MEMS Sensors As Physical Unclonable Functions

Dissertation
zur Erlangung des Grades
des Doktors der Ingenieurwissenschaften
der Naturwissenschaftlich-Technischen Fakultät
der Universität des Saarlandes

von

Oliver Willers

Saarbrücken

2019

Tag des Kolloquiums:	25.10.2019
Dekan:	Univ.-Prof. Dr. rer. nat. Guido Kickelbick
Berichterstatter:	Prof. Dr. H. Seidel
	Prof. Dr. D. Schröder
Vorsitz:	Prof. Dr.-Ing. G. Frey
Akad. Mitarbeiter:	Dr.-Ing. P. Motzki

To my family and friends.

Abstract

A fundamental requirement of any crypto system is that secret-key material remains securely stored so that it is robust in withstanding attacks including physical tampering. In this context, physical unclonable functions (PUFs) have been proposed to store cryptographic secrets in a particularly secure manner.

In this thesis, the feasibility of using microelectromechanical systems (MEMS) sensors for secure key storage purposes is evaluated for the first time. To this end, we investigated an off-the-shelf 3-axis MEMS gyroscope design and used its properties to derive a unique fingerprint from each sensor.

We thoroughly examined the robustness of the derived fingerprints against temperature variation and aging. We extracted stable keys with nearly full entropy from the fingerprints. The security level of the extracted keys lies in a range between 27 bits and 150 bits depending on the applied test conditions and the used entropy estimation method. Moreover, we provide experimental evidence that the extractable key length is higher in practice when multiple wafers are considered. In addition, it is shown that further improvements could be achieved by using more precise measurement techniques and by optimizing the MEMS design.

The robustness of a MEMS PUF against tampering and malicious read-outs was tested by three different types of physical attacks. We could show that MEMS PUFs provide a high level of protection due to the sensitivity of their characteristics to disassembly.

Kurzfassung

Eine grundlegende Anforderung jedes Kryptosystems ist, dass der verwendete geheime Schlüssel sicher und geschützt aufbewahrt wird. Vor diesem Hintergrund wurden physikalisch unklonbare Funktionen (PUFs) vorgeschlagen, um kryptographische Geheimnisse besonders sicher zu speichern.

In dieser Arbeit wird erstmals die Verwendbarkeit von mikroelektromechanischen Systemen (MEMS) für die sichere Schlüsselspeicherung anhand eines 3-achsigen MEMS Drehratensensor gezeigt. Dabei werden die Eigenschaften der Sensoren zur Ableitung eines eindeutigen Fingerabdrucks verwendet.

Die Temperatur- und Langzeitstabilität der abgeleiteten Fingerabdrücke wurde ausführlich untersucht. Aus den Fingerabdrücken wurden stabile Schlüssel mit einem Sicherheitsniveau zwischen 27 Bit und 150 Bit, abhängig von den Testbedingungen und der verwendeten Entropie-Schätzmethode, extrahiert. Außerdem konnte gezeigt werden, dass die Schlüssellänge ansteigt, je mehr Wafer betrachtet werden. Darüber hinaus wurde die Verwendung einer präziseren Messtechnik und eine Optimierung des MEMS-Designs als potentielle Verbesserungsmaßnahmen identifiziert.

Die Robustheit einer MEMS PUF gegen Manipulationen und feindseliges Auslesen durch verschiedene Arten von physikalischen Angriffen wurde untersucht. Es konnte gezeigt werden, dass MEMS PUFs aufgrund der Empfindlichkeit ihrer Eigenschaften hinsichtlich einer Öffnung des Mold-Gehäuses eine hohe Widerstandsfähigkeit gegenüber invasiven Angriffen aufweisen.

Acknowledgements

At this point I would like to thank all those who supported and motivated me during my PhD work. First of all my thanks go to Prof. Dr. Helmut Seidel for supervising my thesis and to Prof. Dr. Helmut Seidel and Prof. Dr. Dominique Schröder for acting as referees for my thesis and for the helpful suggestions and the constructive criticism. Then, I would like to especially thank Dr. Sven Zinober for the excellent guidance and the weekly feedback. I am very thankful to Dr. Jorge Guajardo, Dr. Christopher Huth and Dr. Michael Curcic for the excellent collaboration, the open exchange of ideas, and the fruitful discussions. I would like to thank Dr. Andreas Lassl, Dr. Wenqing Liu, Dr. Reinhard Neul, Thomas Buck, Ingo Herrmann, Ulrich Kunz, Max Schellenberg, Ralf Bößendörfer, Dr. Christoph Schelling, Dr. Tobias Zoller, Sabine Nagel, Dr. Johannes Kenntner, and Dr. Jain Shalabh for sharing their expertise and for always being willing to provide support. Additionally, I would like to thank all colleagues in the Microsystems and Nanotechnologies department at Robert Bosch Corporate Research in Renningen. The constructive and inspiring working environment has also contributed to the success of this thesis. Furthermore, I would like to thank Dr. Roland Müller-Fiedler, Dr. Andre Kretschmann, Dr. Hans Georg Schlager, and Dr. Kilian Bilger for enabling me to perform this work and the ongoing support. I would also like to thank Marius Wolf, Peter Deutsch, and Swarup Nagnath Pulujkar for their diligence and ideas they put into during their internships and theses. Finally, I would like to take this opportunity to thank my family and friends, who are always there for me, even if things are not going perfectly.

Table of Contents

Chapter 1

Introduction

It is well known and widely accepted that one of the biggest challenges in the Internet of Things (IoT) is security. A fundamental assumption in any crypto system is that secret-key material remains securely stored and protected from attackers. The secrecy of cryptographic keys is a fundamental prerequisite to safeguard many higher level mechanisms such as attestation, secure boot and any cryptographic operation which might require a secret or private key (e.g., encryption, signatures, message authentication codes, etc.).

However, guaranteeing security in the IoT is challenging, because the devices are often constrained in computational, memory, and power resources. In this context, secure key storage has been recognized as a major issue by many works [1–5]. This is even more critical in cases where a key cannot be stored locally in digital form because expensive non-volatile memory (NVM) and continuous power supply for tamper-evident memory is unavailable. To this end, physical unclonable functions (PUFs) have been identified as a promising alternative to secure NVM because of their assumed higher security, tamper evidence properties [1–5], and their ease of integration with other hardware security primitives and architectures [6–8].

A PUF is a physical system whose uniqueness and randomness properties are typically due to manufacturing tolerances. When stimulating the PUF with a challenge a unique response can be measured, forming a challenge-response-pair (CRP). This property can be used, e.g., for low cost device authentication [9]. When processing the response further, a cryptographic key can be derived from a PUF. From a security perspective, this is beneficial because the key is only derived when it is needed and thus, it is only available for use for a very short period of time. As a result, an attacker will have less opportunities to recover the key during an invasive attack when compared to a similar key stored in NVM. The susceptibility of NVM to invasive attacks has been demonstrated multiple times [10–12]. PUFs also offer the unclonability property which, com-

bined with a specialized key derivation process, can position them as a trust anchor on top of which a secure environment and secure applications can be built.

The focus of this work is on investigating the feasibility of constructing PUFs based on the uniqueness of microelectromechanical systems (MEMS) sensors. In this approach, the properties (features) of MEMS devices are used in order to create unique bit strings which can be processed as cryptographic keys. This PUF construction is even more relevant when one considers that sensors are already ubiquitous in the IoT world. Since these sensors often provide safety-relevant information (e.g., data for electronic stability control (ESC) and airbag modules) or collect highly private and sensitive data (e.g., sensitive data gathered from smartphones, wearables or body sensors), it is of great importance that the authenticity and confidentiality of the data that they collect cannot be compromised. Moreover, MEMS sensors are typically covered by a mold package preventing physical access to the PUF device. Since MEMS' exact characteristics are also dependent on the packaging processes (e.g., induced thermal stress), their properties are sensitive to disassembly which makes MEMS PUFs promising in terms of the resistance to invasive attacks.

1.1 Research Goals

The main goal of this thesis is to evaluate the feasibility of creating a PUF based on MEMS sensors and, in particular, an off-the-shelf 3-axis MEMS-based gyroscope is the device under test (DUT). The following sub-goals are addressed:

- identifying possible implementation concepts,
- determining MEMS gyroscopes' properties that are suitable to be used for cryptographic key derivation,
- developing an electrical characterization method to be able to measure these properties in a fast and automated manner,
- deriving the MEMS' fingerprints with a suitable quantization procedure which converts the analog feature values into uniformly distributed substrings,
- investigating the uniqueness and robustness of the derived fingerprints against varying temperature conditions, aging, and packaging effects,

- examining the impact of within-batch and batch-to-batch variations on the uniqueness and length of the fingerprints,
- determining the number of stable bits with nearly full entropy that can be derived by estimating the entropy of the fingerprints and the entropy loss due to information reconciliation and randomness extraction,
- assessing the security level of the derived keys with regard to usability in real applications,
- identifying potentials for improvement in terms of the number of derivable bits considering measurement technique and MEMS design,
- analyzing the resistance of MEMS PUFs to different kind of attacks.

1.2 Thesis Outline

This thesis is structured in the following manner. We begin by introducing the state of the art in Chapter 2. In doing so, we give an overview of PUFs, introduce entropy and its estimation, and explain the basics of the key derivation process. Afterwards, we provide fundamentals of MEMS gyroscopes, explain the investigated sensor design, and discuss the characteristics of a MEMS PUF including an overview about related work. In Chapter 3, we explain the used measurement technique, the setup of measured sensor modules, and the performed test procedures. In Chapter 4, the conversion of the measurements values into binary strings is explained and the uniqueness of the obtained fingerprints is analyzed. Additionally, we show the effect of considering sensors from several wafers for the uniqueness of the fingerprints. Based on the derived fingerprints, cryptographic keys are extracted and their robustness and randomness is evaluated in Chapter 5. In Chapter 6, we showcase potentials of improvement regarding the amount of extractable information from MEMS structures by an improved measurement technique and by design optimization. The resistance of a MEMS PUF against physical attacks is analyzed in Chapter 7. The thesis is concluded in Chapter 8.

Chapter 2

Background and State of the Art

In this chapter, basic foundational concepts about PUFs are described. In particular, several state-of-the-art PUF constructions and classifications are introduced. Then, different methods for the evaluation of PUF responses and needed postprocessing steps are discussed. Afterwards, the basics of MEMS gyroscopes are explained and the investigated sensor design is described. Finally, MEMS PUFs are introduced including an overview of related work and a discussion of different implementation concepts.

2.1 Physical Unclonable Functions

PUFs were introduced by Pappu *et al.* at the beginning of this century as physical one-way functions [13, 14]. A PUF exploits variations inherent in the manufacturing process of a physical system. Since these manufacturing variations cannot be controlled even by the device manufacturers themselves, it is infeasible to estimate the exact characteristics of a particular device or to reproduce them, thus making the device unclonable. Moreover, PUFs can provide tamper resistance enabling secure key storage in a device.

To derive a device's unique response (also called fingerprint), a stimulus or a challenge has to be applied to the PUF. The corresponding PUF output is called a response forming a CRP. Dependent on the PUF type the nature of the stimulus can be quite different. For example, an arbiter PUF requires a binary input string determining the evaluated paths of a circuit [1] while an static random access memory (SRAM) PUF requires the addresses of used memory cells [15]. In case of a MEMS PUF, the challenge consists of an electrical signal, e.g., with a dedicated voltage and frequency. The response of a PUF is a binary string. For some PUF types, e.g. arbiter and SRAM PUFs, the output is already binary. Other PUF types, e.g. coating [16] and MEMS

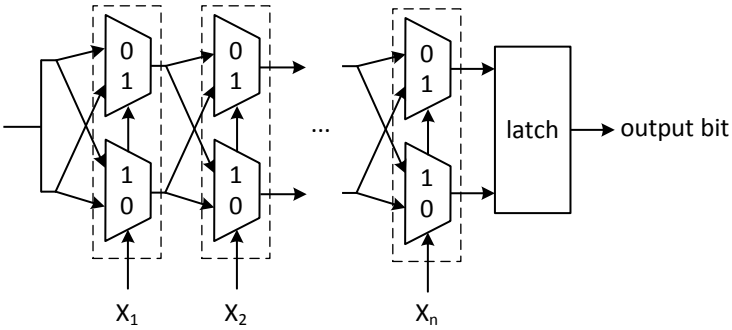


Figure 2.1: Working principle of an arbiter PUF [3].

PUFs, require a postprocessing step in order to quantize analog feature values in an appropriate manner.

2.1.1 Major PUF Constructions

PUFs have received a lot of attention in the past 15 years. In this section, several PUF constructions which have been proposed in the literature are briefly described. Note that further PUF types exist. A comprehensive overview about different PUF constructions proposed in the last years is given in [17, 18].

Optical PUF. In [13], Pappu proposed different methods to implement an optical PUF. For example, transparent structures doped with scattering particles were used. By shining the structures with a laser, a unique speckle pattern is created. However, it is clear that such an optical PUF is hard to integrate into products like sensor nodes.

Arbiter PUF. Arbiter PUFs exploit the delay differences between identically designed paths of integrated circuits [1–3, 19]. The input to an arbiter PUF is an n-bit vector X which defines the used delay paths by controlling multiplexers. In Figure 2.1, each input bit controls two multiplexers. If a bit is zero, the signals are forwarded to the next multiplexers, otherwise the signals are switched to the multiplexer in the opposite path. The PUF is evaluated by applying a signal to the two paths at the same time and the arbiter (latch) measures which path is faster. The generated output is either a one or a zero bit. The input vector X should be chosen randomly. Hence, arbiter PUFs have a large input space (2^n). In order to create a k-bit output, the procedure has to be repeated k times with random input vectors. Alternatively, k single-output circuits can also be used.

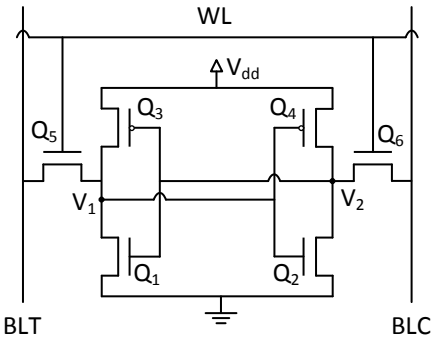


Figure 2.2: Circuit of a six transistor SRAM cell [20].

Ring Oscillator PUF. The working principle of ring oscillator (RO) PUFs is similar to that of arbiter PUFs. This PUF type compares frequencies of ROs which are built by identically designed delay loops [3]. A binary input string determines the evaluated ROs. A RO PUF can also be implemented in an integrated circuit.

Coating PUF. Coating PUFs consist of capacitors, which are covered with a coating doped with dielectric particles (e.g. TiO_2) [16]. The capacitance varies for each device since the particles are randomly distributed. Coating PUFs can provide inherent tamper resistance since an invasive attack would change the capacitance value of the capacitor. However, additional manufacturing steps are necessary.

SRAM PUF. SRAM PUFs were introduced in [4, 15]. An SRAM cell (Figure 2.2) is manufactured in complementary metal–oxide–semiconductor (CMOS) fabrication technology and it consists of six transistors (two cross-coupled inverters $Q_1 - Q_6$ and two access transistors Q_5 and Q_6) [20]. The cross-coupled inverters are connected through the two access transistors to data bit-lines BLT (bit-line true) and BLC (bit-line complement). The access transistors can be controlled via the word-line (WL) signal. Powering-up an SRAM cell, many of the bi-stable memory cells tend with high probability to either one or zero as a result of slight differences in the transistors' threshold voltages [4]. Advantageously, SRAM PUFs can deliver a large number of bits, with the size of an SRAM array as the only limit, and SRAM is inherent in many semiconductor devices. Hence, it does not require additional steps in the manufacturing process.

2.1.2 Definition of PUFs

In [21], the mapping from a challenge to a response performed by the PUF is denoted as a probabilistic function $f : \mathcal{C} \to \mathcal{R}$, where $\mathcal{C}$ is a domain space and $\mathcal{R}$ is an output range of f. The randomized creation process of a new PUF is formally expressed by invoking a manufacturing process $\mathcal{MP}$ which is initialized by parameters *param*. The following definitions provided by Armknecht *et al.* [21] are parametrized by some thresholds δ_i, the number of iterations t, the number of inputs ℓ, the number of devices n, a negligible function $\epsilon(\cdot)$, and the security parameter λ.

Intra Distance Requirement [21]: *Whenever a single PUF is repeatedly evaluated with a fixed input, the maximum distance between the corresponding outputs is at most δ_1. That is for any created PUF $f \leftarrow \mathcal{MP}(param)$ and any $y \in \mathcal{C}$ it holds that*

$$\mathsf{Pr}\left[\max(\{\mathsf{dis}(z_i, z_j)\}_{i \neq j}) \leq \delta_1 \,|\, y \in \mathcal{C}, \{z_i \leftarrow f(y)\}_{1 \leq i \leq t}\right] = 1 - \epsilon(\lambda).$$

Inter Distance I Requirement [21]: *Whenever a single PUF is evaluated on different inputs, the minimum distance among them is at least δ_2. That is for a created PUF $f \leftarrow \mathcal{MP}(param)$ and for any $y_1, ..., y_\ell \in \mathcal{C}$, we have*

$$\mathsf{Pr}\left[\min(\{\mathsf{dis}(z_i, z_j)\}_{i \neq j}) \geq \delta_2 \,\middle|\, \begin{array}{l} y_1, ..., y_\ell \in \mathcal{C}, \\ \{z_i \leftarrow f(y_i)\}_{1 \leq i \leq \ell} \end{array}\right] = 1 - \epsilon(\lambda).$$

Inter Distance II Requirement [21]: *Whenever multiple PUFs are evaluated on a single, fixed input, the minimum distance among them is at least δ_3. That is for any created PUF $f_i \leftarrow \mathcal{MP}(param)$ for $1 \leq i \leq n$ and any $y \in \mathcal{C}$, we have*

$$\mathsf{Pr}\left[\min(\{\mathsf{dis}(z_i, z_j)\}_{i \neq j}) \geq \delta_3 \,|\, y \in \mathcal{C}, \{z_i \leftarrow f_i(y)\}_{1 \leq i \leq n}\right] = 1 - \epsilon(\lambda).$$

Min-Entropy Requirement [21]: *Whenever multiple PUFs are evaluated on multiple inputs, the min-entropy of the outputs is at least δ_4, even if the other outputs are observed. Let $z_{i,j} \leftarrow f_i(y_i)$ be the output of a PUF f_i on input y_i where $f_i \leftarrow \mathcal{MP}(param)$. Then*

$$\mathsf{Pr}\left[\tilde{H}_\infty(z_{i,j} \,|\, \mathcal{Z}_{i,j}) \geq \delta_4 \,\middle|\, \mathcal{Z} := \begin{array}{l} y_1, ..., y_\ell \in \mathcal{C}, \\ \{z_{i,j} \leftarrow f_i(y_i)\}_{1 \leq i \leq n, 1 \leq j \leq \ell}, \\ \mathcal{Z}_{i,j} := \mathcal{Z} \backslash \{z_{i,j}\} \end{array}\right] = 1 - \epsilon(\lambda)$$

holds for sufficiently large δ_4.

Definition I [21]: *A PUF $f : \mathcal{C} \to \mathcal{R}$ has $(\mathcal{MP}, t, n, \ell, \delta_1, \delta_2, \delta_3, \epsilon)$-variance if the PUF's output has inter and intra distances as described above, parameterized by $(\mathcal{MP}, t, n, \ell, \delta_1, \delta_2, \delta_3)$.*

Definition II [21]: *A PUF* $f : \mathcal{C} \rightarrow \mathcal{R}$ *has* $(\mathcal{MP}, n, \ell, \delta_4, \epsilon)$*-min-entropy if the PUF satisfies the min-entropy requirement explained above.*

A prerequisite for a PUF is that its output is unique. This requires that if a PUF has multiple inputs, threshold $\delta_1 < \delta_2$ so that it can be distinguished between noisy responses obtained from the same input and responses obtained from different inputs. Furthermore, $\delta_1 < \delta_3$ is required in order to be able o distinguish between noisy responses obtained from the same input and responses obtained from other PUFs.

The min-entropy requirement implies the unpredictability property of PUFs. It states that even if an attacker could evaluate multiple PUFs on multiple inputs, the PUF output on any new chosen challenge remains unpredictable. Another fundamental property of PUFs is unclonability. This means that it should be infeasible to predefine the exact characteristics or to physically replicate a single PUF even for the manufacturer. The unpredictability and unclonability properties can be formally expressed as in [21].

2.1.3 Types of PUFs

Depending on the number of CRPs, PUFs have been divided into two categories: strong PUFs and weak PUFs (also called obfuscating PUFs [22]). This differentiation was originally introduced in [4] and further developed in [22, 23].

Strong vs. weak PUFs. Strong PUFs are characterized by having a large number of CRPs meaning that the domain space $\mathcal{C}$ and the output range $\mathcal{R}$ are large. As the minimum amount of CRPs a strong PUF needs to have, Guajardo *et al.* proposed 2^{100} [4]. In [24], Rührmair *et al.* describe this amount qualitatively as it should be infeasible to completely measure all CRPs in a limited amount of time (e.g., several days or even weeks) noting that PUFs have a limited read-out speed.

A crucial condition for a strong PUF is that it must be infeasible for an attacker to predict the right response for any new chosen challenge even if the attacker could acquire a considerable number of CRPs before the prediction event[1]. Therefore, the different CRPs have to be independent from each other so that they do not reveal any relevant information about each other according to the min-entropy requirement.

A big advantage of having a strong PUF is that a response can be transmitted without any additional security mechanism because it is assumed that

[1] In order to prevent an attacker to apply arbitrary challenges to the PUF, so-called controlled PUFs were proposed adding a control logic which surrounds the PUF [25].

each CRP will only be used once, e.g., enabling secure authentication. This is especially beneficial in cases where no computational power is available on the device for processing the response and performing cryptographic operations, such as on radio-frequency identification (RFID) tags [9]. In this example, the PUF is randomly read out multiple times in an enrollment stage generating a list of valid CRPs. Based on this list, the device can be authenticated in the field by checking the response to a corresponding challenge.

A promising candidate for building a strong PUF in semiconductor devices was the class of arbiter PUFs due to its complex challenge-response behavior and the large number of possible challenges. However, meeting the requirement of unpredictability is a difficult undertaking since different outputs are usually not fully independent. Hence, multiple attacks have been shown to be feasible by taking advantage of few CRPs as input to powerful machine learning techniques [24, 26–28].

Weak PUFs have only few CRPs or, in some cases, just one[2]. Hence, the security of the PUF cannot be build on the single use of CRPs, but the PUF's output needs to be protected against unauthorized access and it must not be given to the outside world. Then, it can be used as as a secret input for subsequent cryptographic operations. Note that a fundamental property of weak PUFs is inherent tamper resistance in order to assure that the PUF response is kept secret. This property can be formally expressed as in [21].

A popular candidate from this PUF class is the SRAM PUF. However, it has been already shown that it is possible to read out SRAM PUFs by invasive and semi-invasive attacks [29]. Additionally, it was shown in [30] that a physical clone of a SRAM PUF can be produced. As discussed later, a MEMS PUF has to be classified as a weak PUF since it only accepts a limited number of challenges.

Intrinsic vs. extrinsic PUFs. Another classification initially proposed by Guajardo *et al.* [4] divides PUFs into intrinsic and extrinsic PUFs. This classification is based on whether the exploited randomness is inherent in the device's standard manufacturing process (intrinsic) or added in an extra production step (extrinsic). Intrinsic PUFs are, e.g., arbiter, SRAM and MEMS PUFs. Examples for extrinsic PUFs are optical PUFs and coating PUFs. Notice that the evaluation unit which measures the PUF is expected to be embedded into the device in case of intrinsic PUFs.

[2] In the case that there is just one CRP, the inter distance I requirement becomes meaningless since $\ell = 1$.

2.2 MEMS Physical Unclonable Functions

A MEMS PUF is based on the uniqueness of MEMS' characteristics. In contrast to most other PUFs, MEMS PUFs can be based on a variety of mechanical and electrical properties which depend on the sensor type. According to Section 2.1.2, the following requirements can be formulated for a MEMS PUF:

- *Unique.* The PUF response has to be unique per device.
- *Unpredictable.* The PUF response of any device should be unpredictable even if an attacker could measure multiple PUFs before.
- *Uncontrollable.* It should be infeasible to predefine the exact characteristics or to replicate a single PUF device even for the manufacturer.
- *Tamper-resistant.* The PUF should have tamper resistance or tamper evidence properties so that it is infeasible for an attacker to measure the PUF response without changing it.

Additionally, we emphasize that $\delta_1 < \delta_3$ needs to hold across the whole range of environmental conditions for which the device is designed and over its life-time.

Due to their high complexity, the large number and the different nature of characteristics, MEMS PUFs are very promising in terms of uncontrollability so that it should be infeasible to build a physical clone as it was done for SRAM PUFs. Since their exact property values are also dependent on packaging processes (e.g., molding, vacuum bonding, etc.), MEMS PUFs can be hypothesized to be sensitive to invasive attacks providing inherent tamper resistance.

Regarding the concept of CRPs, we can define the measurement of each feature as an individual CRP. However, it is hardly feasible to extract enough information from a single feature to derive a key of reasonable length. Hence, we prefer to define the measurement procedure of all features used as a single CRP. In this case, the number of possible inputs to the PUF $\ell = 1$ so that we only consider inter distance II requirement from Section 2.1.2 hereinafter. Note that even if a MEMS might have enough randomness to derive several independent responses of considerable length from it, it seems to be difficult to meet the very challenging requirement of strong PUFs. Thus, MEMS PUFs are rather candidates for the class of weak PUFs and they clearly belong to the class of intrinsic PUFs.

2.2.1 Related Work

Recently, several studies have investigated the feasibility of deriving fingerprints from MEMS sensors. In particular, most of them are focused on fingerprinting mobile devices, such as smartphones, either with the goal of using the sensor's fingerprints for secure device authentication, e.g., [31–37] or in order to raise awareness about the privacy implications of device tracking [38, 39]. In this context, a variety of MEMS devices have been recently analyzed including accelerometers [31–34, 37–39], microphones [32, 37], gyroscopes [37, 39], magnetometers [36, 37] or a combination of them.

While the principle feasibility of deriving fingerprints from MEMS sensors has been shown several times, the stability of the fingerprints regarding the whole temperature range typically required for consumer sensors (from −40°C to 85°C) or aging was not sufficiently investigated in any of the studies. Moreover, no detailed analysis about the extractable entropy has been performed.

In addition, the proposed procedures have a common limitation: the way the response is measured is not practical for a stand-alone secure storage solution. In particular, it is required that either an external stimulus, e.g. vibration [31, 33, 36–38], sinusoidal tones [32, 37, 39], is provided, a user executes an external action [34, 39] or the device rests in a certain fixed position [32, 34]. The only exception is the proposal of Aysu *et al.* to use the self-test function of accelerometers in which an electrostatic stimulus is generated by the application-specific integrated circuit (ASIC). However, there is not enough evidence to show that sufficient information could be gathered to extract a stable key with a reasonable security level.

2.2.2 MEMS PUF Model

In this work, we are aiming at an embedded solution, independent of external conditions. Thus, we prefer to extract the fingerprint information directly from the MEMS structure by measuring its properties. We assume a MEMS to be placed together with an ASIC and (possibly) with a microcontroller (µC) on a substrate covered by a mold package[3]. Figure 2.3 shows schematically a typical example for such a system in package (SIP). MEMS and ASIC are connected by wire bonds and placed on a land grid array (LGA) substrate. Alternatively, MEMS and ASIC could also be stacked vertically and (possibly) connected by through-silicon vias [41].

[3]Note that smart hubs exist which combine sensors with a µC in a SIP (see e.g., [40])

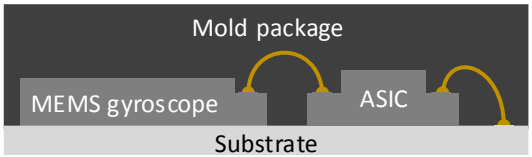


Figure 2.3: Schematic composite of MEMS sensor and ASIC in a SIP.

We assume that all operation required to derive a key from a MEMS and the desired subsequent cryptographic operations are performed within the package (see Figure 2.4) so that neither the PUF response nor the derived secret (private) key is given to the outside world. This can be accomplished by using digital signatures enabling data authenticity and integrity. But also confidentiality can be enabled by public-key based protocols without disclosing the secret (private) key (see e.g., [42]). Furthermore, we assume that digital stored information such as the stimulating signals, the quantization scheme and the helper data needed for error correction, are known to an attacker since he could extract those information by a physical attack.

2.2.3 Implementation Concepts

Adding secure key storage capabilities to existent sensors would provide an additional value, making them *enhanced* sensors. In the longer run, new MEMS concepts could be exclusively designed for secure key storage purposes (*dedicated* MEMS PUF). This would also offer opportunities to increase the number of derivable bits by measures in the design or in the manufacturing process. In total, we see three possible implementation concepts:

- *Dual use.* Using the same MEMS structure for sensing and key storage purposes.
- *Additional structure.* Using an additional MEMS structure (e.g., a separate and decoupled part of the MEMS sensor) for secure key storage.
- *Stand alone.* A new MEMS device for secure key storage only as a product for the security market without sensor functionality.

In the dual use concept, the same MEMS structure is used for sensing and key storage. In this case, no additional MEMS structure would be necessary. However, such a dual use comes up with additional risks as it is shown later in this thesis. In order to overcome the (possible) issue of information leakage by

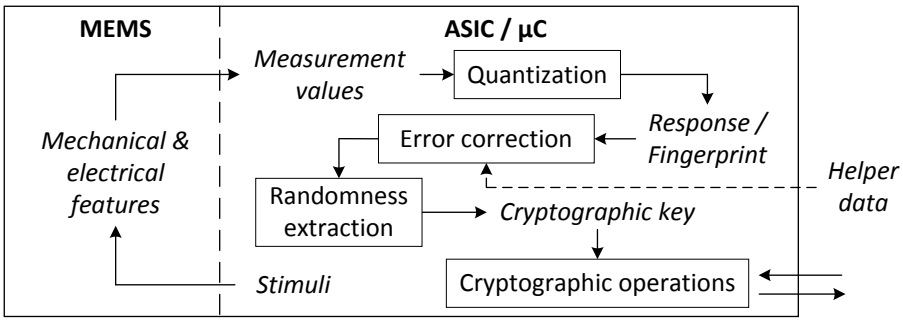


Figure 2.4: Flow chart for deriving a cryptographic key from a MEMS PUF.

the sensor's output, a separate structure could be added to the MEMS which is exclusively used for key derivation. This structure could be a separated and decoupled part within the original sensor. Note that such a structure can be much smaller than the sensing structure as shown later. The third proposal aims at a product of its own (e.g., a security-token) for the security market based on an optimized MEMS structure providing secure key storage capabilities. Note that in the latter two implementation concepts a multi-core approach with multiple decoupled MEMS structures would be possible in order to increase the number of derivable bits.

2.3 Entropy and Its Estimation

In an ideal case, bits derived from PUFs would be independent and identically distributed (IID). However, as a result of bias and/or correlations, PUF responses are usually expected to have reduced entropy meaning that some responses have a higher probability than others. Since a cryptographic key needs to have *nearly* full entropy, an appropriate postprocessing step (randomness extraction, see Section 2.4.3) is necessary if a PUF is used for generating a key. In order to assess the security level of the extracted keys, the entropy of the PUF responses has to be properly estimated.

We consider two possible definitions of entropy: Shannon entropy and min-entropy as discussed in [43]. The concept of (Shannon) entropy was introduced by Shannon in [44]. It has been observed in [45] that Shannon entropy provides a lower bound for the *average* work incurred in guessing a random variable[4].

[4]This observation is attributed in [45] to Massey [46].

The Shannon entropy or just the entropy of a discrete random variable X, is defined as

$$H(X) := -\sum_{x \in \mathcal{X}} \mathsf{Pr}(X = x) \log_2 \mathsf{Pr}(X = x), \tag{2.1}$$

where $\mathcal{X}$ denotes the range of the variable X and $\mathsf{Pr}(X = x)$ is the probability of a possible outcome x of the variable X.

To understand, the concept of min-entropy, it is convenient to think in terms of an adversary trying to guess a secret key used by a cryptographic algorithm. Clearly, the optimal adversary's strategy is to guess the *most* likely key. Thus, in cryptographic applications it is desirable to choose a random key from a uniformly distributed distribution. In particular, if a key is chosen uniformly at random, the adversary's advantage is minimized and his best guess is the inverse over the number of possible keys (e.g., if one chooses a secret key uniformly in $[0, 2^\ell)$, the probability of guessing *any* key is $2^{-\ell}$). Following Dodis *et al.* [43], the min-entropy of a random variable X is defined as

$$H_\infty(X) := -\log_2 \left(\max_{x \in \mathcal{X}} \mathsf{Pr}(X = x) \right), \tag{2.2}$$

which means that the value of $H_\infty(X)$ depends only on the most likely outcome of X.

Following the example above, then the min-entropy of the uniform distribution in $[0, 2^\ell)$ (sometimes written U_ℓ) is ℓ. In other words, we expect that we can extract nearly ℓ uniformly distributed bits from the uniform distribution in the range $[0, 2^\ell)$. It has to be noticed that in practice the min-entropy measure might be too strict as it presupposes that an adversary would know the distribution from which the key originates and, thus, he would know the *most* likely key. This problem has been observed by others in the context of PUFs and biometrics, see e.g., [47–51]. Note that there are distributions that can have a lot of Shannon entropy but just a couple of bits of min-entropy [52, Section 2.2].

The concept of being *nearly* uniform can be formalized via the statistical distance SD between two probability distributions X and Y, which is defined as $\mathrm{SD} := \frac{1}{2} \sum_v |\mathsf{Pr}(X = v) - \mathsf{Pr}(Y = v)|$. If two probability distributions are close to each other, then the statistical distance between them is expected to be small. The statistical distance is useful to define the security of a given key generation process in terms of how close the distribution of output keys (from the key generation process) is to the uniform distribution.

2.3.1 Hamming Distance Measure

The Hamming distance evaluates the distance between binary strings. In particular, given two bit strings w and u of the same length n, $\mathsf{HD}(w, u)$ is the number of positions in which w and u differ. Thus, the Hamming distance is defined as

$$\mathsf{HD}(w, u) = \sum_{i=1}^{n} w_i \oplus u_i, \tag{2.3}$$

and the fractional Hamming distance can be calculated by

$$\text{fractional } \mathsf{HD}(w, u) = \frac{1}{n} \sum_{i=1}^{n} w_i \oplus u_i. \tag{2.4}$$

Note that, the Hamming distance measure does not provide directly an entropy estimate. However, it enables to evaluate the basic suitability of a physical system to be used in PUF applications through the concept of inter and intra distances[5] discussed in Section 2.1.2 [3]. As pointed out in Section 2.2, we only consider inter distance II hereinafter.

