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

# Impact of Localized Electromagnetic Field Measurements on Implementations of Asymmetric Cryptography

**Johann Heyszl**

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

Vorsitzender der Kommission: Univ.-Prof. Dr.-Ing. Ulf Schlichtmann
Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing. Georg Sigl
2. Univ.-Prof. Dr.-Ing. Christof Paar, Ruhr-Universität Bochum

Die Dissertation wurde am 23. Januar 2013 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 16. Juli 2013 angenommen.

# Abstract

Implementations of cryptographic algorithms are threatened by side-channel analysis, which denotes the recovery of secret keys through observations of e.g., the current consumption of a device during cryptographic operations. In this thesis, I investigate the use of high-resolution electromagnetic field measurements for side-channel analysis. Contrary to previous contributions about precise electromagnetic field measurements in side-channel analysis, I specifically concentrate on *localized* aspects of such measurements, which means that the measurements are restricted to a certain spatial extent. Previous publications either conclude that localized measurements of electromagnetic fields are impossible, or show unconvincing, coarse localizations without dedicated exploitation of such localized measurements. In this thesis, I improve the current state of research by investigating the feasibility, quality and dedicated use of localized electromagnetic field measurements.

In a first effort, I performed an extensive study about the strengths and limitations of such measurements. For this, I designed a test setup including a tailored hardware design configured into a depackaged Field Programmable Gate Array (FPGA) and used state-of-the-art high-resolution magnetic field measurement equipment. The measurements are processed in several different ways using statistical analysis to extract the relevant information. From this I am able to clearly demonstrate the feasibility and quality of localized measurements which yields two main results impacting implementations of cryptographic algorithms. First, the side-channel-signal quality can be significantly improved at eligible measurement positions, and second, dedicated side-channel attacks become possible. Additionally, I derive important conclusions about the measurement setup and processing of such traces in side-channel scenarios. Measurements from the frontside of an integrated circuit die using a magnetic sensor coil in the horizontal plane lead to the highest signal-to-noise ratios. High sampling rates are required and no trace compression should be applied. Finally, localized electromagnetic field measurements suffer from fewer parasitics than current consumption measurements using a resistor in the ground or supply voltage.

After establishing the feasibility of localized measurements, I continue to describe how such localized measurements can be used to extract dedicated *location-based side-channel information leakage* from certain cryptographic algorithms. In integrated circuits, logic is located at different distances to a high-precision measurement probe. The location of the side-channel leakage allows to recover information about the secret during a cryptographic computation if the different locations are used in a way that depends on secret information. I describe, how popular exponentiation algorithms, which are e.g., used in elliptic curve cryptography fall into this category and exhibit location-based side-channel leakage. Attacks based on localized measurements are possible even if countermeasures such as exponent blinding are included or protocols restrict adversaries to single observations. To demonstrate this, I use an FPGA-based hardware implementation of an elliptic curve scalar point multiplication algorithm for a practical evaluation. Profiling of the leakage of the device using multiple measurements leads to an eligible measurement position. A profiled template attack exploiting a single localized measurement is able to recover the scalar almost entirely. I suggest a countermeasure which randomizes storage locations and demonstrate how it prevents the described attack.

As an improvement, I present a *non-profiled* side-channel attack to exploit location-based side-channel leakage of exponentiation algorithms. This attack applies well-researched *unsupervised cluster classification* algorithms to recover the secret scalar and does not require profiling, hence, the generation of templates. This clustering-based attack can be used to exploit arbitrary single-execution side-channel leakage. In this way I extend previous work by Walter who, contrarily, used an individual algorithm. I practically demonstrate a successful and complete recovery of the scalar from the previous setup which includes the FPGA-based elliptic curve cryptography implementation and a localized measurement at an eligible position.

The success probability of such single-execution attacks depends on the quality of the side-channel measurements. It follows directly from the localization property, that different measurement positions lead to different observed side-channel information. A concurrent measurement of side-channel leakage at different positions during a single execution leads to more recovered information. The combination of measurements has already been described for other side-channel attacks. I improve the clustering based side-channel attack on exponentiations by combining multiple simultaneous measurements. During a practical study, I use a regular array of measurement positions on the surface of the FPGA from the previous evaluation. The case study shows that three measurements of the same execution from different positions lead to a full recovery of the secret scalar, even though the positions are chosen

without prior profiling of the spatial leakage distribution. Hence, no prior profiling to find the best measurement position is necessary. Instead, multiple measurement probes and a combination of measurements is sufficient. This is a significant threat and might equally apply to other implementations.

*To summarize, I contribute results regarding the strengths and limitations of high-resolution EM measurements for side-channel analysis and describe how location-based single-execution information leakage of cryptographic algorithms can be exploited in dedicated attacks on implementations of asymmetric cryptography. Furthermore, I introduce unsupervised cluster classification algorithms as an attack to exploit such single-execution leakage and as a means to combine simultaneous measurements in such attacks.*

# Kurzfassung

Seitenkanalanalysen stellen eine ernstzunehmende Bedrohung für Implementierungen von kryptographischen Algorithmen dar. Sie ermöglichen die Ermittlung von geheimen Schlüsseln durch Messungen zum Beispiel des Stromverbrauchs während einer kryptographischen Berechnung. In meiner Arbeit untersuche ich die Verwendung von Messungen elektromagnetischer Abstrahlung für die Seitenkanalanalyse. Im Gegensatz zu bisherigen Ergebnissen konzentriere ich mich speziell auf sehr hochauflösende und örtlich-begrenzte Messungen. Bisher wurden mit sehr groben Auflösungen wenig überzeugende Ergebnisse bezüglich der örtlichen Zusammenhänge gezeigt und auch keine spezifischen Nutzen der Ortsauflösung beschrieben. Im Vergleich zum aktuellen Stand der Forschung zeige ich die qualitativen Möglichkeiten und den spezifischen Nutzen solcher örtlich-begrenzten Messungen für die Seitenkanalanalyse kryptographischer Implementierungen.

