Integration of Physically Unclonable Functions (PUFs) in CMOS

Von der Fakultät für Ingenieurwissenschaften
Abteilung Elektrotechnik und Informationstechnik
der Universität Duisburg-Essen

zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften

genehmigte Dissertation

von
Benjamin Willsch

aus
Preetz

Gutachter: Prof. Dr.-Ing. Holger Vogt
Gutachter: Prof Dr. rer. nat. Gregor Schiele

Tag der mündlichen Prüfung: 24.04.2019

DOI: 10.17185/duepublico/70433
URN: urn:nbn:de:hbz:464-20190822-072454-3

Abstract

Silicon-based Physically Unclonable Functions (PUFs) are a new type of cryptographic primitive designed to extract unique key signatures from random process-related variations in the physical quantities of semiconductor devices. The signatures are intrinsically hidden inside the complex manufacturing history of the silicon material making it difficult for an adversary to steal the information enclosed in the microstructure of a PUF device. Thus, the core design provides a secure, forgery-proof storage place for device-specific secrets that does not require peripheral anti-temper protection making the usage of PUFs appealing for low-cost chip authentication purposes and lightweight cryptographic key applications.

Although the concept realization of a PUF is simple in theory, implementing a design in hardware is a challenging task due to the physical nature of the quantities from which the keys are derived. The randomness and uniqueness of the extracted keys are closely related to the uniformity of the manufacturing process. Furthermore, the signature generation is error-prone to noise and changes in the operating conditions of a PUF, thus potentially lowering the ability to distinguish different devices and limiting the usage of the raw signatures for cryptographic key applications.

Since the randomness, uniqueness and stability of a PUF-based key generator are tied to the fabrication process of a device, security-relevant features inevitably depend on the hardware implementation of a PUF. The main subject of this thesis is to analyze, model and resolve this form of implementation dependency in a systematic and mathematically convenient way. For this purpose, a probabilistic model framework is developed to investigate the randomness, uniqueness and reproducibility of binary keys derived from the electrical characteristics of a PUF device. Starting from a random field description of the electrical characteristics of a PUF, the aforementioned performance metrics are modeled in terms of de-

terministic and stochastic process variations. The model is used to establish a link between critical design parameters of a PUF concept such as the key entropy and error correction capacity to the distributional properties of the fabrication process of a PUF design. The stochastic process representing the manufacturing conditions is approximated by parametric probability distributions, whose parameter values can be readily estimated from experimental data using variation decomposition techniques common in spatial statistics. The process model allows to accurately assess the performance of a given PUF concept fabricated in a particular production line and manufacturing process as is demonstrated for the well-known ring oscillator PUF design implemented on FPGAs.

Motivated by the analysis results, a simple pair selection scheme is proposed to increase the randomness and reproducibility of keys extracted from arbitrary analog PUF devices. A high degree of randomness is achieved by pairing PUF devices featuring a similar systematic mismatch in their electrical properties in a way that minimizes the deterministic bias of individual key-bits. In order to increase the stability of the generated keys, higher order pairs are formed which prevent noise induced bit-errors to occur during the key generation process. The length and number of keys obtainable by this quantization procedure scales linearly with the size of a PUF instance. The size limitation is overcome by introducing a boolean function which recursively performs XOR operations on the output-bits of a PUF to produce an exponential number of keys.

Applicability of the two concepts is verified by experimental data obtained for various devices including transistors, resistors, ring oscillators and image sensor based PUFs, implemented and fabricated in different processing technologies. Compared to state-of-the-art solutions evaluated for the same datasets, the randomness, uniqueness and stability of the generated keys could be significantly improved. Extensive measurements demonstrate that the performance of the developed concepts is nearly independent of the used device type and implementation form, whereas existing solutions fail to produce consistent results across the range of tested devices.

Zusammenfassung

Siliziumbasierte Physikalisch Unklonbare Funktionen (PUFs) stellen eine neue Art kryptographischer Primitive dar, welche dazu konzipiert wurden, einzigartige Schlüsselsignaturen aus fertigungsbedingten Schwankungen physikalischer Kenngrößen von Halbleiterbauelementen zu extrahieren. Die Signaturen sind in der komplexen Fertigungshistorie des prozessierten Halbleitermaterials vorborgen, wodurch es einem Angreifer erschwert wird, die in der Mikrostruktur eines PUFs enthaltenen Informationen auszulesen. Als solches bietet das grundlegende Konstruktionsprinzip ein fälschungssicheres Speichermedium für gerätespezifische Geheimnisse, welches keiner zusätzlichen Vorrichtung zum Manipulationsschutz bedarf. Da periphere Schutzvorkehrungen eingespart werden können, ist die Nutzung von PUFs insbesondere für Anwendungen im Bereich der Low-Cost Chip Authentisierung und für die Umsetzung kryptographischen Funktionen auf technisch limitierten Plattformen vorteilhaft.

Obwohl sich die konzeptionelle Realisierung eines PUF in der Theorie als einfach erweist, stellt die Hardware-Implementierung aufgrund der physikalischen Eigenschaften schlüsselrelevanter Kenngrößen eine Herausforderung dar. Die Zufälligkeit und Einzigartigkeit der abgeleiteten Schlüssel steht in einem engen Bezug zur Uniformität des Fertigungsprozesses. Durüberhinaus ist das Auslesen der Schlüsselsignaturen anfällig gegenüber Messrauschen und Änderungen in den Betriebsbedingungen, infolgedessen die Unterscheidbarkeit verschiedener PUF Instanzen abnimmt und die unprozessierten Signaturen nur bedingt für die Generierung kryptographischer Schlüssel geeignet sind.

Weil die Zufälligkeit, Einzigartigkeit und Stabilität eines PUF-basierten Schlüsselgenerators an den Fertigungsprozess des PUF gebunden sind, hängen die sicherheitsrelevanten Eigenschaften eines PUF zwingenderweise von dessen Implementierungsform ab. Der Hauptthemenbereich dieser Arbeit behandelt die sys-

tematische Analyse, Modellierung und Auflösung dieser Implementierungsabhängigkeit. Zu diesem Zweck wurde ein probabilistisches Model entwickelt, in dessen Rahmen sich die Zufälligkeit, Einzigartigkeit und Stabilität binärer Schlüssel untersuchen lassen, welche aus den elektrischen Kenngrößen eines PUF abgeleitet werden. Ausgehend von einer Zufallsfeldbeschreibung der elektrischen Kenngrößen eines PUFs wird gezeigt, wie sich gängige Performance-Metriken unter dem Einfluss deterministischer und stochastischer Prozessschwankungen modellieren lassen. Das Model wird genutzt, um wesentliche Design-Parameter wie die Schlüssel-Entropie und die Fehlerkorrekturkapazität eines PUF-Konzepts mit den stochastischen Verteilungseigenschaften des zugrundeliegenden Fertigungsprozess in Verbindung zu bringen. Der stochastische Prozess, welcher die Fertigungsbedingungen eines PUFs modelliert, wird mittels parametrischer Wahrscheinlichkeitsverteilungen approximiert. Die Modelparameter können unter Verwendung in der räumlichen Statistik geläufigen Variationsanalyseverfahren unmittelbar aus den experimentellen Daten eines PUFs bestimmt werden. Die modellhafte Darstellung des Prozesses ermöglicht es, eine genaue Abschätzung der Performance eines PUF-Konzepts vorzunehmen, welches in einer bestimmten Fertigungslinie und einem bestimmten Herstellungsprozess fabriziert wurde, wie am Beispiel einer FPGA-basierten Implementierung des wohlbekannten Ring-Oszillator PUF Designs gezeigt wird.

Motiviert durch die Analyseergebnisse wird ein einfaches Paarauswahlverfahren entwickelt, um den Schlüsselgenerierungsmechanismus arbiträrer analoger PUF-Designs hinsichtlich der Schlüssel-Zufälligkeit und Reproduzierbarkeit zu verbessern. Ein hohes Maß an Zufälligkeit wird erzielt, in dem paare von PUF-Strukturen ausgewählt werden, die einen vergleichbaren systematischer Mismatch in ihren elektrischen Eigenschaften aufweisen, sodass der deterministische Bias individueller Schlüssel-Bits minimiert wird. Um die Stabilität der Schlüssel zu erhöhen, werden Paare höherer Ordnung geformt, welche das Auftreten von Bit-Fehlern während der Schlüsselgenerierung unterbinden. Die Länge und Anzahl der Schlüssel skaliert linear mit den Abmessungen einer PUF-Instanz. Um die flächenbedingte Limitierung des Verfahrens aufzuheben, wird eine boolsche Funktion definiert, welche rekursiv XOR-Operationen auf die Ausgabebits eines PUFs anwendet und so exponentiell viele Schlüssel erzeugt.

Die Anwendbarkeit beider Konzepte wurde experimentell für eine Vielzahl

verschiedener Halbleiterbauelemente verifiziert. Die getesteten Bauelemente umfassen Transistoren, Widerstände, Ring-Oszillatoren und Sensor-basierte PUFs, welche in unterschiedlichen Fertigungstechnologien implementiert und produziert wurden. Im Vergleich zu aktuell genutzten Verfahren, deren Performance für die gleichen Datensätze evaluiert wurde, konnten deutliche Steigerungen in Bezug auf die Zufälligkeit, Einzigartigkeit und Stabilität der generierten Schlüssel erzielt werden. Umfangreiche Messungen haben gezeigt, dass die Performance der neuentwickelten Konzepte nahezu unabhängig von der Implementierungsform des PUFs ist, wohingegen die Analyseergebnisse der anderen Verfahren zum Teil deutlich zwischen den getesteten Designs variieren.

Contents

Chapter 1

Introduction

1.1 Motivation

New concepts and applications emerging from current digitalization trends such as the Internet of Things and Industry 4.0 strive to fully embed physical systems into the digital world.

Essential to the practical realization of these paradigms are smart devices capable of communicating and exchanging information with each other. Ubiquitous usage of interconnected smart devices allows real-time monitoring and processing of telemetric data opening up countless ways to improve all areas of life including home automation [Piya 13, Soli 13], energy management [Shro 15, Shro 14], transportation [Sher 15, Boja 14] and health care [Isla 15, Smor 17]. For example, having access to real-time information on the living conditions and vital statistics of a person allows to provide remote health diagnosis and support to the elderly, people with mobility disabilities and people living in rural areas with limited access to medical facilities.

At the same time, knowledge of personal and sensitive data gives rise to privacy and security concerns. Personalized information can be used to generate movement and consumer profiles of individuals without their knowledge and consent [Helb 16, Eske 16]. Misuse of the data might result in identity theft [Vida 14, Mahm 15] and manipulation of the data or the devices themselves can pose a real threat to the physical safety of its user. One alarming cyber-physical attack scenario was demonstrated by the authors of [Mill 15] who leveraged security vulnerabilities in the network system of a motorized vehicle to affect steering

and braking of the car. In consequence of the exploit, the manufacturer had to recall 1.4 million automobiles susceptible to this attack [Gree 15].

In order to secure communication channels and to prevent attacks on interconnected devices, not only strong cryptographic protocols are needed but also secure hardware components to implement them. The design of such hardware components and systems is a challenging task, because many casual smart devices (watch, phone, car, etc.) are used in an untrusted environment leaving them prone to physical attacks carried out by an adversary having unrestricted access to these items. Moreover, protecting sensitive data stored in nonvolatile memory against physical attacks requires elaborate and possibly costly anti-temper measures [Mitt 18] which may not be suitable for compact resource-restricted systems.

One promising solution to these issues are Physically Unclonable Functions (PUFs) – a new form of hardware-based cryptographic primitives – specifically designed for low-cost device authentication and lightweight cryptographic key generation. PUFs harness random manufacturing-related differences between nominally identically fabricated physical objects to produce truly random and unique signatures. A distinctive feature of the PUF concept is that the device-specific signature is permanently imprinted into the internal physical characteristics of the PUF hardware during the fabrication process. Since a cryptographic key can be generated from this unique signature whenever needed, no secure storage is required. Furthermore, any attempt to acquire the key by means of physically probing the device might alter the properties of the PUF system, thus effectively changing the key. This form of intrinsic hardware protection layer serves as a convenient anti-temper measure against physical attacks. An adversary trying to fabricate a PUF counterfeit without knowing its signature key faces similar problems. Replicating the processing conditions under which a PUF was produced is a highly difficult task due to the stochastic nature of the fabrication process. Thus, creating a physical clone of a PUF device that outputs the same signature is considered unfeasible in practice, even for the original manufacturer. [Herd 14]

Because of the aforementioned properties, using PUFs to complement and strengthen existing security systems is conceivable, especially with regard to the widespread application of low-cost smart devices [Baba 19] and sensor systems [Seli 11].

1.2 Problem Statement

Despite many appealing features, the core working principle of a PUF suffers from a number of conceptual flaws closely related the implementation and physical realization of a PUF design in hardware.

Implementation dependence

The functionality of conventional cryptographic primitives typically relies on hard-to-solve mathematical problems whose security features are independent of their hardware implementation[1]. In contrast, the security features of a PUF are bound to its hardware implementation. [Katz 12]

Because of this implementation dependence the uniqueness and randomness of the extracted PUF signatures are affected by numerous processing-related factors including the manufacturing equipment, manufacturing tolerances and the manufacturing process. Hence, the performance of a given PUF design does not only depend on its realization, e.g. FPGA vs. microprocessor, but also at which manufacturing site and in which technology it was fabricated. Therefore, licensing or transferring a PUF design to a different manufacturer or technology node is likely to yield varying performance results (Sec. 2.4.3 and 2.5.3). Even the performance of a PUF design adjusted to a specific fabrication technology is likely to change over time due to perpetual modifications to the production process. In particular, process optimization has a detrimental effect on the randomness of a PUF response as optimization procedures typically aim to reduce process variations to increase production yield.

Systematic process variations

The uniqueness and randomness of a PUF response might be further reduced due to the contribution of systematic process variations to the overall process variability [Yin 14]. Systematic process variations such as temperature gradients or optical aberration effects that occur during deposition and lithographic process steps, respectively, are deterministic by nature [Bohm 13]. These types of varia-

[1] As long as no vulnerabilities are introduced by erroneous hardware design, the computational effort to break a given cryptographic algorithm by means of a brute force attack remains the same, whether it is implemented on an FPGA, a microprocessor or any other system.

tions can induce systematic patterns in a PUF response or yield correlation effects between response-bits [Will 18]. Any form of structure imposed on the response generation process can potentially be exploited by an attacker to compromise the security provided by a PUF [Maes 13].

Measurement noise and environmental effects

Another problem related to the physical key generation process of a PUF is its susceptibility to noise and environmental effects yielding irreproducible responses that differ in one or more bits between subsequent queries of the same PUF device. Unlike keys computed by deterministic cryptographic algorithms, PUF responses are extracted from the measurements of one or more physical device characteristics. The measurement process is prone to measurement noise and the physical quantities are subject to change under varying operating conditions of a PUF, e.g. changes in the operating temperature or aging of the PUF. Both measurement noise and environmental effects alter the response of a PUF, possibly resulting in a key that changes every time a readout operation is performed. Therefore, some form of error correction procedure has to be employed for any type of application that requires a fully reproducible key. [Herd 14]

The use of error correction codes is sensible only for a fixed, limited number of bit-errors. For a large bit-error rate[2], the increased areal overhead and power consumption associated with a complex correction procedure limits the usability of a PUF in lightweight cryptographic applications. [Bohm 13]

However, from a statistical point of view, up to 50 % of the response-bits can be erroneous on average when generated under variable operating conditions, hence rendering conventional error correction procedures useless in these extreme cases.

1.3 Thesis Goals

This thesis aims to provide general design concepts on how to construct PUFs that output reliable high-entropy keys independent from their chosen implementation form.

[2]The bit-error rate is defined as the fraction of erroneous response-bits [Bohm 13].

For this purpose two generic PUF concept are proposed. The first PUF concept allows to extract cryptographic keys from arbitrary noisy physical characteristics. A high degree of randomness, uniqueness and reproducibility of the extracted keys is achieved by utilizing a higher order differential quantization scheme. The second PUF concept is designed to enhance the number of keys obtainable from a fixed set of physical parameters. Combinations of the internal parameters of a PUF device are linked by a nonlinear function to produce exponentially many keys that are unique to a given PUF instance.

The performance of the presented concepts is assessed in terms of standard metrics used to quantify the randomness, uniqueness and reproducibility of PUF responses. The metrics are evaluated based on extensive experimental tests carried out for four semiconductor device types fabricated in different process technologies to demonstrate implementation independence of the performance.

Generalization and out-out sample prediction of the experimental test results are achieved by an in-depth analysis of the stochastic properties of the designs. To this end, a new comprehensive statistical model is developed. The model explains the working principle of a PUF in terms of the underlying distributional parameters of the physical quantities from which the secret keys are derived. Within this framework experimental proof is given that both concepts produce keys suitable for the use in cryptographic applications.

Chapter 2

Physically Unclonable Functions (PUFs)

This chapter briefly introduces the basic concept, working principle, properties and performance metrics relevant to PUFs. All of these topics has been covered exhaustively in the literature and an in-depth view on the different subject matters can be found in [Maes 13, Mait 13, Hori 10, Pehl 14, Armk 11] among others. Central ideas presented in the aforementioned references are described and summarized in the following sections.

2.1 Working Principle

PUFs can be thought of as a physical system characterized by some intrinsic parameters β that accepts a challenge C as input and generates a random response R as output (Fig. 2.1). The applied challenge is used to configure the physical

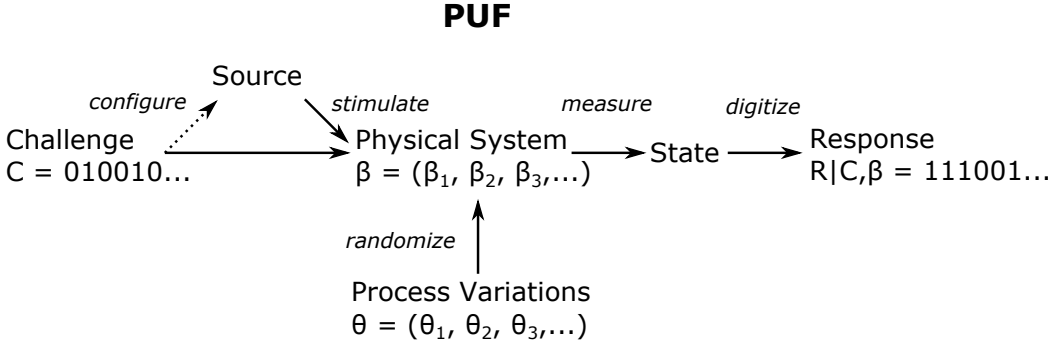


Figure 2.1: Working principle of a generic PUF. Adapted from [Armk 11].

system. After configuration, some form of stimulus is used to measure the current state of the system. Due to the effect of random process variations Θ, the states of nominally identically fabricated systems differ from one another in a random fashion. From these measured differences, a randomly distributed binary response is extracted which is specific to the applied challenge and the physical state of the system. Additional post-processing steps such as privacy amplification and error correction procedures might be employed to increase the randomness and reproducibility of noisy raw responses (Sec. 2.4.1). [Armk 11]

Examples of semiconductor devices used to implement PUFs and their related physical characteristic from which binary responses are extracted will be presented in Sec. 2.4 and 2.5.

2.2 Enrollment and Authentication Protocol

Since the responses of uninitialized PUFs are unknown by design, all PUF devices have to undergo an enrollment phase after fabrication. During the enrollment phase a set of challenge-response pairs (CRPs) is recorded at the manufacturing site and stored in a secure database. Once a PUF instance has been deployed in the field, the authenticity of the device is checked by reconstructing one or more CRPs and comparing them with the prerecorded values in the database. If the difference to the CRPs in the database is sufficiently small, then the authenticity of the device ID is confirmed. [Suh 07]

The closeness of two responses for a given challenge is typically measured in terms of the number of differing bits, i.e. the Hamming distance (HD) between the two binary PUF keys [Bohm 13]. A basic sketch of the enrollment and authentication process is depicted in Fig. 2.2. More advanced protocols are described in [Bohm 13].

2.3 Properties and Performance Metrics

Since PUFs are primarily designed for the use in device authentication tasks and cryptographic applications, the set of CRPs extracted from a PUF must be random and unique to each hardware instance of a given PUF design. In this context, the randomness property refers to the inability to correctly guess

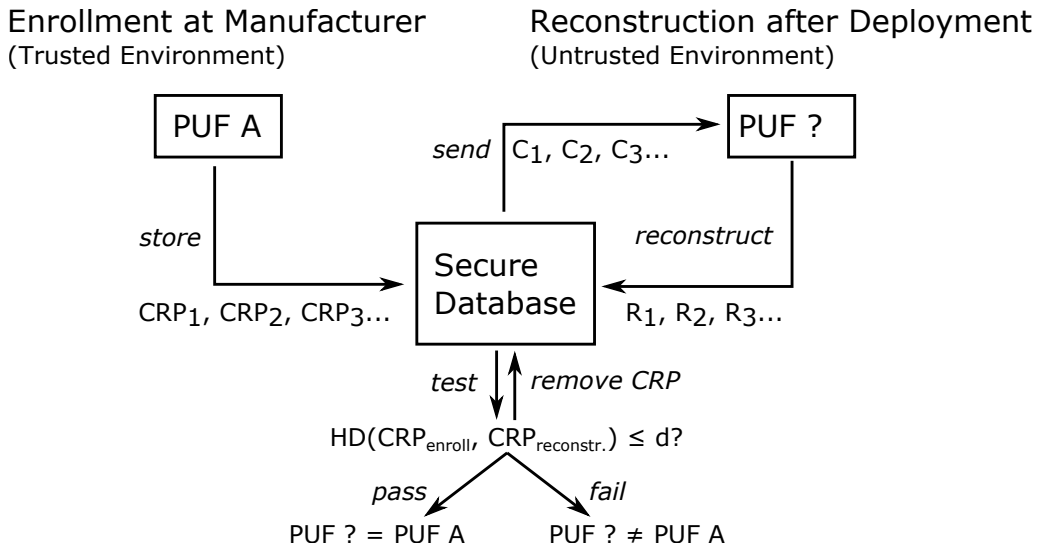


Figure 2.2: Basic enrollment and authentication protocol. Adapted from [Bohm 13].

an unknown CRP with a probability greater than given by pure chance. Once a response was recorded during the enrollment phase, regeneration of the same CRP should always yield the same key, i.e. the response should be fully reproducible. If a response is too noisy, unique identification of a PUF instance based on its CRPs is not possible. [Katz 12]

Additional security relevant properties might be considered, if device authentication involves the public exchange of CRPs. In order to limit the predictability of CRPs, each response of a PUF instance should strongly depend on the corresponding input challenge, i.e. a small change to the challenge should effect the state of all response-bits in a random fashion [Ganj 16, Majz 08b]. Similarly, the challenge-response mechanism should be sufficiently complex to prevent machine learning attacks aiming to predict unobserved responses from known CRPs by modeling the internal structure of a PUF [Ruhr 10].

The security-related properties of a PUF can be quantified in terms of commonly used metrics that can be calculated from observed CRPs of a PUF – independent of its design and implementation form. Since no knowledge about the internal working of the PUF systems is required, the metrics defined in the following subsections provide simple means to evaluate and compare the performance of competing PUF designs in a unified way. [Mait 13]

2.3.1 Randomness

The randomness of a response is measured in terms of the Shannon entropy of the underlying hardware random source used to implement a PUF design. The joint Shannon entropy $H(R)$ of an N-bit PUF response conditional on some challenge C is determined by the equation:

$$H(R) = -\sum_{R \in \{0,1\}^N} P(X = R) \log_2 P(X = R), \tag{2.1}$$

where $P(X = R)$ denotes the probability that a random realization of a PUF instance generates the response $R \in \{0,1\}^N$ given C. [Pehl 14]

The response space for $N = 4$ is graphically illustrated in Fig. 2.3. Each

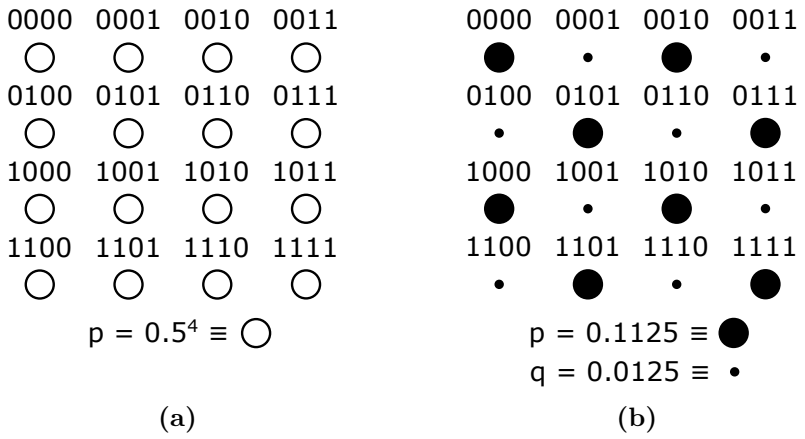


Figure 2.3: Conceptual representation of (a) uniformly and (b) nonuniformly distributed responses. In case of uniformly distributed responses, each response value is equally likely to be extracted from the realizations of a PUF design. For nonuniformly distributed responses, some response values are more likely to occur than others (here indicated by different point sizes).

point marks one of the $2^4 = 16$ possible PUF responses. Figure 2.3a displays the case that all responses are equally likely to be extracted from a PUF design, i.e. the probability mass $P(X = R)$ is distributed uniformly across the entire response space. In this instance, the probability that an attacker correctly guesses a CRP equals $p = 2^{-N}$ which is close to zero for typical response lengths $N \geqslant 64$ [Armk 11]. In contrast, if certain bit-combinations are more frequently observed than others, i.e. a larger fraction of PUF realizations outputs these bit-combinations, then the probability mass is centered at specific locations or parts

of the response space as illustrated in Fig. 2.3b. In this case, the response values are nonuniformly distributed across $\{0,1\}^N$. The unequal distribution of the probability mass imposes additional structure on the CRPs that can be exploited by an attacker [Maes 13]. For example, a priori knowledge of the response bias depicted in Fig. 2.3b suggests that choosing among one of the responses indicated by a larger dot increases the odds of correctly guessing a CRP. In this context the Shannon entropy can be thought of as the number of bits required to encode the information content included in the structure of the response space. An evenly populated response space of dimension 2^N requires N bits to encode all response states. Any form of structure reduces the information content contained in the responses, thus the response space can be compressed and encoded by fewer bits.

If the response-bits are independently distributed, the overall response entropy is simply the sum of the entropy contributed by the individual bits r_i, $i = 1, 2, ..., N$, of a response R[1]:

$$H(R) = \sum_{i=1}^{N} H(r_i) \tag{2.2}$$

with single-bit entropy values

$$H(r_i) = -p_i \log_2 p_i - (1 - p_i) \log_2 (1 - p_i) \tag{2.3}$$

and probabilities p_i, $i = 1, 2, ..., N$, that the i-th response-bit equals a logical one. The maximum entropy is attained in the case of independent and uniformly distributed response-bits, i.e. $p_i = 0.5 \, \forall i = 1, 2, ..., N$, as is immediately apparent from Eq. (2.2) and (2.3). In this instance, $H(R)$ equals the number of response-bits N as previously stated.

The set of single-bit probabilities $\{p_i : i = 1, 2, ..., N\}$ required to compute the response entropy defined in Eq. (2.2) are typically estimated in terms of the uniformity and bit-aliasing measures described in the two sections below. Both measures can be directly calculated from the raw responses of one or more PUFs providing empirical upper bounds on the amount of entropy extractable from a particular design implementation. [Maes 13]

[1] In case of independently distributed response-bits r_i, $i = 1, ..., N$, the probability $P(X = R)$ factorizes into the product $\prod_{i=1}^{N} P(X_i = r_i)$ of the individual bit-probabilities $P(X_i = r_i)$. Substituting the product of probabilities into Eq. (2.1) and summing over the states of the N response-bits yields Eq. (2.2).

Uniformity

The uniformity of a PUF response $R = (r_1, r_2, ..., r_N) \in \{0,1\}^N$ is equal to the mean value of the response [Mait 13][2], i.e.

$$x_U = \frac{1}{N} \sum_{i=1}^{N} r_i \,. \tag{2.4}$$

Assuming the bias towards either logical state to be the same for all response-bits then the uniformity can be taken as estimate $\hat{p} \equiv x_U$ of the probabilities p_i such that $\hat{p} = \hat{p}_i$ holds for $i = 1, 2, ..., N$. Computation of x_U is visualized in Fig. 2.4.

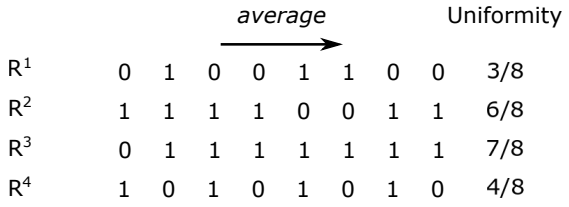


Figure 2.4: Computation of the response uniformity.

Bit-Aliasing

The bit-aliasing measure equals the set of mean values $x_{A,i}$ computed for each bit-position i, $i = 1, 2, ..., N$, of a response, where the average is taken over a number of M PUF responses $R^m \in \{0,1\}^N$, $m = 1, 2, ..., M$ [Mait 13][3]:

$$x_{A,i} = \frac{1}{M} \sum_{m=1}^{M} r_i^m \,. \tag{2.5}$$

The bit-aliasing values serve as bitwise estimates of the probabilities $\hat{p}_i = x_{A,i}$, $i = 1, 2, ..., N$, provided that the bias of each response-bit is the same for all M responses. An example of the summation carried out over the different bit-positions of a response is shown in Fig. 2.5.

2.3.2 Uniqueness

The uniqueness of a set of CRPs obtained from one or more PUF instances is characterized by their pairwise Hamming distance, i.e. the number of bits that

[2]Note that the uniformity measure defined in the reference is given in percent.

[3]Note that the bit-aliasing measure defined in the reference is given in percent.

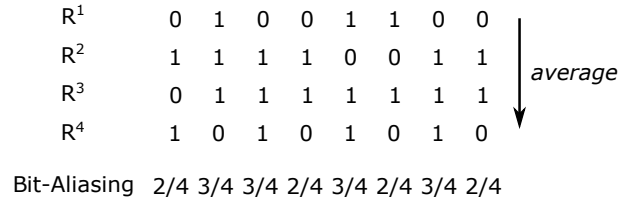


Figure 2.5: Computation of the response bit-aliasing.

any two PUF responses differ from one another. Therefore, the uniqueness of two N-bit responses R^u and R^v can be expressed by their respective inter-Hamming distance $HD_{Inter}(R^u, R^v)$ [Mait 13][4]:

$$HD_{Inter}(R^u, R^v) = \sum_{i=1}^{N} r_i^u \oplus r_i^v \,. \tag{2.6}$$

The symbol $\oplus$ corresponds to the bitwise XOR operation of two boolean values. The subscript i indicates that the XOR operation is applied to the i-th bit of the two responses.

Calculation of the pairwise inter-Hamming distances is depicted in Fig. 2.6. Since the response values are randomly distributed, the inter-Hamming distance

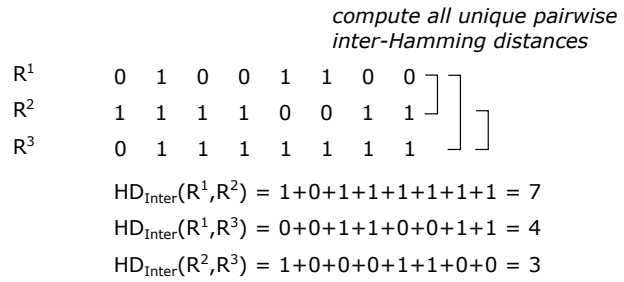


Figure 2.6: Computation of the response inter-Hamming distance.

itself has to be considered as a random variable. In the ideal case of independent and uniformly generated response-bits, the inter-Hamming distances calculated for a large number of CRPs follow a binomial distribution $B(N, 0.5)$ characterized by the response length N and probability $p_{unique} = 0.5$ that a pair of response-bits

[4]In the reference given, the uniqueness is defined as the average inter-Hamming distance of a number of CRPs. In this work, the uniqueness metric is generally evaluated in terms of the distributional properties of the pairwise inter-Hamming distances.

is unique[5]. Since the expected value of the Binomial distribution is equal to $N/2$, two PUF-responses should differ in half of their response-bits on average.

2.3.3 Reproducibility

The reproducibility of a PUF response R is evaluated in terms of the Hamming distance between a reference response R^{ref} recorded during the enrollment phase and a sample R^k of the same response after deployment of the PUF instance. The number of differing response-bits is referred to as intra-Hamming distance $HD_{Intra}(R^{ref}, R^k)$[Mait 13][6]:

$$HD_{Intra}(R^{ref}, R^k) = \sum_{i=1}^{N} r_i^{ref} \oplus r_i^k \,. \tag{2.7}$$

Figure 2.7 graphically displays the computation of the intra-Hamming distance.

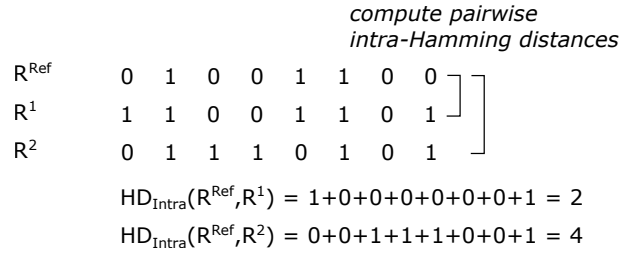


Figure 2.7: Computation of the response intra-Hamming distance.

Ideally, all responses of a PUF are fully reproducible under all specified operating conditions, i.e. the intra-Hamming distance is identically zero.

2.3.4 Challenge-Response Sensitivity

The Strict Avalanche Criterion (SAC) is a standard metric to quantify the extent to which the output-bits of a cryptographic function change upon complementing individual input-bits. A cryptographic function meets the SAC, if changing a single input-bit alters the state of each output-bit with probability 0.5. In the

[5]In the case of independent and uniformly distributed response-bits, the probability p_1 that a response-bits equals a logical 1 is 0.5, therefore the probability that two response-bits taken from two PUF keys u and v differ from each other is $p_{unique} = p_1^u \cdot (1 - p_1^v) + (1 - p_1^u) \cdot p_1^v = 0.5 \cdot 0.5 + 0.5 \cdot 0.5 = 0.5$. Due to the assumed independence it follows that the number of unique response-bits $0 \leqslant k_{unique} \leqslant N$ is binomially distributed, i.e. $X_{k_{unique}} \sim B(N, p_{unique})$.

[6]Note that the intra-Hamming measure defined in the reference is given in percent.

case of PUFs, the input- and output-bits correspond to an M-bit challenge C and an N-bit response R, respectively. Therefore, the SAC can be formulated in terms of the transition probability that a response-bit flips:

$$P\left(r_i|C \oplus r_i|C_j = 1\right) = 0.5 \quad \forall C, C_j \in \{0,1\}^M, i = 1, 2, ..., N\,, \tag{2.8}$$

where C_j denotes a challenge obtained from C by complementing the j-th challenge-bit. The degree to which the SAC is fulfilled can be judged based on the $N \times M$ dependence matrix A whose element a_{ij} corresponds the relative number of times the i-th response-bit changes upon inverting the j-th challenge-bit computed for a large number of challenges C^k, $k = 1, 2, ..., K$:

$$a_{ij} = \frac{1}{K} \sum_{k=1}^{K} r_i|C^k \oplus r_i|C_j^k\,. \tag{2.9}$$

The CRP behavior reacts highly sensitive to the input-challenge, if all matrix elements of A are close to the ideal value of 0.5. [Webs 86]

2.3.5 Complexity of the Challenge-Response Generation

The resilience of a PUF against model attacks is closely tied to the sample size required to model the internal structure of a PUF by learning algorithms [Ganj 15]. The order of the sample size needed to emulate the response behavior of a PUF can be estimated within the Probably Approximately Correct (PAC) learning framework [Vali 84]. Within this framework, building a model which assigns a new, randomly chosen challenge the correct response value with a probability of at least $1 - \delta$, $0 < \delta < 0.5$, and a maximum average prediction error of ε, $0 < \varepsilon < 0.5$, takes $m(\delta, \varepsilon)$ training CRPs in the order of

$$m(\delta, \varepsilon) = \mathcal{O}\left(\frac{1}{\varepsilon} \log \frac{1}{\delta} + \frac{VCdim(F)}{\varepsilon}\right)\,, \tag{2.10}$$

where $VCdim(F)$ denotes the Vapnik-Chervonenkis dimension[7] of a concept class F to be learned [Ehre 88].

Following this notion of complexity, a PUF concept can be considered secure against learning attacks, if the training sample size $m(\delta, \varepsilon)$ exceeds the number of CRPs that can be realistically gathered from a PUF within a finite amount of time.

[7]The Vapnik-Chervonenkis dimension is related to the complexity of the model building process; a mathematical definition is given in the reference cited.

2.4 Weak PUFs

A PUF design whose number of response-bits scales linearly with the physical size of the underlying system is referred to as Weak PUF or Physically Obfuscating Key (POK) [Herd 14]. A typical example are SRAM-based PUF designs, where each SRAM cell produces a single response-bit [Guaj 07]. In this case, doubling the response length increases the number of PUF structures by a factor of two. Due to size and power constraints, Weak PUF designs are usually limited to generate a small number of responses or even just a single CRP [Ruhr 14].

2.4.1 Fuzzy Extractor

Since the functionality of a Weak PUF can be easily emulated in hardware or software once the response is known to an attacker, the raw output of a Weak PUF must never be revealed in public [Herd 14].

For the application in cryptographic procedures such as message authentication algorithms, further requirements have to be fulfilled. Cryptographic functions generally take highly random and fully reproducible keys as input. A high degree of randomness can be achieved by the use of a cryptographic hash function, which takes responses of arbitrary lengths to produce uniformly distributed hashed keys of a fixed length. Since hash functions are designed to map any two non-identical responses to output keys that differ in 50 % of their bits on average, the PUF response must be fully reproducible upon key generation, i.e. all noise-related bit-errors must be corrected prior to hashing. Reproducibility of a PUF response can be achieved by a variety of different error correction schemes including those proposed in [Suh 05, Bosc 08, Maes 09, Yu 10a, Para 11]. The choice of a particular algorithm depends on the expected number of bit-errors to be corrected and resource constraints such as chip area and power consumption. The combination of error correction schemes and privacy amplification procedures including cryptographic hash functions were intensively studied within the scope of so called fuzzy extractors [Dodi 04]. A fuzzy extractor can be considered as a set of methods used to generate a reproducible uniform random key K from a noisy entropy source. Since no assumptions about the physical nature of the input sources are made, fuzzy extractors provide a powerful and flexible framework to implement Weak PUFs. [Maes 13]

The basic working principle of a generic fuzzy extractor is illustrated in Fig. 2.8. During the enrollment phase, a random seed S is generated which is fed to the

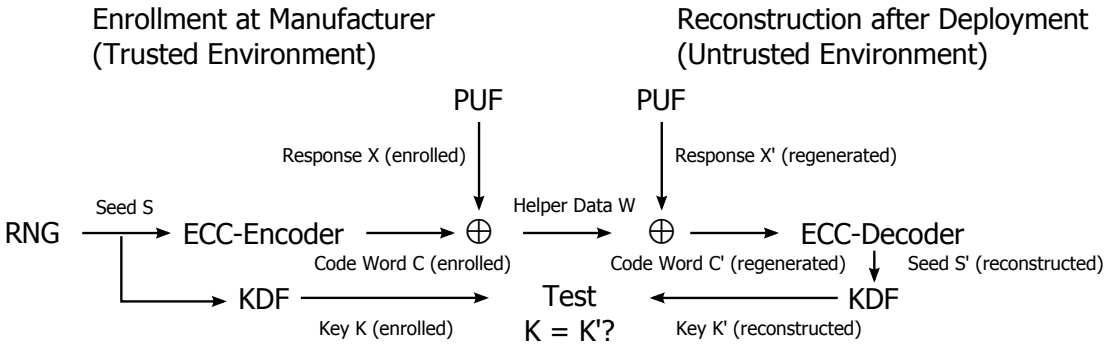


Figure 2.8: PUF-based fuzzy extractor. Adapted from [Maes 15].

randomness extractor KDF to produce a high-entropy key K. In addition, S is encoded by an error correction function to produce a code word C. The resulting code word is XORed with a PUF response X to produce helper data W, which is released to the public. The helper data is used to reconstruct S from a noisy response X' after deployment of the PUF. The error correction decoder function recovers the seed S' from the noisy code word C' using helper data W. Lastly, S' is used to regenerate the cryptographic key K', which is identical to the enrolled key K, unless the error-correction step failed. Note that S must never be made publicly available. In particular, it should not be possible for an attacker to recover the seed from the public helper data. [Maes 15]

Crucial to the implementation of a Weak PUF fuzzy extractor are the bit-error probability and entropy of a PUF design. The maximum number of bit-errors determines the capacity of the error correction algorithm required to guarantee fully reproducible PUF responses. The syndrome data generated by the error correction procedure and the raw response entropy provide a lower limit on the minimum response length necessary to produce a full entropy key. For illustration purposes, the implementation example described in [Maes 12] is repeated here. The authors designed the extractor to produce a 128-bit entropy key from 16×53 ring oscillators. The oscillators generate $42 \times 53 = 2226$ raw response-bits at an entropy rate of $\rho = 97.95\,\%$ yielding a response entropy of 2180.4 bits. To ensure reproducibility of the extracted keys, the maximum failure rate was limited to $p_{fail} \leqslant 1 \times 10^{-9}$, i.e. the maximum probability of one or more bit-errors to occur is less than or equal to p_{fail}. The target error rate was achieved by the use of two

concatenated error codes. The helper data generated by these correction codes leaks 2052 bits, thus reducing the minimum response entropy to 128.4 bits. In a last step, a cryptographic hash function is used to compress the 2226 response-bits into a 128-bit entropy key.

It is apparent from the example given, that proper functioning of the extractor requires consistent estimates of the response entropy and error probability. If the response entropy decreases due to changes in the processing conditions of the PUF devices, additional response-bits are needed to generate full entropy keys. Similarly, if the average bit-error probability increases, then the failure rate might exceed the specified limit. Selection of an error correction code with a higher error capacity increases the number of response-bits needed to compensate entropy loss caused by helper data leakage. Since one target application is to implement lightweight cryptography on resource-constrained platforms, the overhead associated with large safety margins on both design parameters is highly undesirable. Therefore, the main objective is to develop a high-performance PUF concept whose security related features are invariant to changes in the processing conditions and consistent for all realizations of a design.

2.4.2 Designs

Since the general PUF concept first caught attention of the scientific community in 2001 [Papp 02], numerous Weak PUF candidates have been proposed in the literature. A brief overview of selected designs is presented in this section. Similar to the classification system in [Maes 13], the designs are categorized into volatile memory-based PUFs and mixed-signal PUFs. Members of the former group directly output responses in binary form, whereas the latter type derives keys from analog characteristics, which require an additional quantization step to produce binary responses.

Volatile Memory-based PUFs

A large number of PUF concepts were developed for volatile memory components, as these devices can be readily turned into Weak PUFs with little overhead regarding area and power consumption. The most renowned example is the aforementioned SRAM PUF which exploits the random start-up pattern of

SRAM cells to generate secret keys. Similar PUF constructions incorporate cross-coupled latches (Butterfly PUF) [Maes 13], flip-flops [Maes 08], buskeeper cells[8] [Maes 13] and DRAM [Tehr 17]. Common to all these devices is that response-bits are extracted from the random preference of individual memory cells towards one of the two possible operating states upon power-up or reset [Maes 13].

Mixed-Signal PUFs

Mixed-signal PUFs were implemented for a variety of different device types. Unique PUF signatures could be successfully extracted from process-related variations of the electrical properties of FLASH cells [Prab 11, Wang 12, Kim 15, Jia 15], EEPROM cells [Will 18] and simple transistor arrays [Lofs 00]. RRAM [Liu 16] and memristor [Che 14] based PUFs leverage random mismatch in the resistance of the memory cells. The memristor PUF design reported in [Maza 15] exploits variable switching characteristics of memristive nano structures to generate secret bits.

Non memory-based concepts include the ring oscillator PUF, the coating PUF and the metal resistance PUF. The ring oscillator PUF exploits fabrication-related deviations in the frequencies of identically designed ring oscillators from which binary signatures are derived [Suh 07, Maes 13]. The coating PUF consists of a coating layer containing dielectric particles deposited on top of a metallic comb structure; the random placement of the particles within the dielectric causes stochastic variations in the capacitance of the metal structure serving as silicon fingerprint [Roy 09, Skor 06]. Yet another way to generate device-specific random secrets is to measure differences in the resistance of the power grid wires of an IC or entire chip as suggested by the inventors of the metal resistance PUF [Ju 13].

Digitization of the analog PUF characteristics can be achieved by means of a differential- or a threshold-based quantization procedure. A simple differential quantization scheme comprises the pairwise comparison of device parameters to generate a binary PUF response [Suh 07]. In the case of a threshold-based quantization scheme, one might compare the device characteristics to the mean value or median value [Che 14, Will 18] of a set of PUF devices.

[8] A buskeeper cell is designed to keep a bus line from floating by holding the last value sent.

2.4.3 Performance Comparison

A performance comparison of the different Weak PUF designs described in the previous two subsections is given in Tab. 2.1, which is an adapted version of the table presented in [Maes 13]. Listed are the mean inter- and intra-Hamming distances obtained under nominal operating conditions as reported by the specified sources. In addition, information about the implementation technology, number of tested PUF instance and response length is given.

The experimental results show that nearly any type of hardware-based random source can be turned into a Weak PUF. However, generating reproducible high-entropy keys without further processing of the raw output is difficult due to the effect of systematic process variations and noise inherent to the measurement of the physical quantities from which PUF responses are derived. Furthermore, it is apparent that the same PUF design implemented in different technologies is likely to yield different performance results, since the response properties are closely tied to the processing conditions of a PUF device.

Table 2.1: Overview over the performance metrics of various Weak PUF designs. Adapted from [Maes 13].

Device	Ref.	Hardware	No. Devices	No. Resp.	No. Bits	Mean HD_{Inter} in %	Mean HD_{Intra} in %
SRAM	[Guaj 07]	FPGA	1	17	65520	49.97	3.57
SRAM	[Holc 09]	COTS	8	10420	64	43.16	3.83
SRAM	[Holc 09]	μC	3	240	64	49.34	6.44
SRAM	[Seli 11]	ASIC	17	68	64k, 128k	≈ 50	< 4
SRAM	[Su 07]	ASIC	19	19	128	≈ 50.13 ... ≈ 50.55	≈ 3.04...≈ 3.78
SRAM	[Maes 13]	ASIC	192	768	65536	49.59 ... 49.72	5.46 ... 5.47
Flip-Flop	[Maes 08]	FPGA	3	3	4096	≈ 10 ... ≈ 12	< 1
Flip-Flop	[Lees 10]	ASIC	40	40	1024	36	< 4
Butterfly	[Kuma 08]	FPGA	36	36	64	≈ 42.5	< 5
Buskeeper	[Maes 13]	ASIC	192	384	8192	48.88 ... 48.92	4.16 ... 4.17
DRAM	[Tehr 17]	COTS	3	48	128	49.37	2.1 ... 3.1
RRAM	[Liu 16]	ASIC	5	40	128	24.2 ... 49.80	$0^{\dagger\dagger}$
EEPROM	[Will 18]	ASIC	68	68	252	30.24 ... 48.76	n.a.
Transistor	[Lofs 00]	ASIC	55	7260	112	≈ 50	$\lesssim 1.3^{\dagger}$
Ring Oscillator	[Maes 13]	ASIC	192	192	2048	49.60	1.53
Metal Resistance	[Ju 13]	ASIC	63	63	7506	49.87	$\leqslant 5.29$

†Mean value of HD_{Intra} determined from measurements of five chips after baking them at 250 °C for 100 h.

††The RRAM cells are operated in different restive states to increase the reproducibility of the responses.

2.4.4 Advanced Concepts

Different concepts and procedures were developed to increase the randomness and reproducibility of the raw responses of a Weak PUF. The procedures can be divided into two categories depending on whether they are applied to the binary output of a PUF or to the electrical characteristics from which responses are extracted.

One example falling into the former category, is a modified von Neumann debiasing step that turns the raw response of a PUF into a uniform bitstream by altering the response sequence [Maes 15]. Another example is the differential sequential coding approach described in [Hill 16], which aims to increase the reproducibility of PUF responses by processing only the most reliable response-bits effectively reducing the implementation overhead associated with additional error correcting hardware.

The remainder of this section focuses on methods that operate on the analog properties of a PUF. Selection criterion was the applicability of the procedures to arbitrary[9] mixed-signal designs that use some form of pairwise quantization scheme to generate binary responses. The first three methods described in the following sections were designed to increase the entropy of PUF responses. The last three methods intend to improve the reproducibility of the response generation mechanism of mixed-signal PUFs.

Neighbor-Based Pair Selection

The authors of [Mait 11] proposed a simple way to enhance the response randomness. Assuming that systematic process variations change smoothly over the distance between two PUF structures, the systematic pair mismatch can be reduced by selecting pairs of devices located in close spatial proximity to each other. However, since the magnitude and spatial distribution of the electrical characteristics of the PUF structures are generally not known after fabrication, the question remains to which extend systematic process variations cancel out between the selected pairs.

[9] Although the presented methods were originally proposed and evaluated for ring oscillator PUFs, they can be directly transferred to other designs.

Regression-Distiller

A more sophisticated concept was developed by the authors of [Yin 14, Yin 13a] who suggested to use regression analysis to enhance the randomness of PUF-based entropy sources. For this purpose, the effect of systematic process variations is estimated by means of a higher order polynomial regression model. Subtraction of the fitted values from the raw data yields random variation components free of deterministic trends. The random variation components are digitized using a pairwise quantization scheme. The regression coefficients are stored in form of helper data used to compute the residuals whenever the PUF response is generated [Yin 13b]. One apparent drawback of this method is the additional hardware required to compute the regression fit on-chip.

Normalization-based Bias Removal

An alternative to the regression-based distiller approach was described in [Maes 12]. Instead of a regression-based trend analysis, the systematic mismatch between PUF structures is estimated from the average over a large number of measurements and devices. The response randomness of a PUF instance is increased by subtracting the respective mean value from the electrical parameters of each PUF structure. The response-bits are then extracted from the detrended parameter values. This normalization procedure has to be performed on-chip, whenever a PUF response is generated. Estimation of the mean values, on the other hand, only needs to be done once in principle, i.e. for a batch of test chips after design development. The mismatch estimates are stored in form of publicly accessible helper data.

As shown in later sections, the normalization procedure does not account for inter-chip variations of the systematic mismatch between PUF devices due to averaging over the parameter values. Therefore, the response-bias is only partly removed.

Pair Masking

One approach to increase the reproducibility of a PUF response is to extract response-bits from pairs of PUF devices with large absolute differences in their electrical parameters.

In [Suh 07], a 1-out-of-K masking scheme was used to select the most stable pair of parameter values from a group of K PUF structures. Since only one response-bit per group of PUF devices is extracted, a total number of $N \times K$ PUF structures are required to generate N response-bits. Alternatively, response-bits might be extracted from pairs of parameter values that exceed a given threshold [Yin 13b, Yin 10]. Since not all pair combinations necessarily fulfill the threshold requirement, the resulting response length might vary from PUF instance to instance. In both cases, the increase in stability is limited by the variability of the electrical parameters and the number of PUF structures from which pairs are formed.

Index-Based Syndrome Coding

In [Yu 10a], a low-leakage syndrome coding scheme named Index-Based Syndrome Coding (IBS) was presented to decrease the complexity and failure rate of conventional error correction techniques. IBS uses the sign and amplitude of real-valued parameter differences of a group of PUF device pairs to encode the bits of a secret key. A simple encoding function can be realized by assigning the smallest (largest) difference value of a group to a bit-value equal to a logical 0 (1). Decoding of a key given a fuzzy response of the PUF correctly reproduces a bit-value, if the sign of the corresponding pair difference did not change at readout time compared to the sign recorded during the enrollment phase. Since pair differences with large absolute values were selected for the encoding process, changes in the sign are less likely to occur than for small parameter differences close to zero. A modification of the concept provides a flexible framework to adapt IBS to different implementation constraints regarding error correction capacity, size of the helper data and decoder complexity [Hill 12].

As for the pair masking procedure, limiting factors to the robustness of IBS are the magnitude of the process variability and the groups size of parameter differences used for the encoding step.

Robust Pair Selection

The authors of [Yin 09] propose a quantization scheme based on a reconfigurable selection of parameter pairs to generate highly reproducible response-bits. In the event that the sign of a pairwise parameter difference changes within the specified

operating range of the PUF, a different pair is assigned to generate the respective response-bit, hence allowing an error-free reconstruction of the PUF key.

A significant drawback to this approach is the need for secondary tests at non-standard operating settings during the enrollment phase effectively increasing the fabrication costs and hence limiting the appeal for mass production. A simplification of the enrollment process might be achieved by predicting parameter dependencies based on auxiliary characteristics that are less costly to measure than the actual parameter of interest. The authors of [Rahm 15], for example, exploit correlations between aging-related degradation of the frequency of ring oscillator PUFs and frequency changes due to supply voltage variations to determine stable pair combinations; instead of performing lengthy aging test, the device measurements are simply carried out at different voltage operating conditions.

2.5 Strong PUFs

Designs that output a large number of CRPs are labeled Strong PUFs. Due to the large, ideally exponential number of CRPs, the authentication procedure of a PUF instance can be simplified compared to Weak PUFs. Instead of exchanging an obfuscated key derived from a raw response, authentication is typically based on possibly unprocessed CRPs that can be exchanged in the open as described in the simple challenge-response protocol in Sec. 2.2. [Herd 14]

In the following sections, an overview over common native Strong PUF concepts and Weak PUF designs with an enhanced challenge-response space is given. In this work, all hardware realizations of PUFs that implement a complex CRP behavior by design are considered native Strong PUFs. In contrast, the latter type of design utilizes Weak PUFs whose analog characteristics or binary responses are configured in a way that yields a large CRP space.

2.5.1 Native Designs

The archetype of a native Strong PUF was invented by the authors of [Lim 05]; a slightly more abstract notion of which is reformulated in this section: a generic native Strong PUF design resembles a chain-like structure comprised of a number

of configurable segments with two input and output channels forming two identical paths along the chain (Fig. 2.9). The pathway through the segments is set by a challenge which controls whether the output sequence of a segment is inverted. Since each segment is configured individually, the number of possible paths grows exponentially with the number of chain segments. To generate a response-bit, an electrical signal is propagated through both paths. Transmission through the paths is modulated by random process variations resulting in a measurable signal difference at the end of the chain. Quantization of the signal difference yields a single response-bit that depends on the configuration of the segments. An N-bit response can be obtained by concatenating the output of multiple chains or by multiple evaluations of the same chain for different challenges. The totality of all CRPs uniquely defines a Strong PUF instance.

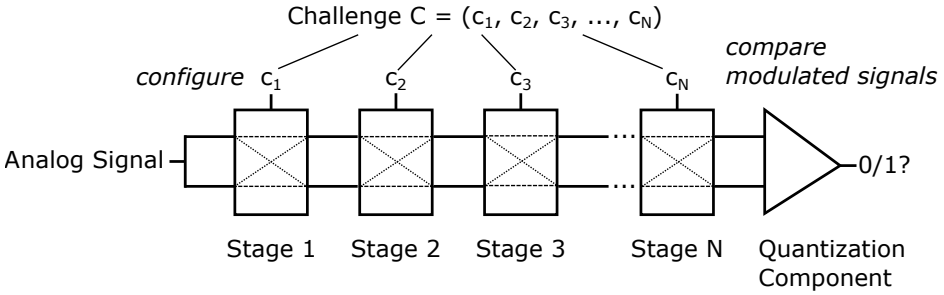


Figure 2.9: Conceptual illustration of a native Strong PUF design. Adapted from [Lim 05].

Arbiter PUF

One of the first designs to implement the aforementioned architecture was the Arbiter PUF introduced in [Lim 05][10]. The design exploits random propagation delays of two identical signals passed through a number of switching stages connected in series. The transmitted signals are fed to an Arbiter that outputs a logical 0 or 1 depending on which signal arrives first. The decision of the Arbiter is taken as response-bit.

As noted by the authors, the design is inherently susceptible to model building attacks due to the linear dependence on the gate delays of the multiplexer stages.

[10]The fundamental design of the delay-based circuit without the arbiter component was first described in [Gass 02].

A small number of observed CRPs ($\approx$ 1000) suffices to determine the internal delay parameters allowing to predict new unobserved CRPs with high accuracy ($\approx 95\,\%$ success rate) [Ruhr 10].

Feedforward Arbiter PUF

In order to increase the model complexity of ordinary Arbiter PUFs, nonlinear system components were implemented by integrating feedforward loops into the signal paths leading to the feedforward Arbiter PUF design described in [Lim 05]. A 64-bit variant of this Arbiter design could be successfully broken for up to eight feedforward stages [Ruhr 10]. The authors of [Alka 17] managed to accurately model 128-bit feedforward Arbiter PUFs including up to six forward loops.

XOR Arbiter PUF

Aside from including nonlinear elements in the signal paths of an Arbiter PUF, multiple output-bits of the same or different Arbiter PUF instances can be XORed to complicate model building attacks [Suh 07]. Analysis of the model complexity showed that the number of CRPs required to mount a successful attack scales exponentially with the number of XOR operations [Ganj 15]. However, the number of XOR operations is limited by the reproducibility of the responses. Experimental results revealed an exponential decrease in the number of reproducible CRPs for an increasing number of XORed output-bits [Zhou 17]. Therefore, the maximum achievable model complexity of XOR Arbiter designs can only be increased up to the point where the generation of stable responses is still feasible [Ganj 15]. Furthermore, it was shown that instabilities in the response generation can be exploited to attack individual PUF instances in a XOR Arbiter ensemble, significantly reducing the number of CRPs required to model multiple XORed Arbiter outputs [Beck 15].

Lightweight Secure PUF

In order to improve the challenge sensitivity of the responses and to increase the model complexity, the basic XOR Arbiter architecture was expanded by interconnecting challenge- and response-bits in form of complicated input- and output-networks [Majz 08a]. Unlike the original design, pairs of challenge-bits are XORed

prior to configuring the multiplexer stages of an Arbiter chain. The transformed challenge is shifted bitwise for each Arbiter chain to generate a unique configuration sequence for each instance. Combination of both operations results in a CRP behavior which satisfies the SAC (Sec. 2.3.4). Resistance against modeling attacks is increased by connecting the output of the Arbiter chains in a network to obfuscate the contributions of the individual chains to the final response-bits. Despite these counter measures, 128-bit lightweight secure PUF constructions implementing up to five XOR operations were successfully attacked by machine learning algorithms [Ruhr 10].

Current Mirror PUF

The current mirror PUF design consists of a series of current mirrors connected in parallel [Kuma 14]. Due to random process variations, the transfer characteristics of the current mirrors mismatch in a random fashion, thus a current sent through the parallel paths will differ in magnitude after reaching the end of the two chains. The difference in the output current is quantized to generate a response-bit. A challenge-response behavior is established by adding a switching block after each pair of current mirrors similar to the Arbiter PUF design. Since the current shift after each stage depends on the magnitude of the current itself, the total change in current along the chains of the proposed PUF architecture is highly nonlinear. However, despite the nonlinearity of the CRP generation process, the current mirror PUF was shown to be vulnerable against model attacks [Guo 16].

Bistable Ring PUF

Another delay-based concept termed the bistable ring PUF was developed by the authors of [Chen 11]. The bistable ring PUF is constructed from pairs of NOR gates arranged in form of a ring. The inputs and outputs of the different pairs of NOR gates are interconnected by a multiplexer and demultiplexer per ring segment. A stimulus is used to force oscillations along the NOR segments and after a predetermined settling time the current state of the ring is evaluated. The logic state of one of the ring segments is taken as response-bit. Since the output state depends on the selected delay path configurable by an external challenge, the number of CRPs grows exponentially with the number of ring segments.