In the following, we denote the distance between responses from different PUFs to the same input as inter Hamming distance $\mathrm{HD}_{\mathrm{inter}}$. When comparing PUF responses, each bit that is compared can be seen as an own random experiment, with the two possible outcomes that bits are equal or different. Hence, if compared PUF responses would be perfectly random, the probability that compared bits are equal (or different) would be 0.5 and the $\mathrm{HD}_{\mathrm{inter}}$ distribution would follow a binomial distribution with $p = 0.5$ (see Figure 2.5).

The intra Hamming distance $\mathrm{HD}_{\mathrm{intra}}$ evaluates the distance between responses from the same PUF to the same input, determining the number of bit-flips as a consequence of measurement noise, aging or temperature drift. Ideally, all values within $\mathrm{HD}_{\mathrm{intra}}$ distribution would be 0.

Based on the $\mathrm{HD}_{\mathrm{intra}}$ and $\mathrm{HD}_{\mathrm{inter}}$ distributions, the false rejection rate (FRR) and false acceptance rate (FAR) can be determined which are common quality measures, well known from biometrics. The FRR denotes the probability that two measurements w and w' from the same instance cannot be matched since their Hamming distance is larger than a threshold t ($\mathsf{HD}(w, w') > t$). The FAR is the probability that measurements from different instances w and u are falsely assumed to originate from the same instance because their Hamming distance is smaller or equal to the threshold t ($\mathsf{HD}(w, u) \leq t$). In order that each PUF instance can be uniquely identified with low error probability, $\mathrm{HD}_{\mathrm{intra}}$ and $\mathrm{HD}_{\mathrm{inter}}$ distributions should overlap only with negligible probability.

[5] Also known as Authentics and Imposters in biometrics [53].

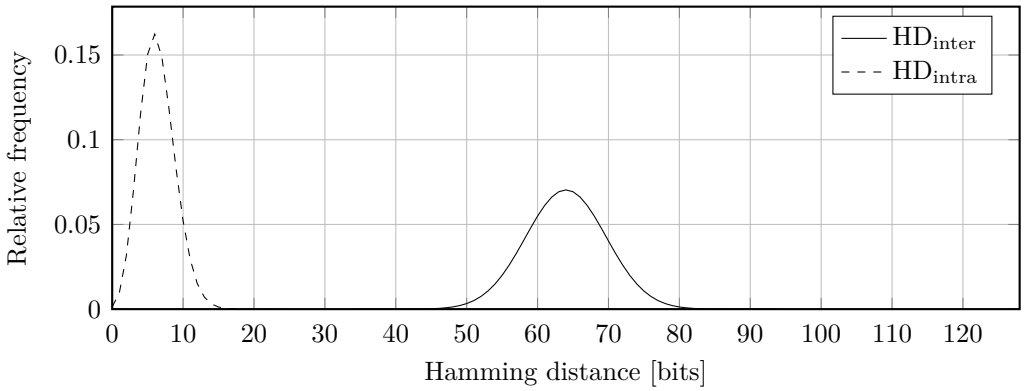


Figure 2.5: Schematic illustration of HD_{intra} and ideal HD_{inter} distribution with response length $n = 128$.

2.3.2 Daugman Method

Estimating the information contained in a binary string is also fundamental in the field of biometrics. In [53], John Daugman proposed a method to estimate the amount of information contained in binary strings derived from iris patterns. This method is based on determining an equivalent binomial distribution to a measured HD_{inter} distribution. The effective number of trials n of the equivalent binomial distribution is then taken as entropy assessment. However, this method requires a constant probability p for the bits to be zero or one. Hence, this is hardly applicable in cases where p-values are not equally distributed, e.g., as a result of spatial correlations or bias, which is the case for MEMS PUFs (see Section 4.4).

2.3.3 Hamming Weight

The Hamming weight (HW) counts the number of bits which are different from zero. Ideally, the HW over all PUF responses is 0.5. This method can be used in order to evaluate a bias in the PUF responses which is a common issue, e.g., of SRAM PUFs (see e.g. [50]). As a result of this analysis the min-entropy can be calculated using Equation (2.2). This evaluation method can be extended by determining the probability of occurrence of longer bit blocks, e.g., entire bytes [54]. In the following, this evaluation method is called most common byte (MCB). It has to be noted that the influence of correlations between bits is not considered by HW. In case of the MCB method, only correlations between adjacent bits can be considered. However, since these estimation methods are

often used in literature [50,54–56], HW and MCB methods are also used in this work.

2.3.4 CTW Compression

The context tree weighting (CTW) method is a lossless compression algorithm which is optimal for stationary ergodic sources [57, 58]. This method dynamically builds a context tree during the encoding process by which the probability of the next symbol is estimated in every node along the tree structure. In the used implementation, two different estimators can be chosen: the Krichevski-Trofimov and the zero-redundancy estimator [59].

CTW compression is often used in the context of PUFs to estimate the entropy of the PUF responses. To this end, all measured PUF responses are concatenated and compressed. Then, the resulting compression rate is used as entropy estimate. This approach is based on the fact that the compressibility of a dataset is strongly related to its entropy [60]. The compressibility of PUF responses using CTW has been shown, e.g., in [61].

This method provides an estimate of the upper bound of the entropy of the PUF responses [62]. Since it is often used in the context of PUFs, CTW compression is considered in this work.

2.3.5 NIST special publication 800-90B

The National Institute of Standards and Technology (NIST) special publication 800-90B provides tests in order to check the IID assumption on random numbers [63]. Additionally, it provides the (probably) most conservative tests for estimating min-entropy of non-IID data. Since PUF data is expected to be non-IID, min-entropy estimation tests are used in this work.

In total, the test suite contains a set of ten diverse and conservative statistical entropy tests from which the minimum estimate is taken as min-entropy assessment. The resulting min-entropy estimate provides a lower bound on min-entropy. In the following, the used tests are briefly described. A comprehensive description is given in [63].

- *Most common value estimate.* The most common value is determined and a confidence interval is calculated for its proportion. The min-entropy is derived from the upper bound of the confidence interval. Note that this test is similar to the HW measure (Section 2.3.3).

- *Collision estimate.* The distance between repeated values within an input string is determined trying to guess the most likely output. This test is sensitive to input strings suffering from bias.

- *Markov estimate.* The Markov estimate determines the dependency of a sample on previous samples. Hence, it efficiently recognizes dependencies within a given input string.

- *Compression estimate.* This test is based on the Maurer Universal Statistic [64]. The min-entropy is estimated based on the repetition intervals of symbols which are related to the compressibility of a data set.

- *t-tuple estimate.* The frequencies of all t-tuples (i.e. pairs, triples, etc.) in a given bit string are determined and the estimated probability that the most common of each t-tuple would be present in random data for every value of t is computed. The t-tuple estimate is particularly sensitive to adjacent tuples in an input string.

- *Longest repeated substring (LRS) estimate.* The frequency of tuples is evaluated. This test considers tuples of larger size than the t-tuple estimate. It is sensitive to correlated tuples as well.

- *Multi most common in window (MCW) prediction estimate.* Several subpredictors try to guess the next output based on a window of previous outputs. The number of correct guesses is evaluated so that the best subpredictor can be dynamically selected to predict the next output.

- *Lag prediction estimate.* This test uses also several subpredictors which guess the next output based on a specified lag.

- *Multiple Markov models with counting (MMC) prediction estimate.* This predictor combines the approach of subpredictors and Markov models in order to make a prediction.

- *LZ78Y prediction estimate* This predictor uses a dictionary which is generated based on LZ78 encoding with the Bernstein's Yabba scheme [65].

It has to be noted that the tests were designed for noise sources having fixed-length bit strings. If several noise sources are used, it is assumed that they are independent. Both requirements cannot be fully met from MEMS PUFs as

Table 2.1: NIST key size recommendation for the cryptographic techniques integer-factorization cryptography (IFC), finite-field cryptography (FFC), elliptic-curve cryptography (ECC) in order to achieve a minimum strength of 128 bits [67].

Date	Symmetric	IFC	FFC		ECC
			private	public	
2016 - 2030 & beyond	AES-128	3072	256	3072	256

discussed later. However, to the best of our knowledge, using tests of NIST 800-90B is currently the most conservative method to estimate the min-entropy of PUF responses.

The test suite explicitly aims at minimizing the probability that the entropy of an input string is greatly overestimated [63]. Hence, it provides a conservative lower bound on min-entropy. This becomes clear if one runs the tests on true random files provided by NIST [66]. For a truly random file, the expected min-entropy result is 1.0 per bit. However, for the file *truerand_1bit.bin* the estimated min-entropy is 0.90 (collision estimate) and for the file *truerand_8bit.bin* it is 0.72 (Markov estimate).

2.3.6 Recommended Key Sizes

Recommendations for key sizes needed to achieve a sufficient security level for the different cryptographic techniques are provided by several academic and private organizations. In Table 2.1, the recommendations from NIST are summarized for achieving a security level of 128 bits which is assumed to be sufficient in the long term (beyond 2030) [67]. Note that the given values (and methods) mean minimal sizes for security.

For symmetric algorithms (e.g., AES) a key size of at least 128 bits is recommended. Regarding asymmetric techniques, for integer-factorization cryptography (IFC) (e.g., RSA) keys of 3072-bit size should be used while for elliptic-curve cryptography (ECC) 256 bits are sufficient. In case of finite-field cryptography (FFC) (e.g., DSA) the size of the private key should be at least 256 bits and the size of the public key 3072 bits.

Another fundamental cryptographic operation is hashing. A hash function maps data of arbitrary size to an output of fixed size which is also called the hash value of the input data. The special class of cryptographic hash functions additionally provides collision resistance which means that it is computational hard to find two different inputs to the function that generate the same hash value. Thus, it is possible, e.g., to check the integrity of a message based on its hash value. Following the NIST recommendation, the hash functions SHA-256 (also SHA-512/256) and SHA3-256 can be used for digital signatures and hash-only applications while SHA-1 is only sufficient for hash-based message authentication codes (HMACs), key derivation functions, and random number generation [67].

2.4 Key Derivation and Fuzzy Extractors

As a consequence of non-uniformities in manufacturing processes, PUF responses are not expected to be uniformly distributed. Moreover, evaluating a PUF is subjected to measurement uncertainties (e.g., measurement noise) and the exact PUF behavior might change slightly over time, e.g, due to temperature variation and aging effects.

Therefore, a PUF response has to be processed through an error correction (or information reconciliation) stage and a randomness extraction (or privacy amplification) stage. Error correction is required because due to slight changes in the PUF's behavior, bits in the response might flip. Bit-flips have to be corrected so that an identical cryptographic key can always be reconstructed. Randomness extraction is carried out to be able to generate an almost uniformly distributed key from a non-uniform input. The combination of these two steps is known as a Fuzzy Extractor [43, 68]. Note that both steps result in information leakage, as shown in [43, 68].

2.4.1 Error Correcting Codes

Error Correcting Codes are used to detect errors occurred in a data set, e.g., during data transmission, and correct them if possible. For this purpose, a message is mapped onto a longer codeword containing redundant information, e.g. additional error correction bits. As long as the errors occurred in the codeword do not exceed the error correcting capability of the code, the message can be reconstructed.

We recall some basic definitions on codes and metric spaces following the treatment of Dodis *et al.* [43,68]. A metric space is a finite message set $\mathcal{M}$ with a distance function $\mathsf{dis} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}^+ = [0, \infty)$, which satisfies the following properties: $\mathsf{dis}(x, y) = 0$ if and only if $x = y$, symmetry $\mathsf{dis}(x, y) = \mathsf{dis}(y, x)$, and triangle inequality $\mathsf{dis}(x, z) \leq \mathsf{dis}(x, y) + \mathsf{dis}(y, z)$. Throughout this work, the Hamming distance is assumed (see Section 2.3.1).

In this metric space, an error correcting code $\mathcal{C}$ is a subset $\{c_0, ..., c_{K-1}\}$ of K elements of $\mathcal{M}$, where $d > 0$ is the minimum distance of the code such that $\mathsf{HD}(c_i, c_j) \geq d$ with $i \neq j$. Encoding $\mathsf{enc}(i) = c_i$ is a mapping from a message i to a codeword c_i. Notice that the encoding always maps its input to the same codeword, $\mathsf{enc}(i) = \mathsf{enc}(j)$ if $i = j$. On the other hand, decoding is a mapping that attempts to find the unique codeword c_i with message i corresponding to a given $w \in \mathcal{M}$ such that $\mathsf{HD}(w, c_i) \leq t$, if c_i exists. The decoding operation is denoted by $c_i = \mathsf{dec}(w)$. The largest integer t such that the code can successfully correct up to $t > 0$ errors, is called the error correcting distance of the code.

As it is standard in the literature, a $[n, k, d = 2t + 1]$-code can detect up to $(d - 1)$ errors and it can correct up to $t \geq \lfloor (d - 1)/2 \rfloor$ errors, where n is the length of a codeword c and k is the length of a message i. The code rate R of a code $\mathcal{C}$ is given by the ratio k/n. Unless, explicitly mentioned, in this work the binary field $\mathbb{F}_2$ is considered.

2.4.2 Information Reconciliation

Dodis *et al.* introduced so-called *secure sketches* by which an input $w \in \mathcal{M}$ can be reconstructed from w', a noisy version of w [43, 68]. Following their definition, a secure sketch is a pair of randomized procedures, *sketch* (SS) and *recover* (Rec). The correctness property of secure sketches guarantees that $\mathsf{Rec}(w', \mathsf{SS}(w)) = w$ if $\mathsf{HD}(w, w') \leq t$. In this procedure $\mathsf{SS}(w)$ has to be generated once during an enrollment stage and stored as side information (also known as helper data).

One of the best known constructions is the code-offset construction (Figure 2.6). For this construction, $\mathsf{SS}(w)$ determines the distance between an initial PUF response w and a valid codeword c which is chosen at random. The response w can be reconstructed from a noisy version w' if the distance $\mathsf{HD}(w, w')$ is equal or smaller than the error correcting distance of the used code t. The exact procedure works as follows:

$\mathsf{SS}(w) = h$

1. A valid codeword c from a linear $[n, k, t]$-code is chosen randomly.

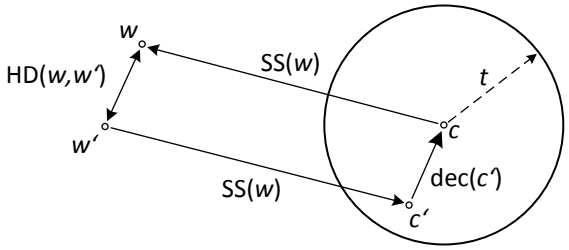


Figure 2.6: Graphical illustration of the code-offset construction by which an initial measurement w can be reconstructed from a noisy version of w, namely w', with the help of so-called helper data $\mathsf{SS}(w)$ which denotes the mapping from w to a valid codeword c. The initial measurement w can be reconstructed as long as the distance $\mathsf{HD}(w, w')$ is equal or smaller than the error correcting capability of the code t.

2. The shift between an initial PUF measurement w and c is calculated by $h = w \oplus c$.

$\mathsf{Rec}(w', h) = w$

1. The noisy codeword c' is calculated by shifting a noisy measurement w' of w using h, $c' = w' \oplus h$.
2. The valid codeword c can be obtained by $\mathsf{dec}(c')$, if $\mathsf{HD}(w, w') \leq t$.
3. By shifting c with h, w can be calculated by $w = c \oplus h$.

The code offset construction can be implemented, e.g., by using a random number generator (RNG) and Bose-Chaudhuri-Hocquenghem (BCH) codes (Figure 2.7). BCH codes are a common choice for error correction with PUFs [3,42], as they are (n, k, t)-codes which guarantee an error-free decoding as long as no more than t bit-flips occur. Such a construction has also been adapted to be secure against (stronger) active attackers, who are allowed to query the PUF multiple times and modify the helper data. This is known as a robust fuzzy extractor [69].

As a result of the stored helper data $\mathsf{SS}(w)$, which are assumed to be public, an entropy loss occurs. Following Dodis *et al.* [43], the residual min-entropy $\tilde{H}_\infty(X|Y)$ of a variable X given a (possibly correlated) variable Y is defined by the concept of average min-entropy as

$$\tilde{H}_\infty(X|Y) := -\log_2 \left(\mathbb{E}_{y \leftarrow Y} \left[\max_{x \in \mathcal{X}} \mathsf{Pr}[X = x|Y = y] \right] \right). \tag{2.5}$$

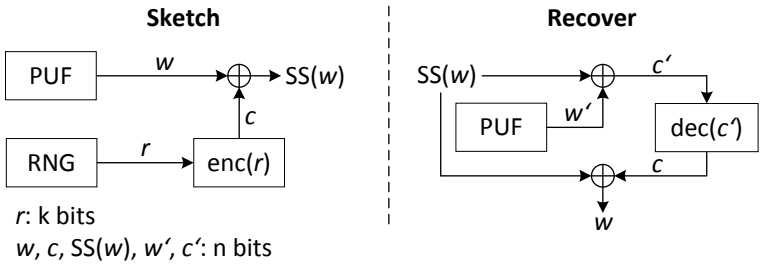


Figure 2.7: Flow diagram of the code offset construction. In an enrollment stage (sketch), helper data $\mathsf{SS}(w)$ are generated denoting the distance between an initial measurement w and a randomly chosen codeword c. By using $\mathsf{SS}(w)$, w can be recovered from a noisy measurement w' if $\mathsf{HD}(w, w') \leq t$, where t is the error correcting capability of the code.

Hence, an average-case $(\mathcal{M}, m, \tilde{m}, t)$-secure sketch can correct up to t errors in any distribution W over $\mathcal{M}$ with min-entropy $H_\infty(W) \geq m$. The security property of the secure sketch guarantees that its output has residual min-entropy $\tilde{H}_\infty(W|\mathsf{SS}(W)) \geq \tilde{m}$, where $m - \tilde{m}$ is the entropy loss of the secure sketch.

In the strict information theoretic sense, the information reconciliation step leaks all the parity bits of the error correction codeword. More specifically, Given an $[n, k, d = 2t + 1]$-code, Dodis *et al.* [43] showed that at most $n - k$ symbols are leaked. However, this is an upper bound on the amount of entropy loss which is maximal for uniformly distributed input strings [70]. Hence, the residual min-entropy $\tilde{m}$ is underestimated by applying the conservative $(n-k)$-bound in cases where the input strings do not have full entropy $m < n$. The possible range in which the residual min-entropy $\tilde{H}_\infty(W|\mathsf{SS}(W))$ lies is bounded by [71]

$$\max(H_\infty(W) - (n - k), 0) \leq \tilde{H}_\infty(W|\mathsf{SS}(W)) \leq \min(k, H_\infty(W)). \tag{2.6}$$

In [71], Delvaux *et al.* presented a method to calculate tighter upper and lower bounds for the entropy loss of PUF responses suffering from correlations or bias.

A worst case and a best case scenario is described for linear codes based on the assumption that is made on the distribution of the most likely PUF responses with regard to cosets. A *coset* of a linear code $\mathcal{C}$ can be constructed by any translation $\tau \in \{0, 1\}^n$, which defines the set $\{\tau \oplus w : w \in \mathcal{M}\}$. Note that two cosets are either identical or disjoint. Each element in a coset has

the same syndrome regarding a linear code $\mathcal{C}$. The coset where the syndrome is zero describes the code $\mathcal{C}$. Consequently, helper data reveal in which coset a particular PUF response w resides. The information loss would be maximal in the case that the most likely 2^{n-k} PUF responses all originate from different cosets defining the lower bound on $\tilde{H}_\infty(W|\mathsf{SS}(W))$. On the other hand, $\tilde{H}_\infty(W|\mathsf{SS}(W))$ is upper bounded by the case that the most likely 2^k PUF responses all map to the same coset and this repeated for all 2^{n-k} cosets.

2.4.3 Privacy Amplification

During the last stage of a fuzzy extractor, the aim is to extract an almost uniformly distributed bit string (ϵ close to uniform as measured by the statistical distance between the uniform and the output distributions) from the corrected PUF response with residual min-entropy $\tilde{m}$. As defined in [43], a (n, m, ℓ, ϵ)-extractor Ext is a function $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^\ell$ such that for every distribution X on $\{0,1\}^n$ with min-entropy $H_\infty(X) \geq m$, the output of $\mathsf{Ext}(X, W)$ with W uniform on $\{0,1\}^d$ is ϵ-close to U_ℓ. Here the first input into Ext is an n-bit sample from a distribution, e.g. a PUF response after error correction, and the second input is a d-bit seed uniformly distributed in U_d.

Dodis *et al.* [43] showed that the optimal extractor choice in the information theoretic sense is a strong randomness extractor instantiated via universal hashing [72]. In this case, the output entropy is $\ell = \tilde{m} - 2\log\frac{1}{\epsilon} + 2$. For small ϵ (e.g.,$\approx 2^{-80}$), this implies discarding 158 of the input bits, which in the current application is not practical.

A practical alternative to that are key derivation functions (KDFs) which use cryptographic hash functions in order to extract a *pseudorandom* key from a non-uniform input [73]. The extracted key is computationally rather than statistically indistinguishable form a random uniform string which is accepted as strong security guarantee. In this setting, security is argued in the random oracle model meaning that cryptographic hash functions are modeled as random functions outputting a random response to every unique query. If the same input is fed to the function multiple times, the function outputs the same response. Assuming that an adversary can query a random oracle, the computational security of the construction depends on the number of queries q an adversary can make.

A implementation of such a KDF is the HMAC-based key derivation function (HKDF) scheme proposed in [48] which we use in this thesis. The construction first extracts a pseudorandom key PRK by applying the HMAC function over a random and public seed W and the source key material X

($PRK = \text{HMAC}(W, X)$). It then expands this initial key to a k-bit output by computing $K(1) = \text{HMAC}(PRK, \text{CTXinfo}||0)$, where CTXinfo is some constant context information (e.g., session identifier, time) and $||$ means string concatenation. Then, dependent on the overall output size L needed, the key can be further expanded by $K(i+1) = \text{HMAC}(PRK, K(i)||\text{CTXinfo}||i)$, where $1 \leq i < t$ is a counter of fixed size (e.g., a byte) and $t = \lceil L/k \rceil$. Finally, the HKDF scheme outputs

$$\text{HKDF}(W, X, CTXinfo, L) = K(1)||K(2)||...||K(t), \tag{2.7}$$

where the value $K(t)$ can be truncated to its first $d = L \mod k$ bits if necessary.

In [48], it is shown that $\mathbf{SD}(\mathcal{Y}, U_\ell) \leq \min\left(q \cdot 2^{-\tilde{m}}, \sqrt{q \cdot \mathsf{Col}(\mathcal{X})}\right)$, where $\mathcal{Y}$ is the HMAC output distribution, q is the number of queries an adversary makes to the random oracle used in HMAC, and $\mathsf{Col}(\mathcal{X}) = \sum_x \Pr(\mathcal{X} = x)^2 \leq 2^{-H_\infty(\mathcal{X})}$ depends on the collision probability of the underlying hash function[6].

2.5 MEMS Gyroscopes

MEMS sensors are silicon based devices which typically combine a microelectromechanical structure with an ASIC used to measure a variety of different physical quantities ranging from acceleration and angular rate to magnetic fields, pressure, humidity, etc..

As mentioned previously, this work focuses on the use of MEMS gyroscopes since this sensor type is supposed to be the most promising one regarding the extractable key length due to its complex mechanical structure. In the past few years MEMS gyroscopes have enabled a large number of applications, including classical automotive safety systems (e.g., ESC) and, more recently, new and exciting features in portable consumer devices.

2.5.1 Operating Principle

The operating principle of MEMS gyroscopes is based on the Coriolis effect. They make use of at least two orthogonal vibration modes of a proof mass which is suspended by several beam springs. In the common operation mode the proof mass is driven into resonance (drive mode) by an electrostatic force, typically accomplished by applying alternating voltages to the comb-drive electrodes of

[6] It is assumed that, e.g., an HMAC with SHA-256, provides security $\varepsilon \approx 2^{256}$ which is reasonable given our current knowledge of the state of the art.

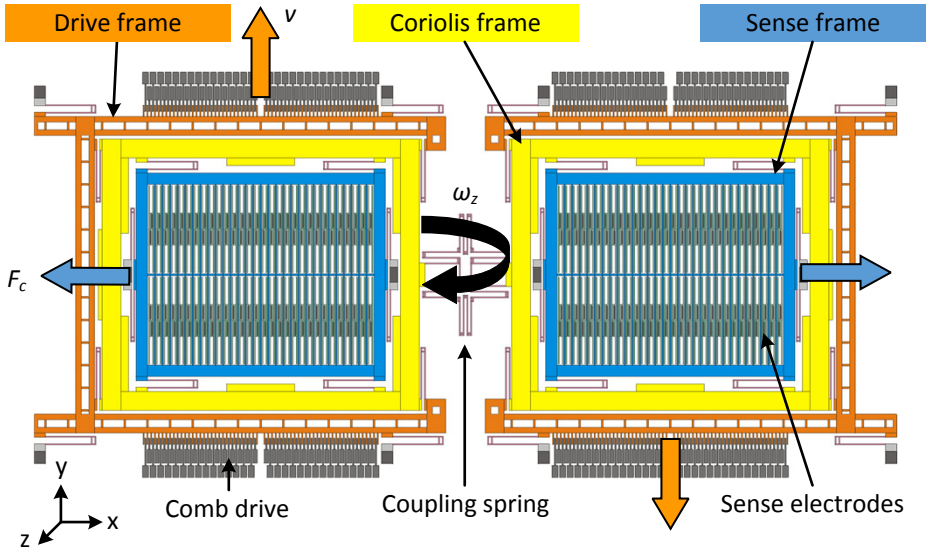


Figure 2.8: Operating principle of a three-frame MEMS gyroscope with an anti-phase oscillation [74].

the drive channel. When the system experiences a rotation, the Coriolis force deflects the proof mass orthogonally to the direction of the drive movement and to the axis of rotation. The Coriolis force $\vec{F}_c$ is proportional to the speed of rotation $\vec{\omega}$ and to the velocity $\vec{v}$ of the proof mass with mass m. The Coriolis force is defined as $\vec{F}_c = 2m(\vec{v} \times \vec{\omega})$.

The deflection causes a change in the capacitance of the sense electrodes. Alternatively, also piezoelectric and piezoresistive measurement concepts exist. In order to increase the deflection of the sense movement, resonance rise is typically also exploited in the detection channel. Then, the structure is designed such that the resonant frequency of the detection mode is close to the resonant frequency of the drive mode. Especially in automotive applications, sensor designs exist that are able to electrostatically tune the resonant frequency of the detection channel. This can be accomplished by applying DC voltages to specific electrodes by which the stiffness of the structure is reduced.

Today's MEMS gyroscopes are very optimized regarding accuracy and reliability. An example of such an optimized structure is shown in Figure 2.8 [74]. Here, the proof mass consists of three frames coupled by U-shaped springs. The design enables the decoupling of drive and sense motion meaning that a movement of the sense frame in the drive direction is suppressed in order to minimize parasitic effects of the drive movement on the measured rate signal.

When a rotation is applied to the system, the Coriolis force acts on the Coriolis frame which then deflects the detection frame. The sensor operates differentially using two identical cores which are coupled by springs. The differential evaluation of the measured signals significantly increases the robustness against external perturbations, such as linear acceleration, vibration, and temperature drift.

Due to the nature of the sensor principle, typically one structure exists per sensitive axis for detection. Since the drive actuation is usually accomplished in the xy-plane (in-plane), the detection for measuring a rotation around the z-axis (z-channel) is also in-plane while the x- and y-channels are based on an out-of-plane movement of the sensing structure. Then, the sense electrodes are located below the proof mass.

2.5.2 Device Under Test

For this work, the DUT was a current MEMS gyroscope designed for the consumer market. The die size is about 2.7 mm^2 and the size of the core is about 1.5 mm^2. The gyroscope has three sensitive axes and it is based on a differential working principle. The sensor structure is depicted in Figure 2.9.

Due to the operating principle of the gyroscope, each detection channel consists of two coupled parts. In the drive channel, comb-electrodes (in the areas drive 1 and drive 2) are used in order to drive the differential parts of the proof mass in opposite directions (anti-phase). The parts x-det and z-det oscillate in the y-direction in an anti-phase manner. The parts y-det of the y-channel perform an anti-phase oscillation in the x-direction. This is accomplished by redirecting the drive movement.

In the detection channels, plate electrodes are used. The sense electrodes of the x- and y- channel are located below the respective parts of the proof mass (in the areas x-det and y-det). In the z-channel, the stator electrodes are located on both sides of the proof mass building parallel plate capacitors (z-det, see also Figure 2.11b). In the second sensing area (z-det 2), the electrodes are then ordered in a mirrored way. Note that x- and z-channel share the same part of the proof mass.

In order to measure frequency modes in the detection channels, the proof mass can also be excited by applying voltages on the detection electrodes. The design allows to drive the differential parts of the x- and y-channel in an in-phase and anti-phase manner. In both cases, different mode types (in-phase and anti-phase modes) can be excited. Due to the design, an in-phase excitation is

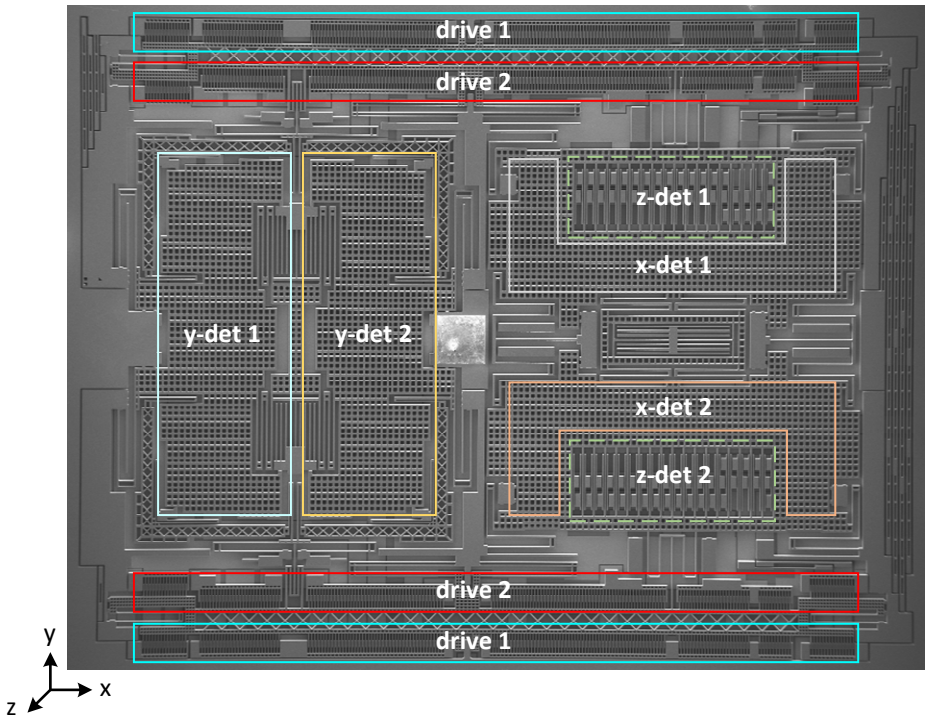


Figure 2.9: DUT: SEM image of the investigated MEMS gyroscope core.

not possible in the drive and z-channel since there the electrostatic forces pull in opposite directions.

2.5.3 Properties

MEMS gyroscopes have a complex structure providing a variety of mechanical properties. Moreover, several electrodes exist in order to drive and measure the mechanical structure. Between the electrodes, electrical properties can be determined. In the following, we comprehensively describe these electrical and mechanical properties.

Capacitances and resistances

Electrical capacitances and line resistances can be measured between the different electrodes which are needed for driving and measuring the sensor. The

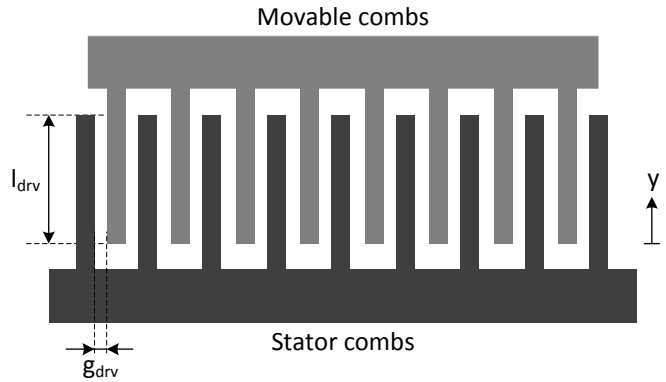


Figure 2.10: Comb electrode used for driving a MEMS gyroscope.

capacitance of a comb-electrode, which is typically used in the drive channel, is defined as

$$C_{drv} = \epsilon_0 N_{drv} \frac{h(l_{drv} + y)}{g_{drv}}, \tag{2.8}$$

where ϵ_0 is the vacuum permittivity[7], N_{drv} is the number of combs, h is the hight of the structure (layer thickness), l_{drv} is the initial overlap, y is the moving direction of the oscillating structure, and g_{drv} is the gap width between the combs of the stator and those of the proof mass (see Figure 2.10).

The capacitance of a plate capacitor, which is typically used in the detection channels, is defined by the plate capacitor equation[8]

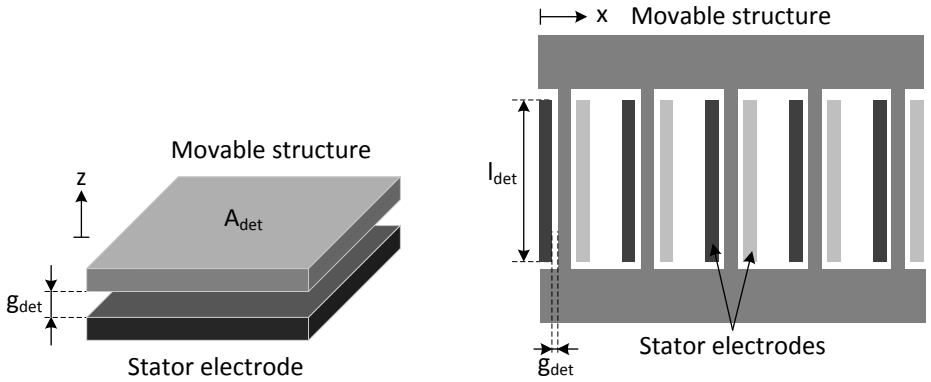
$$C_{det} = \epsilon_0 \frac{A_{det}}{g_{det} + z}, \tag{2.9}$$

where A_{det} is the area of the electrode and g_{det} is the electrode gap (see Figure 2.11a). In case of in-plane detection, z can be replaced by x and $A_{det} = N_{det} l_{det} h$, where N_{det} is the number of plates (see Figure 2.11b). In this case, the differential electrodes are located on both sides of the proof mass building parallel plate capacitors.