In einem ersten Schritt habe ich eine ausführliche praktische Studie über die qualitativen Merkmale solcher Messungen durchgeführt. Zu diesem Zweck habe ich eine spezielle Testschaltung auf einem FPGA (Field Programmable Gate Array) entwickelt und mit einem hochauflösenden Messaufbau für Magnetfelder vermessen. Die Messungen habe ich auf verschiedene Arten statistisch analysiert, um repräsentative Seitenkanalinformationen zu extrahieren. Auf diese Art gelang es mir, die Machbarkeit von örtlich-begrenzten Messungen und deren Qualitäten klar zu zeigen. Zwei vorwiegende Erkenntnisse konnte ich gewinnen. Erstens ist die Qualität von solchen örtlich-begrenzten Messungen für die Seitenkanalanalyse an den entsprechend richtigen Messpunkten deutlich erhöht. Zweitens ermöglicht der Beweis für die Machbarkeit von örtlich-begrenzten Messungen spezifische Angriffe auf Basis von Ortsinformationen. Im Zuge dieser Arbeiten konnte ich außerdem einige grundlegende Aspekte solcher Messungen klären. Messungen eines integrierten Schaltkreises von "oben" mit einer Messspule in horizontaler Ebene führen zu den größten Signal-zu-Rausch Verhältnissen. Außerdem sind im Vergleich zur Strommessung hohe Abtastraten notwendig und eine Kompression der Messdaten ist nicht ratsam. Schlußendlich sind in

solchen örtlich-begrenzten Messungen deutlich weniger parasitäre Effekte zu beobachten als in Strommessungen.

Im nächsten Schritt zeige ich, wie solche örtlich-begrenzten Messungen verwendet werden können, um spezifische ortsabhängige Seitenkanalinformationen von kryptographischen Implementierung zu extrahieren. In integrierten Schaltkreisen liegen einzelne Schaltungselemente an verschiedenen Stellen und damit in unterschiedlicher Distanz zu einer hochpräzisen Messspule. Wenn nun während einer kryptographischen Berechnung verschiedene Schaltungsteile, beispielsweise Speicherzellen, in Abhängigkeit von geheimer Information genutzt werden, kann die gemessene ortsabhängige Seitenkanalinformation zur Ermittlung des geheimen Schlüssels dienen. Dies ist der Fall für viele wichtige Exponentiationsalgorithmen, wie sie beispielsweise in der elliptischen Kurven Kryptographie eingesetzt werden. Angriffe auf Basis dieser ortsabhängigen Seitenkanalinformation sind trotz Gegenmaßnahmen wie zum Beispiel dem bekannten Verschleiern des Exponenten und trotz der Tatsache, dass in den meisten Protokollen der betreffende Exponent in jeder Ausführung neu gewählt wird, möglich. Um dies zu zeigen, habe ich einen elliptischen Kurven Prozessor auf einem FPGA implementiert und hochauflösende Messungen durchgeführt. Im Rahmen eines Template-Angriffs habe ich zuerst eine Charakterisierung an vielen Messpunkten und anschließend einen Angriff an einem geeigneten Messpunkt durchgeführt. Auf diese Art ist es gelungen, den geheimen Exponenten mit nur einer Messung fast vollständig zu ermitteln. Um solcherart Angriffe zu verhindern, schlage ich als Gegenmaßnahme vor, die Speicherorte von Zwischenwerten in der Implementierung der betroffenen Algorithmen an zufälligen Zeitpunkten zu vertauschen. Die Wirksamkeit dieses Vorschlags konnte ich anhand eine praktischen Messung zeigen.

Der zuvor genannte Angriff setzt im ersten Schritt eine Charakterisierung der angegriffenen Implementierung mit bekannten Parametern voraus. Dies stellt eine deutliche Einschränkung für Angreifer dar. In einem weiteren Teil meiner Arbeit stelle ich in diesem Zusammenhang vor, wie man sogenannte Cluster Algorithmen verwenden kann, um Seitenkanalinformationen von Exponentiationsalgorithmen auszunutzen, ohne diese zuvor zu charakterisieren. Die Idee der Anwendung von Cluster Algorithmen ist verwandt mit der Arbeit von Walter, kann im Gegensatz dazu aber unter verschiedensten Umständen sowie zur Ausnutzung von verschiedenen Seitenkanalinformationen angewandt werden und basiert außerdem auf einer etablierten Fachrichtung. In einem praktischen Versuch gelang ein vollständig erfolgreicher Angriff auf die Implementierung elliptischer Kurven Kryptographie unter Ausnutzung der ortsabhängigen Seitenkanalinformation.

Die Erfolgswahrscheinlichkeit solcher Angriffe hängt maßgeblich von der

Qualität der Seitenkanalmessung und der darin enthaltenen Information ab. Aufgrund der gezeigten Ortsabhängigkeit unterscheiden sich Messungen an verschiedenen Messpunkten stark. Mit einer Messung des Magnetfeldes an mehreren Messpunkten gleichzeitig kann zusätzliche Information gesammelt werden. Als weiteren Teil meiner Arbeit zeige ich, wie im Rahmen des zuvor beschriebenen Angriffs mit Hilfe von Cluster Algorithmen mehrere gleichzeitige Messungen genutzt werden können. In einem praktischen Versuch habe ich mehrere Messpunkte in einer geometrischen Anordnung verwendet und gemeinsam ausgewertet. Der Versuch hat gezeigt, dass die Kombination von Messungen zu einem vollständig erfolgreichen Angriff führt, selbst wenn die einzelnen Messungen unzureichende Seitenkanalinformation beinhalten weil keine Charakterisierung durchgeführt wurde, um die Messpunkte zu wählen. Eine wichtige Schlußfolgerung ist nun, dass nicht notwendigerweise geeignete Messpunkte gesucht werden müssen, sondern stattdessen eine Kombination von mehreren Messpunkten vorgezogen werden kann.