As pointed out by different authors, the bistable ring PUF suffers from a

number of security vulnerabilities. Theoretical considerations [Ganj 16] and experimental results [Yama 14] suggest that the response-bits of a bistable ring PUF are dominated by a small number of challenge-bits for a given challenge, hence the design does not meet the SAC. Furthermore, the presence of dominant challenge-bits could be successfully exploited to carry out model building attacks [Ganj 16]. In order to increase the difficulty of modeling the CRP behavior, multiple output-bits might be XORed similar to the XOR Arbiter PUF concept. In case of the bistable ring PUF, XORing more than three response-bits improved the model resilience of the design, as was demonstrated in [Xu 15].

2.5.2 Challenge-Response Enhancement

The CRP space of Weak PUFs can be enhanced by mathematical operations that combine the internal electrical parameters of a PUF device and a user defined challenge. Existing CRP enhancing procedures employ functions that perform multiple comparisons of device parameters, where each comparison is carried out according to a challenge sequence as described in subsequent paragraphs. Since these functions operate on non-binary quantities, their application is limited to mixed-signal PUFs. The very basic design of a PUF with enhanced CRP-space is displayed in Fig. 2.10.

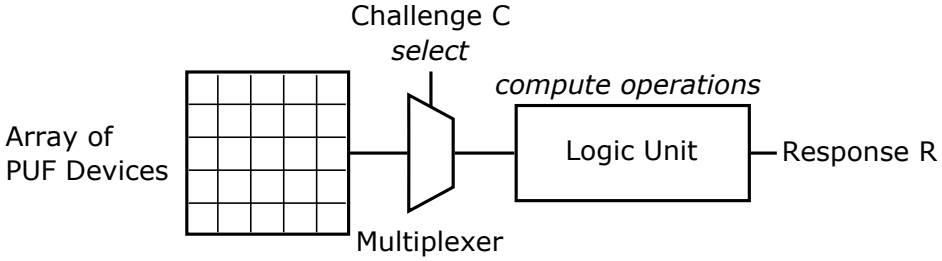


Figure 2.10: Conceptual illustration of a generic challenge-response enhanced PUF design.

Group-Based Enhancement

The CRP space of ring oscillator PUFs can be enlarged by multiple pairwise comparisons of oscillation frequencies as demonstrated in [Dela 16]. A challenge is used to select k oscillators from a set of n devices. Each selected frequency is compared to the remaining ones to produce $n-1$ response-bits. A final response

is obtained by XORing the k responses and a random vector. Since knowledge of the CRPs and the corresponding random vectors allows to model the response behavior, the random vector must be securely generated on-chip[11] whenever a CRP is queried. The authors of the PUF concept suggest to construct random vectors from a $n-1$ bit-sequence obtained by comparing all n ring oscillators to the oscillator whose oscillation frequency is closest to the median value of all tested devices. The resulting sequence is rearranged based on the minimum autocorrelation lag distance between the bits. In a last step, the sequence is shifted according to the sum of the selected oscillator indices modulo $n-1$.

Problematic to this particular choice of random vectors is the correlation structure originating form the shift operation. The random vectors computed for certain challenge combinations leak information about the outcome of the pairwise comparisons of the oscillation frequencies. After only $n+2$ chosen challenge-response queries, the internal parameters of the group-based PUF can be fully reconstructed to build a software model of the CRP behavior. Therefore, the PUF concept provides only a weak level of protection in all use cases, where an attacker is able to send a user-defined set of challenges to the PUF for which he can observe the corresponding responses. [Wisi 18]

Identity-Mapping Enhancement

In [Mait 12], an identity-mapping function was proposed which maps the frequency difference of configurable ring oscillator pairs to a set of real numbers Q. In a first step, m out of a total number of M ring oscillators are selected according to a user-supplied challenge. In the next step, $\binom{m}{t}$ frequency pairs are grouped into a subset S_t for each $t \in \{2, 3, ..., m\}$ yielding t different subsets. Lastly, the weighted sum of the square root of the pair differences is computed for each element in these subsets. The terms of the sum are weighted by the euclidean distance between the physical locations of the ring oscillator pairs. The procedure yields a Q-value for each element in S_t, $t = 2, 3, ..., m$. Therefore, a maximum number of $2^m - m - 1$ Q-values can be obtained in total for all $m-1$ subsets combined. The calculated Q-values are digitized to produce a binary response.

As noted by the authors, the CRPs obtained by this method are not indepen-

[11] Permanent storage of the random vector in an secure NVM would defeat the purpose of using a PUF in the first place.

dently generated which might be exploited by an attacker to predict the CRP behavior.

k-Sum PUF

The k-sum PUF [Yu 10b, Yu 12] operates on process-induced differences in the oscillation frequency of k ring oscillator pairs. A response-bit is obtained by digitizing the sum of all pair differences. In order to generate different CRPs, a k-bit challenge is used to manipulate the sign of the individual pair differences. Similar to the Arbiter PUF, the number of CRPs increases exponentially in the number of pair differences and challenge-bits. Since the response value of a k-sum PUF depends linearly on the frequency differences, the concept is vulnerable to the exact same model building attacks as the basic Arbiter PUF design.

2.5.3 Performance Comparison

Table 2.2 gives an overview over the performance of a variety of Strong PUF designs. Listed are the mean fractional inter- and intra-Hamming metrics calculated for a given number of PUF devices and challenges acquired at nominal operating conditions. The entry labeled number of bits corresponds to the number of response-bits per challenge. The presented information is partly taken and adapted from the corresponding table listed in [Maes 13] with some additions including the Strong PUF designs based on the CRP enhancing concepts described in Sec. 2.5.2.

Objective evaluation and comparison of the implementation dependent Hamming distances are difficult due to the limited amount of test data available for the majority of the concepts. More recent Strong PUF designs such as the current mirror or k-sum PUF, for example, show promising first results regarding their response uniqueness and stability. However, further measurements have yet to be carried out to fully assesses the overall performance of these designs, especially when transferred to different processing technologies.

The need for extensive measurements becomes apparent when comparing the reported inter-Hamming metrics of the different realizations of the Arbiter PUF: the same PUF concept implemented in ASICs and on commercial FPGAs yields average inter-Hamming distances ranging between 1.05 % and 47.13 %. This un-

certainty in key performance characteristics is a pressing issue relevant to all state-of-the-art Strong PUF concepts; the security level warranted by a design is highly dependent on the manufacturing process and technology.

Compared to the intra- and inter-Hamming metrics only limited information is available whether a design complies to the SAC. From the listed concepts, only the lightweight secure PUF was verified to meet the SAC [Majz 08a]. The vast majority of the remaining designs were not analyzed in this regard despite the presumption that the existence of dominant challenge-bits might enable or facilitate model building attacks on Strong PUFs as was shown for the bistable ring PUF [Ganj 16].

Modeling of the challenge-response behavior currently poses a notable threat to existing PUF constructions as is apparent from previous sections. Learning attacks could not only break linear designs, e.g. the Arbiter PUF, but also non-linear constructions like the XOR-Arbiter PUF and the current mirror PUF. Out of the listed constructions, all designs except for the untested identity-mapping PUF were successfully attacked in the past. Naturally, concepts proven to be vulnerable to an attack are not suitable for public challenge-response authentication protocols.

Table 2.2: Overview over the performance metrics of various Strong PUF designs. Adapted from [Maes 13].

Device/Concept	Ref.	Hardware	No. Devices	No. Challenges	No. Bits	Mean HD_{Inter} in %	Mean HD_{Intra} in %
Arbiter	[Gass 04]	FPGA	23	100000	1	1.05	0.1
Arbiter	[Hori 10]	FPGA	45	1024	128	36.75	1.71
Arbiter	[Maes 13]	ASIC	192	256	256	47.13	3.04
Arbiter	[Lim 05]	ASIC	20	100000	1	23	0.7
Arbiter (feedforward)	[Lim 05]	ASIC	20	100000	1	38	4.5
Arbiter (2-XOR)	[Maes 13]	ASIC	192	256	128	49.74	5.89
Current Mirror	[Wang 17]	ASIC	2	10000/50000†	256	51.48	0.16
Bistable Ring	[Chen 11]	FPGA	8	50000	1	44	2.19
Bistable Ring	[Chen 12]	ASIC (simulated)	15	1000	1	50.9	1.3
Bistable Ring	[Xu 15]	FPGA	8	1000000	1	n.a.	≈ 1
Bistable Ring (2-XOR)	[Xu 15]	FPGA	< 8	1000000	1	≈ 50	< 2.5
Bistable Ring (3-XOR)	[Xu 15]	FPGA	< 8	1000000	1	≈ 50	< 2.5
Bistable Ring (4-XOR)	[Xu 15]	FPGA	< 8	1000000	1	≈ 50	≈ 2.5
Bistable Ring (5-XOR)	[Xu 15]	FPGA	< 8	1000000	1	≈ 50	< 5
Bistable Ring (6-XOR)	[Xu 15]	FPGA	< 8	1000000	1	≈ 50	< 5
Bistable Ring (7-XOR)	[Xu 15]	FPGA	< 8	1000000	1	≈ 48	< 5
Bistable Ring (8-XOR)	[Xu 15]	FPGA	1	1000000	1	n.a.	≈ 5
Group-Based Enhancement	[Dela 16]	FPGA	125	32767	15	49.81	1.76
Identity-Mapping	[Mait 12]	FPGA	125	65519	1	49.99	<3
k-Sum	[Yu 12]	ASIC	76††	n.a.	256	48.9 ... 50.1	n.a.

†Number of challenges used to compute the mean value of HD_{Inter} and HD_{Intra}, respectively.

††76 PUFs implemented on 19 ASICs.

Chapter 3

Mathematical Background

This chapter provides a brief overview over the core mathematical concepts and methodologies the proposed model framework presented throughout this work is based on. Emphasis is placed on topics related to statistical probability distributions, stochastic hypothesis tests and spatial data analysis. The material presented throughout the following sections was mainly adapted from [Gelm 14, Scha 05, Tong 90]. Detailed information on the aforementioned topics can be found in the references given.

3.1 Probability Distributions

PUF responses are generated from electrical quantities subjected to stochastic process variations. Therefore, the physical characteristics of the semiconductor devices constituting a PUF and the response-bits derived from said characteristics have to be treated as random variables. Consequently, a probabilistic model descriptions of the response behavior and properties of the underlying manufacturing process is considered essential for the analysis and evaluation of a PUF design. For this purpose, a number of probability distributions are introduced in the following sections which will be made extensive use of in Chap. 4 and 6.

3.1.1 Normal Distribution

Multivariate Case

Within the scope of the proposed Gaussian random field approach (Chap. 4), the multivariate normal distribution plays a central role in modeling the electrical characteristics of a PUF from which binary responses are generated. The formal definition and the most relevant properties of the distribution will be briefly summarized below. The reader is referred to [Tong 90] for a comprehensive and detailed treatment of multivariate normally distributed random variables.

Let $X = [X_1, X_2, ..., X_N]^{\mathrm{T}}$ denote a random vector comprised of N jointly normally distributed random variables X_i, $i = 1, 2, ..., N$, then $X \sim N(\mu_X, \Sigma_X)$ follows a multivariate normal distribution characterized by the probability density function:

$$p(x) = \frac{1}{\sqrt{(2\pi)^N \det(\Sigma_X)}} \cdot \exp\left(-\frac{1}{2}(x-\mu_X)^{\mathrm{T}}\Sigma_X^{-1}(x-\mu_X)\right) . \tag{3.1}$$

The $N \times 1$-dimensional mean vector μ_X is equal to the expected value $E[X]$ of X, i.e.

$$\mu_X = E[X] = [E[X_1], E[X_2], ..., E[X_N]]^{\mathrm{T}} = [\mu_{X_1}, \mu_{X_2}, ..., \mu_{X_N}]^{\mathrm{T}} . \tag{3.2}$$

Σ_X denotes the $N \times N$ covariance matrix of X. The matrix elements $[\Sigma_X]_{ij}$, $i, j = 1, 2, ..., N$ are of the form:

$$[\Sigma_X]_{ij} = E[(X_i - \mu_{X_i})(X_j - \mu_{X_j})] = \mathrm{cov}(X_i, X_j) , \tag{3.3}$$

where the latter expression defines the covariance of two random variables X_i and X_j. [Tong 90]

The probability $P(X_1 \leqslant a_1, X_2 \leqslant a_2, ..., X_N \leqslant a_N)$ that random realizations x_i, $i = 1, 2, ..., N$, of X are equal to or less than some real numbers $\{a_i : i = 1, 2, ..., N\}$ is determined by the cumulative distribution function (CDF)

$$F_X(a_1, a_2, ..., a_N) = P(X_1 \leqslant a_1, X_2 \leqslant a_2, ..., X_N \leqslant a_N) \tag{3.4}$$

$$= \int_{-\infty}^{a_1} dx_1 \int_{-\infty}^{a_2} dx_2 ... \int_{-\infty}^{a_N} dx_N p(x_1, x_2, ..., x_N) \tag{3.5}$$

with probability density function $p(x) := p(x_1, x_2, ..., x_N)$ as defined in Eq. (3.1). [Fabe 12]

Distribution of Linear Combinations As stated previously, quantization of the electrical characteristics of a PUF design often involves sums or differences of subsets of random vectors or variables. A pairwise differential quantization scheme, for example, extracts binary keys from the difference of two sets of electrical device characteristics of a PUF instance.

In case of a multivariate normal random vector X as defined above, any subset or linear combination of subsets formed from X is also multivariate normally distributed. Therefore, given a non-random vector A of dimension $K \times 1$ and a $K \times N$ non-random matrix B, it follows that the linear combination of subsets $Y = A + BX$ is multivariate normally distributed:

$$Y \sim N(\mu_Y, \Sigma_Y) \tag{3.6}$$

with mean vector

$$\mu_Y = A + B\mu_X \tag{3.7}$$

and covariance matrix

$$\Sigma_Y = B\Sigma_X B^{\mathrm{T}}\,, \tag{3.8}$$

which correspond to the transformed distribution parameters of X. [Tong 90]

Conditional Distribution Aside from the unconditional probability $p(x)$ of a single random vector X, an expression for the conditional density function of two jointly normal random vectors X and Y will be needed in later sections to model the reproducibility of PUF responses generated under variable operating conditions (Sec. 4.8.2).

Given two jointly normal random vectors $X = [X_1, X_2, ..., X_M]^{\mathrm{T}}$ and $Y = [Y_1, Y_2, ..., Y_N]^{\mathrm{T}}$, then X conditional on Y is also multivariate normally distributed, i.e.

$$X|Y = \tilde{y} \sim N(\mu_{X|Y}, \Sigma_{X|Y}) \tag{3.9}$$

with $\tilde{y}$ being a random realization of Y. The conditional covariance matrix $\Sigma_{X|Y}$ is obtained by transforming Σ_X according to the equation:

$$\Sigma_{X|Y} = \Sigma_X - \Sigma_{X,Y}\Sigma_{Y,Y}^{-1}\Sigma_{Y,X}\,, \tag{3.10}$$

where

$$\Sigma_{X,Y} = \Sigma_{Y,X}^{\mathrm{T}} = E[(X - \mu_X)(Y - \mu_Y)^{\mathrm{T}}] \tag{3.11}$$

represents the cross-covariance matrix of the random vectors X and Y. The expression for the conditional mean vector $\mu_{X|Y}$ is fully determined by the mean vectors μ_X, μ_Y, the covariance matrices $\Sigma_{X,Y}, \Sigma_Y$ and the realization $\tilde{y}$:

$$\mu_{X|Y} = \mu_X - \Sigma_{X,Y}\Sigma_Y^{-1}(\tilde{y} - \mu_Y)\,. \tag{3.12}$$

Substituting Eq. (3.10) and (3.12) into Eq. (3.1) yields a formula for the conditional probability density function $p(x|y = \tilde{y})$. [Tong 90]

Univariate Case

The univariate marginal distributions of a multivariate normal vector are themselves normally distributed, i.e. each component X_i, $i = 1, 2, ..., N$, of a jointly normal random vector $X = [X_1, X_2, ..., X_N]^{\mathrm{T}}$ is characterized by a probability density function of the form:

$$p(x_i) = \frac{1}{\sqrt{2\pi\sigma_{X_i}^2}} \exp\left(-\frac{(x_i - \mu_{X_i})^2}{2\sigma_{X_i}^2}\right), \tag{3.13}$$

specified in terms of a location and scale parameter equal to the mean μ_{X_i} and variance $\sigma_{X_i}^2$ of X_i, respectively. The mean value μ_{X_i} is determined by the expected value $E[X_i]$ and $\sigma_{X_i}^2 \equiv \mathrm{var}(X_i)$ defines the variance of the random variable X_i. [Tong 90]

The corresponding cumulative distribution function F_{X_i} of X_i is determined by the equation:

$$F_{X_i}(a_i) = \int_{-\infty}^{a_i} dx \frac{1}{\sqrt{2\pi\sigma_{X_i}^2}} \exp\left(-\frac{(x_i - \mu_{X_i})^2}{2\sigma_{X_i}^2}\right) \tag{3.14}$$

$$= \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{a_i - \mu_{X_i}}{\sigma_{X_i}\sqrt{2}}\right)\right], \tag{3.15}$$

where $\mathrm{erf}(x)$, $x \in \mathbb{R}$, denotes the error function:

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x dx \exp(-x^2)\,. \tag{3.16}$$

3.1.2 Lognormal Distribution

Multivariate Case

In Sec. 4.8 and Sec. 6.3, the effect of variable measurement noise on the reproducibility of PUF responses will be analyzed. As motivated in later sections, the

noise distribution models the measurement variance $\sigma^2_{Noise} > 0$ of the electrical characteristics of a PUF. Since the normal distribution is defined on the entire real line, the Gaussian approximation is a poor choice to model σ^2_{Noise} – a strictly positive quantity – especially if the average variance is close to zero. Instead, the multivariate lognormal distribution will serve as standard model to investigate the stochastic properties of positive and possibly correlated random variables.

A random vector $X = [X_1, X_2, ..., X_N]^{\mathrm{T}}$ is said to be multivariate lognormally distributed, if $\log(X) = [\log(X_1), \log(X_2), ..., \log(X_N)]^{\mathrm{T}}$ follows a multivariate normal distribution

$$\log(X) \sim N(\mu, \Sigma) \tag{3.17}$$

specified in terms of the mean vector μ and covariance matrix Σ of the logarithmized random vector. The density function of X is determined by the equation:

$$p(x) = \frac{1}{\sqrt{(2\pi)^N \det(\Sigma)} \prod_{i=1}^{N} x_i} \cdot \exp\left(-\frac{1}{2}(\log(x) - \mu)^{\mathrm{T}} \Sigma^{-1} (\log(x) - \mu)\right) \tag{3.18}$$

for $x > 0$ and zero otherwise. [Tarm 01]

Univariate Case

Similar to the multivariate normal distribution, the marginal distributions of N jointly lognormally distributed random variables $X_1, X_2, ..., X_N$ are fully specified by their respective univariate density functions [Gelm 14]:

$$p(x_i) = \frac{1}{\sqrt{2\pi\sigma_i^2} x_i} \exp\left(-\frac{(x_i - \mu_i)^2}{2\sigma_i^2}\right) \tag{3.19}$$

with $p(x_i) \equiv 0$ for $x_i \leqslant 0$.

3.1.3 Folded Normal Distribution

Following the analysis of the response stability in Sec. 6.3, it will be shown that the bit-error probabilities depend only on the absolute value of the systematic mismatch between the primitive devices constituting a PUF instance. In the case of normally distributed device mismatch, the absolute values of the mismatch follow a folded normal distribution.

The probability density function of a folded normal distribution with mean μ and variance σ^2 is given by the equation [Tsag 14]:

$$p(x) = (x) = \sqrt{\frac{2}{\pi\sigma^2}} \exp\left(-\frac{(x^2+\mu^2)}{2\sigma^2}\right) \cosh\left(\frac{\mu x}{\sigma^2}\right), \tag{3.20}$$

where $\cosh(x)$, $x \in \mathbb{R}$, represents the hyperbolic cosine function defined as

$$\cosh(x) = \frac{e^x + e^{-x}}{2}. \tag{3.21}$$

3.1.4 Poisson Binomial Distribution

The response of a PUF can be thought of as a realization $y \in \{0,1\}^N$ of a set of binary random variables $\{Y_i : i = 1, 2, ..., N\}$ with N being equal to the number of response-bits. In order to evaluate key performance metrics such as the inter- and intra-Hamming distance, the relative frequencies at which different numbers of unique or stable response-bits occur need to be determined.

Assuming independence of the response-bits, the total number of bits X having a certain feature in common, e.g. X = 'number of reproducible bits', can be regarded as the sum of N independent Bernoulli trials, where the outcome of each trial indicates whether the characteristic feature is inherent to the respective bit. In this case, X follows a Poisson binomial distribution defined by the probability mass function [Hong 13]:

$$P(X = k) = \sum_{A \in B_k} \prod_{i \in A} p_i \cdot \prod_{j \in A^c} (1 - p_i), \tag{3.22}$$

where $P(X = k)$ equals the probability that exactly k of the N Bernoulli trials were successful. The success probability of the individual Bernoulli trials is determined by probabilities $p_i \in [0,1]$, $i = 1, 2, ..., N$. The sum is carried out over all k-element subsets taken from $\{1, 2, ..., N\}$, the set of which is denoted by B_k. The superscript c marks the complement of a set, i.e. A^c is the complement of A.

Since the success probabilities do not need to be identical, bit-specific differences in the response characteristics such as variable bit-error probabilities can be easily taken into account by choosing appropriate values for the p_i's .

3.1.5 Binomial Distribution

Simplified models of the response properties might ignore any differences in the distributional characteristics of the individual response-bits, e.g. the error proba-

bility of each response-bit is assumed to be the same for all bits (Sec. 6.3). Consequently, the success probability of each Bernoulli trial representing the outcome of a response-bit being unstable is considered identical, i.e. $p_1 = p_2 = ... = p_N \equiv p$.

In this instance, the summed outcome X of N Bernoulli trials is binomially distributed $X \sim B(N, p)$ with common success probability p and probability mass function

$$P(X = k) = \binom{N}{k} p^k (1-p)^{N-k}, \tag{3.23}$$

where k denotes the number of successes. [Gelm 14]

3.2 Statistical Tests

Statistical tests are designed to assess specific features of a random variable, e.g. its distribution form. For this purpose a test statistic T is defined whose stochastic properties allow to quantify the likeliness that a certain hypothesis related to the features of interest holds. For example, one might test the hypothesis that a random sample follows a Gaussian probability distribution. In order to test the validity of a hypothesis, the likeliness of the observed test statistic T_{obs} computed from a random sample is either compared to the theoretical distribution of the test statistic or an expected empirical distribution of T (Fig. 3.1). If the probability $p = P(T > T_{obs})$, typically referred to as p-value, is lower than a probability $\alpha = P(T > T_{lim})$ that the test statistic T exceeds a critical value T_{lim} defined prior to the test, then the deviation of the test statistic from the expected result is considered statistically significant, i.e. the outcome is deemed unlikely to occur by pure chance alone. In this case, the null hypothesis that no statistical difference between the actual and assumed distribution of the test statistic exists is rejected at significance level α. Conversely, α also determines the probability that the null hypothesis is falsely dismissed. The error committed by rejecting the null hypothesis albeit being true is called a type I error. [Feld 10]

In this work, three statistical tests, namely the Anderson-Darling test, the Moran's I test and the Chi-squared test were used to evaluate the stochastic properties of a PUF. The conceptual ideas behind the different tests will be briefly explained in the subsequent sections.

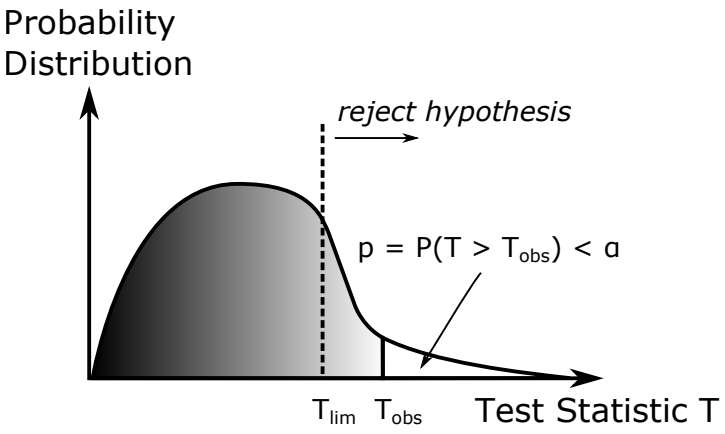


Figure 3.1: Conceptual illustration of a statistical hypothesis test.

3.2.1 Anderson-Darling Test

The Anderson-Darling test for normality is used to test whether the empirical distribution function of a sample deviates from a hypothesized normal distribution $N(\mu, \sigma^2)$ in a statistically significant way. For distribution parameters (μ, σ) estimated from the data, the test statistic A^{*2} is determined by the relation:

$$A^{*2} = A^2 \left(1 + \frac{0.75}{N} + \frac{2.25}{N^2}\right) , \tag{3.24}$$

where N denotes the sample size and A^2 is defined as:

$$A^2 = -N - \frac{1}{N} \sum_{i=1}^{N} (2i - 1)(\ln F_Y(y_i) + \ln(1 - F_Y(y_{n+1-i}))) . \tag{3.25}$$

$F_Y(y)$ is the cumulative distribution function of the standard normal distribution $N(0, 1)$ obtained by setting the mean and standard deviation to $\mu = 0$ and $\sigma = 1$ in Eq. (3.14). The standardized sample values y_i, $i = 1, 2, ..., N$, are obtained by centering and scaling the original sample data $\{x_i : i = 1, 2, ..., N\}$:

$$y_i = \frac{x_i - \hat{\mu}}{\hat{\sigma}} \tag{3.26}$$

given estimates of the mean

$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{3.27}$$

and sample variance

$$\hat{\sigma}^2 = \frac{1}{N - 1} \sum_{i=1}^{N} (x_i - \hat{\mu})^2 . \tag{3.28}$$

The critical values corresponding to different significance levels for which the normality hypothesis is rejected are tabulated in [Step 79]. [Step 79]

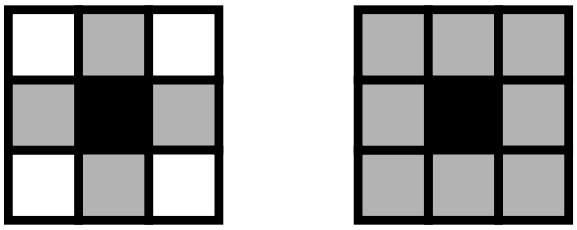
In this work, the Anderson-Darling test is used to examine the distributional properties of the random variation components of the electrical characteristics of a PUF (Sec. 5.3).

3.2.2 Moran's I Test

Moran's I is a standard measure to assess spatial autocorrelation between continuous random variables arranged on a two-dimensional plane [Mora 50]. The test statistic I is computed from the weighted sum over all pairwise cross products of a set of observations $\{x_i : i = 1, 2, ..., N\}$:

$$I = \frac{N}{W} \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij}(x_i - \hat{\mu})(x_j - \hat{\mu})}{\sum_i (x_i - \hat{\mu})^2} \tag{3.29}$$

with $W = \sum_i \sum_j w_{ij}$ and sample mean $\hat{\mu}$ (Eq. (3.27)). The weights w_{ij}, $i, j = 1, 2, ..., N$, define the elements of a $N \times N$ contiguity matrix which determines the spatial neighborhood structure of the observations x_i. Common contiguity definitions include rook- and queen-type neighborhood structures (Fig. 3.2). The former definition type considers all areal units sharing a common border as neighbors (Fig. 3.2a). In the latter case, the neighborhood of a cell is defined as the set of all adjacent areal units which are connected to a common grid vertex (Fig. 3.2b).



(a) **(b)**

Figure 3.2: Graphical illustration of the (a) rook- and (b) queen-type neighborhood definition. The central cell is colored black. Other cells included in the neighborhood of the central cell are displayed in grey. Both figures are adapted from the illustrations in [Scha 05].

The related hypothesis test assumes spatially uncorrelated observations under the null hypothesis conditional on a given neighborhood definition. Further information on how to derive the theoretical distribution of the test statistic is given in [Scha 05]. [Scha 05]

In Sec. 5.3, it will be shown how to apply the test to the experimental data of PUFs in order examine possible correlation effects between the electrical parameters from which response-bits are generated.

3.2.3 Chi-Squared Test

The Chi-squared test is used to determine whether the observed frequencies at which the outcomes of a random variable occur differ from the expected frequencies of random samples drawn from a hypothesized distribution. The test statistic Q is computed according to the equation:

$$Q = \sum_{k=0}^{N} \frac{(m_k - Mp_k)^2}{Mp_k} \tag{3.30}$$

with probability p_k to observe $k \in \{0, 1, ..., N\}$, number of outcomes m_k equal to k and the total number of outcomes $M = \sum_{k=1}^{N} m_k$. The test statistic follows a χ^2-distribution, $Q \sim \chi^2(r)$, specified by the degrees of freedom $r = N - 1 - s$ where s is the numbers of unknown distribution parameters to be estimated from the sample data. If Q exceeds the α-percentile of the χ^2-distribution for a given confidence level α, the null hypothesis that the data was drawn from the hypothesized distribution is rejected. [Feld 10]

The test in the aforementioned form will be used in Sec. 11.1 to evaluate statistical differences between the empirical inter-Hamming distribution calculated from experimental response data and its ideal distribution form, i.e. the binomial distribution $B(N, 0.5)$.

3.3 Change of Variable Transformation

In section 6.3, model equations are derived to investigate the effect of variable bit-error probabilities on the overall reproducibility of a PUF response. In order to determine the density distribution function of the bit-error probabilities, the change of variable transformation is used. The transformation allows to express

the probability density function p_Y of a random variable $Y = f(X)$ in terms of the density function p_X of X. Provided that f is a continuous bijective function, p_Y is computed according to the following equation [Gelm 14]:

$$p_Y(y) = \left| \frac{df(x)}{dx} \right| p_X(f^{-1}(y)) \,. \tag{3.31}$$

3.4 Median Polish Variation Decomposition

Median polish [Cres 93] is an iterative procedure which decomposes noisy elements y_{ij} of a two dimensional dataset $\{y_{ij} : i = 1, 2, ..., M; j = 1, 2, ..., N\}$ into a trend component $\bar{y}_{ij}$ and a random component u_{ij}:

$$y_{ij} = \bar{y}_{ij} + u_{ij} \,. \tag{3.32}$$

The decomposition is achieved by repeatedly subtracting the median from different subsets of the data until the trend components are removed. Detailed information on the algorithm used in this work is given in [Will 17a].

Median polish provides a simple method to estimate the systematic and random mismatch of primitive PUF devices (Sec. 5.3). The decomposition results are used to determine the distribution parameters of the electrical characteristics of a PUF which in turn serve as basis for the model building procedure described in later sections.

Chapter 4

Statistical Model Framework

The black box approach described in Sec. 2.3 is a convenient way to assess the properties of a given PUF design without having the need to specify the effect of the manufacturing process on the internal state of the PUF primitives. However, in order to develop a high-performance PUF concept easily adaptable to different processing conditions, it is essential to establish a functional relationship between the physical properties of a PUF and its security relevant features. Hence, the objective of this chapter is to provide a general framework that links the randomness, uniqueness and reproducibility of PUF responses to the fabrication-related distributional properties of the hardware-based random source from which the signatures are extracted.

In the case of Si-based PUFs, random sources typically originate from the variability of one or more electrical parameters of a device or system due to process variations. The overall variability of an electrical parameter includes both stochastic and deterministic components. Stochastic process variations are essential for the randomness of the extracted responses. The contribution of deterministic components to the parameter variability, in contrast, decreases the entropy of a response-bit by causing a tendency towards one of its logic states, effectively increasing the predictability of a response value. [Yin 14]

Consequently, in order to accurately model a PUF, both types of variation components have to be taken into account. However, modeling the variation components itself is a difficult task, giving rise to three challenges in particular:

- Modern semiconductor fabrication usually involves hundreds of processing steps all of which can contribute to the variability of an electrical param-

eter. Detailed knowledge of the individual contributions to the parameter variability might not be readily available. Furthermore, the functional dependence on the variation components might be unknown.

- The extent to which a variation component associated with a particular processing step contributes to the overall variability of a parameter generally depends on the parameter in question. For example, variations in the oxide thickness typically have a noticeable effect on the variability of the threshold voltage of a transistor, whereas the resistivity of a resistor is unaffected by this type of variation.

- The variability of a parameter often depends on the spatial scale on which individual processing steps operate (wafer level, die level, etc.) [Boni 96]. For example, the variability might change with increasing distance from the center of a wafer. In a similar fashion, structures placed at the edge of an array might display a different trend in their electrical characteristics than devices located close to the center of the array.

Previous attempts to analyze the distributional properties of mixed-signal PUFs were based on simple Gaussian process models of the form

$$Z_j = \mu_j + U_j + \varepsilon_j \,, \tag{4.1}$$

where the electrical parameter Z_j of the j-th device comprising a PUF instance is expressed as the sum of a systematic variation component μ_j, a random component U_j and a noise term ε_j [Mait 10, Yin 14, Scha 18]. The mean component μ_j represents the design value set by the processing technology of a device and the lumped contribution originating from systematic process variations. The components U_j are treated as independent random realizations of a zero-mean Gaussian distribution $N(0, \sigma^2_{U_j})$ characterizing the effect of stochastic process variations. Random measurement errors due to noise, for example, are absorbed in the term $\varepsilon_j \sim N(0, \sigma^2_{Noise,j})$. Figure 4.1 graphically illustrates the distributions of the different model components.

Given a dataset $\{z_j^{uv} : j = 1, 2, ..., 2N; u = 1, 2, ..., M; v = 1, 2, ..., O\}$ of $2N$ electrical characteristics of O PUF instances which were repeatedly measured M times, the parameters μ_j, $\sigma^2_{U_j}$ and $(\sigma^v_{Noise,j})^2$ are typically estimated from the sample mean and variances of suitable subsets of the data [Mait 10, Wild 14].

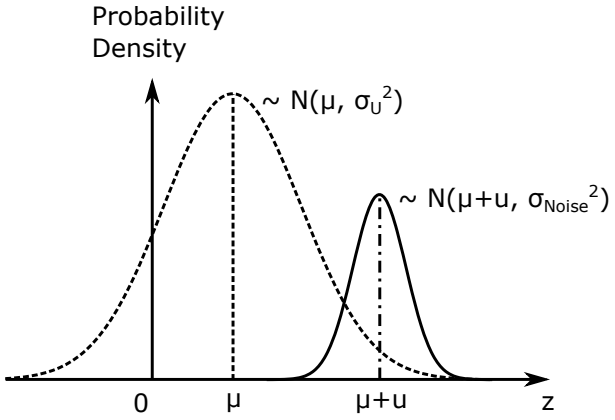


Figure 4.1: Gaussian process model. Adapted from [Delv 15].

An estimate $\hat{\mu}_j$ of the systematic variation component is obtained by averaging the parameter values of the different PUF instances over the individual device indices [Wild 14]:

$$\hat{\mu}_j = \frac{1}{O} \sum_{v=1}^{O} \frac{1}{M} \sum_{u=1}^{M} z_j^{uv} \,. \tag{4.2}$$

The $2N$ parameter variances are estimated by the sample variance $\hat{\sigma}_{U_j}^2$ computed for each device position j [Wild 14]:

$$\hat{\sigma}_{U_j}^2 = \frac{1}{O-1} \sum_{v=1}^{O} \left(\frac{1}{M-1} \sum_{u=1}^{M} z_j^{uv} - \hat{\mu}_j \right)^2 \,. \tag{4.3}$$

In a similar way, the variances of the measurement samples are taken as estimates of the the $O \cdot 2N$ error variances $(\hat{\sigma}_{Noise,j}^v)^2$ [Mait 10]:

$$(\hat{\sigma}_{Noise,j}^v)^2 = \frac{1}{M-1} \sum_{u=1}^{M} \left(z_j^{uv} - \frac{1}{M} \sum_{u=1}^{M} z_j^{uv} \right)^2 \,. \tag{4.4}$$

The simple Gaussian model specified in Eq. (4.1) is fully determined by the aforementioned sets of parameters.

In case of a simple differential quantization scheme, the response entropy of a PUF design can be readily calculated from the previously stated distribution parameters analogous to the computations described in [Wild 14]: under the assumption of independently and normally distributed PUF characteristics, the probability $P(\Delta S \leqslant 0)$ that an average pairwise difference $\Delta S = Z_m - Z_n$ of two

parameters yields a logical 0 is given by the expression:

$$P(\Delta S \leqslant 0) = \int_{-\infty}^{0} d\Delta s \frac{1}{\sqrt{2\pi\hat{\sigma}^2_{\Delta S}}} \exp\left(-\frac{(\Delta s - \hat{\mu}_{\Delta S})^2}{2\hat{\sigma}^2_{\Delta S}}\right) \tag{4.5}$$

$$= \frac{1}{2}\left[1 - \mathrm{erf}\left(\frac{\hat{\mu}_{\Delta S}}{\hat{\sigma}_{\Delta S}\sqrt{2}}\right)\right] \tag{4.6}$$

with mean difference $\hat{\mu}_{\Delta S} = \hat{\mu}_m - \hat{\mu}_n$ and pair variance $\hat{\sigma}^2_{\Delta S} = \hat{\sigma}^2_{U_m} + \hat{\sigma}^2_{U_n}$. $P(\Delta S \leqslant 0)$ corresponds to the shaded area under the graph of the normal probability density function displayed in Fig. 4.2. An upper bound on the Shannon entropy is obtained by substituting the N probabilities $\{P(\Delta S_i \leqslant 0) : i = 1, 2, ..., N\}$ into Eq. (2.3) and adding up the single-bit entropy values as stated in Eq. (2.2).

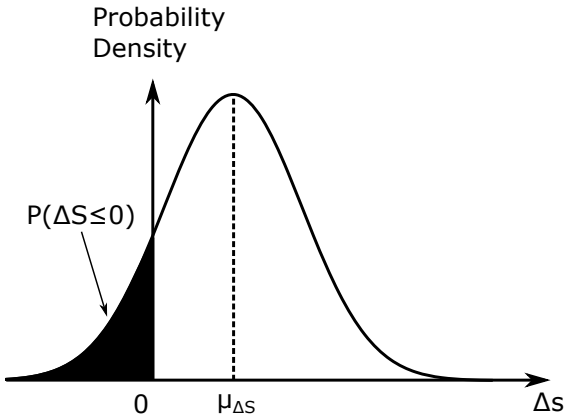


Figure 4.2: Graphical visualization of the single-bit probability $P(\Delta S \leqslant 0)$. Adapted from [Delv 15].

The authors of [Delv 15] proposed a similar model to compute the bit-error probabilities induced by Gaussian measurement noise: the probability $P(X_{flip})$ that a response-bit extracted from a pair difference Δs changes its value upon re-measurement is computed according to the equation:

$$P(X_{flip})|\Delta s = \int_{-\infty}^{0} d\Delta\tilde{s} \frac{1}{\sqrt{2\pi\hat{\sigma}^2_{\Delta S,Noise}}} \exp\left(-\frac{(\Delta\tilde{s} - |\Delta s|)^2}{2\hat{\sigma}^2_{\Delta S,Noise}}\right) \tag{4.7}$$

$$= \frac{1}{2}\left[1 - \mathrm{erf}\left(\frac{|\Delta s|}{\hat{\sigma}_{\Delta S,Noise}\sqrt{2}}\right)\right] \tag{4.8}$$

with total noise variance $\hat{\sigma}^2_{\Delta S,Noise} = \hat{\sigma}^2_{Z_m,Noise} + \hat{\sigma}^2_{Z_n,Noise}$ assuming independence of the parameter samples z_m and z_n. Temporal correlations between simultaneously measured parameter pairs and the corresponding expression of their noise

variance are considered in [Hill 13]. An estimate of the average bit-error probability can be obtained by computing the expected value of the conditional bit-error probability stated in Eq. (4.7) with respect to the distribution of the pair differences Δs as shown in [Scha 18].

Problematic to the aforementioned model approaches is the way how parameters are estimated. Averaging the samples over multiple PUF instances implies that the systematic mismatch associated with a particular device index is the same for all instances and that the random variation components of different PUFs are identically distributed. The underlying distribution parameter model is shown in Fig. 4.3a. In practice, neither of the two implications hold because of nonuniform spatial- and scale-specific processing variations. It is well known that the systematic and random mismatch between devices fabricated on the same die and on different dies depend on the wafer location [Boni 96, Stin 97, Will 17a]. Therefore, averaging over the pairwise mismatch between devices is likely to cancel out portions of the systematic variation component resulting in a low response bias. Consequently, the entropy calculated from these bit-probabilities tend to overestimate the randomness of a PUF response compared to a model which accounts for spatially varying mismatch. For example, in the case of 193 analyzed ring oscillator PUFs implemented on FPGAs, the average entropy estimate obtained under the fixed bias assumption was found to be more than 30 % higher than the corresponding value calculated for a variable device mismatch as will be shown in Chap. 5.

In order to overcome these model limitations, distribution parameters are determined by means of spatial variation decomposition techniques such as the median polish algorithm described in Sec. 3.4. Application of these methods allows to obtain spatially resolved estimates of the systematic and random device mismatch, thus eliminating the need for the fixed bias assumption. The refined model structure is displayed in Fig. 4.3b. In contrast to the existing model, the statistical properties of each PUF instance are defined by its own probability distribution. Thus the effect of spatially inhomogeneous process variations can be easily represented by different sets of distribution parameters.

In the following sections, it is shown how to expand the simple Gaussian model to include spatially dependent systematic mismatch and possibly correlated random variation components of neighboring devices. This form of Gaussian random

field approximation of the underlying manufacturing process provides a mathematically convenient framework to analyze the properties of PUFs on a statistical basis. Within the scope of this framework, model equations are derived to evaluate the response entropy and the inter- and intra-Hamming distance of arbitrary mixed-signal PUF designs. Furthermore, a fully probabilistic formulation of the response stability is given, which incorporates process-related variations in the noise variance, i.e. the noise variance itself is now treated as a random variable. The effect of these variations were not considered in the previous error models proposed by the authors of [Delv 15, Hill 13, Scha 18]. The response error model is further extended to include the effect of arbitrary noise sources such as temperature variations or changes in the supply voltage. The relationship between the response error probabilities and the noise sources is analyzed on the basis of a multiple regression model.

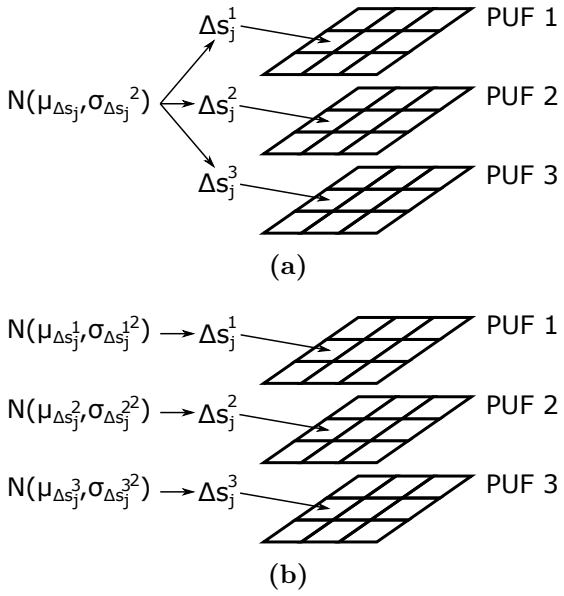


Figure 4.3: Distribution parameter models: (a) the mismatch of devices of different PUF instances positioned at the same spatial location is assumed to follow a common distribution which might differ between device locations. (b) the device mismatch of different PUF instances positioned at the same location is modeled separately for each instance and each device location.

4.1 Gaussian Random Field Approximation

As stated in [Delv 15] the Gaussian distribution assumption is motivated by the central limit theorem according to which the sum of a large number of independent random variations is approximately normally distributed [Brei 69], hence it provides simple means to model the mismatch of the physical characteristics of basic semiconductor devices, e.g. the frequency mismatch of ring oscillators [Wild 14].

Here, the generalized formulation [Brei 69] of the central limit theorem is considered as it is less restrictive with regard to the correlation structure imposed on the random effects. For the following considerations let $\{Z_i : i = 1, 2, ..., 2N\}$ denote a set of random variables associated with the electrical characteristics of $2N$ PUF primitives placed at different spatial locations i on a chip. In the context of semiconductor fabrication, the generalized central limit theorem states that if the numerous small processing-related contributions $X^k = [X_1, X_2, ..., X_{2N}]^{\mathrm{T}}$, $k = 1, 2, ..., K$, adding to the variability of the electrical parameters of a PUF are independently distributed, then the random vector $Z = [Z_1, Z_2, ..., Z_{2N}]^{\mathrm{T}}$ formed from the $2N$ device parameters of a PUF is approximately multivariate normally distributed:

$$Z = X^1 + X^2 + ... + X^K, \quad Z \sim N(\mu_Z, \Sigma_Z) \tag{4.9}$$

with mean vector $\mu_Z = [\mu_1, \mu_2, ..., \mu_{2N}]^{\mathrm{T}}$ and spatial covariance matrix Σ_Z

$$\Sigma_Z = \begin{pmatrix} \sigma_1^2 & \rho_{1,2}\sigma_1\sigma_2 & \cdots & \rho_{1,2N}\sigma_1\sigma_{2N} \\ \rho_{1,2}\sigma_1\sigma_2 & \sigma_2^2 & \cdots & \rho_{2,2N}\sigma_2\sigma_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{1,2N}\sigma_1\sigma_{2N} & \rho_{2,2N}\sigma_2\sigma_{2N} & \cdots & \sigma_{2N}^2 \end{pmatrix}, \tag{4.10}$$

where σ_i^2, $i = 1, 2, ..., 2N$, denotes the variance of the i-th vector component of Z. The correlation coefficient $\rho_{i,j} \in [-1, 1]$, $i, j = 1, 2, ..., 2N$, determines the spatial correlation between the i-th and j-th component of Z.

Figuratively speaking, the parameter value of each device realizing a PUF is regarded as realization of a Gaussian random variable labeled by a corresponding coordinate vector referencing the individual device locations (Fig. 4.4). The elements of μ_Z represent the deterministic components of the parameter values at

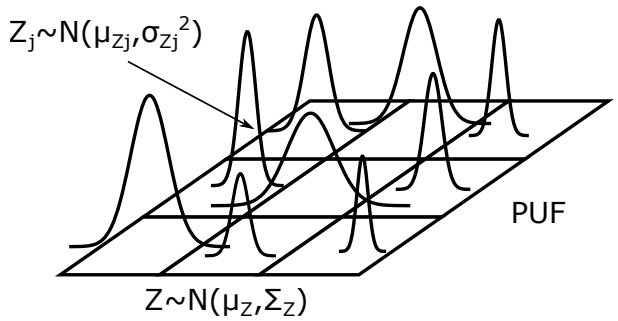


Figure 4.4: Graphical visualization of the Gaussian random field approximation of a PUF.

the different device locations. Σ_Z captures the variability of the parameters and controls the spatial smoothness of the random variation components [Scha 05].

It shall be highlighted that the Gaussian random field approximation greatly simplifies the model building process:

- Following this approximation, any electrical parameter is multivariate normally distributed. Therefore, model inference is independent of the design of a PUF.

- μ_Z and Σ_Z can be obtained by measuring the electrical characteristics of a PUF. Estimation of the distribution parameters is described in Sec. 5.1.3. Knowledge of the random effects associated with the fabrication process is not required.

4.2 Quantization Scheme

In order to generate binary responses, some form of quantization scheme has to be applied to the continuously distributed characteristics of a PUF.

4.2.1 Pairwise Bit-Generation

In the case of mixed-signal Weak PUFs such as the ring oscillator PUF, a simple way to generate a response-bit is to compare pairs of parameter values. For instance, if the inequality

$$z_i - z_j \leqslant 0 \tag{4.11}$$

holds for two parameter realizations z_i and z_j, the pair is assigned a logical 0 and logical 1 otherwise. The maximum number of obtainable bits is $N(2N-1)$ which equals the number of unique pairs that can be formed from a set of $2N$ parameters. However, by comparing the parameter values multiple times, trivial dependencies between the response-bits can arise, e.g. if $z_1 < z_2$ and $z_2 < z_3$ holds, then it follows that $z_2 < z_3$. In order to avoid these dependencies, exclusive pairs might be formed, i.e. each parameter is assigned to one and only one pair. [Suh 07]

Henceforth, selecting pairs of parameter values always implies the formation of exclusive pairs.

4.2.2 Response Distribution

Since the parameter components Z_i, $i = 1, 2, ..., 2N$, are assumed to be jointly normal, the distribution of the pairwise differences of parameter values can be easily determined. Let $S^1 = [Z_{i_1}, Z_{i_2}, ..., Z_{i_N}]^{\mathrm{T}}$ denote the random vector containing the first element of each of the N parameter pairs. Similarly, let $S^2 = [Z_{j_1}, Z_{j_2}, ..., Z_{j_N}]^{\mathrm{T}}$ be the random vector composed of the second element of each pair. It follows that S^1 and S^2 are multivariate normally distributed with distribution parameters $(\mu_{S^1}, \Sigma_{S^1})$ and $(\mu_{S^2}, \Sigma_{S^2})$, respectively. Since S^1 and S^2 are jointly normal, the pair difference vector $\Delta S = S^1 - S^2$ is also multivariate normally distributed[1]

$$\Delta S \sim N(\mu_{\Delta S}, \Sigma_{\Delta S}) \tag{4.12}$$

with mean vector

$$\mu_{\Delta S} = \mu_{S^1} - \mu_{S^2} \tag{4.13}$$

and covariance matrix

$$\Sigma_{\Delta S} = \Sigma_{S^1} + \Sigma_{S^2} - \Sigma_{S^1,S^2} - \Sigma_{S^2,S^1}\,, \tag{4.14}$$

where $\Sigma_{S^1,S^2} = \mathrm{cov}(S^1, S^2)$ and $\Sigma_{S^2,S^1} = \mathrm{cov}(S^2, S^1)$ denote the cross-covariance matrices of S^1 and S^2, respectively. The components of $\Sigma_{\Delta S}$ can be written in the form:

$$[\Sigma_{\Delta S}]_{uv} = \begin{cases} \sigma^2_{S^1_u} + \sigma^2_{S^2_u} - 2\rho_{S^1_u,S^2_u}\sigma_{S^1_u}\sigma_{S^2_u}, & \text{if } u = v \\ \sum\limits_{i,j=1}^{2} (-1)^{i+j}\rho_{S^i_u,S^j_v}\sigma_{S^i_u}\sigma_{S^j_v}, & \text{otherwise} \end{cases} \tag{4.15}$$

[1]This statement trivially follows from the fact that any linear combination of subsets of a multivariate normally distributed random vector is also jointly normal (Sec. 3.1.1).

with parameter variances $\sigma^2_{S_u^i}$ and correlation coefficients $\rho_{S_u^i,S_v^j}$, $i = 1, 2$, $u, v = 1, 2, ..., N$. The correlation coefficients measure the similarity between the random components of parameter value pairs at different spatial locations.

Dependent Variables

Given the distribution of ΔS, one can determine the probability $P(X = R)$ that a PUF instance generates a binary response $R = (r_1, r_2, ..., r_N)$. To compute $P(X = R)$, a vector-valued random function

$$f_{\Delta S}(R) = \left[(-1)^{r_1}\Delta S_1, (-1)^{r_2}\Delta S_2, ..., (-1)^{r_N}\Delta S_N\right]]^{\mathrm{T}} \tag{4.16}$$

is defined that maps a pair difference vector ΔS and a response R to a new pair difference vector $\Delta\tilde{S} = f_{\Delta S}(R)$. The components of $\Delta\tilde{S}$ and ΔS are identical for all components of R equal zero. All other components are changed in their respective sign. Consequently, the distributions of $\Delta\tilde{S}$ and ΔS differ in the sign of the corresponding components of their mean vectors. Furthermore, the sign of each off-diagonal component $[\Sigma_{\Delta\tilde{S}}]_{uv}$ for which $r_u \oplus r_v = 1$ holds has to be negated, i.e.$[\Sigma_{\Delta\tilde{S}}]_{uv} = -[\Sigma_{\Delta S}]_{uv}$. All other components are invariant under the transformation.

It follows that $P(X = R)$ can be calculated from the cumulative distribution function $F_{\Delta\tilde{S}}$ of the transformed pair difference vector $\Delta\tilde{S}$:

$$\begin{aligned} P(X = R) &= P(\Delta\tilde{S}_1 \leqslant 0, \Delta\tilde{S}_2 \leqslant 0, ..., \Delta\tilde{S}_N \leqslant 0) && (4.17)\\ &= F_{\Delta\tilde{S}}(0, 0, ..., 0) && (4.18)\\ &= \int\limits_{\mathbb{R}^{\mathbb{N}}_{-}} d\Delta s \, \frac{1}{\sqrt{(2\pi)^N \det(\Sigma_{\Delta\tilde{S}})}} \\ &\quad \cdot \exp\left(-\frac{1}{2}(\Delta s - \mu_{\Delta\tilde{S}})^{\mathrm{T}}\Sigma^{-1}_{\Delta\tilde{S}}(\Delta s - \mu_{\Delta\tilde{S}})\right) . && (4.19) \end{aligned}$$

Figure 4.5 shows a schematic illustration of the response distribution for $N = 2$. The ellipse displayed in the figure indicates the contour plot of the probability density function $p(\Delta s_1, \Delta s_2)$ of the bivariate normal distribution. The ellipse is centered around the mean vector $\mu_{\Delta S}$ and its shape is determined by the covariance matrix $\Sigma_{\Delta S}$. The intersection between the area under the density function and the coordinate quadrants represents the probability according to which the corresponding response value is realized. For example, the integrated area under

the density function in the first quadrant equals the probability $P(X = 11)$ that a PUF outputs the response $R = 11$.

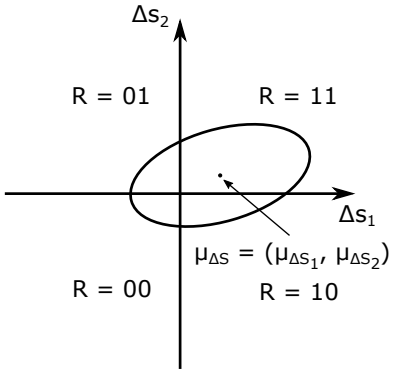


Figure 4.5: Graphical visualization of the response probability $P(X = R)$ for $R \in \{0, 1\}^2$.

In order to determine the properties of a PUF instance, Eq. (4.19) or variants thereof need to be computed for all possible 2^N responses. Although algorithms [Miwa 03, Crai 08] exist that allow efficient calculation of the integral term in Eq. (4.19), the large number of possible responses make this approach computationally unfeasible for all practically relevant key lengths $N \geqslant 64$.

Independent Variables

Efficient computation of the properties of a PUF instance is possible, if $\Sigma_{\Delta\tilde{S}}$ is a diagonal matrix, i.e. $\Sigma_{\Delta\tilde{S}} = \mathrm{diag}(\sigma^2_{\Delta S_1}, \sigma^2_{\Delta S_2}, ..., \sigma^2_{\Delta S_N})$ holds. For arbitrarily selected parameter pairs S^1 and S^2, diagonality of $\Sigma_{\Delta\tilde{S}}$ implies that the random effects contributing to the variability of the electrical device characteristics are spatially uncorrelated, i.e. the covariance matrix Σ_Z is a diagonal. The diagonal elements of $\Sigma_{\Delta\tilde{S}}$ are equal to the sum of the parameter variances:

$$\sigma^2_{\Delta S_i} = \sigma^2_{S_i^1} + \sigma^2_{S_i^2} \,. \tag{4.20}$$

In this case, the cumulative distribution function $F_{\Delta\tilde{S}}$ of $\Delta\tilde{S}$ factorizes. Thus, $P(X = R)$ can be written as a product of the form:

$$P(X = R) = \prod_{i=1}^{N} P(\Delta\tilde{S}_i \leqslant 0) = \prod_{i=1}^{N} F_{\Delta\tilde{S}_i}(0) \,, \tag{4.21}$$

where $P(\Delta\tilde{S}_i \leqslant 0)$, $i = 1, 2, ..., N$, denotes the single-bit probability that a pair difference $\Delta\tilde{S}_i$ results in a logical 0:

$$P(\Delta\tilde{S}_i \leqslant 0) = \int_{-\infty}^{0} d\Delta s_i \frac{1}{\sqrt{2\pi\sigma^2_{\Delta S_i}}} \exp\left(-\frac{(\Delta s_i - \mu_{\Delta\tilde{S}_i})^2}{2\sigma^2_{\Delta S_i}}\right). \tag{4.22}$$

4.3 Uniformity

Dependent Variables

The uniformity X_U of an N-bit response of a PUF instance is defined as the fraction of response-bits k/N, $k = 0, 1, ..., N$, equal to a logical 1 (Eq. (2.4)). Therefore, the distribution of X_U is determined by the probability mass function

$$P\left(X_U = \frac{k}{N}\right) = \sum_{R\in\{0,1\}^N} P(X = R) \cdot \delta_{x_{HW}(R),k}\,, \tag{4.23}$$

where $\delta_{j,k}$ denotes the Kronecker delta and $x_{HW}(R)$ the Hamming weight of a response R. The summation is carried out over all 2^N responses R. $P(X = R)$ equals the probability that a PUF instance outputs a response R as defined in Eq. (4.19).

Independent Variables

Computation of the uniformity property is possible, if the parameter values are independently distributed. In this case, $P(X_U = k/N)$ can be expressed in terms of the single-bit probabilities $P(\Delta S_i > 0) = 1\text{-}P(\Delta S_i \leqslant 0)$, $i = 1, 2, ..., N$ (Eq. (4.22)). It follows, that X_U is Poisson binomially distributed

$$\begin{aligned} P\left(X_U = \frac{k}{N}\right) = \sum_{A\in B_k} \prod_{i\in A} P(\Delta S_i > 0) \\ \cdot \prod_{j\in A^c} (1 - P(\Delta S_j > 0)) \end{aligned} \tag{4.24}$$

with B_k being defined as the set of subsets containing k integer indices taken from $\{1, 2, ..., N\}$. A^c denotes the complement of a set A.

Uniformity Distribution

Since the response uniformity of M PUF instances depends on the set of random variables $\{X_U^i : 1, 2, ..., M\}$, the probability mass function of the response

uniformity is a random function. A realization of the uniformity distribution is obtained by simulating a random sample from each of the M probability mass functions stated in either Eq. (4.23) or (4.24).

4.4 Bit-Aliasing

Dependent Variables

The bit-aliasing $X_{A,i}$ of the i-th response-bit of M PUF instances is determined by the fractional number of bits k/M, $k = 0, 1, ..., M$, equal to 1 (Eq. (2.5)). $X_{A,i}$ can be modeled by a Poisson binomial distribution

$$P\left(X_{A,i} = \frac{k}{M}\right) = \sum_{A \in B_k} \prod_{u \in A} P(X_i^u = 1) \cdot \prod_{v \in A^c} (1 - P(X_i^v = 1)) \,, \tag{4.25}$$

where B_k is the set containing all k-element subsets taken from $\{1, 2, ..., M\}$. $P(X_i^u = 1)$ denotes the probability that the i-th response-bit of PUF instance u is a logical 1, i.e.

$$P(X_i^u = 1) = \sum_{R \in \{0,1\}^N} P(X^u = R) \cdot \delta_{r_i,1} \,. \tag{4.26}$$

It shall be noted that the Poisson binomial model is only valid, if the spatial random components of different PUF instances are independent. This assumption is reasonable, if the random effects are mainly related to local process variations such that correlations between parameter values only exist for devices placed in close proximity to each other. If this assumption fails, then the joint distribution of the parameter values of all correlated PUF instances has to be taken into account when computing the response probabilities. This use case will not be further considered here.