Frequency modes

For small displacements, a MEMS gyroscope can be modeled as a driven harmonic oscillator. Figure 2.12 schematically illustrates a 1-degree-of-freedom

[7] Relative permittivity is neglected due to the very low cavity pressure.
[8] It is assumed that the area of the plate capacitor does not change during oscillation.



(a) Plate capacitor for out-of-plane detection.

(b) Plate capacitors for in-plane detection.

Figure 2.11: Parallel plate capacitors used for detection.

mechanical oscillator with mass m, spring constant k, and damping constant d. The spring causes a restoring force which increases linearly with increasing deflection $F_k = -kx$. The damper also counteracts a deflection proportionally to mass' velocity $F_d = -dv = -d\dot{x}$. Following Newton's second law of motion, the sum of all acting forces is equal to the mass times its acceleration in an inertial reference frame

$$m\ddot{x} = -kx - d\dot{x} + F_{ext}. \tag{2.10}$$

Assuming that no external force would be present and using the common substitutions $\delta := \frac{d}{2m}$ and $\omega_0 := \sqrt{\frac{k}{m}}$, we obtain the differential equation

$$\ddot{x} + 2\delta\dot{x} + \omega_0^2 x = 0, \tag{2.11}$$

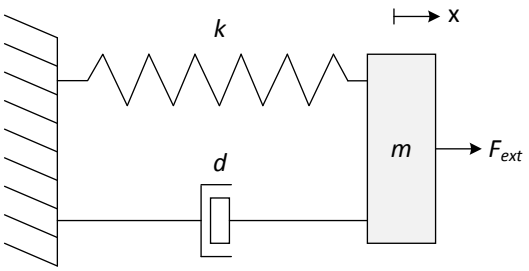


Figure 2.12: 1-degree-of-freedom mechanical oscillator with mass m, spring constant k, and damping constant d.

where δ is the damping coefficient and ω_0 is the resonance frequency of the oscillator without damping [75]. This differential equation has a solution of the form $x(t) = ce^{\lambda t}$ with the characteristic equation $\lambda^2 + 2\lambda\delta + \omega_0^2 = 0$ where

$$\lambda_{1,2} = -\delta \pm \sqrt{\delta^2 - \omega_0^2}. \tag{2.12}$$

Depending on the value of the root, four different cases can be distinguished: *overdamped* ($\delta^2 > \omega_0^2$), *critically damped* ($\delta^2 = \omega_0^2$), *underdamped* ($\delta^2 < \omega_0^2$), and *undamped* ($\delta = 0$) as a special case of the *underdamped* oscillator. MEMS gyroscopes typically operate in a cavity chamber of a few millibar pressure enabling a high sensitivity. Hence, we consider the *underdamped* case only. Then, the resonant frequency is derived as

$$\omega = \sqrt{\omega_0^2 - \delta^2} = \sqrt{\frac{k}{m} - \frac{d^2}{4m^2}}. \tag{2.13}$$

As mentioned previously, the gyroscope is driven by an electrostatic force. Assuming $F_{ext} = F_0 \cos(\Omega t)$ to be a periodic force with angular frequency Ω and amplitude F_0, we obtain an equation of motion of the form

$$\ddot{x} + 2\delta\dot{x} + \omega_0^2 x = \frac{F_0}{m}\cos(\Omega t), \tag{2.14}$$

from which the amplitude response $A(\Omega)$ and the phase response $\varphi(\Omega)$ can be calculated (see e.g., [75]). The amplitude response $A(\Omega)$ derives as

$$A(\Omega) = \frac{F_0/m}{\sqrt{(\omega_0^2 - \Omega^2)^2 + 4\delta^2\Omega^2}}, \tag{2.15}$$

and the phase response $\varphi(\Omega)$ is

$$\tan\varphi(\Omega) = \frac{2\delta\Omega}{\omega_0^2 - \Omega^2}. \tag{2.16}$$

It follows that the resonance curve has its maximum at

$$A_{max} = A(\Omega_{res}) = \frac{F_0/m}{2\delta\sqrt{\omega_0^2 - \delta^2}}, \tag{2.17}$$

where $\Omega_{res} = \sqrt{\omega_0^2 - 2\delta^2}$.

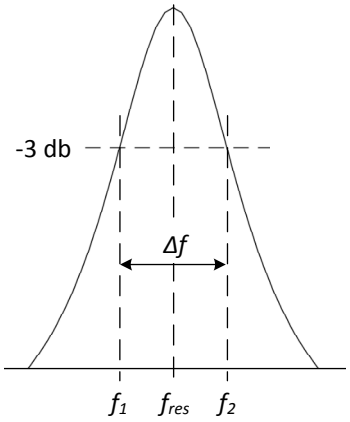


Figure 2.13: Determination of the quality factor Q of a resonance mode using the 3 db method.

Another important property of a MEMS resonator is the quality factor (also Q-factor) of a mode which is defined by the energy loss relative to the stored energy due to damping

$$Q := 2\pi \frac{\text{energy stored}}{\text{energy dissipated per cycle}}. \tag{2.18}$$

The quality factor can be determined from the decay time of the oscillation after turning off the driving force or from the resonance curve, e.g. by using the 3 db method (see Figure 2.13). The Q-factor describes the bandwidth of the resonance curve and it can be calculated by the ratio of the resonance frequency f_{res} to the peak's width $\Delta f = f_2 - f_1$ at $-3db$ relative to the peak's maximum

$$Q := \frac{f_{res}}{\Delta f}. \tag{2.19}$$

High Q-values are important for MEMS gyroscopes since the oscillation amplitude at resonance increases proportional to Q. Hence, a higher sensitivity can be reached by increasing the Q-factor.

Due to their complex structure, MEMS gyroscopes have a large number of resonant modes in practice. The resonant modes of such a multiple-degree-of-freedom system can be calculated by solving the differential equation

$$M\ddot{x} + D\dot{x} + Kx = 0, \tag{2.20}$$

where M is the mass matrix, D is the damping matrix, and K is the stiffness matrix of the system. However, for a modal analysis using finite element

method (FEM) the damping matrix D is usually neglected. The difference in an obtained frequency positions is

$$\omega = \omega_0 \sqrt{1 - \frac{1}{2Q^2}}. \tag{2.21}$$

Since typical Q-factor values range from several hundred up to several thousand, the difference is in the sub-hertz range only. When neglecting D, the FEM can be used in order to solve the simplified equation

$$M\ddot{x} + Kx = 0. \tag{2.22}$$

Notice that the exact characteristics vary from sensor to sensor due to tolerances in the manufacturing process. Simulations can just be used to calculate the characteristics of an idealized sensor structure. Nevertheless, in order to estimate the distribution of the sensors' properties, Monte Carlo simulations are used in practice with the range of tolerances in the manufacturing process as input parameters.

Resonance frequency tuning

As mentioned in Section 2.5.1, the characteristic of frequency tuning is important, e.g., for sensor designs which are exploiting resonance rise in the detection channels. Applying an electrical voltage to a plate electrode generates an electrostatic force F_{el} which counteracts the spring force F_{mech}

$$F_{sum} = F_{mech} + F_{el} = -kx + \frac{\partial}{\partial x}\frac{1}{2}CU^2. \tag{2.23}$$

Hence, the effective stiffness of the structure is reduced by

$$-\frac{\partial F_{sum}}{\partial x} = k_{eff} = k - \frac{1}{2}\frac{\partial^2 C}{\partial x^2}U^2. \tag{2.24}$$

With Equation (2.9), we obtain k_{eff} for a plate capacitor

$$k_{eff} = k - \frac{1}{2}\frac{\partial^2}{\partial x^2}U^2 = k - \frac{\epsilon_0 A_{det}}{(g_{det} + x)^3}U^2 \approx k - \frac{\epsilon_0 A_{det}}{g_{det}^3}U^2. \tag{2.25}$$

As can be seen from Equation (2.25), k_{eff} changes proportional to U^2 and, thus, the frequency shift follows a parabola, whereby the resonance frequency is maximal for $U = 0$. The exact parameters of the parabola are slightly different for each mode and sensor.

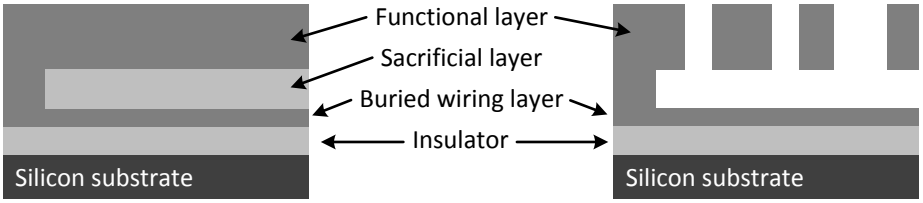


Figure 2.14: Schematic cross-section of a MEMS gyroscope manufactured in surface-micromachining technique before (left) and after (right) structuring.

Quadrature signals

In the operating mode, the quadrature signal is an error signal caused by a deviation from the sensor's ideal moving direction and it can be measured via the electrodes of the detection channels. The quadrature movement is caused by asymmetries of the sensor structure due to process imperfections, generating a signal in the sense path even when the angular rate is zero [76]. In the operating mode, the quadrature signal can be distinguished from the rate signal since it is proportional to the structure's displacement. Hence, it has a 90° phase shift with respect to the angular rate signal which is proportional to the structure's velocity.

2.5.4 Manufacturing

The high performance and small areas of todays MEMS gyroscopes are enabled by highly optimized micromachining technologies. The sensor structures are usually made on top of a silicon wafer by deposition and selective etching (surface-micromachining). The main materials used are polysilicon and thermal oxide. Figure 2.14 shows a schematic cross-section[9] as described in [41, Chapter 28]. The first step is the deposition of an insulation layer such as silicon dioxide on the silicon wafer. Next step is the deposition of a first polysilicon layer and a silicon dioxide layer. Both layers are structured by photolithography processes and reactive ion etching. The polysilicon layer is also called buried wiring layer and it provides electrical contacting to the functional layer. The structured silicon dioxide layer (also called sacrificial layer) provides the anchor holes for the functional layer. Then, the thick mechanical polysilicon layer and a metallization are deposited and structured followed by removing

[9] It has to be noted that Figure 2.14 shows a simplified layer stack meaning that process flows and layer stacks are more complex in practice (see e.g., [77]).

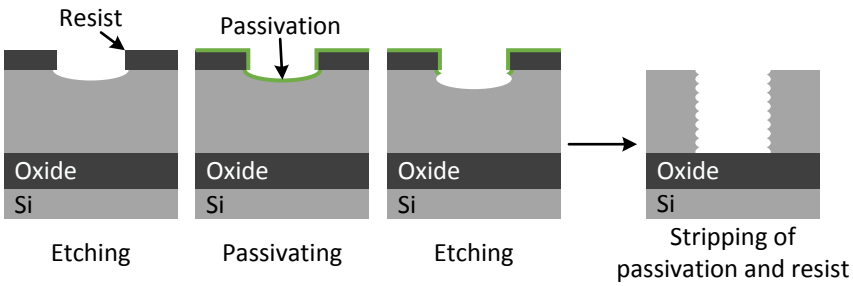


Figure 2.15: Schematic process flow of DRIE.

the sacrificial layer in order to release the functional layer. Finally, the sensing elements are encapsulated by a capping wafer in a vacuum bonding process.

Fundamental technologies in the manufacturing of MEMS gyroscopes are the deposition of the used materials as well as the structuring of the layers. As a consequence of process tolerances, the individual device characteristics vary. Epitaxial deposition and deep reactive ion etching (DRIE) of the mechanical polysilicon layer are of special significance, since they are mainly responsible for variations in the sensor parameter values.

The mechanical polysilicon layer is deposited with the epi-poly process. The thickness of this layer is typically in the order of 20 µm [78]. Process tolerances lead to a slight variation of the layer thickness. The final layer thickness can be controlled in a chemical mechanical polishing (CMP) step enabling a thickness uniformity in the order of ±0.5 µm [41, Chapter 7].

DRIE combines physical and chemical processes in order to etch material from the wafer. One of the main DRIE techniques is the so-called *Bosch process* which enables etching of deep and near-vertical trenches [79]. The anisotropic nature of the process is accomplished by a combination of alternating etching and sidewall passivation steps (see Figure 2.15). During etching, the ion bombardment removes the passivation layer from the trench bottom only while the sidewalls remain protected preventing lateral etching. Oxide (e.g., SiO_2) can be used as etch stop in order to manufacture trenches of fixed depth [41, Chapter 23].

The DRIE process is subjected to different kinds of variations (see Figure 2.16) [80]. In the following, four types of them are described:

- *Structure width variation.* Due to a varying etch undercut of the mask, the actual structure widths are reduced (Figure 2.16a). The narrowing is typically in the order of 0.3 µm and it varies by about ±0.15 µm depending

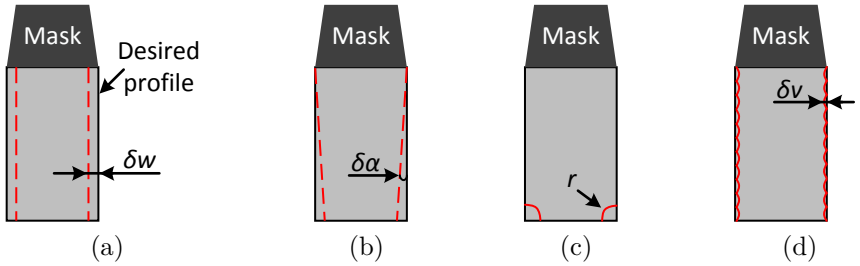


Figure 2.16: Sources of MEMS properties' variation based on process imperfections of DRIE: structure width variation (a), sidewall angle variation (b), notching effect (c), and sidewall scalloping (d) [80].

on the position of the die on the wafer, on the etching process parameters, and on the etching tool itself.

- *Sidewall angle variation.* Ideally, the DRIE process generates vertical sidewalls ($\alpha = 90°$). However, the actual sidewall angles are subjected to minor variations in the order of $\pm 0.3°$ (Figure 2.16b) [41, Chapter 23]. This is caused by the slightly oblique incidence angle of the ions coming from the ion source.

- *Notching effect.* When the DRIE process reaches an insulator surface, lateral etching is forced due to charging effects (Figure 2.16c) [41, Chapter 23].

- *Sidewall scalloping.* The alternating etching and passivation steps of the DRIE process cause scalloping of the sidewalls (Figure 2.16d). The resulting sidewall roughness is typical in the lower nanometer range [41, Chapter 23].

The combination of process tolerances for layer deposition, for photolithography, and for etching leads to a variation of the geometric dimensions of the structures. For the position of resonance modes, structure width variation is the predominant factor. Resonance frequencies typically vary between $\pm 1\,\%$ and $\pm 5\,\%$ [81]. Additionally, electrical properties are affected due to changes in the effective electrode areas and gaps.

As mentioned in Section 2.5.3, the quadrature error is a result of asymmetries in the structure. Different effects cause in-plane and out-of-plane quadrature, respectively. In-plane quadrature is mainly caused by varying widths of identical springs across a sensor as a result of structure width variation. Out-of-plane

quadrature is caused by non-ideally vertical sidewall slopes forcing an out-of-plane movement of an in-plane oscillator.

Other influencing factors for mechanical parameters include variations of Young's modulus of polysilicon, of the cavity pressure, of residual stress in the mechanical layers and of stress caused by subsequent processes, such as packaging and soldering. An additional factor for the variation of electrical properties are process tolerances affecting the doping concentrations [41, Chapter 6].

On a wider perspective, manufacturing tolerances can be divided into four categories with increasing level of expected variation [82]:

- within-die variations,
- within-wafer variations,
- within-batch variations,
- batch-to-batch variations.

Within-die variations are small variations to which a single sensor is subjected, e.g., leading to asymmetries in the structures. Within-wafer variations describe the variation between sensors originating from the same wafer. Especially for this category, some of the effects seem to be systematic, such as structure width variation [83], caused by non-uniformities in the process, e.g., temperature and plasma non-uniformity [84]. As a result, sensors that are positioned close together on a wafer are expected to be more similar to each other than those being further apart. Within-batch variations characterize variations between sensors from different wafers but manufactured in the same batch. Such variations are also caused by non-uniformities. Batch-to-batch variations are caused by long-term fluctuations of the processes.

Chapter 3

Experimental Setup

In this chapter, a new electrical measurement method is introduced which was developed in this work and was published in [85]. Wafer-level and module-level measurement techniques are explained and the dedicated setup of packaged sensor modules used to obtain the experimental data is described. Finally, an overview of the robustness tests that were performed on the sensor modules is given. Observe that the sensor modules were subjected to these tests in order to investigate the stability of the derived fingerprints to varying temperature and aging conditions.

3.1 Measurement Method

The mechanical parameters introduced in Section 2.5 can be extracted by a dynamic characterization of the sensor structure [86, 87]. The characterization can be performed by either optical or electrical measurement methods [87–89]. In order to be able to perform the characterization in a highly automated and cost efficient way, electrical characterization is the preferred method and was used in this work.

In the following, an electrical method is presented, which was developed as a part of this thesis, enabling the rapid dynamic characterization which has been a time-consuming and expensive procedure in the past. The main component of the setup was a Red Pitaya which is an open-source-software measurement and control tool. The Red Pitaya and further equipment such as a PXI-1042 station from National Instruments (NI) with switch matrices and programmable power supply cards were controlled via Python(x, y) running on a standard notebook. The block diagram of the measurement method is shown in Figure 3.1.

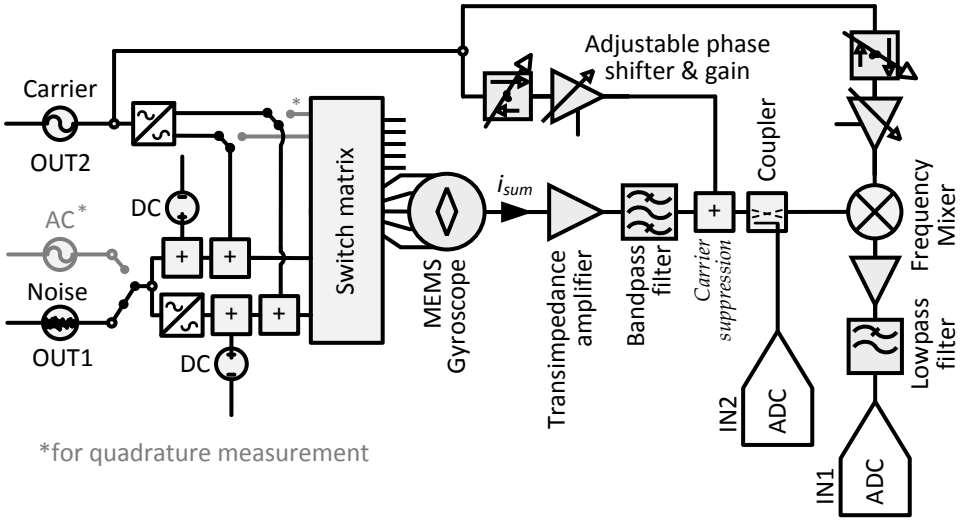


Figure 3.1: Block diagram of the measurement method. For the measurement of in-phase modes, inverting of noise and carrier has to be skipped.

3.1.1 Noise Excitation

For dynamic characterization, the micromechanical structure is usually excited electrostatically. This can be achieved by applying biased alternating voltages[1] U_p^* and U_n^* on a differential pair of electrodes[2](denoted as p and n). As mentioned in Section 2.5.2, two different types of frequency modes exist, namely in-phase and anti-phase modes. Depending on which mode type is measured, the alternating part of one excitation voltage has to be inverted (for anti-phase modes) or $U_p^* = U_n^*$ (for in-phase modes)

$$\begin{aligned} U_p^*(t) &= U_{DC} + U_{AC}\sin(\Omega t) \\ U_n^*(t) &= U_{DC} \pm U_{AC}\sin(\Omega t), \end{aligned} \tag{3.1}$$

where Ω is the angular frequency of the periodic AC voltage and U_{DC} and U_{AC} represent the voltage amplitudes.

[1] In principle, the DC voltage can also be applied on the proof mass.
[2] A single-sided excitation is possible as well. However, the effective electrostatic force is only half as big in this case.

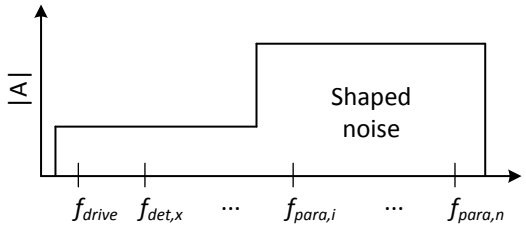


Figure 3.2: Schematic illustration of the synthesized truncated white noise which allows for applying higher voltages without destroying the sensor due to a reduced excitation of the pronounced drive and detection modes.

The minus sign of $\pm$ applies to the case of anti-phase modes throughout the following sections. In case of comb-electrodes (Equation (2.8)), the acting electrostatic force is

$$F_{el,drv}(t) = \frac{2\epsilon_0 N_{drv} h U_{DC} U_{AC} \sin(\Omega t)}{g_{drv}}, \tag{3.2}$$

where ϵ_0 is the vacuum permittivity, N_{drv} is the number of combs, h is the hight of the structure (layer thickness), and g_{drv} is the gap width between the combs of the stator and those of the proof mass (see also Section 2.5.3).

For plate capacitors (Equation (2.9)) used in the detection channels, when using a Taylor approximation and truncating the Taylor series at the second order[3], the electrostatic force is

$$F_{el,det}(t) \approx \frac{2\epsilon_0 A_{det} U_{DC} U_{AC} \sin(\Omega t)}{g_{det}^2}, \tag{3.3}$$

where A_{det} is the area of the electrode and g_{det} is the electrode gap (see also Section 2.5.3).

For the developed measurement method, a synthesized truncated white noise (shaped noise) with a dedicated amplitude spectrum concentrated in the specific frequency range of interest was used for the excitation. This signal was provided by the Red Pitaya (OUT1). The shaping of the noise was achieved digitally by performing an inverse Fourier transformation of the complex input spectra. In particular, the objective of shaping the noise is to reduce the excitation of the pronounced drive (f_{drive}) and detection modes ($f_{det,x/y/z}$) and

[3]The Taylor series can be truncated at the second order because of the small oscillating amplitudes due to the used noise excitation.

to increase the excitation of higher (parasitic) modes ($f_{para,i}$). This allows for applying significantly higher voltages without destroying the sensor, leading to an increase of the mechanical amplitude of the higher modes (see Figure 3.2). In Equation (3.1), the shaped noise replaces the alternating voltage component.

In order to distribute the excitation voltages to the different electrodes, we used a PXI-1042 station from NI which controls three switch matrices M9128A from Agilent. The DC voltages were provided by a NI PXI-4110 programmable DC power supply card.

3.1.2 Carrier Frequency

For measuring the structure's movement, a carrier frequency was used which is amplitude modulated by the oscillation of the MEMS structure and the sum current i_{sum} was measured at the proof mass. The carrier frequency was superposed with the excitation voltages extending Equation (3.1) to

$$\begin{aligned} U_p(t) &= U_{DC} + U_{AC}\sin(\Omega) + U_c\sin(\omega t) \\ U_n(t) &= U_{DC} \pm U_{AC}\sin(\Omega) \pm U_c\sin(\omega t), \end{aligned} \tag{3.4}$$

enabling to drive and measure the structure on the same electrodes. The carrier frequency was provided by the Red Pitaya (OUT2) and it was set to 10.7 MHz (1.5 V_{pp}) enabling a high bandwidth for the measurements. The relatively high frequency of the carrier signal leads to a higher electrical current that is proportional to this frequency. Increasing the signal-to-noise-ratio (SNR) is beneficial to observe the very small signals rising from the higher frequency modes.

In order to measure the quadrature signals, the carrier signal was applied to the sensing electrodes of the detection channels while driving the sensor in the common drive mode as described by Cigada *et al.* [90]. In order to achieve the best possible repeatability of the quadrature measurement, a constant vibration amplitude was adjusted for the drive mode through controlling the output signal of the drive channel.

3.1.3 Derivation of the Sum Current

As mentioned previously, the used method is based on measuring the sum current i_{sum} at the proof mass. The equivalent circuit of the sensor is shown in Figure 3.3. The sum current i_{sum} is defined by

$$i_{sum}(t) = \sum_{j=1}^{n} \frac{dQ_j(t)}{dt} = \sum_{j=1}^{n} \frac{d(C_j(t)\cdot U_j(t))}{dt}. \tag{3.5}$$

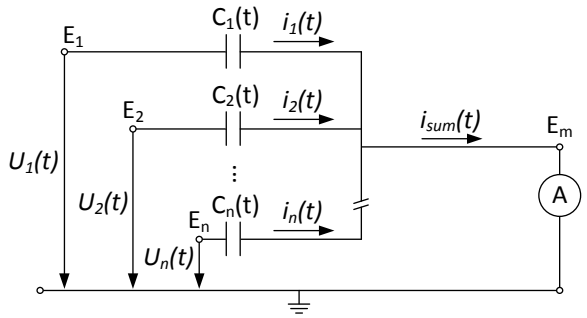


Figure 3.3: Equivalent circuit of a MEMS gyroscope with current measurement at the proof mass.

For this thesis, the different channels of the sensor were measured successively and excitation voltages according to Equation (3.4) were always applied to a pair of electrodes in a particular channel. In this section, i_{sum} is derived for the cases:

(i) comb-electrodes (drive channel), anti-phase modes,

(ii) plate electrodes (detection channels), anti-phase modes,

(iii) plate electrodes (detection channels), in-phase modes.

The capacitance of a comb-electrode is defined in Equation (2.8) and the electrode overlap is changing with

$$\begin{aligned} y_p(t) &= A(\Omega)\sin(\Omega t - \varphi(\Omega)) \\ y_n(t) &= \pm A(\Omega)\sin(\Omega t - \varphi(\Omega)), \end{aligned} \tag{3.6}$$

as a consequence of the structure's oscillation depending on the mechanical phase response $\varphi(\Omega)$ (see Equation (2.16)) and the mechanical amplitude response $A(\Omega)$ (see Equation (2.15)) which significantly increases when Ω is at a resonance frequency of the system.

Using Equations (2.8) and (3.4) to (3.6) gives the equation for the sum current i_{sum} for the case (i)

$$\begin{aligned} i_{sum,(i)}(t) = \frac{\epsilon_0 N_{drv} A(\Omega) h}{g_{drv}} [&(\omega + \Omega)(U_c \sin(\omega t + \Omega t - \varphi(\Omega))) \\ &-(\omega - \Omega)(U_c \sin(\omega t - (\Omega t - \varphi(\Omega))))] + f(\Omega t). \end{aligned} \tag{3.7}$$

Note that a movement of a mode can be observed in the equally spaced sideband around the carrier (see Figure 3.4). Since the carrier signals of U_p and

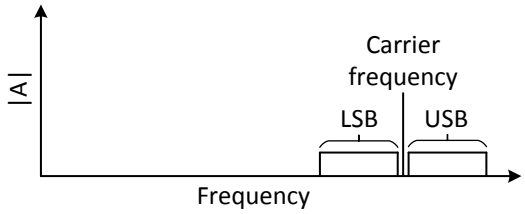


Figure 3.4: Schematic illustration of modulated amplitude spectrum with carrier frequency and lower and upper sidebands (LSB and USB).

U_n have a 180° phase shift, the signal of the carrier itself is rejected due to the addition of individual currents. This is beneficial since it allows for using a high carrier amplitude without overloading the transimpedance amplifier that acquires i_{sum}. Note that additional terms $f(\Omega t)$ can be neglected since these signals are in the baseband and they are filtered by a subsequent bandpass filter.

The capacitance of a plate electrode in a detection channel is defined in Equation (2.9). The displacements in the x-direction (or in the z-direction, respectively) can be defined as those in the y-direction (see Equation (3.6)). For the anti-phase and in-phase modes in the detection channels (cases (ii) and (iii)), Equation (2.8) is replaced by Equation (2.9) in Equation (3.5) and a Taylor approximation truncated at the second order is used for calculating i_{sum}. Thus, for the case (ii), i_{sum} is

$$\begin{aligned} i_{sum,(ii)}(t) \approx -\frac{\epsilon_0 A_{det} A(\Omega)}{g_{det}^2}[&(\omega+\Omega)(U_c \sin(\omega t + \Omega t - \varphi(\Omega))) \\ &-(\omega-\Omega)(U_c \sin(\omega t - (\Omega t - \varphi(\Omega))))] + f(\Omega t). \end{aligned} \tag{3.8}$$

For in-phase modes (case (iii)), $U_p = U_n$ and $y_p = y_n$. Then, i_{sum} can be derived as

$$\begin{aligned} i_{sum,(iii)}(t) \approx -\frac{\epsilon_0 A_{det} A(\Omega)}{g_{det}^2}[&(\omega+\Omega)(U_c \sin(\omega t + \Omega t - \varphi(\Omega))) \\ &-(\omega-\Omega)(U_c \sin(\omega t - (\Omega t - \varphi(\Omega))))] \\ &-\frac{2\epsilon_0 A_{det} \omega U_c}{g_{det}} \cos(\omega t) + f(\Omega t), \end{aligned} \tag{3.9}$$

whereas an additional term at the carrier frequency ω arises due to the addition of the two in-phase carrier signals.

3.1.4 Signal Processing

The sum current i_{sum} measured at the proof mass was converted to an electrical voltage by a current-to-voltage converter and amplified (transimpedance amplifier DHPCA-S from FEMTO, with transimpedance gain $g_a = U_{OUT}/I_{IN} = 10^4 V/A$). After the transimpedance amplifier, a bandpass filter (SBP-10.7+ from Mini Circuits) was utilized to delete the parts of the signal in the baseband $f(\Omega t)$ which are caused by the excitation voltages.

Next step is the further suppression of the carrier signal. In particular, this is necessary in case (iii) in order to reject the signal at the carrier frequency ω. However, this is also recommended for the other cases (i) and (ii) because the carrier signal is usually not canceled out completely due to a slight mismatch of the sensor's electrode pairs as a consequence of manufacturing tolerances. The elimination of the carrier signal is important because its amplitude is large compared to the amplitude of the sideband signals that contain the needed information. After canceling the carrier, the signal can be further amplified without overloading the second amplifier stage or the input of the Red Pitaya (IN1).

For carrier suppression and analog demodulation, the carrier signal was split twice directly after OUT2 (Mini Circuits ZFRSC-42-S+). The phase and amplitude of these signals have to be adjusted properly. This can be achieved by using two I&Q-modulators (Mini Circuits ZFMIQ-10M) that are controlled via DC voltages. These voltages can be provided from the four pulse-width modulators (PWMs) of the Red Pitaya (additional low-pass filtering of the PWM output is necessary). In order to be able to make use of the full output range of the I&Q-modulators, also negative DC voltages are necessary. To accomplish this, a dedicated board was designed using two NE5532P integrated circuits (ICs) from NI that added a negative DC bias to the PWM output signals. The optimal PWM setting for the carrier suppression has to be adjusted for each sensor individually. Hence, the signal was decoupled (Mini Circuits ZFDC-15-6-S+) to the second input of the Red Pitaya (IN2) after the carrier suppression. Based on this signal, the optimal PWMs setting for the carrier suppression could be determined dynamically on the Red Pitaya.

For the analog demodulating of the signal, a frequency mixer (Mini Circuits ZX05-1L-S+) was used. After the demodulation, the signal was amplified and low-pass filtered before it was acquired by the Red Pitaya (IN1). For this step, a dedicated board was designed based on four THS3115 high-speed ICs from NI for signal amplification. We used a sample rate of 1.95 Msps[4] which allows

[4]The sample rate can be set by decimation of the maximal sample rate (125 Msps).

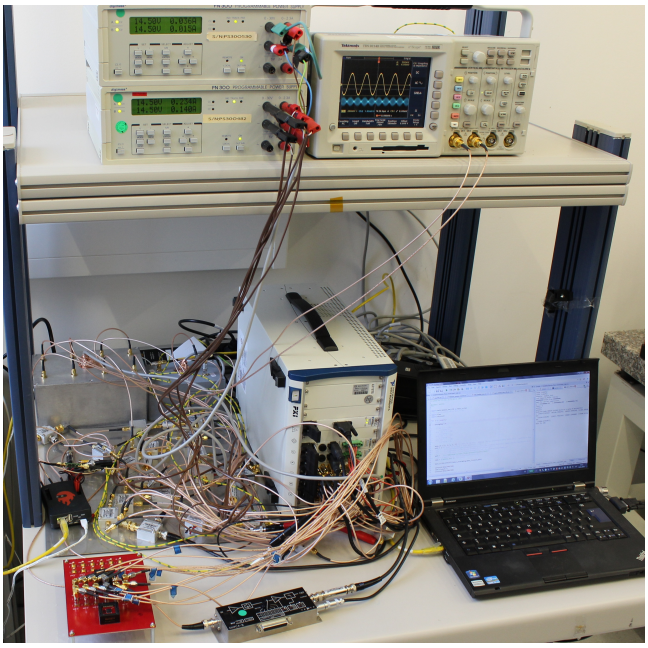


Figure 3.5: Picture of the measurement setup.

for measuring the amplitude spectrum of the signal with a resolution of 2 Hz within a second and a bandwidth of more than 900 kHz. Assuming a complete suppression of the carrier, the demodulated and low-pass filtered voltage signal is derived as

$$u_{(i)}(t) = g_a \frac{\epsilon_0 N_{drv} A(\Omega) h}{g_{drv}} \Omega U_c \cos(\Omega t - \varphi(\Omega)), \tag{3.10}$$

for the case (i) and

$$u_{(ii),(iii)}(t) \approx -g_a \frac{\epsilon_0 A_{det} A(\Omega)}{g_{det}^2} \Omega U_c \cos(\Omega t - \varphi(\Omega)), \tag{3.11}$$

for the cases (ii) and (iii).