Zusammenfassend zeige ich Stärken und Schwächen von hochauflösenden und örtlich-begrenzten Messungen des elektromagnetischen Feldes für die Seitenkanalanalyse, und wie ortsabhängige Seitenkanalinformationen für spezifische Angriffe auf Implementierungen von asymmetrischen kryptographischen Algorithmen verwendet werden können. Außerdem zeige ich die Anwendung von Cluster Algorithmen für Angriffe und die Möglichkeit bei deren Anwendung gleichzeitige Messungen zu kombinieren.

# Acknowledgements

I sincerely appreciate the past years of supervision and promotion by my dissertation adviser Prof. Dr.-Ing. Georg Sigl. I would like to thank my second examiner Prof. Dr.-Ing. Christof Paar.

I would also like to thank my highly appreciated colleagues from Fraunhofer AISEC, Technische Universität München, and Infineon Technologies AG for collaboration on scientific publications, support and opportunities. Especially so, I would like to thank Dr. techn. Stefan Mangard for his valuable scientific supervision and guidance, Dr. rer. nat. Frederic Stumpf for supervising my scientific work from the very beginning, Dr.-Ing. Andreas Ibing for introducing me to the field of pattern classification, Benedikt Heinz, Dominik Merli and Fabrizio De Santis for discussions, collaboration on publications and support, Gerald Holweg, Walter Kargl and Prof. Dr. rer. nat. Claudia Eckert for giving me opportunities, Dr. rer. nat. Guido Stromberg and Günter Hofer for encouraging me to pursue a doctorate, and Robert Hesselbarth as well as Konstantin Böttinger for proof-reading my dissertation.

# Nomenclature

AES ..... Advanced Encryption Standard

BER ..... Bit Error Rate

CPA ..... Correlation-based Power Analysis

DCA ..... Differential Cluster Analysis

DFA ..... Differential Fault Attack

DLP ..... Discrete Logarithm Problem

DPA ..... Differential Power Analysis

DSA ..... Digital Signature Algorithm

ECC ..... Elliptic Curve Cryptography

ECDLP ..... Elliptic Curve Discrete Logarithm Problem

ECSM ..... Elliptic Curve Scalar Multiplication

ITMIA ..... Itoh-Tsujii Multiplicative Inverse Algorithm

LDA ..... Linear Discriminant Analysis

LUT ..... Look-Up Table

MIA ..... Mutual Information Analysis

PCA ..... Principal Component Analysis

RSA ..... Rivest Shamir Adleman

SNR ..... Signal-to-Noise Ratio

SPA ..... Simple Power Analysis

# List of Figures

# List of Tables

# List of Algorithms

# Contents

# Chapter 1

# Introduction

The need for information security in daily life is increasing continuously. We rely on electronic information systems for monetary transactions, identification purposes and communication and many other purposes. We buy goods via internet portals, access our bank accounts to authorize transactions via internet, and perform daily payments using electronic bank cards or credit cards. We use electronic passports which are safer against counterfeit and use electronic car keys and building access tokens. We communicate using mobile handsets and voice over IP. In all such applications, information security is crucial and established through information security engineering and, ultimately, cryptography.

Information security denotes among other properties the confidentiality, integrity and availability of information. In most of the mentioned applications, the communication channels, or devices, are accessible to people who shall not be able access our personal information either constantly, or during a certain time. This personal information may be our bank account number, authorization codes, or identification information which could be of value to opponents. In case of communication over the internet for instance, a third party could eavesdrop at every single routing node on a connection between two communicating parties.

Cryptography, or more specifically, cryptographic algorithms are used to ensure confidentiality as well as integrity of information. Early cryptographic algorithms date back to ancient times. The Caesar cipher [PHS03] is a simple substitution cipher where the letters in the alphabet are substituted. It could provide confidentiality against simple adversaries from ancient times when a message was carried by a herald over long distances and interception had to be expected. A long history of improvements, cryptanalysis and defeats of cryptographic algorithms followed. Decades of research in cryptography during the late 20th century lead to a selection of algorithms, which are

cryptanalytically secure. But this is not the end of the story.

Cryptanalytic security was exclusively important until the 90s, when side-channel analysis, or more general, physical cryptanalysis emerged as a major threat. Physical cryptanalysis refers to analyzing and breaking cryptographic security by using physical, or implementation aspects of secure devices. The reason why this emerged is that devices which are used for information security started to be increasingly embedded, pervasive and, thus, accessible. Think of electronic passports, credit cards, and mobile phones for instance where it is obvious that such devices may get into the hands of adversaries. The most popular part of physical cryptanalysis is *side-channel analysis*. As an illustrative example, safe-breakers which use a stethoscope in order to listen to the sounds of a mechanical lock can be mentioned. This approach can be denoted as a side-channel attack, however, does not have anything to do with cryptography. In cryptographic side-channel attacks, adversaries observes the physical properties of devices during a cryptographic operation, e.g., the time consumption, current consumption, or electromagnetic field, to recover the secret cryptographic key material using visual inspection or statistical evaluation methods. Side-channel attacks became known after Kocher published the first article in 1996 [Koc96]. In the years since then, there have been over 700 published side-channel attacks [Wag12]. This number is still increasing and, therefore, physical implementation security is equally important as the mathematical security of cryptographic algorithms today. Chapter 2 provides a longer introduction into information security, cryptography and physical cryptanalysis.