Independent Variables

If the parameter values of each of the M PUF instances are spatially uncorrelated, then $P(X_i^u = 1) = P(\Delta S_i^u > 0)$ holds and Eq. (4.25) can be written in the form:

$$P\left(X_{A,i} = \frac{k}{M}\right) = \sum_{A \in B_k} \prod_{u \in A} P(\Delta S_i^u > 0) \cdot \prod_{v \in A^c} (1 - P(\Delta S_i^v > 0)) \,. \tag{4.27}$$

Bit-Aliasing Distribution

A realization of the bit-aliasing distribution of M PUF instances is generated by simulating a random sample from each of the N probability mass functions defined in Eq. (4.25) or (4.27).

4.5 Entropy

Given either the response probability $P(X = R)$ (Eq. (4.19)) or the single-bit probabilities $P(\Delta S_i \leqslant 0)$, $i = 1, 2, ..., N$ (Eq. (4.22)), the entropy can be directly calculated according to the equations stated in Sec. 2.3.1.

4.6 Uniqueness

Dependent Variables

The probability $P(X_{HD}^{u,v} = k)$ that the pairwise Hamming distance $X_{HD}^{u,v}$ between the responses X^u and X^v of two PUF instances u and v amounts to exactly k bits is given by the expression:

$$P(X_{HD}^{u,v} = k) = \sum_{R,R' \in \{0,1\}^N} P(X^u = R) P(X^v = R') \cdot \delta_{x_{HD}(R,R'),k} \,, \tag{4.28}$$

where $x_{HD}(R, R')$ denotes the Hamming distance between the two responses R and R'. Note that the equation only holds, if the response-bits of the two PUF instances u and v are independently distributed, otherwise the joint distribution of the pair differences of both PUF instances has to be considered.

Independent Variables

If the pair differences are uncorrelated, $P(X_{HD}^{u,v} = k)$ can be computed from the individual pair probabilities $P(X_i^u \neq X_i^v)$ that the i-th response-bit of PUF

instances u and v differ from each other. $P(X_i^u \neq X_i^v)$ is equal to the probability that either the response-bit of the first instance is a logical zero and that of the second one a logical 1 or the other way round:

$$\begin{aligned} P(X_i^u \neq X_i^v) = P(\Delta S_i^u \leqslant 0)(1 - P(\Delta S_i^v \leqslant 0)) \\ + (1 - P(\Delta S_i^u \leqslant 0))P(\Delta S_i^v \leqslant 0)\,. \end{aligned} \tag{4.29}$$

If the response-bits of PUF-instances u and v are independently distributed, then $P(X_{HD}^{u,v} = k)$ follows a Poisson binomial distribution:

$$\begin{aligned} P(X_{HD}^{u,v} = k) = \sum_{A \in B_k} \prod_{i \in A} P(X_i^u \neq X_i^v) \\ \cdot \prod_{j \in A^c} (1 - P(X_j^u \neq X_j^v))\,, \end{aligned} \tag{4.30}$$

where B_k is the set of subsets including k integer indices taken from $\{1, 2, ..., N\}$.

Inter-Hamming Distance Distribution

The distribution of the family of inter-Hamming distributions is approximated by simulating random draws from each of the $M(M-1)/2$ probability mass functions $P(X_{HD}^{u,v} = k)$ defined by Eq. (4.28) and (4.30), respectively.

4.7 Reproducibility

In case of the pairwise quantization scheme, a bit generated from a given pair difference Δs changes its value, if the added effect of the noise source $\delta\Delta s$ causes a change in the sign of Δs:

$$\delta\Delta s > |\Delta s|, \quad \text{if } \Delta s \leqslant 0 \tag{4.31}$$

$$\delta\Delta s \leqslant -\Delta s, \quad \text{if } \Delta s > 0\,. \tag{4.32}$$

The probability that the events stated in Eq. (4.31) and (4.32) occur determines the probability that a bit generated form Δs is unstable. With regard to the reproducibility of a response, determination of the probability $P(X_{flip} = r_{flip})$ that a number of response-bits indicated by a vector $r_{flip} \in \{0, 1\}^N$ change their logic state is of primary interest.

Dependent Variables

In order to determine $P(X_{flip} = r_{flip})$ the joint distribution $p(\Delta s, \delta\Delta s, u)$ of ΔS, $\delta\Delta S$ and other random variables U related to the former two quantities need to be specified.In order to model the joint density, it is convenient to express $p(\Delta s, \delta\Delta s, u)$ in terms of conditional probabilities:

$$p(\Delta s, \delta\Delta s, u) = p(\delta\Delta s|\Delta s, u)p(\Delta s|u)p(u) \tag{4.33}$$

$$= p(\Delta s|\delta\Delta s, u)p(\delta\Delta s|u)p(u)\,, \tag{4.34}$$

where $p(\cdot|\cdot)$ and $p(\cdot)$ denote the conditional and marginal distribution of their respective arguments.

This form of hierarchical description is commonly used throughout statistical data analysis to simply the model building process of complex problems. Instead of having to specify the joint probability of a possibly large number of random variables, the model can be broken down into smaller sub-models corresponding to a set of hierarchical stages, where the density function associated with each stage might be conditionally dependent on the outcome of other stages. [Gelm 14]

Within the scope of the proposed framework, using a conditional formulation facilitates the integration of random effects originating from other sources such as the process-related uncertainty of the noise variances into the proposed model framework. Proceeding from the representation of the joint distribution stated in Eq. (4.33), $P(X_{flip} = r_{flip})$ can be written in the form:

$$P(X_{flip} = r_{flip})|\Delta s, u = \int d\delta\Delta s\, p(\delta\Delta s|\Delta s, u) \tag{4.35}$$

for given realizations of Δs and u. The integration region $D_i(r_{flip,i})$ of each of the N tuples $(\Delta s_i, \delta\Delta s_i)$ are limited by the constraints given in Eq. (4.31) and (4.32) depending on the corresponding component of the indicator vector $r_{flip,i}$. The component value $r_{flip,i} = 1$ indicates that the i-th response-bit is unstable, hence D_i is defined as:

$$D_i(r_{flip,i} = 1) = \begin{cases} \{\delta\Delta s_i \in \mathbb{R} : \delta\Delta s_i > |\Delta s_i|\}, & \text{if } \Delta s_i \leqslant 0 \\ \{\delta\Delta s_i \in \mathbb{R} : \delta\Delta s_i \leqslant -\Delta s_i\}, & \text{otherwise}\,. \end{cases} \tag{4.36}$$

Accordingly, $r_{flip,i} = 0$ denotes a stable response-bit. In this case, D_i is deter-

mined by the following equations:

$$D_i(r_{flip,i}=0)=\begin{cases}\{\delta\Delta s_i\in\mathbb{R}:\delta\Delta s_i\leqslant|\Delta s_i|\}, & \text{if } \Delta s_i\leqslant 0\\ \{\delta\Delta s_i\in\mathbb{R}:\delta\Delta s_i>-\Delta s_i\}, & \text{otherwise.}\end{cases} \tag{4.37}$$

Since Δs, $\delta\Delta s$ and u are realizations of random variables, the probability $P(X_{flip}=r_{flip})$ is also a stochastic quantity. In practice, one can resort to simulation methods to approximate the distribution of $P(X_{flip}=r_{flip})$. For this purpose random samples are sequentially drawn from the density functions stated in Eq. (4.33) or (4.34). In case of Eq. (4.33), the sampling procedure involves the following steps:

- draw a random sample $\bar{u}$ from $p(u)$
- draw a random sample $\Delta\bar{s}$ from $p(\Delta s|\bar{u})$
- compute $P(X_{flip}=r_{flip})|\Delta\bar{s},\bar{u}$ according to Eq. (4.35)
- repeat the previous steps.

Given a set of realizations of all probabilities $P(X_{flip}=r_{flip})$, the overall reproducibility of a PUF instance can be evaluated. Let $R_k=\{r\in\{0,1\}^N:\sum_{i=1}^N r_i=k\}$ denote the set of indicator vectors, which feature exactly k ones, then the probability $P(X_{unstable}=k)$ that k bits of a PUF response are unstable is determined by the sum:

$$P(X_{unstable}=k)=\sum_{r_{flip}\in R_k}P(X_{flip}=r_{flip})\,. \tag{4.38}$$

$P(X_{unstable}=k)$ is determined conditional on the probabilities $P(X_{flip}=r_{flip})$, $r_{flip}\in R_k$. Since R_k contains k-permutations of N elements, computation of $P(X_{unstable}=k)$ is only possible for values of k for which $|N/2-k|$ is large. For example, let $N=256$ and $k=0,1,...5$, then the integral in Eq. (4.35) has to be evaluated more than 1.7×10^8 times per PUF instance, which is computationally unfeasible. Therefore, instead of solving Eq. (4.35), one could use the conditional and marginal distributions stated in Eq. (4.33) to simulate a set of noisy responses to approximate the distribution of the intra-Hamming distances.

Independent Variables

The expressions derived in the previous section can be greatly simplified, if the joint density $p(\Delta s, \delta\Delta s, u)$ factorizes into N componentwise density products:

$$p(\Delta s, \delta\Delta s, u) = \prod_{i=1}^{N} p(\delta\Delta s_i|\Delta s_i, u_i)p(\Delta s_i|u_i)p(u_i)\,. \tag{4.39}$$

In this case, the probability $P(X_{flip,i})$ that the i-th response-bit flips can be computed independently from the state of the remaining bits. Conditional on a tuple of realizations $(\Delta s_i, u_i)$, the probability $P(X_{flip,i})$ can be written as:

$$P(X_{flip,i}) = \begin{cases} \int_{-\Delta s_i}^{\infty} d\delta\Delta s_i\, p(\delta\Delta s_i|\Delta s_i, u_i)\,, & \text{if } \Delta s_i \leqslant 0 \\ \int_{-\infty}^{-\Delta s_i} d\delta\Delta s_i\, p(\delta\Delta s_i|\Delta s_i, u_i)\,, & \text{otherwise}\,. \end{cases} \tag{4.40}$$

The main benefit provided by separable bit-probabilities is the ability to model $P(X_{unstable} = k)$ in a computationally convenient way. Given a set of single-bit probabilities $\{P(X_{flip,i}) : i = 1, 2, ..., N\}$, the number of unstable response-bits $X_{unstable}$ is Poisson binomially distributed with probability mass function:

$$\begin{aligned} P(X_{unstable} = k) = \sum_{A \in B_k} \prod_{i \in A} P(X_{flip,i}) \\ \prod_{j \in A^c} (1 - P(X_{flip,i})) \end{aligned} \tag{4.41}$$

with B_k denoting the set of subsets comprised of k integer indices taken from $\{1, 2, ..., N\}$. Unlike the expression in Eq. (4.38), the Poisson binomial distribution can be computed; different computation methods are described in [Chen 97, Hong 13].

Intra-Hamming Distance Distribution

The underlying family of intra-Hamming distance distributions can be approximated by sampling random draws from each of the M probability mass functions $P(X_{unstable} = k)$ stated in Eq. (4.38) and (4.41), respectively.

4.8 Noise Source Distributions

In order to evaluate the reproducibility of a PUF response, the probability distributions of the noise source $p(\delta\Delta s|\Delta s, u)$, the pair differences $p(\Delta s|u)$ and other

related random variables $p(u)$ need to be specified. In the following two sections the effect of a lognormally and a normally distributed random source are considered. The lognormal distribution provides a generic model for an electrical noise source. In a complementary way, the normal distribution serves as a flexible model to incorporate the effect of varying operating conditions.

4.8.1 Lognormally Distributed Noise Source

Dependent variables

Noise-related deviations $\delta\Delta s_i$, $i = 1, 2, ..., N$, from the respective pair differences Δs_i are modeled as independent normally distributed random variables

$$\delta\Delta s \sim N(0, D_{\sigma^2_{Noise}}) \,. \tag{4.42}$$

Here 0_N denotes the zero vector of size N. The diagonal covariance matrix can be written in the form

$$D_{\sigma^2_{Noise}} = \mathrm{diag}(\sigma^2_{Noise,1}, ..., \sigma^2_{Noise,N}) \,, \tag{4.43}$$

where $\sigma^2_{Noise,i} = \sigma^2_{s_i^1} + \sigma^2_{s_i^2}$, $i = 1, 2, ..., N$, is the sum of the noise variances of the i-th pair difference Δs_i. It follows that the conditional probability $p(\delta\Delta s|\sigma^2_{noise})$ is determined by the expression:

$$\begin{aligned} p(\delta\Delta s|\sigma^2_{noise}) = & \frac{1}{\sqrt{(2\pi)^N \prod_{i=1}^{N} \sigma^2_{Noise,i}}} \\ & \cdot \exp\left(-\frac{1}{2}\delta\Delta s^{\mathrm{T}} D^{-1}_{\sigma^2_{Noise}} \delta\Delta s\right) . \end{aligned} \tag{4.44}$$

Since Δs is taken as mean value of repeated measurement samples, Δs is assumed to be independent of the random vectors $\sigma^2_{s^1}$ and $\sigma^2_{s^2}$, i.e. the approximation $p(\Delta s|\sigma^2_{noise}) \approx p(\Delta s)$ holds. Within the scope of the Gaussian random field approximation, $p(\Delta s)$ corresponds to the density function of the multivariate normal distribution $\Delta S \sim N(\mu_{\Delta S}, \Sigma_{\Delta S})$:

$$p(\Delta s) = \frac{\exp\left(-\frac{1}{2}(\Delta s - \mu_{\Delta S})^{\mathrm{T}} \Sigma^{-1}_{\Delta S}(\Delta s - \mu_{\Delta S})\right)}{\sqrt{(2\pi)^N \det(\Sigma_{\Delta S})}} \tag{4.45}$$

with mean vector $\mu_{\Delta S}$ and covariance matrix $\Sigma_{\Delta S}$.

Lastly, the probability density function $p(\sigma^2_{Noise})$ needs to be specified. The density function of the sum of two random variables[2] $\sigma^2_{s^1}$ and $\sigma^2_{s^2}$ can be expressed in terms of their joint distribution $p(\sigma^2_{s^1}, \sigma^2_{s^2})$:

$$p(\sigma^2_{Noise}) = \int\limits_{\mathbb{R}^N_+} d\sigma^2_{s^2} p(\sigma^2_{Noise} - \sigma^2_{s^2}, \sigma^2_{s^2}) \tag{4.46}$$

for $\sigma^2_{Noise} - \sigma^2_{s^2} > 0$ and 0 otherwise. The variance of a random variable is a positive quantity, therefore the joint density $p(\sigma^2_{s^1}, \sigma^2_{s^2})$ must be a strictly positive function. A convenient way to model the tuple of random vectors $(\sigma^2_{s^1}, \sigma^2_{s^2})$ is to assume a multivariate lognormal distribution

$$\begin{pmatrix} \sigma^2_{s^1} \\ \sigma^2_{s^2} \end{pmatrix} \sim \text{Lognormal}\left(\mu_s = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \Sigma_s = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} \right) \tag{4.47}$$

with joint distribution density:

$$\begin{aligned} p(\sigma^2_{s^1}, \sigma^2_{s^2}) = & \frac{1}{\sqrt{(2\pi)^{2N} \det(\Sigma_s)} \prod_{i=1}^{N} \sigma^2_{s^1_i} \sigma^2_{s^2_i}} \\ & \exp\left(-\frac{1}{2} \left(\begin{pmatrix} \log(\sigma^2_{s^1}) \\ \log(\sigma^2_{s^2}) \end{pmatrix} - \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \right)^{\mathrm{T}} \right. \\ & \left. \cdot \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}^{-1} \left(\begin{pmatrix} \log(\sigma^2_{s^1}) \\ \log(\sigma^2_{s^2}) \end{pmatrix} - \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \right) \right), \end{aligned} \tag{4.48}$$

if $\sigma^2_{s^1}, \sigma^2_{s^2} > 0$ and 0 otherwise. By substituting $\sigma^2_{s^1} = \sigma^2_{Noise} - \sigma^2_{s^2}$ into Eq. (4.48), the joint density function can be rewritten in the form:

$$\begin{aligned} p(\sigma^2_{Noise} - \sigma^2_{s^2}, \sigma^2_{s^2}) = & \frac{1}{\sqrt{(2\pi)^{2N} \det(\Sigma_s)} \prod_{i=1}^{N} (\sigma^2_{Noise,i} - \sigma^2_{s^2_i}) \sigma^2_{s^2_i}} \\ & \exp\left(-\frac{1}{2} \left(\begin{pmatrix} \log(\sigma^2_{Noise} - \sigma^2_{s^2}) \\ \log(\sigma^2_{s^2}) \end{pmatrix} - \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \right)^{\mathrm{T}} \right. \\ & \left. {\Sigma_s}^{-1} \left(\begin{pmatrix} \log(\sigma^2_{Noise} - \sigma^2_{s^2}) \\ \log(\sigma^2_{s^2}) \end{pmatrix} - \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \right) \right) \end{aligned} \tag{4.49}$$

for $\sigma^2_{Noise} - \sigma^2_{s^2} > 0$ and 0 otherwise. Estimates of the distributional parameters μ_s and Σ_s can be obtained by applying median polish to the logarithmized sample variance of repeatedly measured parameter values (Sec. 5.3.2).

[2] In general, the probability density function of the sum of two random variables X and Y is simply obtained by computing the derivative of the cumulative distribution function $F_Z = P(Z \leqslant z)$ with $Z = X + Y$ and $z = x + y$ [Fabe 12].

Given the distribution functions $p(\delta\Delta s|\sigma^2_{noise})$, $p(\Delta s)$ and $p(\sigma^2_{Noise})$ defined in Eq. (4.44), (4.45) and (4.46), respectively, the effect of the noise source is fully characterized.

Independent Variables

The joint density factorizes into the componentwise product of density functions

$$p(\delta\Delta s_i|\sigma^2_{Noise,i})p(\Delta s_i)p(\sigma^2_{Noise,i})\,, \tag{4.50}$$

if both variance-covariance matrices Σ_s and $\Sigma_{\Delta S}$ are diagonal. A diagonal covariance matrix of the form $\Sigma_s = \mathrm{diag}(\tilde{\sigma}^2_{s_1^1},...,\tilde{\sigma}^2_{s_N^1},\tilde{\sigma}^2_{s_1^2},...,\tilde{\sigma}^2_{s_N^2})$ implies that the probability density function $p(\sigma^2_{Noise})$ defined in Eq. (4.46) can be written as the product of the N marginal distributions $p(\sigma^2_{noise,i})$:

$$\begin{aligned} p(\sigma^2_{noise,i}) = \int_0^\infty d\sigma^2_{s_i^2} \frac{1}{2\pi\tilde{\sigma}_{s_i^1}\tilde{\sigma}_{s_i^2}(\sigma^2_{Noise,i}-\sigma^2_{s_i^2})\sigma^2_{s_i^2}} \\ \exp\left(-\frac{1}{2\tilde{\sigma}^2_{s_i^1}}(\log(\sigma^2_{Noise,i}-\sigma^2_{s_i^2})-\mu_{s_i^1})^2\right. \\ \left.-\frac{1}{2\tilde{\sigma}^2_{s_i^2}}(\log(\sigma^2_{s_i^2})-\mu_{s_i^2})^2\right)\,, \end{aligned} \tag{4.51}$$

if $\sigma^2_{Noise,i}-\sigma^2_{s_i^2} > 0$ and 0 otherwise. Similarly, a diagonal structure of $\Sigma_{\Delta S}$ allows to reformulate the density function

$$p(\Delta s) = \prod_{i=1}^{N} p(\Delta s_i) \tag{4.52}$$

in terms of the density functions $p(\Delta s_i)$, $i = 1, 2, ..., N$, of the N parameter difference pairs:

$$p(\Delta s_i) = \frac{1}{\sqrt{2\pi\sigma^2_{\Delta S_i}}}\exp\left(-\frac{(\Delta s_i-\mu_{\Delta S_i})^2}{2\sigma^2_{\Delta S_i}}\right)\,. \tag{4.53}$$

By assumption, the components of the deviation vector $\delta\Delta s$ are modeled as independent random variables with density functions

$$p(\delta\Delta s_i|\sigma^2_{Noise,i}) = \frac{1}{\sqrt{2\pi\sigma^2_{Noise,i}}}\exp\left(-\frac{\delta\Delta s_i^2}{2\sigma^2_{Noise,i}}\right) \tag{4.54}$$

for $i = 1, 2, ..., N$. Therefore, the case of independent variables described in the previous section applies.

4.8.2 Normally Distributed Noise Source

Dependent Variables

Given a set of variables x such as the operating temperature or the device age that affect the electrical characteristics of a PUF, the objective is to determine the joint probability $p(\Delta s, \delta\Delta s(x))$ of the N parameter differences Δs measured under reference conditions x_0 during the enrollment phase and its deviations $\delta\Delta s(x)$ when measured under conditions $x \neq x_0$ after deployment of the PUF. The relationship between x and the parameter values is evaluated in terms of a linear regression model. The regression coefficients are treated as random fields whose mean vectors and covariance structure are used to derive an expression for $p(\Delta s, \delta\Delta s(x))$.

Let $\tilde{z}_j = [\tilde{z}_j^1, \tilde{z}_j^2, ..., \tilde{z}_j^M]^{\mathrm{T}}$ denote a vector whose M components correspond to noisy measurement samples of the j-th parameter value taken at operating conditions $x = (x_1, x_2, ..., x_M)$. The relation between $\tilde{z}_j$ and x modeled in terms of the regression equation [Timm 02]:

$$\tilde{z}_j = X\beta_j + \varepsilon_j \tag{4.55}$$

with regression coefficient vector

$$\beta_j = [\beta_j^0, \beta_j^1, ..., \beta_j^K]^{\mathrm{T}}, \tag{4.56}$$

random error vector

$$\varepsilon_j = [\varepsilon_j^1, \varepsilon_j^2, ..., \varepsilon_j^M]^{\mathrm{T}} \tag{4.57}$$

and design matrix

$$X = \begin{pmatrix} b_0(x_1) & b_1(x_1) & \cdots & b_K(x_1) \\ b_0(x_2) & b_1(x_2) & \cdots & b_K(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ b_0(x_M) & b_1(x_M) & \cdots & b_K(x_M) \end{pmatrix}, \tag{4.58}$$

where $\{b_k(\cdot) : k = 0, 1, ..., K\}$, denotes a set of basis functions, e.g. polynomials $b_k(x_l) = x_l^k$.

For the following analysis, the error components $\{\varepsilon_j^u : u = 1, 2, ..., M\}$ are assumed to be independently and normally distributed, i.e.

$$\varepsilon_j | \{\sigma_{\tilde{z}_j^u}^2 : u = 1, 2, ..., M\} \sim N\left(0_M, D_{\sigma_{\tilde{z}_j}^2}\right) \tag{4.59}$$

with diagonal matrix $D_{\sigma^2_{\tilde{z}_j}}$ defined as

$$D_{\sigma^2_{\tilde{z}_j}} = \mathrm{diag}(\sigma^2_{\tilde{z}_j^1}, \sigma^2_{\tilde{z}_j^2}, ..., \sigma^2_{\tilde{z}_j^M}) \tag{4.60}$$

and error variances $\sigma^2_{\tilde{z}_j^u}$, $u = 1, 2, ..., M$. Note that the distribution of the error terms has to be taken conditionally on the error variances, since they are also considered as random variables due to the effect of stochastic process variations (Sec. 4.8.1). In case of unequal error variances, weighted least squares provides the best linear unbiased estimator $\hat{\beta}_j$ of the regression coefficients [Gelm 14]:

$$\hat{\beta}_j = (X^{\mathrm{T}} W X)^{-1} X^{\mathrm{T}} W \tilde{z}_j \,, \tag{4.61}$$

where W denotes a diagonal weight matrix whose components W_{uu}, $u = 1, 2, ..., M$, are set equal to the inverse error variances [Abdi 07], i.e.

$$W_{uu} = \sigma^{-2}_{\tilde{z}_j^u} \,. \tag{4.62}$$

The next step is to determine the distributional properties of $\hat{\beta}_j$ taken into account process-related variability of the regression coefficients β_j. Motivated by the Gaussian random field approximation, the lumped effect of systematic and random process variations is expressed in terms of an additive model with Gaussian random effects, i.e. β_j is written as the sum of a deterministic component μ_{β_j} and a normally distributed stochastic component U_{β_j}:

$$\beta_j = \mu_{\beta_j} + U_{\beta_j} \,. \tag{4.63}$$

By substituting Eq. (4.55) and (4.63) into (4.61), $\hat{\beta}_j$ can be expressed in terms of μ_{β_j} and U_{β_j}:

$$\hat{\beta}_j = \mu_{\beta_j} + U_{\beta_j} + (X^{\mathrm{T}} W X)^{-1} X^{\mathrm{T}} W \varepsilon_j \,. \tag{4.64}$$

Since it is reasonable to assume the random effects U_{β_j} and the error vector ε_j to be conditionally independent given the sample variances $\sigma^2_{\tilde{z}_j^u}$, $u = 1, 2, ..., M$, it follows from Eq. (4.64) that $\hat{\beta}_j$ is multivariate normally distributed:

$$\hat{\beta}_j \sim N(\mu_{\hat{\beta}_j}, \Sigma_{\hat{\beta}_j}) \tag{4.65}$$

with mean vector $\mu_{\hat{\beta}_j}$

$$\mu_{\hat{\beta}_j} = \mu_{\beta_j} \tag{4.66}$$

and covariance matrix $\Sigma_{\hat{\beta}_j}$ given by the equation

$$\Sigma_{\hat{\beta}_j} = \Sigma_{\beta_j} + ((X^{\mathrm{T}}WX)^{-1}X^{\mathrm{T}}W)D_{\sigma^2_{\tilde{z}_j}}((X^{\mathrm{T}}WX)^{-1}X^{\mathrm{T}}W)^{\mathrm{T}}, \tag{4.67}$$

where the first term corresponds to the covariance structure resulting from the random effects U_{β_j} and the second one to the transformed error variance $D_{\sigma^2_{\tilde{z}_j}}$.

Considering the variation decomposition of $\tilde{z}_j$ into a systematic component $\mu_{\tilde{z}_j}$, multivariate normal random effects $U_{\tilde{z}_j}$ and an error term ε_j, it follows from Eq. (4.61) that the distribution parameters $\mu_{\hat{\beta}_j}$ and $\Sigma_{\hat{\beta}_j}$ can be identified with the mean vector $\mu_{\tilde{z}_j}$ and covariance matrix $\Sigma_{\tilde{z}_j}$ of $\tilde{z}_j$. The mean vector is determined by the expression

$$\mu_{\hat{\beta}_j} = (X^{\mathrm{T}}WX)^{-1}X^{\mathrm{T}}W\mu_{\tilde{z}_j}\,. \tag{4.68}$$

The covariance matrix can be written in the form

$$\Sigma_{\hat{\beta}_j} = ((X^{\mathrm{T}}WX)^{-1}X^{\mathrm{T}}W)(\Sigma_{\tilde{z}_j} + D_{\sigma^2_{\tilde{z}_j}})((X^{\mathrm{T}}WX)^{-1}X^{\mathrm{T}}W)^{\mathrm{T}}. \tag{4.69}$$

It is apparent from the two equations that the mean vector and the covariance matrix of $\hat{\beta}_j$ are fully determined by the mean vector, covariance matrix and error variances of $\tilde{z}_j$. All three quantities can be estimated from the electrical measurement data of a PUF.

Further model development requires specification of the joint distribution of the regression coefficient estimates $\hat{\beta}_j$, $j = 1, 2, ..., 2N$, to take into account the spatial aspect of the problem at hand. First of all, it is assumed that the $2N$ random effects $U_{\tilde{z}_j}$ follow a multivariate normal distribution. This approximation is based on the assumption that the random effects at different operating conditions are highly correlated (Sec. 5.3.3). Secondly, the combined random effects $U_{\tilde{z}} = [U^{\mathrm{T}}_{\tilde{z}1}, ..., U^{\mathrm{T}}_{\tilde{z}2N}]^{\mathrm{T}}$ and error terms $\varepsilon = [\varepsilon_1^{\mathrm{T}}, \varepsilon_2^{\mathrm{T}}, ..., \varepsilon_{2N}^{\mathrm{T}}]^{\mathrm{T}}$ are considered independent conditional on the sample variances $\sigma^2_{\tilde{z}_j^u}$, $j = 1, 2, ..., 2N$, $u = 1, 2, ..., M$. Lastly, the combined error term ε is modeled by a jointly zero-mean Gaussian distribution:

$$\varepsilon \sim N(0_{2MN}, \Sigma_\varepsilon)\,. \tag{4.70}$$

The covariance matrix Σ_ε can be written in block form:

$$\Sigma_\varepsilon = \begin{pmatrix} \Sigma_{\varepsilon_1} & \Sigma_{\varepsilon_1,\varepsilon_2} & \cdots & \Sigma_{\varepsilon_1,\varepsilon_{2N}} \\ \Sigma_{\varepsilon_2,\varepsilon_1} & \Sigma_{\varepsilon_2} & \cdots & \Sigma_{\varepsilon_2,\varepsilon_{2N}} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{\varepsilon_{2N},\varepsilon_1} & \Sigma_{\varepsilon_{2N},\varepsilon_2} & \cdots & \Sigma_{\varepsilon_{2N}} \end{pmatrix} \tag{4.71}$$

with covariance matrices $\Sigma_{\varepsilon_j} := D_{\sigma^2_{\tilde{z}_j}}$, $j = 1, 2, ..., 2N$, (Eq. (4.60)) and cross-covariance matrices $\Sigma_{\varepsilon_j,\varepsilon_k}$, $j, k = 1, 2, ..., 2N$, $j \neq k$. The cross-covariance matrices allow to include spatiotemporal correlations between parameter values measured in close succession at different spatial locations on a chip.

Under these assumption, the coefficient vector $\hat{\beta} = [\hat{\beta}_1^{\mathrm{T}}, \hat{\beta}_2^{\mathrm{T}}, ..., \hat{\beta}_{2N}^{\mathrm{T}}]^{\mathrm{T}}$ is multivariate normally distributed

$$\hat{\beta} \sim N(\mu_{\hat{\beta}}, \Sigma_{\hat{\beta}})\,. \tag{4.72}$$

The $(K+1) \times 2N$ components of the mean vector

$$\mu_{\hat{\beta}} = \begin{pmatrix} \mu_{\hat{\beta}_1} \\ \mu_{\hat{\beta}_2} \\ \vdots \\ \mu_{\hat{\beta}_{2N}} \end{pmatrix} \tag{4.73}$$

are defined according to Eq. (4.68):

$$\mu_{\hat{\beta}_j^u} = \sum_{t=0}^{K} \sum_{s=1}^{M} [(X^{\mathrm{T}} W X)^{-1}]_{ut} b_t(x_s) \sigma^{-2}_{\tilde{z}_j^s} \mu_{\tilde{z}_j^s}\,. \tag{4.74}$$

The covariance structure of $\hat{\beta}$ can be represented by a block matrix $\Sigma_{\hat{\beta}}$ similar to the one of the combined error term:

$$\Sigma_{\hat{\beta}} = \begin{pmatrix} \Sigma_{\hat{\beta}_1} & \Sigma_{\hat{\beta}_1,\hat{\beta}_2} & \cdots & \Sigma_{\hat{\beta}_1,\hat{\beta}_{2N}} \\ \Sigma_{\hat{\beta}_2,\hat{\beta}_1} & \Sigma_{\hat{\beta}_2} & \cdots & \Sigma_{\hat{\beta}_2,\hat{\beta}_{2N}} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{\hat{\beta}_{2N},\hat{\beta}_1} & \Sigma_{\hat{\beta}_{2N},\hat{\beta}_2} & \cdots & \Sigma_{\hat{\beta}_{2N}} \end{pmatrix}. \tag{4.75}$$

The block diagonal components correspond to the $K \times K$ covariance matrices defined in Eq. (4.69). Each matrix $\Sigma_{\hat{\beta}_j}$ captures the covariance structure between different coefficient estimates at a given spatial location j. The off-diagonal block elements define the cross-covariance $\Sigma_{\hat{\beta}_j,\hat{\beta}_k}$ between the regression coefficient vectors at two different spatial locations j and k. Computation of $\mathrm{cov}(\hat{\beta}_j, \hat{\beta}_k)$ yields the following expression for $\Sigma_{\hat{\beta}_j,\hat{\beta}_k}$:

$$\begin{aligned} \Sigma_{\hat{\beta}_j,\hat{\beta}_k} &= \Sigma_{\beta_j,\beta_k} \\ &\quad + ((X^{\mathrm{T}} W X)^{-1} X^{\mathrm{T}} W) \Sigma_{\varepsilon_j,\varepsilon_k} ((X^{\mathrm{T}} W X)^{-1} X^{\mathrm{T}} W)^{\mathrm{T}} \end{aligned} \tag{4.76}$$

with Σ_{β_j,β_k} being the cross-covariance matrix of the parameters vectors β_j and β_k. In order to determine $\Sigma_{\hat{\beta}_j,\hat{\beta}_k}$ from regression data, Eq. (4.76) needs to be reformulated in terms of the measurable parameter values:

$$\begin{aligned}\Sigma_{\hat{\beta}_j,\hat{\beta}_k} = &((X^{\mathrm{T}}WX)^{-1}X^{\mathrm{T}}W)(\Sigma_{\tilde{z}_j,\tilde{z}_k} + \Sigma_{\varepsilon_j,\varepsilon_k}) \\ &\cdot ((X^{\mathrm{T}}WX)^{-1}X^{\mathrm{T}}W)^{\mathrm{T}}.\end{aligned} \tag{4.77}$$

The cross-covariance matrix $\Sigma_{\tilde{z}_j,\tilde{z}_k}$ determines the correlation between two parameters $\tilde{z}_j$ and $\tilde{z}_k$ measured at different spatial locations and operating conditions.

To simplify notation in subsequent sections, the components of the covariance and cross-covariance matrices are restated in terms of the regression variances and correlation coefficients. Let $\sigma^2_{\hat{\beta}_j^u}$, $u = 0, 1, ..., K$, be the variances of the regression coefficient estimates defined as

$$\begin{aligned}\sigma^2_{\hat{\beta}_j^u} &= \mathrm{var}(\hat{\beta}_j^u) \\ &= \sum_{q,t=0}^{K}\sum_{r,s=1}^{M}[(X^{\mathrm{T}}WX)^{-1}]_{ut}b_t(x_s)\sigma^{-2}_{\tilde{z}_j^s}[\Sigma_{\tilde{z}_j} + D_{\sigma^2_{\tilde{z}_j}}]_{sr} \\ &\quad \cdot \sigma^{-2}_{\tilde{z}_j^r}b_q(x_r)[(X^{\mathrm{T}}WX)^{-1}]_{uq}\,.\end{aligned} \tag{4.78}$$

Furthermore, let $\rho_{\hat{\beta}_j^u,\hat{\beta}_k^v}$, $u, v = 0, 1, ..., K$ denote the correlation coefficient of $\hat{\beta}_j^u$ and $\hat{\beta}_k^v$:

$$\rho_{\hat{\beta}_j^u,\hat{\beta}_k^v} = \frac{1}{\sigma_{\hat{\beta}_j^u}\sigma_{\hat{\beta}_k^v}}\,\mathrm{cov}(\hat{\beta}_j^u, \hat{\beta}_k^v) \tag{4.79}$$

$$\begin{aligned}&= \frac{1}{\sigma_j^u\sigma_k^v}\sum_{q,t=0}^{K}\sum_{r,s=1}^{M}[(X^{\mathrm{T}}WX)^{-1}]_{ut}b_t(x_s)\sigma^{-2}_{\tilde{z}_j^s} \\ &\quad \cdot [\Sigma_{\tilde{z}_j,\tilde{z}_k} + \Sigma_{\varepsilon_j,\varepsilon_k}]_{sr}\sigma^{-2}_{\tilde{z}_j^r}b_q(x_r)[(X^{\mathrm{T}}WX)^{-1}]_{vq}\end{aligned} \tag{4.80}$$

with $\Sigma_{\tilde{z}_j,\tilde{z}_j} \equiv \Sigma_{\tilde{z}_j}$ and $\Sigma_{\varepsilon_j,\varepsilon_j} \equiv \Sigma_{\varepsilon_j}$. Given these two definitions, the matrix elements $\Sigma_{\hat{\beta}_j^u,\hat{\beta}_j^v}$, $u, v = 0, 1, ..., K$, can be expressed as

$$\Sigma_{\hat{\beta}_j^u,\hat{\beta}_j^v} = \begin{cases}\sigma^2_{\hat{\beta}_j^u}, & \text{if } u = v \\ \rho_{\hat{\beta}_j^u,\hat{\beta}_j^v}\sigma_{\hat{\beta}_j^u}\sigma_{\hat{\beta}_j^v}, & \text{otherwise.}\end{cases} \tag{4.81}$$

Similarly, the components of the cross-covariance matrix $\Sigma_{\hat{\beta}_j^u,\hat{\beta}_k^v}$, $u, v = 0, 1, ..., K$, are determined by the following equation:

$$\Sigma_{\hat{\beta}_j^u,\hat{\beta}_k^v} = \rho_{\hat{\beta}_j^u,\hat{\beta}_k^v}\sigma_{\hat{\beta}_j^u}\sigma_{\hat{\beta}_k^v}\,. \tag{4.82}$$

Note that $\Sigma_{\hat{\beta}}$ generally contains the spatial autocorrelation structure of the random field and, in addition, the cross-correlation effects between the random field evaluated at different operating conditions. Therefore, the spatial mean, the variance-covariance structure and the cross-covariance structure of the random fields have to be estimated simultaneously from the data. If the random effects of the random fields at different operating conditions are linearly related, then the distribution parameters can be easily estimated from the decomposed variation data obtained by median polishing as shown in later sections.

Given the coefficient vector $\hat{\beta}$, a regression model for the pair differences $\Delta\tilde{s}$ at operating conditions x can be formulated:

$$\Delta\tilde{s} = \Delta\hat{\beta}^0 + \sum_{k=1}^{K} b_k(x)\Delta\hat{\beta}^k \tag{4.83}$$

with coefficient pair differences $\Delta\hat{\beta}^0 = \hat{\beta}^0_{S^1} - \hat{\beta}^0_{S^2}$ and $\Delta\hat{\beta}^k = \hat{\beta}^k_{S^1} - \hat{\beta}^k_{S^2}$, where $\hat{\beta}^u_{S^1} = [\hat{\beta}^u_{S^1_1}, \hat{\beta}^u_{S^1_2}, ..., \hat{\beta}^u_{S^1_N}]$ and $\hat{\beta}^u_{S^2} = [\hat{\beta}^u_{S^2_1}, \hat{\beta}^u_{S^2_2}, ..., \hat{\beta}^u_{S^2_N}]$ denote the vectors containing the first and second element of the N pairs of the u-th regression coefficient, respectively. Without loss of generality, $b_0(\cdot)$ was set to 1 in Eq. (4.83).

The next step is to express Δs and $\delta\Delta s$ in terms of the regression coefficients. According to Eq. (4.83), $\Delta\tilde{s} = \Delta s = \Delta\hat{\beta}^0$ holds for $x = x_0$, hence Δs can be associated with $\Delta\hat{\beta}^0$. Similarly, $\sum_{k=1}^{K} b_k(x)\Delta\hat{\beta}^k$ yields an expression for $\delta\Delta s(x)$. Based on these relations, the probability density functions $p(\delta\Delta s|\Delta s)$ and $p(\Delta s)$ can be specified. The conditional probability density $p(\delta\Delta s|\Delta s)$ is derived from the joint density $p(\Delta s, \delta\Delta s)$. Since $\hat{\beta}$ is assumed to be multivariate normal, ΔS and $\delta\Delta S(x)$ also follow a multivariate normal distribution[3]:

$$\begin{pmatrix} \Delta S \\ \delta\Delta S(x) \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu_{\Delta S} \\ \mu_{\delta\Delta S} \end{pmatrix}, \begin{pmatrix} \Sigma_{\Delta S} & \Sigma_{\Delta S,\delta\Delta S(x)} \\ \Sigma^{\mathrm{T}}_{\Delta S,\delta\Delta S(x)} & \Sigma_{\delta\Delta S(x)} \end{pmatrix} \right) . \tag{4.84}$$

The mean vector components $\mu_{\Delta S}$ and $\mu_{\delta\Delta S(x)}$ are given by the relations

$$\mu_{\Delta S} = \mu_{\Delta\hat{\beta}^0} = \mu_{\hat{\beta}^0_{S^1}} - \mu_{\hat{\beta}^0_{S^2}} \tag{4.85}$$

and

$$\mu_{\delta\Delta S(x)} = \sum_{k=1}^{K} b_k(x)\mu_{\Delta\hat{\beta}^k} = \sum_{k=1}^{K} b_k(x)\left(\mu_{\hat{\beta}^k_{S^1}} - \mu_{\hat{\beta}^k_{S^2}}\right) , \tag{4.86}$$

[3]Any subset of jointly normal random variables follows a jointly normal distribution (Sec. 3.1.1).

respectively. The covariance matrix $\Sigma_{\Delta S}$ is determined by the variance of ΔS:

$$\begin{aligned}\Sigma_{\Delta S} &= \mathrm{var}(\Delta S) && (4.87)\\ &= \mathrm{var}(\Delta\hat{\beta}^0) && (4.88)\\ &= \Sigma_{\hat{\beta}^0_{S^1}} + \Sigma_{\hat{\beta}^0_{S^2}}, - \Sigma_{\hat{\beta}^0_{S^1},\hat{\beta}^0_{S^2}} - \Sigma_{\hat{\beta}^0_{S^2},\hat{\beta}^0_{S^1}}\ , && (4.89)\end{aligned}$$

where $\Sigma_{\hat{\beta}^0_{S^1}}$ and $\Sigma_{\hat{\beta}^0_{S^2}}$ denote the covariance matrices of the regression coefficient vectors $\hat{\beta}^0_{S^1}$ and $\hat{\beta}^0_{S^2}$, respectively. With a slightly modified notation for the regression parameter variances and correlation coefficients defined in Eq. (4.78) and (4.80), respectively, the components of the covariance matrix $\Sigma_{\hat{\beta}^u_{S^i_r},\hat{\beta}^u_{S^i_s}}$, $u = 0, 1, ...K$, $i = 1, 2$, can be written in the form

$$\Sigma_{\hat{\beta}^u_{S^i_r},\hat{\beta}^u_{S^i_s}} = \begin{cases} \sigma^2_{\hat{\beta}^u_{S^i_r}}, & \text{if } r = s \\ \rho_{\hat{\beta}^u_{S^i_r},\hat{\beta}^u_{S^i_s}}\sigma_{\hat{\beta}^u_{S^i_r}}\sigma_{\hat{\beta}^u_{S^i_s}}, & \text{otherwise} \end{cases} \tag{4.90}$$

for all coefficient pair components $r, s = 1, 2, ..., N$. Accordingly, $\Sigma_{\hat{\beta}^0_{S^1},\hat{\beta}^0_{S^2}}$ and $\Sigma_{\hat{\beta}^0_{S^2},\hat{\beta}^0_{S^1}}$ represent the cross-covariance matrices of the regression coefficient vectors $\hat{\beta}^0_{S^1}$ and $\hat{\beta}^0_{S^2}$, whose components $\Sigma_{\hat{\beta}^u_{S^i_r},\hat{\beta}^v_{S^j_s}}$, $u, v = 0, 1, ...K$, $i, j = 1, 2$, $r, s = 1, 2, ..., N$, are determined by the following equation:

$$\Sigma_{\hat{\beta}^u_{S^i_r},\hat{\beta}^v_{S^j_s}} = \rho_{\hat{\beta}^u_{S^i_r},\hat{\beta}^v_{S^j_s}}\sigma_{\hat{\beta}^u_{S^i_r}}\sigma_{\hat{\beta}^v_{S^j_s}} \tag{4.91}$$

with $u \neq v$ and $i \neq j$. The covariance matrix $\Sigma_{\delta\Delta S(x)}$ is obtained by computing the variance of $\delta\Delta S(x)$:

$$\begin{aligned}\Sigma_{\delta\Delta S(x)} &= \mathrm{var}\left(\delta\Delta S(x)\right) && (4.92)\\ &= \mathrm{var}\left(\sum_{k=1}^{K} b_k(x)\Delta\hat{\beta}^k\right) && (4.93)\\ &= \sum_{k=1}^{K} b_k^2(x)\left(\Sigma_{\hat{\beta}^k_{S^1},\hat{\beta}^k_{S^1}} + \Sigma_{\hat{\beta}^k_{S^2},\hat{\beta}^k_{S^2}}\right)\\ &\quad + \sum_{\substack{k,l=1\\ k\neq l}}^{K} b_k(x)b_l(x)\left(\Sigma_{\hat{\beta}^k_{S^1},\hat{\beta}^l_{S^1}} + \Sigma_{\hat{\beta}^k_{S^2},\hat{\beta}^l_{S^2}}\right) && (4.94)\\ &\quad - \sum_{k,l=1}^{K} b_k(x)b_l(x)\left(\Sigma_{\hat{\beta}^k_{S^1},\hat{\beta}^l_{S^2}} + \Sigma_{\hat{\beta}^k_{S^2},\hat{\beta}^k_{S^1}}\right)\ .\end{aligned}$$

Lastly, computation of the covariance of ΔS and $\delta\Delta S$ yields an expression for the cross-covariance matrix $\Sigma_{\Delta S,\delta\Delta S}$:

$$\Sigma_{\Delta S,\delta\Delta S(x)} = \operatorname{cov}\left(\Delta S, \delta\Delta S(x)\right) \tag{4.95}$$

$$= \operatorname{cov}\left(\Delta\hat{\beta}^0, \sum_{k=1}^{K} b_k(x)\Delta\hat{\beta}^k\right) \tag{4.96}$$

$$\begin{aligned} = \sum_{k=1}^{K} b_k(x) \Big(&\Sigma_{\hat{\beta}^0_{S^1},\hat{\beta}^k_{S^1}} + \Sigma_{\hat{\beta}^0_{S^2},\hat{\beta}^k_{S^2}} \\ &-\Sigma_{\hat{\beta}^0_{S^1},\hat{\beta}^k_{S^2}} - \Sigma_{\hat{\beta}^0_{S^2},\hat{\beta}^k_{S^1}}\Big) \,. \end{aligned} \tag{4.97}$$

For known values of $\mu_{\Delta S}$, $\mu_{\delta\Delta S}$, $\Sigma_{\Delta S}$, $\Sigma_{\delta\Delta S(x)}$ and $\Sigma_{\Delta S,\delta\Delta S(x)}$, the conditional joint density $p(\delta\Delta s|\Delta s)$ can be specified. The conditional distribution of two jointly normal random vectors follows a normal distribution, hence $\delta\Delta S$ conditional on ΔS is also multivariate normally distributed[4]:

$$p(\delta\Delta s|\Delta s) = \mathcal{N}\left(\delta\Delta s|\mu_{\delta\Delta S|\Delta S}, \Sigma_{\delta\Delta S(x)|\Delta S}\right) \tag{4.98}$$

with conditional mean vector $\mu_{\delta\Delta S|\Delta S}$

$$\mu_{\delta\Delta S(x)|\Delta S} = \mu_{\delta\Delta S(x)} + \Sigma_{\delta\Delta S(x),\Delta S}\Sigma^{-1}_{\Delta S}\left(\Delta s - \mu_{\Delta S}\right) \tag{4.99}$$

and covariance matrix

$$\Sigma_{\delta\Delta S(x)|\Delta S} = \Sigma_{\delta\Delta S(x)} - \Sigma^{\mathrm{T}}_{\Delta S,\delta\Delta S(x)}\Sigma^{-1}_{\Delta S}\Sigma_{\Delta S,\delta\Delta S(x)} \,. \tag{4.100}$$

The random vector ΔS also follows a multivariate normal distribution with density function:

$$p(\Delta s) = \mathcal{N}\left(\Delta s|\mu_{\Delta S}, \Sigma_{\Delta S}\right) \,. \tag{4.101}$$

The definitions of the mean vector $\mu_{\Delta S}$ and covariance matrix $\Sigma_{\Delta S}$ are given in Eq. (4.85) and (4.89), respectively. The normally distributed noise source is fully determined by the density functions $p(\delta\Delta s|\Delta s)$ (Eq. (4.98)) and $p(\Delta s)$ (Eq. (4.101)).

Independent Variables

A componentwise factorization of the joint probability density function

$$p(\Delta s, \delta\Delta s(x)) = \prod_{i=1}^{N} p(\delta\Delta s_i(x)|\Delta s_i)p(\Delta s_i) \tag{4.102}$$

[4]See the paragraph on conditional normal distributions in Sec. 3.1.1.

is achieved, if the corresponding covariance matrices $\Sigma_{\Delta S}$ and $\Sigma_{\delta\Delta S|\Delta S}$ are diagonal, which requires the off-diagonal elements of $\Sigma_{\Delta S}$, $\Sigma_{\delta\Delta S}$ and $\Sigma_{\Delta S,\delta\Delta S(x)}$ to vanish. As can be seen from Eq. (4.89), (4.94) and (4.97) diagonalization of all matrices is accomplished, if the regression coefficient estimates belonging to different pairs are uncorrelated, i.e.

$$\rho_{\hat{\beta}^u_{S^i_r},\hat{\beta}^v_{S^j_s}} = 0 \tag{4.103}$$

holds for all $u, v = 0, 1, ..., K$, $i, j = 1, 2$ and $r, s = 1, 2, ..., N$ with $r \neq s$. According to Eq. (4.80), the correlation coefficients are identically zero, if

$$\Sigma_{\tilde{z}_{S^i_r},\tilde{z}_{S^j_s}} + \Sigma_{\varepsilon_{S^i_r},\varepsilon_{S^j_s}} = 0_{N,N} \tag{4.104}$$

holds, which generally requires that each term itself vanishes. In this case, the expressions for $\Sigma_{\Delta S}$ and $\Sigma_{\delta\Delta S|\Delta S}$ can be simplified as follows. The diagonal elements $[\Sigma_{\Delta S}]_{rr}$, $r = 1, 2, ..., N$, are given by

$$[\Sigma_{\Delta S}]_{rr} = \sigma^2_{\hat{\beta}^0_{S^1_r}} + \sigma^2_{\hat{\beta}^0_{S^2_r}} - 2\rho_{\hat{\beta}^0_{S^1_r},\hat{\beta}^0_{S^2_r}} \sigma_{\hat{\beta}^0_{S^1_r}} \sigma_{\hat{\beta}^0_{S^2_r}} \,. \tag{4.105}$$

According to Eq. (4.100), the matrix elements $[\Sigma_{\delta\Delta S|\Delta S}]_{rr}$, $r = 1, 2, ..., N$, are of the form

$$\begin{aligned}
[\Sigma_{\delta\Delta S|\Delta S}]_{rr} = &\sum_{k=1}^{K} b_k^2(x) \left(\sigma^2_{\hat{\beta}^k_{S^1_r}} + \sigma^2_{\hat{\beta}^k_{S^2_r}} \right) \\
&+ \sum_{k\neq l=1}^{K} \sum_{i=1}^{2} b_k(x) b_l(x) \rho_{\hat{\beta}^k_{S^i_r},\hat{\beta}^l_{S^i_r}} \sigma_{\hat{\beta}^k_{S^i_r}} \sigma_{\hat{\beta}^l_{S^i_r}} \\
&- \sum_{k,l=1}^{K} \sum_{i\neq j=1}^{2} b_k(x) b_l(x) \rho_{\hat{\beta}^k_{S^i_r},\hat{\beta}^l_{S^j_r}} \sigma_{\hat{\beta}^k_{S^i_r}} \sigma_{\hat{\beta}^l_{S^j_r}} \\
&- \frac{\left(\sum_{k=1}^{K} b_k(x) \sum_{i,j=1}^{2} (-1)^{i+j} \rho^{0,k}_{S^i_u,S^j_u} \sigma^0_{S^i_u} \sigma^k_{S^j_u} \right)^2}{\sigma^2_{\hat{\beta}^0_{S^1_r}} + \sigma^2_{\hat{\beta}^0_{S^2_r}} - 2\rho_{\hat{\beta}^0_{S^1_r},\hat{\beta}^0_{S^2_r}} \sigma_{\hat{\beta}^0_{S^1_r}} \sigma_{\hat{\beta}^0_{S^2_r}}} \,.
\end{aligned} \tag{4.106}$$

The corresponding mean vector components $[\mu_{\delta\Delta S|\Delta S}]_r$, $r = 1, 2, ..., N$, are:

$$\begin{aligned}
[\mu_{\delta\Delta S|\Delta S}]_r = &\sum_{k=1}^{K} b_k(x) \left(\mu_{\hat{\beta}^k_{S^1_r}} - \mu_{\hat{\beta}^k_{S^2_r}} \right) \\
&- \frac{\sum_{k=1}^{K} b_k(x) \sum_{i,j=1}^{2} (-1)^{i+j} \rho_{\hat{\beta}^0_{S^i_r},\hat{\beta}^k_{S^j_r}} \sigma_{\hat{\beta}^0_{S^i_r}} \sigma_{\hat{\beta}^k_{S^j_r}}}{\sigma^2_{\hat{\beta}^0_{S^1_r}} + \sigma^2_{\hat{\beta}^0_{S^2_r}} - 2\rho_{\hat{\beta}^0_{S^1_r},\hat{\beta}^0_{S^2_r}} \sigma_{\hat{\beta}^0_{S^1_r}} \sigma_{\hat{\beta}^0_{S^2_r}}} \\
&\cdot \left(\Delta s_r - \mu_{\hat{\beta}^0_{S^1_r}} + \mu_{\hat{\beta}^0_{S^2_r}} \right) \,.
\end{aligned} \tag{4.107}$$

Provided that the requirements stated above are fulfilled, the reproducibility of PUF response can be evaluated within the scope of the single-bit probability model described in Sec. 4.7.

4.9 Summary

A detailed model framework capable of simulating the response randomness, uniqueness and reproducibility of arbitrary mixed-signal PUFs was presented. The conceptual idea behind the framework is based on the assumption that the manufacturing-related variability of the analog characteristics of a PUF can be expressed in terms of a Gaussian random field.

The effect of nonuniform systematic process variations and spatially correlated random mismatch, are modeled in terms of a spatially dependent mean and a non-diagonal covariance structure. Simultaneous estimates of both distribution parameters are obtained by the use of standard variation decomposition procedures well-known in the area of spatial statistics. Application of these methods results in an individual set of distribution parameters for each PUF instance. Differences in the parameters between PUF instances naturally reflect spatial or temporal disparities in the processing conditions responsible for inter-die and batch-to-batch variation, for example.

Given the distribution type and parameter estimates, the response probabilities that a PUF instance outputs a particular bit-sequence were derived in the case of dependent and independent random variables modeling the electrical device parameters of a PUF. Conditional on the response probabilities, expressions for the inter- and intra-Hamming distance were formulated. In the general use case of unequal response probabilities, it was shown that the distribution of the Hamming distances are random realization of a family of possible distributions, members of which can be simulated by nested Monte Carlo sampling.

The intra-Hamming distance was evaluated for two types of noise sources representing the effect of measurement noise and variable operating conditions on the reproducibility of a PUF response, respectively. The former noise source, was modeled in terms of lognormally distributed noise variances to account for spatially dependent response error probabilities. Response errors caused by changes in the operating conditions were analyzed by means of multiple linear weighted

regression with random coefficients. The random variable treatment provides a simple way to include systematic and random mismatch between temperature or voltage coefficients.

Chapter 5

Model Building Example: Ring Oscillator PUF

In this chapter, the proposed Gaussian random field model procedure (Chap. 4) is applied to the ring oscillator PUF implementation described in [Mait 10]. Validation of the model is based on the uniformity, bit-aliasing, uniqueness and reproducibility of PUF responses computed for two different pair selection schemes. The choice of the aforementioned metrics is motivated by the ability to compute these measures directly from the observed raw responses of a PUF realization, thus facilitating the comparison between the empirical estimates and the theoretical values predicted by the model in a straight-forward way.

5.1 Description of the Model Building Process

5.1.1 Data Acquisition

Prerequisite to the subsequent analysis is access to experimental data of a number of PUF instances. The experimental data should include electrical measurements of all primitive devices realizing a PUF instance. In order to determine the effect of measurement noise on the stability of a PUF design, repeated samples of the electrical characteristics have to be acquired. Similarly, modeling the reproducibility of the PUF responses under variable operating conditions requires multiple measurements of the electrical characteristics at different operating points. If regression analysis is to be employed, the number of operating points and the

spacing between them should reflect the functional relationship between the electrical parameters of a PUF and the operating conditions. For example, in order to model a nonlinear dependency measurements at three or more different operating conditions are required.

5.1.2 Data Processing

Cleansing

The distribution form and the distribution parameters used to model the response properties of a PUF are estimated from experimental data, thus outliers due to systematic measurement errors are likely to affect and falsify the statistical inference based on these estimates. Consequently, erroneous records should be removed in order to avoid measurement artifacts in the analysis results.

Computation of the Noise Variances

The noise error model developed in Sec. 4.8.1 requires estimates of the variance of the measurement noise as input. In practice, the noise variances are estimated from the sample variance (Eq. (3.28)) of the repeated measurement samples of each electrical characteristic.

5.1.3 Distribution Parameter Estimation

The electrical data obtained from a PUF instance is modeled as Gaussian random field, i.e. the measured characteristics of each PUF primitive or suitable transformations thereof are considered as realization of a multivariate normal distribution defined by the distribution parameters μ_Z and Σ_Z (Sec. 4.1).

Estimation of the distribution parameters of a Gaussian random field is a well known problem in spatial statistics. A variety of readily applicable parametric and nonparametric methods exist that can be applied to the problem at hand (see [Scha 05] for a detailed overview). By the use of these methods, μ_Z and Σ_Z can be determined from a single realization of the underlying random field. The lack of samples from which to estimate the covariance structure is taken into account by assuming local stationarity[1] of the random components U_{Z_i}, $i = 1, 2, ..., 2N$, i.e.

[1] In this context, spatial stationarity means that the covariance structure of the random

the random components are assumed to have equal variances. Thus, the random field approach is less restrictive regarding the distributional properties of a PUF than conventional methods which assume common distribution parameters for all PUF instances[2].

Within the scope of this work, median polish (Sec. 3.4) was used to determine estimates of μ_Z and Σ_Z. The choice of this method was motivated by the flexibility of the algorithm to adapt to different spatial trend surfaces while preserving local features without having the need to specify a parametric form of μ_Z and Σ_Z. Therefore, the procedure can be directly applied to the measured electrical parameters of a PUF. In case of the noise variances, the data is logarithmized prior to the polishing in order normalize the data[3]. If the measurements were taken at different operating conditions, polishing is performed separately on each dataset for all operating points.

The polishing algorithm decomposes the measurement values of the $2N$ electrical parameters z_i, $i = 1, 2, ..., 2N$, of a PUF instance into $2N$ trend components $\bar{z}_i$ and random effects u_i (Eq. (3.32)). The trend components can be associated with the mean vector μ_Z of the Gaussian random field and the variance of the random effects yields an estimate of the spatial covariance matrix Σ_Z of the random components.

The structure of the covariance matrix is examined by the use of Moran's I test for spatial autocorrelation (Sec. 3.2.2). In order to calculate the test statistic, the spatial contiguity of the primitive devices realizing a PUF needs to be specified. Since the devices are typically placed in form of an array shaped layout to minimize the effect of systematic mismatch, the queen-type neighborhood definition provides a natural choice to measure the spatial proximity between the primitives. Based on the chosen neighborhood definition, Moran's I test is performed for 1st and higher order sets of neighbors[4]. A positive test result at a given significance level indicates spatially correlated random effects. In this

components only depends on the distance between two points of the random field and that the mean of the random field is constant over the spatial domain of interest [Scha 05].

[2]Refer to the brief discussion at the beginning of Chap. 4.