A picture of the measurement setup is shown in Figure 3.5. Further devices which were used are Mini Circuits ZFL-500HLN+ amplifiers for adjusting the signal level of the carrier, Mini Circuits ZFBT-4R2GW+ Bias Tees for combining AC and DC voltages, Mini Circuits ZFRSC-42-S+ and ZFSCJ-2-2-S Splitters for splitting (and converting in case of anti-phase modes) the excitation and carrier signals, and for carrier suppression. For providing power supply voltages for the measurement boards and amplifiers, two Digimess PN300 power

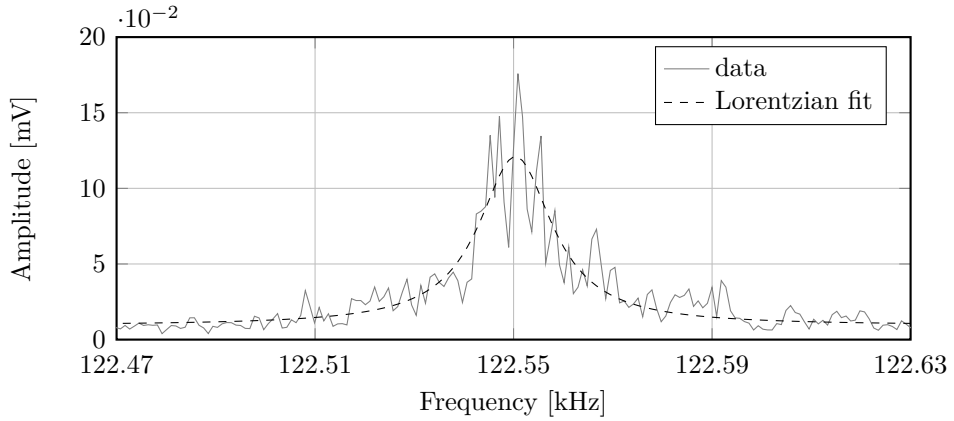


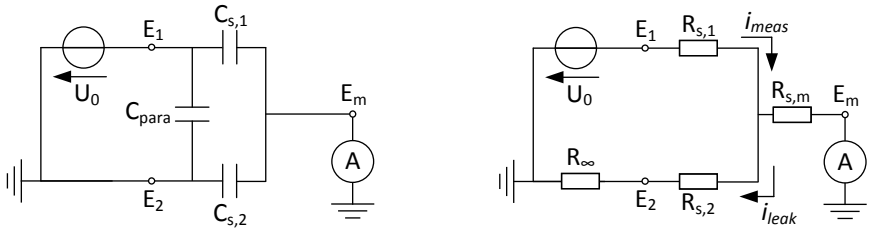
Figure 3.6: Measured frequency mode with corresponding Lorentzian fit.

supply devices were used. Moreover, coaxial cables served for transmitting high frequency (HF) signals.

The measurement data were transmitted to the notebook for storing and evaluation. The amplitude spectrum ($|V(\Omega)|$) was calculated with a fast Fourier transform (FFT)[5] from which the position and amplitude of the structure's frequency modes were determined. In order to increase the repeatability of the measurements, we approximated the resonance curves using a Lorentzian fit and took its peak value as actual mode position (see Figure 3.6). It has to be noted that the measured current is proportional to the structure's velocity and not to the deflection. In order to get a signal which is proportional to the deflection, an additional integration of the signal is necessary.

In addition, the quality factors of the modes were determined from the resonance curves using the 3 db method (see Section 2.5.3). The characteristic of frequency tuning was determined by measuring the pronounced detection mode position for three different DC voltages in each detection channel. Then, a parabola $y(x) = ax^2 + bx + c$ was fitted to the data from which the parameters a and b were taken as features. Furthermore, the quadrature amplitudes were measured in the time domain while driving a sensor at a particular drive frequency.

[5]It has to be noted that for practical applications with constrained memory resources memory-saving alternatives might be chosen, such as zoom FFT.



(a) Measurement of C_s.

(b) Measurement of R_s.

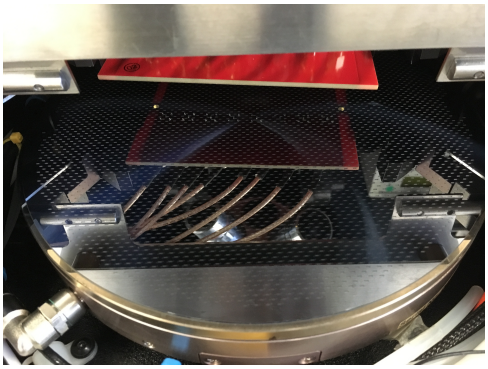
Figure 3.7: Equivalent circuits for the precise measurement of C_s and R_s.

3.2 Impedance Analyzer Measurement Technique

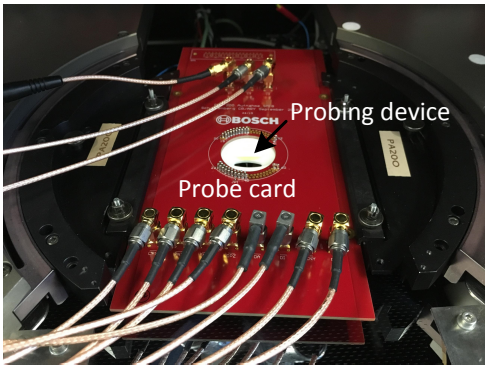
The electrical resistances and capacitances between the sensor electrodes and the proof mass were measured with an Agilent 4294A precision impedance analyzer [91]. In particular, the impedance analyzer measures the real part (resistance R) and imaginary part (reluctance jX) of impedance $Z = R + jX$ and derives from them resistance and capacitance values. The underlying equivalent circuit model consists of a resistance R_s and a capacitance C_s connected in series, where R_s represents the line resistance of the sensor and C_s depicts the capacitance of the capacitor formed by the electrodes.

Since MEMS gyroscopes are typically driven and measured via differential lines, the parasitic capacitance C_{para} or, respectively, the leakage current I_{leak} have to be considered. Thus, for an accurate measurement of the capacitance $C_{s,1}$ between an electrode E_1 and the proof mass E_m, the differential electrode E_2 should be grounded so that E_2 and E_m have the same potential (Figure 3.7a). Otherwise, $C_{s,2}$ could distort the measurement. In this case C_{para} changes the measurement only marginally since $C_{para} \ll C_{s,1}$. On the other hand, for a precise measurement of $R_{s,1}$, E_2 should be connected to ground with a high resistance ($R_\infty = 10M\Omega$) in order to minimize the leakage current i_{leak} that would falsify the measured value of $R_{s,1}$ (Figure 3.7b). Note that E_2 should not remain unconnected in order to avoid electrostatic charging effects.

The impedance analyzer has four measurement terminals (high current H_c, high potential H_p, low current L_c and low potential L_p). In order to achieve highest measurement accuracy, the high and low side ports should not be interconnected until just before the DUT. However, since the measurement signals had to be distributed dynamically to the different electrodes, leading all four wires each up to the sensor pads was not possible with the used setup. Hence, an open and a short calibration of the setup was necessary in order to elimi-



(a) Measured wafer fixed on a movable chuck.



(b) Probe card.

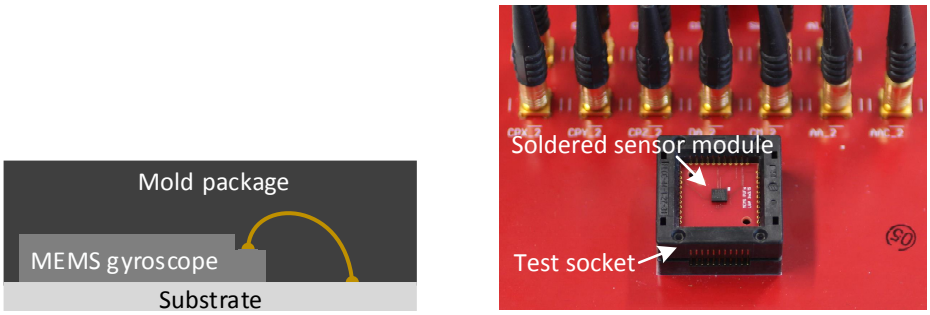
Figure 3.8: View of a probe station which enables the automated measurement of silicon wafers.

nate resistances and stray capacitances of the setup. For the open calibration, a capacitance measurement was performed without contacting the sensor pads. The short calibration was performed with a special chip having a short-circuited pad area.

Measured capacitance values ranged from several hundred fF to few pF while resistances were in the kΩ-range. For this range, the uncertainty of measurement is about ±1 % (see [91, Figure 1-21]).

3.3 Wafer-Level Measurement Technique

MEMS sensors are manufactured on silicon wafers (Figure 3.8a). They can be measured in a highly automated manner directly on the wafers using probe stations, such as PA 200 by Süss Micro Tec. On the probe station, the wafer is fixed on a movable chuck by a vacuum pump. Moreover, a probe card and a contacting device with probes mounted on the probe card are necessary in order to directly access the sensor pads (Figure 3.8b). Enabling the automation of the measurements, an initial alignment of the wafer and a setting of the chuck height have to be made. Based on a map of the wafer, the chuck positions the desired sensor under the probe card automatically and it pushes the sensor pads smoothly against the contact probes. Note that the noise floor is usually higher on wafer-level compared to module-level measurements since the probes are acquiring interference signals from the environment as well.



(a) Schematic layout.

(b) Board with test socket for module-level measurements.

Figure 3.9: Schematic layout of the dedicated packaged sensor modules and measurement board with test socket.

3.4 Packaged Sensor Modules and Module-Level Measurement Technique

In contrast to wafer-level, module-level usually means that the MEMS silicon die is mounted together with an ASIC on a substrate and enclosed by a molding compound. Additionally, sensors are typically soldered on a bigger printed circuit board (PCB) in practical applications.

In this case, different materials, e.g., with different thermal expansion coefficients are rigidly connected. This is expected to affect the physical characteristics of a MEMS PUF temporarily, e.g. due to thermal stress induced by varying temperature, and permanently, e.g. due to heavy thermal loads and/or thermal activated changes over time. Thus, the long-term stability and the temperature stability of MEMS fingerprints should be investigated on soldered modules.

Since common ASICs cannot measure all the features of interest, we built up dedicated modules (see Figure 3.9a) in order to be able to directly access the sensor pads with the described measurement setup using a dedicated measurement board and a test socket (see Figure 3.9b).

We built 468 dedicated $3.5 \times 3 \times 1$ mm^3 LGA packaged sensor modules in a standard packaging process (wafer sawing, die attach, Au-wire bonding, molding, high temperature storage at 170°C for 4 h, laser marking, sawing). The used substrate material (thickness 0.25 mm) was HL832NS [92], which has a low thermal mismatch to the used molding compound G760L [93]. After the packaging process, we dried the sensor modules for 24 h at 120°C and soldered

them onto 17.5 × 17.5 × 1.6 mm^3 PCBs in order to consider the influence of rigid solder joints to a common FR4 (Tg 170°C) PCB. Before soldering, the PCBs were dried for 4 h at (130°C).

3.5 Test Procedures

As mentioned in Section 2.2, a core requirement for a MEMS PUF is that a cryptographic key that is derived from it can be reconstructed across the whole range of environmental conditions in which the device is supposed to operate. Hence, we performed the following tests which are typically required for consumer applications.

Baseline measurements

We initially performed a baseline measurement for all 468 sensors at room temperature (RT). These measurements are the basis for the evaluation in the following chapters.

Repeatability at RT (REP)

This test evaluates the stability of the PUF responses at RT. It determines the repeatability of the measurements which is limited by the accuracy of the measurement method and the setup. In total, we performed 1500 measurements across 5 sensors.

Temperature dependency (TD)

To investigate the stability of the PUF responses across the temperature range from −40°C to 85°C, which is often required for consumer applications, we measured 10 sensors in a Vötsch VTM7004 heating oven which has a cooling speed of 4.7 K/min and a heating speed of 5.3 K/min, respectively. Five of the sensors were measured in a single cycle (20°C, −40°C, 20°C, 85°C, 20°C) while the others were measured in five temperature cycles so that we had 30 measurements at −40°C and at 85°C in total. This test causes non-permanent changes in the physical characteristics of a sensor (e.g., due to thermal stress, temperature dependence of material properties, and increased measurement noise). As a result, the stability of the PUF responses is expected to be reduced.

High temperature storage life (HTSL)

Following the Solid State Technology Association (JEDEC) Standard JESD22-A103E [94], 50 sensor modules were stored at a temperature of 125°C (condition A) for over 1000 h without any electrical conditions applied. This test causes

thermally activated changes over time, e.g., changing stress conditions. It leads to a permanent change in the physical characteristics of a sensor and, thus, it reduces the stability of the PUF responses.

Temperature cycling (TC)

In order to estimate the effect of alternating high- and low-temperature extremes, we conducted 1000 temperature cycles between −40°C and 125°C on 50 sensors according to the JEDEC Standard JESD22-A104 [95] (test condition G, soak mode 4, soak time 15 min, load transfer time 5 min). A thermal shock test chamber TSS-70-130 from CTS was used (two chamber device). Before the tests were started, the modules were dried at 120°C for 24 h to be in a dry and comparable initial state. Due to rapid temperature variation and different coefficients of thermal expansion, high mechanical stress is induced. As a result, this test causes permanent changes in the physical characteristics (e.g., due to delamination) of a sensor reducing the stability of the PUF responses.

Chapter 4

Deriving MEMS Fingerprints

In this chapter, the used quantization method for deriving binary strings from the analog measurement values of MEMS sensors is explained. Then, MEMS properties are identified which are suitable for PUF applications. Afterwards, the uniqueness and robustness of the derived fingerprints is evaluated by calculating inter and intra Hamming distances. Additionally, the effect of considering sensors from several wafers out of different batches compared to sensors from a single wafer is investigated. Parts of this chapter were published in [85, 96, 97].

4.1 Multi-Bit Quantization

Due to the noisy nature of physical measurements a proper quantization of the analog PUF responses is necessary. The quantization has to be carried out per feature and the generated bit strings are concatenated, creating the entire response of the PUF. The first stage is common to analog PUFs (e.g., coating PUFs [98]) and also to quantizing analog features in the field of biometrics [99].

To accomplish this, the distribution of a feature across all entities is divided into K discrete bins and a bit string of length $N = \log_2(K)$ generated by a Gray code is assigned to each of them (Figure 4.1). This ensures that the Hamming distance between adjacent bins is always one [100]. As a result only one bit-flip occurs when a measurement is shifted into an adjacent bin. This enables the generation of stable binary substrings per feature with a limited fraction of bit-flips when considering noise and other effects such as aging and temperature variations.

Furthermore, the quantization can provide uniformly distributed bit strings by arranging the bins according to an equal probability of occurrence [98, 101]. The width of the most narrow bins depends on the feature stability across all possible operating conditions. There exists a clear trade-off between making the bins narrower in order to increase the number of bits that can be derived per

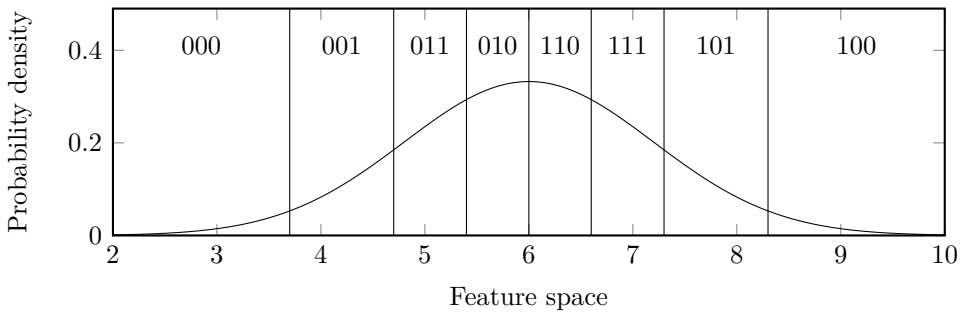


Figure 4.1: Exemplary illustration of the multi-bit quantization scheme for a Gaussian distributed feature.

feature and widening the bin widths to decrease the probability of quantization errors.

Ultimately, the number of derivable bits N per feature X is proportional to the base-2 logarithm of the ratio of the width W_X of the global distribution of X and the feature stability defined by $X'_{i,j} = x_{0,j} - x_{i,j}$, where $x_{0,j}$ is the reference measurement of sensor j and $x_{i,j}$ is a repeated measurement of the same feature and sensor, where $i \geq 1$. Note that the maximum shift $\max(X'_{i,j})$ over all tests is considered as reference for adjusting the bin widths.

The quantization can be optimized by setting the narrowest bin widths depending on the value $\max(X'_{i,j})$ multiplied by an adjustment parameter p_a. Notice that large values of the parameter p_a lead to a reduction of bit-flips while at the same time reducing the number of derivable bits and vice versa. Finally, a fine adjustment of the bin widths has to be performed so that the number of bins K is equal to a power of two and all possible substrings (2^N) occur with the same probability.

In order to arrange the bins according to an equal probability of occurrence, the probability distribution has to be determined for each feature. Generally, manufacturing variations are expected to be normally distributed. However, the actual distribution can deviate from this, e.g., due to a limited sample size or non-uniformities in the manufacturing processes. To ensure that the quantization scheme generates uniformly distributed bit strings and, thus, no entropy reduction occurs, a kernel smoothing technique was used to estimate the features' probability density [102].

In kernel smoothing technique, the probability density function of a random variable is represented in a non-parametric manner enabling to properly

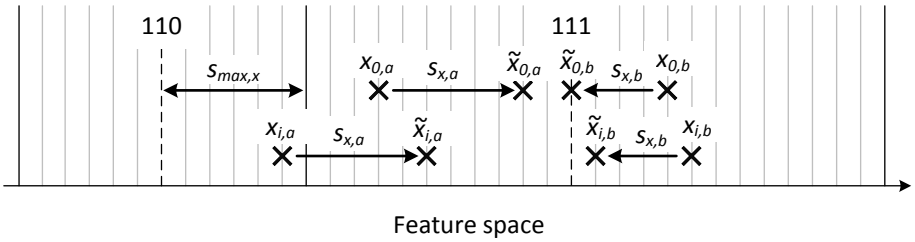


Figure 4.2: Schematic illustration of shift value usage. $x_{0,a}$ and $x_{0,b}$ represent initial measurements of a feature X originating from different sensors a and b. $x_{i,a}$ and $x_{i,b}$ are repeated measurements.

describe the data without making assumptions about the data's distribution. The kernel density is given by

$$\hat{f}_h(z) = \frac{1}{nd} \sum_{i=1}^{n} K\left(\frac{z - z_i}{d}\right), \tag{4.1}$$

where z_i are random samples from an unknown distribution with $i \in [1, n]$, n is the sample size, $K(\cdot)$ is the kernel smoothing function, and d is the bandwidth which determines the smoothness of the probability density curve.

The measurements resolution is significantly higher than the width of the bins (resolution is illustrated by gray vertical lines in Figure 4.2). In order to further reduce the error probability, a fixed shift value $s_{x,j}$ was used per sensor j and feature X. This value was derived from an initial measurement $x_{0,j}$ during an enrollment stage. It was chosen such that an initial feature value $x_{0,j}$ was shifted by $s_{x,j}$ towards the center of the bin in which it lies ($\tilde{x}_{0,j} = x_{0,j} + s_{x,j}$). The value $s_{x,j}$ was then added to all following measurements $x_{i,j}$ of the same feature and sensor ($\tilde{x}_{i,j} = x_{i,j} + s_{x,j}$). Note that the shift value must not be bigger than half the width of the narrowest bin $s_{x,j} \leq s_{max,x}$ (see Figure 4.2). In this case, it does not leak any information about the expected position of a feature within the quantization scheme [103].

4.2 Identifying Suitable Features

As mentioned in Section 4.1, a fundamental measure that determines the suitability of a feature to be used in PUF applications is the ratio of the stability of a feature $\max(X'_{i,j})$, to its variability, expressed by W_X. For the following analysis, we defined $W_X = P_{97.5} - P_{2.5}$ determining the range that covers 95 %

of the measurements of a particular feature, whereas $P_{97.5}$ is the 97.5th and $P_{2.5}$ the 2.5th percentile.

Another important criterion is the correlation between different features which reduces entropy. Most of the properties explained in Section 2.5.3 are dependent on several sources of variation. Hence, some of the properties which are dependent on the same predominant sources are expected to correlate. However, small within-die variations counteract correlation.

The correlation coefficient $\rho_{X,Y}$ between two variables X and Y with N measurement values is given by

$$\rho_{X,Y} = \frac{1}{N-1} \sum_{i=1}^{N} \left(\frac{x_i - \mu_X}{\sigma_X} \right) \left(\frac{y_i - \mu_Y}{\sigma_Y} \right), \tag{4.2}$$

where μ_X and μ_Y are the mean and σ_X and σ_Y are the standard deviations of X and Y. The importance of $\rho_{X,Y}$ becomes obvious if one assumes that two features would correlate with 100 %. As a result, a feature value could be predicted based on the other so that its information is redundant and its contribution to the entropy of the PUF fingerprint is zero.

The ratio $\left| W_X / X'_{i,j} \right|$ was defined as relative variability τ, taking into account that the larger the parameter τ, the more stable bits can be derived from a particular feature. We found that mode amplitudes and quality factors have rather low τ-values. However, it has to be noted that this might be a result of the used measurement method. Since those values are dependent on measured absolute values, other measurement methods (e.g., precise frequency sweeping) might be necessary to determine them with sufficient accuracy. However, this would mean that those features would have to be measured successively making the measurement time-consuming when having a large number of frequency modes. Moreover, we found that resonance frequency tuning properties and electrical resistances have too low τ-values as well, especially if long-term and temperature stability are considered.

Electrical capacitances have a better relative variability than resistances. However, an additional and time-consuming measurement step is necessary in order to measure them with sufficient accuracy (see Section 3.2). Since the number of capacitances is rather small and they correlate with certain frequency modes, they are not expected to contribute much to the uniqueness of the fingerprints. Thus, we decided not to consider them further.

Summarizing the above, the following analysis is based on 13 frequency modes and 3 quadrature signals. The measured frequency modes were in the range between 20 kHz and 180 kHz.

4.2.1 Temperature Dependency

The used features are sensitive to temperature variations. Especially, this applies to the frequency modes since their position is proportional to the root of the Young's modulus which decreases with raising temperature [104]. However, the relationship is approximately linear for the used features so that we apply a linear correction scheme. For this purpose, a global temperature compensation parameter was derived per feature from a small subset of sensors. The same compensation parameters were then used for all sensors[1]. Note that only compensated data are considered in the following evaluation.

4.2.2 Feature Ratios

As mentioned previously, high feature correlation leads to a reduced bit-entropy and, hence, to a reduced security level of the derived keys. The positions of frequency modes are mainly dependent on the structure widths of the springs (see Section 2.5.4). Since the structure width variations across a wafer are significantly higher than those across a single die, the different frequency modes are correlated.

For this reason, we calculated ratios between the different frequency modes moving the focus to their relative position[2]. In this way, correlations are significantly reduced and within-die variations become more important. Additionally, frequency ratios are less sensitive to temperature variations as the temperature dependency of the different frequency modes is roughly similar. However, the compensating effect is not sufficient when considering large temperature variations as we did, so that a linear compensation scheme is still preferably.

For the following evaluation, all possible ratio combinations were used meaning that $\frac{13 \cdot 12}{2}$ ratios were calculated. In principle, the use of all combinations is acceptable because each bit string indicates only that a feature value x_i lies in a certain bin (see Section 4.1). Considering three variables x_1, x_2 and x_3, this means that even knowing the intervals in which the ratios x_1/x_2 and x_1/x_3 lie, there is still some information left in the ratio x_2/x_3 as shown hereafter.

Assuming $x_1 \in [l_1, u_1]$, $x_2 \in [l_2, u_2]$, and $x_3 \in [l_3, u_3]$, then $x_1/x_2 \in [\frac{l_1}{u_2}, \frac{u_1}{l_2}]$, $x_1/x_3 \in [\frac{l_1}{u_3}, \frac{u_1}{l_3}]$, and $x_2/x_3 \in [\frac{l_2}{u_3}, \frac{u_2}{l_3}]$. Based on the knowledge of $[\frac{l_1}{u_2}, \frac{u_1}{l_2}]$ and

[1] This means that no individual compensation parameters would have to be derived for each PUF instance but it is sufficient to determine those parameters on a small subset.

[2] Note that ratios of non-independent normal variables are usually not normally distributed, which has to be considered for a proper quantization of such features (see e.g., [105])

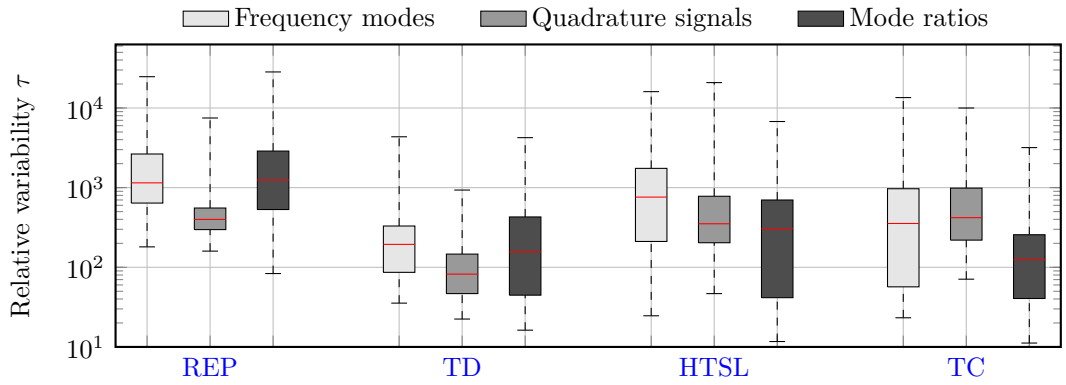


Figure 4.3: Relative variability τ of 13 frequency modes, 3 quadrature signals and 78 mode ratios for the performed test procedures repeatability at RT (REP), temperature dependency (TD), high temperature storage life (HTSL), and temperature cycling (TC).

$[\frac{l_1}{u_3}, \frac{u_1}{l_3}]$ only, x_2/x_3 can just be predicted to be in $[\frac{l_1}{u_1}\frac{l_2}{u_3}, \frac{u_1}{l_1}\frac{u_2}{l_3}]$, where $u_1 > l_1$ and, thus, $\frac{u_1}{l_1}\frac{u_2}{l_3} > \frac{u_2}{l_3}$ and $\frac{l_1}{u_1}\frac{l_2}{u_3} < \frac{l_2}{u_3}$.

4.2.3 Relative Variability Results

Figure 4.3 shows the relative variability τ of frequency modes, quadrature signals, and ratios of the frequency modes for the different test procedures, explained in Section 3.5. The widths W_X of the global distributions is based on module-level measurements of 468 sensors (baseline measurements). The feature stability was calculated from $X'_{i,j}$ for each measurement within a particular test procedure. The boxes define the range that covers 50 % of the data (P_{25} and P_{75}) and the median is given by the horizontal red line within a particular box. The whiskers cover 95 % of the data ($P_{2.5}$ and $P_{97.5}$).

As expected, the relative variability of frequency modes and mode ratios tend to decrease considering temperature variation and aging. However, the majority of τ-values still lies in a usable range. The relative variability of quadrature signals significantly decreases only in the TD test. Note that one reason for the partly high variation of the parameter τ is the fact that several features were evaluated together having different SNRs and sensitivities with respect to particular test conditions.

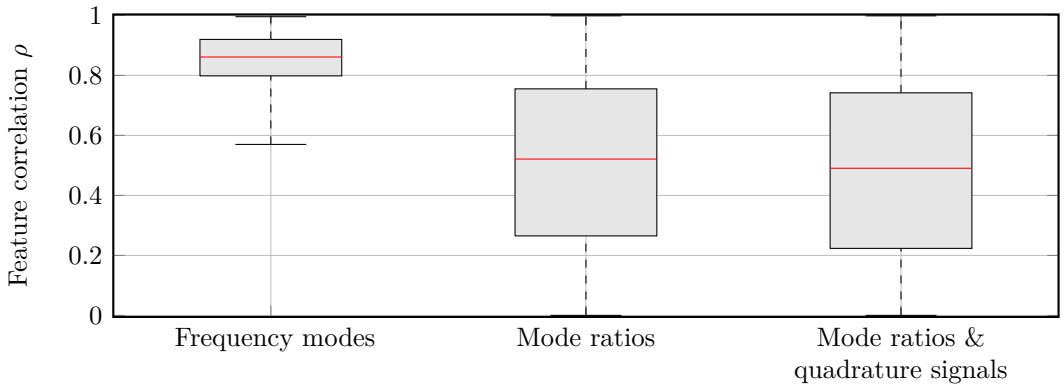


Figure 4.4: Feature correlation ρ considering 13 frequency modes only, their ratios and the ratios evaluated together with 3 quadrature signals.

4.2.4 Feature Correlation Results

In Figure 4.4, the distribution of the correlation coefficients (absolute values) between the features is shown for considering frequency modes, mode ratios, and mode ratios together with quadrature signals. The boxes indicate the lower and upper quartile (the median is given by the red line) while the whiskers show the minimum and the maximum of the distribution. It can be seen that in the case of mode ratios, the average feature correlation is significantly reduced compared to frequency modes. Quadrature signals are less correlated with each other and with frequency modes. Hence, the average feature correlation is further reduced when quadrature signals are also considered.

While using feature ratios has been shown to be beneficial, the information that is added by using all possible ratio combinations will tend towards zero at some point. Hence, using all possible ratios might cause an increase of redundant information reducing the entropy of the PUF response. Thus, this might not lead to an optimal result although the response length of the PUF can be significantly increased in this way. Hence, an upper limit ρ_{max} for acceptable correlations was introduced. Features which show a stronger correlation than that were rejected.

4.2.5 Feature Selection

As a result of the previous analysis, we decided to use frequency mode ratios and quadrature signals for the following evaluation. Absolute positions of frequency modes were not considered any further due to the high average

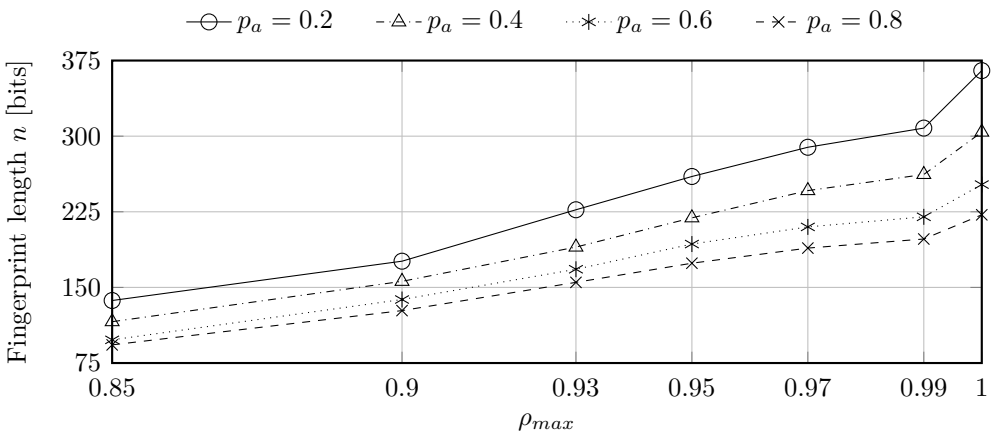


Figure 4.5: Dependence of fingerprint length n on the choice of the correlation upper limit ρ_{max} and the adjustment parameter p_a.

correlation between them. Moreover, their information is contained in mode ratios. Besides, mode ratios provide the advantage that minor variation in temperature (variation in room temperature) can be compensated. As mentioned previously, a correlation upper limit ρ_{max} was used in order to reject features that are strongly correlated.

4.3 Setting Values for Parameters p_a and ρ_{max}

As mentioned in Section 4.1, the narrowest bins of the quantization scheme are dependent on the feature stability $\max(X'_{i,j})$ and a fine adjustment can be made by multiplying $\max(X'_{i,j})$ with an adjustment parameter p_a. Additionally, an appropriate value for the used correlation upper limit ρ_{max} has to be determined. In the following, the dependence of the fingerprint length, number of bit-flips, and entropy of the fingerprints on those two parameters is evaluated.

4.3.1 Influence on Fingerprint Length

Figure 4.5 shows the length n of the derived fingerprints depending on the parameters ρ_{max} and p_a. As expected, n increases with an increasing ρ_{max}-value since more features are used in this case. Furthermore, it can be seen that with a decreasing p_a-value the fingerprint length increases as well since more bits are derived per feature.

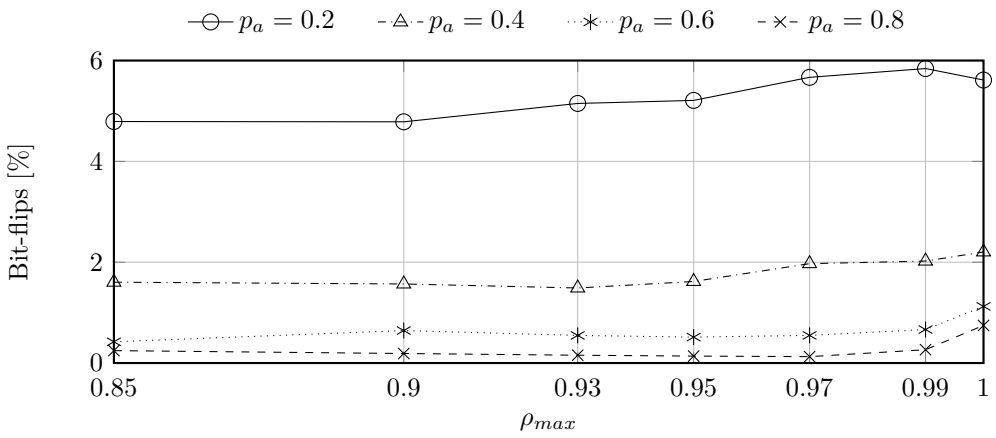


Figure 4.6: Average number of bit flips in the derived fingerprints relative to the fingerprint length n depending on the correlation upper limit ρ_{max} and the adjustment parameter p_a.

4.3.2 Influence on Bit-Flip Probability

The mean number of bit flips in relation to the fingerprint length n is shown in Figure 4.6. The values are averaged over the performed robustness tests TD, HTSL, and TC. Due to larger bin widths, the probability for bit-flips decreases with increasing p_a-values. Besides, it can be seen that the bit-flip probability slightly increases with an increasing ρ_{max}-parameter. The reason for this is the way in which features that are stronger correlated than ρ_{max} are rejected: if the correlation between two features is too high, the feature with the worse relative variability is rejected. Hence, more and more features with a low relative variability are used with increasing ρ_{max}-values.

4.3.3 Influence on Entropy

As mentioned in Section 2.3, high correlations between features reduce the entropy of the fingerprints. In Figure 4.7, $H_\infty(X)$ is estimated by determining the MCB in the concatenated fingerprints (from the baseline measurements). As expected, $H_\infty(X)$ tends to decline for higher ρ_{max}-values. However, the decrease is not monotone. At this point, we want to emphasize that $H_\infty(X)$ of a source can just be estimated from a sample with limited size. The estimate depends on more factors such as the nature of a particular test, the size of the substrings and the ordering of the features. A refined analysis on that is made in Section 5.1.