In this thesis, I investigate *high-resolution electromagnetic field measurements* of cryptographic implementations which are used for side-channel analysis. I concentrate specifically on *localized* measurements, which means that the measurements are restricted to a certain spatial extent. Previous publications have either advertised this without practical results [GMO01], concluded that localized measurements of electromagnetic fields are impossible [SGM09], or have shown unconvincing results with coarse localizations [KS11].

In the first main Chap. 3, I present the results of an extensive study about the strengths and limitations of high-resolution electromagnetic field measurements for side-channel analysis. The background of electromagnetic fields in general as well as their application in side-channel analysis is also introduced in this chapter. I am able to clearly demonstrate the feasibility and quality of localized measurements. Additionally, I derive several conclusions about the measurement setup and processing of traces. This resulted in a contribution, *Strengths and Limitations of High-Resolution Electromagnetic Field Measurements for Side-Channel Analysis* to the CARDIS conference

in 2012 [HMH+12b].

After establishing the feasibility of localized measurements of electromagnetic fields, I present how such localized measurements can be used to extract dedicated *location-based side-channel information leakage* from certain cryptographic algorithms in Chap. 5. I describe, how popular exponentiation algorithms exhibit location-based side-channel leakage, which had been unknown previously. To demonstrate this I use a hardware implementation of an elliptic curve scalar point multiplication algorithm for a practical evaluation which I describe in Chap. 4. Chapter 4 also contains an introduction into elliptic curve cryptography and a survey about the state-of-the art in protecting implementations against physical cryptanalysis. The presented work resulted in the contribution, *Localized EM Analysis of Cryptographic Implementations* to the CT-RSA conference in 2012 [HMH+12a].

As a further contribution, I present a *non-profiled* side-channel attack which is able to successfully exploit the location-based side-channel leakage from a single measurement in the subsequent Chap. 6. This attack applies well-researched unsupervised cluster classification algorithms to recover the secret exponent, or scalar and does *not require profiling*, such as the generation of templates. I perform a practical experiment and demonstrate a successful attack.

The success probability of such single-execution attacks depends on the quality of the measurements. A concurrent measurement of side-channel leakage at different positions during a single execution leads to more recovered information which can be combined. I show how *multiple simultaneous measurements can be combined in the presented non-profiled single-execution attack based on unsupervised clustering* in Chap. 7. Again, the described method is evaluated successfully in a practical attack on an FPGA-based ECC implementation using high-resolution measurement equipment for electromagnetic fields.

In the last Chap. 8, I draw conclusions and mention topics for future work.

# Chapter 2

# Background

This chapter serves as an introduction into information security in general, cryptography, as well as applied cryptography and physical cryptanalysis. More background information is provided within the subsequent chapters when relevant along with related work. Chapter 4 for instance presents the background of elliptic curve cryptography and Chap. 3 includes the background on electromagnetic fields. I start with a discussion of information security and cryptography in Sect. 2.1. Then, in Sect. 2.2, I introduce the field of physical implementation security of cryptographic algorithms including side-channel analysis and fault attacks.

## 2.1 Information Security and Cryptography

Information security denotes the confidentiality, integrity and availability of information and is important in cases of personal as well as public interest. Nowadays, information is stored in electronic devices and transmitted through electronic data connections which are, in theory, also accessible to people with no right to access the contained information. If for instance the internet is used as a transport mechanism for information, it could be eavesdropped at every single routing node on a connection between two communicating parties. Cryptography, or more specifically, cryptographic algorithms are used to ensure confidentiality as well as integrity of information. Early cryptographic algorithms date back to ancient times like for example the Caesar cipher [PHS03]. The Caesar cipher is a simple substitution cipher where the letters in the alphabet are substituted. It could provide confidentiality against simple adversaries from ancient times when a message was carried by a herald over long distances and interception had to be expected. A long history of developing better cryptographic algorithms, cryptanalysis

and also breaking some of them along the way followed.

Kerckhoffs postulated a principle in 1883 [Ker83] which was later regarded as the most important basic principle in cryptography. It states that the security of a cryptographic system must only depend on one input, the secret key, a number from a certain number space, while the cryptographic algorithm itself is public. The public access encourages scientific analysis of the algorithm. If a cryptographic algorithm is still secure under this postulation, it can be used between different entities and a lost secret key can easily be replaced by a new one. This is extremely important when using cryptography for information security because it supports widespread usage.

From a historical perspective, symmetric cryptography is prevailing. Symmetric cryptography is also called private key-, or secret key cryptography. One secret key is used for encryption as well as for decryption and is shared between two or more parties. This requires confidential key establishment. The historical examples in the next section below belong to this class of cryptographic algorithms.

### 2.1.1 History and Security Levels

Information security is generally defined under the black-box security model. In this model, the algorithm is known to the adversary and the adversary may access the device to choose inputs and read outputs. In this model, the security of cryptographic algorithms is usually assessed by the computational effort which is required to break the cipher.

The required computational effort can for instance be reduced by analytical observations, which reduce the computational effort, or search space to break the secret key. Ancient ciphers such as the Caesar cipher were only exposed to the computing power of a human being. In this respect they could provide acceptable security at the time.

Later examples of cryptographic systems include the famous electromechanical Enigma machine which was invented by Arthur Scherbius in 1918 [KD02] and used by the German military information service during the second world war. Driven by the high value of the protected information during the submarine war in the northern atlantic, enormous effort was put into the cryptanalysis of the machine's mechanical algorithm. This and significant improvements in computing power lead to a break of the original Enigma and later, also the improved version [PHS03].

With the exponential development of the computational power of electronic computers, the demands for the security of cryptographic algorithms are compelled to increase at the same pace. The available computational power is a measure of information security because it relates to the time it

takes to perform the operations which are required to break an algorithm. Besides computational advancements, continuous improvements in the cryptanalysis of algorithms reduce the number of operations which are required to break them. It took many decades until the mathematical, or cryptanalytical security of cryptographic algorithms reached a level which is acceptable today and safe against computational threats for years.