[3]The noise variances are assumed to be lognormally distributed (Sec. 4.8.1), thus the log-transformed data follows a normal distribution.

[4]The set of 1st order neighbors of a given PUF primitive includes all other devices in the immediate vicinity of the primitive. Higher orders are defined accordingly, e.g. 2nd order neighbors comprise all next nearest devices to the primitive.

case, the covariance structure can be parameterized by means of simultaneous or conditional autoregressive models [Scha 05]. If Moran's I test turns out negative, the effect of spatial autocorrelation is assumed to be negligible. Under this assumption, the resulting Gaussian random field model can be expressed in terms of a spatially varying mean vector μ_Z and independent and identically distributed zero-mean random components featuring a common variance σ_Z^2. Consequently, Eq. (4.9) can be written in the form

$$Z \sim N(\mu_Z, \sigma_Z^2 \mathbb{1}_{2N\times 2N})\,, \tag{5.1}$$

where $\mathbb{1}_{m\times n}$ denotes the unity matrix of size $m \times n$. Given that the random effects are statistically independent, the Anderson-Darling test for normality (Sec. 3.2.1) is employed to examine the validity of the Gaussian random field approximation.

In addition, Quantile-Quantile (QQ) plots are used to visually inspect the distribution form of the random effects. The QQ-plot is a graphical tool to evaluate the likeliness of random sample data being drawn from a hypothesized distribution. For this purpose, the empirical quantiles of the sample are compared to the theoretical quantiles of the hypothesized distribution. If the assumed distribution form provides a reasonable fit to the observed data, then the graph of the sample quantiles should closely follow the identity line. Deviations from the identity line indicate a possible misspecification of the sample distribution. [Fabe 12]

5.1.4 Computation of the Performance Metrics

Given estimates of the distribution parameters, computation of the different response properties and performance metrics described in Chap. 4 is straightforward.

First of all, the appropriate set of model equations has to be selected. The choice depends on whether the parameter values were tested positively for spatial autocorrelation effects. If this is the case, the equations derived for dependent random variables have to be used. As explained in the previous chapter, exponentially many computing operations are required to calculate the performance metrics, thus exact computation of the equations is generally unfeasible. In practice, the response properties can be approximated by means of Monte Carlo simulations. For this purpose, the random field representation of a PUF (Eq. (4.45)) is

used to simulate new random realizations of the electrical parameters of each device from which binary responses are extracted. The properties of the simulated PUF instances are then analyzed by means of the empirical evaluation methods described in Sec. 2.3, e.g. the inter- and intra-Hamming metrics.

If the spatial correlation effects are found to be statistically insignificant, the parameter values can be treated as independent random variables. Consequently, the model equations for independent random variables apply. Since efficient computation of the performance metrics is only possible under the assumption of statistical independence, parameters might be approximated by independent random variables, even if they are correlated given that the spatial correlation effects are sufficiently weak. In order to assess the validity of this approximation, the distribution of the calculated test statistic can be compared to the hypothesized distribution obtained for uncorrelated data. The approximation is considered reasonable, if the difference between the quantiles of the two distributions is small. Naturally, the model results have to be consistent with the experimental data.

Assuming the approximation holds, the next step is to determine the sets of probabilities characterizing the response-bias and the bit-error probabilities defined in Eq. (4.22) and Eq. (4.40), respectively.

The former set of probabilities allows to compute the bitwise response entropy (Eq. (2.2)), the probability mass function of the response uniformity (Eq. (4.24)) and the mass function of the bit-aliasing measure (Eq. (4.27)) for each PUF instance. The mass function of the inter-Hamming distance between two PUF responses (Eq. (4.30)) is calculated for the single-bit probabilities of all unique pairs of PUF instances. The second set of probabilities is used to compute the mass function of the intra-Hamming distance (Eq. (4.41)) for each PUF response. Depending on the type of noise source used to determine the error probabilities, the resulting mass function quantifies either the effect of measurement noise or the effect of variable operating conditions on the stability of a PUF response.

Lastly, the overall distribution of each performance metric can be simulated by generating random samples from the mass function of each PUF instance.

5.1.5 Software Implementation

The stochastic model framework and the different parts of the model building procedure were implemented in the general-purpose programming languages Python

[Pyth 18] and R [R Co 18].

Handling and processing of the experimental data was accomplished by the use of the Python packages 'pandas' [McKi 10] and 'numpy' [Mill 11]. The former package, in particular, provides a great number of functions to manage and manipulate large datasets containing measurement data of hundreds of PUF instances acquired under various operating conditions.

The median polish procedure used to estimate the distribution parameter of the random field model was implemented as user-defined function in Python. The procedure corresponds to the basic polish algorithm described in [Will 17a]. Moran's I test for spatial correlation is performed using the predefined function 'moran.test' included in the R package 'spdep' [Biva 13]. The Anderson-Darling test statistic A^2 as defined in Eq. (3.25) is calculated via the built-in function 'anderson' of the module 'scipy.stats' contained in the Python package 'scipy' [Jone 01].

Furthermore, all probability distributions except for the Poisson binomial distribution were implemented by their respective distribution function included in the 'scipy.stats' module. The R package 'poibin' [Hong 13] was used to carry out all calculations involving the Poisson binomial distribution. Each of these predefined distribution functions provides subroutines for the computation of the probability mass/density and cumulative distribution function. In addition, all distribution functions feature a built-in pseudorandom number generator to draw random samples from a given distribution. The random number functionality was used to carry out Monte Carlo simulations of the response properties and performance metrics such as the distribution of the inter- and intra-Hamming distance.

5.2 Dataset

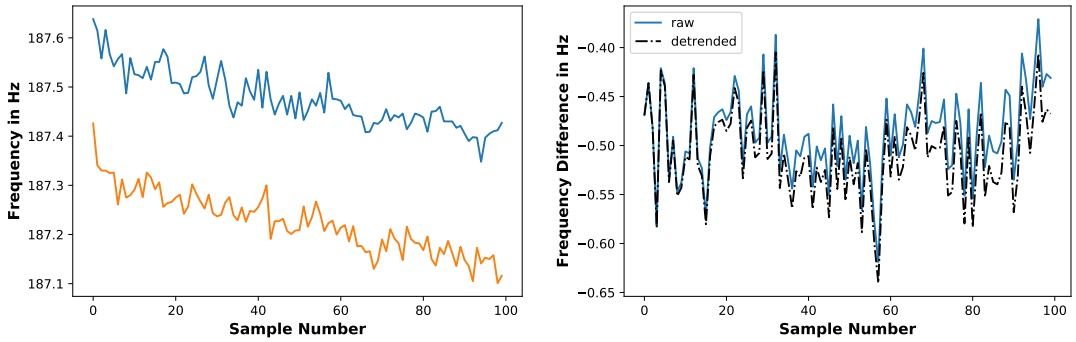
The subsequent analysis was carried out for a ring oscillator PUF dataset made publicly available by the authors of [Mait 10]. The dataset contains the measured oscillation frequencies of 193 ring oscillator PUF instances implemented on commercially available FPGAs fabricated in a 90 nm technology. Each PUF instance is comprised of 512 ring oscillators arranged in a 32×16 array. For each ring oscillator, 100 oscillation frequency measurements were taken under standard operating conditions, i.e. at room temperature ($\sim 25\,^{\circ}\mathrm{C}$) and at a supply voltage

level of 1.2 V. In addition, the frequency of 5 ring oscillator PUF instances were measured at different operating temperatures and supply voltage settings. For a fixed supply voltage level of 1.2 V, temperature dependent measurements were performed at 35 °C, 45 °C, 55 °C and 65 °C. In a similar fashion, room temperature measurements were made at supply voltage levels of 0.96 V, 1.08 V, 1.20 V, 1.32 V and 1.44 V.

5.3 Distribution Parameter Estimation

5.3.1 Oscillation Frequency

The raw measurement data of individual ring oscillators showed a decrease in the oscillation frequency for subsequent measurement samples possibly caused by self-heating of the devices (Fig. 5.1a). Computation of the sample mean and variance,



(a) (b)

Figure 5.1: Time series of (a) the measured frequency samples of two ring oscillators and (b) the pairwise differences of the samples. In the latter case, the differences were calculated from the raw samples and the detrended frequencies.

however, requires stationary samples in order to determine unbiased estimates of both quantities. Therefore, the series of frequency values was detrended using a low-order polynomial fit. In practice, first to third order polynomials were sufficient to remove the observed temporal trend in the data. The sample variance required for the analysis presented in the following sections can be directly calculated from the residuals of the polynomial fit. The mean estimates correspond to the fit values at the different time steps of the measurement series, hence the

mean oscillation frequency is time dependent. In principle, the spatial analysis of the data has to be carried out for each set of ring oscillators of a PUF at different time steps. However, the differential pair selection scheme allows to significantly simplify the model building process because the temporal drift in the frequencies approximately cancels out for the pairwise differences of measurement samples.

Figure 5.1b shows two series of pairwise frequency differences calculated for two sets of measurement samples. One of the two series was computed from the raw frequency measurement samples (Fig. 5.1a). The other series was calculated from the difference of the detrended frequency samples. As can be seen from Fig. 5.1b, the detrended series closely matches the graph of the series determined from the raw data. Hence, the same stationary behavior observed for the measured frequencies differences could be reproduced to a good approximation by simply eliminating the temporal trends from the data of the individual time series. Therefore, the mean frequency of individual oscillators can be determined from the detrended measurement series, effectively obviating the need to explicitly model the time dependence of the samples.

In a next step, the mean and standard deviation were estimated from the detrended frequency samples. The processed frequency data of each PUF instance was approximated by a Gaussian random field of dimension 32×16 with individual mean vector μ_Z and covariance matrix Σ_Z. Application of the median polish algorithm to the data yielded the decomposition of the oscillation frequency into a deterministic and a random component. The deterministic frequency component includes the nominal oscillation frequency set by the fabrication technology and contributions originating from deterministic process variations, e.g. spatial heat gradients in a furnace or lithographic effects [Bohm 13]. In contrast, the random frequency component accounts for stochastic processing effects such as fluctuations in the doping concentration or variation in the line edge roughness of semiconductor structures [Sedc 06]. The spatial distributions of the raw frequency data, the systematic components and the random components of one of the 193 analyzed PUF instance are shown in Fig. 5.2a to 5.2c, respectively. The deterministic component was taken as estimate of μ_Z. Σ_Z was determined from the random frequency components. In order to determine the covariance structure, Moran's I test for spatial autocorrelation was applied to the random components of the individual PUF instances. The test was performed for nearest

neighbors at different spatial lags up to a maximum lag order of 6. The results showed that in 98 % of the performed test, no significant correlations were found at a significance level of 0.01.

To reassess normality of the oscillation frequencies, the random components were plotted in form of QQ-plots (Sec. 5.1.3). Figure 5.3 shows the QQ-plot of the computed 512 random components of a PUF instance. As can be seen from the figure, the data can be well approximated by a normal distribution. Slight deviations from normality can be observed at the tail regions of the distribution, i.e. the sample distribution features longer tails than the hypothesized normal distribution. The results were quantitatively confirmed by the Anderson-Darling test which generally rejected the null hypothesis of normally distributed random effects at 0.01 significance level, mainly because of the distribution tails. Therefore, the normality assumption is only approximately valid. However, as is demonstrated in later sections, the Gaussian model holds reasonably well when comparing the empirical response properties of the tested ring oscillator PUF design to the results obtained under the proposed analysis framework.

Based on these results Σ_Z was modeled by a diagonal matrix $\sigma_Z^2 \mathbb{1}_{512\times512}$, where σ_Z^2 denotes the variance of the random components. The parameter estimates were validated by generating random draws from the resulting multivariate normal distribution $N(\mu_Z, \sigma_Z^2 \mathbb{1}_{512\times512})$. Application of the two sample Anderson-Darling test to the measured frequency data and 100 simulated samples per PUF instance revealed no significant differences between the underlying population distributions at a significance level of 0.01. Furthermore, visual comparison of the measurement and the simulated datasets showed reasonable agreement with regard to their spatial features (compare Fig. 5.2a and 5.2d).

5.3.2 Sample Variance

Application of the distribution model described in Sec. 4.8.1 requires the distribution form of the sample noise and the spatial properties of the logarithmized sample variances $\log(\sigma_z^2)$ to be specified.

In order to test the assumption of normally distributed measurement noise, the Anderson-Darling test statistic was computed for the detrended repeatedly measured frequency samples of 193 PUF instances. The hypothesis tests were evaluated at a confidence level of 0.01. On average, the normality hypotheses

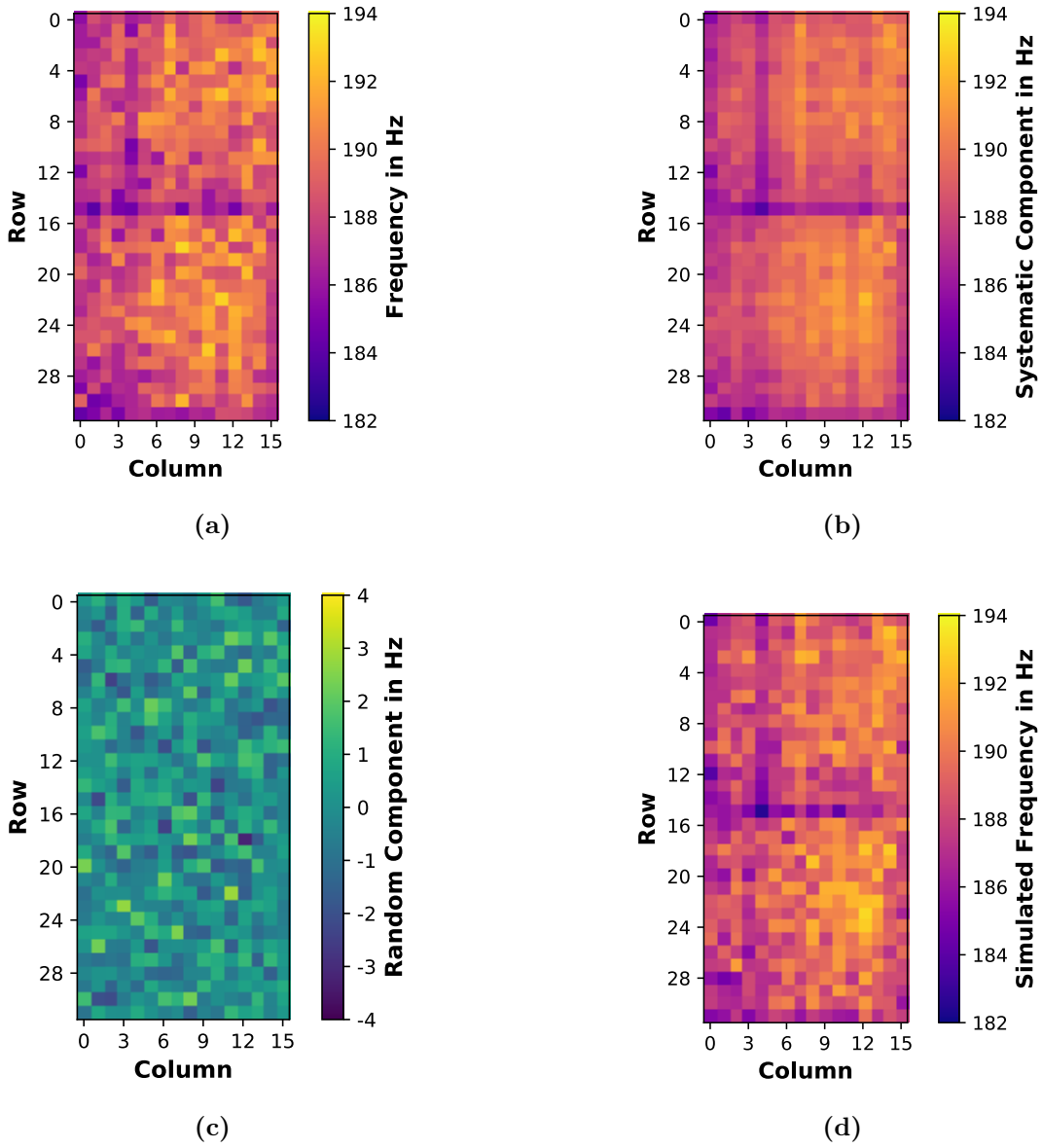


Figure 5.2: Spatial variation decomposition of the (a) oscillation frequencies z of 512 ring oscillators into (b) systematic components μ_Z and (c) random components U_Z. (d) shows the spatial distribution of a simulated dataset.

was rejected in 0.9 % of the test cases, which is below the expected type I error probability of 1 %. Hence, deviations from normality were found to be statistically insignificant. Furthermore, examination of the QQ-plots of randomly chosen frequency samples suggested normality of the data.

The distributional properties of $\log(\sigma_z^2)$ were analyzed based on the systematic

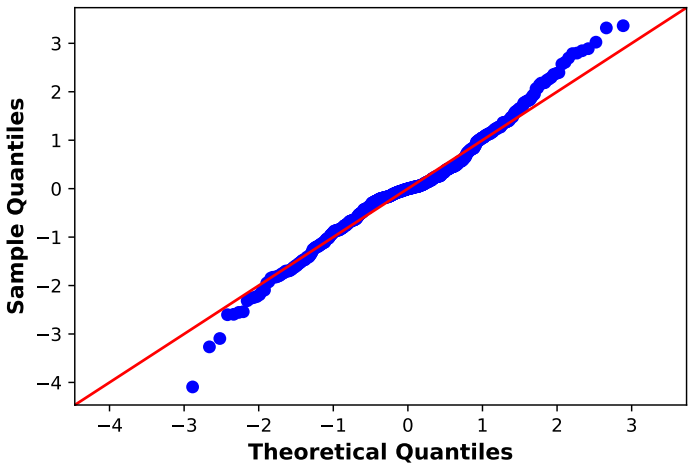


Figure 5.3: QQ-plot of the random components U_Z of 512 ring oscillator frequencies. In case of normally distributed data, the QQ-plot should follow the 45° line $y = x$.

and random variation components obtained by median polishing of the data. The spatial distribution of the logarithmized sample variance of a ring oscillator PUF instance before and after variation decomposition is shown in Fig. 5.4a to 5.4c.

The systematic variation components were identified with the mean vector of $\log(\sigma_z^2)$. The spatial covariance structure of the random effects was examined by means of statistical autocorrelation and normality tests, which were applied to the random effects of the logarithmized sample variances. Moran's I autocorrelation test statistic was calculated for a queen-type weight matrix with a maximum lag distance of up to 6 neighbors. Evaluated at a confidence level of 0.01, the results showed significant spatial correlation in 3.6 % of the performed tests. Since only a small number of tests turned out positive, correlation effects were neglected to simplify subsequent analysis of the noise-related intra-Hamming distance. The Anderson-Darling test was used to test the random effects for normality at a significance level of 0.01. The normality assumption was rejected in 3.1 % of the performed tests exceeding the chosen confidence level of 0.01. Therefore, the results indicate that a small fraction of sample distributions show statistically significant deviations from normality. However, analysis of the QQ-plots of the random effects showed no marked differences from a set of Gaussian distributed random samples, therefore the normal approximation was deemed a suitable fit to model the data. The covariance structure of the logarithmized random effects

was specified in terms of a diagonal matrix with a common variance representing the case of independently and identically distributed Gaussian random variables. Simulated datasets could reproduce the underlying spatial trend of the raw data as suggested by the comparison of Fig. 5.4a and 5.4d.

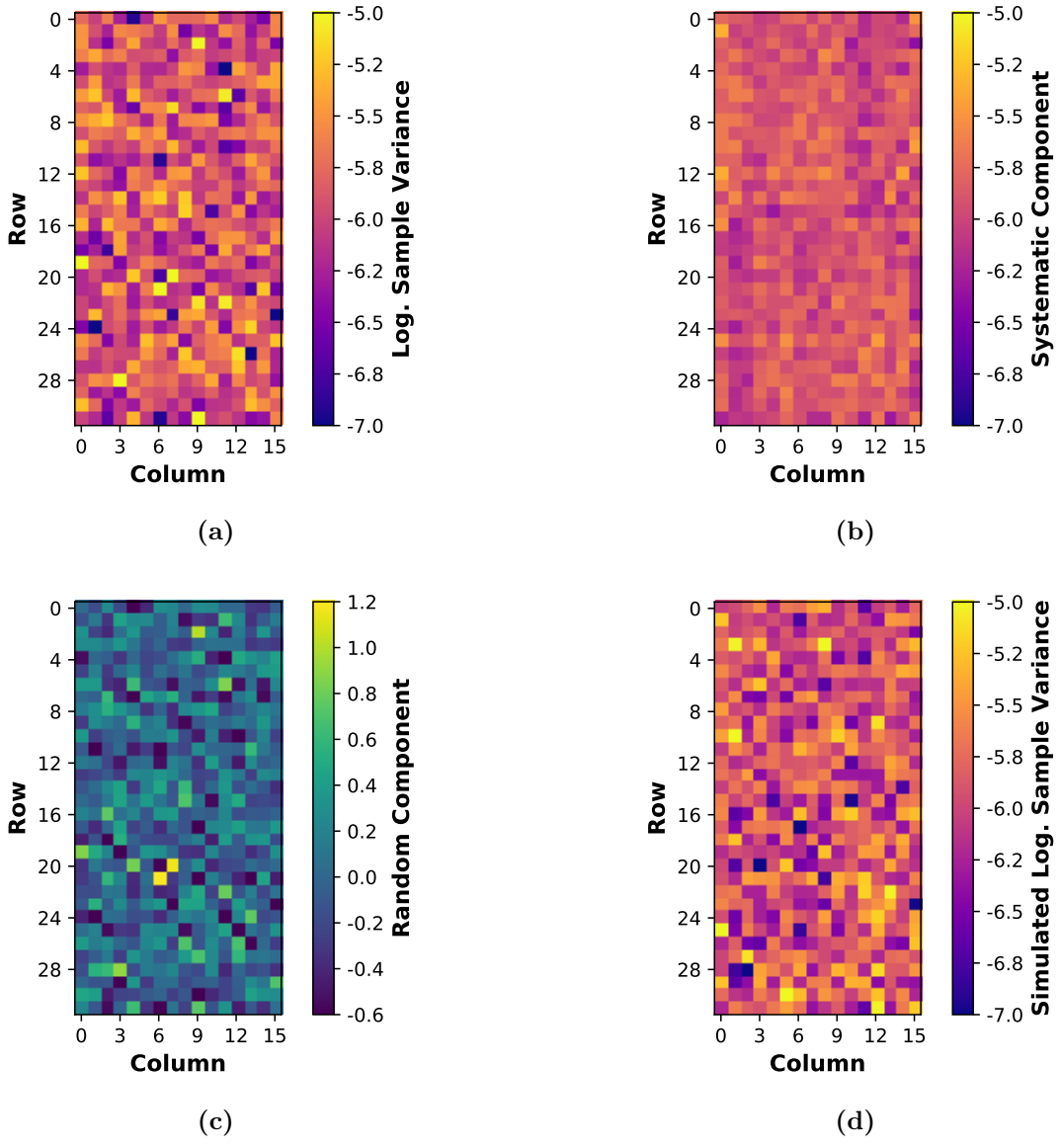
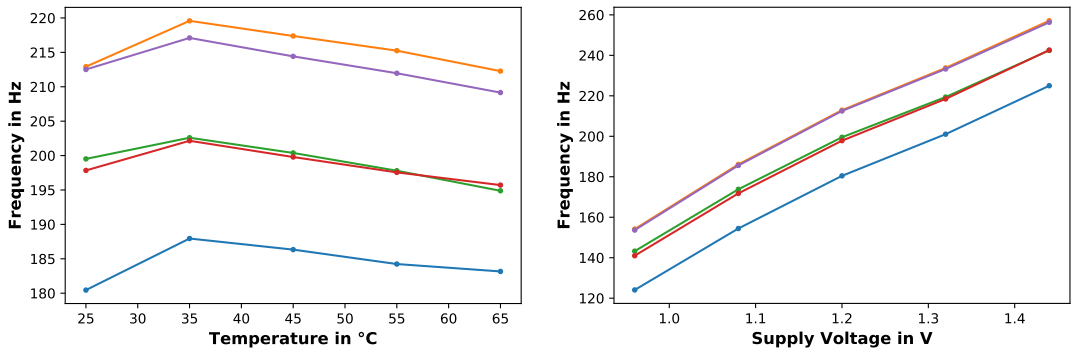


Figure 5.4: Spatial variation decomposition of the (a) logarithmized noise variance $\log \sigma_z^2$ of 512 ring oscillators into (b) systematic components $\mu_{\log \sigma_z^2}$ and (c) random components $U_{\log \sigma_z^2}$. (d) shows the spatial distribution of a simulated dataset.



(a) (b)

Figure 5.5: Dependence of the average oscillation frequency of 5 PUF instances on (a) the operating temperature and (b) the supply voltage level. The displayed frequencies were averaged over all 512 ring oscillators of a PUF instance.

5.3.3 Regression Parameters

In this example, multiple linear regression analysis was performed to include temperature and supply voltage effects on the stability of a PUF response. The form of the model equations can be readily determined by visual inspection of the data. As can be seen from Fig. 5.5a, the ring oscillator frequency increases linearly with increasing temperatures for $T \leqslant 35\,°\mathrm{C}$. For higher temperatures a linear decrease in the frequency was observed. Due to the change in the slope of the graph, the temperature behavior was modeled by a piecewise first order polynomial model [Mugg 08]. The dependence on the supply voltage level is shown in Fig. 5.5b. Taking into account deviations from the linear behavior for supply voltages larger than 1.2 V, the relation was approximated by a quadratic polynomial. The resulting linear regression model can be written as follows:

$$\tilde{z}_j^i = \beta_j^0 + \beta_j^1(T_i - T_0) + \beta_j^2\gamma^i + \beta_j^3(V_i - V_0) + \beta_j^4(V_i - V_0)^2 + \varepsilon_j^i\,, \tag{5.2}$$

where $\tilde{z}_j^i$, $i = 1, 2, ..., M$, $j = 1, 2, ..., 512$, denotes the i-th frequency sample of the j-th ring oscillator of a PUF instance taken at operating temperature T_i and supply voltage level V_i. The standard operating temperature T_0 and supply voltage level V_0 were set to $T_0 = 25\,°\mathrm{C}$ and $V_0 = 1.2\,\mathrm{V}$, respectively. The control variable γ_i is defined as

$$\gamma_i = (T_i - T_{bp})\alpha_i \tag{5.3}$$

with indicator value α_i

$$\alpha_i = \begin{cases} 1, & \text{if } T > T_{bp} \\ 0, & \text{otherwise.} \end{cases} \tag{5.4}$$

The inclusion of γ_i in the regression model causes a change in the slope of the temperature curve, if the operating temperature exceeds the break point temperature $T_{bp} = 35\,^{\circ}\mathrm{C}$. The corresponding design matrix X is of the form:

$$X = \begin{pmatrix} 1 & T_1 - T_0 & \gamma_1 & V_1 - V_0 & (V_1 - V_0)^2 \\ 1 & T_2 - T_0 & \gamma_2 & V_2 - V_0 & (V_2 - V_0)^2 \\ \vdots & \vdots & \vdots & & \\ 1 & T_M - T_0 & \gamma_M & V_M - V_0 & (V_M - V_0)^2 \end{pmatrix}. \tag{5.5}$$

The main objective of the analysis presented in the remainder of this section is to determine the distribution parameters $\mu_{\tilde{z}}$ and $\Sigma_{\tilde{z}}$ of the frequency samples obtained under different operating conditions and the covariance structure Σ_ε of the error terms. The distributional properties of the error terms were investigated in terms of the Anderson-Darling test for normality and the Kendall rank correlation test [Kend 38, Kend 45]. The two tests were applied to the detrended frequency data of the five PUF instances. Statistical hypothesis tests were performed at a confidence level of 0.01. The normality test yielded negative results for all PUF instances, i.e. the null hypothesis of normally distributed samples could not be rejected. Visual analysis of the QQ-plots of randomly chosen noise samples, suggested that the data could be well-fitted by zero-mean univariate normal distributions characterized by noise variances $\sigma^2_{\tilde{z}^i_j}$, $i = 1, 2, ..M$, $j = 1, 2, ..., 512$. Estimates of the noise variances were directly calculated from the detrended frequency samples.

The rank correlation coefficients were computed to test for spatiotemporal dependencies between the sets of repeatedly measured frequency samples. Kendall's test was performed for all unique pair combinations of the detrended frequency samples of each PUF instance. The null-hypothesis that the error terms are statistically independent was rejected for 1.5 % to 2.2 % of the sample pairs. According to the test outcomes, a small fraction of the error terms exhibited statistically significant correlation effects. Despite the positive test results, the error terms were considered approximately independent in order to facilitate computation of the intra-Hamming distance. Hence, the covariance matrix Σ_ε of a PUF instance

was approximated by a $(M \cdot 512) \times (M \cdot 512)$ diagonal matrix, i.e.

$$\Sigma_\varepsilon = \mathrm{diag}(\sigma^2_{\tilde{z}^1_1}, \sigma^2_{\tilde{z}^2_1}, ..., \sigma^2_{\tilde{z}^M_1}, ..., \sigma^2_{\tilde{z}^1_{512}}, \sigma^2_{\tilde{z}^2_{512}}, ..., \sigma^2_{\tilde{z}^M_{512}}) \,. \tag{5.6}$$

Under the assumption of jointly normal error terms, the distribution of ε was modeled by a zero-mean multivariate normal distribution (Eq. (4.70)) with covariance matrix Σ_ε as defined above.

Analysis of the mean vector $\mu_{\tilde{z}}$ and covariance matrix $\Sigma_{\tilde{z}}$ was based on the spatial distribution of the oscillation frequencies. Estimates of both quantities were obtained by applying median polishing to the denoised frequency samples measured at different operating temperatures. The systematic frequency components were taken as estimate of $\mu_{\tilde{z}}$ from which the components of $\mu_{\hat{\beta}}$ were calculated (Eq. (4.68)).

$\Sigma_{\tilde{z}}$ was determined from the random frequency components. The spatial covariance structure of the random effects was evaluated in terms of the Moran's I and the Anderson-Darling test statistics as described in the previous section. Again, the two tests were carried out at a significance level of 0.01. Computation of Moran's I for a queen-type contiguity definition, revealed statistically significant correlation effects for one of the five PUF instance at a lag distance of six. The results indicated that some of the random components were spatially correlated, typically at larger lag distances. Since only a small number of tests were positive, the random effects were considered approximately independent to simplify computation of the error probabilities in later sections.

The Anderson-Darling test did not reveal statistically significant deviations from normality. The respective QQ-plots were analyzed to verify normality of the data.

Motivated by the outcome of the two tests, the spatial covariance matrix $\Sigma_{\tilde{z}}|x$ at operating conditions $x = (T, V)$ was modeled by a diagonal matrix with variance $\sigma^2_{\tilde{z}}|x$, which was found to be dependent on the operating conditions x (Fig. 5.6). In order to specify the covariance matrix of the regression coefficient estimates, additional information about the correlation structure between the random effects at different operating conditions is required (Eq. (4.105), (4.106) and (4.107)). Since the random effects of each PUF instance were considered as repeated draws from the same zero-mean normal distribution, the pairwise correlation coefficients between the random effects at the various operating conditions

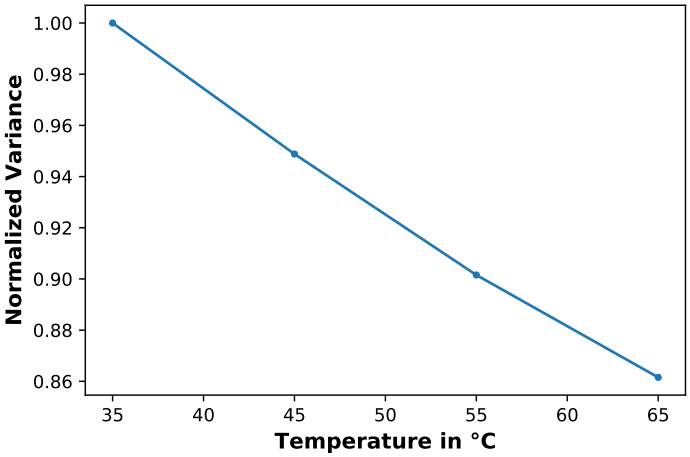


Figure 5.6: Dependence of the normalized variance of the random components of 512 ring oscillator frequencies on the operating temperature. The variance displayed at each temperature point was averaged over 5 PUF instances.

could be directly computed. The correlation coefficients calculated for the different operating temperatures were close to 1, i.e. the random effects were found to be highly correlated (Fig. 5.7).

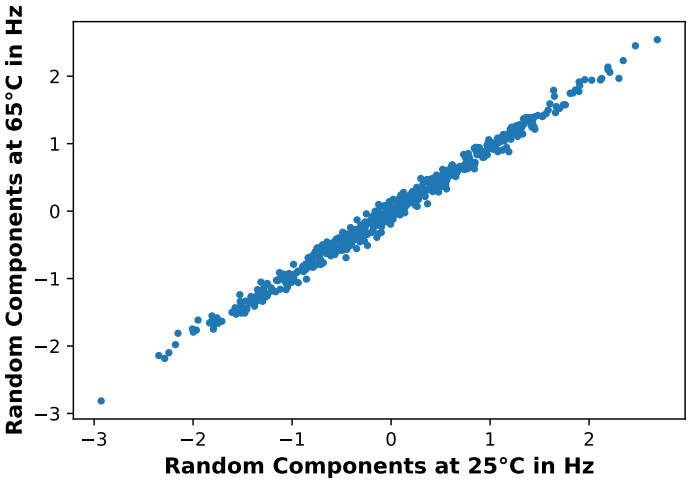


Figure 5.7: Scatter plot of the random components of 512 ring oscillator frequencies of a PUF instance determined at $T = 25\,°\mathrm{C}$ and $T = 65\,°\mathrm{C}$.

Therefore, the random effects $U_{\tilde{z}_j^i}$, $j = 1, 2, ..., 512$, at a given operating condition x_i could be expressed as scaled version of the random variables $U_{\tilde{z}_j^0}$ obtained

under reference conditions:

$$U_{\tilde{z}_j^i} = \theta_j^i U_{\tilde{z}_j^0} \,. \tag{5.7}$$

The scaling factor θ_j^i accounts for the fact that the variance of the random effects generally depends on the operating conditions of a PUF (Fig. 5.6). Since the random effects $U_{\tilde{z}_j^0}$, $j = 1, 2, ..., 512$, were modeled by a multivariate normal distribution, it follows that the components of the combined vector $U_{\tilde{z}} = [U_{\tilde{z}_1}^{\mathrm{T}}, ..., U_{\tilde{z}_{2N}}^{\mathrm{T}}]^{\mathrm{T}}$ are also jointly normal[5]. Furthermore, the random effects at different spatial locations were found to be uncorrelated, as

$$\mathrm{cov}(U_{\tilde{z}_j^r}, U_{\tilde{z}_k^s}) \propto \mathrm{cov}(U_{\tilde{z}_j^0}, U_{\tilde{z}_k^0}) \approx 0 \tag{5.8}$$

holds for all position indices $j, k = 1, 2, ..., 512$ with $j \neq k$. Hence, all cross-covariance matrices $\Sigma_{\tilde{z}_j, \tilde{z}_k}$ vanish and the elements of covariance matrix $[\Sigma_{\tilde{z}_j}]_{uv}$ are of the form

$$[\Sigma_{\tilde{z}_j}]_{uv} = \begin{cases} \sigma_{\tilde{z}}^2 | x_u, & \text{if } u = v \\ \sigma_{\tilde{z}} | x_u \cdot \sigma_{\tilde{z}} | x_v, & \text{otherwise.} \end{cases} \tag{5.9}$$

Since the cross-covariance matrices of the error terms and random effects are identical to the zero matrix, it follows from Eq. (4.77) that the covariance matrix of the regression coefficients $\Sigma_{\hat{\beta}}$ can be written in block diagonal form. The block diagonal entries are determined by Eq. (4.81). Given the estimates of $\mu_{\hat{\beta}}$ and $\Sigma_{\hat{\beta}}$, the distribution of the regression coefficients $\hat{\beta}$ is fully characterized (Eq. (4.72)).

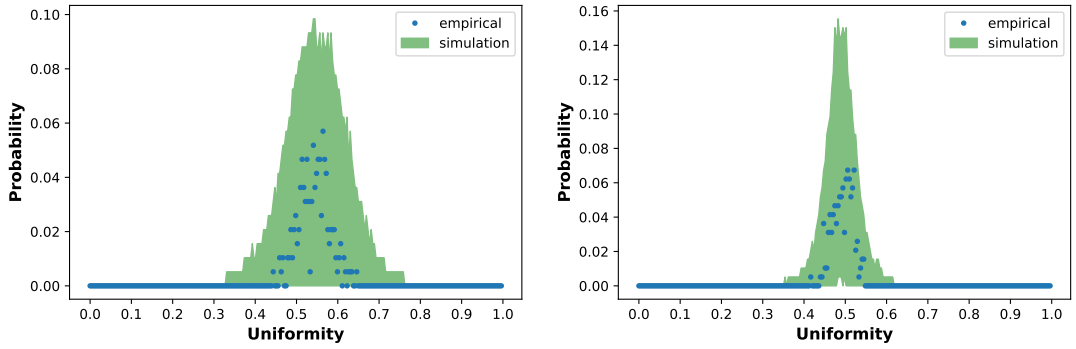
5.4 Quantization Scheme

The basic exclusive pairwise quantization scheme described in Sec. 4.2 was used to generate 256 response-bits for each of the 193 PUF instances. The response properties were examined for two different types of pair selection schemes. The first selection scheme forms pairs of ring oscillators located in two spatially adjacent columns of the 32×16 device array. For the second selection scheme, a fixed set of randomly chosen ring oscillators pairs was used.

[5]Since every linear combination of components of a jointly normal random vector follows a univariate normal distribution (Sec. 3.1.1), the linear relation stated in Eq. (5.7) is trivially fulfilled.

5.5 Uniformity

The distribution of the response uniformity of the 193 PUF instances was simulated by generating 100000 samples from the set of probability mass functions $\{P(X_U^u = k/N) : u = 1, 2, ..., 193\}$. Since the pair differences were found to be spatially independent, $P(X_U^u = k/N)$ was computed according to Eq. (4.24). The 100000 simulated realizations of the uniformity distribution and the empirically observed mass function are shown in Fig. 5.8. Due to the large number of samples, the graphs of the individual distributions are displayed as filled plot area. As can be seen from the figure, the simulation results reproduce the shape of the graph of the empirical distribution for both pair selection schemes, i.e. the empirical distribution is included in the area covered by the simulated graphs.



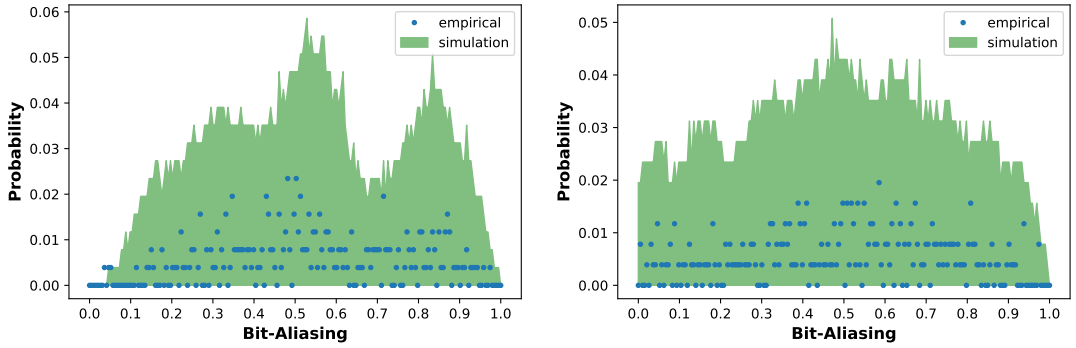
(a) **(b)**

Figure 5.8: Simulated and empirical distributions of the response uniformity computed for the (a) column-wise and (b) random pair selection scheme.

5.6 Bit-Aliasing

The distribution family of the bit-aliasing density was simulated by drawing random samples from the probability mass functions $P(X_{A,i} = k)$, $i = 1, 2, ..., 256$, derived for the case of independent response-bits (Eq. (4.27)). Since a small fraction of the PUF instances were tested positively for pairwise correlated random components, the results are only approximately valid. Figure 5.9 displays the results of 100000 simulated probability mass functions and the observed empirical mass function. Despite correlation effects between some of the PUF instances,

the model yields a plausible approximation of the underlying bit-aliasing distribution family for the two pair selection types. In particular, the large range of bit-aliasing values with non-zero probability mass is correctly replicated by the model.



(a) (b)

Figure 5.9: Simulated and empirical bit-aliasing distributions computed for the (a) column-wise and (b) random pair selection scheme.

5.7 Entropy

The distribution of the response entropy of the 193 PUF instances was determined for the following three use cases:

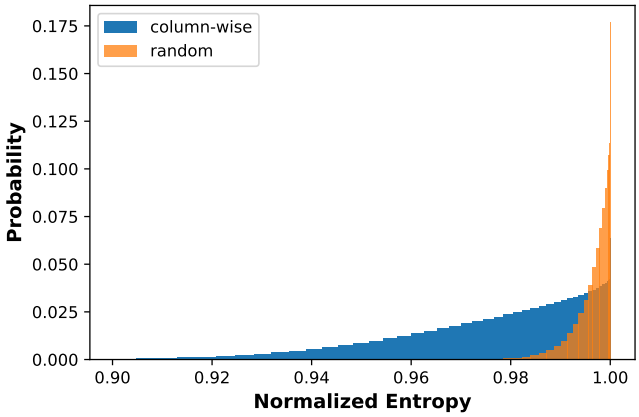
1. The response-bits of each PUF instance are affected by a common bias. The bias values are allowed to vary between different PUF instances.

2. Each response-bit of a set of PUF instances is affected by a common bias. The bias of different sets of response-bits might vary.

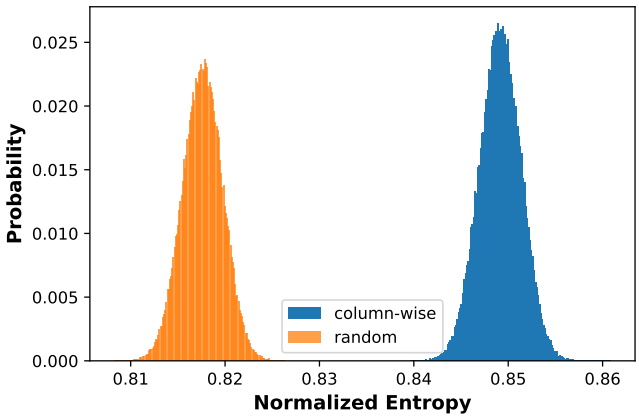
3. The bias of each response-bit of each PUF instance is individually determined by its respective single-bit probability (Eq. (4.22)).

The first two use cases correspond to the two most prevalent ways to estimate the response entropy from either the response uniformity or the bit-aliasing metric. In this instance, the response entropy of the PUF instances was calculated from the simulated samples used to determine the uniformity and bit-aliasing

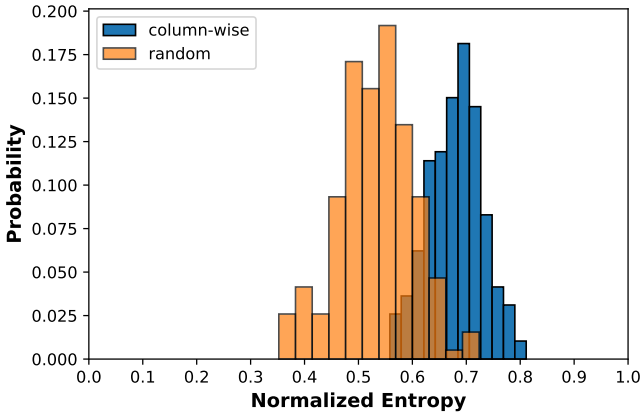
distributions, respectively. In the last use case, the entropy was computed according to Eq. (2.2). The results obtained for the column-wise and random pair selection scheme are displayed in Fig. 5.10. It is evident from Fig. 5.10a that the common bias assumption yielded the overall highest response entropy estimate for both pair selection schemes. The distributions are severely skewed towards maximal entropy values ($H > 0.9$) due to the high response uniformity (Fig. 5.8). Although both distributions are of similar shape, the distribution resulting from the column-wise pair selection scheme is heavier tailed than the one obtained for a randomized selection of pairs. The long tail can be explained by the comparably higher variability of the bias distributions (Fig. 5.8a) and the noticeable shift in the center of the probability mass towards uniformity values around 0.55. In contrast, the response entropy computed for a bit-dependent bias (Fig. 5.10b) was symmetrically distributed with significantly lower mean values close to 0.82 and 0.85, respectively. Since the bit-aliasing metric incorporates both the effect of systematic intra- and inter-die variation, lower entropy estimates were to be expected, especially since substantial probability mass is located at low and high bit-aliasing values indicating a bias of the response-bits towards a logical 0 and 1, respectively. Furthermore, it can be seen that the overall entropy is lower for the random pair selection scheme than for the column-wise selection, which was not the case for the results derived under the common bias assumption. In this case, the lower entropy can be attributed to the higher variability of the bit-aliasing distributions (compare Fig. 5.9a and 5.9b). Lastly, the entropy distributions of the 193 PUF instances computed according to Eq. (2.2) are plotted in Fig. 5.10c. On average, the column-wise and random pair selection scheme produced 0.68 and 0.53 bits of entropy per response-bit, respectively. The respective minimum entropy values were 0.56 and 0.35. Comparing these results to the ones obtained under the common and bitwise bias assumptions, it is apparent that both calculation methods grossly overestimate the response entropy. Out of the two empirical estimation procedures, the bitwise bias approximation yields a tighter upper bound on the response entropy, which is in agreement with the theoretical considerations presented in [Maes 13].



(a)



(b)

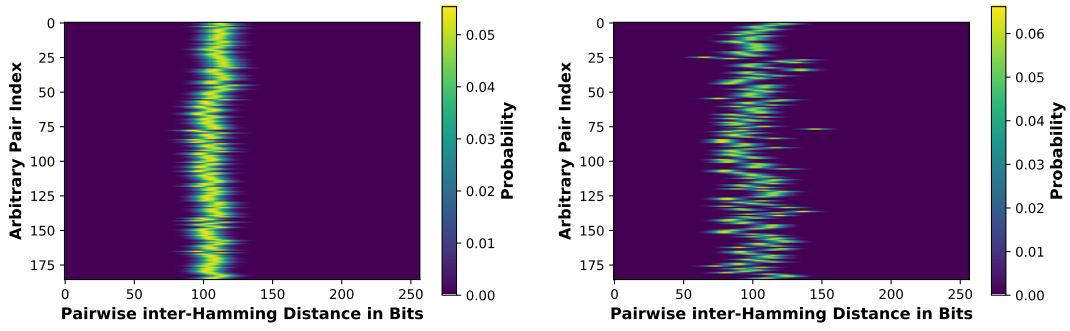


(c)

Figure 5.10: Normalized entropy distributions. (a) and (b) were obtained under the common and bitwise bias assumption, respectively. The canonical response entropy is displayed in (c).

5.8 Uniqueness

The uniqueness of the PUF responses was evaluated in terms of the pairwise inter-Hamming distances and the overall distribution of the inter-Hamming metric. The probability mass functions $P\left(X_{HD}^{u,v} = k\right)$ of the pairwise Hamming distance of two PUF instances u and v were computed according to Eq. (4.30). Due to correlations between the random components of some of the PUF instances, said equation is only approximately correct. To illustrate the differences in the shapes of the distribution functions obtained for the column-wise and random quantization scheme, 186 out of the $193 \cdot 192/2 = 18528$ unique pair distributions are shown in Fig. 5.11a and 5.11b, respectively. In the two figures, each row depicts the probability mass of a pairwise inter-Hamming distribution corresponding to different Hamming distances. As can be seen from the figures, both pair selection



(a) **(b)**

Figure 5.11: Pairwise inter-Hamming distance distributions computed for the (a) column-wise and (b) random pair selection scheme.

schemes result in overall low pair Hamming distances with comparable expected values around 100 bits. Furthermore, it is apparent that the pairwise Hamming distances are not identically distributed as indicated by the vertical misalignment of the color gradient of each row. Differences in the pairwise inter-Hamming distributions were found to be more pronounced in the case of the random pair selection scheme (Fig. 5.11b).

The distribution of the inter-Hamming distance was approximated by drawing random samples from each of the 18528 pairwise inter-Hamming distance probability densities. Figures 5.12a and 5.12b display the resulting distributions obtained for 100000 randomly sampled probability mass functions for each pair selec-

tion type. In addition, the measured realization of the empirical inter-Hamming distribution function and the respective binomial fit are plotted.

As is apparent from the two figures, the area spanned by the 100000 simulated distribution functions follows the shape of the empirical distribution, even if the pairwise inter-Hamming distances are not identically distributed (Fig. 5.12b). In the latter case, the proposed model yields a more realistic approximation of the inter-Hamming distance than a simple binomial distribution which completely fails to capture the characteristic shape of the empirical probability mass function. Furthermore, this example illustrates that if the pairwise Hamming distances are not identically distributed, then the resulting inter-Hamming distance metric can not be adequately described by a single distribution but rather by a family of possible distributions. Thus, characteristic quantities such as the mean and variance of the inter-Hamming distance depend on the random realization of one of the possible inter-Hamming distributions. In this example, considerable variation in the distribution of the mean inter-Hamming distances of the simulated samples was observed as displayed in Fig. 5.13. Therefore, evaluating the uniqueness of a PUF design in terms of the mean and variance of the average inter-Hamming distance is more conclusive than just reporting the empirical mean inter-Hamming distance determined from a single realization of measured PUF response data which provides no information about the uncertainty in the estimate.

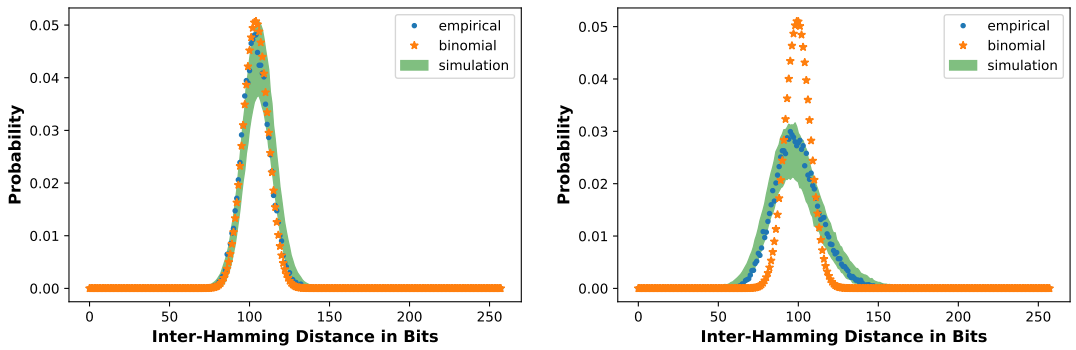


Figure 5.12: Simulated and empirical inter-Hamming distributions computed for the (a) column-wise and (b) random pair selection scheme.

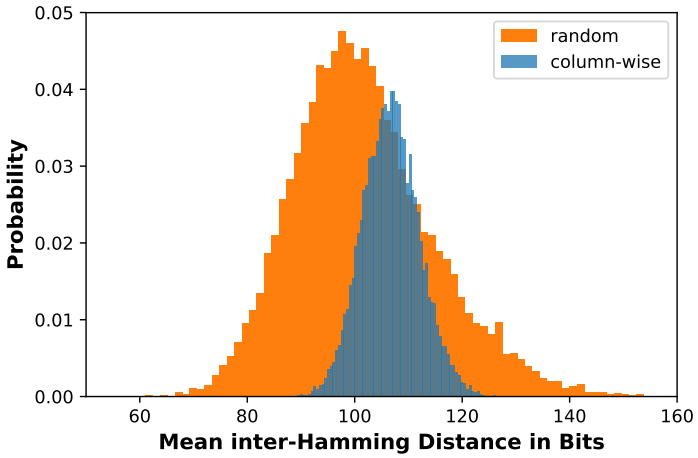


Figure 5.13: Distribution of the mean inter-Hamming distances computed for the column-wise and the random pair selection scheme.

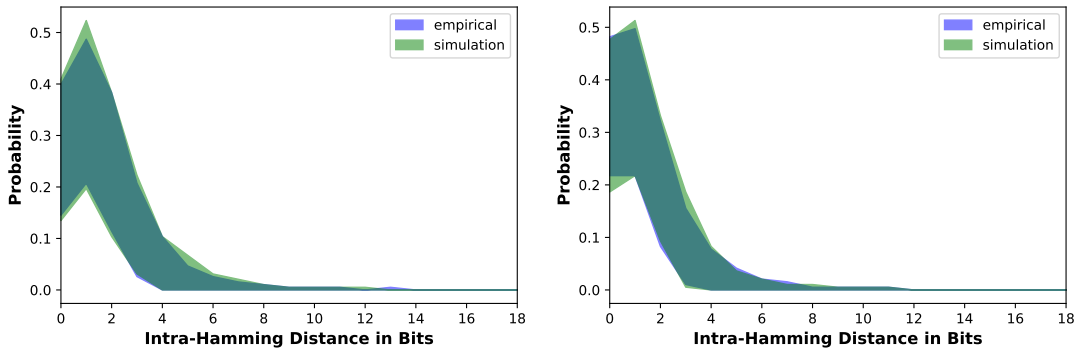
5.9 Reproducibility

5.9.1 Effect of Measurement Noise

The reproducibility of the PUF responses was evaluated in terms of the intra-Hamming distribution computed from the responses of 193 ring oscillator PUFs measured under standard operating conditions. The joint distribution of the oscillation frequencies and noise variances were approximated by the bitwise density functions $p(\delta\Delta s_i|\sigma^2_{noise,i})p(\Delta s_i)p(\sigma^2_{noise,i})$ (Sec. 4.8.1) with conditional and marginal distributions calculated according to Eq. (4.54) ,(4.53) and (4.51), respectively. The distribution of the noise-related Hamming distances was simulated from 100000 random realizations of the intra-Hamming distributions of the 193 PUF instances. The simulation results obtained for the column-wise and the random pair selection scheme are displayed in Fig. 5.14a and 5.14b, respectively. The ensemble of the 100000 mass functions are depicted as a single filled space to increase visibility of the overall trend of the distributions. Furthermore, 100000 randomly sampled probability mass functions computed from observed realizations of the oscillation frequency and noise variances are shown. The empirical intra-Hamming distributions were computed from the bit-error probabilities defined in Eq. (4.7).

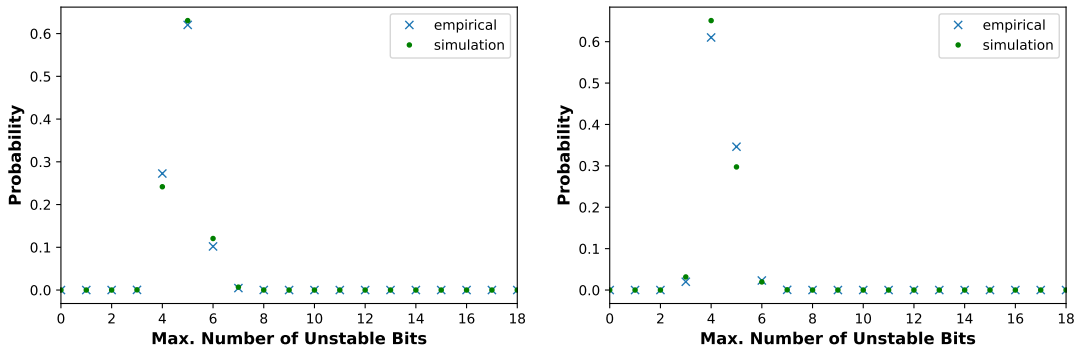
As can be seen from the two figures, the simulated distributions follow nearly the same trend as the empirical intra-Hamming distributions. The same behavior

can be observed for the distribution functions of the maximum number of bit-errors computed from the 99-percentile of the simulated and empirical Hamming probability functions. The bit-errors are almost identically distributed as shown in Fig. 5.15. In the case of the analyzed PUF design and pair selection schemes, the results suggest that process-induced variations of the oscillation frequencies and noise variances have negligible effects on the estimated bit-error distributions.



(a) (b)

Figure 5.14: Simulated and empirical intra-Hamming distributions computed for the (a) column-wise and (b) random pair selection scheme at standard operating temperature and supply voltage level.



(a) (b)

Figure 5.15: Maximum number of bit-errors computed for the (a) column-wise and (b) random pair selection scheme at standard operating temperature and supply voltage level.

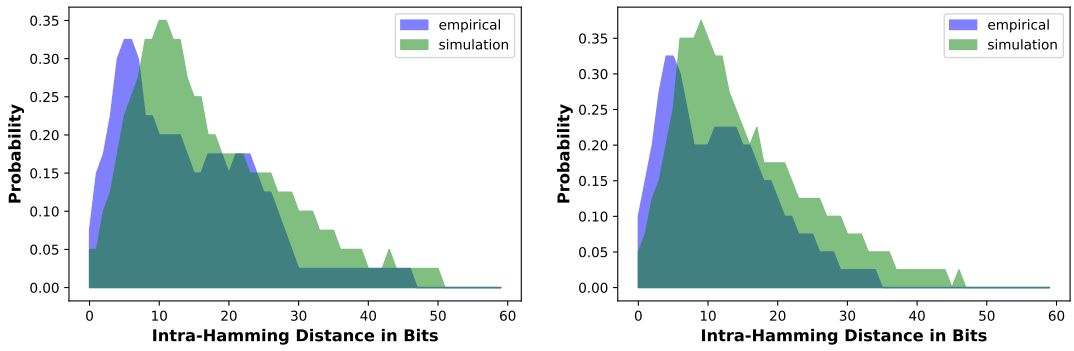
5.9.2 Effect of Temperature and Supply Voltage Variations

The stability of five PUF responses regenerated under different chip temperatures T_l and supply voltage levels V_l were evaluated in terms of the intra-Hamming distance. The cumulative intra-Hamming distance was computed for the operating conditions $T = 35\,°\mathrm{C}, 45\,°\mathrm{C}, 55\,°\mathrm{C}, 65\,°\mathrm{C}$ at a supply voltage level of $1.2\,\mathrm{V}$ and $V = 0.96\,\mathrm{V}, 1.08\,\mathrm{V}, 1.32\,\mathrm{V}, 1.44\,\mathrm{V}$ at $T = 25\,°\mathrm{C}$. The reference responses were measured under normal conditions $T_0 = 25\,°\mathrm{C}$ and $V_0 = 1.2\,\mathrm{V}$. In accordance with the approximations made in Sec. 5.3.3, simulation of the individual intra-Hamming distribution of the different PUF devices at the various operating conditions was accomplished by the multistage sampling procedure described in Sec. 4.7. In order to compute the single-bit error probabilities $P(X_{flip})$ (Eq. (4.40)), the required parameter values were randomly generated from the parameter distributions defined in Eq. (4.98) and (4.101), respectively. In total, 1000000 random intra-Hamming distances were simulated to approximate the underlying distribution family. The resulting intra-Hamming probability mass functions computed for the two pair selection schemes are displayed in Fig. 5.16a and 5.16b, respectively. In addition, 1000000 intra-Hamming distance distributions simulated from the empirically determined single-bit error probabilities $\tilde{P}(X_{flip})|x_k, x_0$ computed at operating conditions $x_l = (T_l, V_l)$, $l = 0, 1, ..., M$, are shown. The empirical bit-probabilities were determined according to the relation

$$\begin{aligned}\tilde{P}(X_{flip})|x_k, x_0 = {} & P(\Delta s(x_0) \leqslant 0)P(\Delta s(x_k) > 0) \\ & + P(\Delta s(x_0) > 0)P(\Delta s(x_k) \leqslant 0)\end{aligned} \tag{5.10}$$

with noise-related error probabilities $P(\Delta s(x_l) \leqslant 0) = 1 - P(\Delta s(x_l) > 0)$, $l = 0, 1, ..., M$, defined in Eq. (4.7). Both sets of graphs are plotted as filled areas spanned by the 1000000 simulated probability functions.