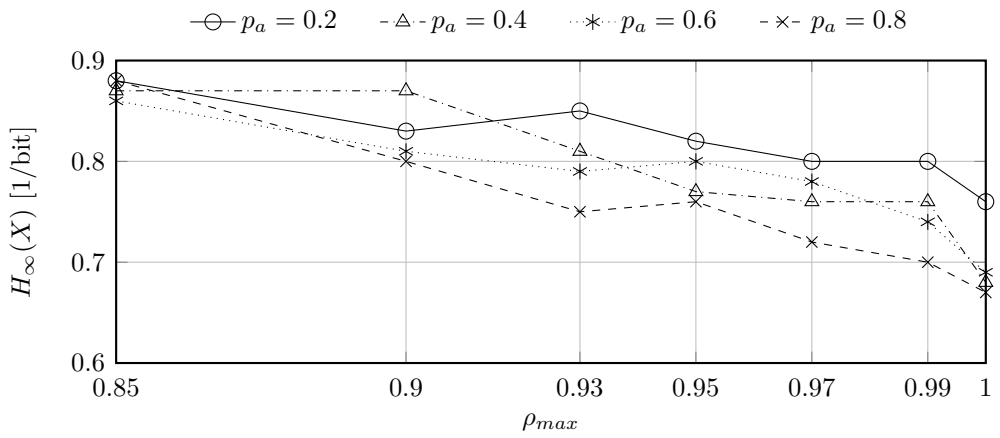


Figure 4.7: Min-entropy per bit depending on the correlation upper limit ρ_{max} and the adjustment parameter p_a estimated with the most common byte method.

Additionally, Figure 4.7 shows that $H_\infty(X)$ tends to be higher for smaller p_a-values. The reason for this is that the influence of measurement noise increases for smaller bin widths.

4.3.4 Parameter Definition

For the following analysis, the parameters ρ_{max} and p_a were fixed. We set $\rho_{max} = 0.95$ and $p_a = 0.6$ as a result of preliminary investigations regarding the residual min-entropy $\tilde{m}$. In this case 50 frequency ratios based on 13 frequency modes and 3 quadrature signals are used. It has to be noted that finding an optimum value for $\tilde{m}$ is non-trivial since $\tilde{m}$ depends on the used entropy estimation method and on the assumptions that are made on the amount of entropy that is leaked by the helper data (see Equation (2.6)).

4.4 Hamming Distance Distributions

A prerequisite for a physical system to be used in PUF applications is that each physical instance can be identified uniquely. As discussed in Section 2.3.1, this can be evaluated through the concept of inter and intra Hamming distances. In the following, we discuss the modeling of $\mathrm{HD}_{\mathrm{intra}}$ and $\mathrm{HD}_{\mathrm{inter}}$ distributions by which we determined the FRR and FAR.

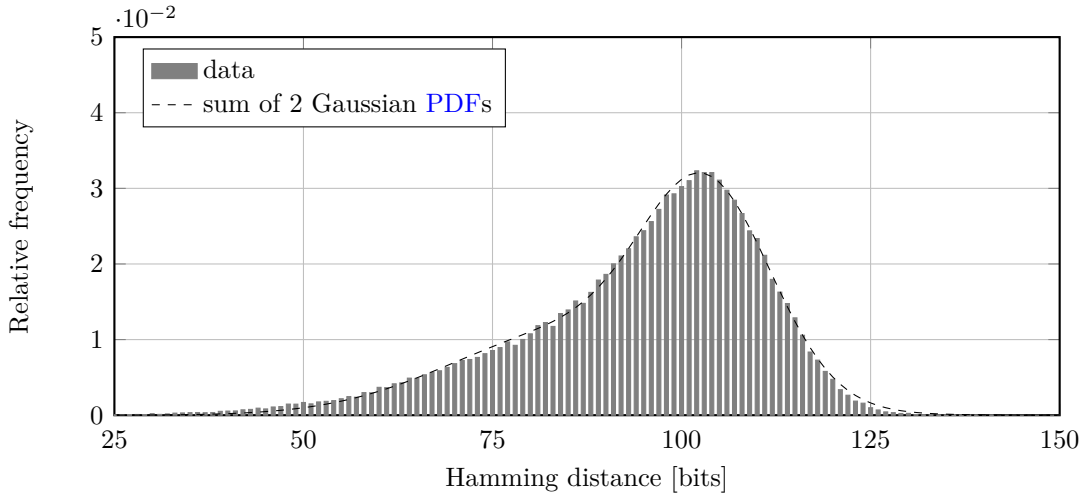


Figure 4.8: Inter Hamming distance distribution of baseline measurements with sum of two Gaussian PDFs (fit parameters for Equation (4.3): $a_1 = 0.026$, $b_1 = 103$, $c_1 = 12$, $a_2 = 0.011$, $b_2 = 86$, $c_2 = 23$).

4.4.1 Modeling of Inter Hamming Distance Distribution

Figure 4.8 shows the $\mathrm{HD}_{\mathrm{inter}}$ distribution based on baseline measurements. The fingerprint length n is 189 bits. As mentioned in Section 2.3.1, given two bit strings with independent and identically distributed bits, $\mathrm{HD}_{\mathrm{inter}}$ distribution is expected to follow a binomial distribution with $p = 0.5$ and, thus, it can be approximated by a Gaussian probability density function (PDF). However, depending on the used quantization scheme and the nature of the data, this approximation is not necessarily accurate. Note that in the given quantization scheme (as seen in Figure 4.1) the bins are narrower in the middle of the distribution than at the edges. As a given feature value approaches either edge of the quantization scheme, the expected Hamming distance to another correlated feature decreases due to the increase in the surrounding bin width.

While the Hamming distances between responses with low or high average feature values are reduced due to this phenomenon, the Hamming distances between responses with medium feature averages are not strongly affected. As such, the combined distribution appears multi-modal in nature, and has to be modeled with a sum of Gaussian curves. For the used data, we found that the sum of two Gaussian PDFs

$$P(x) = a_1 e^{-(x-b_1)^2/2c_1^2} + a_2 e^{-(x-b_2)^2/2c_2^2} \tag{4.3}$$

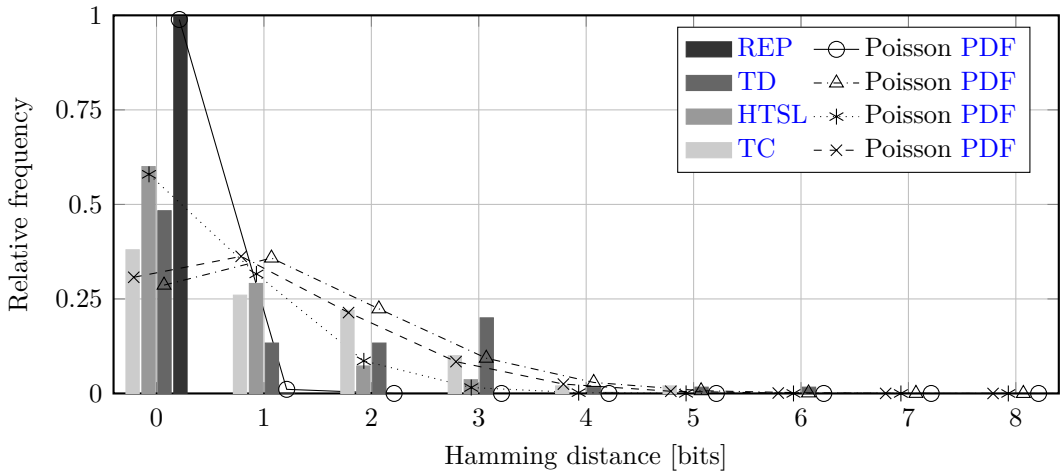


Figure 4.9: Intra Hamming distance distributions with fitted Poisson PDF of the performed robustness tests repeatability at RT (REP) (average success probability $\lambda = 0.011$), temperature dependency (TD) ($\lambda = 1.25$), high temperature storage life (HTSL) ($\lambda = 0.55$), and temperature cycling (TC) ($\lambda = 1.18$).

provides already an accurate fit.

4.4.2 Modeling of Intra Hamming Distance Distribution

When comparing two bit strings w and w' derived from the same instance, each bit that is compared can be considered as its own Bernoulli trial, with a success denoted by a bit-flip. Due to the nature of the quantization scheme each Bernoulli trial has its own probability of success p_i. Hence, the $\mathrm{HD}_{\mathrm{intra}}$ distribution follows the Poisson binomial distribution. Since p_i values are rather small and $n > 100$, we can utilize the Poisson PDF

$$P(x) = e^{-\lambda}\lambda^x/x!, \tag{4.4}$$

as a good approximation of the Poisson binomial as described in [106] with an average success probability λ.

Figure 4.9 shows the $\mathrm{HD}_{\mathrm{intra}}$ distributions with corresponding Poisson fit for the performed robustness tests. Note that each sensor has individual values for p_i because p_i is dependent on the position of a feature value within the quantization scheme. This means that λ is not necessarily the same for each sensor.

Table 4.1: Results for false rejection rate (FRR) and false acceptance rate (FAR) at the equal error rate (EER) point for the performed robustness tests REP, TD, HTSL, and TC.

	REP	TD	HTSL	TC
n [bits]	189	189	189	189
t [bits]	3	9	7	9
FRR	$6.1 \cdot 10^{-10}$	$8.3 \cdot 10^{-7}$	$1.2 \cdot 10^{-7}$	$5.0 \cdot 10^{-7}$
FAR	$6.8 \cdot 10^{-8}$	$5.8 \cdot 10^{-7}$	$3.0 \cdot 10^{-7}$	$5.8 \cdot 10^{-7}$

However, the sum of several Poisson distributions is still Poisson distributed so that this approach remains valid.

4.4.3 Equal Error Rate

The exact values of FRR and FAR are dependent on the choice of the threshold t. Choosing a high t value increases the FAR while decreasing the FRR and vice versa. For the sake of comparability, t is often set such that FRR and FAR are equal (under the restriction that $t \in \mathbb{N}$). This point is also known as equal error rate (EER).

All values of FRR and FAR for the performed tests are below 1 ppm (Table 4.1). The highest error rate results from the TD test which corresponds to the relative variability results in Section 4.2.3. Note that the bin widths of the quantization scheme were not optimized for each individual test but the same quantization scheme was used in all cases.

4.5 Impact of Inter-Wafer Variations

So far, the analysis was based on sensors from a single wafer. In this section, we show that considering a single wafer can be seen as a worst case scenario in terms of the number of derivable bits and the statistical probability for an authentication error as a result of the reduced feature variation and spatial correlations between sensors from the same wafer. To this end, wafer-level measurements were carried out on more than seven thousand sensors from four wafers which were produced in two different batches.

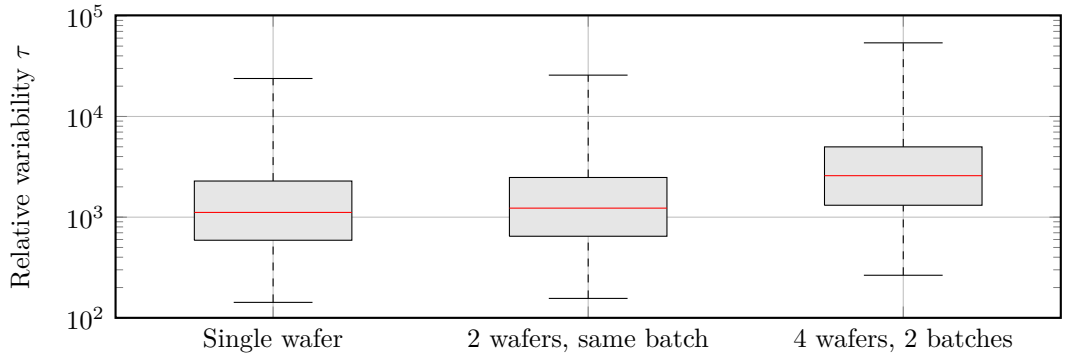


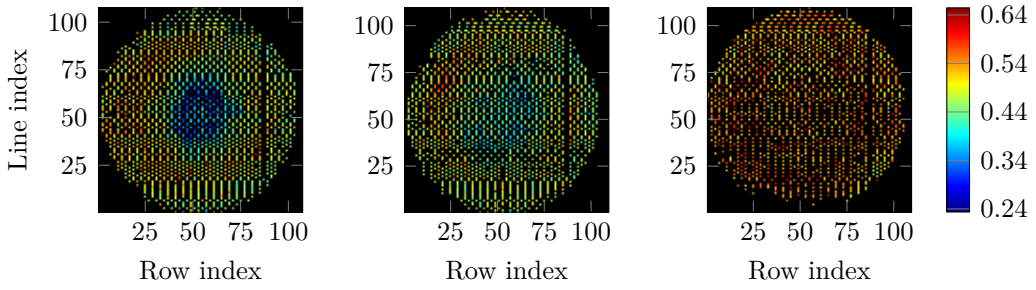
Figure 4.10: Relative feature variability τ for a single wafer, two wafers from the same batch, and four wafers out of two different batches.

It has to be noted that the results cannot be compared directly to baseline measurements on module-level due to the higher noise floor of wafer-level measurements. As a result fewer frequency modes could be reliably measured leading to a reduced fingerprint length. Moreover, the feature stability is slightly reduced and the feature variability is slightly increased by noise. However, the results show a clear trend since these effects apply to both the measurement of a single wafer and the measurement of several wafers in the same way.

4.5.1 Impact on Relative Variability

Figure 4.10 shows the distribution of the relative feature variability τ measured on a single wafer, two wafers from the same batch, and four wafers out of two different batches. The relative variability τ is determined and presented as in Section 4.2.3. Frequency mode ratios and quadrature signals are evaluated together and the feature stability is based on the repeatability of the measurements at room temperature.

As expected, the distribution of τ-values significantly increases when considering four wafers from two different batches due to the higher level of within-batch and batch-to-batch variations. This leads to an increase in the length n of the derived fingerprints. When initializing the quantizations scheme individually for the three mentioned cases based on the respective W_X-values, the fingerprint length increases by 15 % from 172 bits for a single wafer to 197 bits considering all measured wafers (177 bits when considering two wafers from the same batch).



(a) Reference wafer. (b) Wafer from the same batch. (c) Wafer from an other batch.

Figure 4.11: Fractional Hamming distances of the MEMS fingerprints to a reference sensor (row index=63, line index=61) on the left wafer.

4.5.2 Impact on Error Rate

For the following evaluation a fixed quantization scheme was used based on the W_X-values considering all four measured wafers. As mentioned previously, the sensor's fingerprints are not fully independent. The reason for this is systematic non-uniformities of within-wafer variations leading to spatial correlations. As a result, sensors which are close together on a wafer, tend to have a reduced Hamming distance relatively to the average (Figure 4.11a). This still applies in weakened form when comparing wafers from the same batch (Figure 4.11b). However, spatial correlations disappear when comparing wafers from different batches (Figure 4.11c).

This local dependency leads to the multimodal nature of the $\mathrm{HD}_{\mathrm{inter}}$ distribution. We show this by separating the $\mathrm{HD}_{\mathrm{inter}}$ distribution into three different sub-distributions:

- *Within-wafer*. Hamming distances between sensors from the same wafer.
- *Within-batch*. Hamming distances between sensors from different wafers but from the same batch.
- *Batch-to-batch*. Hamming distances between sensors from different wafers out of different batches.

In Figure 4.12, it becomes clear that considering wafers from different batches shifts the sum distribution to the right as a result of the increased Hamming

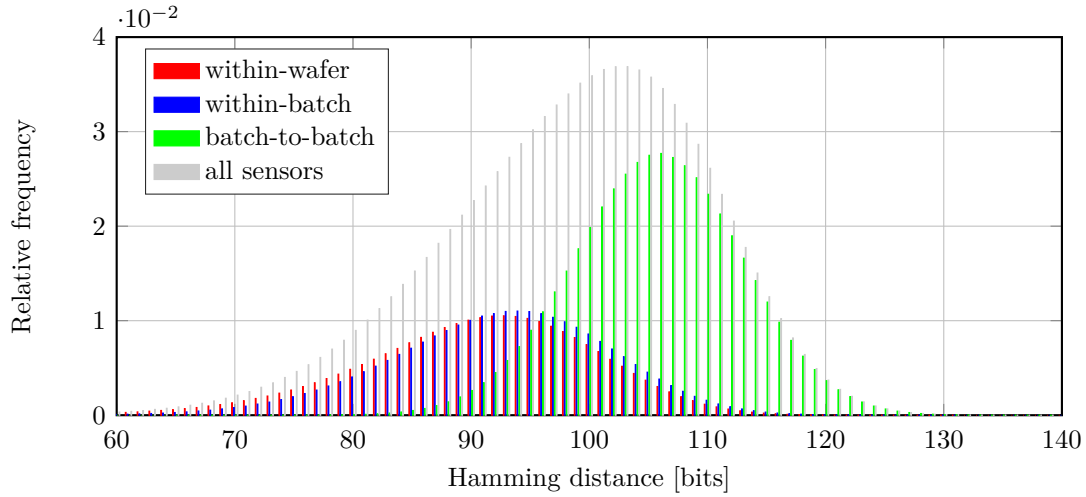
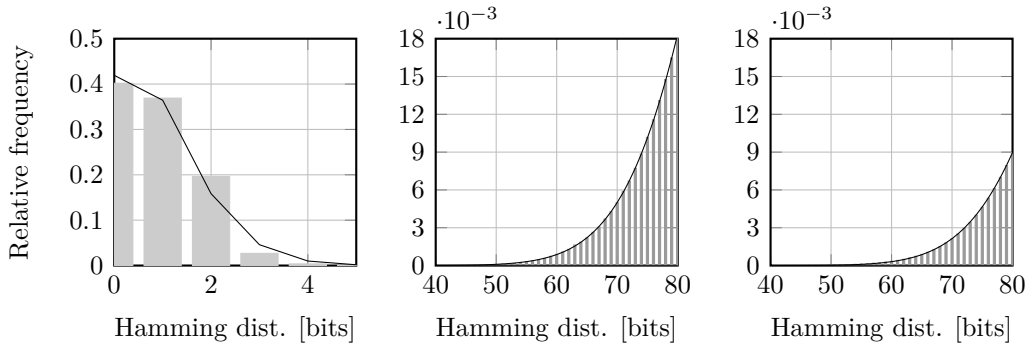


Figure 4.12: Inter Hamming distance distributions for the cases *within-wafer*, *within-batch*, *batch-to-batch*, and the sum of them ($n = 197$).

distances between sensors out of different batches. This also affects significantly the FAR which can be seen in Figure 4.13. The proportion of reduced Hamming distances significantly decreases when considering all measured wafers (Figure 4.13c) compared to a single wafer (Figure 4.13b). For a fixed threshold, e.g., $t = 12$ this means that the FAR is decreased by almost two orders of magnitude from $1.15 \cdot 10^{-9}$ to $7.22 \cdot 10^{-11}$ ($1.59 \cdot 10^{-10}$ when considering two wafers from the same batch). Note that the $\mathrm{HD}_{\mathrm{intra}}$ distribution (Figure 4.13a) is not dependent on the examined cases since the feature stability remains constant. The results are summarized in Table 4.2.

Based on this evaluation, it can be concluded that the influence of spatial correlations would be further reduced when considering multiple batches as shown hereafter. Assuming the number of sensors per wafer to be n_s, the number of wafers within a batch to be n_w, and the number of batches to be n_b, then the number of inter Hamming distances suffering from spatial correlations (distances between sensors within wafers and batches) increases linearly with the number of batches $n_b \left(n_w \frac{n_s(n_s-1)}{2} + n_s^2 \frac{n_w(n_w-1)}{2} \right)$. On the other hand, the number of Hamming distances between sensors from wafers corresponding to different batches is $\frac{n_b(n_b-1)}{2}(n_s^2 \cdot n_w^2)$ and, hence, it increases proportionally to the square of the number of batches. Since n_s and n_w are limited by the wafer size and the batch size of the manufacturing process, a large number of units,



(a) HD_{intra} distribution.

(b) HD_{inter} distribution for a single wafer.

(c) HD_{inter} distribution of all measured sensors.

Figure 4.13: Intra Hamming distance distribution with Poisson fit ($\lambda = 0.87$) and inter Hamming distance distributions with a Gaussian fit of the left tail.

which is typical for MEMS applications, can just be manufactured by producing multiple batches in practice.

Table 4.2: Results for false rejection rate (FRR) and false acceptance rate (FAR) for considering a single wafer, two wafers from the same batch, and four wafers from two different batches.

	Single wafer	2 wafers, same batch	4 wafers, 2 batches
n [bits]	172	177	197
t [bits]	12	12	12
FRR	$1.16 \cdot 10^{-11}$	$1.16 \cdot 10^{-11}$	$1.16 \cdot 10^{-11}$
FAR	$1.15 \cdot 10^{-9}$	$1.59 \cdot 10^{-10}$	$7.22 \cdot 10^{-11}$

Chapter 5

Key Extraction

In this chapter, the number of stable bits with nearly full entropy that can be extracted from the MEMS fingerprints is estimated. Initially, a detailed analysis of the entropy inherent in the MEMS fingerprints is provided. Afterwards, the error correction using the code offset construction with BCH codes and the entropy leakage due to the helper data are discussed. In addition, the probabilities that the error correction might fail are determined. Finally, randomness extraction results are provided and the security level of the generated keys is discussed.

5.1 Min-Entropy Estimation

For entropy estimation, all fingerprints, which were derived from the baseline measurements, were concatenated and analyzed as one long bit string. The entropy estimation tests can be performed bit-wise or on larger bit blocks. For the results shown in this section, bit-wise and a byte-wise evaluation was carried out.

We performed the HW, the MCB, the CTW compression, and the NIST 800-90B min-entropy estimation tests. The Daugman method was not considered since it is not clear how the test could by applied to a multimodal and asymmetric $\mathrm{HD}_{\mathrm{inter}}$ distribution. Furthermore, the impact of ordering the features within the responses on the test results was examined. Since the entropy was estimated on the concatenated fingerprints, we also considered the effect of ordering the fingerprints within the entire bit string. Additionally, we verified the results by performing a Monte-Carlo simulation in order to make sure that the test results were not significantly influenced by the limited size of the measured data.

5.1.1 Estimation Results for Measured Data

Entropy estimation results for measured data are based on the responses derived from baseline measurements (468 sensors). Since the fingerprint length n is 189 bits, the size of the input string on which the tests were performed is about 88.5 kbit. Note that the entropy estimation results are given per bit.

The derived responses consist of a concatenation of individual substrings derived from the sensors' properties. We assume the order of the substrings to be the same for all sensors, but the order itself to be arbitrary. Also the order of the fingerprints within the concatenated bit string, which is the input to the entropy tests, is arbitrary. To examine if a particular ordering affects the test results, features in the responses were randomly rearranged and the order of the fingerprints within the entire bit string was varied. Random ordering of features and responses was carried out and the entropy was estimated in 100 test runs. The distributions of the results are shown in Figure 5.1 for the different tests.

We found that rearranging the features based on their correlations with one another leads to lower entropy estimates. Ordering the features in that manner was achieved by rearranging features so that the first diagonal of the feature correlation matrix was maximized. Regarding the positioning of the responses, lowest entropy estimates result from ordering the derived responses according to the original position of the sensors on the wafer. The ordering was carried out so that responses of sensors which were adjacent on the wafer are close to each other within the input string. Thus, a combination of ordering the substrings after correlations and ordering the responses within the entire input string after sensors' original position on the wafer led to the lowest entropy estimation results. It should be noted that the effect of ordering the features is higher than that of rearranging the entire responses within the input string.

Hamming weight

The estimated min-entropy based on the HW test is 1.0 independent from ordering. This result means that the number of ones equals the number of zeros indicating that the derived responses do not suffer from bias. It has to be noted that correlations cannot be sufficiently analyzed in this way.

For a byte-wise evaluation (MCB), the estimated min-entropy is reduced (Figure 5.1). When maximizing adjacent correlations, there is a noticeable degradation in min-entropy to 0.76. The reason for this is that this method is sensitive to correlations between adjacent substrings of smaller size than the examined block size. Note that correlations can be better analyzed, the longer

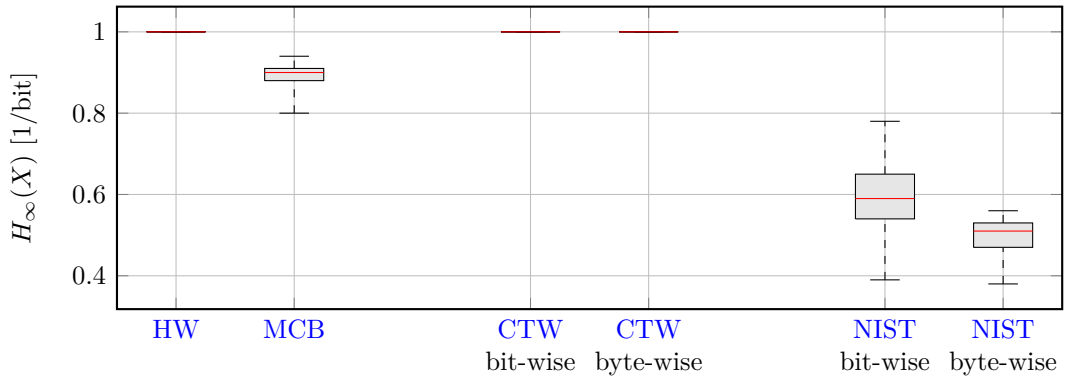


Figure 5.1: Bit-wise and byte-wise entropy estimation results for random ordering of features and responses. The boxes indicate the lower and upper quartile while the whiskers show the minimum and the maximum of the test results. The median is given by the red line. For HW and CTW tests, all estimates are close to 1.0 so that whiskers and boxes are not visible with this scale.

the examined block size is in relation to the length of the substrings. We performed tests with a maximum block size of 8 bits. Using larger block sizes require too much computational power and time.

CTW compression

CTW compression test on measured data shows an overall incompressibility of the responses for bit-wise and byte-wise evaluation largely independent from rearranging the features and responses. Note that Krichevski-Trofimov as well as the zero-redundancy estimator and different tree depths were used (maximum tree depth = 12) [59]. For maximizing adjacent correlations and rearranging responses according to sensors' wafer position, the test indicates an entropy of 0.99 for bit-wise and 1.0 for byte-wise evaluation.

NIST 800-90B min-entropy estimation

As expected, the lowest estimates result from NIST tests. Table 5.1 shows the min-entropy estimation results from the different tests for a bit-wise and byte-wise evaluation in the case where adjacent correlations are maximized and responses are ordered according to sensors' original position on the wafer.

For bit-wise evaluation, the lowest results are given by the t-tuple and LRS estimates. As mentioned in Section 2.3.1, these tests are sensitive to repeated substrings within a given input string. If the features are ordered such that

Table 5.1: Min-entropy estimates per bit from tests contained in NIST's special publication 800-90B.

Estimators	bit-wise	byte-wise
Most common value	0.99	0.73
Collision	0.78	n.a.[1]
Markov	0.99	0.38
Compression	0.84	0.51
t-tuple	0.48	0.76
LRS	0.43	0.44
Multi MCW Prediction	0.70	0.63
Lag Prediction	0.81	0.42
Multiple MMC Prediction	0.72	0.63
LZ78Y Prediction	0.72	0.63

correlated features are adjacent, longer substrings based on several features can occur with higher probability than others within a subsection of the responses.

When evaluating the concatenated responses byte-wise, the lowest min-entropy estimate results from the Markov estimate which determines the dependency of a sample on the previous samples. Since the number of considered samples is limited and the quantization scheme outputs approximately uniformly distributed substrings per feature, there are almost no increased dependencies when evaluating the input string bit-wise. However, when evaluating the responses byte-wise, the Markov estimate efficiently recognizes dependencies between adjacent correlated substrings.

The min-entropy estimation results from random ordering of features and responses are subjected to a relatively high variability for bit-wise as well as byte-wise evaluation (see Figure 5.1). Hence, it is just possible to define a range in which the actual min-entropy resides. For the case in which features and responses are systematically ordered, the minimum estimation result is 0.43 for bit-wise and 0.38 for byte-wise evaluation. The overall minimum received during random ordering of features and responses is 0.39 for bit-wise and 0.38 for byte-wise evaluation.

[1] Collision estimate is not working for this alphabet size.

As mentioned in Section 2.3.5, tests of NIST 800-90B assume that a used noise source generates fixed-length bit strings and, if several noise sources are used, that they are independent. In case of evaluating concatenated PUF responses, the definition of a noise source is not entirely clear. It might be possible to define each PUF instance as a separate noise source but also different features, from which substrings within a PUF response are derived, might be seen as individual noise sources. Regardless if particular substrings or an entire PUF response is considered as a noise source, to meet the requirement of independence is hardly possible for silicon-based PUFs in general because spatial dependencies are inherent in virtually every silicon manufacturing process.

In the case of MEMS PUFs, substrings derived from different features are not of a fixed length. Hence, dependencies between features and responses and the varying length of derived substrings may be the reason for a partly high variation of the min-entropy estimation results of NIST 800-90B tests. However, to the best of our knowledge and as mentioned in Section 2.3.5, using tests of NIST 800-90B is currently the most conservative estimation method for PUF responses.

As discussed in Section 2.3.5, tests from NIST 800-90B provide a conservative lower bound on min-entropy. As indicated by the test results obtained from evaluating true random files, the min-entropy estimates from NIST 800-90B tests might be overly pessimistic for practical applications. For this reason and for the sake of comparability to other works, which did not consider tests from NIST 800-90B, the security level of the derived keys is determined for several entropy estimates in Section 5.3. In particular, the lowest results obtained from CTW compression, MCB, and NIST 800-90B are considered.

5.1.2 Estimation Results for Simulated Data

As mentioned, the size of the measured data is about 88.5 kbit. In order to validate the results received on measured data, a Monte-Carlo simulation was performed. The simulation allows to test if the results of the entropy estimation are affected from the limited size of measured data. For the simulation we assumed that the frequency modes exhibit a normal distribution and quadrature values exhibit a half-normal distribution. The Monte-Carlo simulation was carried out by using the Cholesky decomposition of the measured features' correlation matrix C and the measured feature means (μ_j) and standard deviations (σ_j). In particular, the procedure is as follows:

Table 5.2: Entropy estimates per bit on simulated data (80 Mbit) of different tests for bit-wise and byte-wise evaluation.

	HW	MCB	CTW	NIST[2]
bit-wise	0.99		0.99	0.32
byte-wise		0.75	0.96	0.37

- generation of a normally distributed random number matrix $R_{i,j}$, where i is the number of simulated responses and j is the number of features,
- Cholesky decomposition of the correlation matrix $C = GG^T$,
- multiplying matrix $R_{i,j}$ with G to receive the normally distributed random number matrix $R(C)_{i,j}$ considering measured correlations $R(C)_{i,j} = RG$,
- generation of matrix $S_{i,j}$ containing simulated features, where $S_{i,j} = \mu_j + \sigma_j R(C)_{i,j}$ for frequency modes, and $S_{i,j} = abs(\sigma_j R(C)_{i,j})$ for quadrature signals.

In this way about 80 Mbit of data were simulated. This makes it also possible to fulfill the NIST recommendation of providing one million bits for estimation and performing a sanity check of the initial estimate on regenerated data [63]. Entropy estimation was performed bit-wise as well as byte-wise and the features were rearranged such that adjacent correlations are maximized.

As it can be seen from Table 5.2, the entropy estimates based on simulated data are just marginally smaller than those obtained on measured data. Hence, we can assume that the estimates on measured data are not significantly affected by the limited data size.

5.1.3 Comparison to Entropy Rates of SRAM PUFs

Table 5.3 shows a comparison of the entropy estimates on measured data to entropy results of SRAM PUFs given in literature. It can be seen that the results obtained on measured MEMS PUF data are significantly better than that for SRAM PUFs regarding HW and MCB estimation methods. The reason for this is that SRAM PUFs are usually suffering from bias which can be reduced by debiasing methods as proposed, e.g., in [107]. As it can be seen on the HW

[2]Results were verified by the restart test on regenerated data.

Table 5.3: Entropy estimates per bit for HW, MCB, CTW, and NIST testing methods for measured data in comparison to results of SRAM PUFs from literature.

PUF	HW	MCB	CTW	NIST
SRAM	0.34-0.46 [50] 0.75 [55] 0.04-0.06 [56]	0.05-0.15 (0.26[3]) [108] 0.03 [54]	1.0 [55] 0.98-1.0 [109]	n.a.
MEMS	1.0	0.76-0.94	0.99-1.0	0.38-0.78

test result, the derived MEMS PUF responses are largely bias-free due to the optimized quantization scheme.

Unfortunately, no results of NIST 800-90B min-entropy estimation tests are known for SRAM PUFs. Note that SRAM PUFs are often assumed to be free of correlations relying on the randomness of dopant fluctuations in CMOS fabrication processes [54]. Other possibly less uniform effects on threshold variability are neglected, e.g., oxide thickness variations and line edge roughness [84].

5.2 Error Correction

For the error correction part of the fuzzy extractor, we used the code offset construction in order to correct all bit-flips occurred in the PUF responses (see Section 2.4.2). The error correction was implemented in MATLAB. In particular, the PUF responses were divided into three blocks of 63-bits length.

When considering repeatability at RT (REP) only, all bit-flips that occurred could be corrected using a $[n = 63, k = 51, t = 2]$-BCH code which can correct up to two bit-flips per 63-bit block. To be robust against major temperature variation (TD) and aging (HTSL, TD) as well, the use of a $[n = 63, k = 39, t = 4]$-BCH code was necessary.

5.2.1 Error Modeling

In this section, the probability that error correction might fail is determined for each test procedure. We found that at most one bit-flip occurs per substring as a result of the used quantization scheme. Hence, we determined the probability

[3]Improvement due to conditioning component: XORing of adjacent bytes.

that one bit might flip for each substring and assumed the probability to be negligible that more than one bit-flip occurs.

The probabilities were determined for each test procedure separately by averaging the number of bit-flips per substring across all measurements performed within a respective test. For the case that a substring overlaps between two blocks, we assumed this substring as two separate ones. The probability per block Pr_b that at most t bit-flips occur can be calculated with the probability mass function of the Poisson binomial distribution

$$\mathsf{Pr}_b(X \leq t) = \sum_{l=0}^{t} \sum_{A \in F_l} \prod_{i \in A} p_i \prod_{j \in A^c} (1 - p_j), \tag{5.1}$$

where X is the number of bit-flips in n bits, F_l contains all subsets of size l that can be selected from $\{1, 2, ..., n\}$ and A^c is the complement of A. The overall error rate $\mathsf{ER}_{\mathsf{ec}}$ for the error correction process is then

$$\mathsf{ER}_{\mathsf{ec}} = 1 - \prod_{b=1}^{3} \mathsf{Pr}_b(X \leq t). \tag{5.2}$$

The resulting error probabilities are summarized in Table 5.5. The probabilities are at most in the lower ppm range which is usually acceptable for consumer applications. The highest probability of $4.2 \cdot 10^{-6}$ was obtained for the REP test when using a $[n = 63, k = 51, t = 2]$-BCH code. However, when using a $[n = 63, k = 39, t = 4]$-BCH code for this test as well, the error probability decreases to $1.1 \cdot 10^{-16}$.