This security level can actually be expressed by the number of operations which have to be performed in order to break a system to recover a key. This does not necessarily equal the bit-size of employed parameters. For instance Elliptic Curve Cryptography (ECC) with a secret key size of 224-bit has a security level of 112-bit, and AES with a key size of 128-bit has a security level of 128-bits. At a security level of 128-bits, an adversary must perform an estimated $2^{127}$ operations to compromise the security with a success probability of 50% [ANS05, p. 6]. An acceptable security level today are at least 80-bit [MOVR01].

A recent example for a cryptographic algorithm which is now considered insecure due to the available computational power and modern cryptanalysis is the DES algorithm which employs secret keys of only 56 bits. This means that only $2^56$ keys have to tried to find the correct one which is within reach of special purpose computing systems. DES was replaced by the AES [NIS01] algorithm which is state-of-the-art in symmetric cryptography.

### 2.1.2 Asymmetric Cryptography

In 1978 Rivest, Shamir et Adleman [RSA78] published the RSA algorithm. RSA is an asymmetric cryptographic algorithm, or public key cryptographic algorithm. Contrary to previous, symmetric cryptographic algorithms, this cryptographic algorithm was the first to build upon number theory.

Cryptographic schemes like authentification, signatures, or key agreement which we use everyday can only be build upon asymmetric cryptographic primitives. A key pair of two different keys which are mathematically related are used. One of them, the public key, can be distributed to the public. The other one, the private key, must be kept secret by the owner of the key pair. The most popular asymmetric algorithms rely on number theory and the private key can only be derived from the public key by breaking a hard mathematical problem. There are two number-theoretic problems which the most important algorithms build upon:

- The *integer factorization problem* for large integers. The hardness of this problem lays the ground for the security of the RSA cryptosystem. The factorization of the public key equals the private key and is often

described as a trap-door to invert the basic one-way function of the RSA cryptosystem.

- The Diffie-Hellman, ElGamal, DSA and ECC cryptosystems are based on the hardness of the *discrete logarithm problem* which means to find the discrete logarithm, thus, secret exponent, given a base and result. The exponent equals the secret key.

### 2.1.3 Cryptanalysis

Cryptanalysis is the analysis of cryptographic systems with the aim of reducing their security up to breaking the system entirely and recovering the secret key. It is performed using methods from various fields of expertise and also includes the use of computational power. Cryptographic algorithms which are based on number theoretic problems are attacked using mathematical theorems. Other cryptographic algorithms are attacked using dedicated cryptanalytical methods such as differential cryptanalysis.

Cryptanalytic security means that algorithms are secure against adversaries in a black-box attack setting. In this setting, the adversary may access, or choose, input as well as output data but is unable to access intermediate values during computation. However, following Kerckhoffs' principle, the adversary does not know the secret key he is trying to break.

After many years of cryptanalysis, AES, RSA, and ECC are regarded as cryptanalytically secure. While AES provides a large security margin today, the parameter sizes for RSA and ECC are steadily increasing to comply with growing computational power.

## 2.2 Physical Cryptanalysis

Cryptanalytic, or mathematical security was exclusively important until the 90s, when side-channel analysis, or more general, physical cryptanalysis emerged as a big threat. Physical cryptanalysis refers to analyzing and breaking cryptographic security by using physical, or implementation aspects of secure devices. This field was more or less created after Kocher published Simple Power Analysis (SPA) and timing attacks [Koc96]. Nowadays, the physical implementation security against physical cryptanalysis is as important as the mathematical security of cryptographic algorithms.

Physical cryptanalysis changes the adversarial model from a black-box model to a grey one, where the adversary has access to certain physical characteristics of the device during cryptographic operations. In such cases, an

adversary circumvents the black-box scenario and recovers information from, e.g., physical side-channels during the cryptographic computation. Such approaches actually date back very far. A well-known example are safe-breakers which use a stethoscope in order to listen to the sounds of a mechanical lock. However, this did not have anything to do with cryptography then.

Nowadays, electronic devices which are used to provide information security by means of cryptographic algorithms are embedded, portable, and often physically accessible to adversaries. Credit cards or electronic passports are popular examples and it is obvious that such devices may get into the hands of adversaries. This means that an adversary may measure, or alter the physical properties of the secure device with the goal of extracting the contained secret information which would enable him to forge a copy of the device for instance. Therefore, the computer chips in such embedded devices include dedicated, so-called, smartcard chips to protect information security. Another application of such chips which are dedicated for embedded information security are embedded computing platforms in mobile handsets for instance which are becoming increasingly complex. It is a common strategy for such embedded platforms, to use one distinct encapsulated element, a smartcard chip, as a provider for all security mechanisms. *Therefore, an increasing demand for highly secure elements and for research and development in countermeasures against physical cryptanalysis is already observed and can be expected in future.*

This section starts with a differentiation, short history, and classification of physical cryptanalysis. The main parts concentrate on side-channel analysis in Sect. 2.2.1, fault attacks in Sect. 2.2.3, along with corresponding countermeasures in Sect. 2.2.2 and Sect. 2.2.4.

### Classification of Cryptographic Physical Attacks

The field of physical cryptanalysis can generally be divided into:

1. *Passive side-channel analysis*
   The terms side-channel analysis and side-channel attacks are used in the same way because analysis is mostly done during actual, or simulated attacks. In passive side-channel analysis, an adversary observes the physical properties of a secure device during a cryptographic operation passively. The observation may for instance include the time duration of the operation, the current (power) consumption of the device, or the electromagnetic field of the device. The adversary then recovers information about the secret key from those observations using visual inspection or statistical evaluation methods.