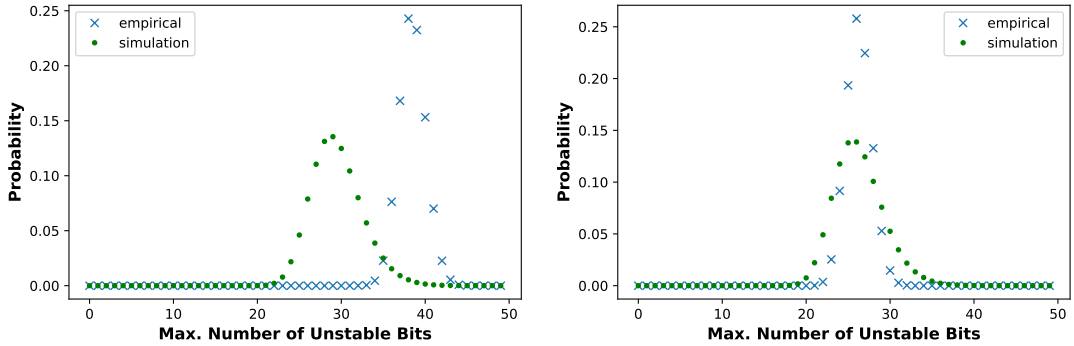
As can be seen from the figures, the shape of the empirical intra-Hamming distributions is correctly reproduced by the model. However, a higher variability of the simulated probability functions can be observed. The increased spread of the intra-Hamming distances accounts for processing-related variations in the temperature and voltage coefficients of the oscillation frequencies that are not included in the standard error-estimates which assume fixed parameter values.



(a) **(b)**

Figure 5.16: Simulated and empirical intra-Hamming distributions computed for the (a) column-wise and (b) random pair selection scheme at different operating temperatures and supply voltage levels.

The corresponding distribution of the maximum bit-errors computed from the 99-percentile of the simulated intra-Hamming distributions is shown in Fig. 5.17.



(a) **(b)**

Figure 5.17: Maximum number of bit-errors computed for the (a) column-wise and (b) random pair selection scheme at different operating temperatures and supply voltage levels.

As is apparent from the figure, taking into account random parameter variations can significantly alter the location and scale of the bit-error distribution. In case of the column-wise pair selection scheme, it was found that the simulated error mass function was centered at overall lower error values than the empirical distribution (Fig. 5.17a). The simulated distribution function spreads error

probability mass across different realizations of the oscillation frequencies and regression coefficients, effectively reducing the probability of large error numbers to occur. In contrast the empirical distribution overemphasizes the effect of measurement noise on the bit-stability of a given fixed realization of oscillation frequencies. The differences in the error mass function are more pronounced for frequency pairs with a small systematic mismatch, which are naturally more susceptible to measurement noise. Therefore, the discrepancy between the simulated and empirical distribution is less pronounced for the random pair selection scheme featuring a higher average mismatch than the column-wise formation of pairs. As shown in Fig. 5.17b, the random pair selection scheme resulted in a simulated density function with an expected value similar to the one of the empirical distribution but with a higher variance due to the uncertainty in the oscillation frequencies and regression coefficients.

5.10 Summary

The stochastic model framework proposed in the previous chapter was validated for experimental data of 193 ring oscillator PUFs implemented on commercial off-the-shelf FPGAs.

Application of the model showed that spatially varying systematic variation components need to be considered in order to compute realistic estimates of the response entropy and uniqueness. Most notably, entropy calculations based on the response uniformity and bit-aliasing metrics were found to overestimate the actual response entropy. Furthermore, it was shown that nonuniform processing variations can lead to a high prediction uncertainty of characteristic performance measures such as the response uniformity, bit-aliasing and inter-Hamming distance.

The intra-Hamming distance and maximum bit-error densities were simulated for Gaussian measurement noise and variable operating conditions including temperature variations and changes in the supply voltage. The analysis results showed that the variability of the oscillation frequencies and the noise variances had little impact on the bit-error estimates obtained under standard operating conditions. However, including process-induced variability of the temperature and voltage coefficients in the simulation significantly changed the probability

distribution of the expected number of bit-errors. Therefore, a fully probabilistic analysis is deemed essential to assert confidence in the estimates determined by empirical performance metrics such as the inter- and intra-Hamming distances computed from the experimental response data of a PUF.

Concerning the design and implementation of a PUF, the stochastic model approach provides vital information on the process-related variability of the performance metrics not captured by the empirical analysis methods. Compared to the latter type of methods, crucial design parameters such as the error correction capacity do not need to be inferred from possibly biased and unrepresentative sample characteristics of a small number of PUF instances. Instead, design parameters can be directly estimated from the distributional properties of a PUF concept realized in a particular processing technology. The statistical uncertainties in the performance estimates of a PUF and their impact on the implementation of a PUF design will be further investigated in the following chapters.

Chapter 6

PUF Design Development and Analysis

Developing a compact PUF design that generates reproducible random keys is a difficult task due to the physical nature of the electrical parameters from which the keys are extracted. As demonstrated by the experimental results shown in the previous sections, the performance of a PUF is highly susceptible to systematic device mismatch, measurement noise and environmental effects. Parameter variability attributed to the effect of deterministic process variations reduces the randomness and uniqueness of the PUF responses. Measurement noise and changes in the operating conditions of a PUF call for some form of error correction resulting in larger and more complex designs.

To ensure proper functionality, two design parameters are of key importance: (i) the number of semiconductor devices required to produce unique, full entropy keys[1] and (ii) the error correction capacity needed to ensure accurate key regeneration under varying operating conditions.

It is intuitively clear that the larger the systematic mismatch between primitives pairs is, the more pairs are needed to counterbalance the reduced bit-entropy. Similarly, a response mechanism susceptible to measurement noise requires a higher error correction capacity than a PUF designed to generate responses selectively from a set of stable bit-values. To make practical use of these generic statements a flexible mathematical model is needed which allows to quantify the

[1] A key is said to have full entropy when the entropy of the underlying response generation mechanism equals the key length.

effect of these factors on the design parameters of a PUF concept.

Building on the statistical analysis presented in the previous chapters, two models are proposed to investigate the relationship between the design parameters of a PUF and the latent characteristics of the manufacturing process used to implement the design in hardware. The models are based on a number of assumptions about the statistical properties of a PUF that reflect the amount of information available at different design stages of the development process. The assumption are closely tied to the information on the spatial statistics and stochastic properties of a PUF design fabricated in given processing technology.

The first model assumes the electrical parameters associated with security-related features of a PUF to be fixed and identical for each response-bit of a PUF instance. For example, the mean of the Gaussian random field model used to characterize these parameters is assumed to be spatially independent, hence all response-bits are identically distributed. As is apparent from the results shown in the previous section, a model which ignores spatial trends is overly simplistic and not suitable to accurately evaluate the performance of a PUF. In particular, this analysis approach is likely to overestimate the response entropy and uniqueness of a PUF design. Nonetheless, it can be useful to obtain crude estimates of the design parameters in the initial design phase, when access to experimental data is scarce. For example, at the very beginning of the design process, knowledge of the parameter distribution of a PUF might be limited to some average values for the deterministic and random matching properties of a PUF primitive obtained from computer aided process simulations.

The second model incorporates knowledge about the stochastic properties of the parameters that govern the distribution of the electrical characteristics of a PUF. Based on preliminary measurement data obtained from a small number of hardware prototypes, the distributional form of relevant parameters such as the deterministic and random mismatch of the PUF primitives can be readily determined. These results allow to estimate the design parameters for arbitrary stochastic process models by means of Monte Carlo simulations.

Application of the two models is demonstrated for the ring oscillator PUF design introduced in the previous chapter. Unless stated otherwise, it is assumed that the entropy, uniqueness and reproducibility of a PUF response can be adequately described at the individual bit-level. In this case, the Gaussian random

field framework presented in the last chapter can be reused.

6.1 Randomness

The response entropy is an important design parameter that determines the number of primitive pairs required to construct a PUF that outputs a full entropy key of a predetermined length. Lets assume that a cryptographic application requires an M-bit entropy key. If each primitive pair constituting a PUF instance produces $H = 1\,\text{bit}$ entropy, then M pairs are required to generate a full entropy key. In practice, the entropy H per primitive pair is likely to be lower, i.e. $0 < H < 1$, due to the effect of deterministic process variations. In this case, at least $N \geqslant \lceil 1/H \rceil$ primitive pairs are needed to generate a full bit of entropy. In the following sections, the relationship between N and the distribution parameters $(\mu_{\Delta S}, \sigma_{\Delta S})$ characterizing the effect of deterministic and random device mismatch is investigated.

6.1.1 Fixed Parameters

The entropy of a response-bit (Eq. (2.3)) is linked to the two process-related parameters $\mu_{\Delta S}$ and $\sigma_{\Delta S}$ through the single-bit probability $p_0 := P(\Delta s \leqslant 0)$ that a pair difference Δs produces a logical 0 (Eq. (4.22)):

$$p_0 = \frac{1}{2}\left(1 - \operatorname{erf}\left(\frac{1}{\sqrt{2}}\bar{\mu}_{\Delta S}\right)\right) \tag{6.1}$$

with $\bar{\mu}_{\Delta S}$ being defined as

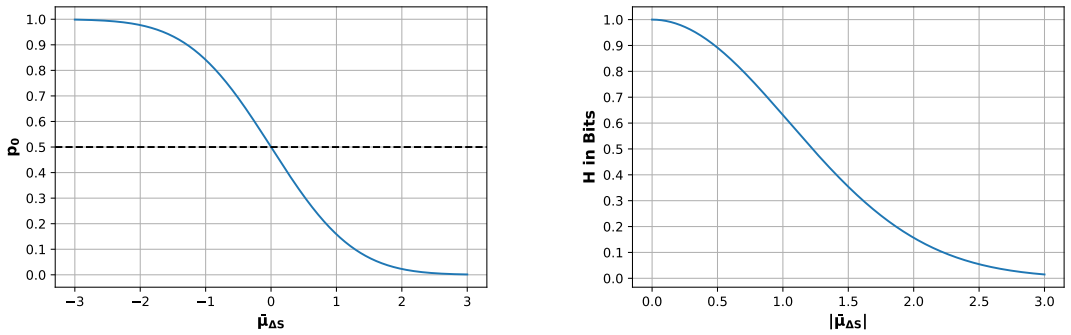
$$\bar{\mu}_{\Delta S} = \frac{\mu_{\Delta S}}{\sigma_{\Delta S}}\,. \tag{6.2}$$

The variable $\bar{\mu}_{\Delta S}$ can be interpreted as the relative bias of a bit-value towards either logical state. The dependence of p_0 on $\bar{\mu}_{\Delta S}$ is shown in Fig. 6.1a. In case $\bar{\mu}_{\Delta S} < 0$ holds, the zero state is favored. For a large negative bias $\bar{\mu}_{\Delta S} \lesssim -3$ the bit-value is a logical 0 with a probability close to 1. If the relative bias is positive, then the resulting bit-value tends towards a logical 1. A relative bias of $\bar{\mu}_{\Delta S} \gtrsim 3$ produces a logical 1 with almost absolute certainty. Consequently, only if the bias is zero, both logic states are equally likely.

By substituting the expression of p_0 into Eq. (2.3) the single-bit entropy can be written as a function of $\bar{\mu}_{\Delta S}$:

$$H\left(\bar{\mu}_{\Delta S}\right) = 1 - \left(1 - \operatorname{erf}\left(\frac{1}{\sqrt{2}}\bar{\mu}_{\Delta S}\right)\right) \log_2 \left(\sqrt{1 - \operatorname{erf}\left(\frac{1}{\sqrt{2}}\bar{\mu}_{\Delta S}\right)}\right) - \left(1 + \operatorname{erf}\left(\frac{1}{\sqrt{2}}\bar{\mu}_{\Delta S}\right)\right) \log_2 \left(\sqrt{1 + \operatorname{erf}\left(\frac{1}{\sqrt{2}}\bar{\mu}_{\Delta S}\right)}\right) . \tag{6.3}$$

From $\operatorname{erf}(-x) = -\operatorname{erf}(x)$, it follows that H is symmetric in $\bar{\mu}_{\Delta S}$. Furthermore, $H \leqslant 1\,\text{bit}$ holds for all $\bar{\mu}_{\Delta S}$. The maximum bit-entropy is only obtained in the complete absence of systematic process variations, i.e. $\bar{\mu}_{\Delta S} = 0$. The dependence of H on $|\bar{\mu}_{\Delta S}|$ is shown in Fig. 6.1b.



(a) **(b)**

Figure 6.1: Dependence of (a) the probability p_0 and (b) the single-bit entropy H on the relative bias $\bar{\mu}_{\Delta S}$.

In the initial design phase knowledge of $\mu_{\Delta S}$ and $\sigma_{\Delta S}$ might be limited to some average values characterizing the intra-die variability of the electrical parameters of a PUF. In this case, each response-bit is assigned a common entropy value. The functional dependence between the number of primitive device pairs N required to produce 1 bit of entropy and $|\bar{\mu}_{\Delta S}|$ is displayed in Fig. 6.2. As is apparent from the figure, the number of primitive pairs increases sublinearly for $|\bar{\mu}_{\Delta S}| < 1.23$. At a relative absolute bias of 1.23 which corresponds to $H = 0.5\,\text{bit}$ the number of required primitive pairs doubles. Even larger bias values result in a superlinear increase in N. Within the scope of the fixed parameter model, it can be concluded that the common relative bitwise bias of a PUF should not surpass a value of 1.23 in order to limit the area overhead and power consumption of a design.

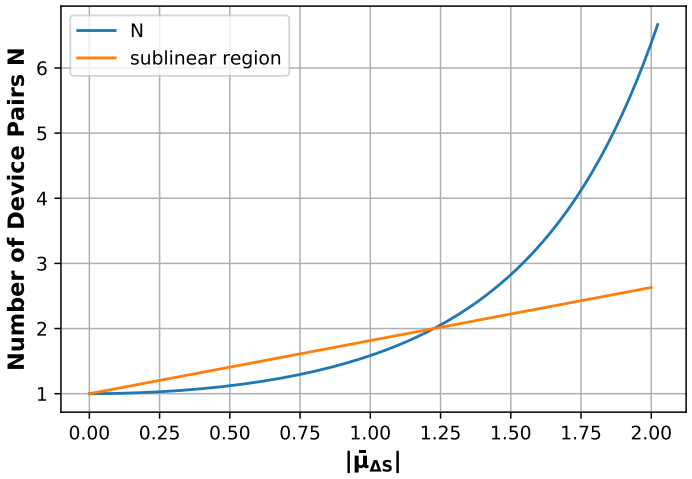


Figure 6.2: Number of device pairs N required to produce 1 bit of entropy dependent on the absolute relative bias $|\bar{\mu}_{\Delta S}|$.

6.1.2 Parameter Distribution

In practice, the relative bias is highly likely to differ between the response-bits due to spatial anisotropic processing effects (heat gradients, nonuniform layer deposition, etc.) which naturally occur throughout fabrication. For example, the distribution of the relative bias computed for 256 pair differences of 193 ring oscillator PUFs shows a wide range of possible values as can be seen in Fig. 6.3. According to the fixed parameter model described in the previous section, only 22 % of the primitive pairs would generate more than 0.5 bit of entropy. However, the normalized response entropy values $\tilde{H} = \sum_{i=1}^{256} H_i/256$ of the PUF instances were found to be much higher on average with a mean value of $\tilde{H} = 0.69$ (Fig. 5.10c). This example illustrates that if the number of bits suffering from a large bias is sufficiently small, then the overall entropy loss will be less severe than for a global bias of similar magnitude.

A more refined model needs to take the distributional properties of the relative bias into account. If $\bar{\mu}_{\Delta S}$ follows a parametric probability density function $p(\bar{\mu}_{\Delta S}|\theta)$ defined by a set of distribution parameters θ, then a functional relationship between θ and the number of primitives pairs N required to produce M bits of entropy can be established.

Model inference is based on Monte Carlo simulations. Random bias samples $\{\bar{\mu}_{\Delta S}^{jk} : j = 1, 2, ..., J\}$ are drawn from different members of the distribution family

conditional on a variety of distribution parameters θ_k, $k = 1, 2, ..., K$. From the bias samples the normalized entropy $h = H/M$ and the normalized number of primitive pairs $n = N/M$ are computed[2]. The dependence on $\bar{\mu}_{\Delta S}$ is determined by standard regression or interpolation techniques. The resulting model can be used to formulate design criteria that limit the possible parameter space of θ to values whose corresponding bias distributions are compliant with resource constraints imposed by the target application of the PUF. The steps involved in the model building process are illustrated in the following example.

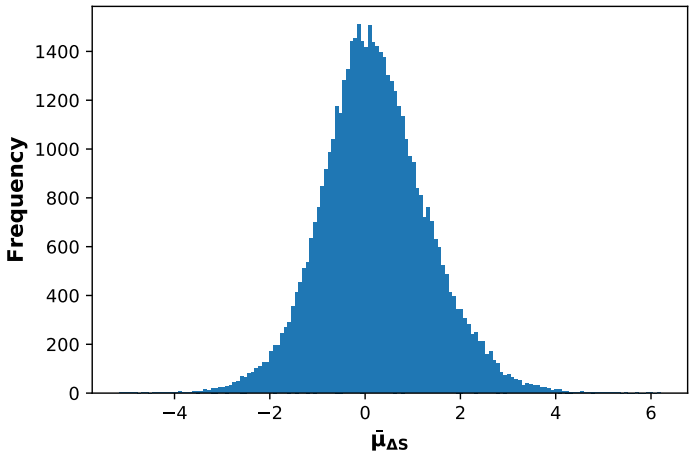


Figure 6.3: Relative bias distribution computed for 256 × 193 ring oscillator pairs.

Example

Let's assume a PUF-based random source is designed to output $M = 256$ bit of entropy per response. Considering the size of the primitives used to build the PUF the implementation overhead might limit the average number of primitive pairs to at most 2.5 pairs per bit entropy, i.e. $n_{max} = 2.5$. Motivated by the bias distribution of the ring oscillator PUF (Fig. 6.3), $\bar{\mu}_{\Delta S}$ is assumed to be normally distributed with distribution parameter $\theta = (\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}})$ corresponding to the distribution mean and standard deviation, respectively. The aim is to determine bounds on θ that warrant design-conform bias distributions.

[2]Here, it is assumed that M is sufficiently large such that small sample size effects can be neglected.

First off, the parameter space $(\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}}) \in \mathbb{R} \times \mathbb{R}_{>0}$ is constrained to a subset of distribution parameter values relevant to the PUF design. Solving the equation

$$\left[H\left(|\bar{\mu}_{\Delta S}|\right)\right]^{-1} - n_{max} = 0 \tag{6.4}$$

for $|\bar{\mu}_{\Delta S}|$ yields a maximum relative bias value of $\bar{\mu}_{\Delta S}^{max} \approx 1.41$. Based on $\bar{\mu}_{\Delta S}^{max}$ the bounds of $\mu_{\bar{\mu}_{\Delta S}}$ and $\sigma_{\bar{\mu}_{\Delta S}}$ are set to $\mu_{\bar{\mu}_{\Delta S}}^{max} = \sigma_{\bar{\mu}_{\Delta S}}^{max} = 2\bar{\mu}_{\Delta S}^{max}$ such that sufficient probability mass of the sampled probability distributions will be placed within and close to the bounds of the confining parameter region determined by n_{max}. Furthermore, $\mu_{\bar{\mu}_{\Delta S}}$ can be limited to positive values since H only depends on the absolute value of $\bar{\mu}_{\Delta S}$. For the model building process, a set of 200 distribution parameters $\{\theta_i = (\mu_{\bar{\mu}_{\Delta S}}^i, \sigma_{\bar{\mu}_{\Delta S}}^i) : i = 1, 2, ..., K\}$ was sampled from the resulting parameter subspace. Each distribution parameter θ_i defines a normal distribution

$$N\left(\bar{\mu}_{\Delta S} | \mu_{\bar{\mu}_{\Delta S}}^i, (\sigma_{\bar{\mu}_{\Delta S}}^i)^2\right) \tag{6.5}$$

from which each of which 256×10000 random bias samples were generated. The bias samples were used to compute the normalized entropy h and number of primitive pairs n.

For a given distribution parameter θ_i, the maximum number of primitive pairs was estimated from the corresponding sample distribution of n_i conditional on θ_i. Since the maximum of the sample distribution generally depends on the sample size and the sampling method, n_{max} was estimated from the α-percentile $n_{perc}(\alpha)$, $\alpha \in [0, 100]$, of the sampling distribution. In this example the 99-th percentile $n_{perc}(99\,\%)$ was chosen, i.e. $n \leqslant n_{perc}(99\,\%)$ holds for 99 % of the observed samples.

In a last step, multiple linear regression analysis was employed to model the dependence of $n_{perc}(0.99)$ on $(|\mu_{\bar{\mu}_{\Delta S}}|, \sigma_{\bar{\mu}_{\Delta S}})$. In this instance, a second order polynomial function

$$f(\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}}) = 1 + \beta_{12}|\mu_{\bar{\mu}_{\Delta S}}|\sigma_{\bar{\mu}_{\Delta S}} + \beta_{01}\mu_{\bar{\mu}_{\Delta S}}^2 + \beta_{02}\sigma_{\bar{\mu}_{\Delta S}}^2 \tag{6.6}$$

provided a good fit to the percentile data as indicated by an adjusted R-squared value of 0.885. The graph of the fitted function with $\beta_{12} = -2.0288$, $\beta_{01} = 1.7397$ and $\beta_{02} = 0.8566$ is shown in Fig. 6.4. In addition, three contour lines corresponding to $n_{perc}(0.99) = 1.25, 2.0, 2.5$ pairs per bit entropy are shown. All distribution parameters enclosed by the contour line with $n_{perc}(0.99) = 2.5$ result

in bias parameter distributions that satisfy the design criterion $n_{perc}(0.99) \leqslant n_{max} = 2.5$. It shall be highlighted that this result is independent of the device type used as PUF primitive.

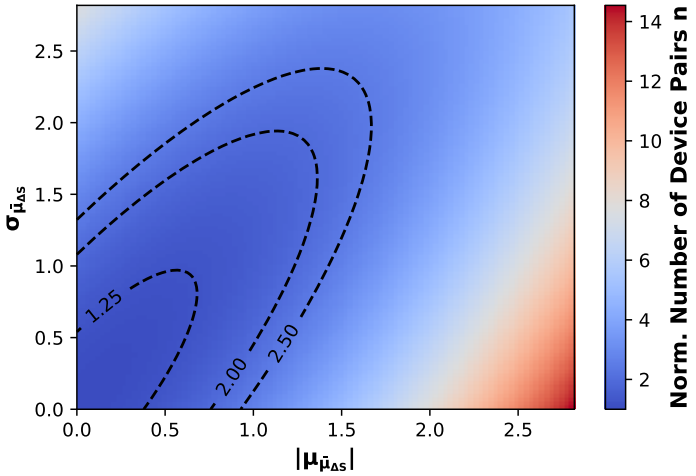


Figure 6.4: Normalized number of device pairs n required to produce 1 bit of entropy dependent on the distribution parameters $(|\mu_{\bar{\mu}_{\Delta S}}|, \sigma_{\bar{\mu}_{\Delta S}})$ of the relative bias distribution of a PUF design.

6.2 Uniqueness

Due to the effect of systematic process variations bit-patterns of two different PUF keys might share either a smaller or larger number of unique response-bits than expected in the case of randomly generated keys. If any two N-bit long PUF responses differ by only k unique response-bits, then at most $I_{unique} = \binom{N}{k}$ PUF instances can be uniquely identified based on their keys. Similarly, if all but k response-bits were to be unique, then the number of distinguishable PUF instances would also be limited to $\binom{N}{N-k} = \binom{N}{k}$. In both cases, additional response-bits are required to differentiate a larger number of PUFs, hence increasing the size of the design. In the following sections, generic methods are developed to quantify the effect of process variations on the minimum number of unique keys that can be extracted from a given PUF design.

6.2.1 Fixed Parameters

In order to gain an intuitive understanding of how systematic process variations affect the uniqueness of a PUF response, it is assumed that all bits of a PUF instance are subjected to a common bias that might differ between PUF instances. In this case, the probability $p_{unique}(\bar{\mu}^u_{\Delta S}, \bar{\mu}^v_{\Delta S})$ (Eq. (4.29)) that any of the response-bits of two PUF instances u and v differ from one another can be expressed in terms of the common relative biases $\bar{\mu}^u_{\Delta S}$ and $\bar{\mu}^v_{\Delta S}$ of the two PUF instances:

$$\begin{aligned} p_{unique}(\bar{\mu}^u_{\Delta S}, \bar{\mu}^v_{\Delta S}) &= p^u_0(\bar{\mu}^u_{\Delta S}) \cdot (1 - p^v_0(\bar{\mu}^v_{\Delta S})) + (1 - p^u_0(\bar{\mu}^u_{\Delta S})) \cdot p^v_0(\bar{\mu}^u_{\Delta S}) \\ &= \frac{1}{2}\left(1 - \operatorname{erf}\left(\frac{1}{\sqrt{2}}\bar{\mu}^u_{\Delta S}\right) \operatorname{erf}\left(\frac{1}{\sqrt{2}}\bar{\mu}^v_{\Delta S}\right)\right) . \end{aligned} \tag{6.7}$$

The dependence of p_{unique} on the relative biases $\bar{\mu}^u_{\Delta S}$ and $\bar{\mu}^v_{\Delta S}$ is depicted in Fig. 6.5. As can be seen from the figure, the events that the bits are identical

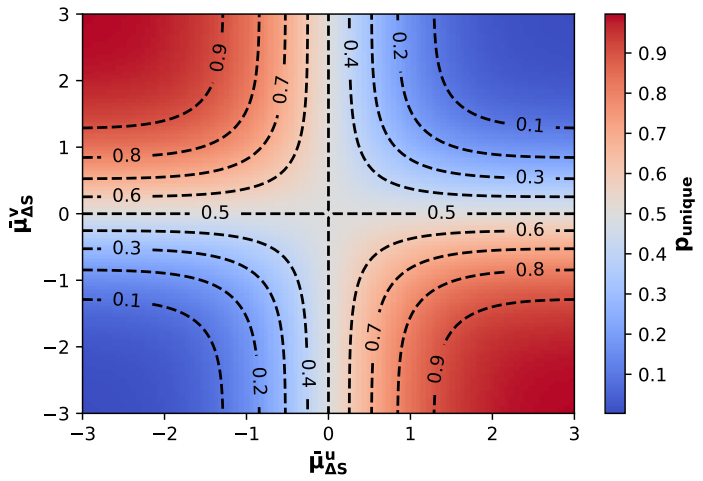


Figure 6.5: Probability p_{unique} dependent on the relative bias values $\bar{\mu}^u_{\Delta S}$ and $\bar{\mu}^v_{\Delta S}$ associated with two PUF instances denoted by superscripts u and v, respectively.

and that they differ from each other are equally likely, if either of the two PUF instances is bias-free. If the two bias values have the same sign, then the probability distributions of the corresponding pair differences are shifted towards the same logical state for both PUF instances. As a result, the probability that the bits are identical is higher than by pure chance. In contrast, if the bias values

differ in their sign, then the distributions are shifted towards opposite logical states. Consequently, the bit-values are more likely to be different.

In order to determine the number of pair primitives required to uniquely distinguish a number of PUF instances for different bias values, the minimum number of bits n_{unique} that can be used to identify a PUF instance needs to be computed. For a given response length N and probability p_{unique}, the number of unique response-bits k_{unique} follows a binomial distribution $B(k|N, p_{unique})$. Here, n_{unique} is determined from the quantiles taken at the distribution tails. Let $n_{perc}(\alpha)$ be the α-percentile of the distribution, then n_{unique} is defined as:

$$n_{unique}|N, p_{unique} = \min(n_{perc}(\alpha)|N, p_{unique}, N - n_{perc}(100\,\% - \alpha)|N, p_{unique})\,, \tag{6.8}$$

where α is set to a small value such as 1 %. If $n_{unique} = n_{perc}(1\,\%)$ holds, then the number of unique response-bits of two PUF instances will be at least n_{unique} with a probability of 99 %. The other possible bound $N - n_{perc}(100\,\% - \alpha)$ takes into account that the effective number of bits usable to differentiate a set of PUF keys decreases for an increasing number of unique response-bits as explained at the beginning of this section. For given values of n_{unique} and N, the minimum number of uniquely identifiable PUF instances I_{unique} is determined by the binomial coefficient

$$I_{unique}|N, p_{unique} = \binom{N}{n_{unique}}\,. \tag{6.9}$$

The dependence of I_{unique} on N and p_{unique} is shown in Fig. 6.6. As can be seen from the figure, the largest value of I_{unique} is always obtained for $p_{unique} = 0.5$, independent of the key length. Deviations from $p_{unique} = 0.5$ due to biased pairs of response-bits result in an increased key length to achieve the same value of I_{unique} as obtained under bias-free conditions. Furthermore, the number of additional response-bits required to compensate bias effects significantly increases for larger values of I_{unique} as indicated by the narrowing contour lines shown in the figure.

Based on the preceding analysis, the number of primitive pairs N required to identify a minimum number of I_{unique} PUF instances can be easily computed for different bias values. For example, one might want to determine N such that n_{unique} is equal to the maximum number of unique response-bits n_{unique}^{max} obtainable from a full entropy key of length M_{full}. Given a set of relative bias values, the

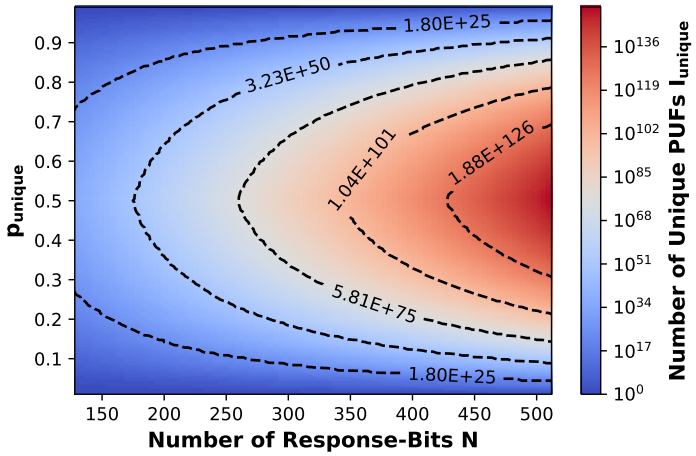


Figure 6.6: Uniquely identifiable PUF instances I_{unique} dependent on the number of response-bits N and probability p_{unique}.

corresponding probabilities p_{unique} are calculated according to Eq. (6.7). Then, these probability values are used to compute n_{unique} as defined in Eq. (6.8). Since the percentiles of the binomial distribution do not scale linearly in the response length, n_{unique} is determined for different response lengths N_i in the range of $M_{full} \leqslant N_i < N_{max}$. The maximum response length N_{max} is limited by design constraints regarding chip area and power consumption. Lastly, for each relative bias pair the smallest response length for which $n_{unique} \geqslant n_{unique}^{max}$ holds is selected.

As an example, the distribution obtained for $\bar{\mu}_{\Delta S}^{u}, \bar{\mu}_{\Delta S}^{v} \in [-3, 3]$, $M_{full} = 256\,\text{bit}$ and $M_{max} = 4096\,\text{bit}$ is shown in Fig. 6.7. Note that the maximum values of N are capped at 4096 bit. From the figure, it is apparent that the minimum response length requires one of the two relative bias values to be exactly zero. If both bias value are non-zero, further response-bits are needed to compensate for the effect of systematic process variations.

6.2.2 Parameter Distribution

As stated in Sec. 6.1.2, the assumption of a common relative bias generally does not hold in practice. To incorporate the effect of spatially dependent bias values, it is assumed that the relative bias distributions of all PUF instances can be modeled by a common distribution family. Since the number of unique response-bits depends on the bias distributions of two PUFs instances, the possibly differing distribution parameters of both instances need to be considered.

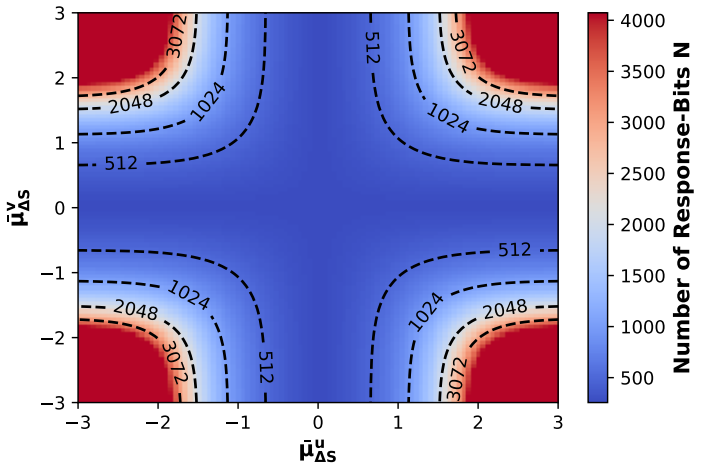


Figure 6.7: Number of response-bits N required to uniquely identify the same number of PUF instances as for 256-bit full-entropy keys dependent on the relative bias values $\bar{\mu}^{u}_{\Delta S}$ and $\bar{\mu}^{v}_{\Delta S}$ associated with two PUF instances denoted by superscripts u and v, respectively.

The relationship between the quantities is modeled using a nested sampling approach similar to the one described in Sec. 6.1.2. For I distribution parameter tuples (θ^1_i, θ^2_i), $i = 1, 2, ..., I$, J sets of random bias samples $(\bar{\mu}^{1ij}_{\Delta S}, \bar{\mu}^{2ij}_{\Delta S})$, $j = 1, 2, ..., J$, are simulated from the corresponding relative bias distributions $p(\bar{\mu}^1_{\Delta S}|\theta^1_i)$ and $p(\bar{\mu}^2_{\Delta S}|\theta^2_i)$. The bias samples are used to compute the N single-bit probabilities $p^{ij}_{unique,n}$, $n = 1, 2, ..., N$, according to Eq. (6.7). Given the parameter vector $p^{ij} = (p^{ij}_{unique,1}, p^{ij}_{unique,2}, ..., p^{ij}_{unique,N})$, the minimum number of unique response-bits n^{ij}_{unique} is calculated from the percentiles of the Poisson binomial distribution $GB(p^{ij})$ analogous to the procedure explained in Sec. 6.2.1. The estimate of n^{i}_{unique} for the i-th distribution parameter tuple is determined from the α-percentile of the J sampled n^{ij}_{unique} values. To simulate worst-case conditions, α is typically set to values in the lower percentage range. For example, $\alpha \leqslant 5\,\%$ means that the estimated minimum number of bits is larger than n^{i}_{unique} with a probability of $100\,\% - \alpha = 95\,\%$. Four-dimensional interpolation techniques can be employed to predict n_{unique} for out-of-sample distribution parameters.

Example

In practice, one might be interested in the distribution parameter subspace for which at least I_{unique} PUF keys can be uniquely identified given a specific re-

sponse length. Continuing the example from the last section, the ring oscillator PUF is designed to produce 256-bit keys that uniquely distinguish a minimum number of $I_{unique} = 10^{60}$ PUF instances. According to Eq. (6.9), at least 62 unique response-bits n^{min}_{unique} are needed[3]. To fulfill this requirement, constraints have to be placed on the allowed value range of the bias distribution parameters $\theta^1 = (\mu^1_{\bar{\mu}_{\Delta S}}, \sigma^1_{\bar{\mu}_{\Delta S}})$ and $\theta^1 = (\mu^2_{\bar{\mu}_{\Delta S}}, \sigma^2_{\bar{\mu}_{\Delta S}})$ of any two PUF instances. Starting point for the following analysis is the distribution parameter subspace $\{(\mu^1_{\bar{\mu}_{\Delta S}}, \sigma^1_{\bar{\mu}_{\Delta S}}, \mu^2_{\bar{\mu}_{\Delta S}}, \sigma^2_{\bar{\mu}_{\Delta S}}) : -1.8 \leqslant \mu^1_{\bar{\mu}_{\Delta S}}, \mu^2_{\bar{\mu}_{\Delta S}} \leqslant 1.8, 0 < \sigma^1_{\bar{\mu}_{\Delta S}}, \sigma^2_{\bar{\mu}_{\Delta S}} \leqslant 2.5\}$. The parameter bounds were chosen to include the parameter subspace Ω_{random} which results in bias distributions compliant with the size limitation of 2.5 primitive pairs per bit response-entropy (Fig. 6.4). From this parameter space, 400 random parameter tuples $(\mu^{1i}_{\bar{\mu}_{\Delta S}}, \sigma^{1i}_{\bar{\mu}_{\Delta S}}, \mu^{2i}_{\bar{\mu}_{\Delta S}}, \sigma^{2i}_{\bar{\mu}_{\Delta S}})$, $i = 1, 2, ..., 400$, were drawn. For each parameter tuple, 256×10000 bias sample pairs $(\bar{\mu}^{1ij}_{\Delta S}, \bar{\mu}^{2ij}_{\Delta S})$, $j = 1, 2, ..., 10000$, were generated. Given the bias samples, the bit-values n^i_{unique} were calculated from the 1 %-percentile of the sample distributions as explained in the previous subsection. Linear interpolation of the initial sampling points $\{((\mu^{1i}_{\bar{\mu}_{\Delta S}}, \sigma^{1i}_{\bar{\mu}_{\Delta S}}, \mu^{2i}_{\bar{\mu}_{\Delta S}}, \sigma^{2i}_{\bar{\mu}_{\Delta S}}), n^i_{unique}) : i = 1, 2, ..., 400\}$ was used to compute n_{unique} at evenly spaced grid positions within the parameter subspace Ω_{random}. In a next step, the distribution parameter values of the grid points for which $n_{unique} \geqslant n^{min}_{unique}$ holds were determined. The resulting distribution parameter space is shown in Fig. 6.8. To aid visualization, the four-dimensional parameter space is projected into three-dimensional subspaces with coordinate axes $\mu^2_{\bar{\mu}_{\Delta S}}, \sigma^2_{\bar{\mu}_{\Delta S}}$ and $\sigma^1_{\bar{\mu}_{\Delta S}}$ for different values of $\mu^1_{\bar{\mu}_{\Delta S}}$. The projections are displayed in form of heatmaps with $\mu^2_{\bar{\mu}_{\Delta S}}$ and $\sigma^2_{\bar{\mu}_{\Delta S}}$ corresponding to the x- and y-axis, respectively. $\sigma^1_{\bar{\mu}_{\Delta S}}$ is represented graphically by the color gradient of the heatmap. As can be seen from Fig. 6.8a and 6.8b, the distribution parameter space of one of the two PUF instances is only limited by the size constraints on the key entropy, if the bias distribution of the other one is centered close to zero, i.e. the response-bits of the PUF instance are nearly bias-free. As the mean parameters $|\mu^1_{\bar{\mu}_{\Delta S}}|$ and $|\mu^2_{\bar{\mu}_{\Delta S}}|$ increase (Fig. 6.8c and 6.8d), the respective bias distributions are shifted towards higher bias values. Consequently, the variance of the bias distributions must increase, such that sufficient probability mass is placed at smaller bias val-

[3]To simplify the analysis it is assumed that the PUF responses are fully reproducible, e.g. after application of an error correction procedure.

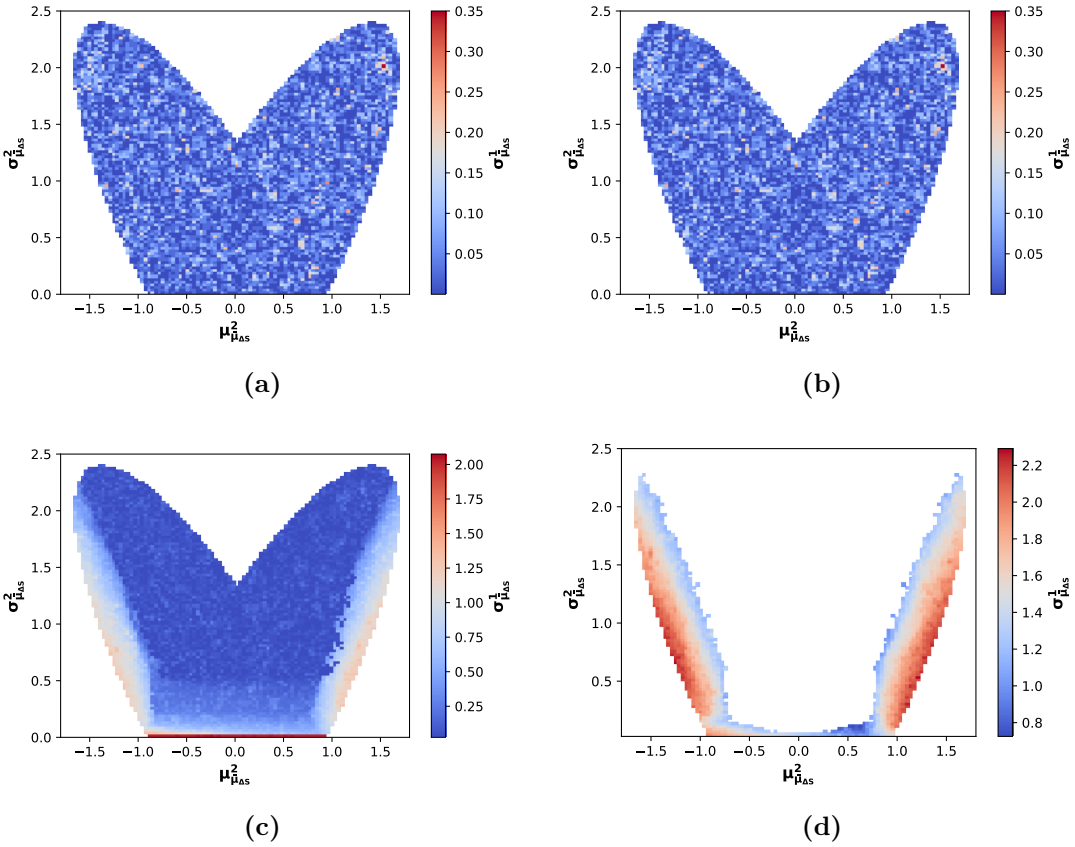


Figure 6.8: Projection of the four-dimensional distribution parameter subspace for which $n_{unique} = 62$ bit holds. The projection levels correspond to the absolute relative bias values (a) $|\mu^1_{\bar{\mu}_{\Delta S}}| = 0$, (b) $|\mu^1_{\bar{\mu}_{\Delta S}}| = 0.5$, (c) $|\mu^1_{\bar{\mu}_{\Delta S}}| = 1.0$ and (d) $|\mu^1_{\bar{\mu}_{\Delta S}}| = 1.5$.

ues to produce the same number of unique response-bits. In Fig. 6.8d large parts of the parameter space are prohibited due to the size constraints and the lower bound $n_{unique} \geqslant 62$.

6.3 Reproducibility - Measurement Noise

Commonly used error correction techniques are limited in the maximum number of bit-errors they can detect and correct. Therefore, detailed knowledge of the maximum number of bits expected to change their value upon key regeneration is required to properly size the applied correction code.

6.3.1 Fixed Parameters

Starting point for the following analysis is the probability $p_{unstable}|\Delta s, \sigma^2_{Noise} := P(X = \text{'unstable'})|\Delta s, \sigma^2_{Noise}$ that a bit is unstable (Eq. (4.40)). In case of Gaussian measurement noise, the equation can be rewritten in the form:

$$p_{unstable}|\Delta\bar{s}, \bar{\sigma}_{Noise} = \frac{1}{2}\left(1 - \operatorname{erf}\left(\frac{1}{\sqrt{2}}\frac{|\Delta\bar{s}|}{\bar{\sigma}_{Noise}}\right)\right), \tag{6.10}$$

where $\bar{\sigma}_{Noise}$ and $\Delta\bar{s}$ are defined as follows:

$$\bar{\sigma}_{Noise} = \frac{\sigma_{Noise}}{\sigma_{\Delta S}} \tag{6.11}$$

$$\Delta\bar{s} = \frac{\Delta s}{\sigma_{\Delta S}}. \tag{6.12}$$

The variable $\bar{\sigma}^2_{Noise}$ equals the noise variance scaled by the random parameter variance, hence $\bar{\sigma}_{Noise}$ can be interpreted as relative noise standard deviation. In this section, it is assumed that σ^2_{Noise} is constant (non-random) and the same for all pair differences from which response-bits are generated. This assumption of a common noise variance is reasonable, if the measurement error is solely attributed to the sampling process of a readout circuit shared by all devices comprising a PUF instance. $\Delta\bar{s}$ denotes the observed pair differences normalized by their random standard deviations. Under the Gaussian random field assumption, $\Delta\bar{s}$ is normally distributed

$$\Delta\bar{s} \sim N(\bar{\mu}_{\Delta S}, 1) \tag{6.13}$$

with the mean value being defined by the relative bias $\bar{\mu}_{\Delta S} = \mu_{\Delta S}/\sigma_{\Delta S}$. Therefore, the absolute values of the pair differences $|\Delta\bar{s}|$ follow a folded normal distribution (Sec. 3.1.3). The probability density function computed for different values of $\bar{\mu}_{\Delta S}$ is depicted in (Fig. 6.9).

By using the change of variable technique (Sec. 3.3), the density distribution function of $p_{unstable}$ can be determined:

$$\begin{aligned} f(p_{unstable}) = 2\bar{\sigma}_{Noise} \exp\left(-\frac{\bar{\mu}^2_{\Delta S}}{2} + \operatorname{erf}^{-1}(1 - 2p_{unstable})^2(1 - \bar{\sigma}^2_{Noise})\right) \\ \cdot \cosh\left(\sqrt{2\pi}\bar{\mu}_{\Delta S}\bar{\sigma}_{Noise} \operatorname{erf}^{-1}(1 - 2p_{unstable})\right) \end{aligned} \tag{6.14}$$

for $p_{unstable} \in [0, 0.5]$ and 0 otherwise. The expression is defined conditional on $\bar{\mu}_{\Delta S}$ and $\bar{\sigma}^2_{Noise}$. The graph of the density function plotted for different values of $(\bar{\mu}_{\Delta S}, \bar{\sigma}^2_{Noise})$ is shown in Fig. 6.10. As can be seen from the figures, the

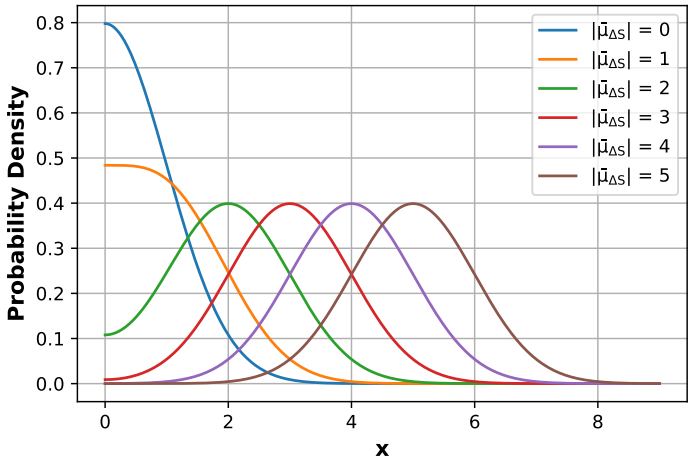
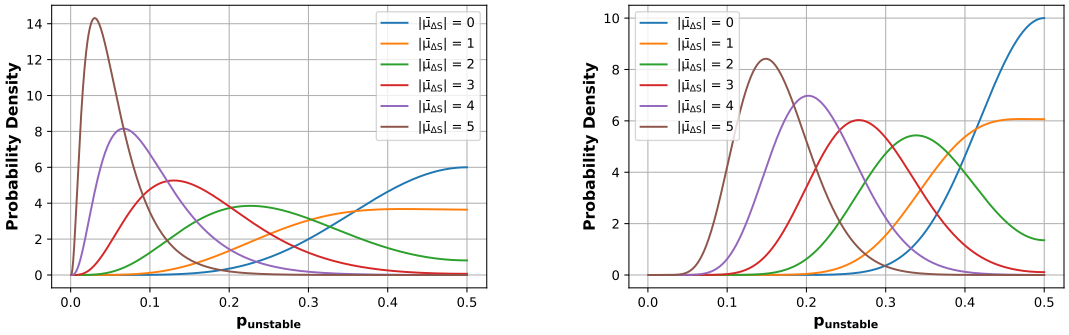


Figure 6.9: Probability density function of the folded normal distribution computed for relative bias values $\bar{\mu}_{\Delta S} = 0, 1, ..., 5$.

distribution shifts towards $p_{unstable} = 0$ for increasing bias values. This behavior is to be expected, since large bias values are more likely to generate pair differences farther away from the quantization level, effectively reducing the probability of bit-flips to occur. In contrast, larger noise variances $\bar{\sigma}^2_{Noise}$ generally increase the likelihood that a bit is unstable as the comparison of Fig. 6.10a and 6.10b reveals.



(a) **(b)**

Figure 6.10: Probability density function $f_{p_{unstable}}$ computed for relative bias values $\bar{\mu}_{\Delta S} = 1, 2, ..., 5$ at fixed relative noise standard deviations (a) $\bar{\sigma}_{Noise} = 3$ and (b) $\bar{\sigma}_{Noise} = 5$.

The next step is to estimate the number of unstable response-bits of an N-bit long key dependent on a set of parameter tuples $\{(\bar{\mu}^i_{\Delta S}, \bar{\sigma}^i_{Noise}) : i = 1, 2, ..., I\}$. First off, a set of J random samples $\{p^{ij}_{unstable} : j = 1, 2, ..., J\}$ is simulated from the

distribution $f(p^i_{unstable})$ conditional on the different relative bias values $\bar{\mu}^i_{\Delta S}$ and noise values $\bar{\sigma}^i_{Noise}$. Then, for each sample of $p^{ij}_{unstable}$ the corresponding binomial distribution function $B(k|N, p^{ij}_{unstable})$ is computed, where $k = 0, 1, ..., N$, denotes the number of unstable response-bits. The maximum number of unstable bits is defined as the α-percentile $n^i_{unstable} = n_{perc}(\alpha)|N, p^{ij}_{unstable}$ of the respective binomial distribution. According to this definition, the probability that more than $n^i_{unstable}$ response-bits are unstable is less than α percent. For a worst-case analysis, α is set to values close to 100 %, e.g. $\alpha = 99\,\%$ as used within this work.

Since the relation of the percentiles of a binomial distribution and N is non-linear, the maximum number of unstable response-bits generally depends on the length of a PUF response as shown in Fig. 6.11. Figure 6.12, for example, displays the dependence of $n_{unstable}$ on $\bar{\mu}_{\Delta S}$ and $\bar{\sigma}_{Noise}$ computed for a 256-bit response normalized to the response length.

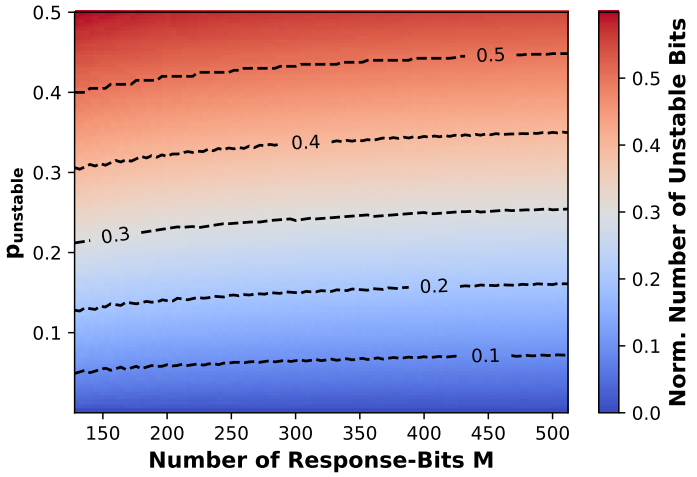


Figure 6.11: Normalized number of unstable bits $n_{unstable}$ dependent on the number of response-bits N and probability $p_{unstable}$.

6.3.2 Parameter Distribution

The restrictions imposed by the common bias and noise variance assumptions introduced in the previous subsection can be removed, if the distribution form of $\bar{\mu}_{\Delta S}$ and $\bar{\sigma}_{Noise}$ is known. Within the scope of this model, σ_{Noise} is treated as random variable due to process-related variations in the output impedance of the primitive devices or circuit components constituting a PUF instance. For

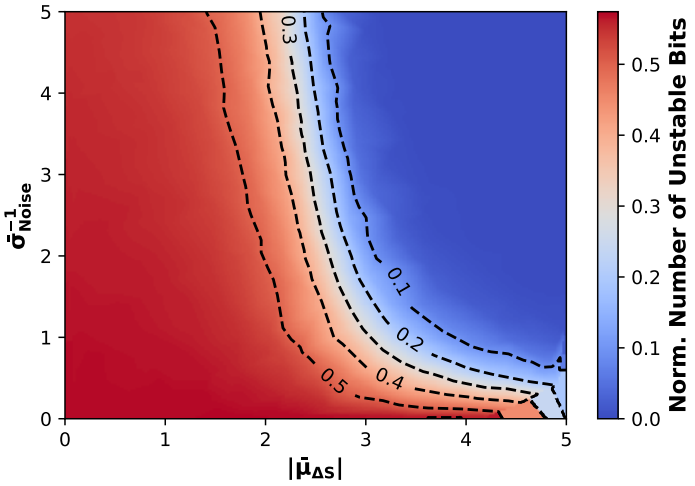


Figure 6.12: Normalized number of unstable bits $n_{unstable}$ dependent on relative bias $\bar{\mu}_{\Delta S}$ and relative noise standard deviation $\bar{\sigma}_{Noise}$.

example, the noise variance σ^2_{Noise} calculated for the oscillation frequency of 256 ring oscillator primitive pairs can not be expressed by a common value as is apparent from the distribution shown in Fig. 5.4.

Similar to the model building procedures described in previous sections, a hierarchical sampling strategy is employed to determine the dependence of $n_{unstable}$ on the distribution parameters θ_{Bias} and θ_{Noise} of the relative bias and noise distribution, respectively. In a first step, $N \times J$ samples pairs $(\bar{\mu}^{ij}_{\Delta S}, \bar{\sigma}^{ij}_{Noise}), j = 1, 2, ..., J$, are generated from the bias and noise distributions defined by a set of I distribution parameter tuples $\{(\theta^i_{Bias}, \theta^i_{Noise}) : i = 1, 2, ..., I\}$. At the next level, the set of corresponding error probabilities $\{p^{ij}_{unstable,k} : k = 1, 2, ..., N\}$ is computed according to Eq. (6.10) from which the probability vector $p^{ij} = (p^{ij}_{unstable,1}, p^{ij}_{unstable,2}, ..., p^{ij}_{unstable,N})$ is formed. Then, n^{ij}_{Noise} is calculated from the α-percentile of the Poisson binomial probability mass function $GB(k|N, p^{ij})$ which determines the distribution of the number of unstable response-bits k. The maximum number of unstable response-bits n^i_{Noise} are estimated from the α-sample percentile of the n^{ij}_{Noise} values in a similar fashion to the examples given in the previous sections.

Following these steps, multivariate interpolation can be used for out-of-sample prediction of n_{Noise} for distribution parameters not covered by the initial simulation runs. Since characterization of the noise source requires a large number of

parameters compared to the models described in the previous sections, application of interpolation techniques is recommended to minimize simulation times.

Example

In [Suh 05], a single-chip secure processor was proposed which uses a PUF circuit to facilitate authentication of the computing hard- and software to the user[4]. Reproducibility of the PUF keys was ensured by the use of a BCH$(255, 63, 61)$ error correction code[5] that corrects up to 30 erroneous bits out of 255 response-bits.

In this example, the subspace of the distribution parameters is determined for which n_{Noise} is compliant with the error correction capacity of 30 bit. As in the preceding sections, the relative bias was modeled by a normal distribution. In addition, the distribution form of the relative noise variance $\bar{\sigma}^2_{Noise}$ had to be determined. By means of Monte Carlo simulations, it was found that $\bar{\sigma}^2_{Noise}$ can be well approximated by a lognormal distribution[6]. The the lognormal distribution is defined by the location parameter $l \in \mathbb{R}$ and the scale parameter $s \in \mathbb{R}_{>0}$. Based on the experimental data, l and s were modeled by a normal and a lognormal distribution with distribution parameters (μ_l, σ_l) and (μ_s, σ_s), respectively.

It follows from the foregoing, that n_{Noise} depends on the six distribution parameters $(\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}}, \mu_l, \sigma_l, \mu_s, \sigma_s)$. As in the previous sections, the mean and standard deviation of the bias distribution were limited to $\mu_{\bar{\mu}_{\Delta S}} \in [0, 2]$ and $\sigma_{\bar{\mu}_{\Delta S}} \in (0, 1.8]$, respectively. The limits of the remaining parameters were estimated from a series of simulations carried out for four combinations of $\mu_{\bar{\mu}_{\Delta S}}$ and $\sigma_{\bar{\mu}_{\Delta S}}$, i.e. $(0, 0.5)$, $(0, 1.5)$, $(1, 0.5)$ and $(1, 1.5)$. Based on the simulation results, the respective parameter ranges of the noise distribution were set to $\mu_l \in [-14, 0]$, $\sigma_l \in [0.02, 0.18]$, $\mu_s \in [-2, -1]$ and $\sigma_s \in [0.01, 0.04]$.

From this initial parameter space 400 random points $(\mu^i_{\bar{\mu}_{\Delta S}}, \sigma^i_{\bar{\mu}_{\Delta S}}, \mu^i_l, \sigma^i_l, \mu^i_s, \sigma^i_s)$, $i = 1, 2, ..., 400$, were selected. For these parameter tuples 256×10000 sam-

[4]The authors chose to implement an Arbiter-based PUF design for their secure computing platform. However, Weak PUF concepts are also suitable candidates to realize the security requirements of their target application, since the proposed chip provides access to a cryptographic hash function.

[5]For more information on BCH codes, the reader is referred to [Bose 60].

[6]In general, the sum of two lognormally distributed random variables does not necessarily need to follow a lognormal distribution [Lo 12].

ples $(\bar{\mu}_{\Delta S}^{ij}, l^{ij}, s^{ij})$, $j = 1, 2, ..., 10000$, were generated. Bias values $\bar{\mu}_{\Delta S}^{ij}$ were drawn from the normal distribution $N(\bar{\mu}_{\Delta S}^{i}|\mu_{\bar{\mu}_{\Delta S}}^{i}, (\sigma_{\bar{\mu}_{\Delta S}}^{i})^2)$. Samples of the distribution parameters of the noise source distribution (l^{ij}, s^{ij}) were simulated from the normal distribution model $N(l^i|\mu_l^i, (\sigma_l^i)^2)$ and the lognormal distribution model Lognormal$((s^i)^2|\mu_s^i, (\sigma_s^i)^2)$, respectively. Then, for each tuple (l^{ij}, s^{ij}) a sample of the noise standard deviation $\bar{\sigma}_{Noise}^{ij}$ was generated from the lognormal distribution Lognormal$((\bar{\sigma}_{Noise}^{i})^2|l^{ij}, (s^{ij})^2)$. In a next step, the realizations $\{(\bar{\mu}_{\Delta S}^{ij}, \bar{\sigma}_{Noise}^{ij}) : i = 1, 2, ..., 400; j = 1, 2, ..., 10000\}$ were used to compute the probability vectors p^{ij} and the corresponding number of unstable response-bits n_{Noise}^{ij} as explained at the beginning of this subsection.