Regarding the robustness tests, the highest error probability of $4.0 \cdot 10^{-6}$ was obtained for the TC test even if a higher FRR was determined for the TD test in Section 4.4.3. The reason for this is the distribution of bits with a higher error probability across the three 63-bit blocks. Note that it is beneficial if those bits are rather equally distributed across the different blocks.

5.2.2 Residual Min-Entropy

As discussed in Section 2.4.2, entropy loss occurs due to the helper data needed for the error correction. This entropy loss is limited by the conservative $(n-k)$-bound. Since this upper bound is overly pessimistic for non-IID data, the entropy loss was estimated using a method introduced by Delvaux *et al.* [71].

We used algorithms *BoundWorstCase2* and *BoundBestCase2* [71, Algorithms 3 and 4] which apply to linear codes in order to calculate upper and lower bounds

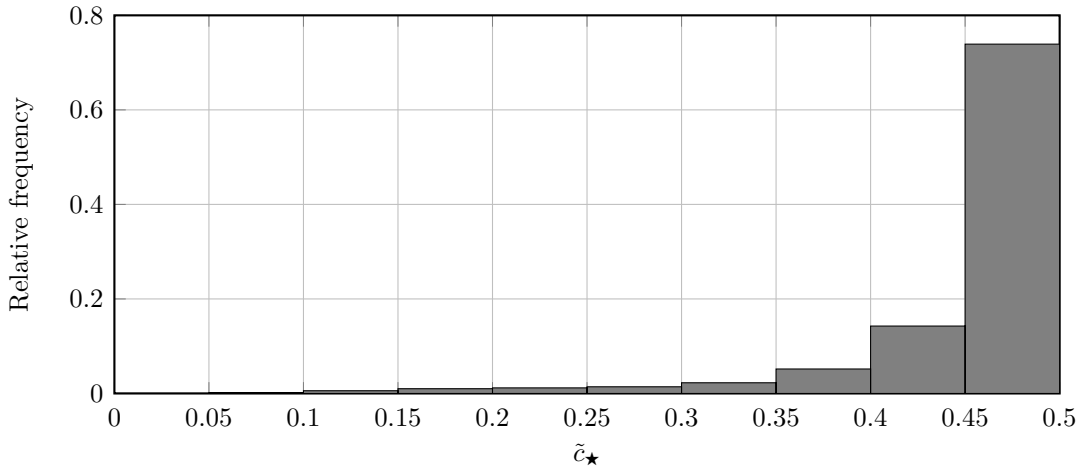


Figure 5.2: Distribution of correlations $\tilde{c}_\star$ between bits in measured responses.

on the residual min-entropy $\tilde{m}$. The algorithms are derived for a constant correlation parameter c, whereas $c = \mathsf{Pr}(X_i = X_{i+1})$ with $i \in [1, n-1]$. For a constant parameter c, the associated min-entropy m is $-log_2(\frac{1}{2}(1-c_\star)^{n-1})$, with $c_\star = \min(c, 1-c)$. Note that $c = \frac{1}{2}$ corresponds to a uniform distribution.

Since the measured distribution does not have a constant correlation parameter, we calculated corresponding $c_\star$-values from the min-entropy estimates m in Section 5.1

$$c_\star = 1 - \sqrt[n-1]{2 \cdot 2^{-m}}. \tag{5.3}$$

The resulting lower and upper bounds of the residual min-entropy $\tilde{m}$ are given in Table 5.5. The absolute best case estimate of $\tilde{m}$ is 150 bits for REP and 114 for TD, HTSL and TC while the absolute worst case is 48 bits for REP and 27 for TD, HTSL and TC.

Additionally, the actual correlations $\tilde{c}_{i,j} = \mathsf{Pr}(X_i = X_j)$ with $i, j \in [1, n]$ between bits in the derived responses were calculated for all possible $\frac{n(n-1)}{2}$ bit combinations, where $\tilde{c}_\star = \min(\tilde{c}_{i,j}, 1-\tilde{c}_{i,j})$. The distribution of actual bit correlations $\tilde{c}_\star$ is shown in Figure 5.2. As can be seen, most of $\tilde{c}_\star$-values are between 0.45 and 0.5 with an average of 0.453. In this light, the $c_\star$-values calculated from MCB and NIST 800-90B min-entropy estimations seem to be sufficiently conservative.

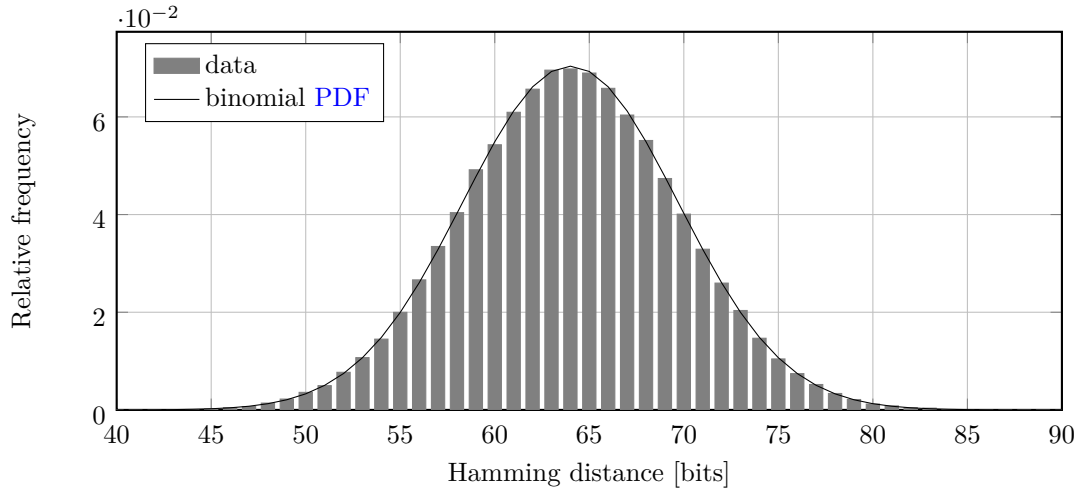


Figure 5.3: Inter Hamming distance distribution of the extracted keys and ideal binomial PDF with $n = 128$ and $p = 0.5$.

5.3 Randomness Extraction

For randomness extraction, the HMAC-based key derivation function (HKDF) scheme discussed in Section 2.4.3 was used. We implemented the HKDF scheme in MATLAB. As proposed in [48], the HKDF was instantiated using HMAC-SHA512 for extracting PRK and HMAC-SHA256 for computing key material of 256-bit length which was truncated to 128 bit. The public random seed W had 256 bits, following Håstad *et al.* [52] and Aysu *et al.* [50], to derive a final 128-bit key. The context information CTXinfo was set to zero. Figure 5.3 shows the $\mathrm{HD}_{\mathrm{inter}}$ distribution of extracted keys with an ideal binomial PDF with $n = 128$ and $p = 0.5$.

The security of the used construction is argued in the random oracle model and it is dependent on the number of queries q that an adversary can make to the extractor, where $\mathbf{SD}(\mathcal{Y}, U_\ell) \leq \min\left(q \cdot 2^{-\tilde{m}}, \sqrt{q \cdot \mathsf{Col}(\mathcal{X})}\right)$ [48] (see also Section 2.4.3). However, in case of a MEMS PUF, an adversary does not have access to the extractor's output since this is the secret key. Preventing access to the extractor's output needs to be ensured at the system level by the tamper-resistance property of the PUF. Thus, we assume q to be rather small meaning that it is possible to extract cryptographic keys with a security level of nearly the residual min-entropy $\tilde{m}$ (see Section 5.2.2).

5.4 Discussion of Obtained Security Levels

As shown in Table 5.5, the resulting security levels vary by roughly a factor of 4 depending on the used entropy estimation method and the assumption that is made on the helper data leakage (leading to lower and upper bounds in the residual min-entropy). However, it can clearly be seen that this large variation is mainly caused by the low values obtained from the NIST 800-90B min-entropy estimation tests. As mentioned in Section 2.3.5 and further discussed in Section 5.1.1, these tests are not intended to be applied on multiple non-independent noise sources with varying bit lengths. This could explain why the test results show a great scattering depending on the ordering of the features and responses. In addition, given the results obtained by applying these tests on true random files provided by NIST, their estimation results seem to be overly pessimistic (see Section 2.3.5). Based on these findings, we tend to assume that the actual min-entropy of the MEMS PUF responses to be higher than the values obtained from the NIST 800-90B min-entropy estimation tests, which are listed in Table 5.5.

Considering the HW, MCB, and CTW estimation methods only, as it was done by other works that evaluated PUF responses (see e.g., SRAM PUFs in Section 5.1.3), the obtained min-entropy estimation results are quite good compared to other PUF constructions such as the SRAM PUF (see Section 5.1.3). In this case, the resulting security level is at least 111 bits considering the repeatability at RT. To be robust against major temperature variation and aging, the obtained security level is reduced to 78 bits due to the increased error correcting capability needed. However, it should be noticed that this is still a considerable security level[5] given the fact that this was achieved with sensors from a single wafer.

5.5 Inter-Wafer Measurements

The entropy estimates of the wafer-level measurements are given in Table 5.4. The entropy was estimated for the three cases

- single wafer,
- 2 wafers from same batch,
- 4 wafers from 2 batches.

[5]Note that prior to 2014, a security level of 80 bits was considered to be sufficient [110].

Table 5.4: Entropy estimation results per bit of wafer-level measurements.

	Single wafer	2 wafers, same batch	4 wafers, 2 batches
n [bits]	172	177	197
HW	1.0	1.0	1.0
MCB	0.88	0.90	0.92
CTW	1.0	1.0	1.0
NIST	0.36	0.38	0.40

Features were ordered so that adjacent correlations were maximized and responses were arranged according to sensors' original position on the wafers. Entropy estimation tests were carried out bit-wise as well as byte-wise. In Table 5.4, only the lowest estimates of MCB and NIST estimation methods are considered. The results of HW and CTW compression estimation methods are constant 1.0 for all three cases. The estimates obtained from MCB and NIST methods slightly increase when considering 4 wafers from 2 batches compared to the cases single wafer and 2 wafers from same batch.

As discussed in Section 4.5, wafer-level measurements cannot be compared directly to measurements on module-level. Thus, the absolute value of increase in the security level of extracted keys cannot be appropriately determined. However, it is possible to show a relative change because the same limitations apply to both the evaluation of a single wafer and multiple wafers.

Since the stability of PUF responses is not affected (see Section 4.5) of the examined cases and conservatively assuming that the entropy per bit remains constant when considering multiple wafers, the security level of extracted keys increases proportional to the response length. Hence, it can be assumed that the security level increases by roughly 15 % when considering four wafers from two batches instead of a single wafer. Additionally, it can be assumed that the security level will further increase when more wafers and batches are considered.

Table 5.5: Upper and lower bounds of residual min-entropy $\tilde{m}$ for minimum min-entropy results obtained from used estimation methods MCB*, CTW† and NIST 800-90B$^{\blacklozenge}$ and performed test procedures repeatability at RT (REP), temperature dependency (TD), high temperature storage life (HTSL) and temperature cycling (TC) with associated results for the error rate of the error correction process $\mathsf{ER}_{\mathsf{ec}}$, the false rejection rate (FRR) and the false acceptance rate (FAR).

	REP			TD			HTSL			TC		
n [bits]	189			189			189			189		
m [1/bit]	0.76^{*}	$0.99^{\dagger}$	$0.38^{\blacklozenge}$	0.76^{*}	$0.99^{\dagger}$	$0.38^{\blacklozenge}$	0.76^{*}	$0.99^{\dagger}$	$0.38^{\blacklozenge}$	0.76^{*}	$0.99^{\dagger}$	$0.38^{\blacklozenge}$
$c_{\bigstar}$	0.408	0.497	0.226	0.408	0.497	0.226	0.408	0.497	0.226	0.408	0.497	0.226
$[n, k, t]$-BCH code	3 x $[63, 51, 2]$			3 x $[63, 39, 4]$			3 x $[63, 39, 4]$			3 x $[63, 39, 4]$		
$\tilde{m}$ [bits] (lower bound)	111	150	48	78	114	27	78	114	27	78	114	27
$\tilde{m}$ [bits] (upper bound)	114	150	57	96	114	45	96	114	45	96	114	45
$\mathsf{ER}_{\mathsf{ec}}$	$4.2 \cdot 10^{-6}$			$1.8 \cdot 10^{-7}$			$5.2 \cdot 10^{-7}$			$4.0 \cdot 10^{-6}$		
FRR	$6.1 \cdot 10^{-10}$			$8.3 \cdot 10^{-7}$			$1.2 \cdot 10^{-7}$			$5.0 \cdot 10^{-7}$		
FAR	$6.8 \cdot 10^{-8}$			$5.8 \cdot 10^{-7}$			$3.0 \cdot 10^{-7}$			$5.8 \cdot 10^{-7}$		

Chapter 6

Towards Practical MEMS PUFs

In this chapter, we discuss additional techniques that can be used to increase the number of bits that can be derived from a MEMS structure. Generally, an increase in the number of derivable bits can be achieved in three ways: improving the measurement circuit, optimizing the MEMS structure design or else widening the fabrication tolerances in the technology used to manufacture MEMS devices.

An improvement on the measurement side can be based on using another measurement method, optimized measurement equipment or modified evaluation methods (e.g., a longer averaging period). This can result in two types of effects: on the one hand, the stability of a feature can be increased by improving the measurement accuracy. On the other hand, this might lead to an increased number of features because additional features might be measurable due to a higher bandwidth and/or a reduced noise floor. In Section 6.1, we focus on the number of measurable frequency modes and we show that more frequency modes can be measured when using an alternative measurement technique.

An optimization of the MEMS design can effect an increase of features' variability and/or an increase of the number of features (e.g., by designing the MEMS structure such that more measurable frequency modes exist). This can also be achieved by worsening the manufacturing accuracy. However, it is rather unlikely that MEMS manufacturers would be willing to intentionally worsen their processes and, thus, we do not consider this approach.

Note that the approach of optimizing the MEMS design can hardly be used if the same MEMS structure acts as both a sensor and a PUF because it will very likely lead to poor sensor performance. In Section 6.2, we show a dedicated MEMS design which was optimized for PUF applications.

6.1 Improving the Measurement Circuitry

In order to investigate the improvement potential that a different measurement concept would have, we used an optical measurement technique which is based on laser-Doppler vibrometry (LDV) [111]. LDV characterization technique enables to measure the velocity (and displacement after integration) of MEMS structures in a non-contacting manner [112]. This has the advantage that parasitic electrical effects on the evaluation side are omitted leading to an improved SNR enabling picometer resolution. Moreover, this method is independent from the presence and the position of sensing electrodes and, thus, movements of parts of the structure can also be measured, which cannot be detected electrically. In the following, we briefly describe the used setup and compare the measurement resolution on an exemplary measurement to those obtained by the used electrical method.

6.1.1 Experimental Setup

The used optical method is based on the LDV characterization principle. In this method, back-scattered light from the moving structure is analyzed in order to get information about the structure's velocity and displacement. A comprehensive description of the LDV characterization principle can be found, e.g., in [113].

A major drawback of standard LDV technique is that visible light is used. Hence, only uncapped wafers can be measured in a vacuum chamber since the silicon cap is opaque to visible light. The special feature of the used measurement technique[1] is that infrared (IR) light is used which passes through the silicon cap but which is back-scattered from the mechanical polysilicon layer [111]. The experimental setup of the IR-laservibrometer measurement is shown in Figure 6.1.

For the sake of comparability, we used the same sensor module for both electrical and optical characterization. Furthermore, we used the same measurement board and we applied the same excitation voltages. Since IR light cannot pass the mold package, the package was removed by chemical etching[2] before the measurements.

[1] The IR-laservibrometer measurement technique is developed within project IRIS which is funded by the German Federal Ministry of Education and Research (grant number 13N13562).

[2] Decapsulation of MEMS by chemical etching is described in Section 7.1.1 in more detail.

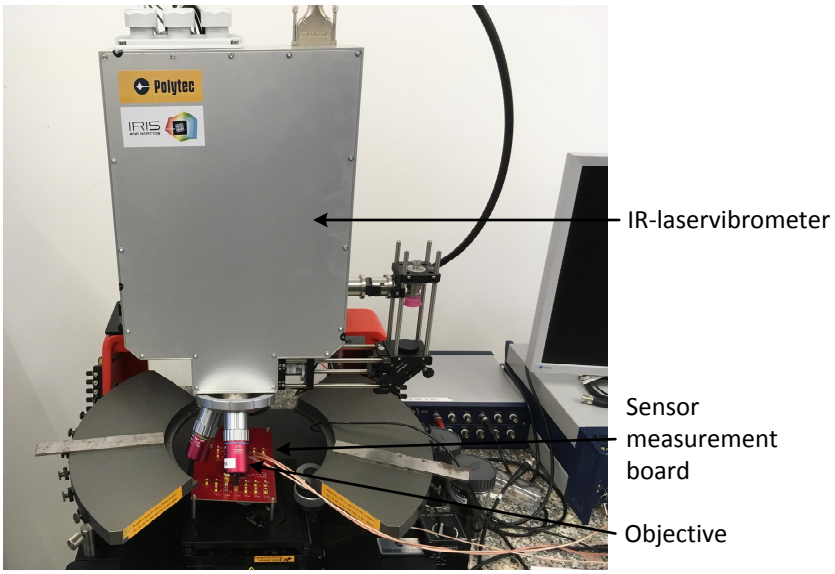


Figure 6.1: IR-laservibrometer measurement setup.

6.1.2 Results

Figure 6.2 shows a small section of the measured frequency spectrum where a frequency mode with a relatively low amplitude is located. The measurement was performed in the x-channel (out-of-plane oscillation) using the electrical and the IR-laservibrometer measurement method. As can be seen, the SNR is significantly higher for the measurement obtained with the IR-laservibrometer. As a result of the low SNR, such a frequency mode cannot be reliably detected with the electrical measurement method and, thus, it was not considered in the analyses in Chapters 4 and 5. Notice that the frequency mode shown in Figure 6.2 is just one example and the same applies to many of the frequency modes of the sensor structure.

6.2 Optimizing the MEMS design

In order to show the potential for improvement from modifying the MEMS structure, we designed a dedicated MEMS structure. Afterwards, the structure was manufactured in a standard manufacturing process. This means that no worsening of the processes was made and the increase of the feature variability was achieved by modifications in the design only.

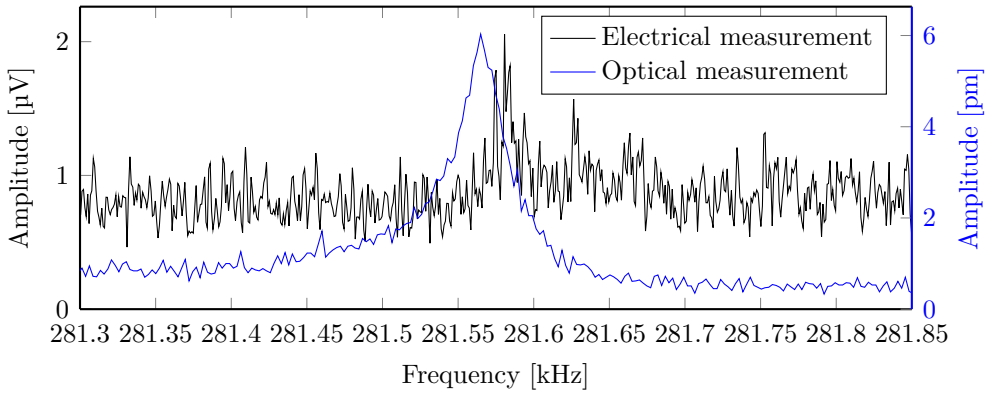


Figure 6.2: Comparative measurement of the used electrical measurement method and the IR-laservibrometer.

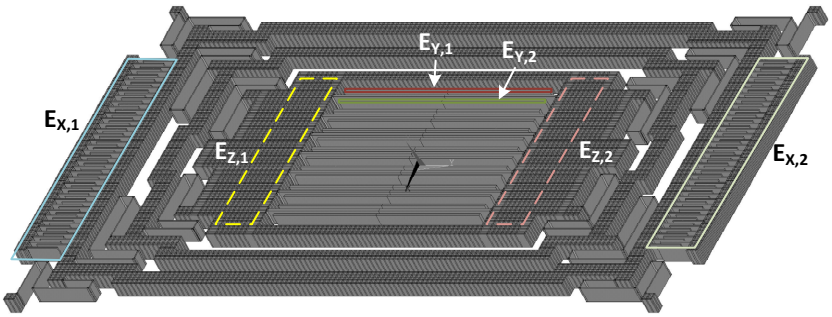


Figure 6.3: Dedicated MEMS design optimized for the use in PUF application.

6.2.1 Dedicated MEMS PUF Design

We designed a dedicated MEMS structure with the aim of providing a high number of frequency modes on a small area while increasing the variability of the mode positions. Figure 6.3 shows the designed MEMS structure. It consists of 3-masses which are linked by double U-springs and the whole structure is suspended by four double U-springs at the outside corners. A method used to increase the variability of the frequency modes was the use of a very narrow beam width in order to increase the percentage influence of the structure width variation. To still enable a robust process we used an aspect-ratio not smaller than 10:1 which means a minimum beam width of 2 µm for a layer thickness of 20 µm.

The structure can be driven and measured in-plane by the electrode pairs $E_{X,1}/E_{X,2}$, $E_{Y,1}/E_{Y,2}$ and out-of-plane by $E_{Z,1}/E_{Z,2}$. Note that the electrodes

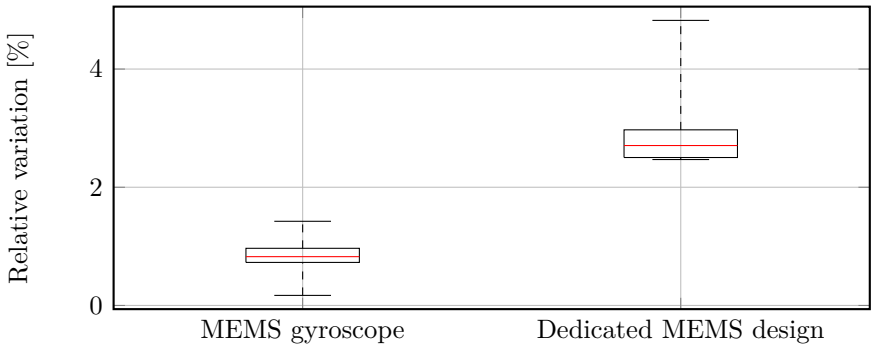


Figure 6.4: Comparison of the relative variation of frequency mode positions for the designed MEMS structure and the investigated MEMS gyroscope.

$E_{Z,1}/E_{Z,2}$ are located below the structure. The die size was 1.5 mm^2 which correspond to roughly 55% of the area of the originally investigated gyroscope (see Section 2.5.2).

The structure contains 12 pronounced frequency modes roughly between 7 kHz and 63 kHz (see Appendix A). All frequency modes could be reliably excited and detected with the developed electrical method, with the exception of mode 1 whose signal peak was too small due to the very low oscillation amplitude of the outer frame. The mechanical structure was designed in a way that the usable frequency modes are close together. This is in contrast to the structure of a MEMS gyroscope where the focus is on the drive and detection modes and all further frequency modes are shifted as far as possible away from them.

6.2.2 Results

In total, we manufactured and measured 69 units of the designed MEMS structure on a single wafer. To show the effect of the applied measures, we analyzed the relative variation of the mode positions and compared this to the results obtained from the baseline measurements of the investigated MEMS gyroscope (13 frequency modes). In Figure 6.4, the boxes indicate the lower and upper quartile, the whiskers show the minimum and the maximum of the relative mode variation and the median is given by the red line. The results show the effectiveness of the used methodology in the design. The variation of the frequency mode positions increases by roughly a factor of 2 on the average.

6.3 Discussion

We could showcase the potential of increasing the number of derivable bits by measures in the measurement technique and in the MEMS design. By using another measurement technique with an improved measurement accuracy, additional features of a MEMS gyroscope could be measured.

Furthermore, we presented a dedicated MEMS design optimized for the use as PUF. It provides 11 usable frequency modes on an area which is only slightly bigger than 50% of the area of the originally investigated MEMS gyroscope. Additionally, we showed that the relative variation of frequency mode positions can be increased by measures in the design. Notice that a further increase of feature's variation could be achieved by measures in the manufacturing processes which are in fact highly optimized to minimize process variations. Additionally, several of such MEMS structures could be placed together on a single die and used for key derivation.

Chapter 7

Physical Attacks on MEMS PUFs

In this chapter, we investigate the susceptibility of MEMS PUFs to physical attacks[1]. Generally, various kinds of physical attacks exist. In [115], physical attacks are categorized as follows:

- *Side-channel attacks.* Side channel attacks use leaked signals like power, current, and electromagnetic radiation emanated during normal operation of the system, which can be used to reveal secret information. These attacks are often performed in a non-invasive manner which makes them hard to detect.

- *Software attacks.* Software attacks are also of a non-invasive nature. They explore vulnerabilities in security protocols or implementation of crypto-algorithms.

- *Fault generation attacks.* Fault generation attacks focus on generating faults in the system, e.g., by running the system in abnormal environmental conditions. It causes system malfunction which might leak information.

- *Microprobing attacks.* This kind of attack requires direct access, e.g. to the electrodes of a device. This attack is typically performed in an invasive manner since protective coatings, e.g. a mold package, have to be removed.

- *Reverse engineering* This attack is also of an invasive nature since the attacker needs to learn the internal structures and functionalities of a device.

[1]The results of this chapter were partly produced in the context of a master thesis [114].

Depending on the type of an attack, specialized equipment and knowledge of hardware, data analysis, and cryptographic algorithms are needed. Based on the resistance to attacks, devices are divided into different levels, e.g., security level 1 (lowest security level) to security level 5 (provides protection against attacks of any type and from any direction) in [116]. Note that a fundamental characteristic of a device to be categorized into a higher security level is tamper resistance.

The aim of performing attacks are also quite different. Usual goals are summarized in [115] and briefly mentioned in the following:

- *Theft of service.* Theft of service is often motivated by access restrictions, e.g., to illegally access pay TV. When an adversary can bypass an access restriction it usually causes great losses for the service provider since the security vulnerability is often shared in a bigger community.

- *Cloning and overbuilding.* This attack deals with cloning of a product, which is also related to reverse engineering to some extent. For example, it enables pirate companies to save development costs.

- *Intellectual property (IP) piracy.* IP piracy means the extraction of information from a product and the illegal usage of the gained insights. The extracted information can also be used for reverse engineering.

- *Denial of service.* This attack aims at disturbing the functionality of a product temporarily or permanently. This kind of attack is often performed by hacker groups.

In this chapter, our aim is to evaluate the tamper evidence property of MEMS PUFs by performing invasive and semi-invasive attacks. In particular, the analysis is focused on:

- PUF package decapsulation to examine the tamper-proof property of a MEMS PUF.

- Piezo shaker measurements to obtain side channel information from MEMS gyroscopes' standard output signals.

- Magnetic field probing to derive information about measured MEMS properties via the magnetic field.

7.1 Decapsulation

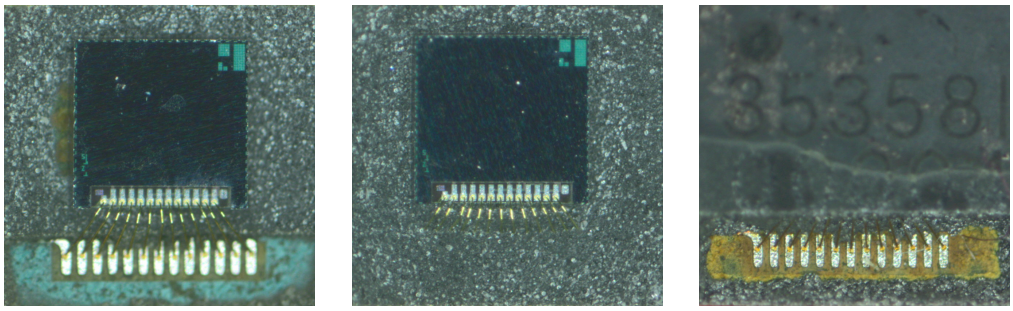
As mentioned earlier, it was hypothesized that the mold package provides inherent tamper protection to a MEMS PUF. In order to investigate this assumption, an attack was conducted to decapsulate the PUF module and measure the effect of the attack on the PUF properties. For this experiment, we used the packaged sensor modules described in Section 3.4. This also enables the analysis of the change of the derived fingerprints based on the performed baseline measurements.

MEMS packages are largely plastic molded with epoxy molding compound. The mold package is a composite structure of different materials such as epoxy polymers, polyimide, silicon dioxide, Teflon, ceramics, and other additives. Decapsulation is the removal of the epoxy molding compounds covering the MEMS package for close examination of die condition, bond condition, die pads, and leads [117]. Decapsulation of a module gives access to the electrodes and the bond wires of a MEMS sensor. An access to the internals of a MEMS PUF makes it possible for an adversary to conduct a microprobing attack to electrically measure the MEMS or just to observe the electrical signals transmitted via the bond wires. This kind of attack is of interest for all three implementation concepts (dual use, additional structure and stand alone) described in Section 2.2.2.

In order to decapsulate the sensor modules, we used laser-assisted wet chemical etching. The MEMS sensors could still be accessed via the test sockets so that the physical properties of the decapsulated modules could be measured by the developed module-level measurement setup (see Chapter 3). Notice that in practice, an attacker would have to use microprobes to contact the sensor or ASIC pads or the bond wires. Observe that an attacker does not need to decapsulate the whole device but it can be sufficient to expose certain (small) areas. In total, 14 sensor modules were decapsulated, measured and the sensor fingerprints were derived using the quantization scheme instantiated from the baseline measurements (see Chapter 4). Then, we compared the derived fingerprints of the same sensors before and after decapsulation.

7.1.1 Laser Assisted Wet Chemical Etching

In the wet chemical etching process, an epoxy compound can be removed by hot nitric acid or fuming sulphuric acid or a mixture of both. In the used laser assisted process, at first, a laser beam was used to pattern the mold package and remove hundreds of µm thickness of mold material. Afterwards, drops of



(a) Etching of complete package (type 1).

(b) Exposure of bond-wires and sensorpads (type 2).

(c) Exposure of substrate electrodes only (type 3).

Figure 7.1: Sensor modules after laser assisted wet chemical etching with different levels of decapsulation.

hot sulphuric acid or nitric acid were poured into the cavity and kept for a limited time so that epoxy compound dissolved in it. Since the use of a laser can lead to the formation of a conductive carbon layer, the sensor modules were cleaned with drops of polyimide remover after etching. At the last step, the senor modules were dried for 60 minutes at high temperature (125°C), to get rid of the moisture.

Laser-assisted wet chemical etching resulted in precise rectangular windows that were formed by the laser whereas a uniformly etched surface was the result of non-assisted wet chemical etching. As shown in Figure 7.1, we decapsulated the sensor modules in three different ways in order to be able to differentiate the effects of decapsulation in a certain area, if any. In particular, the different types of decapsulation were:

- *Type 1:* Etching of the complete package so that the MEMS die, its pads, and the substrate pads were exposed (5 modules, Figure 7.1a),
- *Type 2:* Exposure of the MEMS die, its pads, and partly the bond wires (6 modules, Figure 7.1b),
- *Type 3:* Exposure of substrate pads and partly the bond wires so that the mold compound enclosing the MEMS die remained untouched (3 modules, Figure 7.1c).

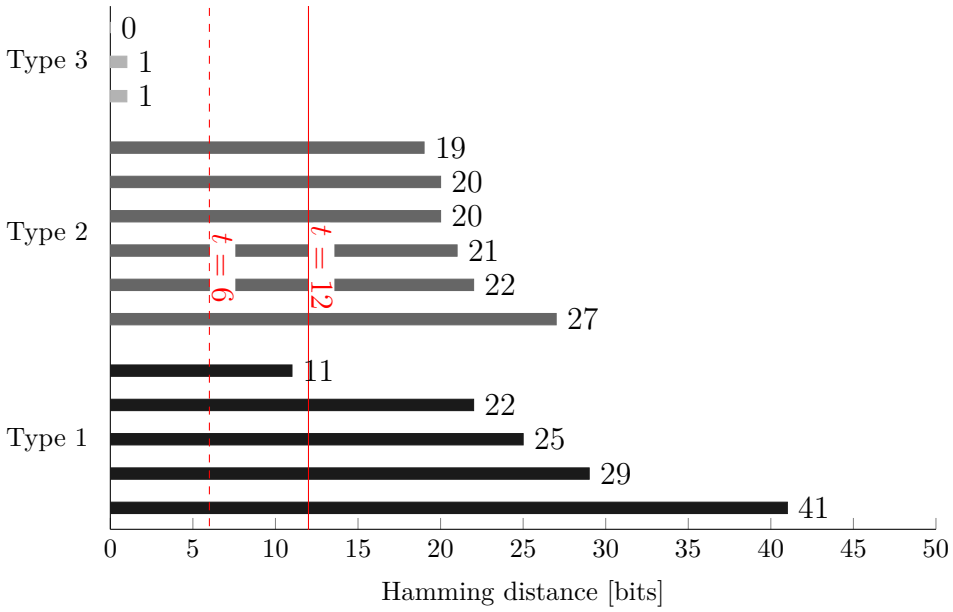


Figure 7.2: Hamming distance between fingerprints derived from sensor modules before and after decapsulation. Error correction capability t needed to reliably correct bit-flips in the PUF responses measured in the repeatability at RT (REP) test ($t = 6$) and in the temperature dependency (TD), high temperature storage life (HTSL) and temperature cycling (TC) tests ($t = 12$) is represented by the two vertical red lines.

7.1.2 Effect of Decapsulation

For analyzing the effect of decapsulation, we calculated the Hamming distance between fingerprints derived from the sensor modules before and after decapsulation. The result of this analysis is shown in Figure 7.2 for each sensor module separately. Note that we measured each sensor module at least 5 times after etching and the distance shown in Figure 7.2 indicates the minimum distance obtained from the repeated measurements for each module. Figure 7.2 also contains the total maximum error correcting capability $t = 12$ of three blocks of a $[n = 63, k = 39, t = 4]$-BCH code used in Section 5.2 (three blocks of a $[n = 63, k = 51, t = 2]$-BCH code was sufficient when stability at RT is only considered).