2. *Active fault attacks*
   In active fault attacks, an adversary deliberately injects faults into the device during a cryptographic computation. From the behavior of the device as a result to the fault injection, or from the faulty outputs, the adversary is able to recover information about the secret. Simple fault attacks just alter the control flow of the device while more sophisticated attacks employ faults in the processed data.

### History

Physical cryptanalysis became a widespread concern after the publication of Simple Power Analysis (SPA) and timing attacks by Kocher [Koc96] in 1996. This was the first public report on passive side-channel analysis of cryptographic algorithm implementations. The beginning of physical cryptanalysis also includes the first fault attack by Boneh et al. [BDL97] in 1997.

Kocher et al. [KJJ99] published the classical version of Differential Power Analysis (DPA) in 1999 which is still the most powerful side-channel based cryptanalytic method. Mangard et al. [MOP07] wrote a reference book for side-channel analysis in 2007.

### Physical Attacks which are Unrelated to Cryptography

TEMPEST is the codename for an operation by the U.S. National Security Agency which aimed at securing electronic communications equipment against eavesdropping via magnetic- or electric field radiation in the late 1960s. While such attacks can be considered as physical attacks against devices which protect information security, such attacks have nothing to do with cryptography. Hence they are not within the physical cryptanalysis category.

There are also invasive attacks on secure devices which have nothing to do with cryptography, hence, do not fall under the category of physical cryptanalysis. Examples for such attacks, include reverse engineering of integrated circuits using using light- or electron-microscopy and software tools. The aim of reverse engineering is to gain knowledge about where sensitive information is stored, or handled. Then, probing or forcing attacks with sophisticated needle equipment lead to eavesdropping on, e.g., secret key material. This sometimes requires the use of a focused ion beam in order to remove, or bridge active shielding from a smartcard device. With the same focused ion beam, connections can be made and pads to establish a needle contact can be build.

### 2.2.1 Passive Side-Channel Analysis

Side-channel analysis falls under the non-invasive category because it does not generally require modification of the attacked device. There are exceptions, where semi-invasive methods like removing plastic packaging lead to better side-channel measurement results.

Passive side-channel analysis exploits information which is leaked through physical side-channels such as the time consumption, power consumption, or electromagnetic radiation. All side-channel attacks on cryptographic devices require, that the adversary knows which algorithm is implemented. This knowledge may be derived in a preceding step before the actual attack. The next section presents different sources of information leakage through side-channels.

#### Sources of Information Leakage

The first discovered side-channel was the amount of time a computation requires. If the timing of cryptographic computations, or parts of such computations depend on secret data, an adversary can recover information about the secret data by analyzing the timing.

However, the most important side-channel is the power, or more precisely, the current consumption of a cryptographic device. Integrated digital circuits are predominantly implemented as complementary metal oxide semiconductors. Such circuits have a small static power consumption and a bigger part, which is the dynamic power consumption. The dynamic power consumption mostly consists of switching currents. Switching currents occur, when outputs of logic gates change their value. A value change requires charging, or de-charging of an output capacitance which consists of the wire capacitances and the input capacitances of the connected gates. In a digital hardware circuit, there are gates which change their value at regular intervals, such as the clock supply network, and there are parts which change their value according to the functionality of the circuit.

Usually, digital circuits can be segmented into control-path parts and data-path parts. In both cases, gates change their output values according to the design's functionality. The value changes, thus, charging currents depend on the values which are processed and the operations which are performed. Both, processed data and performed operations may depend on the secret data. Hence, the current consumption is a physical source of information leakage. Passive side-channel attacks aim at exploiting such data-dependent currents, thus, information leakage.

An important aspect is the ratio of the measured amplitudes of the secret-

dependent currents which are to be exploited and the measured amplitudes of currents which are due to non-secret-dependent switches and measurement noise. This signal-to-noise ratio significantly influences the success probability of side-channel attacks.

Electric currents generate proportional co-centric magnetic fields and radial electric fields around them. An adversary may measure the current consumption of the device, or the electric as well as magnetic field to gain information about the processed data values. Side-channel attacks which can be applied to current consumption measurements can as well be applied to measurements of electromagnetic radiation. This was established by Quisquater et Samyde [QS01].

However, contrary to currents in wires which can be described using scalar values, electric and magnetic fields are vector fields in three-dimensional space. In this way, electromagnetic fields provide a richer source of information than current flows and specific properties can be used for attacks. This is discussed in greater detail in Sect. 3.1 and is the main target of research for this thesis.

### Side-Channel Measurement

The timing of a computation can be measured using an oscilloscope, or sometimes, simply using the built in PC clock.

The current consumption of an integrated circuit is typically measured by inserting a small measurement resistor with a resistance of $\approx 10$ Ohm into the power supply [MOP07]. The smaller the resistance, the smaller the implication on the equivalent of the source's inner resistance. The resistance must, however, be large enough to cause a measurable voltage drop. The voltage drop over the resistor is proportional to the supply current and, thus, also proportional to the power consumption of the device. In most cases, a measurement in the ground supply line is preferable, since a common ground level for the laboratory equipment and simple passive probes can be employed.

Electromagnetic fields or radiation are measured using hand-made, or commercially available magnetic coils and electric field probes. This is addressed in more detail in Sect. 3.1.2. Those measurements usually require amplification and are usually afflicted with a higher noise level. However, measurements of electromagnetic fields provide significant advantages for side-channel analysis of cryptographic implementations. This is discussed in this thesis.

A side-channel measurement setup will always include a digital sampling oscilloscope and computer-controlled recording and storing of data.

In some cases, trace compression is used to reduce the amount of recorded data to save storage space or computation time. Such methods would for instance only store the highest values during every clock cycle. However, trace compression reduces the contained information and may not lead to meaningful results.