The maximum number of erroneous bits n_{Noise}^{i} resulting from the i-th distribution parameter tuple $(\mu_{\bar{\mu}_{\Delta S}}^{i}, \sigma_{\bar{\mu}_{\Delta S}}^{i}, \mu_l^i, \sigma_l^i, \mu_s^i, \sigma_s^i)$ was estimated from the 99 %-percentile of the respective sample values n_{Noise}^{ij}, $j = 1, 2, ..., 10000$. Lastly, the parameter space was confined to those points for which $n_{Noise} \leqslant 30\,\text{bit}$ and $(\mu_{\bar{\mu}_{\Delta S}}^{i}, \sigma_{\bar{\mu}_{\Delta S}}^{i}) \in \Omega_{random}$ holds.

A meaningful graphical representation of the complete solution space is no longer possible, due to the large number of distribution parameters. In order to reduce the dimensionality of the problem, the four parameters $(\mu_l, \sigma_l, \mu_s, \sigma_s)$ were set to the median values calculated from their respective empirical distributions of the 193 PUF instances. The distributions of n_{Noise} obtained for the corresponding projection in the $(\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}})$-plane are displayed in Fig. 6.13. Linear interpolation was used to increase the number of depicted sample points.

As can be seen from the figure, the largest number of bit-errors was attained at small mean and standard deviation values close to zero. Since bias distributions characterized by small distribution parameters are sharply peaked at zero bias values, it follows that realizations of the pair differences are centered around the quantization threshold. Consequently, the response-bits generated from these distributions are highly susceptible to measurement noise resulting in a large number of bit-errors. Increasing the mean absolute bias while keeping the standard deviation fixed shifts the distribution of the pair differences away from the quantization threshold, thus reducing the likeliness of bit-errors to occur. A larger variability of the bias distribution, on the other hand, places probability mass closer to the threshold in case of a non-zero mean bias, hence partially compensating the effect of large mean bias values. This behavior is well reflected by the graph of n_{Noise};

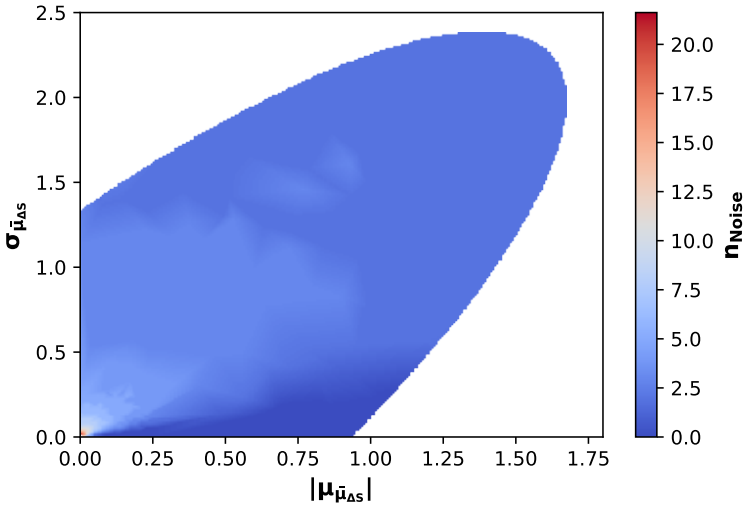


Figure 6.13: Projection of the six-dimensional parameter space of n_{noise} into the $(\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}})$-plane.

the smallest number of bit-errors was attained at large $\mu_{\bar{\mu}_{\Delta S}}$ and small values of $\sigma_{\bar{\mu}_{\Delta S}}$. In contrast larger standard deviation values tend to increase the number of bit-flips. Overall, the error values displayed in the graph were found to be smaller than the maximum capacity of the BCH$(255, 63, 61)$ error correction code.

Projection at other distribution parameter settings (corner values, interquartile range, etc.) can be analyzed in a similar fashion to get additional insight on the parameter boundaries compliant with all design limitations.

6.4 Summary

Different models were developed to link general design parameters of a PUF to its security relevant properties including the response randomness, uniqueness and reproducibility. Depending on the amount of available information on the distributional properties of the electrical characteristics of a PUF primitive, a number of simulation based analysis tools and design guidelines were proposed to ensure proper functionality of a PUF design. Furthermore, it was shown that high-performance PUF designs with minimal implementation overhead impose strict limitations on the distribution parameters of the underlying stochastic process which determines the electrical properties of the primitive devices.

Chapter 7

Weak PUF Concept

The experimental results and design considerations presented in the previous chapters clearly demonstrated the close relationship between the randomness, uniqueness and stability of signatures extracted from a PUF and its implementation form in hardware. The analysis of the design space, in particular, showed that extensive control over the fabrication process is necessary in order to realize high-performance designs. The strong dependence on the fabrication technology paired with the inability to control the processing conditions in a way that benefits the realization of a PUF design is highly undesirable for a number of reasons:

- Process-related variations used to generate PUF keys are likely to have a negative effect on the performance of conventional semiconductor devices. Process optimization and tighter manufacturing controls aim to reduce process variability in order to improve yield and reliability of integrated circuits. For PUFs, a reduction in the magnitude of random process variations is detrimental to the response entropy and uniqueness of the extracted keys.

- Performance results obtained from the experimental evaluation of a PUF design are tied to the processing conditions under which the tested PUF instances were manufactured. Since the processing conditions are specific to a particular combination of production line, technology and equipment, inference based on these results cannot be easily generalized.

- Transferring a PUF design to different production lines or technologies is likely to yield inconsistent performance results. Arguably, any form of pro-

cess or technology transfer is difficult, however PUFs are particularly affected by this problem, since the manufacturing technology usually can not be adapted to fit the need for large process variations required for efficient random key generation.

To illustrate the underlying problem, a simple PUF design based on integrated high-R poly resistors is considered. In case of this particular design, secret keys might be derived from differences in the voltage drop across nominally identically designed resistors whose resistance values are subjected to random process variations.

The nominal resistance R of a homogenous poly layer characterized by resistivity ρ, length L, width W and thickness t can be computed according to the following equation[1] [Hast 01]:

$$R = \frac{\rho}{t}\frac{L}{W}\,. \tag{7.1}$$

In practice, each of the four parameters is affected by process variations, resulting in a range of possible resistance values due to the propagation of uncertainty in ρ, L, W and t. Consequently, the variance of the resistance σ_R^2 is a function of the individual parameter variances:

$$\sigma_R^2 = f(\sigma_\rho^2, \sigma_L^2, \sigma_W^2, \sigma_t^2)\,. \tag{7.2}$$

The parameter variances are determined by the process settings and the precision of the manufacturing equipment operated at these settings. The variance σ_t^2, for example, depends on the precision of the used deposition system to produce a poly layer of nominal thickness t. In a similar fashion, σ_ρ^2 and the other two variances σ_L^2 and σ_W^2 originate from the limited precision of the implanter and the lithographic process equipment, respectively.

Analog circuits designed for non-PUF applications typically require matching resistors, hence the fabrication process and machines are adjusted to produce the target resistance required by the circuit design while minimizing σ_R^2. In contrast, for PUFs a large value of σ_R^2 is desirable to produce stable high-entropy response-bits. Since the processing conditions apply to all fabricated resistors, selective changes in the processing settings to increase the variability of PUF related

[1] Here, resistance contributions related to the terminal contacts of a resistor are ignored.

devices are not feasible. Increasing the parameter variability by introducing additional processing steps requires extra lithographic processes that, in turn, result in higher production costs. Moreover, semiconductor foundries might not allow the required modifications of their standard technologies. For these reasons, the designer of a PUF system usually has little to no control over the process technology to improve the performance of a PUF. The remaining possibility to enlarge the parameter variation is to modify layout dependent variance components such as σ_L^2 and σ_W^2, for example. By choosing a small and short sized resistor design close to the CD limits of the manufacturing technology, lithographic process variations can be exploited to increase σ_R^2. Since the resistor dimension is bounded by the minimum size required to produce a functional device, the maximal achievable values of σ_L^2 and σ_W^2 are limited. Furthermore, the maximum values σ_L^2 and σ_W^2 are tightly connected to the processing technology and manufacturing equipment, hence the problem of implementation dependent performance results remains unsolved.

7.1 Working Principle

In order to overcome the challenges stated in the beginning of this chapter, a generic Weak PUF concept is proposed which realizes a high-entropy hardware source capable of generating unique and stable response-bits independent of its design implementation.

For this purpose, three transformations were developed to modify the underlying probability distribution of the electrical parameters from which PUF-responses are generated without changing the actual manufacturing process. Within the context of the distribution parameter models introduced in Chap. 6, the transformations can be thought of as algebraic operations that confine the distribution parameters characteristic of an arbitrary fabrication process to values within the parameter subspace for which security relevant features of a PUF design are close to the optimum. In practice, this means that the relative bias (Sec. 6.1.1) and the error probabilities (Sec. 6.3.1) of all response-bits are reduced in order to achieve a high response entropy, uniqueness and stability. Figure 7.1 illustrates the effect of the transformations on the distribution parameters of two fabrication processes A and B.

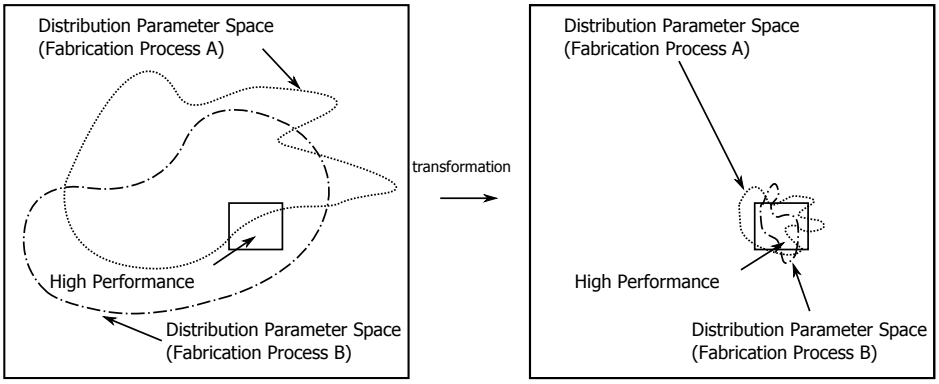


Figure 7.1: Transformation of the distribution parameter space conceptually illustrated for two fabrication processes A and B. The area highlighted by the rectangle indicates distribution parameters required to implement high-performance PUFs.

The transformations are realized by choosing specific combinations of device parameter pairs based on their distributional features. The first transformation selects parameter pairs featuring a similar systemic mismatch to reduce the overall response bias which in turn increases the uniqueness and entropy of the PUF responses. As shown in Sec. 6.3, the bit-error probabilities are highest for a relative bias close to zero, thus a second transformation is required to increase the reproducibility of the PUF responses. For this purpose, higher order pair combinations are formed that shift the parameter differences away from the quantization threshold. Since the resulting sequence of pair differences is entirely deterministic, a final randomization transformation must be applied to recover the initial response entropy. A detailed description of the three transformations is given in the following subsections.

Lastly, it shall be noted that no assumptions are made about the physical nature of the electrical parameters from which the responses are extracted, thus the transformation can be readily implemented for any type of PUF design using a differential quantization scheme.

7.1.1 Randomness Enhancement

The degree of randomness and uniqueness of individual response-bits is determined by the absolute systematic mismatch between parameter values relative to the variance of their random variation components (Sec. 6.1). In case of spatially

dependent process variations, the magnitude of the mismatch heavily depends on the selection of the parameter pairs. If the trend components of the parameter pairs do not cancel out, a low response entropy and uniqueness are to be expected as the empirical and theoretical analysis results of the ring oscillator PUF implementation showed (Chap. 4 and 6). To solve the problem of ambiguous pair selection schemes, a simple matching procedure is proposed to minimize the absolute difference in the trend components of all pairs.

The first step in finding the optimum selection of pairs is to determine the systematic trend components from the measured electrical characteristics of a PUF instance. For this purpose, measurement data recorded during the enrollment phase of a PUF instance can be used. Alternatively, measurements of the parameter values might be carried out for PUF devices fabricated on one or more reference wafers. Furthermore, it is conceivable to perform the subsequent analysis on simulation data taking into account the effect of spatially varying process conditions. Within the scope of this work the first option was chosen, since the presented methodology can be directly applied to the measurement data of individual PUF instances. For the purpose of estimating the trend components, variation decomposition techniques such as median polishing can be used.

Given the trend components $\{\mu_k : k = 1, 2, ..., 2N\}$ of the $2N$ characteristic parameters of a PUF instance, the selection of pairs is based on the sorted sequence of the trend components:

$$\mu_{(1)} \leqslant \mu_{(2)} \leqslant \mu_{(3)} \leqslant ... \leqslant \mu_{(2N)} \,. \tag{7.3}$$

The set of pair combinations

$$I = \{(i_1, j_1), (i_2, j_2), ..., (i_N, j_N)\} \tag{7.4}$$

that minimizes the systematic mismatch $|\mu_{(k)} - \mu_{(l)}|$, $k, l = 1, 2, ..., 2N$, $k \neq l$, between any two parameter values is obtained by simply forming exclusive pairs from subsequent trend components, i.e.

$$(\mu_{(1)}, \mu_{(2)}), (\mu_{(3)}, \mu_{(4)}), ..., (\mu_{(2N-1)}, \mu_{(2N)}) \,. \tag{7.5}$$

For illustration purposes, the randomness enhancement transformation was applied to the measurement data of a ring oscillator PUF instance containing $32 \times 16 = 512$ oscillation frequencies (Sec 5.2). Figure 7.2 shows the graph of

the systematic mismatch of the 512 oscillation frequencies obtained after sorting. As is apparent from the figure, the difference between subsequent frequencies is minimal compared to any other arrangement of the values. The corresponding distribution of the pairwise differences of the sorted mismatch values is displayed in Fig. 7.3a. For comparison purposes, the distribution of the differences computed for a column-wise selection of the mismatch pairs is plotted. Evidently, the mismatch distribution obtained by the matching pair selection procedure was sharply peaked around zero. In contrast, the column-wise pair selection scheme did not fully eliminate the mismatch between the oscillations frequencies as indicated by the large spread in the calculated trend components. The corresponding distributions of the observed pair differences are depicted in Fig. 7.3b. In case of the matching procedure, the detrended pair differences were symmetrically distributed around zero as to be expected in the case of normally distributed random pair values. Note that the variance of pair differences was smaller compared to the distribution obtained for the column-wise pair selection scheme, because the matching procedure removed the contributions of spatially varying device mismatch from the overall variability of the difference values.

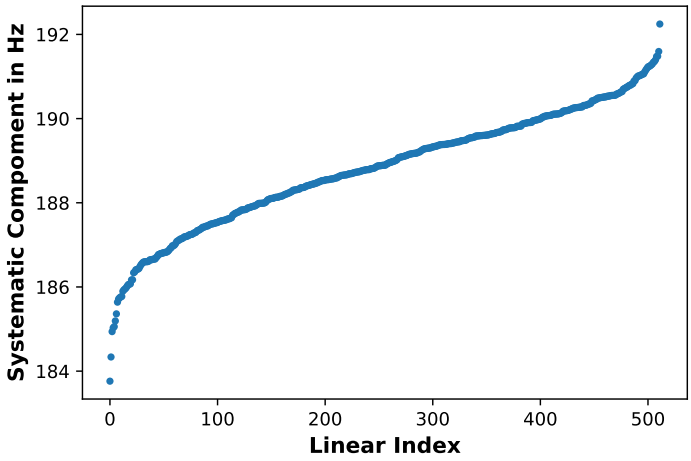
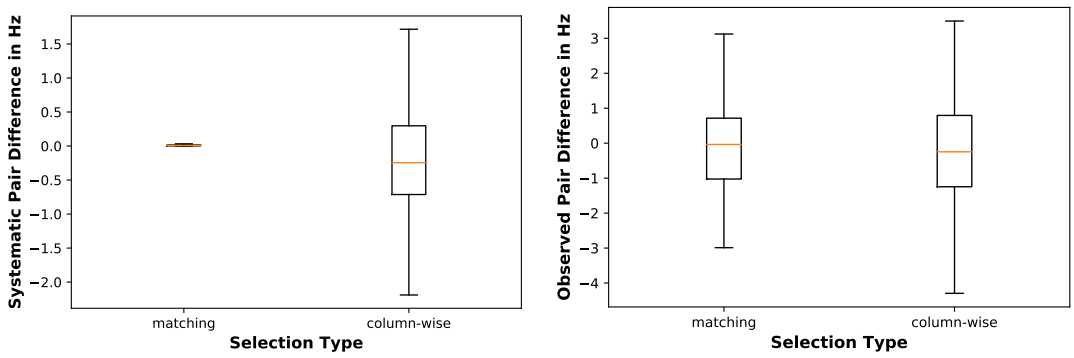


Figure 7.2: Graph of the sorted systematic variation components of 512 ring oscillator frequencies of a PUF instance.



(a) **(b)**

Figure 7.3: (a) Distribution of the systematic pair differences of 512 ring oscillator frequencies computed for matching pairs and for the column-wise pair selection scheme. (b) shows the corresponding distributions of the observed pair differences.

7.1.2 Reproducibility Enhancement

The distributions of matched trend components are centered around the quantization threshold. Consequently, the pair differences are more prone to noise-related fluctuations and environmental effects than prior to the transformation. In order to improve the reproducibility of the responses, higher order pair differences are formed. By specifically selecting pair combinations with large absolute differences in their parameter values, the probability density of the underlying pair distribution is shifted away from the quantization threshold. The transformed distribution features a range of parameter values around the quantization threshold for which the probability density equals zero. This gap in the density function counterbalances changes in the parameter differences, thus effectively increasing the stability of a PUF response. The width of the gap depends on the order of the pair differences.

Computation of K-th order pair differences of a set of $2N$ parameter values is accomplished by performing the following steps:

1. Compute the first order pair differences Δs_k, $k = 1, 2, ..., N$, of the measured parameters values s_l, $l = 1, 2, ..., 2N$. The difference of the m-th pair is calculated according to the equation

$$\Delta s_m = s_{i_m} - s_{j_m} \,, \tag{7.6}$$

where the pair indices $(i_m, j_m) \in I$ were determined by the matching pair procedure described in the previous section.

2. Invert the sign of all negative pair differences. The sign is inverted by swapping the order of the pair indices, i.e.

$$(i_k, j_k) \rightarrow (j_k, i_k), \text{ if } \Delta s_i < 0\,. \tag{7.7}$$

3. Sort the absolute pair differences in ascending order:

$$\Delta\tilde{s}_{(1)} \leqslant \Delta\tilde{s}_{(2)} \leqslant \Delta\tilde{s}_{(3)} \leqslant \Delta\tilde{s}_{(4)} \leqslant ... \leqslant \Delta\tilde{s}_{(N-1)} \leqslant \Delta\tilde{s}_{(N)} \tag{7.8}$$

with

$$\Delta\tilde{s}_{(m)} = \begin{cases} s_{i_m} - s_{j_m}, & \text{if } s_{i_m} > s_{j_m} \\ s_{j_m} - s_{i_m}, & \text{otherwise.} \end{cases} \tag{7.9}$$

4. Invert the sign of every second pair difference. Again, a change in the sign is realized by reversing the order of the pair indices. The resulting sequence can be written in the form

$$\Delta\tilde{s}_{(1)} \leqslant -|\Delta\tilde{s}_{(2)}| \leqslant \Delta\tilde{s}_{(3)} \leqslant -|\Delta\tilde{s}_{(4)}| \leqslant ... \leqslant \Delta\tilde{s}_{(N-1)} \leqslant -|\Delta\tilde{s}_{(N)}| \tag{7.10}$$

5. Second order pairs are obtained by calculating the $N/2$ differences

$$-|\Delta\tilde{s}_{(N)}|-\Delta\tilde{s}_{(1)}, \Delta\tilde{s}_{(N-1)}+|\Delta\tilde{s}_{(2)}|, -|\Delta\tilde{s}_{(N-2)}|-\Delta\tilde{s}_{(3)}, ..., \Delta\tilde{s}_{\left(\frac{N}{2}+1\right)}+|\Delta\tilde{s}_{\left(\frac{N}{2}\right)}|\,. \tag{7.11}$$

Iterative computation of the pair differences of lower order pairs yields higher order combinations.

Since the absolute values of the higher order pairs are larger than those of the constituting lower order terms, the distance between the pair differences and the quantization threshold increases with each iteration. Consequently, the distribution density of the transformed parameter pairs falls steeply to zero in the vicinity of the threshold.

Figure 7.4 shows the resulting sequences of 2nd to 4th order pair differences calculated for the frequency data of the same PUF instance to which the randomness enhancement transformation was applied in the previous section. The gap between the positive and negative pair differences is clearly visible in the figure. Furthermore, it can be seen that the gap width increases for higher order pairs at the expense of halving the number of pair differences for each new order.

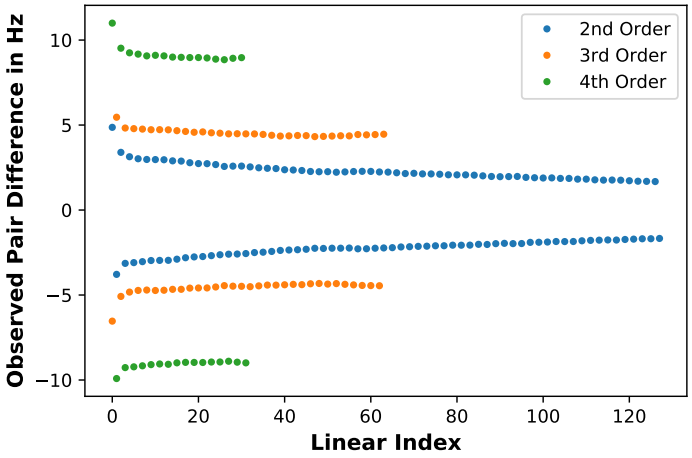


Figure 7.4: Sequences of the observed pair differences computed for 2nd, 3rd and 4th order pairs formed from 512 ring oscillator frequencies.

7.1.3 Response Randomization

The stability enhancement procedure produces a noise-resistant sequence of higher order pair differences which is fully deterministic by construction, i.e. the resulting responses have zero entropy. The response entropy inherent to the initial parameter distribution was transferred to the ordering of the corresponding pair indices resulting in a random sequence of index values. In order to use the PUF-responses for cryptographic applications, the entropy bound to the random sequence of indices has to be transferred back to the higher order parameter differences.

Randomization of a K-th order differences sequence is achieved by reordering the corresponding set of pair indices

$$I^{(K)} = \{J_1, J_2, ..., J_M\}, \tag{7.12}$$

where the index tuple $J_m = (i_m^{(1)}, j_m^{(1)}, i_m^{(2)}, j_m^{(2)}, ..., i_m^{(K)}, j_m^{(K)})$ determines the value of the m-th difference term. The indices are reordered in two steps:

1. The ordering of pairs contained in an index tuple J_m, $m = 1, 2, ..., M$, is reversed, if the number $i_m^{(1)} i_m^{(2)} ... i_m^{(K)}$ obtained by concatenating[2] the first

[2]To avoid ambiguity between the resulting values, the individual numbers are zero-padded prior to concatenation, e.g. the number 1 is written as 001 assuming that the total number of values is less than 1000.

pair elements is larger than the number $j_m^{(1)} j_m^{(2)} ... j_m^{(K)}$ of the second pair elements, i.e.

$$(i_m^{(1)}, j_m^{(1)}, i_m^{(2)}, j_m^{(2)}, ..., i_m^{(K)}, j_m^{(K)}) \rightarrow (j_m^{(1)}, i_m^{(1)}, j_m^{(2)}, i_m^{(2)}, ..., j_m^{(K)}, i_m^{(K)}), \quad (7.13)$$

if $i_m^{(1)} i_m^{(2)} ... i_m^{(K)} > j_m^{(1)} j_m^{(2)} ... j_m^{(K)}$ holds.

2. The index tuples $J_1, J_2, ..., J_M$ are ordered according to the numbers formed from the concatenated index pairs, where the number of the m-th index tuple J_m equals $i_m^{(1)} j_m^{(1)} i_m^{(2)} j_m^{(2)} ... i_m^{(K)} j_m^{(K)}$ given $i_m^{(1)} i_m^{(2)} ... i_m^{(K)} > j_m^{(1)} j_m^{(2)} ... j_m^{(K)}$.

The first step changes the sign of a random number of pair differences resulting in uniformly distributed bit-strings. The second step obfuscates the order of the absolute values of the pair differences.

Application of the randomization transformation to the sequence of the first order pairs calculated in the previous section resulted in the randomized sequence shown in Fig. 7.5. Since the randomization transformation only changed the

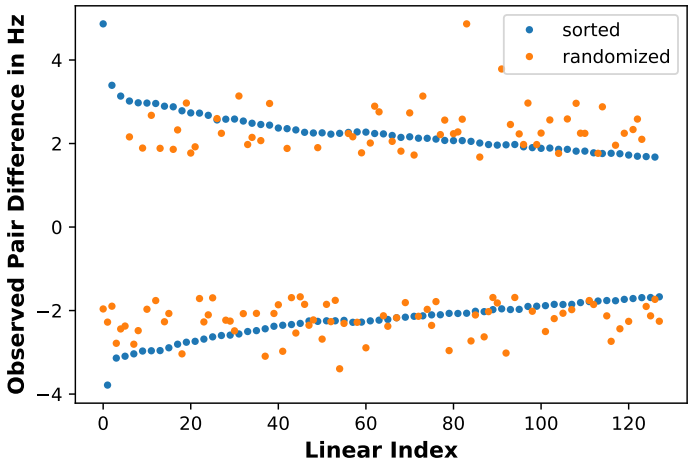


Figure 7.5: Sequence of the sorted and randomized 1st order pair differences formed from 512 ring oscillator frequencies.

sign and order of the pair differences, the gap width between the values was preserved. The random spread in the transformed pair differences suggests that the entropy contained in the sequence of the pair indices could be successfully recovered. Further analysis of the responses data extracted from the transformed pair differences will be carried out in Chap. 10.

7.2 Implementation

The basic circuit components required to implement a generic version of the presented Weak PUF concept are displayed in Fig. 7.6. The sources provide

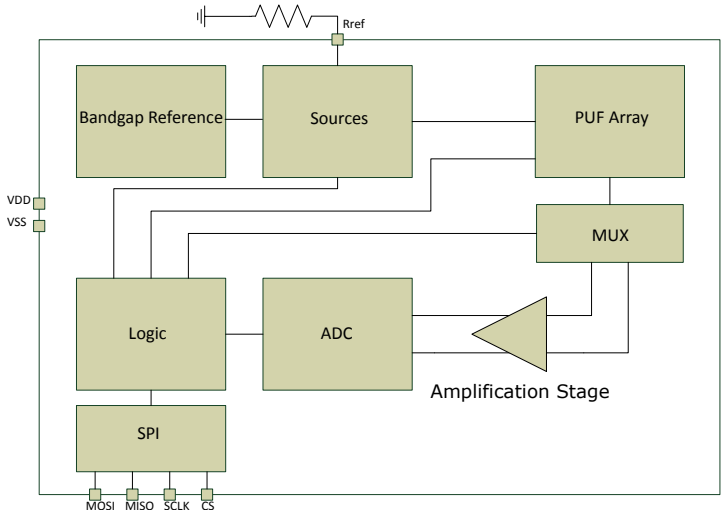


Figure 7.6: Schematic of the Weak PUF circuit components. Adapted from [Will 17b].

an electrical stimulus to the semiconductor devices located in the PUF array. A multiplexer is used to select pairs of array elements whose analog response signals are fed to the input of a differential ADC. The resulting binary signature is processed by a logic block to generate the final PUF response. The logic block primarily consists of a binary subtractor to compute higher order pair differences and registers needed to store intermediate outputs of the arithmetic operations. In order to minimize the silicon footprint of the PUF circuitry, the sets of indices used to define the higher order pair differences are determined off-chip during the enrollment phase. For example, the algorithm to find the array indices as described in Sec. 7.1.1 to 7.1.3 can be executed on an external microcontroller.

A rough estimation of the size of the readout circuit and the processing logic is given in Tab. 7.1. The listed gate counts were estimated for a third order pair quantization scheme that extracts a 32-bit response from 16×16 primitives per PUF instance using a differential 13-bit ADC.

As can be seen from the table, the area overhead of the proposed concept is mainly attributed to the size of the ADC. Therefore, this implementation form is best used in combination with sensor-based applications featuring on-chip analog-

Table 7.1: Weak PUF readout and processing logic overhead.

Component	Gate Count
Pair Selector	200
Differential 13-Bit ADC	32000
13-Bit Binary Subtractor	1000
Register for Data Storage	384
Total	33584

to-digital signal conversion. A lightweight variant might be employed, if no ADC is available. The pair selection procedure of the lightweight variant is nearly identical to the higher order difference algorithm with two exceptions:

1. Only first order pairs are considered. Therefore step 5 of the reproducibility enhancement sequence described in Sec. 7.1.2 is omitted.

2. Instead of computing higher order pairs, the M smallest differences in the sorted series of pair values (Eq. (7.10)) are discarded.

Quantization of the first order differences is achieved by means of a simple comparator, hence no ADC and processing logic are required. Disadvantageous to this approach is that the maximum gap width is limited by the variability of the pair differences, whereas the gap width of the original method is only limited by the number of pairs used to calculate higher order differences.

Chapter 8

Strong PUF Concept

Strong PUFs rely on the same working principle as Weak PUFs, i.e. they exploit process-related variations in the physical characteristics of a hardware system to generate random binary keys. Therefore, Strong PUF designs suffer from the same implementation dependent performance issues stated in the previous chapter. For example, consider the performance differences between the ASIC- and a FPGA-based design implementation of the simple arbiter PUF (Tab. 2.2): the mean inter-Hamming distance of the FPGA implementation was found to be as low as 1.05 %, whereas the ASIC implementation scored a mean inter-Hamming distance up to 47.13 %. In both cases, the experimentally determined inter-Hamming metric was lower than the ideal value of 50 %.

Another challenge related to the development of Strong PUF concepts is to design a hardware system that generates complex and highly challenge-sensitive CRPs resistant against machine learning attacks. A common approach to solve these problems is to increase the complexity of the challenge-response mechanism, often at the expense of a significantly reduced response stability [Zhou 17] and more elaborate PUF designs [Majz 08a]. This approach further contributes to the problem of implementation dependent performance issues.

Within the scope of this work, a new Strong PUF concept was developed to enhance the challenge-response behavior of any of Weak PUF design. Application of the concept to the Weak PUF design presented in Chap. 7 enables implementation of independent Strong PUF functionality robust against learning attacks as will be proven in Sec. 11.3. In the following sections, two basic concept implementations are described.

8.1 Working Principle

The proposed challenge-response mechanism is realized in form of a function that maps N pairs of signed or binary parameter values $\{(\beta_{1i}, \beta_{2i}) : i = 1, 2, ..., N\}$ of a system and a user defined N-bit challenge $C = (c_1, c_2, ..., c_N) \in \{0, 1\}^N$ to a response-bit $R \in \{0, 1\}$ (Fig. 8.1).

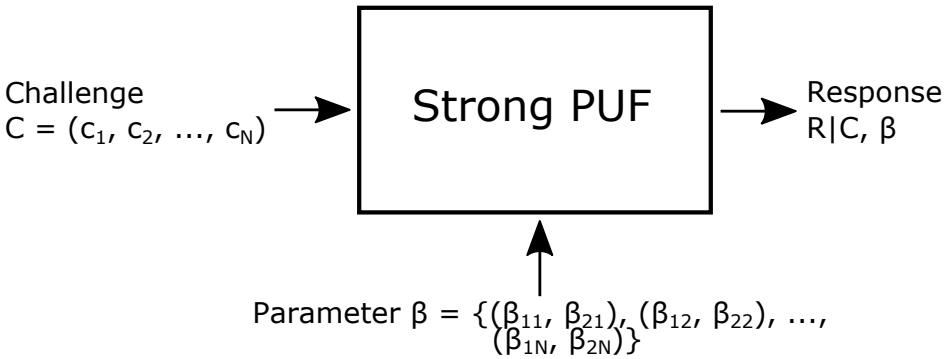


Figure 8.1: Schematic representation of the Strong PUF challenge-response mechanism.

In case of hardware-based Strong PUFs, β corresponds to a set of physical quantities whose signs or parities are randomly distributed due to the effect of stochastic process variations. Examples for real valued quantities are differences in the voltage drop across nominally identically fabricated resistors or differences in the oscillation frequencies of ring oscillators. Binary quantities can be obtained by applying some form of quantization scheme to the analog characteristics of a device.

From the set of parameter values β, N pair elements are selected according to a challenge C. If the i-th challenge-bit c_i equals a logical zero, then the first element β_{1i} of the i-th parameter pair (β_{1i}, β_{2i}) is chosen. Correspondingly, β_{2i} is selected, if $c_i = 1$ holds. The N parameter values are connected by logical XOR operations or elementwise multiplication, depending on whether the elements are binary or non-binary quantities. The resulting sign or parity of the connected parameter values is taken as response-bit. Without loss of generality subsequent analysis assumes the parameter values to be of digital nature.

In order to generate an M-bit response, the single response-bits of M Strong PUF instances can be concatenated. Alternatively, the response-bits obtained for M different challenges applied to the same PUF instance can be used to produce

a response of the desired length – similar to the standard native Strong PUF design described in Sec. 2.5.1.

8.1.1 Single Chain Function

A generic realization of the concept is shown in Fig. 8.2. The Strong PUF is comprised of N segments forming a chain-like structure. Each chain link contains two independent signed or binary parameter values. The parameters are connected by logic XOR operations. The state of the chain is configured according to the challenge-bits applied to the N links as shown in the figure. The response value of

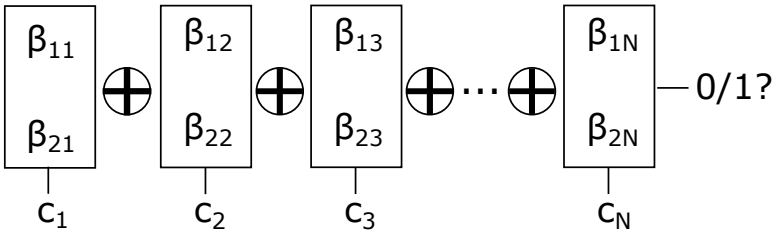


Figure 8.2: Single chain comprised of N links.

a given chain configuration formed from $2N$ binary parameter values is computed as follows:

$$R = \bigoplus_{k=1}^{N} \left(\bar{c}_k \beta_{1k} \oplus c_k \beta_{2k} \right) , \tag{8.1}$$

where $\bar{c}$ denotes the negated state of a boolean value c.

Example

In this example, a single chain featuring $N = 4$ links with parameter values $\beta = \{(\beta_{11} = 0, \beta_{21} = 0), (\beta_{12} = 1, \beta_{22} = 1), (\beta_{13} = 0, \beta_{23} = 1), (\beta_{14} = 0, \beta_{24} = 1)\}$ is considered. Furthermore, let $C_1 = (0, 1, 0, 0)$, $C_2 = (0, 1, 1, 1)$, $C_3 = (0, 1, 0, 1)$ denote a set of challenges, then the corresponding response values R_1, R_2 and R_3 can be readily computed:

$$R_1 = \beta_{11} \oplus \beta_{22} \oplus \beta_{13} \oplus \beta_{14} = 0 \oplus 1 \oplus 0 \oplus 0 = 1 \tag{8.2}$$

$$R_2 = \beta_{11} \oplus \beta_{22} \oplus \beta_{23} \oplus \beta_{24} = 0 \oplus 1 \oplus 1 \oplus 1 = 1 \tag{8.3}$$

$$R_3 = \beta_{11} \oplus \beta_{22} \oplus \beta_{13} \oplus \beta_{24} = 0 \oplus 1 \oplus 0 \oplus 1 = 0 \,. \tag{8.4}$$

8.1.2 Twisted Chain Function

In order to increase the resistance against model building attacks, the N segments of two single chains are piecewise intertwined as shown in Fig. 8.3. A simple mixing of the i-th chain segment of the first chain can be achieved by swapping the parameter values between the respective segments of the two chains. The segments are swapped, if the XOR connection of the response-bit R^1_{i-1} computed for the preceding $i-1$ sections of the first chain and the response-bit R^2_{i-2} calculated up to the $i-2$-th segment of the second chain equal a logical 1. In this case, the response value is obtained by XORing the parity S^2_i of the i-th segment of the second chain with R^1_{i-1}. Otherwise, no swap occurs and the response value equals the XOR combination of the parity value S^1_i of the i-th segment of the first chain and R^1_{i-1}. A similar swapping procedure is applied to the second chain.

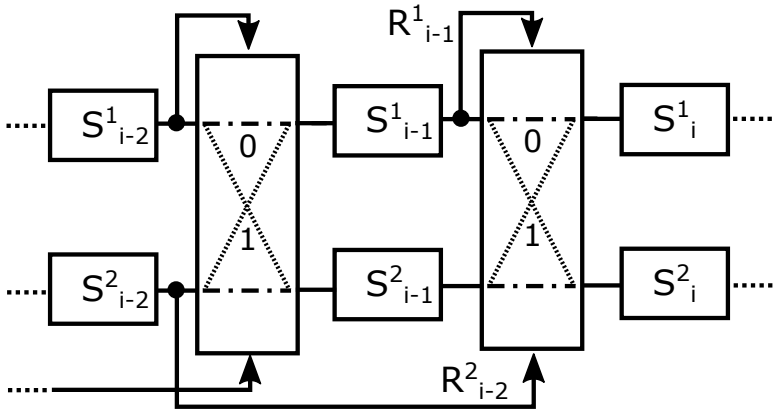


Figure 8.3: Feedforward links of the i-th segment of the first chain.

The response values R^1_i and R^2_i, $i = 1, 2, ..., N$, of the two chains, can be expressed in terms of two coupled recursive equations:

$$R^1_i = R^1_{i-1} \oplus \overline{(R^1_{i-1} \oplus R^2_{i-2})} S^1_i \oplus (R^1_{i-1} \oplus R^2_{i-2}) S^2_i \tag{8.5}$$

$$R^2_i = R^2_{i-1} \oplus \overline{(R^2_{i-1} \oplus R^1_{i-2})} S^2_i \oplus (R^2_{i-1} \oplus R^1_{i-2}) S^1_i \tag{8.6}$$

The parity S^j_i of the i-th segment of the j-th chain is computed according to the following equation:

$$S^j_i = \bigoplus_{k_i} (\bar{c}_{k_i} \beta_{1k_i} \oplus c_{k_i} \beta_{2k_i}) \ . \tag{8.7}$$

Here, the index tuple (k_i, j) denotes the k-th parameter pair of the i-th segment of the j-th chain. The response-bits R^1_0, R^1_{-1}, R^2_0 and R^2_{-1} are set equal to the

parity of a larger section of non-overlapping chain segments in order to minimize the bias of the initial bit-values. For example, the initial value for R_0^1 might be set to the parity computed for half the segments of the first chain and R_{-1}^1 is initialized to the parity of the remaining half. The response-bits R_0^2 and R_{-1}^2 are computed from the corresponding segments of the second chain.

In order to meet the SAC, the chains should be evaluated for the set of all challenge variants obtained by subsequently rotating the sequence of challenge-bits by one position either to the left or to the right.

8.1.3 Example

In this example, a simple feedforward construction with four segments per chain is considered. Each chain segment consists of a single parameter pair. The first chain is characterized by the following set of parameter values:

$$\beta^1 = \{(\beta_{11}^1 = 0, \beta_{21}^1 = 0), (\beta_{12}^1 = 1, \beta_{22}^1 = 1), (\beta_{13}^1 = 0, \beta_{23}^1 = 1), (\beta_{14}^1 = 0, \beta_{24}^1 = 1)\}.$$

The parameters of the second chain are set to the values:

$$\beta^2 = \{(\beta_{11}^2 = 1, \beta_{21}^2 = 0), (\beta_{12}^2 = 0, \beta_{22}^2 = 0), (\beta_{13}^2 = 1, \beta_{23}^2 = 1), (\beta_{14}^2 = 1, \beta_{24}^2 = 0)\}.$$

Given these two sets of parameter values, the challenge-response pairs (C, R_4^1) and (C, R_4^2) for a challenge $C = (c_1, c_2, c_3, c_4) = (0, 1, 1, 0)$ are calculated as follows:

Computation of $\mathbf{S_i^1}$ and $\mathbf{S_i^2}$ for $\mathbf{i = 1, 2, 3, 4}$

$$(S_1^1, S_2^1, S_3^1, S_4^1) = (\beta_{11}^1 = 0, \beta_{22}^1 = 1, \beta_{23}^1 = 1, \beta_{14}^1 = 0) \tag{8.8}$$

$$(S_1^2, S_2^2, S_3^2, S_4^2) = (\beta_{11}^2 = 1, \beta_{22}^2 = 0, \beta_{23}^2 = 1, \beta_{14}^2 = 1) \tag{8.9}$$

Computation of $\mathbf{R_{-1}^1}$ and $\mathbf{R_{-1}^2}$

$$R_{-1}^1 = S_1^1 \oplus S_2^1 = 0 \oplus 1 = 1 \tag{8.10}$$

$$R_{-1}^2 = S_1^2 \oplus S_2^2 = 1 \oplus 0 = 1 \tag{8.11}$$

Computation of $\mathbf{R_0^1}$ and $\mathbf{R_0^2}$

$$R_0^1 = S_3^1 \oplus S_4^1 = 1 \oplus 0 = 1 \tag{8.12}$$

$$R_0^2 = S_3^2 \oplus S_4^2 = 1 \oplus 1 = 0 \tag{8.13}$$

Computation of $\mathbf{R_1^1}$ and $\mathbf{R_1^2}$

$$R_1^1 = R_0^1 \oplus \overline{(R_0^1 \oplus R_{-1}^2)} S_1^1 \oplus (R_0^1 \oplus R_{-1}^2) S_1^2 = 1 \oplus \overline{(1 \oplus 1)} 0 \oplus (1 \oplus 1) 1 = 1 \quad (8.14)$$
$$R_1^2 = R_0^2 \oplus \overline{(R_0^2 \oplus R_{-1}^1)} S_1^2 \oplus (R_0^2 \oplus R_{-1}^1) S_1^1 = 0 \oplus \overline{(0 \oplus 1)} 1 \oplus (0 \oplus 1) 0 = 0 \quad (8.15)$$

Computation of $\mathbf{R_2^1}$ and $\mathbf{R_2^2}$

$$R_2^1 = R_1^1 \oplus \overline{(R_1^1 \oplus R_0^2)} S_2^1 \oplus (R_1^1 \oplus R_0^2) S_2^2 = 1 \oplus \overline{(1 \oplus 0)} 1 \oplus (1 \oplus 0) 0 = 1 \quad (8.16)$$
$$R_2^2 = R_1^2 \oplus \overline{(R_1^2 \oplus R_0^1)} S_2^2 \oplus (R_1^2 \oplus R_0^1) S_2^1 = 0 \oplus \overline{(0 \oplus 1)} 0 \oplus (0 \oplus 1) 1 = 1 \quad (8.17)$$

Computation of $\mathbf{R_3^1}$ and $\mathbf{R_3^2}$

$$R_3^1 = R_2^1 \oplus \overline{(R_2^1 \oplus R_1^2)} S_3^1 \oplus (R_2^1 \oplus R_1^2) S_3^2 = 1 \oplus \overline{(1 \oplus 0)} 1 \oplus (1 \oplus 0) 1 = 0 \quad (8.18)$$
$$R_3^2 = R_2^2 \oplus \overline{(R_2^2 \oplus R_1^1)} S_3^2 \oplus (R_2^2 \oplus R_1^1) S_3^1 = 1 \oplus \overline{(1 \oplus 1)} 1 \oplus (1 \oplus 1) 1 = 0 \quad (8.19)$$

Computation of $\mathbf{R_4^1}$ and $\mathbf{R_4^2}$

$$R_4^1 = R_3^1 \oplus \overline{(R_3^1 \oplus R_2^2)} S_4^1 \oplus (R_3^1 \oplus R_2^2) S_4^2 = 0 \oplus \overline{(0 \oplus 1)} 0 \oplus (0 \oplus 1) 1 = 1 \quad (8.20)$$
$$R_4^2 = R_3^2 \oplus \overline{(R_3^2 \oplus R_2^1)} S_4^2 \oplus (R_3^2 \oplus R_2^1) S_4^1 = 0 \oplus \overline{(0 \oplus 1)} 1 \oplus (0 \oplus 1) 0 = 0 \quad (8.21)$$

In a next step, the computation of the chain parities are continued for the challenge $C' = (c_4, c_1, c_2, c_3) = (0, 0, 1, 1)$ obtained by rotating C one bit-position to the right. The sets of chain segments are selected in accordance with new challenge C':

$$(S_1^1, S_2^1, S_3^1, S_4^1)|C' = (\beta_{11}^1 = 0, \beta_{12}^1 = 1, \beta_{23}^1 = 1, \beta_{24}^1 = 1) \quad (8.22)$$
$$(S_1^2, S_2^2, S_3^2, S_4^2)|C' = (\beta_{11}^2 = 0, \beta_{12}^2 = 0, \beta_{23}^2 = 1, \beta_{24}^2 = 1) \quad (8.23)$$

Given this selection of chain segments, the challenge-response pairs (C', R_4^1) and (C', R_4^2) are iteratively calculated from previous response-bits with $R_0^1|C' = R_4^1|C$, $R_{-1}^1|C' = R_3^1|C$, $R_0^2|C' = R_4^2|C$ and $R_{-1}^2|C' = R_3^2|C$. The whole procedure is repeated for the rotated challenges $C'' = (c_3, c_4, c_1, c_2) = (1, 0, 0, 1)$ and $C''' = (c_2, c_3, c_4, c_1) = (1, 1, 0, 0)$.

8.2 Implementation

Realization of the twisted chain structure with basic logic gates is straightforward. Rewriting Eq. (8.5) and (8.6) in the following form

$$R_i^1 = R_{i-1}^1 \oplus S_i^1 \oplus (R_{i-1}^1 \oplus R_{i-2}^2)(S_i^1 \oplus S_i^2) \tag{8.24}$$

$$R_i^2 = R_{i-1}^2 \oplus S_i^2 \oplus (R_{i-1}^2 \oplus R_{i-2}^1)(S_i^2 \oplus S_i^1)\,, \tag{8.25}$$

suggests that each of the two chains can be constructed from four XOR-gates and one AND-gate as shown in Fig. 8.4. The response-bits are generated by sequen-

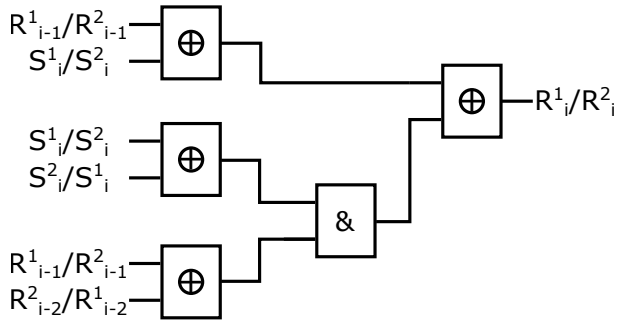


Figure 8.4: Basic logic components needed to implement the proposed twisted chain Strong PUF concept.

tially evaluating both chains. The resulting output values are then fed back to the corresponding inputs of the first set of XOR-gates. The response values of previous two iterations are temporarily stored in a register and updated accordingly. Two multiplexer are used to switch through the different chain segments S_i^1, S_i^2 at each iteration. The state of the chain segments can be evaluated on the fly when their respective cycle is due, hence no additional storage is required.

Chapter 9

Concept Implementation

Both PUF concepts introduced in the previous two chapters were validated by experimental data obtained from four different types of PUF primitives including nMOS transistors, high-R poly resistors, image sensors and ring oscillators. A description of each device is given in the following sections; key features are listed in Tab. 9.1. Furthermore, the various operating conditions under which the individual device tests were carried out are briefly summarized.

Table 9.1: Overview over process technology and implementation form of the tested devices.

Device	Design	Technology	Implementation
nMOS Transistor	custom	350 nm CMOS (in-house)	ASIC
High-R Poly Resistor	custom	350 nm CMOS (in-house)	ASIC
Image Sensor	standard	500 nm CMOS (in-house)	ASIC
Ring Oscillator	standard	90 nm CMOS (external)	FPGA

9.1 nMOS Transistor PUF

The nMOS transistor PUF design is comprised of a 16×16 array of over-critically sized digital nMOS transistors fabricated in a standard 350 nm in-house CMOS bulk technology. The nominal gate-length of $L = 280\,\text{nm}$ and gate-width of $W = 360\,\text{nm}$ after poly sizing were chosen to be smaller than the minimum al-

lowed feature size to enhance the effect of CD variations on the electrical characteristics of the devices.

For their use as PUFs, the transistors were operated in a diode-connected state by shortening the drain and gate terminals. The drain voltage at a fixed drain current of 10 µA was taken as electrical parameter from which binary responses were generated. The source voltage was set to ground. The transistor arrays were electrically characterized using a simple integrated readout circuit composed of current sources, row and column multiplexers to select individual transistors and an ADC to measure the drain voltage. Further details on the readout circuit are given in [Will 17b]. The devices were characterized under standard conditions at an operating temperature of 25 °C and an ADC readout voltage of 3.3 V. Additional measurements were performed at different chip temperatures in the range of 0 °C to 85 °C and three supply voltage settings $V_{dd} = 3.14\,\text{V}, 3.3\,\text{V}$ and 3.46 V. An overview over all test conditions is given in Tab. 9.2.

Table 9.2: Overview over transistor device tests.

Temperature in °C	Supply Voltage in V	Number of Devices
0	3.14	5
0	3.46	5
25	3.14	5
25	3.3	68
25	3.46	5
35	3.3	47
50	3.3	47
65	3.3	47
80	3.14	5
80	3.3	47
80	3.46	5

9.2 High-R Poly Resistor PUF

The resistor PUF design contained 256 high resistance poly resistors arranged in a 16 × 16 array. The design was implemented and processed in a proprietary

350 nm in-house CMOS bulk technology.

The width of the resistors was set to the minimum allowed size of 600 nm to increase variability of the poly resistance due to lithographic process variations. Furthermore, the overlap of the salicide blocker over the resistor was limited to 200 nm (minimum allowed overlap: 600 nm) to allow for a partial silicidation of the poly lane resulting in additional variability of the resistance.

Binary keys were extracted from random mismatch in the voltage drop across the resistors operated at a fixed terminal current of 10 µA. On-chip measurements of the voltage were performed using the same readout circuit as described in the previous section.

Standard measurements were carried out at an operating temperature of 25 °C and a readout voltage of 3.3 V. Further data samples were acquired at operating temperatures between 0 °C and 80 °C. The supply voltage of the ADC was varied between three levels $V_{dd} = 3.14\,\mathrm{V}, 3.3\,\mathrm{V}, 3.46\,\mathrm{V}$. The measurements performed within the scope of this work are listed in Tab. 9.3.

Table 9.3: Overview over resistor device tests.

Temperature in °C	Supply Voltage in V	Number of Devices
0	3.14	5
0	3.46	5
25	3.14	5
25	3.3	50
25	3.46	5
35	3.3	23
50	3.3	23
65	3.3	23
80	3.14	5
80	3.3	23
80	3.46	5

9.3 Image Sensor PUF

The sensor PUF implementation is based on optical sensors fabricated in an in-house 500 nm CMOS bulk technology. In this case, a row of 128 optically screened dark pixel values served as PUF primitive from which the responses were computed.

The dark pixel values of the sensor elements were measured using an integrated readout circuit. Measurements were taken at −40 °C and 80 °C. At both operating temperatures, the relative illumination intensity of the sensor was varied between 0 % and 100 %. Table 9.4 provides a summary of all measurements.

Since no room temperature data was available, the measurements performed at −40 °C and zero percent illumination were taken as reference operating conditions for which the reproducibility of the PUF responses is evaluated in later sections.

Table 9.4: Overview over sensor device tests.

Temperature in °C	Relative Illumination in %	Number of Devices
-40	0	7852
-40	20	7852
-40	50	7852
-40	80	7852
-40	100	7852
80	0	7852
80	20	7852
80	50	7852
80	80	7852
80	100	7852

9.4 Ring Oscillator PUF

The PUF concepts were verified for ring oscillators implemented on off-the-shelf Xilinx FPGAs manufactured in a 90 nm technology [Mait 10][1]. As stated previ-

[1] In the reference given, experimental results of only 125 FPGAs instances were reported. The published dataset contains measurements of 193 PUF instances.

ously, secret response-bits were extracted from the frequencies of the oscillator devices synthesized on the FPGA instances. Detailed information about the measurements was given in Sec. 5.2. A summary of the test dataset containing ring oscillator frequency samples acquired under different operating conditions can be found in Tab. 9.5.

Table 9.5: Overview over ring oscillator device tests

Temperature in °C	Supply Voltage in V	Number of Devices
25	0.96	5
25	1.08	5
25	1.2	193
25	1.32	5
25	1.44	5
35	1.2	5
45	1.2	5
55	1.2	5
65	1.2	5

Chapter 10

Experimental Weak PUF Concept Validation

The Weak PUF concept presented in Chap. 7 was implemented and tested for the different device types described in Chap. 9. In addition, the performance of the following PUF concepts (Sec. 2.4.4) were analyzed for comparison purposes:

- The neighbor-based pair selection scheme. In this instance, the pairs were formed from PUF structures placed in neighboring columns within a device array.
- The regression-distiller method. In contrast to the original distiller concept, elastic net regularization [Frie 10] was used to generate sparse regression models including up to 4th order polynomials. The regularization parameters were determined by simple 3-fold cross validation [Arlo 10]. A decisive advantage over the ordinary least squares solution proposed by the authors of the distiller method is the lowered risk of overfitting the systematic trend components, thus preserving the variability of the random device mismatch from which response-bits are extracted.
- The normalization-based bias removal procedure.
- The 1-out-of-K masking approach. Similar to the setup in [Suh 07] a group size of $K = 8$ PUF structures was chosen. With the exception of the sensor-based PUF design, the groups were selected from non-overlapping sets of 4×2 neighboring PUF structures. In case of the image sensor PUF, the 128 sensor pixel per PUF instance were simply divided into groups of 8 pixels.

Since the error correction capacity of index-based syndrome coding is equivalent to the pair-masking approach, only the latter concept was evaluated. The robust pair selection scheme was excluded from the analysis as the method was considered impractical for industrial-scale production due to the extensive device tests at different operating conditions needed to implement the concept.

The performance of all concepts was evaluated with regard to the response randomness, uniqueness and reproducibility. The randomness was measured in terms of the relative parameter pair bias introduced in Chap. 6. The uniqueness and reproducibility of the PUF responses were assessed by means of the inter- and intra-Hamming distance, respectively.

Computation of the performance metrics was based on real measurement data. The binary PUF responses, however, were calculated off-chip due to the lack of the required post-processing logic to fully implement the concepts in hardware. To facilitate a direct comparison between the results of the different device types, each PUF response was generated from 128 parameter pairs per PUF instance. In case of the ring oscillator PUF design, the set of 256 oscillation frequency pairs was limited to 128 pairs. Since the sensor-based PUF design featured only 64 dark pixel pairs per sensor, the responses of two sensor devices were concatenated to form a single PUF instance.

10.1 Randomness

Evaluation of the response randomness of the four PUF designs was based on the test data obtained under device-specific standard operating conditions (Chap. 9). The number of PUF instances used to compute the bias distributions are listed in Tab. 10.1.

From these datasets, the relative bias distributions of all PUF instances of a given design were computed to quantify the degree of randomness exhibited by the different device types. In case of the proposed Weak PUF concept, the column-wise pair selection scheme and the 1-out-of-8 masking approach, the pair bias was estimated according to the polishing procedure described in Chap. 5. For the other two methods, bias estimates were obtained from the detrended data after application of the concepts in order to determine the remaining systematic mismatch between the PUF structures. For this purpose, the detrended regression

residuals and the normalized parameter values were analyzed.

The box plots of the bias distributions obtained for the proposed pair matching procedure (Sec. 7.1.1) are displayed in Fig. 10.1a. The corresponding plots for the other four concepts are shown in Fig. 10.1b to 10.1e.

It is apparent from the bias distributions in Fig. 10.1a that the matching procedure successfully eliminated the systematic mismatch between the parameter pairs for all four device types. The bias distributions are tightly centered around a relative bias of zero, as indicated by the vanishing box heights and the small distances between the box whiskers.

In contrast, the other four quantization concepts did not fully detrend all pair differences resulting in a large variability of the calculated pair biases. The relative bias calculated for the column-wise pair selection scheme, the regression-distiller method and the normalization procedure were found to be similarly distributed with absolute values in the range between two to three. Therefore, the application of the distiller and normalization procedure did not improve on the results obtained for the simple column-wise quantization scheme, despite the implementation overhead associated with both methods. The masking approach yielded the overall largest mismatch as the procedure does not compensate for systematic mismatch between devices. Consequently, the procedure preferably selects biased pairs because the observed absolute differences are likely to be larger than those of pairs featuring a small mismatch.

Comparison of all five concepts clearly showed that the proposed matching pair scheme resulted in the highest response randomness – independent of the device type and electrical characteristic from which response-bits were extracted.

Table 10.1: Overview over the number of PUF instances used to compute the relative parameter bias and inter-Hamming distributions.

Device Type	Number of PUF Instances
Transistor	68
Resistor	50
Sensor	3926
Oscillator	193

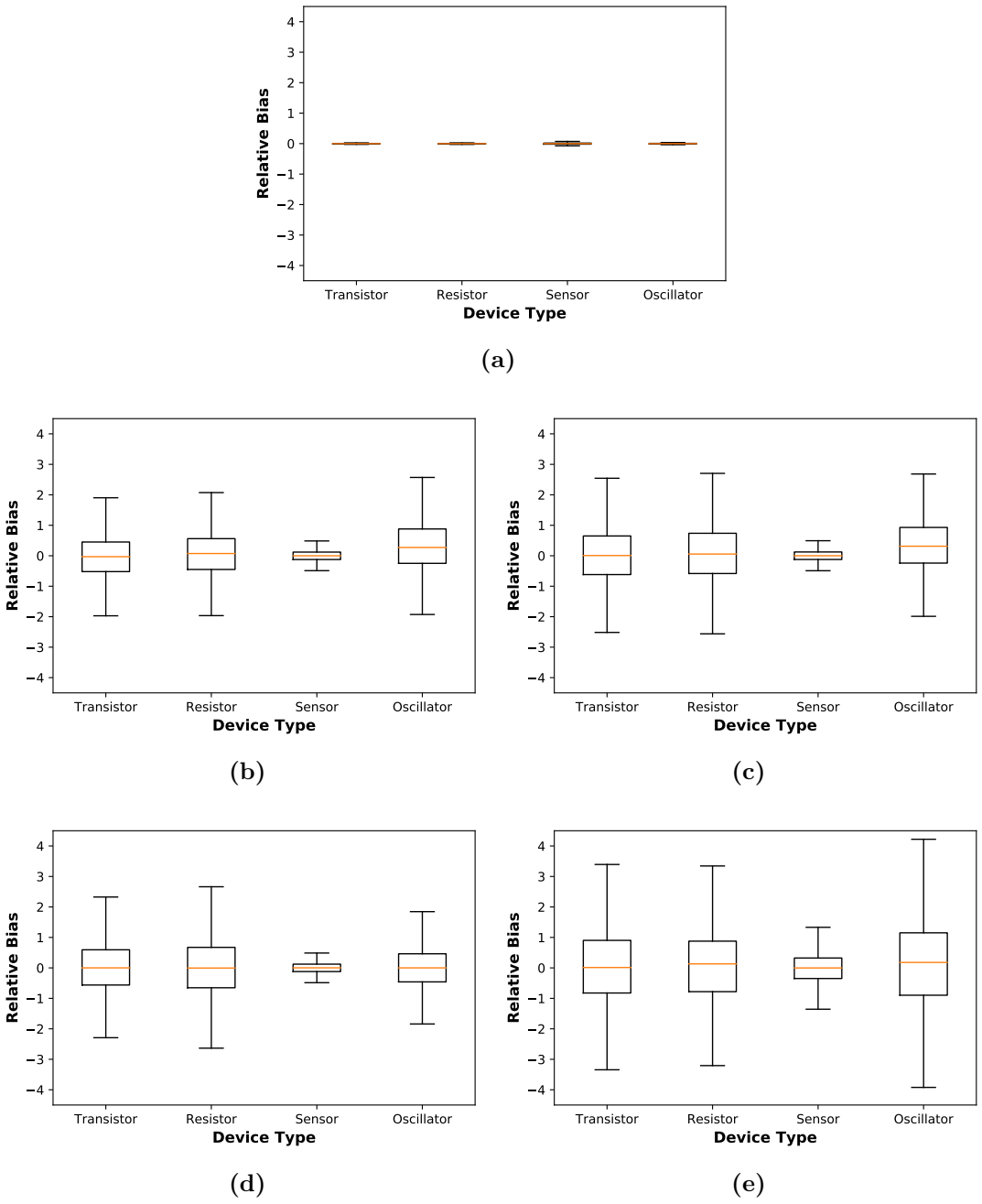


Figure 10.1: Relative bias distributions obtained for (a) the proposed Weak PUF concept implementation, (b) the column-wise pair selection scheme, (c) the regression-distiller method, (d) the normalization-based bias removal procedure and (e) the 1-out-of-8 masking approach.