With one exception, the obtained Hamming distances, after decapsulating the sensor modules according to type 1 and type 2, are significantly above the error

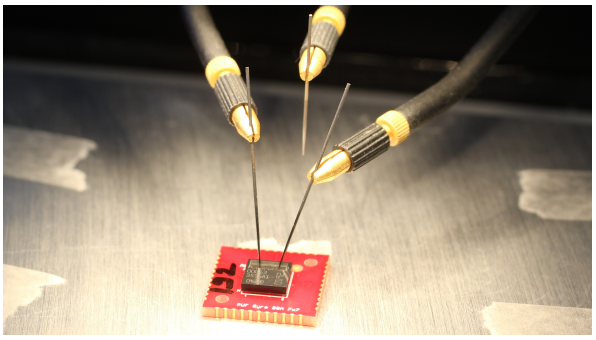


Figure 7.3: Etched sensor module contacted with microprobes.

correcting capability, meaning that an attacker could not easily reconstruct the key based on this measurement. However, since the Hamming distances are rather small to the total length of the fingerprints, he could take the obtained fingerprint w'_0 as a starting point for a brute force attack.

If we assume that an attacker would have access to the postprocessing environment (which means he has to feed-in only quantized PUF responses into the fuzzy extractor to get a final key), he has to guess a bit string w' which is close enough to the sensor fingerprint w before decapsulation ($\mathsf{HD}(w, w') \leq t$). In order to do that he has to guess $\mathsf{HD}(w, w'_0) - t$ flipped bits out of the $n = 189$ bits long fingerprint. Based on that, the effort of this attack can be calculated using binomial coefficients. In particular, the number of trials T an attacker needs to perform is

$$\mathsf{T} = \sum_{i=1}^{\mathsf{HD}(w,w'_0)-t} \frac{n!}{(n-i)!\,i!}. \tag{7.1}$$

For example, considering a Hamming distance of 20 bits and the parameter $t = 12$, this means that $\mathsf{T} \approx 3.63 \cdot 10^{13}$ trials are required which corresponds to a remaining security level of roughly 46 bits. However, it is possible to further reduce the necessary number of trials by making use of cosets. Since an attacker knows, based on the stored helper data, the coset in which w resides, he just needs to consider bit strings originating from this coset.

7.1.3 Microprobing

As mentioned in Section 7.1, in order to capture the key or key material, an attacker would have to physically connect the sensor using microprobes. In order to simulate this attack, we built up a microprobing setup and measured

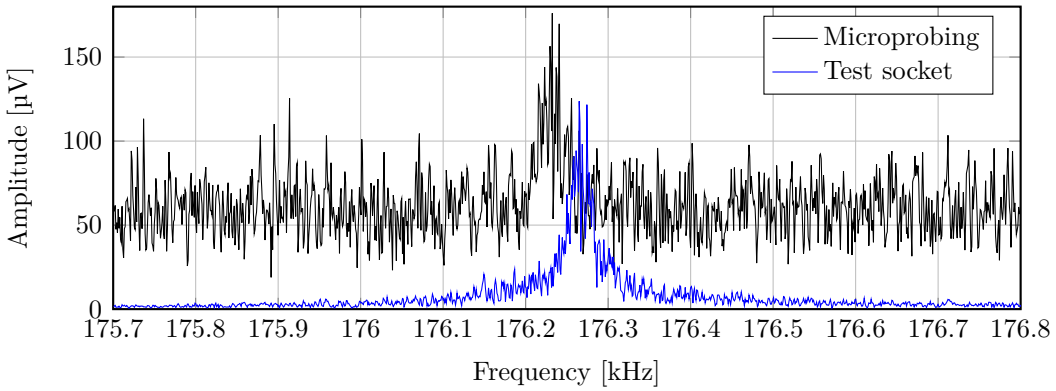


Figure 7.4: Comparison of a frequency mode measured with the same measurement method. In the one case, the sensor pads were contacted via a test socket and, in the other case, via microprobes.

the frequency spectrum of a decapsulated sensor module (Figure 7.3). In that way, we could make electrical contact with the sensor module via fine-tipped needles. The output of the needles was connected to the measurement setup so that we could use the developed measurement method for characterizing the sensor.

Notice that we could only measure frequency modes in that way because two additional probes would be necessary for the measurement of the quadrature signals (2 probes for driving the sensor, 1 probe for measuring the current at the proof mass, and 2 probes for feeding the carrier to a detection channel). However, to add two additional probes was not possible due to lack of space.

Figure 7.4 shows the comparison between microprobe measurements and measurement in a test socket with one frequency mode. The noise level observed in microprobe measurements is significantly higher compared to the measurement in a test socket. As a result, the frequency modes are harder to detect. In addition, we found a small shift in several frequency mode positions. We assume that the reason for this is a slight drop in the applied DC voltages due to the contact resistance between the probes and the substrate pads.

7.1.4 Discussion

Considering type 1 and 2 of decapsulating the sensor modules, the results verified that the MEMS package provides some inherent tamper protection to a MEMS PUF. However, an attacker can significantly improve his chance to

guess the right key by using the measured fingerprint as a starting point for a brute force attack. If an attacker can get access to the MEMS electrodes without affecting the mold compound which directly encapsulates the MEMS die (decapsulation type 3), the key can be reconstructed from the measurement. However, this will be hard to achieve in practice since MEMS die and ASIC are often stacked and, due to a higher packing density, the connecting bond wires are much shorter than those we used to build the modules for this thesis (see e.g., [77]).

Additionally, it should be noticed that a considerable effort is necessary for performing the measurements. As mentioned, the measurement requires microprobing in practice which places high demands in terms of needed equipment and data analysis. Note that in the case of having a complete PUF module, an alternative to characterize the MEMS structure with external equipment would be to observe the measurement signals acquired from the ASIC. However in this case, probing has to be carried out without affecting the impedance of the system.

7.2 Piezo Shaker Measurement

Even though the mold package provides tamper protection to a MEMS PUF, the regular output of a sensor could leak information about the used key material. As an attacker, the idea is to apply an external stimulus while observing the rate signals of a MEMS gyroscope to obtain information about its physical characteristics. To this end, we used a piezo shaker which acts as a source of external force on the gyroscope and we acquired the sensor output corresponding to the shaker movement. Note that this attack represents a threat to the dual use implementation concept only since in this case a regular sensor output is accessible.

7.2.1 Experimental Setup of Piezo Shaker Measurements

In this experiment, an off-the-shelf three-axis sensor module was used which contains an ASIC and the MEMS gyroscope used as DUT throughout this thesis. For the measurement purpose, the sensor module was fixed on the shaker surface with a double-sided tape (Figure 7.5). The sensor module was measured three times in different orientations such that at all three x-, y-, and z-axes of the gyroscope coincided respectively with the up-down movement of

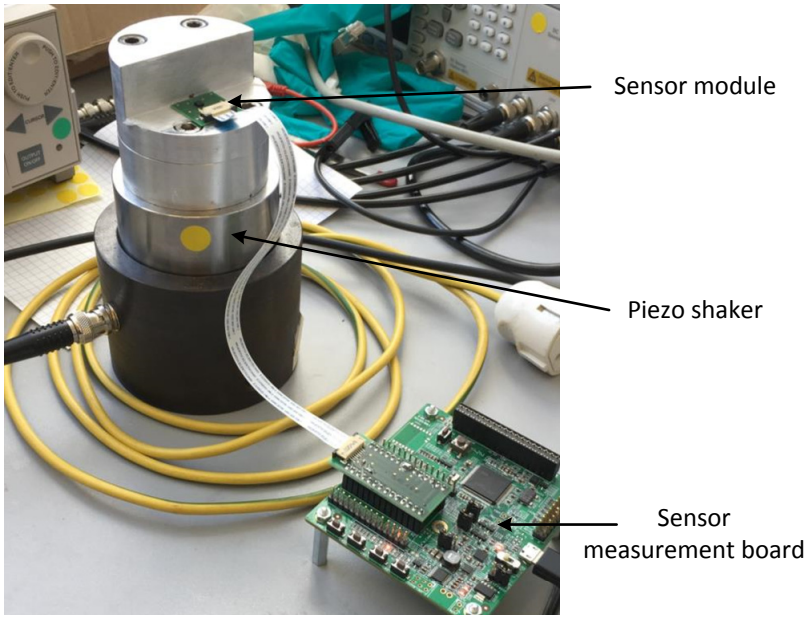


Figure 7.5: Piezo shaker measurements setup.

the shaker. Taking measurements in this way facilitated to record all possible responses of a gyroscope to shaker movement.

The measurement setup mainly comprises a piezo shaker, arbitrary waveform generator, and sensor development board (Figure 7.6). An excitation signal from the arbitrary waveform generator set the piezo shaker into vibration mode. The gyroscope's output was recorded using a standard sensor measurement

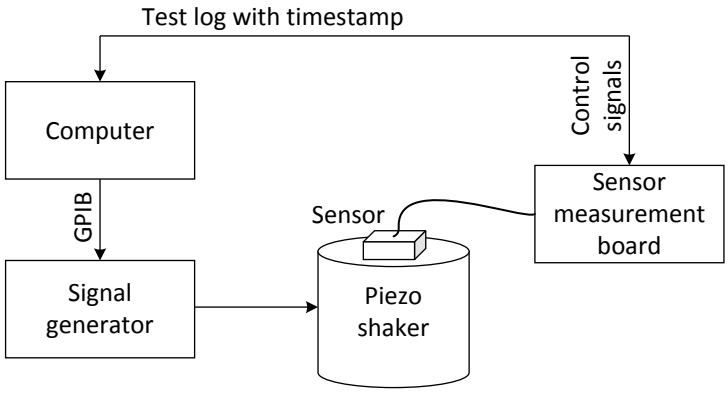


Figure 7.6: Block diagram of piezo shaker measurements.

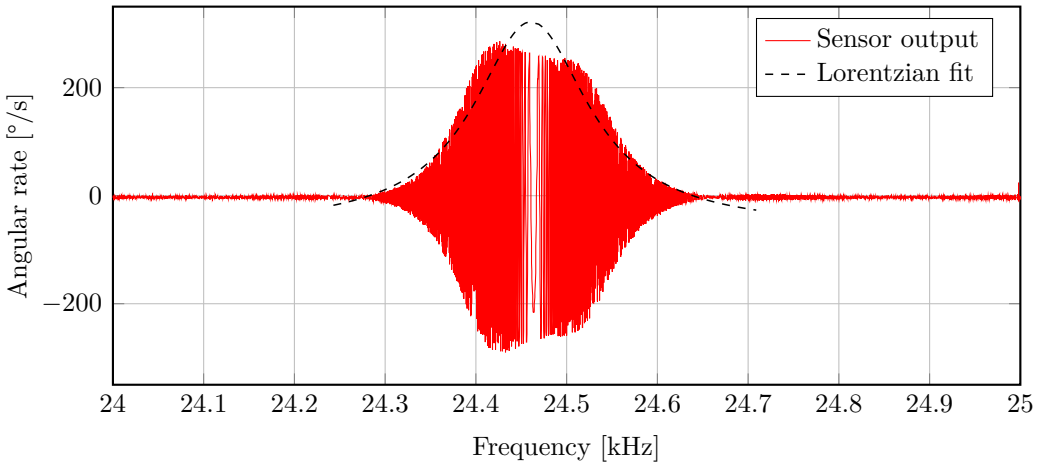


Figure 7.7: Gyroscope's regular output to vibrating frequency sweep around a frequency mode position of the sensor structure with Lorentzian curve fit.

board which provide an interface between a computer and a sensor module. The amplified output of the waveform generator acted as an input to the piezo shaker.

Due to the output of the waveform generator, it is possible to vary the vibration amplitude and frequency of the piezo shaker. The used piezo shaker has a working frequency range of up to a few hundred kilohertz. In order to excite frequency modes, we swept the vibrating frequency around the expected mode positions.

7.2.2 Shaker Measurement Results

In off-the-shelf sensor modules, there exists a built-in sampler unit with a sampling frequency around 6 kHz. Since the sampling frequency is low compared to the applied vibrating frequencies, an undersampled signal burst is obtained as a gyroscope's response. As an example, a response of gyroscope to shaker movement along the x-axis is as shown in Figure 7.7.

We checked signal bursts along x-, y-, and z-axes of three off-the-shelf sensor modules in specific frequency ranges around expected mode positions. By evaluating the regular sensor output, we found evidence that roughly half of the

positions of the 13 frequency modes used in this thesis can be approximately estimated in this way[2].

7.2.3 Discussion

As mentioned, the results of piezo shaker measurements suggest that during normal functioning of the sensor, side information can be derived from the sensor output. However, based on the experiments performed, it is not possible to quantify the amount of information leakage. Nevertheless, we can conclude that several problems remain for an attacker to guess the key:

1. He has to guess the resonance peak of the modes out of the rate signal which does not exhibit a smooth Lorentian curve in many cases.

2. Several frequency modes do not affect the regular rate signal so that they remain unknown.

3. Other features might be (additionally) used, e.g. quadrature signals, which cannot be extracted by this kind of attack.

7.3 Magnetic Field Probe to Read Ground Current

Every current carrying conductor produces a magnetic field depending on its physical and electrical characteristics. Inside a MEMS package, bond wires interconnect sensor and ASIC pads and the electrical current flowing through the bond wires generate a magnetic field which is a typical source of information for non-invasive side-channel attacks [115].

A MEMS gyroscope is usually connected to a number of bond wires which carry various signals like excitation signals and ground current. In this experiment, the feasibility of extracting information about the key material (e.g., frequency mode positions) from outside the package is evaluated by measuring and simulating the magnetic field around a bond wire which carries the ground current. This kind of attack affects all three proposed implementation concepts if bond wires are used.

There are a number of ways to sense magnetic fields. For this experiment, we focused on search coils or magnetic pickup loops. The magnetic pickup loop is a simple, low cost, and easy to manufacture magnetometer. The working

[2] A comparison with the actual frequency mode positions of measured sensors was not possible due to unavailability of these data.

principle is based on fundamental Faraday's law of induction. In particular, the induced voltage in the coil V_{in} with area A is proportional to the negative rate of change of magnetic flux Φ [118]

$$V_{in} = -n\frac{d\Phi}{dt} = -nA\frac{dB}{dt} = -\mu_0 nA\frac{dH}{dt}, \tag{7.2}$$

where, μ_0 is the permeability of free space, n is the number of turns, B is the magnetic flux density, and H is the magnetic field strength.

The sensitivity of such a coil sensor can be significantly increased by using a coil with a ferromagnetic core which concentrates the flux inside the coil. In this case, Equation (7.2) can be rewritten as [118]

$$V_{in} = -\mu_0\mu_r nA\frac{dH}{dt}, \tag{7.3}$$

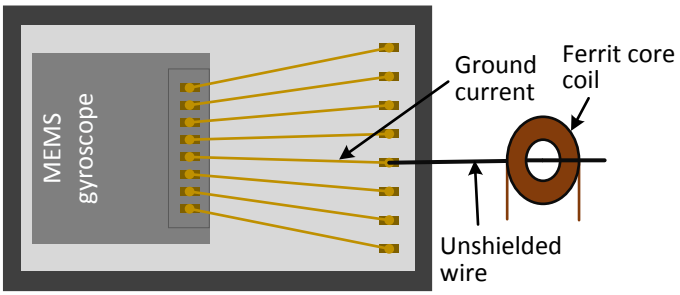
where μ_r is the relative permeability which can be in the range of 10^5 for some soft magnetic materials (e.g., permalloy which is a highly ferromagnetic nickel–iron alloy).

To be able to measure the magnetic field around a bond wire with high accuracy, it would be beneficial for an attacker to put the coil sensor as close as possible to the wire. However, when measuring the ground current from outside the package, the package dimensions limit the distance to the wires. In the following sections, we evaluate the feasibility of reading the ground current from outside the package by an experimental and simulative analysis. The simulations of the magnetic field probe were realized in COMSOL multiphysics.

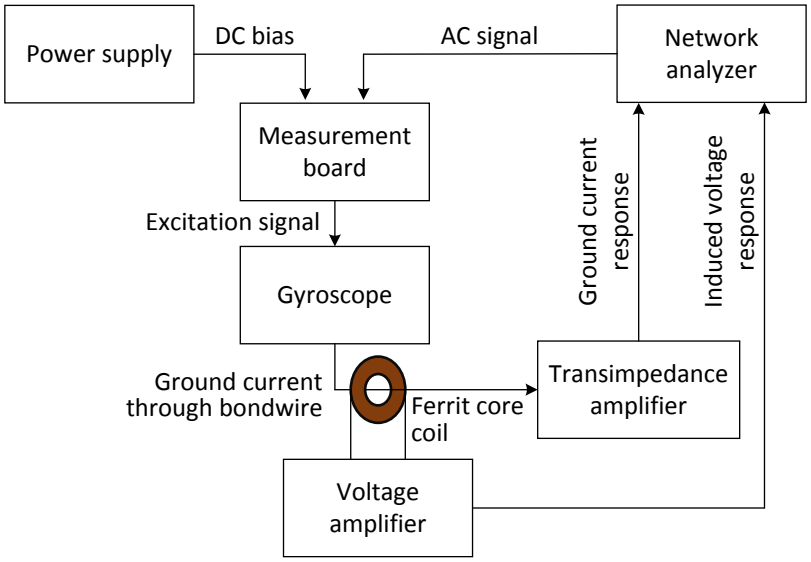
7.3.1 Extended Bond Wire Experiment

In order to build a valid simulation model, we initially performed an experiment to directly measure the electrical current and the magnetic field of a bond wire. For this purpose, we took one of the dedicated sensor modules and connected an unshielded wire to the connector of the test socket carrying the ground current. As shown in Figure 7.8a, this extended bond wire connection was passed through a ferrite core coil.

A block diagram of the extended bond wire experiment is shown in Figure 7.8b. As an excitation signal, a combination of the DC bias voltage and the AC voltage was imposed to excite the mechanical structure. For the measurement, we used a network analyzer 4395A from Agilent which performed a frequency sweep.



(a) Illustration of extended bond wire experiment.



(b) Block diagram of extended bondwire experiment.

Figure 7.8: Principle of the extended bond wire experiment.

In order to simulate the PUF measurement, the signal level of the applied excitation voltages had to be determined. Observe that the magnetic field depends on the frequency of the ground current. Thus, the implementation of the measurement concept has to be considered. Since we expect that in a system implementation the ASIC would work without a carrier, we estimated the AC level of the excitation signal necessary to still be able to electrically detect the used frequency modes in the baseband. As a result, we set the AC voltage amplitude used for the frequency sweep to 100 µV.

With this configuration, we carried out a measurement for two mode positions, one at 23 824 Hz (drive mode) and the other at 113 879 Hz (higher para-

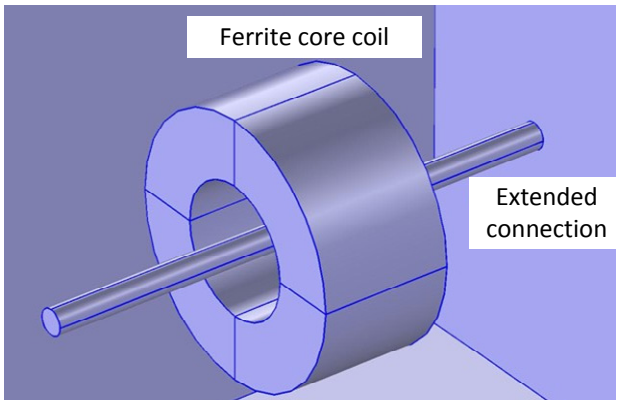


Figure 7.9: Simulation model of extended bondwire experiment.

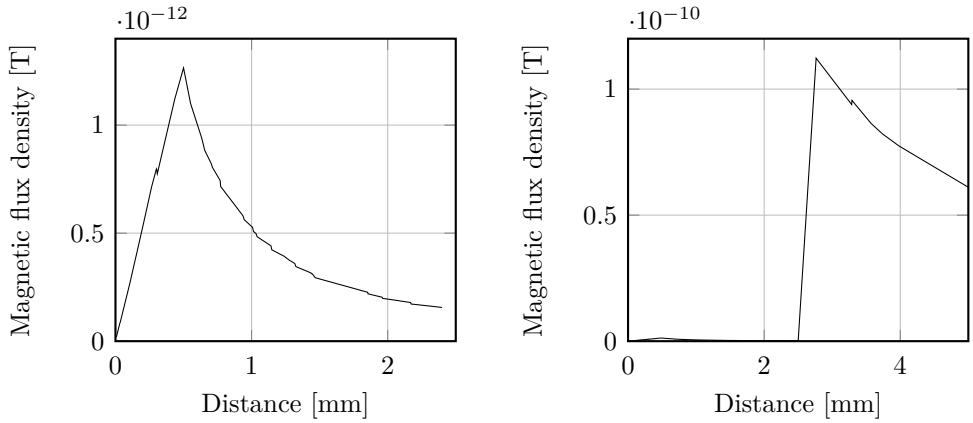
sitic mode). The frequency mode positions obtained from the direct measurement of the ground current and from the measurement with the coil matched. The measured ground current peak was 3.4 nA for the drive mode and 2.5 nA for the high-frequency mode. The voltage peak induced in the coil at low frequency was 16.6 nV and at high-frequency mode position 60.0 nV.

7.3.2 Simulation of Extended Bond Wire Experiment

The extended bond wire experiment was replicated by a simulation model consisting of a current carrying wire and a ferrite core coil enclosed inside an air-box (Figure 7.9). The air-box was used to simulate the space surrounding the wire and the coil. Also, it defined a boundary for the field such that all magnetic field generated remains inside the box only. The conducting wire had a diameter of 1 mm. The ferrite core had an inner diameter of 5 mm, an outer diameter of 10 mm, a width of 5 mm, 105 number of turns, and a relative permeability μ_r of 850. The readings for the current through the bond wire and the corresponding frequency mode position measured in Section 7.3.1 were utilized for the simulation.

The simulation model was built iteratively. Initially, a model with only a conducting wire placed inside an air-box was simulated. In Figure 7.10a, we observe that the magnetic flux density is zero at the center of the wire and increases linearly as we move along the radius. Outside the wire the flux density falls off.

In the next step, we simulated the magnetic flux density profile with a ferrite core put around the wire. The use of high permeability material increases the



(a) Magnetic flux density profile without ferrite core coil.

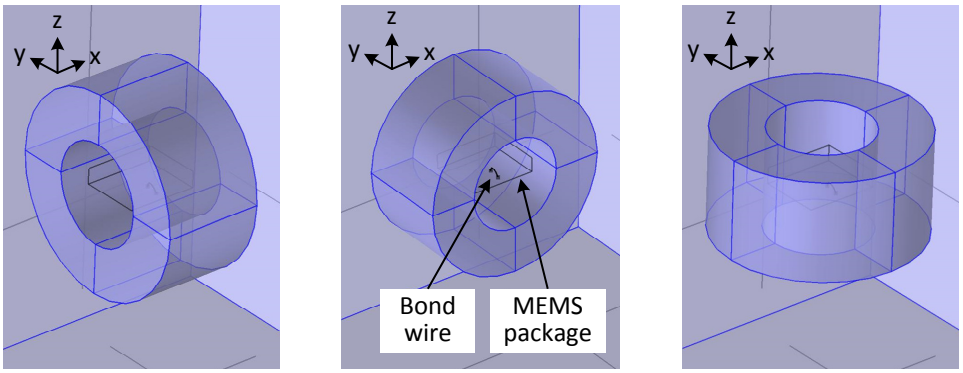
(b) Magnetic flux density profile with ferrite core coil.

Figure 7.10: Comparison of the generated magnetic flux density with and without ferrite core for the ground current of the drive mode at 23 824 Hz depending on the distance to the center of the conducting wire.

flux concentration. As shown in Figure 7.10b, there is a significant increase in the flux density in the region of the ferrite core. The simulated voltage peak for the drive mode was 16.8 nV and 59.4 nV for the high-frequency mode. The small deviation between measured and simulated results indicates a good accuracy of the simulation model.

7.3.3 Simulation of a packaged MEMS PUF Module

The extended bond wire experiment showed the feasibility of detecting frequency modes. However, this experiment implicates that the coil has to be positioned directly around the bond wire carrying the ground current in practice when the sensor part is connected to an ASIC. On the one hand, this would be a challenging task and, on the other hand, it would require an invasive attack to get access to the bond wires. Thus, we build another model, simulating the case that an attacker would put a coil around the entire MEMS package, as shown in Figure 7.11. In this case, the dimensions of the MEMS package limit the distance of the coil to the bond wire. We assumed the package dimensions to be $3.5 \times 3 \times 1$ mm^3 which is a common package size for consumer sensors.



(a) Coil position perpendicular to x-axis.

(b) Coil position perpendicular to y-axis.

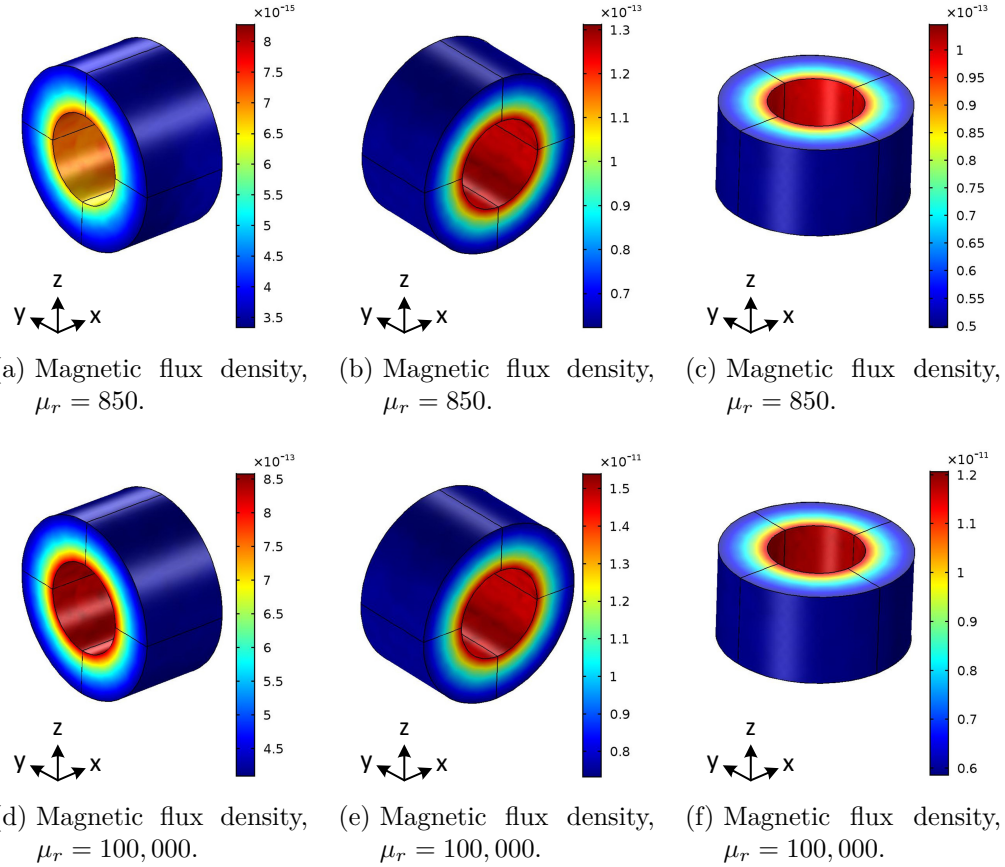
(c) Coil position perpendicular to z-axis.

Figure 7.11: Simulation model for a packaged PUF module with a ferrite core coil placed in three different orientations.

For the gold bond wire, we considered a diameter value of 15 µm. Additionally, we placed the coil in three different orientations (see Figure 7.11).

In a first simulation run, the ferrite material and number of turns of the coil were kept the same as before. Also the core dimensions were the same since the inner diameter was big enough to fit around the MEMS package. However, the distance to the wire was significantly increased in this model compared to the extended bond wire experiment due to the small diameter of the bond wire. We simulated the model for both the drive mode at 23 824 Hz and the high-frequency mode at 113 879 Hz.

The results of simulation presented in Figure 7.12 were obtained for the drive mode position. The magnetic flux density values in the core region for respective coil position are shown in Figures 7.12a, 7.12b and 7.12c. The maximum induced voltage of 17.3 pV (61.1 pV for the high-frequency mode) is obtained for the coil position shown in Figure 7.12b. However, in a real application, when a PUF module is soldered on a PCB, it will be easier for an adversary to put a coil in the position as shown in Figure 7.12c. In this case the simulated induced voltage is 13.8 pV for the drive frequency and 48.7 pV for the high-frequency mode. A drop in the induced voltages by up to roughly three orders of magnitude is observed when compared with the value obtained by the extended bond wire experiment which is a result of the increased distance between the current carrying wire and the coil.



(a) Magnetic flux density, $\mu_r = 850$.

(b) Magnetic flux density, $\mu_r = 850$.

(c) Magnetic flux density, $\mu_r = 850$.

(d) Magnetic flux density, $\mu_r = 100,000$.

(e) Magnetic flux density, $\mu_r = 100,000$.

(f) Magnetic flux density, $\mu_r = 100,000$.

Figure 7.12: Simulation results for the magnetic flux density in the coil core placed around a PUF module with common package dimensions.

In a second simulation run, the parameters of the coil were adapted. As mentioned previously, materials with relative permeability as high as 100,000 exist. Thus, we set $\mu_r = 100,000$. In addition, the number of turns were increased to 1000 which still seems to be a reasonable number for the dimensions of the used coil. The simulation results for the magnetic flux density in the coil core obtained for the drive mode are shown in Figures 7.12d, 7.12e and 7.12f. These results demonstrate that the use of a high permeability core material can significantly improve the flux density values in the core region. Also, induced voltage values are raised up to a few tens of nanovolt. Finally, it should be noticed that simulations did not take into consideration the skin effect, any losses due to eddy currents, and hysteresis in the core material.

7.3.4 Discussion

The results obtained from the experiments indicate the presence of a weak magnetic field around the MEMS package leading to voltages in the range of picovolt induced in the coil. By using a high permeability core material and a large number of turns, the induced voltages can be increased up to a few nanovolt which is in a measurable range. Although simulation results can detect a weak magnetic field around the MEMS package, in practice the ability of a pickup coil to measure weak signals is limited by the presence of noise. In the simulation model, noise factors and their effects were not considered. There are various types and sources of noise, such as magnetic noise, thermal noise, interference from signals carried by other bond wires, interference from ASIC communication signals, and interference of other electronic measurement devices. Furthermore, in a real application, a MEMS PUF will likely be soldered on a PCB. The physical dimensions of the PCB, the mold package, and, maybe, other components on the PCB could make it difficult to put the coil in the required position close to the PUF. However, the performed experiment indicates that the magnetic field of the current carrying bond wires might be a possible attack path for a MEMS PUF.

Chapter 8

Conclusion and Outlook

8.1 Summary of Main Contributions

A fundamental requirement for the success of the Internet of Things (IoT) is that the security of its components can be guaranteed. The necessary tools for this are cryptographic methods such as authentication and encryption, which require cryptographic keys that have to be securely stored resistant against attackers. In this context, so-called physical unclonable functions (PUFs) are considered to be particularly promising to make this possible. PUFs are based on inherent variations in the manufacturing process of a device. As a result, each device is unique which means that it has a unique fingerprint. In addition, PUFs are unclonable and often resistant to different kind of attacks, making them more secure compared to conventional storage technologies.

This thesis provides solid evidence that MEMS sensors, which are ubiquitous in the IoT, can be used as PUFs under practical test conditions. We show that every single MEMS sensor has a unique fingerprint from which a cryptographic key can be extracted. A basic requirement for a MEMS PUF is that a key derived from it can be reliably reconstructed across the whole range of operation conditions and over the sensors' life-time. Thus, we analyzed for the first time the robustness of keys extracted from MEMS gyroscopes considering temperature stability and long-term stability. In particular, we performed measurements across a temperature range from $-40°C$ to $85°C$ and standardized aging tests, namely temperature cycles (1000 cycles between $-40°C$ and $125°C$) and high temperature storage (1000 h at $125°C$). The test results show that the extracted keys can be reliably reconstructed with an error probability in the lower ppm range.

Regarding the usability of the extracted keys in cryptographic operations, a security level of 128 bits would be desirable. Even though the derived fingerprints have a length of 189 bits, the security level of the extracted keys is lower

since binary strings derived from the sensors are not entirely random. The reason for this is that some of the used MEMS characteristics exhibit correlations. Additionally, due to noise and changes in the exact characteristic values caused by temperature variation and aging, an error correction step is necessary to correct single bit-flips in the strings which leaks some information about the key material. As a result, the usable key length is reduced.

In order to determine the security level of the extracted keys, the entropy of the derived fingerprints has to be estimated. We did this by using the estimation methods Hamming weight (HW), most common byte (MCB), and context tree weighting (CTW) which are often used in the context of PUFs. Since those methods cannot sufficiently consider the effect of correlations in the derived bit strings, we additionally carried out min-entropy estimation tests mentioned in NIST special publication 800-90B. We found that the estimated entropy shows significant variation depending on the used estimation method which can be explained by the nature of the different tests. Summarizing the results from HW, MCB and CTW estimation methods, the obtained security level of the extracted keys is at least 78 bits in the worst case. When considering estimates from the NIST tests, this drops down to 27 bits.

Summarizing, we could demonstrate that it is possible to extract a considerable number of robust bits with nearly full entropy from a current MEMS-based gyroscope. In addition, we could show that considering sensors from just a few or even a single wafer can be seen as a worst case scenario due to spatial correlations between sensors originating from the same wafer. As a result, we can expect that the security levels obtained represent lower bounds for the extractable key length from a MEMS PUF and the actual number of bits will be higher in practice when the used sensors originate from multiple wafers manufactured in multiple batches. Furthermore, we provide evidence that the number of bits that can be derived from a sensor can be further increased by improving the used measurement technique.

While the main focus of this thesis lies on the investigation of a MEMS structure which was actually designed for sensing angular rates, it is also possible to build a dedicated MEMS structure only for secure key storage. For this purpose, we designed and manufactured a MEMS structure optimized for the use as PUF while at the same time optimizing the design for minimal area. In an implementation, such a structure or several of them could be placed aside a sensor structure or used to create a high-end stand-alone product, e.g. a security-token. Note that based on the results of this thesis it is already possible to build a MEMS-based PUF providing a 128 bit key by combining several MEMS structures (multi-core approach).

An important property of a MEMS PUF is its resistance to invasive and semi-invasive attacks. To analyze the robustness of a MEMS PUF against such attacks, we simulated three different physical attacks trying to capture information about the characteristics used for deriving the fingerprint.