### Classification of Side-Channel Attacks

Side-channel attacks can be classified according to the following properties:

1. *Number of observed executions*
   Depending on the application and cryptographic protocol, an adversary may be restricted to a single observed execution or be allowed to observe multiple executions with the same secret.

2. *Leakage characteristic*
   An important property of side-channel attacks is whether the adversary is allowed to precisely characterize the leakage of a device. He can use the same or a similar device to do so. Other attacks use a heuristic model of the side-channel leakage instead, or require no model to be used at all.

3. *Number of stochastic variables*
   The number of stochastic/statistic variables which are used in the attack is another important property as well as the assumption about how the variables, which can for instance be samples from a measurement, are distributed. The common assumption is a Gaussian distribution.

In the following, I first give an overview about the properties of a few important side-channel-attacks and then discuss them in more detail.

Simple Power Analysis (SPA) [KJJ99], template attacks [CRR03, ARR03], and algebraic side-channel attacks [RS09] recover the secret key by exploiting the observation of only a single-execution. Regarding implementations of asymmetric algorithms, an adversary may only observe a single execution with the same secret in many cases. This is due to the employed protocols or countermeasures.

Template attacks [CRR03, ARR03] as well as the stochastic approach by Schindler et al. [SLP05] characterize the leakage function of a device during a profiling phase on a fully accessible, identical device and using many observations. The attack itself is also performed on a single observation and is multi-variate because many samples of the observation are considered at the same time.

Differential Power Analysis (DPA) [KJJ99], Correlation Power Analysis (CPA) [BCO04], Mutual Information Analysis (MIA) [GBTP08, BGP+11], and robust SCA [DPRS11] distinguish the secret key by observing multiple executions with constant secret and use a heuristic model of the device's leakage. Those attacks are usually uni-variate.

Side-channel-based, internal collision attacks [SWP03, SLFP04] require neither profiling nor a leakage model but assume that processing the same values leads to similar leakages. Such attacks are possible in a single observation as well as a many observations context.

### Timing Analysis

Timing attacks exploit data-dependent variances in the computation time of a cryptographic device. Most exponentiation algorithms for public key cryptography process the secret in small parts. This fact lead to the first published timing attack of Kocher [Koc96]. This attack targets the square-and-multiply exponentiation computation which is used for Diffie-Hellman and RSA cryptosystems. This algorithm processes the secret exponent bit-wise and, depending on the bit's value, either a square, or a square-and-multiply operation is performed. Time measurements allow an adversary to exploit the different computation times of both cases and to recover the secret exponent.

Timing attacks also apply to implementations of symmetric cryptographic algorithms when operations are data-dependent. This was described by Koeune et al. [KQQ99] for the case of AES. An important general source of timing leaks in software implementations is the cache. This was first described by Bonneau et Mironov [BM06].

### SPA

In Simple Power Analysis (SPA), an adversary aims at deriving the secret directly from the trace. A single trace, or only a few traces are employed and in most cases, a visual inspection is sufficient to recover the secret. This implies that the information leakage is significant. SPA is an important threat to implementations of public key cryptography. Kocher et al. [KJJ99] demonstrate this on the example of exponentiation algorithms which is similar to Kocher's timing attack [Koc96]. If the adversary is able to distinguish between squares and multiplies in the power trace, he can recover the secret exponent. Implementations of symmetric cryptography mostly employ constant program flows which are independent of processed data values.

### DPA, CPA, MIA

Differential Power Analysis (DPA) is the generic term for a family of similar attacks. DPA uses multiple measurements with varying input values and exploits differences using statistical methods. It requires that the secret remains equal over those multiple executions of the cryptographic algorithm. In this way, low signal-to-noise-ratios of the information leakage per observation can be overcome by simply increasing the number of employed observations. Resistance of an implementation against SPA and timing attacks does not prevent DPA. DPA usually targets implementations of symmetric cryptographic algorithms.

During all DPA methods, an adversary uses information about the algorithm and known input values, or alternatively output values. He then attacks an intermediate value of the algorithm which depends on known input/output data and small parts of the unknown key. Based on guesses about the value of this part of the key, he can hypothetically compute the intermediate value for every observation. If this intermediate value is computed as a non-linear function of the guessed key parts, a key guess with small errors will lead to significantly wrong intermediate values.

A leakage model is used to derive a hypothetical power consumption value from an intermediate value. The original DPA from Kocher et al. [KJJ99] uses a single bit model where the assumption is that the power consumption is different for 1- and 0-values of one bit. The attack is completed by checking, if this assumption fits to the measurements using a statistical difference-of-means test [MOP07]. Since every such bit which is used in the end depends on, e.g., one complete byte of input data and one complete key byte, it still reveals a complete key byte.

A later attack in the DPA class is Correlation-based Power Analysis (CPA) [BCO04]. This attack uses more than one bit of the intermediate value. Hence, it is necessary to find a function which maps multi-bit values to hypothetical power consumption values. The most popular leakage models are the Hamming weights of values or Hamming distances of successive values. Using a leakage model, an adversary derives power values, which depend on the guessed key. Those power values are very inaccurate estimations of the real power consumption when interpreted as absolute values. However, the relative differences in power consumptions usually fit well to real measurements. A statistical distinguisher such as the Pearson correlation coefficient is used in the next step. For the correct key guess, this distinguisher reveals correlations between estimated, hypothetical power differences and the actual measurements. By repeating this for different key guesses exhaustively, the correct key is distinguished.

Classic DPA attacks [MOS09, KJJ99, BCO04] assume a Gaussian distribution of the leakage observations and employ most likelihood estimations.