10.2 Uniqueness

The inter-Hamming distance distributions were determined from the empirical PUF data obtained under standard operating conditions (Tab. 10.1). The inter-Hamming mass functions calculated for 32-bit responses generated from third order pair differences are displayed in Fig. 10.2a. The mass functions obtained for the column-wise pair selection scheme, the distiller method, the normalization procedure and the masking approach are plotted in Fig. 10.2b to 10.2e, respectively. The ideal probability function plotted in each figure corresponds to the binomial distribution $B(32, 0.5)$ obtained in the case of unique response-bits.

As can be seen from these figures, only the inter-Hamming distances calculated from the differential pair responses match the ideal distribution for all four device types (Fig. 10.2a). In comparison, the inter-Hamming mass distributions of the other bit-generation mechanisms show noticeable deviations from the ideal graph for response data of the oscillator and sensor devices.

The graphical results are quantified using the Chi-squared test (Sec. 3.2.3) in order to formally assess whether the observed deviations from the binomial distribution $B(32, 0.5)$ are statistically significant. The test statistics and their respective p-values are reported in Tab. 10.2.

As indicated by the extremely low p-values, the inter-Hamming distributions computed for the oscillator and sensor primitives differed significantly from the assumed binomial distribution in all test cases but the ones evaluated for the proposed differential PUF concept. The tests statistics computed from the transistor and resistor responses, on the other hand, did not reveal significant deviations from the binomial distribution $B(32, 0.5)$ at a significance level of 0.01 in the case of the distiller and normalization procedure. In these instances, observed discrepancies between the empirical and ideal mass functions are most likely attributed to the sample variance of the response data used to calculate the respective empirical distributions. Naturally, the displayed variations are more pronounced for the smaller datasets of the transistor and resistor PUF populations including only 68 and 50 instances, respectively. In comparison, the masking approach failed the test for each device type most likely due to the overall large relative bias of the response-bits as was shown in Sec. 10.1.

Taking into account the experimental results of the previous section it is ev-

ident that the inter-Hamming metric is only loosely related to the randomness of a PUF-based entropy source, i.e. a high degree of uniqueness does not imply randomly generated response-bits. The existence of strongly biased response-bits is not accurately reflected by the pairwise inter-Hamming distance, if the systematic mismatch between devices of different PUF instances varies (Sec. 6.2.2). In the case of the transistor and resistor designs, for example, no common trend in the spatial mismatch between the analyzed PUF devices was found. Therefore, the overall large number of biased pairs had little effect on the response uniqueness, especially due to the moderate number of response samples. The oscillator and sensor devices, in contrast, showed a systematic pattern in their spatial mismatch distributions. Due to the consistency in the trend pattern between the PUF instances, the response-bias failed to cancel out in the computation process of the pairwise Hamming distance, hence lowering the response uniqueness. The decrease in the response uniqueness is particularly visible for the inter-Hamming distributions of the ring oscillator PUF in Fig. 10.2b, 10.2c and 10.2e. Only the differential PUF concept and the normalization procedure could mitigate the effect of global systematic trends. The former concept locally compensates the device mismatch of individual PUF instances, whereas the latter method subtracts the lumped global bias averaged over all PUF instance from each response-bit. By construction, the normalization approach only works if PUF devices taken from different wafer and production batches are subjected to the same global bias. Since the overall mismatch between the sensor pixels varies between wafers and batches, the normalization approach only yields suboptimal results regarding the response uniqueness when compared to the differential quantization scheme.

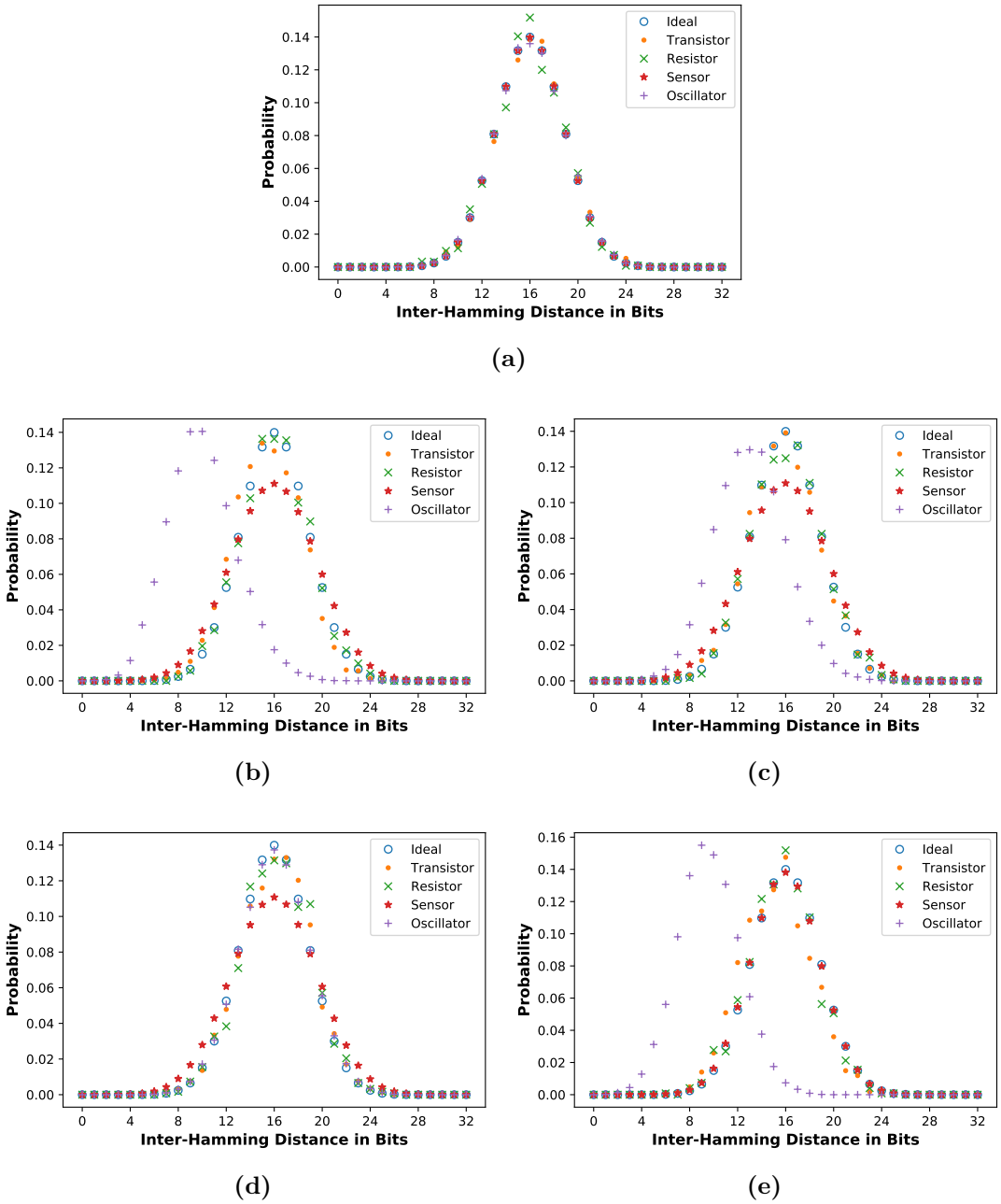


Figure 10.2: Inter-Hamming distance probability mass functions obtained for (a) the proposed Weak PUF concept implementation, (b) the column-wise pair selection scheme, (c) the regression-distiller method, (d) the normalization-based bias removal procedure and (e) the 1-out-of-8 masking approach.

Table 10.2: Chi-squared test results of the empirical inter-Hamming metrics.

Concept	Transistor		Resistor		Sensor		Oscillator	
	Q-Value	p-Value	Q-Value	p-Value	Q-Value	p-Value	Q-Value	p-Value
3. Order Differences	23.19	0.87	23.16	0.87	30.13	0.56	26.32	0.75
Column-wise Pair Selection	105.36	0.00	11.68	1.00	1576376.33	0.00	1676265.75	0.00
Regression Distiller	31.34	0.50	16.85	0.99	1599622.61	0.00	42580.45	0.00
Normalization Approach	23.67	0.86	26.12	0.76	1612149.69	0.00	236.29	0.00
1-out-of-8 Masking	217.52	0.00	37.30	0.24	5191.99	0.00	2109221.71	0.00

10.3 Reproducibility

The stability of the PUF-based key generators was tested for response data measured under standard conditions and for averaged parameter samples acquired at varying operating settings. In both cases, 32-bit responses obtained from multiple PUF instances were used to evaluate the respective distributions of the empirical intra-Hamming distances.

10.3.1 Noise

The noise-related intra-Hamming distance distributions were computed from up to 100 response samples per PUF instance and device type. The number of PUF instances used to compute the intra-Hamming metric are listed in Tab. 10.3. The empirical mass functions obtained for the various device types and PUF concepts are visualized in Fig. 10.3a to 10.3e. The corresponding mean and maximum[1] bit-error values are summarized in Tab. 10.4.

As demonstrated by the results, the highest reproducibility of the generated PUF responses could be achieved for the third order differential pair selection scheme (Fig. 10.3a) and the 1-out-of-8 masking approach (Fig. 10.3e). The empirical mass functions of both concepts peaked at small intra-Hamming distances, indicating a high degree of noise resilience of the respective response mechanisms. The average number of bit-errors were in the range of 0 to 0.6 bit and thus noticeably smaller than the error values obtained for the column-wise pair selection scheme, the distiller method and the normalization procedure. Furthermore, application of the differential and masking quantization scheme resulted in the overall smallest maximum number of bit-errors. Depending on the tested device, the maximum number of bit-errors fell in the range of 0 to 12 bit out of 32 response-bits. Compared to the other three concepts, the maximum number of bit-errors was found to be 1 to 3 bit smaller on average.

In case of the differential quantization scheme, the robustness of the PUF responses can be further enhanced by increasing the pair order from which response-bits are generated. The dependence of the mean and maximum number of bit-errors on the pair order is depicted in Fig. 10.4a and 10.4b, respectively.

[1] As stated in previous sections, the maximum number of bit-errors was defined as the 99-percentile of the intra-Hamming distribution.

First order pairs only remove the systematic bias of the response-bits and do not provide any form of noise reduction, hence the likeliness of bit-errors to occur is larger than for higher order differences. This example illustrates the relationship between the bias of response-bits and their susceptibility to measurement noise (Sec. 6.3.1), i.e. the smaller the bias the higher the number of expected bit-errors. By increasing the pair order, the measured pair differences are shifted away from the quantization threshold, hence effectively lowering the mean and maximum number of bit-errors. Error-free regeneration of the transistor, sensor and oscillator responses was possible at minimum pair orders around 3 and 4. In case of the highly noise-sensitive resistor design more than six pair orders were required to ensure full reproducibility of the PUF responses. Since the effect of measurement noise can be reduced by averaging over multiple samples, it is advised to choose a lower pair order and increase the integration time of the read-out circuit in order to minimize the number of PUF devices needed to generate a sufficient number of response-bits.

Table 10.3: Overview over the number of PUF instances used to compute the noise-related intra-Hamming distributions.

Device Type	Number of PUF Instances
Transistor	58
Resistor	28
Sensor	3926
Oscillator	193

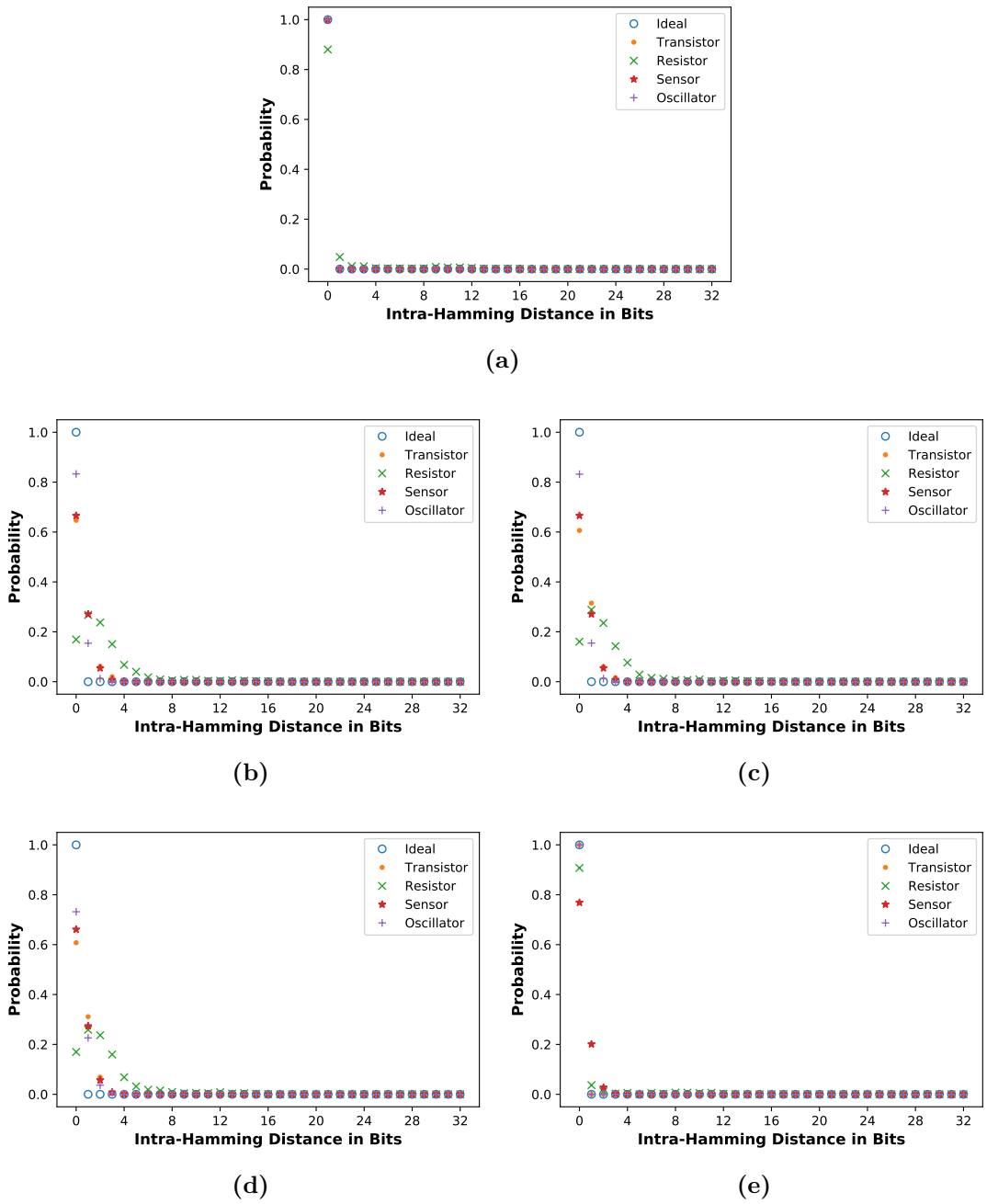
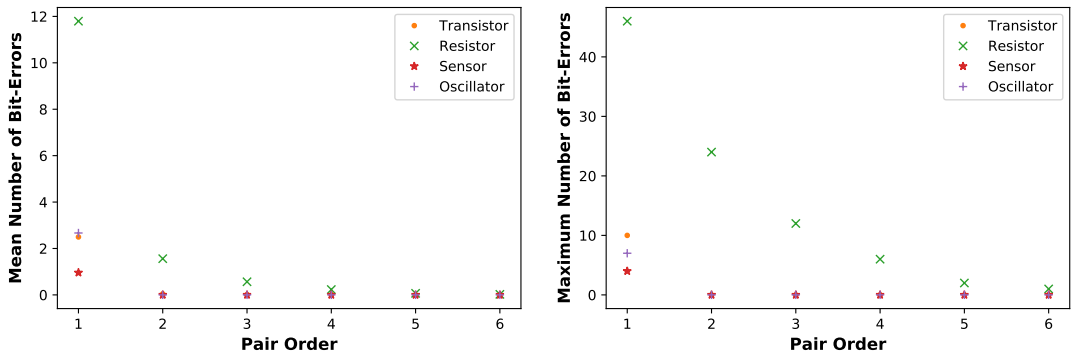


Figure 10.3: Noise-related intra-Hamming distance probability mass functions obtained for (a) the proposed Weak PUF concept implementation, (b) the column-wise pair selection scheme, (c) the regression-distiller method, (d) the normalization-based bias removal procedure and (e) the 1-out-of-8 masking approach.

Table 10.4: Mean and maximum number of bit-errors computed for the intra-Hamming distributions due to measurement noise.

Concept	Transistor		Resistor		Sensor		Oscillator	
	Mean	Max.	Mean	Max.	Mean	Max.	Mean	Max.
3. Order Differences	0.0160	0	0.5662	12	0.0015	0	0	0
Column-wise Pair Selection	0.4576	3	2.2875	13	0.4086	2	0.1797	2
Regression Distiller	0.4945	3	2.2647	13	0.4086	2	0.1814	2
Normalization Approach	0.4908	3	2.3132	13	0.4145	2	0.3161	2
1-out-of-8 Masking	0	0	0.4162	10	0.2647	2	0	0



(a) (b)

Figure 10.4: Dependence of the (a) mean and (b) maximum number of noise-induced bit-errors on the pair difference order.

10.3.2 Operating Conditions

The intra-Hamming distance distributions were determined for PUF responses generated from averaged parameter samples measured under different operating conditions. A summary of the analyzed datasets is given in Tab. 10.5. The intra-Hamming distances computed for stability-enhanced third order pairs are displayed in Fig. 10.5a. The corresponding mass functions of the other four concepts are shown in Fig. 10.5b to 10.5e. An overview over the related mean and maximum bit-error metrics is given Tab. 10.6.

Consistent with the analysis results presented in the previous sections, application of the differential quantization scheme and the masking approach generally yielded a lower average and a lower maximum number of bit-errors than the other three concepts. The mean number of bit-errors was around five or more times smaller than the numbers obtained for the column-wise pair selection scheme, the distiller method and the normalization procedure. The maximum number of errors were often around 2 to 8 bit lower depending on the device type.

It is important to point out the limited error reduction capability of the masking approach regarding the sensor response data. The method barely improves on the average bit-error rate obtained for the column-wise quantization scheme, the distiller method and the normalization concept. The maximum number of bit-errors is identical to the three aforementioned approaches. This example

shows that the bit-masking does not necessarily warrant lower error probabilities, because the magnitude of the pair differences depends on the variability of the random realizations and the size of the subsets of devices from which pairs are formed. Since large pair differences are less likely to be observed in small subsets, large shifts in the parameter differences due to changes in the operating conditions can not be compensated. In contrast, the proposed differential quantization concept provides simple directives to systematically construct sets of pairs with large absolute differences from the totality of all devices comprising a PUF. Therefore, it is possible to increase the response stability by including more pairs in the computation process, whereas the masking approach needs to discard larger number of pairs to improve the response stability.

Similar to the results shown in the previous section, the occurrence of erroneous response-bits can be reduced by forming higher order pairs. The relation between the mean and maximum number of bit-errors and the pair order is illustrated in Fig. 10.6a and 10.6b, respectively. As can be seen from the figures, the transistor and oscillator PUFs require a minimum pair order of four to five to ensure full reproducibility of the responses. For the resistor and sensor designs, pair orders of six were sufficient to produce stable response-bits. Unlike noise-induced bit-flips, errors caused by variable operating conditions cannot be reduced by increasing the sample integration time. Consequently, redundancy in form of additional PUF devices that do not contribute to the overall response entropy is unavoidable, if error-free keys are required.

Table 10.5: Overview over the number of PUF instances used to compute the intra-Hamming distributions at variable operating conditions.

Device Type	Varied Operating Conditions	Number of Devices
Transistor	temperature and supply voltage	5
	temperature (fixed supply voltage)	47
Resistor	temperature and supply voltage	5
	temperature (fixed supply voltage)	23
Sensor	temperature and illumination	3926
Oscillator	temperature (fixed supply voltage)	5
	supply voltage (fixed temperature)	5

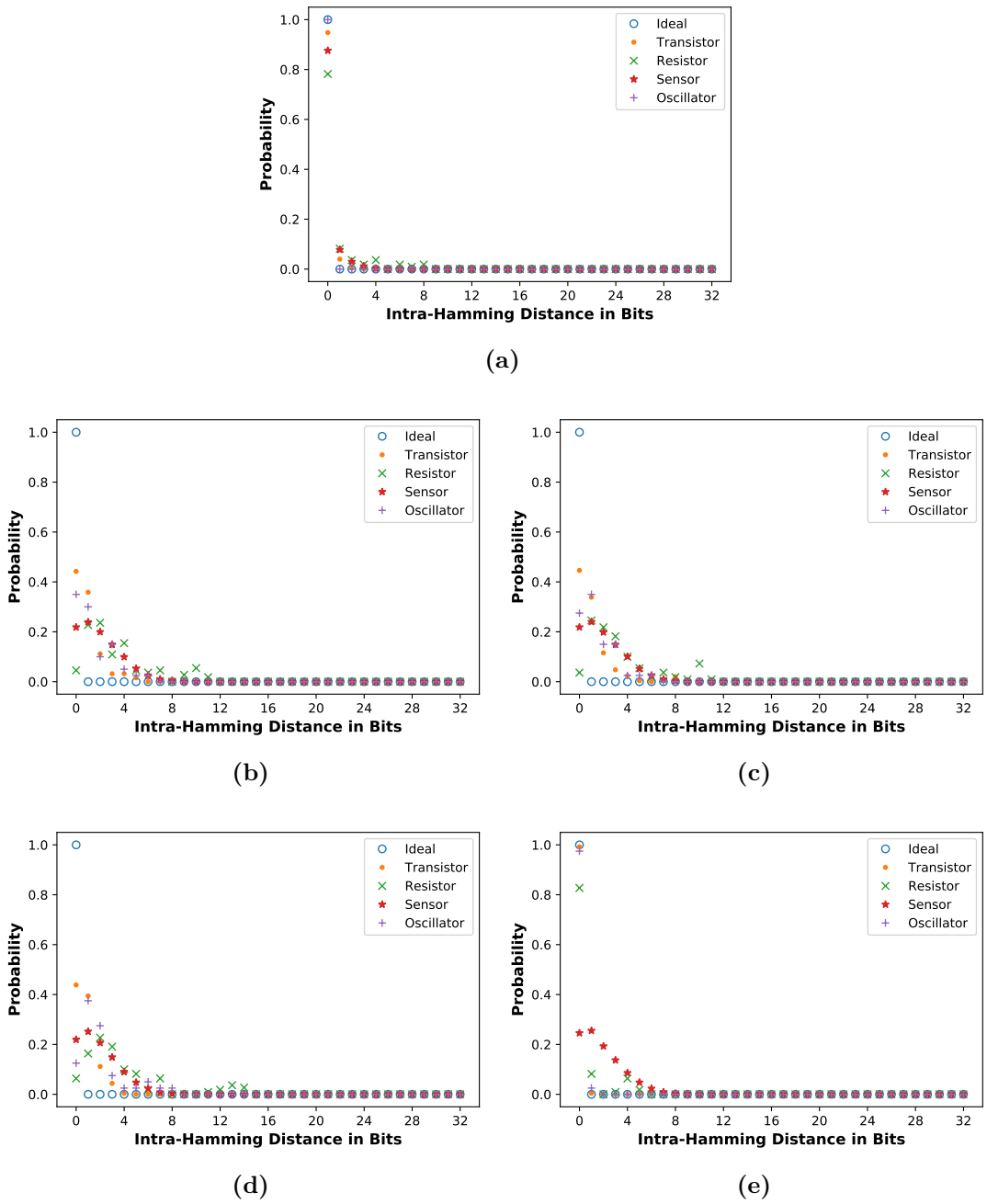
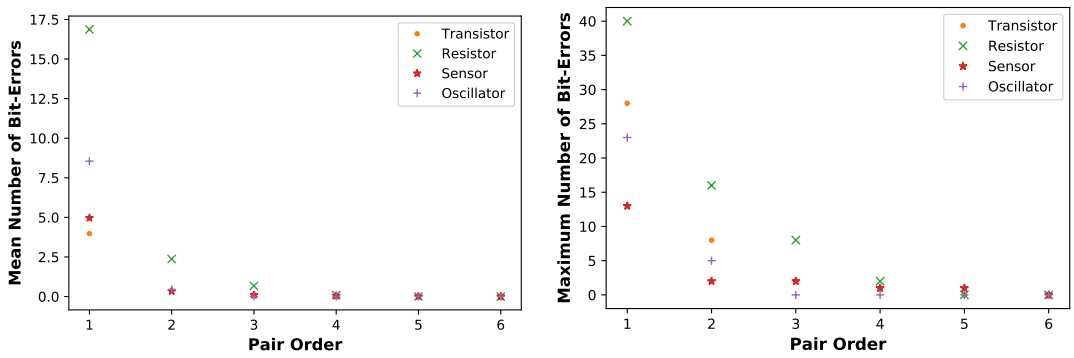


Figure 10.5: Intra-Hamming distance probability mass functions accumulated over various operating conditions for (a) the proposed Weak PUF concept implementation, (b) the column-wise pair selection scheme, (c) the regression-distiller method, (d) the normalization-based bias removal procedure and (e) the 1-out-of-8 masking approach.

Table 10.6: Mean and maximum number of bit-errors computed for the intra-Hamming distributions due to changes in the operating conditions.

Concept	Transistor		Resistor		Sensor		Oscillator	
	Mean	Max.	Mean	Max.	Mean	Max.	Mean	Max.
3. Order Differences	0.1195	2	0.6727	8	0.2030	3	0	0
Column-wise Pair Selection	0.9482	5	3.4000	11	2.0353	7	1.4250	6
Regression Distiller	1.0159	8	3.3182	10	2.0332	7	1.4750	6
Normalization Approach	0.8446	4	3.7273	14	1.9518	7	2.0500	8
1-out-of-8 Masking	0.0279	0	0.4545	5	1.8686	7	0.0250	1



(a) (b)

Figure 10.6: Dependence of the (a) mean and (b) maximum number of bit-errors due to variable operating conditions on the differential pair order.

10.4 Summary

The proposed Weak PUF concept was experimentally evaluated for four different types of primitives including transistors, resistors, sensors and ring oscillators. Application of a higher-order matching pair selection scheme produced random, unique and stable responses. All three properties were found to be independent of the device type and the processing technology.

The results were compared to the performance metrics calculated for a simple column-wise pair selection scheme and three state-of-the-art quantization methods, namely the regression distiller procedure, the normalization bias removal technique and a 1-out-of-8 masking scheme. The analysis showed that all four methods produced biased response-bits for all device types. Particularly noteworthy is the poor performance of the regression and normalization methods, which were specifically developed to eliminate systematic mismatch between PUF devices. In case of the sensor and oscillator PUFs, a severe reduction in the response uniqueness could be observed due to uncompensated bias in the generated response-bits.

With regard to the stability of the three tested state-of-the-art response mechanisms, only the error rates of the masking approach were found to be lower than those of the standard column-wise pair selection scheme. Although the masking approach generally improved upon the reproducibility of the responses, the

performance was not consistent for all device types and noise sources. In case of the sensor response data acquired under variable operating conditions, no reduction in the number of the response errors could be achieved compared to the other three methods. In contrast, the newly developed differential quantization scheme was able to reproduce the response data of each device at all tested operating conditions by forming higher order pairs. The only difference between the device types was the minimum number of pairs required to achieve error-free regeneration of the PUF responses.

Chapter 11

Experimental Strong PUF Concept Validation

In this chapter, the applicability of the newly developed Strong PUF concept is experimentally validated. The results are compared to the basic Arbiter PUF (Sec. 2.5.1), the group-based CRP enhancement concept, the identity-mapping procedure and the k-sum PUF (see the respective subsections in Sec. 2.5.2).

The performance analysis of the Arbiter PUF was based on the response data provided by the authors of [Hori 10], who implemented 45 instances of a 64 stage Arbiter design on commercial FPGAs. The dataset contains 128×1024 randomly generated CRPs per PUF instance. In order to facilitate comparison with the sensor PUF design composed of only 64 device pairs usable for the response generation, each CRP included in the Arbiter dataset was limited to the first 64-bits.

The proposed Strong PUF concept, the group-based CRP enhancement approach, the identity-mapping procedure and the k-sum PUF were applied to the experimental data obtained from the transistor, resistor, sensor and ring oscillator devices presented in Chap. 9. In case of the proposed Strong PUF concept, 64-bit twisted chain functions (Sec. 8.1.2) were constructed from the response data of the differential Weak PUF realizations examined in Chap. 10. The individual chains were comprised of 64 coupled segments configurable by a 64-bit challenge. Each segment was built from two Weak PUF response-bits, thus implementing both chains required a total number of 256 response-bits. For a given challenge, the output of the first chain was taken as response-bit if both chains produced

the same bit-value, otherwise the response-bit was set equal to the output of the second chain. Each chain structure was evaluated for 64 randomly chosen challenges to generate a 64-bit response. The group-based, identity-mapping and k-sum PUF concepts were implemented according to the response generation algorithms described in their respective research paper[1]. The group-based and identity-mapping algorithms were used to generated CRPs composed of 64-bit long challenges and responses. Since the basic k-sum PUF produces only one response-bit per challenge, the output values obtained for 64 randomly selected challenges were concatenated to form a full response. The performance of the various Strong PUF implementations was evaluated with regard to their applicability in public challenge-response protocols – the main area of application of a Strong PUF. Therefore, the analysis focused on the uniqueness, challenge sensitivity and complexity of the measured CRP data. The reproducibility of the responses was not investigated because the occurrence of bit-errors can be suppressed by the use of suitable error correction procedures, e.g. the 1-out-of-K masking scheme or higher order pair differences, as was demonstrated in the previous chapter. The uniqueness of the CRPs was measured in terms of the inter-Hamming distance between responses of different PUFs for a fixed challenge and between the responses of a given PUF instance obtained for different challenges. In the following sections, the two inter-Hamming metrics are denoted by $HD_{Inter-PUF}$ and $HD_{Inter-Challenge}$, respectively. The sensitivity with which response values react to changes of individual challenge-bits was analyzed by means of the SAC (Sec. 2.3.4). The complexity of the CRP generation mechanisms was evaluated within the scope of the PAC model framework (Sec. 2.3.5). Lastly, it shall be noted that the CRPs obtained for all concepts but the Arbiter PUF were calculated off-chip, as the processing logic needed to fully implement the designs in hardware was not available at the time of writing.

[1]References are given in Sec. 2.5.2.

11.1 Uniqueness

11.1.1 Inter-Hamming Distance between PUFs

The inter-Hamming metric $HD_{Inter-PUF}$ of the different PUF concepts was evaluated for 10000 CRPs acquired from multiple PUF instances realized for each semiconductor device type. Table 11.1 provides an overview over the number of tested devices. The resulting empirical inter-Hamming distributions $HD_{Inter-PUF}$ are presented in Fig. 11.1. Figure 11.1a shows the distribution obtained for the new Strong PUF concept. The corresponding distributions computed for the group-based CRP enhancement approach, the identity-mapping procedure, the k-sum PUF and the Arbiter PUF are displayed in Fig. 11.1b to 11.1e, respectively. The plotted binomial distribution $B(64, 0.5)$ serves as benchmark to qualitatively assess the uniqueness of the CRPs.

In the case of the proposed PUF concept, the group-based CRP mechanism and the identity-mapping approach no visible difference from the graph of the ideal distribution could be observed. In contrast, the corresponding distributions obtained for the other two design types showed marked deviations from the binomial distribution: the mass functions were either shifted in their center location or broadened with respect to the scale of the ideal distribution. Particularly noteworthy is the discrepancy between the graphs obtained for the k-sum PUF which failed to match the reference curve for any of the tested implementations. Despite subpar results, the k-sum PUF clearly outperformed the Arbiter design. The CRPs produced by the Arbiter PUF varied only in a small fraction of response-bits, i.e. the responses of different PUF instances generated for a given challenge were nearly identical. Since all challenges were randomly generated, the low degree of uniqueness exhibited by the CRPs could be attributed to a severe bias of the responses.

As in the previous chapter, the Chi-squared test was used to measure the goodness of fit between the empirical and the ideal inter-Hamming distribution. The test results are listed in Tab. 11.2. In accordance with the analysis of the distribution graphs, the Arbiter PUF and the k-sum PUF completely failed the test at a significance level of 0.01. In the case of the proposed Strong PUF concept, the null hypothesis could not be rejected for any of the device types. Surprisingly, the group-based approach and the identity-mapping procedure did not pass the

test for some device implementations despite matching distribution functions. In particular, they failed the test for the sensor-based PUF primitive featuring the overall largest sample population of all device types. The excessively high test statistics were caused by the large number of pairwise comparisons between PUF responses ($> 10^6$) effectively inflating the observed differences between the mass functions. Therefore, the test results were highly sensitive even to small deviations[2] from the ideal distribution. The question remains, whether a failed test result is of practical relevance regarding possible attack vectors given that the empirical and ideal inter-Hamming distributions closely match.

Table 11.1: Overview over the number of PUF instances and CRPs used to compute $HD_{Inter-PUF}$ and $HD_{Inter-Challenge}$.

Device Type	Number of Devices	CRPs
Transistor	68	10000
Resistor	50	10000
Sensor	7436	10000
Oscillator	100	10000

[2]The relative absolute differences between the distributions were around ~ 0.01 in magnitude.

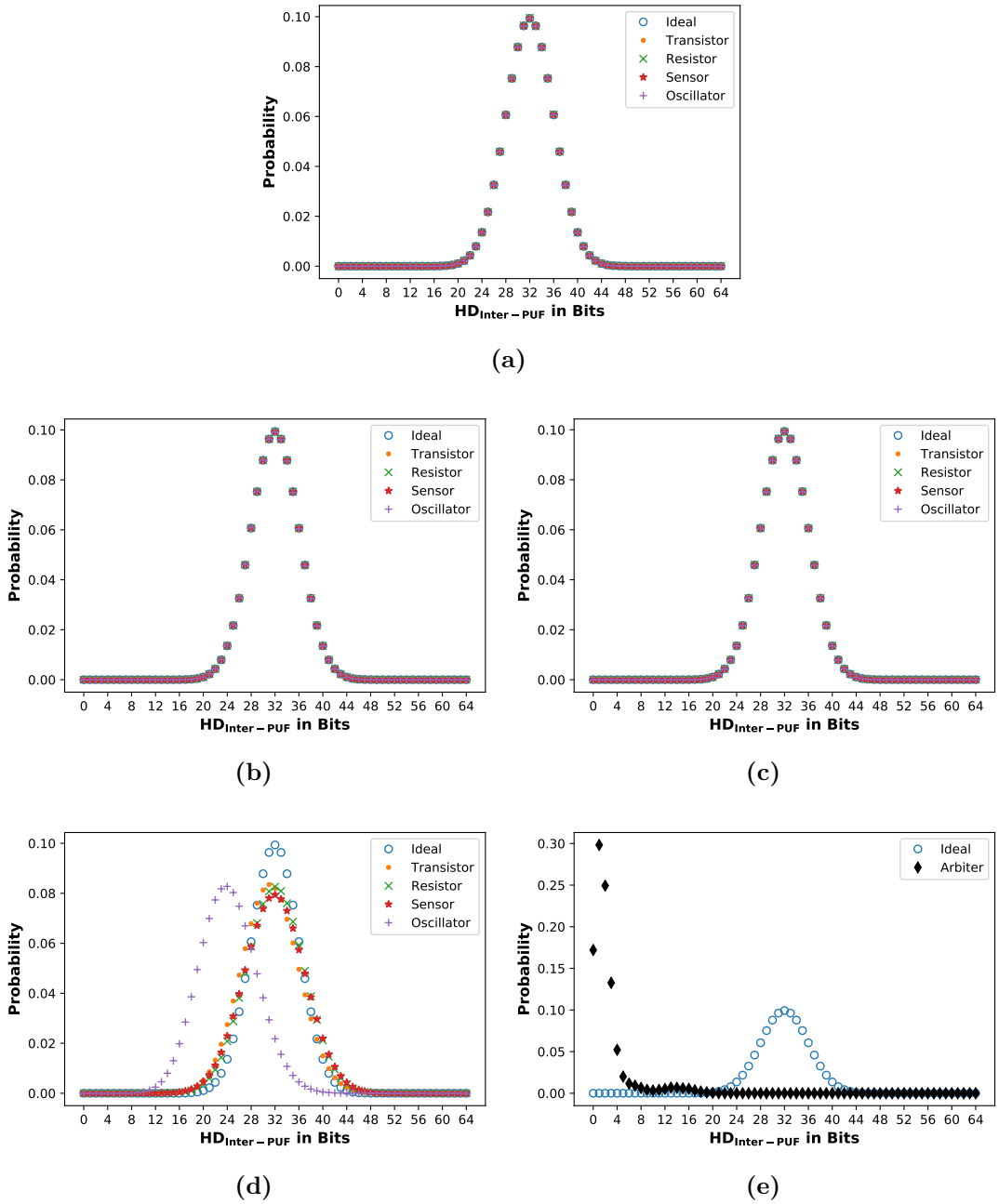


Figure 11.1: Inter-Hamming distance $HD_{Inter-PUF}$ probability mass functions obtained for (a) the proposed Strong PUF concept, (b) the group-based concept, (c) the identity-mapping concept, (d) the k-sum PUF and (e) the Arbiter PUF.

Table 11.2: Chi-squared test results of the empirical inter-Hamming metrics $HD_{Inter-PUF}$.

Concept	Transistor		Resistor		Sensor		Oscillator		Arbiter	
	Q-Value	p-Value	Q-Value	p-Value	Q-Value	p-Value	Q-Value	p-Value	Q-Value	p-Value
Twisted Chains	28.12	1.00	40.74	0.99	67.68	0.35	75.15	0.16	n.a	n.a
Group-Based	47.78	0.94	29.56	1.00	4.60×10^5	0.00	4.93×10^3	0.00	n.a	n.a
Identity-Mapping	64.16	0.47	41.60	0.99	44916.05	0.00	74.20	0.18	n.a	n.a
k-Sum	4.20×10^6	0.00	1.32×10^6	0.00	4.07×10^{15}	0.00	6.38×10^{10}	0.00	n.a	n.a
Arbiter	n.a	n.a	n.a	n.a	n.a	n.a	n.a	n.a	5.80×10^{23}	0.00

11.1.2 Inter-Hamming Distance between Challenges

The inter-Hamming distances $HD_{Inter-Challenge}$ were calculated from the same CRP datasets (Tab. 11.1) used to evaluate $HD_{Inter-PUF}$ in the previous section. The empirical mass functions obtained for the different PUF concepts and device types are plotted in Fig. 11.2a to 11.2e. The ideal graph drawn in each figure represents the probability mass function of the binomial distribution $B(64, 0.5)$ as before.

It is evident from the plots that the newly developed chain PUF, the group-based concept and the identity-mapping approach produce highly unique CRPs as implied by the matching empirical and ideal graphs. Taking into account the results shown in the previous section, all three concepts achieved a consistently high performance with respect to the inter-Hamming metrics $HD_{Inter-PUF}$ and $HD_{Inter-Challenge}$.

In contrast, the CRPs generated by the k-sum PUF only met the $HD_{Inter-Challenge}$ criterion, whereas the uniqueness regarding the $HD_{Inter-PUF}$ metric was less than optimal. The discrepancy between the results can be explained in terms of systematic inter-chip process variations causing similar device mismatch between different PUF instances. Consequently, responses extracted from two or more PUF instances can appear more similar for a given challenge than CRPs generated by a single PUF instance. This example emphasizes the necessity to test the inter-Hamming measure for a large number of PUF instances and CRPs, since performance test applied to small-sized samples might not accurately reflect the properties of the underlying population of PUF instances.

The Arbiter PUF design generated the least unique CRPs of all tested concepts. The empirical distributions of the inter-Hamming distances $HD_{Inter-Challenge}$ visibly differed from the ideal mass function. The poor performance of the Arbiter PUF indicates that the design implementation suffers from both systematic intra- and inter-chip mismatch. As a result, CRPs extracted from the same or different PUF instances only feature a small fraction of response-bits unique to each challenge and PUF.

Quantitative evaluation of the similarity between the empirical and ideal distributions was based on the Chi-squared test as in the previous section. The results are summarized in Tab. 11.3. Out of the five Strong PUF concepts, only the k-sum PUF passed all tests. The Arbiter PUF design failed the test confirming the

results of the graphical analysis. The twisted chain design, the group-based concept and the identity-mapping procedure, in comparison, failed the test for two to three out of the four implementation forms. Since the respective inter-Hamming distributions graphically matched the ideal binomial probability function, it remains questionable, if the test results have practical ramifications on the security and applicability of the aforementioned concepts.

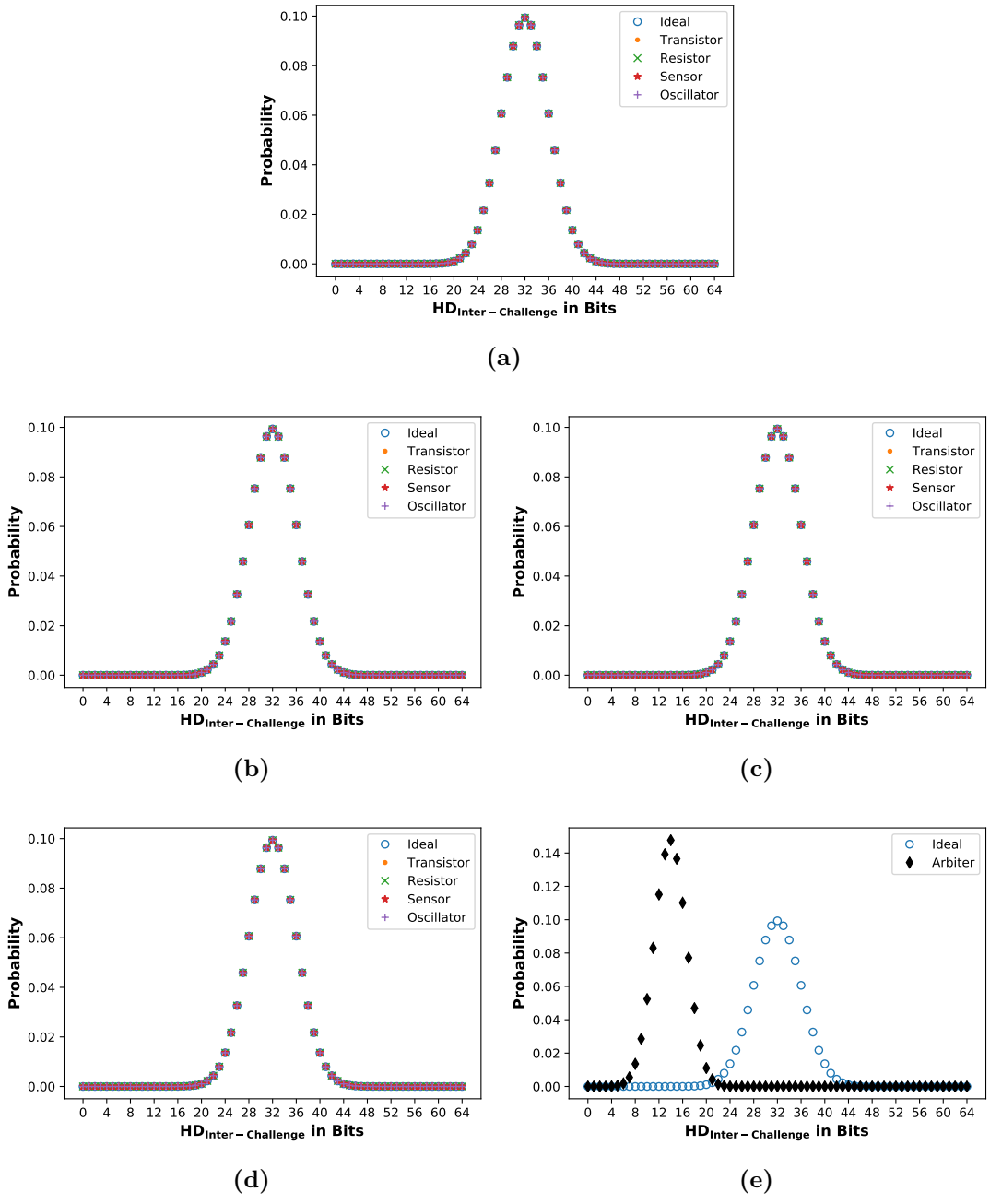


Figure 11.2: Inter-Hamming distance $HD_{Inter-Challenge}$ probability mass functions obtained for (a) the proposed Strong PUF concept, (b) the group-based concept, (c) the identity-mapping concept, (d) the k-sum PUF and (e) the Arbiter PUF.

Table 11.3: Chi-squared test results of the empirical inter-Hamming metrics $HD_{Inter-Challenge}$.

Concept	Transistor		Resistor		Sensor		Oscillator		Arbiter	
	Q-Value	p-Value	Q-Value	p-Value	Q-Value	p-Value	Q-Value	p-Value	Q-Value	p-Value
Twisted Chains	41.05	0.99	42.02	0.98	2.62×10^{10}	0.00	2.62×10^{10}	0.00	n.a	n.a
Group-Based	276.23	0.00	1.36×10^{4}	0.00	2.44×10^{5}	0.00	48.24	0.93	n.a	n.a
Identity-Mapping	120.85	0.00	58	0.68	1773.11	0.00	58.33	0.68	n.a	n.a
k-Sum	57.48	0.70	49.07	0.92	62.31	0.54	52.25	0.85	n.a	n.a
Arbiter	n.a	n.a	n.a	n.a	n.a	n.a	n.a	n.a	1.09×10^{14}	0.00

11.2 Challenge-Response Sensitivity

The challenge-response sensitivity was evaluated for the different implementation forms of the newly developed Strong PUF concept, the group-based approach, the identity-mapping procedure and the k-Sum PUF. The dataset of the Arbiter PUF did not provided sufficient CRPs, hence the design could not be tested. However, since the response mechanism is equivalent to that of the k-sum PUF, analysis results obtained for the k-sum design are transferable to the Arbiter PUF.

Each PUF concept was evaluated for 10000 random CRPs extracted from a maximum number of 50 PUF instances per semiconductor device type. For each PUF instance and set of CRPs, the dependence matrix was calculated according to the definition given in Eq. (2.9). The distribution of the elements of the dependence matrix, i.e. the estimated transition probabilities, computed for the different Strong PUF concepts and device types are displayed in Fig. 11.3a to 11.3d.

In the case of the twisted chain PUF design and the identity mapping procedure, the distributions of the response transition probabilities were tightly centered around the ideal value of 0.5, thus both concepts fulfilled the SAC (Sec. 2.3.4) on average for all tested device types. In comparison to the aforementioned concepts, the CRP behavior of the group-based approach reacted less sensitive to small modifications of the applied challenges as indicated by the notable spread in the observed transition probabilities.

The transition probabilities were overall lowest for the k-sum PUF as can be clearly seen from Fig. 11.3d. Due to the linear dependence of the response on the challenge-bits and the internal parameters of the PUF device, inverting only a single challenge-bit has little to no effect on the response value for any given challenge. This low challenge sensitivity is common to all linear Arbiter-like Strong PUF systems as was shown by the authors of [Majz 08a]. In order to achieve a transition probability of 0.5 half the number of challenge-bits needs to be flipped on average. Consequently, neither the k-sum PUF nor the Arbiter PUF meet the SAC. In order to increase the CRP sensitivity of these designs, input- and interconnected networks similar to the lightweight secure PUF (Sec. 2.5.1) might be added to the basic designs in order to propagate changes to the challenge to multiple entry points throughout the response generation mechanism.

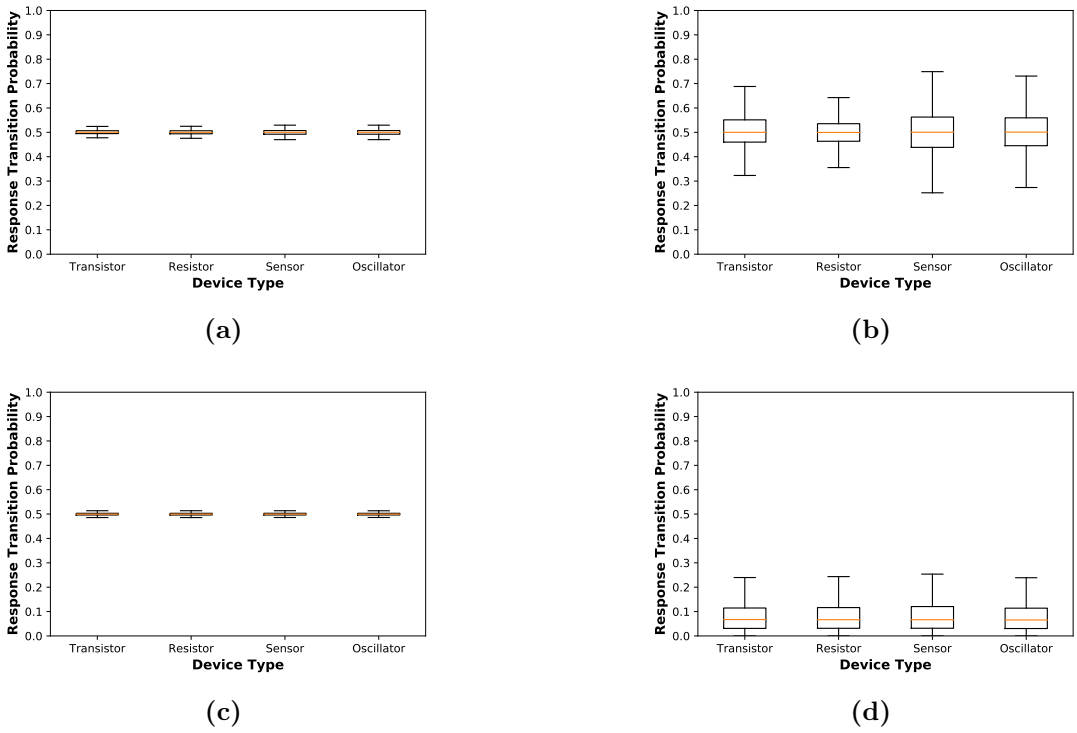


Figure 11.3: Transition probabilities of the response-bits computed for (a) the proposed Strong PUF concept, (b) the group-based concept, (c) the identity-mapping concept and (d) the k-sum PUF.

11.3 Complexity of the Challenge-Response Generation

In this section the challenge-response complexity of the proposed twisted chain PUF is determined and compared to the other PUF concepts.

The subsequent analysis assumes a twisted chain PUF implementation composed of $\tilde{l}$ coupled segments per chain, where the state of each chain segment is set by one challenge-bit. The set of recursive formulas defining the response values generated by the two chains (Eq. (8.5) and (8.6)) can be expressed as ring-sum-expansion[3]:

$$R_i^1 = R_{i-1}^1 \oplus S_i^1 \oplus R_{i-1}^1 S_i^1 \oplus R_{i-1}^1 S_i^2 \oplus R_{i-2}^2 S_i^1 \oplus R_{i-2}^2 S_i^2 \tag{11.1}$$

$$R_i^2 = R_{i-1}^2 \oplus S_i^2 \oplus R_{i-1}^2 S_i^2 \oplus R_{i-1}^2 S_i^2 \oplus R_{i-2}^1 S_i^1 \oplus R_{i-2}^1 S_i^1 \,. \tag{11.2}$$

S_i^j denotes the parity of the i-th segment of chain j as defined in Eq. (8.7). Recursively substituting R_i^1 and R_i^2 in the two equations yields terms such as $S_1^{1/2} S_2^{1/2} S_3^{1/2} ... S_k^{1/2}$ including the parity values of up to $\tilde{l}$ chain segments for a fixed challenge. It follows from the proof given in [Fisc 92] that the number of CRPs m_{Chain} needed to emulate the response mechanism of the twisted chain PUF is in the order of

$$m_{Chain}\left(\tilde{l}\right) = \mathcal{O}\left(\tilde{l}^{\tilde{l}}\right) \,. \tag{11.3}$$

Consequently, the complexity of the response mechanism scales exponentially with the number of chain segments $\tilde{l}$.

In case of the other four tested PUF concepts, only the complexity of the Arbiter PUF and the k-sum PUF were reported in the literature. The number of training samples $m_{Arbiter-like}$ required to model the CRP behavior of the two designs is in the order

$$m_{Arbiter-like}(n, l) = \mathcal{O}\left((n+1)^l\right) \,, \tag{11.4}$$

where n denotes the number of configurable stages of the design and l is equal to the number of PUF instances whose output values are XORed to generate a response-bit [Ganj 15]. As can be seen from the equation, the number of CRPs

[3]The ring-sum-expansion of a boolean functions is a form of representation that contains only unnegated boolean variables connected by logical XOR and AND operations.

depends linearly on the number of stages n and increases exponentially with the number of XORed response-bits. In the case of the tested Arbiter PUF and k-sum design, $n = 64$ and $l = 1$ holds.

A comparison of the results is given in Tab. 11.4. Since the complexity of the basic Arbiter and k-sum PUF is linear in the number of system parameters, model building attacks are always feasible due to practical limitations on the area and power consumption of a PUF design. Consequently, the response-bits of multiple PUF instances must be XORed in order to prevent model attempts. As stated by the authors of [Ganj 15], the maximum number of XORed Arbiter PUFs fabricated in 65 nm technology should be around 12 as the noise-induced error probability approaches 0.5. Motivated by this example 12 XORed 64-bit k-sum PUFs built from 64 primitive pairs per instance are considered here. The construction of the design requires at least $N_{12-XOR} = 12 \times 64 = 768$ primitive pairs. The number of CRPs needed to break the PUF is in the order of $m_{12-XOR}(64, 12) \sim 5.688 \times 10^{21}$.

In comparison, a twisted chain design made up of 74 2-bit segments takes only $N_{Chain} = 74 \times 2 \times 2 = 296$ primitive pairs to implement and withstands modeling attacks up to $m_{Chain}(74) \sim 2.104 \times 10^{138}$ CRPs. These sample calculations demonstrate that the response complexity of the proposed concept implementation exceeds the corresponding value of a 12-XOR k-sum PUF design by orders of magnitude while reducing the number of required primitive pairs to construct the PUF by more than 60 %.

Table 11.4: Complexity of the challenge-response mechanism of different Strong PUF concepts.

Device/Concept	Number of Training CRPs
Twisted Chains	$\mathcal{O}\left(\tilde{l}^{\tilde{l}}\right)$
Group-Based	n.a.
Identity-Mapping	n.a.
Arbiter / k-Sum [Ganj 15]	$\mathcal{O}\left(n+1\right)$
Arbiter / k-Sum (l-XOR) [Ganj 15]	$\mathcal{O}\left((n+1)^l\right)$

11.4 Summary

Five Strong PUF concepts were analyzed with regard to their applicability for basic public challenge-response protocols. Within the scope of the presented analysis, the challenge-response properties of the newly proposed chain PUF concept, the group-based CRP enhancement approach, the identity-mapping procedure, the k-sum PUF and the basic Arbiter PUF were investigated. All concepts with the exception of the Arbiter PUF were tested for different realizations including transistor, resistor, sensor and ring oscillator primitives.

The performance of the concepts was evaluated based on the uniqueness, challenge-response sensitivity and complexity of experimentally determined CRPs.

In the case of the inter-Hamming metrics, the best results were obtained for the chain PUF, the group-based concept and the identity-mapping procedure; all three concepts produced highly unique responses. The CRPs generated by the k-sum and Arbiter designs revealed either instance- or challenge-specific response patterns, effectively lowering the ability to differentiate the digital signatures extracted from multiple PUFs. In particular, the performance results of the k-sum PUF showed that a high average inter-Hamming distance between CRPs obtained from individual PUFs does not imply a high uniqueness of responses generated from different PUF instances given a specific challenge[4].

The challenge-response sensitivity was quantified by means of the SAC. From all tested concepts, only the chain PUF and the group-based approach satisfied the SAC on average, indicating a strong dependence on the applied challenges. The CRPs generated by the identity-mapping concept and the k-sum PUF responded only weakly to changes of the challenge-bits. Therefore, the latter two concepts are more prone to differential cryptanalysis, as attackers can identify dominant challenge-bits by observing transitions in the responses when successively complementing all bits of a challenge. In order improve the sensitivity of the k-sum PUF, advanced design concepts similar to the feedforward and XOR Arbiter PUF might be implemented (Sec. 2.5.1).

Lastly, the complexity of the response mechanism of the different concepts was compared. Although any Strong PUF concept designed for the use in public key

[4]Therefore, inter-Hamming metrics reported for a small population of PUF instances should be interpreted with caution, e.g. the analysis results of the current mirror PUF listed in Tab. 2.2.

exchange protocols must be resistant against modeling attacks, the group-based approach and the identity-mapping function have yet to be analyzed in this regard. Therefore, these concepts might not be used in security-critical procedures and applications irrespective of the other performance results. In contrast, the complexity of the CRPs extracted from the chain PUF was proven to grow exponentially in the number of chain segments, rendering model attempts impractical in theory. Compared to a state-of-the-art 12-XOR k-sum PUF design composed of 64 stages, the complexity of the chain PUF was found to be orders of magnitude higher. At the same time, the concept required less than 40 % of the primitives pairs needed to implement the 12-XOR k-sum PUF design.

In conclusion, the analysis results showed that the proposed PUF concept fulfills all performance criteria of an ideal Strong PUF design independent of its implementation form.

Chapter 12

PUF Optimization

In this chapter, a generic optimization methodology is presented to complement the proposed Weak PUF concept by providing means to improve the layout design of PUF primitives and to select the ideal operating point at which responses should be extracted. Application of the methodology allows to determine the set of layout parameters for which the lowest relative systematic mismatch between PUF primitives is obtained at a given operating point. Conversely, the method can be used to compute the optimum bias point of any given PUF design at which the systematic mismatch between device pairs is dominated by random effects originating from stochastic process variations. To make full use of both options, the simultaneous optimization of layout and operating parameters is described in Sec. 12.1. As a simple application example, the procedure is used to calculate the ideal gate-length and operating settings of a transistor-based PUF primitive. The results of the optimization procedure are shown in Sec. 12.2.

12.1 Methodology

The response entropy extractable from a set of pair differences is closely related to the distribution parameters of the corresponding bias distribution of the primitive pairs (Sec. 6.1).

Due to spatiotemporal changes in the processing conditions, the device mismatch is likely to vary between dies, wafers and batches, hence it is reasonable to consider the two distribution parameters $(\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}})$ as realization of a latent stochastic process that governs said changes in the bias distributions. Thus,

to characterize the overall relative systematic mismatch of a PUF design implemented in a particular processing technology, it is necessary to evaluate the distributional properties of the distribution parameters.

In the optimization context, the location and scale of the distribution parameter distribution are of prime interest. The smaller the absolute location values of the distribution parameter distributions are, the sharper are the resulting bias distributions centered around zero bias. The scale determines the variability of the bias distributions of individual PUF instances, i.e. the scale parameter controls the degree to which bias distributions of a number of PUF instances differ from each other.