First, we decapsulated packaged sensor modules and compared the sensor's fingerprint before and after etching in order to verify our hypothesis that a MEMS PUF is sensitive to invasive attacks since a change in the internal environment (e.g., stress conditions) would change the exact MEMS characteristics. This attack analyzed the threat that an attacker could measure the MEMS by getting access to the electrical interconnection between MEMS and ASIC die. We found that in fact the change in the fingerprint due to decapsulation is big enough in most cases so that an attacker could not simply reconstruct the original fingerprint of the sensor if the mold package in the direct environment of the MEMS die is affected. However, taking this fingerprint as a starting point for a brute force attack significantly increases the chance of an adversary to find the right bit combination. It has to be noticed that this attack still requires difficult microprobing which we could omit since we could use our dedicated PUF modules enabling to access the sensor pads via a measurement board.

In a second attack scenario, we investigated if information about the key material could be extracted from the regular sensor output. To this end, we used a piezo shaker which mechanically excited the MEMS sensor externally. Note that this scenario is only relevant if the same structure is used for both sensing and key generation. The results show that at least some information about the used characteristics can be extracted in this way. So we can conclude that a separate MEMS structure for key generation or even a stand-alone device would be desirable to meet highest security requirements when a key has to be securely stored.

A third attack analyzed the threat that information about the key material could be gained by measuring unavoidable magnetic fields around the MEMS package. We show by measurements and simulations that the magnetic field around the current carrying bond wires could be high enough to be measured. However, it is not clear if this attack would work in practice when additional effects like noise and interference from other signals are present. In conclusion, the results obtained from the different attacks performed show that a MEMS PUF provides a high level of protection against physical attacks.

8.2 Recommendations for Further Research

As mentioned, we found that the entropy estimation results show a high variation across the performed tests. Additionally, the estimates of the tests provided by NIST special publication 800-90B were significantly lower than those of the other tests. As discussed in Section 5.4, the NIST tests are originally not intended for this type of data. Moreover, their estimates seem to be overly pessimistic as shown by the results obtained by running these tests on true random files. At this point we can conclude that there is some further research necessary to estimate the entropy of correlated binary strings more accurately. Since spatial dependencies are inherent in almost every semiconductor manufacturing process, new estimation methods should also be considered for the evaluation of other PUF technologies which, up to now, were assumed to be free of correlations.

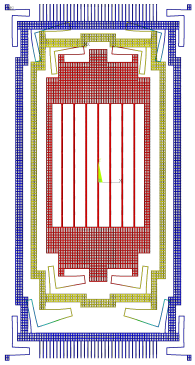
Furthermore, the feasibility of extracting a 128-bit or even longer key from a MEMS sensor in practice can be investigated. Regarding the number of derivable bits, we could show that there is evidence that the extractable key length will be significantly higher when the sensors originate from multiple wafers. In addition, the fingerprints of different sensors could be combined if available, e.g. on sensor nodes. Besides, the potential of a dedicated MEMS device for secure key storage only as a product for the security market can be examined.

We could show that a MEMS PUF provides inherent resistance against physical attacks. However, regarding the results obtained from the attacks performed, additional measures could be investigated to further increase the robustness of a MEMS PUF, especially when the same MEMS structure is used as both a sensor and a PUF. Such additional measures could be appropriate filtering of the sensor output or the selection of features about which no information is contained in the regular output. Besides, further integration and new packaging concepts could be considered making the microprobing attack harder to mount, e.g., when MEMS and ASIC dies are stacked vertically and connected by through silicon vias which are not accessible from outside. Regarding the magnetic field probe attack, more investigations on a final MEMS PUF module are necessary in order to be able to consider effects such as noise and interference from other signals in detail.

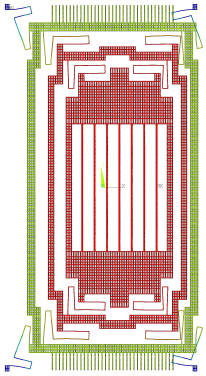
Furthermore, the temperature and the long-term stability needs to be verified when the measurement unit (ASIC) itself is part of the PUF module because a change of the ASIC properties could negatively affect the stability of the derived fingerprints.

Appendix A

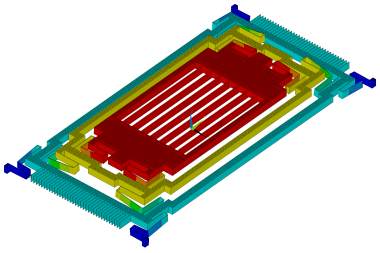
Dedicated MEMS PUF Design Simulation



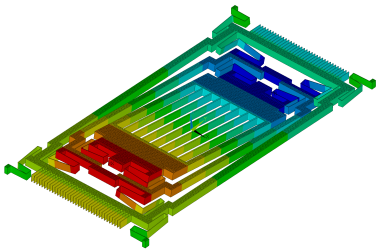
(a) Mode 1 @7645 Hz.



(b) Mode 2 @7840 Hz.

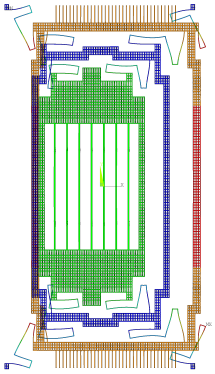


(c) Mode 3 @11035 Hz.

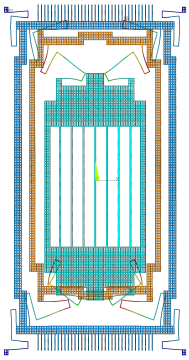


(d) Mode 4 @19354 Hz.

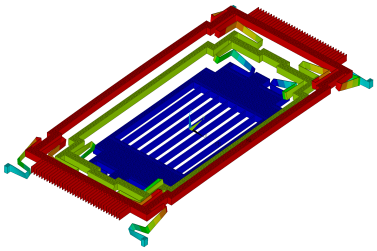
Figure A.1: FEM-simulation with ANSYS of frequency modes 1-4.



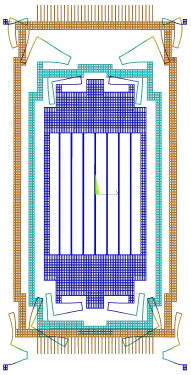
(a) Mode 5 @25776 Hz.



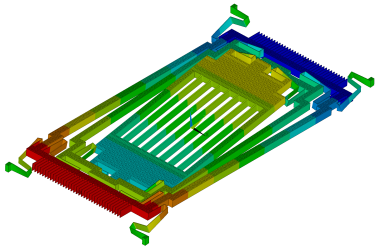
(b) Mode 6 @27203 Hz.



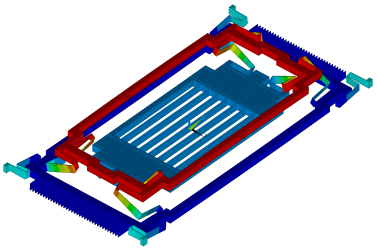
(c) Mode 7 @33161 Hz.



(d) Mode 8 @39199 Hz (out-of-plane translational).

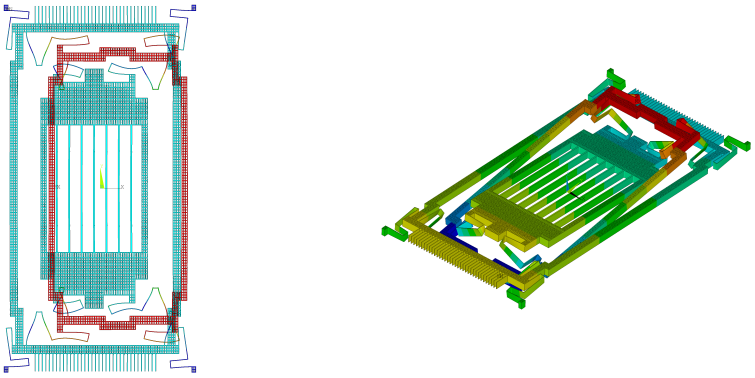


(e) Mode 9 @43429 Hz.



(f) Mode 10 @48137 Hz.

Figure A.2: FEM-simulation with ANSYS of frequency modes 5-10.



(a) Mode 11 @49554 Hz. (b) Mode 12 @62127 Hz.

Figure A.3: FEM-simulation with ANSYS of frequency modes 11-12.

Bibliography

[1] B. Gassend, D. E. Clarke, M. van Dijk, and S. Devadas, "Silicon physical random functions," in *Proceedings of the 9th ACM Conference on Computer and Communications Security – CCS 2002*, V. Atluri, Ed. ACM, Nov. 2002, pp. 148–160.

[2] J. Lee, D. Lim, B. Gassend, G. Suh, M. van Dijk, and S. Devadas, "A technique to build a secret key in integrated circuits for identification and authentication applications," in *2004 Symposium on VLSI Circuits, Digest of Technical Papers*, June 2004, pp. 176–179.

[3] G. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *44th Design Automation Conference – DAC 2007.*, June 2007, pp. 9–14.

[4] J. Guajardo, S. Kumar, G.-J. Schrijen, and P. Tuyls, "FPGA intrinsic PUFs and their use for IP protection," in *Cryptographic Hardware and Embedded Systems – CHES 2007*, ser. LNCS, P. Paillier and I. Verbauwhede, Eds., vol. 4727. Springer Berlin Heidelberg, 2007, pp. 63–80.

[5] B. Gassend, M. V. Dijk, D. Clarke, E. Torlak, S. Devadas, and P. Tuyls, "Controlled physical random functions and applications," *ACM Transactions on Information and System Security*, vol. 10, no. 4, pp. 3:1–3:22, Jan. 2008.

[6] K. Eldefrawy, G. Tsudik, A. Francillon, and D. Perito, "SMART: Secure and minimal architecture for (establishing dynamic) root of trust," in *19th Network and Distributed System Security (NDSS) Symposium.* The Internet Society, Feb. 2012.

[7] F. F. Brasser, B. E. Mahjoub, A. Sadeghi, C. Wachsmann, and P. Koeberl, "TyTAN: Tiny trust anchor for tiny devices," in *Proceedings of the 52nd Annual Design Automation Conference – DAC 2015.* ACM, June 2015, pp. 34:1–34:6.

[8] N. Asokan, F. F. Brasser, A. Ibrahim, A. Sadeghi, M. Schunter, G. Tsudik, and C. Wachsmann, "SEDA: Scalable embedded device attestation," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security – CCS 2015*, Oct. 2015, pp. 964–975.

[9] S. Devadas, E. Suh, S. Paral, R. Sowell, T. Ziola, and V. Khandelwal, "Design and implementation of PUF-based "unclonable" RFID ICs for anti-counterfeiting and security applications," in *IEEE International Conference on RFID*, Apr. 2008, pp. 58–64.

[10] D. Samyde, S. P. Skorobogatov, R. J. Anderson, and J. Quisquater, "On a new way to read data from memory," in *Proceedings of the First International IEEE Security in Storage Workshop – SISW 2002*. IEEE Computer Society, Dec. 2002, pp. 65–69.

[11] S. P. Skorobogatov, "Data remanence in flash memory devices," in *Cryptographic Hardware and Embedded Systems – CHES 2005*, ser. Lecture Notes in Computer Science, J. R. Rao and B. Sunar, Eds., vol. 3659. Springer, Aug. 2005, pp. 339–353.

[12] J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Calandrino, A. J. Feldman, J. Appelbaum, and E. W. Felten, "Lest we remember: cold-boot attacks on encryption keys," *Communications of the ACM*, vol. 52, no. 5, pp. 91–98, 2009.

[13] S. R. Pappu, "Physical one-way functions," Ph.D. dissertation, Massachusetts Institute of Technology, 2001.

[14] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical one-way functions," *Science*, vol. 297, no. 5589, pp. 2026–2030, 2002.

[15] D. E. Holcomb, W. P. Burleson, and K. Fu, "Initial SRAM state as a fingerprint and source of true random numbers for RFID tags," in *Proceedings of Conference on Radio Frequency Identification Security*, 2007.

[16] P. Tuyls and B. Škorić, "Secret key generation from classical physics: Physical uncloneable functions," in *AmIware Hardware Technology Drivers of Ambient Intelligence*, S. Mukherjee, R. M. Aarts, R. Roovers, F. Widdershoven, and M. Ouwerkerk, Eds. Springer Netherlands, 2006, pp. 421–447.

[17] C. Boehm, *Physical Unclonable Functions in Theory and Practice.* Springer, 2013.

[18] R. Maes, *Physically Unclonable Functions: Constructions, Properties and Applications.* Springer, 2013.

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

[20] Y. Taur and T. H. Ning, "Memory devices," in *Fundamentals of Modern VLSI Devices.* Cambridge University Press, 1998, ch. 9.

[21] F. Armknecht, D. Moriyama, A.-R. Sadeghi, and M. Yung, "Towards a unified security model for physically unclonable functions," in *Proceedings of the RSA Conference on Topics in Cryptology - CT-RSA 2016 - Volume 9610.* New York, NY, USA: Springer-Verlag New York, Inc., 2016, pp. 271–287.

[22] U. Rührmair, J. Sölter, and F. Sehnke, "On the foundations of physical unclonable functions," Cryptology ePrint Archive, Report 2009/277, 2009.

[23] F. Armknecht, R. Maes, A. Sadeghi, O.-X. Standaert, and C. Wachsmann, "A formalization of the security features of physical functions," in *IEEE Symposium on Security and Privacy – SP 2011*, May 2011, pp. 397–412.

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

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

[26] U. Rührmair, J. Sölter, F. Sehnke, X. Xu, A. Mahmoud, V. Stoyanova, G. Dror, J. Schmidhuber, W. Burleson, and S. Devadas, "PUF modeling attacks on simulated and silicon data," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 11, pp. 1876–1891, 2013.

[27] F. Ganji, S. Tajik, and J. Seifert, "PAC learning of arbiter PUFs," *Journal of Cryptographic Engineering*, vol. 6, no. 3, pp. 249–258, 2016.

[28] F. Ganji, S. Tajik, F. Fäßler, and J. Seifert, "Having no mathematical model may not secure PUFs," *Journal of Cryptographic Engineering*, vol. 7, no. 2, pp. 113–128, 2017.

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

[30] C. Helfmeier, C. Boit, D. Nedospasov, and J.-P. Seifert, "Cloning physically unclonable functions," in *IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2013, pp. 1–6.

[31] A. Aysu, N. F. Ghalaty, Z. Franklin, M. P. Yali, and P. Schaumont, "Digital fingerprints for low-cost platforms using MEMS sensors," in *Proceedings of the Workshop on Embedded Systems Security – WESS 2013*. ACM, 2013.

[32] H. Bojinov, Y. Michalevsky, G. Nakibly, and D. Boneh, "Mobile device identification via sensor fingerprinting," *CoRR*, vol. abs/1408.1416, 2014.

[33] T. Van Goethem, W. Scheepers, D. Preuveneers, and W. Joosen, "Accelerometer-based device fingerprinting for multi-factor mobile authentication," in *Engineering Secure Software and Systems*, J. Caballero, E. Bodden, and E. Athanasopoulos, Eds. Springer International Publishing, 2016, pp. 106–121.

[34] T. Hupperich, H. Hosseini, and T. Holz, "Leveraging sensor fingerprinting for mobile device authentication," in *Detection of Intrusions and Malware, and Vulnerability Assessment*, J. Caballero, U. Zurutuza, and R. J. Rodríguez, Eds. Springer International Publishing, 2016, pp. 377–396.

[35] Z. Ba and K. Ren, "Addressing smartphone-based multi-factor authentication via hardware-rooted technologies," in *37th IEEE International Conference on Distributed Computing Systems – ICDCS 2017*, June 2017, pp. 1910–1914.

[36] G. Baldini, F. Dimc, R. Kamnik, G. Steri, R. Giuliani, and C. Gentile, "Identification of mobile phones using the built-in magnetometers stimulated by motion patterns," *Sensors*, vol. 17, no. 4, p. 783, 2017.

[37] I. Amerini, R. Becarelli, R. Caldelli, A. Melani, and M. Niccolai, "Smartphone fingerprinting combining features of on-board sensors," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 10, pp. 2457–2466, Oct. 2017.

[38] S. Dey, N. Roy, W. Xu, R. R. Choudhury, and S. Nelakuditi, "Accelprint: Imperfections of accelerometers make smartphones trackable," *Network and Distributed System Security (NDSS) Symposium*, 2014.

[39] A. Das, N. Borisov, and M. Caesar, "Tracking mobile web users through motion sensors: Attacks and defenses," in *23rd Network and Distributed System Security (NDSS) Symposium*, 2016.

[40] Bosch Sensortec, "BNO055 - Smart Hubs & ASSNs," accessed 2018-03-01. [Online]. Available: https://www.bosch-sensortec.com/bst/products/all_products/bno055

[41] V. Lindroos, M. Tilli, A. Lehto, and T. Motooka, *Handbook of Silicon Based MEMS Materials and Technologies*. Elsevier Inc., 2010.

[42] J. Guajardo, S. Kumar, G.-J. Schrijen, and P. Tuyls, "Physical unclonable functions and public-key crypto for FPGA IP protection," in *International Conference on Field Programmable Logic and Applications (FPL)*, Aug. 2007, pp. 189–195.

[43] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data." *SIAM Journal on Computing*, vol. 38, no. 1, pp. 97–139, 2008, an extended abstract appears in [68].

[44] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[45] R. Maes, V. van der Leest, E. van der Sluis, and F. M. J. Willems, "Secure key generation from biased PUFs," in *Cryptographic Hardware and Embedded Systems – CHES 2015*, ser. LNCS, T. Güneysu and H. Handschuh, Eds., vol. 9293. Springer, Sept. 2015, pp. 517–534.

[46] J. L. Massey, "Guessing and entropy," in *IEEE International Symposium on Information Theory (ISIT)*, 1994.

[47] Y. Dodis, R. Gennaro, J. Håstad, H. Krawczyk, and T. Rabin, "Randomness extraction and key derivation using the cbc, cascade and HMAC modes," in *Advances in Cryptology – CRYPTO 2004*, ser. LNCS, M. K. Franklin, Ed., vol. 3152. Springer, Aug. 2004, pp. 494–510.

[48] H. Krawczyk, "Cryptographic Extraction and Key Derivation: The HKDF Scheme," in *Advances in Cryptology – CRYPTO 2010*, ser. LNCS, T. Rabin, Ed., vol. 6223. Springer, 2010, pp. 631–648.

[49] R. Maes, A. Van Herrewege, and I. Verbauwhede, "PUFKY: A fully functional PUF-based cryptographic key generator," in *Cryptographic Hardware and Embedded Systems – CHES 2012*, ser. LNCS, E. Prouff and P. Schaumont, Eds., vol. 7428. Springer, Sept. 2012, pp. 302–319.

[50] A. Aysu, E. Gulcan, D. Moriyama, P. Schaumont, and M. Yung, "End-to-end design of a PUF-based privacy preserving authentication protocol," in *Cryptographic Hardware and Embedded Systems – CHES 2015*, ser. LNCS, T. Güneysu and H. Handschuh, Eds., vol. 9293. Springer, Sept. 2015, pp. 556–576.

[51] R. Maes, V. van der Leest, E. van der Sluis, and F. M. J. Willems, "Secure key generation from biased PUFs: extended version," *Journal of Cryptographic Engineering*, vol. 6, no. 2, pp. 121–137, 2016, a preliminary version appears in [45].

[52] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby, "A pseudorandom generator from any one-way function," *SIAM Journal on Computing*, vol. 28, no. 4, pp. 1364–1396, 1999.

[53] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1148–1161, 1993.

[54] D. E. Holcomb, W. P. Burleson, and K. Fu, "Power-up SRAM state as an identifying fingerprint and source of true random numbers," *IEEE Transactions on Computers*, vol. 58, no. 9, pp. 1198–1210, Sept. 2009.

[55] M. Claes, V. van der Leest, and A. Braeken, "Comparison of SRAM and FF PUF in 65nm technology," in *Information Security Technology for Applications*, P. Laud, Ed. Springer Berlin Heidelberg, 2012, pp. 47–64.

[56] D. Li, Z. Lu, X. Zou, and Z. Liu, "PUFKEY: A high-security and high-throughput hardware true random number generator for sensor networks," *Sensors*, vol. 15(10), pp. 26 251–26 266, 2015.

[57] F. M. J. Willems, Y. M. Shtarkov, and T. J. Tjalkens, "The context-tree weighting method: basic properties," *IEEE Transactions on Information Theory*, vol. 41, no. 3, pp. 653–664, May 1995.

[58] ——, "Context weighting for general finite-context sources," *IEEE Transactions on Information Theory*, vol. 42, no. 5, pp. 1514–1520, Sept. 1996.

[59] E. Franken and M. Peeters, "Context tree weighting implementation version 0.1," Eindhoven University of Technology, Nov. 2002.

[60] T. M. Cover and J. A. Thomas, *Elements of information theory*. Wiley-Interscience, 2006.

[61] T. Ignatenko, G. j. Schrijen, B. Skoric, P. Tuyls, and F. Willems, "Estimating the secrecy-rate of physical unclonable functions with the context-tree weighting method," in *IEEE International Symposium on Information Theory*, July 2006, pp. 499–503.

[62] S. Katzenbeisser, Ü. Kocabaş, V. Rožić, A.-R. Sadeghi, I. Verbauwhede, and C. 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*, E. Prouff and P. Schaumont, Eds. Springer Berlin Heidelberg, 2012, pp. 283–301.

[63] M. S. Turan, E. Barker, J. Kelsey, K. A. McKay, M. L. Baish, and M. Boyle, "Recommendation for the entropy sources used for random bit generation (second draft)," National Institute of Standards and Technology, Special Publication 800-90B, 2016.

[64] U. M. Maurer, "A universal statistical test for random bit generators," *Journal of Cryptology*, vol. 5, no. 2, pp. 89–105, Jan. 1992.

[65] D. Salomon, "Dictionary methods," in *Data Compression: The Complete Reference*. Springer London, 2007, pp. 171–261.

[66] Github, "SP800-90B Entropy Assessment," accessed 2018-03-05. [Online]. Available: https://github.com/usnistgov/SP800-90B_EntropyAssessment

[67] E. Barker, "Recommendation for key management," National Institute of Standards and Technology, Special Publication 800-57 Part 1 Rev. 4, Jan. 2016.

[68] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Advances in cryptology – EUROCRYPT 2004*. Springer, 2004, pp. 523–540.

[69] Y. Dodis, J. Katz, L. Reyzin, and A. Smith, "Robust fuzzy extractors and authenticated key agreement from close secrets," in *Advances in Cryptology – CRYPTO 2006*. Springer, 2006, pp. 232–250.

[70] L. Reyzin, "Entropy loss is maximal for uniform inputs," Boston University, Technical Report BUCS-TR-2007-011, Sept. 2007.

[71] J. Delvaux, D. Gu, I. Verbauwhede, M. Hiller, and M. M. Yu, "Efficient fuzzy extraction of PUF-induced secrets: Theory and applications," in *Cryptographic Hardware and Embedded Systems – CHES 2016*, ser. LNCS, B. Gierlichs and A. Y. Poschmann, Eds., vol. 9813. Springer, Aug. 2016, pp. 412–431.

[72] L. Carter and M. N. Wegman, "Universal classes of hash functions," *Journal of Computer and System Sciences*, vol. 18, no. 2, pp. 143–154, 1979.

[73] Y. Dodis, R. Gennaro, J. Håstad, H. Krawczyk, and T. Rabin, "Randomness extraction and key derivation using the CBC, cascade and HMAC modes," in *Advances in Cryptology – CRYPTO 2004*, M. Franklin, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 494–510.

[74] R. Neul, U. Gomez, K. Kehr, W. Bauer, J. Classen, C. Doring, E. Esch, S. Gotz, J. Hauer, B. Kuhlmann, C. Lang, M. Veith, and R. Willig, "Micromachined gyros for automotive applications," in *IEEE Sensors*, Oct. 2005, pp. 527–530.

[75] F. Kuypers, *Klassische Mechanik*, 5th ed. Wiley, 1997.

[76] E. Tatar, S. Alper, and T. Akin, "Quadrature-error compensation and corresponding effects on the performance of fully decoupled MEMS gyroscopes," *Journal of Microelectromechanical Systems*, vol. 21, no. 3, pp. 656–667, June 2012.

[77] A. Lahrach, "Bosch IMU in iPhone X," System Plus Consulting, Reverse Costing & Technology Analysis, Jan. 2018.

[78] R. Fraux, "Bosch BMI160 vs ST LSM6DSM," System Plus Consulting, Reverse Costing & Technology Analysis, Dec. 2017.

[79] F. Laermer and A. Schilp, "Method of anisotropically etching silicon," US patent 5,501,893, Mar., 1996.

[80] J. Xu and J. M. Tsai, "A process-induced-frequency-drift resilient 32 kHz MEMS resonator," *Journal of Micromechanics and Microengineering*, vol. 22, no. 10, p. 105029, 2012.

[81] W. T. Hsu and A. R. Brown, "Frequency trimming for MEMS resonator oscillators," in *IEEE International Frequency Control Symposium Joint with the 21st European Frequency and Time Forum*, May 2007, pp. 1088–1091.

[82] G. S. May and C. J. Spanos, "Statistical process control," in *Fundamentals of Semiconductor Manufacturing and Process Control*. John Wiley & Sons, Inc., 2006, ch. 6, pp. 181–227.

[83] D. M. Tanner, A. C. Owen, and F. Rodriguez, "Resonant frequency method for monitoring MEMS fabrication," *Proc.SPIE*, vol. 4980, pp. 4980 – 4980 – 9, 2003.

[84] M. Orshansky, S. R. Nassif, and D. Boning, "Front end variability," in *Design for Manufacturability and Statistical Design: A Constructive Approach*. Springer US, 2008, ch. 2, pp. 11–41.

[85] O. Willers, M. Curcic, and H. Seidel, "Fingerprinting MEMS gyroscopes," in *Proceedings of the 11th Smart Systems Integration (SSI)*, 2017, pp. 133 – 140.

[86] N. Yazdi, F. Ayazi, and K. Najafi, "Micromachined inertial sensors," *Proceedings of the IEEE*, vol. 86, no. 8, pp. 1640–1659, Aug. 1998.

[87] P. J. Petersan and Steven M., "Measurement of resonant frequency and quality factor of microwave resonators: Comparison of methods," *Journal of Applied Physics*, vol. 84, no. 6, pp. 3392 – 3402, 1998.

[88] J. S. Burdess, A. J. Harris, D. Wood, R. J. Pitcher, and D. Glennie, "A system for the dynamic characterization of microstructures," *Journal of Microelectromechanical Systems*, vol. 6, no. 4, pp. 322–328, Dec. 1997.

[89] C. Rembe, R. Kant, and R. S. Muller, "Optical measurement methods to study dynamic behavior in MEMS," in *Proc.SPIE*, vol. 4400, 2001, pp. 127–137.

[90] A. Cigada, E. Leo, and M. Vanali, "Electrical method to measure the dynamic behaviour and the quadrature error of a MEMS gyroscope sensor," *Sensors and Actuators A: Physical*, vol. 134, no. 1, pp. 88 – 97, 2007.

[91] Agilent Technologies, Inc., *Agilent Impedance Measurement Handbook*, 4th ed., June 2009.

[92] Mgc.co.jp, "Non-haloganated low CTE BT resin laminate for IC plastic packages," accessed 2017-07-24. [Online]. Available: http://www.mgc.co.jp/eng/products/lm/btprint/lineup/hfbt.html

[93] Sumibe.co.jp, "Epoxy resin molding compounds for encapsulation of semiconductor devices," accessed 2017-07-24. [Online]. Available: http://www.sumibe.co.jp/english/product/it-materials/epoxy/sumikon-eme/

[94] JEDEC Standard - JESD22-A103E, "High Temperature Storage Life," JEDEC Solid State Technology Association, Oct. 2015, Revision of JESD22-A103D, Dec. 2010.

[95] JEDEC Standard - JESD22-A104D, "Temperature Cycling," JEDEC Solid State Technology Association, Mar. 2009, Revision of JESD22-A104C, May 2005.

[96] O. Willers, C. Huth, J. Guajardo, and H. Seidel, "MEMS gyroscopes as Physical Unclonable Functions," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security – CCS 2016*. ACM, 2016, pp. 591–602.

[97] ——, "Impact of inter-wafer variations on MEMS fingerprints," in *Proceedings of the 12th Smart Systems Integration*, 2018, pp. 261 – 268.

[98] P. Tuyls, G. J. Schrijen, B. Skoric, J. van Geloven, N. Verhaegh, and R. Wolters, "Read-proof hardware from protective coatings," in *Cryptographic Hardware and Embedded Systems - CHES 2006*, ser. LNCS,

L. Goubin and M. Matsui, Eds., vol. 4249. Springer, October 10-13, 2006, pp. 369–383.

[99] Y.-J. Chang, W. Zhang, and T. Chen, "Biometrics-based cryptographic key generation." in *IEEE International Conference on Multimedia and Expo (ICME)*, vol. 3, 2004.

[100] M. Gardner, "The binary gray code," in *Knotted Doughnuts and Other Mathematical Entertainments*. W. H. Freeman and Co., 1986, ch. 2.

[101] C. Chen, R. N. J. Veldhuis, T. A. M. Kevenaar, and A. H. M. Akkermans, "Multi-bits biometric string generation based on the likelihood ratio," in *First IEEE International Conference on Biometrics: Theory, Applications, and Systems*, Sept. 2007, pp. 1–6.

[102] A. W. Bowman and A. Azzalini, *Applied Smoothing Techniques for Data Analysis*. New York, Oxford University Press Inc., 1997.

[103] J. M. G. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates," in *Proceedings of the 4th International Conference on Audio-and Video-Based Biometrie Person Authentication – AVBPA 2003*, ser. LNCS, J. Kittler and M. S. Nixon, Eds., vol. 2688. Springer, June 2003, pp. 393–402.

[104] W. N. Sharpe, M. A. Eby, and G. Coles, "Effect of temperature on mechanical properties of polysilicon," in *Transducers '01 Eurosensors XV: The 11th International Conference on Solid-State Sensors and Actuators*, E. Obermeier, Ed. Springer Berlin Heidelberg, 2001, pp. 1338–1341.

[105] G. Marsaglia, "Ratios of normal variables," *Journal of Statistical Software*, vol. 16, 2006.

[106] J. L. Hodges and L. L. Cam, "The Poisson approximation to the Poisson binomial distribution," *The Annals of Mathematical Statistics*, vol. 31, no. 3, pp. 737–740, Sept. 1960.

[107] R. Maes, V. van der Leest, E. van der Sluis, and F. Willems, "Secure key generation from biased PUFs: extended version," *Journal of Cryptographic Engineering*, vol. 6, no. 2, pp. 121–137, Jun 2016.

[108] A. Aysu, E. Gulcan, D. Moriyama, P. Schaumont, and M. Yung, *End-To-End Design of a PUF-Based Privacy Preserving Authentication Protocol*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 556–576.

[109] G. J. Schrijen and V. van der Leest, "Comparative analysis of sram memories used as puf primitives," in *2012 Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2012, pp. 1319–1324.

[110] E. Barker and A. Roginsky, "Transitions: Recommendation for transitioning the use of cryptographic algorithms and key lengths," National Institute of Standards and Technology, Special Publication 800-131A Rev. 1, Nov. 2015.

[111] M. Wolfer, C. Hepp, M. Reimann, U. Kunz, and C. Rembe, "Testing a capped mems gyroscope by an infrared technique," in *2015 Transducers - 2015 18th International Conference on Solid-State Sensors, Actuators and Microsystems (TRANSDUCERS)*, June 2015, pp. 2240–2243.

[112] E. M. Lawrence, K. E. Speller, and D. Yu, "MEMS characterization using laser Doppler vibrometry," in *SPIE 4980, Reliability, Testing, and Characterization of MEMS/MOEMS II*, 2003, pp. 4980 – 4980 – 12.

[113] M. Bauer, F. Ritter, and G. Siegmund, "High-precision laser vibrometers based on digital Doppler signal processing," in *SPIE 4827, Fifth International Conference on Vibration Measurements by Laser Techniques: Advances and Applications*, 2002, pp. 4827 – 4827 – 12.

[114] S. N. Pulujkar, "Physical attacks on MEMS PUFs," Master's thesis, Robert Bosch GmbH Corporate Sector Research and Advance Engineering, Hochschule Furtwangen University, 2017.

[115] S. Skorobogatov, "Physical attacks and tamper resistance," in *Introduction to Hardware Security and Trust*, M. Tehranipoor and C. Wang, Eds. Springer, 2012.

[116] FIPS PUB 140-2, "Security requirements for cryptographic modules," National Institute of Standards and Technology, May 2001.

[117] X. Ma, D. G. Yang, and G. Q. Zhang, "Decapsulation methods for Cu interconnection packages," in *13th International Conference on Electronic Packaging Technology High Density Packaging*, Aug. 2012, pp. 1387–1391.

[118] S. Tumanski, "Induction coil sensors – a review," *Measurement Science and Technology*, vol. 18, no. 3, p. R31, 2007.

List of Abbreviations

AC	alternating current
ASIC	application-specific integrated circuit
BCH	Bose-Chaudhuri-Hocquenghem
CMOS	complementary metal–oxide–semiconductor
CMP	chemical mechanical polishing
CRP	challenge-response-pair
CTW	context tree weighting
DC	direct current
DRIE	deep reactive ion etching
DUT	device under test
ECC	elliptic-curve cryptography
EER	equal error rate
ESC	electronic stability control
FAR	false acceptance rate
FEM	finite element method
FFC	finite-field cryptography
FFT	fast Fourier transform
FRR	false rejection rate
HF	high frequency
HKDF	HMAC-based key derivation function
HMAC	hash-based message authentication code
HTSL	high temperature storage life

HW Hamming weight

IC integrated circuit

IFC integer-factorization cryptography

IID independent and identically distributed

IoT Internet of Things

IP intellectual property

IR infrared

JEDEC Solid State Technology Association

KDF key derivation function

LDV laser-Doppler vibrometry

LGA land grid array

LRS longest repeated substring

MCB most common byte

MCW most common in window

MEMS microelectromechanical systems

MMC Markov model with counting

NI National Instruments

NIST National Institute of Standards and Technology

NVM non-volatile memory

PCB printed circuit board

PDF probability density function

PWM pulse-width modulator

PUF physical unclonable function

REP repeatability at RT

RFID radio-frequency identification

RNG random number generator

RO ring oscillator

RT room temperature

SEM scanning electron microscope

SIP system in package

SNR signal-to-noise-ratio

SRAM static random access memory

TC temperature cycling

TD temperature dependency

µC microcontroller

List of Figures

List of Tables