Within the scope of this work, the mean vector of the distribution parameter distribution is taken as location vector. The scale of the distribution is estimated by the covariance matrix of the two distribution parameters. The choice of location and scale is motivated by the intuitive meaning of the mean and covariance matrix in the case of bivariate Gaussian-like distributions, i.e. unimodal, elliptical distributions. If these assumptions hold, then the mean vector $[\mu_1, \mu_2]^{\mathrm{T}}$ indicates the center of the probability distribution and the two eigenvalues (λ_1, λ_2) of the covariance matrix determine the variance of the distribution parameters along the two axis of the elliptical distribution [Tong 90].

In order to ensure an overall high and consistent level of response entropy, the distribution parameter distribution should be sharply peaked around the zero vector. Equivalently, the four distribution parameters μ_1, μ_2, λ_1 and λ_2 should all be close to zero. Hence, the goal of the optimization procedure is to determine the set of layout and operating settings x that produce the lowest distribution parameter values. The functional relationship between x and the distribution parameters can be determined from the experimental data of different PUF layout variants measured at variable operating points. The dependence on x is modeled by means of standard regression analysis or interpolation techniques.

Let the relation between the four parameter values and x be expressed in terms of the respective functions $f_{\mu_1}(x)$, $f_{\mu_2}(x)$, $f_{\lambda_1}(x)$ and $f_{\lambda_2}(x)$, then a simple way to define the optimization objective function $f_{opt}(x)$ is to write $f_{opt}(x)$ as weighted sum of the four parameter functions, i.e.:

$$f_{opt}(x) = w_1 f_{\mu_1}(x) + w_2 f_{\mu_2}(x) + w_3 f_{\lambda_1}(x) + w_4 f_{\lambda_2}(x) \tag{12.1}$$

with $\sum_{i=1}^{4} w_i = 1$. Here, all components are considered equally important, thus

all weight factors are set to 0.25. Taking into account the constraints imposed by the design rules and operating limits of a PUF primitive, the optimization problem can be formulated as follows [Boyd 04]:

$$\underset{x}{\text{minimize}} \quad f_{opt}(x) \tag{12.2}$$

$$\text{subject to} \quad a_i(x) \leqslant 0, \quad i = 1, \ldots, M \tag{12.3}$$

$$b_j(x) = 0, \quad j = 1, \ldots, N \tag{12.4}$$

with inequality and equality constraints defined by the functions a_i and b_j, respectively. The optimization problem can be solved using standard algorithms readily implemented in computer languages such as Python[1].

12.2 Example: nMOS Transistor PUF

The proposed methodology is used to determine the operating point and gate-length of a transistor-based PUF primitive at which the relative systematic mismatch in the drain current of selected device pairs is minimized.

12.2.1 Dataset

The optimization procedure was applied to the electrical data of 31 PUF instances fabricated in a standard 350 nm CMOS bulk technology. Each PUF instance contained 100 identically designed nMOS transistors arranged in a 10×10 array. The transistor primitives of the 31 PUF instance were fabricated at 15 different nominal gate-lengths $L = 200\,\text{nm}, 220\,\text{nm}, ..., 480\,\text{nm}$. The 15×31 transistor arrays were measured on wafer-level to acquire the output characteristics of the individual transistors. The sweeping range of the gate voltage V_g and source-drain voltage V_{ds} was adjusted to the different gate-lengths to generate sufficiently large drain currents I_{ds} between 1 µA to 100 µA.

12.2.2 Parameter Estimation

In a next step, median polish was applied to the drain currents of each PUF instance recorded at the various parameter levels $x = (V_g, V_{ds}, L)$. The decom-

[1]The module 'scipy.optimize' of the Python package 'scipy' [Jone 01], for example, offers a number of built-in functions to solve constrained optimization problems.

position results were used to estimate the systematic current mismatch $\mu_{\Delta S}$ and the variance of the random components $\sigma^2_{\Delta S}$ as described in Sec. 5.3. From these estimates, the empirical bias distribution of 50 matching first order pairs per PUF instance were computed. The sample mean and standard deviation were taken as distribution parameters $(\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}})$ of the bias distributions.

The distribution of the distribution parameters obtained for two sets of layout configurations are shown in Fig. 12.1. As is apparent from the figure, noticeable

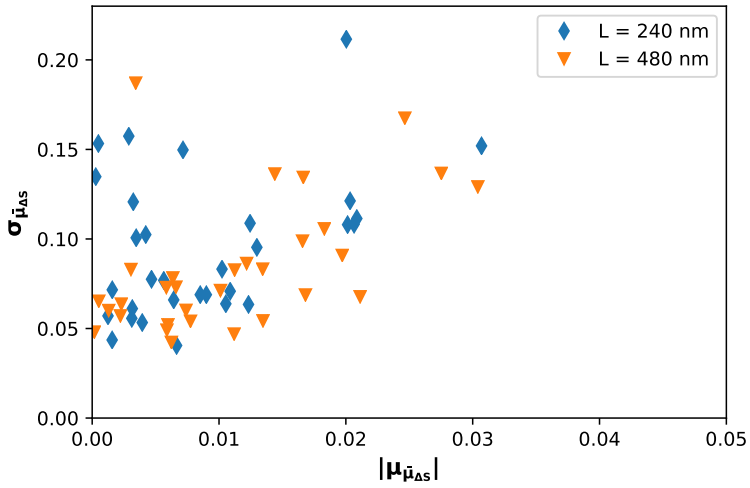


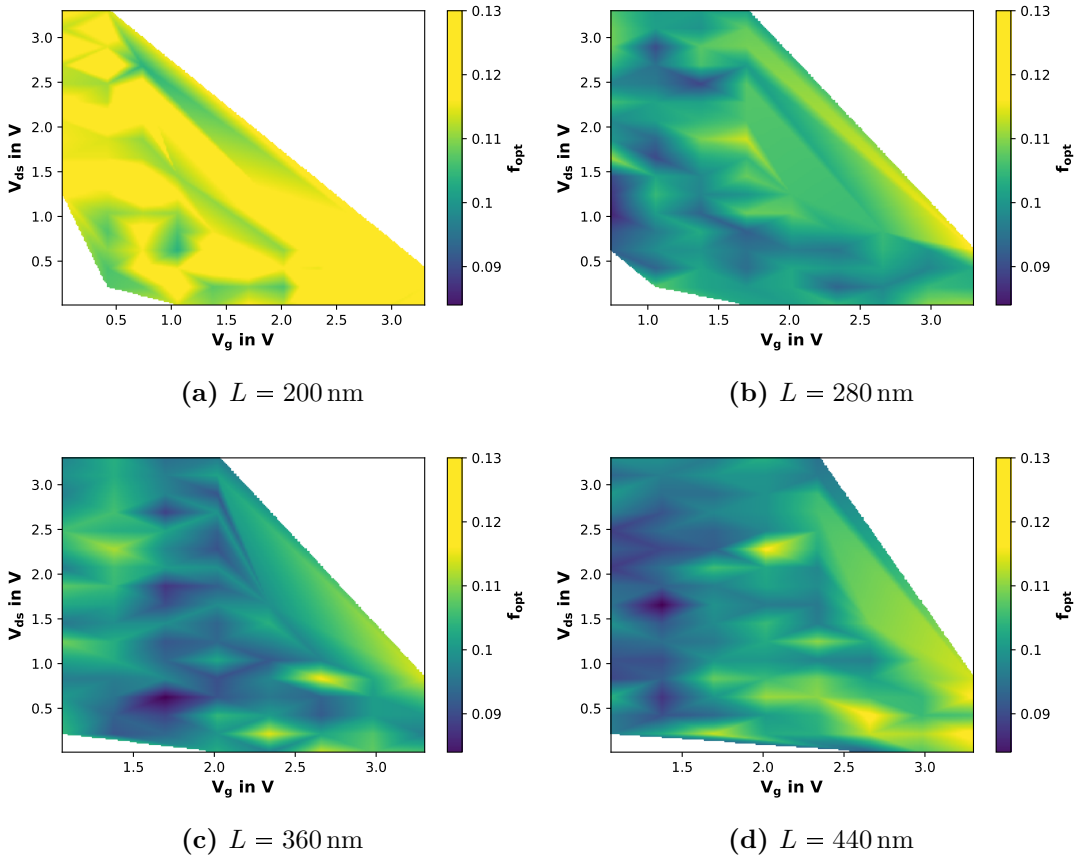
Figure 12.1: Scatter plot of the distribution parameters $(\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}})$ obtained for 2×31 PUF instances designed with nominal gate-lengths $L = 240\,\text{nm}$ and $L = 480\,\text{nm}$, respectively. The transistor primitives were operated at $V_g = 1.38\,\text{V}$ and $V_{ds} = 1.655\,\text{V}$.

spread in the absolute values of the distribution parameters can be observed. In order to obtain robust estimates of the location and covariance matrix of $|\mu_{\bar{\mu}_{\Delta S}}|$ and $\sigma_{\bar{\mu}_{\Delta S}}$, the Minimum Covariance Determinant algorithm [Hube 10] was applied to the square-root transform of the absolute mean and standard deviation values. Computation of the two eigenvalues λ_1 and λ_2 of the covariance matrix concluded the parameter estimation phase.

12.2.3 Optimization

The four distribution parameters μ_1, μ_2, λ_1 and λ_2 were computed for a total of 730 different combinations of V_g, V_{ds} and L. Projections of the corresponding objective function values $y_i = f_{opt}(V_g^i, V_{ds}^i, L^i)$, $i = 1, 2, ..., 730$, onto the V_g-V_{ds}

plane at four different gate-lengths are displayed in Fig. 12.2. It is apparent from



(a) $L = 200\,\text{nm}$ **(b)** $L = 280\,\text{nm}$

(c) $L = 360\,\text{nm}$ **(d)** $L = 440\,\text{nm}$

Figure 12.2: Objective function f_{opt} dependent on the operating parameters (V_g, V_{ds}) evaluated at fixed gate-lengths (a) $L = 200\,\text{nm}$, (b) $L = 280\,\text{nm}$, (c) $L = 360\,\text{nm}$ and (d) $L = 440\,\text{nm}$.

the figures that both the gate-length and the operating point of the devices have a noticeable effect on the graph of the objective function.

Due to the complex relationship between the distribution parameters and (V_g, V_{ds}, L), three dimensional linear interpolation was used to model $f_{\mu_1}(x)$, $f_{\mu_2}(x)$, $f_{\lambda_1}(x)$ and $f_{\lambda_2}(x)$. The resulting optimization problem to be solved can be written in the form:

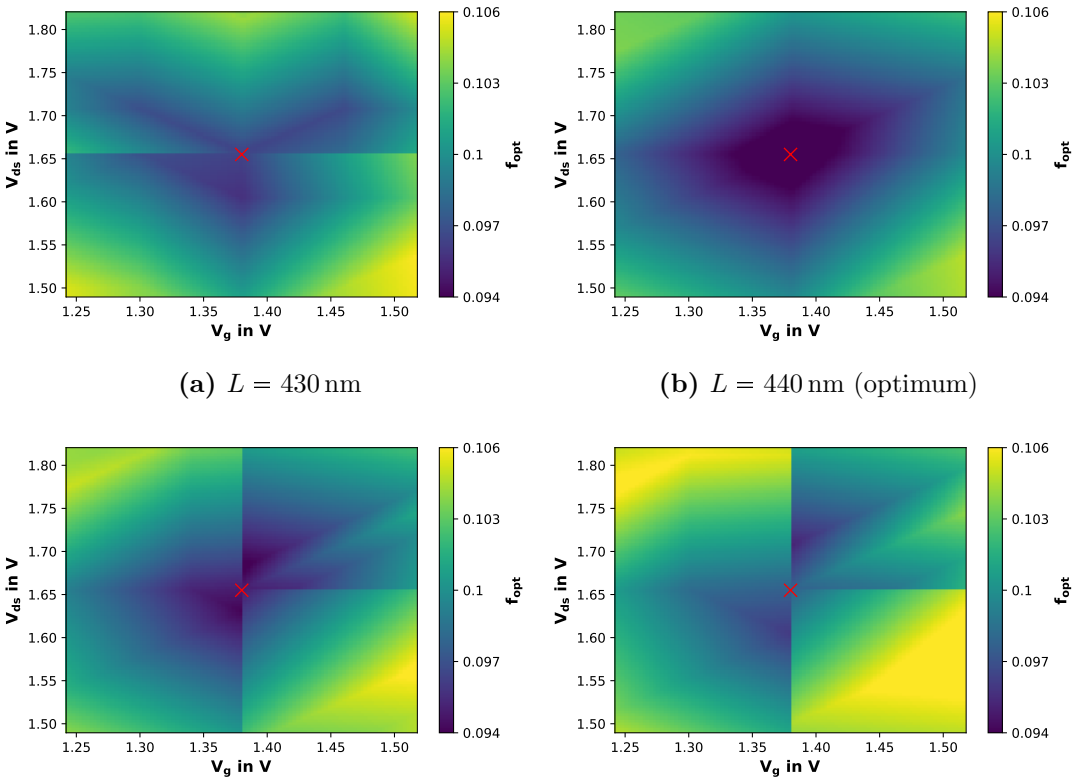
$$\underset{(V_g,V_{ds},L)}{\text{minimize}} \quad f_{opt}(V_g, V_{ds}, L) \tag{12.5}$$

$$\text{subject to} \quad 0\,\text{V} < V_g, V_{ds} \leqslant 3.3\,\text{V} \tag{12.6}$$

$$L \geqslant 200\,\text{nm} \tag{12.7}$$

Optimization of the objective function was repeated for 29 initial parameter combinations in order determine the parameter settings that globally minimize f_{opt}. The initial settings were obtained by adding uniformly distributed noise to all empirical parameter settings for which the objective function value was found to be less than 0.1.

The minimum objective function value of 0.092148 was attained at $V_g = 1.38\,\mathrm{V}$, $V_{ds} = 1.655\,\mathrm{V}$ and $L = 440\,\mathrm{nm}$. Two dimensional surface plots computed for parameter settings close to the optimal solution are shown in Fig. 12.2. The plots were generated at four gate-lengths $L = 430\,\mathrm{nm}, 440\,\mathrm{nm}, 445\,\mathrm{nm}$ and $450\,\mathrm{nm}$; a red cross marks the optimal operating point of the transistor design variant with a gate-length of 440 nm. As can be seen from the plots, evaluation of the objective



(a) $L = 430\,\mathrm{nm}$ **(b)** $L = 440\,\mathrm{nm}$ (optimum)

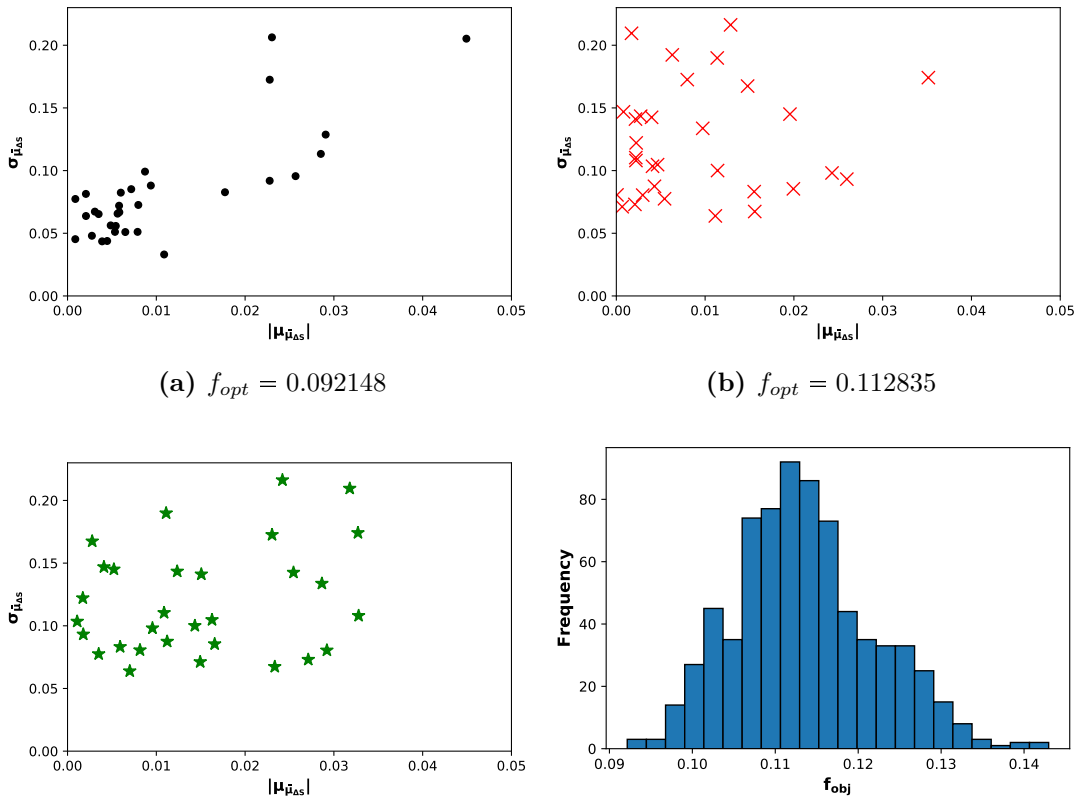
(c) $L = 445\,\mathrm{nm}$ **(d)** $L = 450\,\mathrm{nm}$

Figure 12.3: Objective function f_{opt} close to the optimal parameter settings ($V_g = 1.38\,\mathrm{V}, V_{ds} = 1.655\,\mathrm{V}, L = 440\,\mathrm{nm}$) evaluated at fixed gate-lengths (a) $L = 430\,\mathrm{nm}$, (b) $L = 440\,\mathrm{nm}$, (c) $L = 445\,\mathrm{nm}$ and (d) $L = 450\,\mathrm{nm}$.

function at the proposed solution yields a true minimum of the optimization

problem. Furthermore, the graphs suggest that small deviations from the function minimum due to lithographic variations of the gate-length and miss-adjusted voltage sources still result in near-optimal results: variations in the gate-length in the range of 430 nm to 450 nm and deviations from the optimal bias conditions within 5 % did not increase the objective function values beyond 0.103.

To qualitatively examine the relationship between the objective function values and the form of the distribution parameter distribution, three scatter plots of $|\mu_{\bar{\mu}_{\Delta S}}|$ and $\sigma_{\bar{\mu}_{\Delta S}}$ are shown in Fig. 12.4. The plots displayed in Fig. 12.4a to 12.4c correspond to objective function values of 0.092148, 0.112835, 0.142967, respectively. The distribution of the 730 empirically determined objective function values is depicted in Fig. 12.4d for reference purposes.



(a) $f_{opt} = 0.092148$ **(b)** $f_{opt} = 0.112835$

(c) $f_{opt} = 0.142967$ **(d)**

Figure 12.4: Scatter plot of the distribution parameters $(\mu_{\bar{\mu}_{\Delta S}}, \sigma_{\bar{\mu}_{\Delta S}})$ corresponding to the objective function values (a) $f_{opt} = 0.092148$, (b) $f_{opt} = 0.112835$ and (c) $f_{opt} = 0.142967$. The empirical distribution of the objective function values is shown in (d).

As expected, the overall lowest spread in the distribution parameter distribution and the smallest absolute values were observed for the minimum objective function value (Fig. 12.4a). Larger function values resulted in marked dispersion of the related distribution parameter distributions. Furthermore, $|\mu_{Bias}|$ and σ_{Bias} were shifted to larger values compared to the optimal solution (Fig. 12.4b and Fig. 12.4c).

12.3 Summary

A simple optimization procedure was developed to determine the design parameters and operating settings of competing primitive design variants best suited for the use as PUF. Aim of the optimization process was to create a PUF design that outputs high-entropy responses with a consistent performance over the entire population of PUF instances fabricated in a particular processing technology. For this purpose the design parameters and operating settings were chosen in a way that minimizes the mean and standard deviation of the bias distribution of the underlying PUF population. Application of the methodology was illustrated for a transistor-based primitive design for which the optimal gate-length and bias point (V_g, V_{ds}) was calculated.

Chapter 13

Conclusions and Future Work

Silicon-based PUFs extract random device-specific responses from complex variations in the physical properties of semiconductor devices and circuits fabricated in modern sub-micron VLSI manufacturing technologies. By construction, the properties of the generated responses are closely related to the latent stochastic process representing the production conditions and technology in which a PUF design is fabricated. Hence, understanding the interconnections between the manufacturing process, the physical response mechanism of a PUF and the design parameters of its implementation form is essential to evaluate and develop secure high-performance concepts.

The relationship between the stochastic properties of the manufacturing process and the randomness, uniqueness and reproducibility of PUF responses was analyzed within the scope of a novel random field model framework. Within the scope of the proposed framework, process-induced device mismatch is jointly modeled in terms of a possibly nonnormal distribution characterized by a spatially varying mean function and correlated random components. The deterministic mean function and covariance structure of the random components can be directly estimated from experimental data using spatial variation decomposition techniques such as median polish. Estimation of individual distribution parameters for each PUF instance provides a simple way to correct for spatiotemporal variability of the processing conditions in the distribution model.

The effect of inhomogeneous processing conditions on the response entropy and the inter- and intra-Hamming distance was investigated for experimental data of 193 ring oscillator PUFs tested on commercial FPGAs.

Application of the newly developed framework to ring oscillator PUF data showed that existing model approaches assuming a fixed bitwise bias tended to overestimate the response entropy by up to 33 % compared to estimates calculated for a variable bias model. In case of the intra- and inter-Hamming distance metrics, it was found that the respective probability mass functions could not be approximated by a simple binomial model, if the pairwise mismatch between PUF devices varies. Instead, the distribution of the Hamming distances itself has to be treated as a random realization of a family of possible distributions. Consequently, the uncertainty of related parameter estimates such as the mean inter-Hamming distance or the maximum number of bit-errors originates not only from the variance of the sampled test data but also from the fabrication-related variance of the electrical characteristics of a PUF due to inhomogeneous processing conditions.

Following this random field approach, hierarchical Monte Carlo simulations were performed to analyze the response randomness, uniqueness and reproducibility with respect to the number of response-bits and the error correction capacity required to generate stable full entropy keys suitable to identify a given number of PUF instances. The design space related to the response entropy and number of unique response-bits was simulated for variable systematic device mismatch characterized by a family of Gaussian distributions, each member of which represents one possible realization of the underlying manufacturing process. The Gaussian model was extend by lognormal distributions to investigate the effect of spatially inhomogeneous noise variances on the error correction capacity.

The resulting surrogate models were used to determine bounds on the distribution parameters for which the system size and error capacity comply with user-defined design specifications. It was shown that a high entropy generation rate can only be achieved for tightly centered mismatch distributions with a near-zero mean value. At the same time, a low systematic mismatch results in a high bit-error probability which in turn increases the error capacity needed to mitigate noise effects. Due the disjunctive nature of the parameter subspaces, simultaneous optimization of the entropy generation rate and the error capacity is generally not possible. Furthermore, enforcing these parameter constraints for arbitrary manufacturing technologies proves to be difficult in practice as the required level of control over the processing conditions is technically limited and process changes

might be incompatible with the fabrication of non-PUF semiconductor devices.

In order to resolve these issues, a novel response generation scheme was developed that does not require any modification to the fabrication process. A generic reproducible high-entropy source is realized on the basis of simple algebraic transformations of the electrical characteristics of a PUF. In a first step, the systematic mismatch between the primitive devices constituting a PUF instance is determined by means of variation decomposition techniques. Conditional on the estimated deterministic variation components, pairs of primitives are chosen in a way that minimizes the overall systematic mismatch between the devices. The next step is to compensate for the reduced response stability by extracting bits from higher order pair differences maximizing the absolute difference in their observed electrical characteristics. Since no assumptions are made about the type of the physical characteristics used to generate responses, the proposed PUF design is independent from its implementation regarding the choice of primitive devices and processing technologies.

Implementation independence of the design concept was experimentally verified for transistor, resistor, oscillator and sensor-based primitives tested at different operating conditions. Analysis of the relative response bias and the standard inter- and intra-Hamming performance metrics proved that the generated responses of each primitive device type were highly random, unique and reproducible. The main difference between the devices was the minimum pair order for which error-free regeneration of the responses was achieved.

In order to balance out the increased area overhead associated with higher order pair selection schemes, a new concept was introduced to enable challenge-response functionality for Weak PUF designs. By enhancing the number of CRPs, authentication protocols suitable for Strong PUFs can be employed, thus eliminating the need for computationally and resource-intensive cryptographic hash functions. The proposed Strong PUF design was constructed from cross-coupled parity functions that operate on the response-bits generated by arbitrary Weak PUFs. Practical realization of the boolean functions in hardware only requires basic logic gates, hence offering a lightweight approach to expand the challenge-response space of existing Weak PUFs. Experimental results obtained for the aforementioned primitive devices demonstrated that the generated CRPs uniquely identify a PUF instance. By construction, the responses were found to

be highly sensitive to small variations of the input challenge, hence suggesting an unpredictable CRP behavior. Evaluation of the challenge sensitivity showed that the SAC was fulfilled on average. Furthermore, proof was given that the construction is resilient against learning attacks as the number of training samples increases exponentially in the number of intertwined elements of the coupled parity functions, thus modeling the CRP mechanism becomes computationally unfeasible.

At the time of writing, the two presented PUF concepts only exist in software because the readout circuitry and post-processing logic have yet to be designed and fabricated. In future work, all required components will be implemented on-chip to fully evaluate the performance of the concepts in hardware. Further research will focus on developing new PUF primitives including optical and MEMS based devices for the use in sensor applications with the goal of providing secure on-chip encryption functionality of sensor data and wireless communication channels in sensor networks.

List of Figures

List of Tables

List of Abbreviations

ADC	Analog-to-Digital Converter
ASIC	Application-specific Integrated Circuit
BCH	Bose-Chaudhuri-Hocquenghem
CD	Critical Dimension
CDF	Cumulative Distribution Function
CRP	Challenge-Response Pair
CMOS	Complementary Metal-Oxide-Semiconductor
COTS	Commercial Off-the-Shelf
DRAM	Dynamic Random-Access Memory
EEPROM	Electrically Erasable Programmable Read-Only Memory
FPGA	Field-Programmable Gate Array
HD	Hamming Distance
IBS	Index-Based Syndrome Coding
IC	Integrated Circuit
MEMS	Microelectromechanical Systems
μC	Microcontroller
QQ	Quantile-Quantile
nMOS	Negative Channel Metal-Oxide-Semiconductor

NVM Nonvolatile Memory

PUF Physically Unclonable Function

PAC Probably Approximately Correct

RRAM Resistive Random-Access Memory

SAC Strict Avalanche Criterion

SRAM Static Random-Access Memory

VLSI Very Large Scale Integration

XOR Exclusive Or

Bibliography

[Abdi 07] H. Abdi. "Method of least squares". In: N. J. Salkind, Ed., *Encyclopedia of measurement and statistics*, Chap. Least Squares, Method of, SAGE Publications, Inc., Thousand Oaks, California 91320, 2007.

[Alka 17] M. S. Alkatheiri and Y. Zhuang. "Towards fast and accurate machine learning attacks of feed-forward arbiter PUFs". In: *2017 IEEE Conference on Dependable and Secure Computing*, pp. 181–187, Aug 2017.

[Arlo 10] S. Arlot and A. Celisse. "A survey of cross-validation procedures for model selection". *Statist. Surv.*, Vol. 4, pp. 40–79, 2010.

[Armk 11] F. Armknecht, R. Maes, A. Sadeghi, F. Standaert, and C. Wachsmann. "A Formalization of the Security Features of Physical Functions". In: *2011 IEEE Symposium on Security and Privacy*, pp. 397–412, May 2011.

[Baba 19] A. Babaei and G. Schiele. "Physical Unclonable Functions in the Internet of Things: State of the Art and Open Challenges". *Sensors*, Vol. 19, No. 14, 2019.

[Beck 15] G. T. Becker. "The Gap Between Promise and Reality: On the Insecurity of XOR Arbiter PUFs". In: T. Güneysu and H. Handschuh, Eds., *Cryptographic Hardware and Embedded Systems – CHES 2015*, pp. 535–555, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[Biva 13] R. S. Bivand, E. Pebesma, and V. Gomez-Rubio. *Applied Spatial Data Analysis with R.* Vol. 10 of *Use R!*, Springer-Verlag New York, 2 Ed., 2013.

[Bohm 13] C. Böhm and M. Hofer. *Physical Unclonable Functions in Theory and Practice*. Springer-Verlag New York, 1 Ed., 2013.

[Boja 14] T. M. Bojan, U. R. Kumar, and V. M. Bojan. "An internet of things based intelligent transportation system". In: *2014 IEEE International Conference on Vehicular Electronics and Safety*, pp. 174–179, Dec 2014.

[Boni 96] D. S. Boning and J. E. Chung. "Statistical metrology: Understanding spatial variation in semiconductor manufacturing". In: *Microelectronic Manufacturing Yield, Reliability, and Failure Analysis II*, pp. 16–27, Proc. SPIE, 1996.

[Bosc 08] C. Bösch, J. Guajardo, A.-R. Sadeghi, J. Shokrollahi, and P. Tuyls. "Efficient Helper Data Key Extractor on FPGAs". In: E. Oswald and P. Rohatgi, Eds., *Cryptographic Hardware and Embedded Systems – CHES 2008*, pp. 181–197, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[Bose 60] R. Bose and D. Ray-Chaudhuri. "On a class of error correcting binary group codes". *Information and Control*, Vol. 3, No. 1, pp. 68 – 79, 1960.

[Boyd 04] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, The Edinburgh Building, Cambridge, CB2 8RU, UK, 2004.

[Brei 69] L. Breiman. *Probability and Stochastic Processes: With a View Toward Applications. The Houghton Mifflin Series in Statistics*, Houghton Mifflin Company, Boston, 1 Ed., 1969.

[Che 14] W. Che, J. Plusquellic, and S. Bhunia. "A non-volatile memory based physically unclonable function without helper data". In: *2014 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 148–153, Nov 2014.

[Chen 11] Q. Chen, G. Csaba, P. Lugli, U. Schlichtmann, and U. Rührmair. "The Bistable Ring PUF: A new architecture for strong Physical Un-

clonable Functions". In: *2011 IEEE International Symposium on Hardware-Oriented Security and Trust*, pp. 134–141, June 2011.

[Chen 12] Q. Chen, G. Csaba, P. Lugli, U. Schlichtmann, and U. Rührmair. "Characterization of the bistable ring PUF". In: *2012 Design, Automation Test in Europe Conference Exhibition (DATE)*, pp. 1459–1462, March 2012.

[Chen 97] S. X. Chen and J. S. Liu. "STATISTICAL APPLICATIONS OF THE POISSON-BINOMIAL AND CONDITIONAL BERNOULLI DISTRIBUTIONS". *Statistica Sinica*, Vol. 7, No. 4, pp. 875–892, 1997.

[Crai 08] P. Craig. "A New Reconstruction of Multivariate Normal Orthant Probabilities". *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, Vol. 70, No. 1, pp. 227–243, 2008.

[Cres 93] N. A. C. Cressie. *Statistics for Spatial Data. Wiley series in probability and mathematical statistics. Applied probability and statistics section*, John Wiley & Sons, Inc., revised Ed., 1993.

[Dela 16] M. Delavar, S. Mirzakuchaki, and J. Mohajeri. "A Ring Oscillator-Based PUF With Enhanced Challenge-Response Pairs". *Canadian Journal of Electrical and Computer Engineering*, Vol. 39, No. 2, pp. 174–180, Spring 2016.

[Delv 15] J. Delvaux, D. Gu, D. Schellekens, and I. Verbauwhede. "Helper Data Algorithms for PUF-Based Key Generation: Overview and Analysis". *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 34, No. 6, pp. 889–902, June 2015.

[Dodi 04] Y. Dodis, L. Reyzin, and A. Smith. "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data". In: C. Cachin and J. L. Camenisch, Eds., *Advances in Cryptology - EUROCRYPT 2004*, pp. 523–540, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[Ehre 88] A. Ehrenfeucht, D. Haussler, M. Kearns, and L. Valiant. "A General Lower Bound on the Number of Examples Needed for Learning". In:

Proceedings of the First Annual Workshop on Computational Learning Theory, pp. 139–154, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.

[Eske 16] S. Eskens. "Profiling the European Citizen in the Internet of Things: How Will the General Data Protection Regulation Apply to This Form of Personal Data Processing, and How Should It?". *Available at SSRN: https://ssrn.com/abstract=2752010 or http://dx.doi.org/10.2139/ssrn.2752010*, February 29, 2016.

[Fabe 12] M. H. Faber. *Statistics and Probability Theory: In Pursuit of Engineering Decision Support*. Vol. 18 of *Topics in Safety, Risk, Reliability and Quality*, Springer Netherlands, 1 Ed., 2012.

[Feld 10] R. M. Feldman and C. Valdez-Flores. *Applied Probability and Stochastic Processes*. Springer-Verlag Berlin Heidelberg, 2 Ed., 2010.

[Fisc 92] P. Fischer and H. U. Simon. "On Learning Ring-Sum-Expansions". *SIAM Journal on Computing*, Vol. 21, No. 1, pp. 181–192, 1992.

[Frie 10] J. Friedman, T. Hastie, and R. Tibshirani. "Regularization paths for generalized linear models via coordinate descent". *Journal of statistical software*, Vol. 33, No. 1, pp. 1–22, 2010.

[Ganj 15] F. Ganji, S. Tajik, and J.-P. Seifert. "Why Attackers Win: On the Learnability of XOR Arbiter PUFs". In: M. Conti, M. Schunter, and I. Askoxylakis, Eds., *Trust and Trustworthy Computing*, pp. 22–39, Springer International Publishing, Cham, 2015.

[Ganj 16] F. Ganji, S. Tajik, F. Fäßler, and J.-P. Seifert. "Strong machine learning attack against PUFs with no mathematical model". In: *Proceedings of the 18th International Conference on Cryptographic Hardware and Embedded Systems — CHES 2016 - Volume 9813*, pp. 391–411, Springer-Verlag New York, Inc., New York, NY, USA, 2016.

[Gass 02] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas. "Silicon Physical Random Functions". In: *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 148–160, ACM, New York, NY, USA, 2002.

[Gass 04] B. Gassend, D. Lim, D. Clarke, M. van Dijk, and S. Devadas. "Identification and authentication of integrated circuits". *Concurrency and Computation: Practice and Experience*, Vol. 16, No. 11, pp. 1077–1098, 2004.

[Gelm 14] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian Data Analysis. Chapman & Hall/CRC Texts in Statistical Science Series*, CRC Press, 6000 Broken Sound Parkway NW, Suite 300. Boca Raton, FL 33487, 3 Ed., 2014.

[Gree 15] A. Greenberg. "After Jeep Hack, Chrysler Recalls 1.4M Vehicles for Bug Fix". `https://www.wired.com/2015/07/jeep-hack-chrysler-recalls-1-4m-vehicles-bug-fix/`, 24. July 2015. [Online; accessed 2018-05-12].

[Guaj 07] J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls. "FPGA Intrinsic PUFs and Their Use for IP Protection". In: P. Paillier and I. Verbauwhede, Eds., *Cryptographic Hardware and Embedded Systems - CHES 2007*, pp. 63–80, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[Guo 16] Q. Guo, J. Ye, Y. Gong, Y. Hu, and X. Li. "Efficient Attack on Non-linear Current Mirror PUF with Genetic Algorithm". In: *2016 IEEE 25th Asian Test Symposium (ATS)*, pp. 49–54, Nov 2016.

[Hast 01] A. Hastings. *The Art of Analog Layout*. Prentice Hall, Upper Saddle River, New Jersey 07458, 1 Ed., 2001.

[Helb 16] N. Helberger. "Profiling and Targeting Consumers in the Internet of Things - A New Challenge for Consumer Law". *Available at SSRN: `https://ssrn.com/abstract=2728717` or `http://dx.doi.org/10.2139/ssrn.2728717`*, February 6, 2016.

[Herd 14] C. Herder, M.-D. M. Yu, F. Koushanfar, and S. Devadas. "Physical Unclonable Functions and Applications: A Tutorial". *Proceedings of the IEEE*, Vol. 102, No. 8, pp. 1126–1141, Aug 2014.

[Hill 12] M. Hiller, D. Merli, F. Stumpf, and G. Sigl. "Complementary IBS: Application specific error correction for PUFs". In: *2012 IEEE International Symposium on Hardware-Oriented Security and Trust*, pp. 1–6, June 2012.

[Hill 13] M. Hiller, G. Sigl, and M. Pehl. "A new model for estimating bit error probabilities of Ring-Oscillator PUFs". In: *2013 8th International Workshop on Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC)*, pp. 1–8, July 2013.

[Hill 16] M. Hiller, M.-D. M. Yu, and G. Sigl. "Cherry-Picking Reliable PUF Bits With Differential Sequence Coding". *IEEE Transactions on Information Forensics and Security*, Vol. 11, No. 9, pp. 2065–2076, Sep. 2016.

[Holc 09] 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, Sep. 2009.

[Hong 13] Y. Hong. "On Computing the Distribution Function for the Poisson Binomial Distribution". *Comput. Stat. Data Anal.*, Vol. 59, pp. 41–51, March 2013.

[Hori 10] Y. Hori, T. Yoshida, T. Katashita, and A. Satoh. "Quantitative and Statistical Performance Evaluation of Arbiter Physical Unclonable Functions on FPGAs". In: *2010 International Conference on Reconfigurable Computing and FPGAs*, pp. 298–303, Dec 2010.

[Hube 10] M. Hubert and M. Debruyne. "Minimum covariance determinant". *Wiley Interdisciplinary Reviews: Computational Statistics*, Vol. 2, No. 1, pp. 36–43, 2010.

[Isla 15] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K.-S. Kwak. "The Internet of Things for Health Care: A Comprehensive Survey". *IEEE Access*, Vol. 3, No. , pp. 678–708, 2015.

[Jia 15] S. Jia, L. Xia, Z. Wang, J. Lin, G. Zhang, and Y. Ji. "Extracting Robust Keys from NAND Flash Physical Unclonable Functions". In:

J. Lopez and C. J. Mitchell, Eds., *Information Security*, pp. 437–454, Springer International Publishing, Cham, 2015.

[Jone 01] E. Jones, T. Oliphant, P. Peterson, *et al.* "SciPy: Open source scientific tools for Python". `http://www.scipy.org/`, 2001. [Online; accessed 2018-01-03].

[Ju 13] J. Ju, R. Chakraborty, C. Lamech, and J. Plusquellic. "Stability analysis of a physical unclonable function based on metal resistance variations". In: *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pp. 143–150, June 2013.

[Katz 12] 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: E. Prouff and P. Schaumont, Eds., *Cryptographic Hardware and Embedded Systems – CHES 2012*, pp. 283–301, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[Kend 38] M. G. Kendall. "A NEW MEASURE OF RANK CORRELATION". *Biometrika*, Vol. 30, No. 1-2, pp. 81–93, 06 1938.

[Kend 45] M. G. Kendall. "The Treatment of Ties in Ranking Problems". *Biometrika*, Vol. 33, No. 3, pp. 239–251, 1945.

[Kim 15] M.-S. Kim, D.-I. Moon, S.-K. Yoo, S.-H. Lee, and Y.-K. Choi. "Investigation of Physically Unclonable Functions Using Flash Memory for Integrated Circuit Authentication". *IEEE Transactions on Nanotechnology*, Vol. 14, No. 2, pp. 384–389, March 2015.

[Kuma 08] S. S. Kumar, J. Guajardo, R. Maes, G. Schrijen, and P. Tuyls. "Extended abstract: The butterfly PUF protecting IP on every FPGA". In: *2008 IEEE International Workshop on Hardware-Oriented Security and Trust*, pp. 67–70, June 2008.

[Kuma 14] R. Kumar and W. Burleson. "On design of a highly secure PUF based on non-linear current mirrors". In: *2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pp. 38–43, May 2014.

[Lees 10] V. van der Leest, G.-J. Schrijen, H. Handschuh, and P. Tuyls. "Hardware Intrinsic Security from D Flip-flops". In: *Proceedings of the Fifth ACM Workshop on Scalable Trusted Computing*, pp. 53–62, ACM, New York, NY, USA, 2010.

[Lim 05] D. Lim, J. W. Lee, B. Gassend, G. E. Suh, M. van Dijk, and S. Devadas. "Extracting secret keys from integrated circuits". *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 13, No. 10, pp. 1200–1205, Oct 2005.

[Liu 16] R. Liu, H. Wu, Y. Pang, H. Qian, and S. Yu. "A highly reliable and tamper-resistant RRAM PUF: Design and experimental validation". In: *2016 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pp. 13–18, May 2016.

[Lo 12] C. F. Lo. "The Sum and Difference of Two Lognormal Random Variables". *Journal of Applied Mathematics*, Vol. 2012, pp. 1–13, 2012. Article ID 838397.

[Lofs 00] K. Lofstrom, W. R. Daasch, and D. Taylor. "IC identification circuit using device mismatch". In: *2000 IEEE International Solid-State Circuits Conference. Digest of Technical Papers (Cat. No.00CH37056)*, pp. 372–373, Feb 2000.

[Maes 08] R. Maes, P. Tuyls, and I. Verbauwhede. "Intrinsic PUFs from flip-flops on reconfigurable devices". In: *3rd Benelux Workshop on Information and System Security*, 2008.

[Maes 09] R. Maes, P. Tuyls, and I. Verbauwhede. "Low-Overhead Implementation of a Soft Decision Helper Data Algorithm for SRAM PUFs". In: C. Clavier and K. Gaj, Eds., *Cryptographic Hardware and Embedded Systems - CHES 2009*, pp. 332–347, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[Maes 12] R. Maes, A. Van Herrewege, and I. Verbauwhede. "PUFKY: A Fully Functional PUF-Based Cryptographic Key Generator". In: E. Prouff

and P. Schaumont, Eds., *Cryptographic Hardware and Embedded Systems – CHES 2012*, pp. 302–319, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[Maes 13] R. Maes. *Physically Unclonable Functions: Constructions, Properties and Applications*. Springer-Verlag Berlin Heidelberg, 1 Ed., 2013.

[Maes 15] R. Maes, V. van der Leest, E. van der Sluis, and F. Willems. "Secure Key Generation from Biased PUFs". In: T. Güneysu and H. Handschuh, Eds., *Cryptographic Hardware and Embedded Systems – CHES 2015*, pp. 517–534, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[Mahm 15] R. Mahmoud, T. Yousuf, F. Aloul, and I. Zualkernan. "Internet of things (IoT) security: Current status, challenges and prospective measures". In: *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*, pp. 336–341, Dec 2015.

[Mait 10] A. Maiti, J. Casarona, L. McHale, and P. Schaumont. "A large scale characterization of RO-PUF". In: *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pp. 94–99, June 2010.

[Mait 11] A. Maiti and P. Schaumont. "Improved Ring Oscillator PUF: An FPGA-friendly Secure Primitive". *Journal of Cryptology*, Vol. 24, No. 2, pp. 375–397, Apr 2011.

[Mait 12] A. Maiti, I. Kim, and P. Schaumont. "A Robust Physical Unclonable Function With Enhanced Challenge-Response Set". *IEEE Transactions on Information Forensics and Security*, Vol. 7, No. 1, pp. 333–345, Feb 2012.

[Mait 13] A. Maiti, V. Gunreddy, and P. Schaumont. "A Systematic Method to Evaluate and Compare the Performance of Physical Unclonable Functions". In: P. Athanas, D. Pnevmatikatos, and N. Sklavos, Eds., *Embedded Systems Design with FPGAs*, pp. 245–267, Springer New York, New York, NY, 2013.

[Majz 08a] M. Majzoobi, F. Koushanfar, and M. Potkonjak. "Lightweight secure PUFs". In: *2008 IEEE/ACM International Conference on Computer-Aided Design*, pp. 670–673, Nov 2008.

[Majz 08b] M. Majzoobi, F. Koushanfar, and M. Potkonjak. "Testing Techniques for Hardware Security". In: *2008 IEEE International Test Conference*, pp. 1–10, Oct 2008.

[Maza 15] A. Mazady, M. T. Rahman, D. Forte, and M. Anwar. "Memristor PUF - A Security Primitive: Theory and Experiment". *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, Vol. 5, No. 2, pp. 222–229, June 2015.

[McKi 10] W. McKinney. "Data Structures for Statistical Computing in Python". In: S. van der Walt and J. Millman, Eds., *Proceedings of the 9th Python in Science Conference*, pp. 51 – 56, 2010.

[Mill 11] K. J. Millman and M. Aivazis. "Python for Scientists and Engineers". *Computing in Science Engineering*, Vol. 13, No. 2, pp. 9–12, March 2011.

[Mill 15] C. Miller and C. Valasek. "Remote Exploitation of an Unaltered Passenger Vehicle". *Black Hat USA*, 2015.

[Mitt 18] S. Mittal and A. I. Alsalibi. "A Survey of Techniques for Improving Security of Non-volatile Memories". *Journal of Hardware and Systems Security*, Vol. 2, No. 2, pp. 179–200, Jun 2018.

[Miwa 03] T. Miwa, A. J. Hayter, and S. Kuriki. "The Evaluation of General Non-Centred Orthant Probabilities". *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, Vol. 65, No. 1, pp. 223–234, 2003.

[Mora 50] P. A. P. Moran. "Notes on Continuous Stochastic Phenomena". *Biometrika*, Vol. 37, No. 1/2, pp. 17–23, 1950.

[Mugg 08] V. M. Muggeo. "segmented: An R Package to Fit Regression Models with Broken-Line Relationships". *R news*, Vol. 8, No. 1, pp. 20–25, 2008.

[Papp 02] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld. "Physical One-Way Functions". *Science*, Vol. 297, No. 5589, pp. 2026–2030, 2002.

[Para 11] Z. S. Paral and S. Devadas. "Reliable and efficient PUF-based key generation using pattern matching". In: *2011 IEEE International Symposium on Hardware-Oriented Security and Trust*, pp. 128–133, June 2011.

[Pehl 14] M. Pehl, A. R. Punnakkal, M. Hiller, and H. Graeb. "Advanced performance metrics for Physical Unclonable Functions". In: *2014 International Symposium on Integrated Circuits (ISIC)*, pp. 136–139, Dec 2014.

[Piya 13] R. Piyare. "Internet of Things: Ubiquitous Home Control and Monitoring System using Android based Smart Phone". *International Journal of Internet of Things*, Vol. 2, No. 1, pp. 5–11, 2013.

[Prab 11] P. Prabhu, A. Akel, L. M. Grupp, W.-K. S. Yu, G. E. Suh, E. Kan, and S. Swanson. "Extracting Device Fingerprints from Flash Memory by Exploiting Physical Variations". In: J. M. McCune, B. Balacheff, A. Perrig, A.-R. Sadeghi, A. Sasse, and Y. Beres, Eds., *Trust and Trustworthy Computing*, pp. 188–201, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[Pyth 18] Python Core Team. "Python: A General Purpose Programming Language". `https://www.python.org/`, 2018.

[R Co 18] R Core Team. "R: A Language and Environment for Statistical Computing". `http://www.R-project.org/`, 2018.

[Rahm 15] M. T. Rahman, D. Forte, F. Rahman, and M. Tehranipoor. "A pair selection algorithm for robust RO-PUF against environmental variations and aging". In: *2015 33rd IEEE International Conference on Computer Design (ICCD)*, pp. 415–418, Oct 2015.

[Roy 09] D. Roy, J. H. Klootwijk, N. A. M. Verhaegh, H. H. A. J. Roosen, and R. A. M. Wolters. "Comb Capacitor Structures for On-Chip Physical Uncloneable Function". *IEEE Transactions on Semiconductor Manufacturing*, Vol. 22, No. 1, pp. 96–102, Feb 2009.

[Ruhr 10] 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*, pp. 237–249, ACM, New York, NY, USA, 2010.

[Ruhr 14] U. Rührmair and D. E. Holcomb. "PUFs at a glance". In: *2014 Design, Automation Test in Europe Conference Exhibition (DATE)*, pp. 1–6, March 2014.

[Scha 05] O. Schabenberger and C. A. Gotway. *Statistical Methods for Spatial Data Analysis. Chapman & Hall/CRC Texts in Statistical Science Series*, Chapman & Hall/CRC, 6000 Broken Sound Parkway NW, Suite 300. Boca Raton, FL 33487, 1 Ed., 2005.

[Scha 18] A. Schaub, J.-L. Danger, S. Guilley, and O. Rioul. "An Improved Analysis of Reliability and Entropy for Delay PUFs". In: *2018 21st Euromicro Conference on Digital System Design (DSD)*, pp. 553–560, Aug 2018.

[Sedc 06] P. Sedcole and P. Y. K. Cheung. "Within-die Delay Variability in 90nm FPGAs and Beyond". In: *2006 IEEE International Conference on Field Programmable Technology*, pp. 97–104, Dec 2006.

[Seli 11] G. Selimis, M. Konijnenburg, M. Ashouei, J. Huisken, H. de Groot, V. van der Leest, G. Schrijen, M. van Hulst, and P. Tuyls. "Evaluation of 90nm 6T-SRAM as Physical Unclonable Function for secure key generation in wireless sensor nodes". In: *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*, pp. 567–570, May 2011.

[Sher 15] J. Sherly and D. Somasundareswari. "Internet of things based smart transportation systems". *International Research Journal of Engineering and Technology*, Vol. 2, No. 7, pp. 1207–1210, 2015.

[Shro 14] F. Shrouf, J. Ordieres, and G. Miragliotta. "Smart factories in Industry 4.0: A review of the concept and of energy management approached in production based on the Internet of Things paradigm".

In: *2014 IEEE International Conference on Industrial Engineering and Engineering Management*, pp. 697–701, Dec 2014.

[Shro 15] F. Shrouf and G. Miragliotta. "Energy management based on Internet of Things: practices and framework for adoption in production management". *Journal of Cleaner Production*, Vol. 100, pp. 235 – 246, 2015.

[Skor 06] B. Škorić, S. Maubach, T. Kevenaar, and P. Tuyls. "Information-theoretic analysis of capacitive physical unclonable functions". *Journal of Applied Physics*, Vol. 100, No. 2, p. 024902, 2006.

[Smor 17] G. Smorodin, O. Kolesnichenko, Y. Kolesnichenko, L. Myakinkova, N. Prisyazhnaya, D. Yakovleva, L. Mazelis, A. Martynov, V. Pulit, D. Danilova, N. Litvak, and S. Balandin. "Internet of Things: Modem paradigm of health care". In: *2017 21st Conference of Open Innovations Association (FRUCT)*, pp. 311–320, Nov 2017.

[Soli 13] M. Soliman, T. Abiodun, T. Hamouda, J. Zhou, and C.-H. Lung. "Smart Home: Integrating Internet of Things with Web Services and Cloud Computing". In: *2013 IEEE 5th International Conference on Cloud Computing Technology and Science*, pp. 317–320, Dec 2013.

[Step 79] M. A. Stephens. "The Anderson-Darling Statistic". Tech. Rep. 39, Department of Statistics, Stanford University, Stanford, CA, 31. October 1979.

[Stin 97] B. E. Stine, D. S. Boning, and J. E. Chung. "Analysis and decomposition of spatial variation in integrated circuit processes and devices". *IEEE Transactions on Semiconductor Manufacturing*, Vol. 10, No. 1, pp. 24–41, Feb 1997.

[Su 07] Y. Su, J. Holleman, and B. Otis. "A 1.6 pJ/bit 96% Stable Chip-ID Generating Circuit using Process Variations". In: *2007 IEEE International Solid-State Circuits Conference. Digest of Technical Papers*, pp. 406–611, Feb 2007.

[Suh 05] G. E. Suh, C. W. O'Donnell, I. Sachdev, and S. Devadas. "Design and implementation of the AEGIS single-chip secure processor using

physical random functions". In: *32nd International Symposium on Computer Architecture (ISCA'05)*, pp. 25–36, June 2005.

[Suh 07] G. E. Suh and S. Devadas. "Physical Unclonable Functions for Device Authentication and Secret Key Generation". In: *2007 44th ACM/IEEE Design Automation Conference*, pp. 9–14, June 2007.

[Tarm 01] G. Tarmast. "Multivariate log-normal distribution". In: *53rd Session of the International Statistical Institute (Seoul)*, 2001.

[Tehr 17] F. Tehranipoor, N. Karimian, W. Yan, and J. A. Chandy. "DRAM-Based Intrinsic Physically Unclonable Functions for System-Level Security and Authentication". *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 25, No. 3, pp. 1085–1097, March 2017.

[Timm 02] N. H. Timm. *Applied Multivariate Analysis. Springer Texts in Statistics*, Springer-Verlag New York, 1 Ed., 2002.

[Tong 90] Y. L. Tong. *The Multivariate Normal Distribution. Springer Series in Statistics*, Springer-Verlag New York, 1 Ed., 1990.

[Tsag 14] M. Tsagris, C. Beneki, and H. Hassani. "On the Folded Normal Distribution". *Mathematics*, Vol. 2, No. 1, pp. 12–28, 2014.

[Vali 84] L. G. Valiant. "A Theory of the Learnable". *Commun. ACM*, Vol. 27, No. 11, pp. 1134–1142, Nov. 1984.

[Vida 14] S. Vidalis and O. Angelopoulou. "Assessing Identity Theft in the Internet of Things". *Journal of IT Convergence Practice (INPRA)*, Vol. 2, No. 1, pp. 15–21, March 2014.

[Wang 12] Y. Wang, W.-k. Yu, S. Wu, G. Malysa, G. E. Suh, and E. C. Kan. "Flash Memory for Ubiquitous Hardware Security Functions: True Random Number Generation and Device Fingerprints". In: *2012 IEEE Symposium on Security and Privacy*, pp. 33–47, May 2012.

[Wang 17] Z. Wang, Y. Chen, A. Patil, C.-H. Chang, and A. Basu. "Current mirror array: A novel lightweight strong PUF topology with enhanced

reliability". In: *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–4, May 2017.

[Webs 86] A. F. Webster and S. E. Tavares. "On the Design of S-Boxes". In: H. C. Williams, Ed., *Advances in Cryptology — CRYPTO '85 Proceedings*, pp. 523–534, Springer Berlin Heidelberg, Berlin, Heidelberg, 1986.

[Wild 14] F. Wilde, M. Hiller, and M. Pehl. "Statistic-based security analysis of ring oscillator PUFs". In: *2014 International Symposium on Integrated Circuits (ISIC)*, pp. 148–151, Dec 2014.

[Will 17a] B. Willsch, J. Hauser, S. Dreiner, A. Goehlich, H. Kappert, and H. Vogt. "Analysis of Semiconductor Process Variations by Means of Hierarchical Median Polish". In: *2017 Austrochip Workshop on Microelectronics (Austrochip)*, pp. 1–5, Oct 2017.

[Will 17b] B. Willsch, K.-U. Müller, Q. Zhang, J. Hauser, S. Dreiner, A. Stanitzki, H. Kappert, R. Kokozinski, and H. Vogt. "Implementation of an Integrated Differential Readout Circuit for Transistor-Based Physically Unclonable Functions". In: *2017 Austrochip Workshop on Microelectronics (Austrochip)*, pp. 58–63, Oct 2017.

[Will 18] B. Willsch, M. te Heesen, J. Hauser, S. Dreiner, H. Kappert, and H. Vogt. "Evaluation of a median threshold based EEPROM-PUF concept implemented in a high temperature SOI CMOS technology". In: *2018 13th International Conference on Design Technology of Integrated Systems In Nanoscale Era (DTIS)*, pp. 1–6, April 2018.

[Wisi 18] N. Wisiol and M. Margraf. "Attacking RO-PUFs with Enhanced Challenge-Response Pairs". In: L. J. Janczewski and M. Kutyłowski, Eds., *ICT Systems Security and Privacy Protection*, pp. 122–126, Springer International Publishing, Cham, 2018.

[Xu 15] X. Xu, U. Rührmair, D. E. Holcomb, and W. Burleson. "Security Evaluation and Enhancement of Bistable Ring PUFs". In: S. Mangard and P. Schaumont, Eds., *Radio Frequency Identification*, pp. 3–16, Springer International Publishing, Cham, 2015.

[Yama 14] D. Yamamoto, M. Takenaka, K. Sakiyama, and N. Torii. "Security evaluation of bistable ring PUFs on FPGAs using differential and linear analysis". In: *2014 Federated Conference on Computer Science and Information Systems*, pp. 911–918, Sep. 2014.

[Yin 09] C.-E. Yin and G. Qu. "Temperature-aware cooperative ring oscillator PUF". In: *2009 IEEE International Workshop on Hardware-Oriented Security and Trust*, pp. 36–42, July 2009.

[Yin 10] C.-E. Yin and G. Qu. "LISA: Maximizing RO PUF's secret extraction". In: *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pp. 100–105, June 2010.

[Yin 13a] C.-E. Yin and G. Qu. "Improving PUF security with regression-based distiller". In: *2013 50th ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1–6, May 2013.

[Yin 13b] C.-E. Yin, G. Qu, and Q. Zhou. "Design and implementation of a group-based RO PUF". In: *2013 Design, Automation Test in Europe Conference Exhibition (DATE)*, pp. 416–421, March 2013.

[Yin 14] C.-E. Yin and G. Qu. "Obtaining Statistically Random Information From Silicon Physical Unclonable Functions". *IEEE Transactions on Emerging Topics in Computing*, Vol. 2, No. 2, pp. 96–106, June 2014.

[Yu 10a] M.-D. M. Yu and S. Devadas. "Secure and robust error correction for physical unclonable functions". *IEEE Design Test of Computers*, Vol. 27, No. 1, pp. 48–65, Jan 2010.

[Yu 10b] M.-D. M. Yu and S. Devadas. "Recombination of Physical Unclonable Functions". *35th Annual GOMACTech Conference*, pp. 22–25, March 2010. Reno, NV.

[Yu 12] M.-D. M. Yu, R. Sowell, A. Singh, D. M'Raïhi, and S. Devadas. "Performance metrics and empirical results of a PUF cryptographic key generation ASIC". In: *2012 IEEE International Symposium on Hardware-Oriented Security and Trust*, pp. 108–115, June 2012.

[Zhou 17] C. Zhou, K. K. Parhi, and C. H. Kim. "Secure and reliable XOR arbiter PUF design: An experimental study based on 1 trillion challenge response pair measurements". In: *2017 54th ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1–6, June 2017.

Publications and Patents

[1] B. Willsch, M. te Heesen, J. Hauser, S. Dreiner, H. Kappert, and H. Vogt, "Evaluation of a median threshold based EEPROM-PUF concept implemented in a high temperature SOI CMOS technology," in *2018 13th International Conference on Design Technology of Integrated Systems In Nanoscale Era (DTIS)*, pp. 1–6, April 2018.

[2] B. Willsch, J. Hauser, S. Dreiner, A. Goehlich, H. Kappert, and H. Vogt, "Analysis of semiconductor process variations by means of hierarchical median polish," in *2017 Austrochip Workshop on Microelectronics (Austrochip)*, pp. 1–5, Oct 2017.

[3] B. Willsch, K.-U. Müller, Q. Zhang, J. Hauser, S. Dreiner, A. Stanitzki, H. Kappert, R. Kokozinski, and H. Vogt, "Implementation of an integrated differential readout circuit for transistor-based physically unclonable functions," in *2017 Austrochip Workshop on Microelectronics (Austrochip)*, pp. 58–63, Oct 2017.

[4] B. Willsch, J. Hauser, S. Dreiner, A. Goehlich, and H. Vogt, "Statistical tests to determine spatial correlations in the response behavior of puf," in *2016 12th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, pp. 1–4, June 2016.

[5] B. Willsch and D. Stefan, "Verfahren und Vorrichtung zur Nutzung von primitiven Physikalisch Unklonbarer Funktionen." FH170902PDE / kg, 2018 (patent pending).

[6] B. Willsch and D. Stefan, "Vorrichtung und Verfahren zur Erzeugung Physikalisch Unklonbarer Funktionen." FH180801PDE / kg, 2018 (patent pending).