Mutual Information Analysis (MIA) [GBTP08, BGP+11] does not make any assumptions about the distribution of measurement values and is able to detect arbitrary kinds of leakages. It is based on the joint entropy of intermediate values and measurement values and MIA is regarded as an information-based distinguisher. However, it requires an extensive amount of computation which rather makes it a valuable tool for design verification than for actual attacks.

#### Template Attacks and Stochastic Modeling

In template attacks [CRR03, ARR03] and when employing the stochastic approach [SLP05], the leakage characteristic of a cryptographic device is profiled before the actual attack. This of course requires, that an adversary has access to an identical, or even the same device for profiling. Templates must usually be built for every value of the secret key or even every combination of secret key and input value [MOP07]. Many observations with completely known inputs and secrets are used in order to build multi-variate Gaussian templates. During the actual attack, a single observation is matched to the templates by for instance using a sum-of-squared-error matching criterion.

In the stochastic approach, the coefficients of a stochastic model of the leakage of the device are recovered using linear regression. This results in one model, where the base functions of the input and secret data are linearly combined. This model is matched against one observation during the attack to recover the secret.

#### Side-Channel-Based Collision Attacks

Regular collision attacks are a cryptanalytical tool and target output values of algorithms [MOVR01]. Side-channel-based, internal collision attacks [SLFP04, SWP03, MME10, Bog08] are based on distinguishing equal intermediate values during cryptographic computations without leakage modeling or profiling. These collisions depend on secret keys and known input values. Combined with analytical properties of the attacked algorithm such internal collisions allow to recover the secret.

### 2.2.2 Side-Channel Countermeasures

Protection against side-channel attacks can be introduced on different abstraction levels of an application or cryptographic device.

1. Protocol-level
2. Algorithm-level
3. Implementation-level (source-code level)
4. Gate-level
5. Transistor-level

Usually the design, or cost effort is higher in lower abstraction levels. However, application properties, standardization, or licensing fees may restrict the possibilities on higher abstraction levels and force protection at low levels.

### Classification of Side-Channel Countermeasures

There are two general approaches for side-channel countermeasures:

1. *Prevent that the adversary collects sufficient information about a secret.* Such principles are relevant in the upper abstraction layers of a secure system. In the case of ECC-based protocols this is an inherent feature. They change their secret scalar (exponent) in every execution, thus, limit the the information leakage to a single observation. However, the same principle can be used to modify protocols based on symmetric cryptographic algorithms so that the key is also changed frequently. This idea was filed as a patent in 1999 by Kocher [Koc03]. Methods which use this approach are currently a topic of high research activity under the term leakage resilience.

   If an attack, however, can be successful using only single observations, this approach does not help.

2. *Make the side-channel observations independent of the processed data.* In cases, where the information leakage of secret data cannot be restricted on the upper abstraction levels, this is the only way to proceed. There are two major methods:

   (a) *Hiding* refers to reducing the signal-to-noise-ratio of the exploited leakage signal. This can be achieved by introducing additional, possibly superficial, noise or by reducing the leakage signal. In amplitude-based hiding, additional noise is for instance introduced by noise engines. Other forms of hiding include time-based hiding. In this case, operations or data values which are possibly targeted by an adversary for exploitation are misaligned in the time

domain. This prevents the adversary from being able to combine the leakage during multiple observations at the same time. Examples include randomizing of the operation sequence, introducing no-operation cycles, or jitter in the clock frequency [MOP07].

A general downside of hiding countermeasures is that low signal-to-noise-ratios can generally be coped with by for instance using more measurements or better measurement equipment. A simple example for this is the averaging of repeated measurements with equal input values to reduce amplitude-based hiding.

The probably most important aspect of hiding is to strictly avoid data-dependent operation sequences. The reason is that data-dependent operation sequences or timings can be exploited particularly easy using SPA, or timing attacks.

Differential logic styles aim at equalizing the current consumption so that different values result in equal current consumption. Such measures reside on the lowest abstraction levels of an implementation and require a significant design and cost effort. Kirschbaum [Kir11] provides an extensive study of PA-resistant logic styles.

(b) *Masking* refers to preventing data-dependent information leakage by changing the values of the processed data values. A mask value is used and combined with the actual data value. Under the assumption that the mask value is unknown, the processed, masked value is not related to the original value anymore.

Therefore, side-channel attacks targeting such a value are prevented. Masking can be performed on different abstraction layers. So-called second order attacks target masked implementations and try to combine the leakage of the mask and the masked value in order to achieve a signal which contains information about the original value.

### 2.2.3 Active Fault Attacks

Fault attacks are categorized into non-invasive, semi-invasive, and invasive attacks [KK99].

1. *Non-invasive fault attacks* use methods which do not require any modification of the device. Examples are glitches or spikes on the voltage supply as well as malformed clock supply curves. Those methods produce faults which cannot be restricted to parts of the device.

2. *Semi-invasive fault attacks* require decapsulating the integrated circuit in order to be able to use fault induction by optical means. Methods include the application of focused laser light which usually supports to restrict the fault induction to a confined area.

3. *Invasive fault attacks* require permanent alterations of an integrated circuit. Chemical etching, focused ion or laser beams are examples for methods which are used to access a circuit for a fault attack. The generated fault in the circuit operation can still either be permanent or transient.

### Fault Model

Fault attacks require a certain outcome of induced faults in order to be able to recover the secret through its exploitation. The outcome of a fault induction can be described through a fault model. A fault model may describe the abilities of an adversary as well as requirements for a certain attack. The easier it is for an adversary to achieve a certain fault model, the more powerful is the attack which relies on this model. A fault model includes the following properties [KOP10]:

- *Affected bits* - The fault may affect specific, arbitrary, single, multiple, or all bits within the device.
- *Effect* - The affected bit's values may be toggled, set to a fixed, or inconclusive value.
- *Permanence* - The effect of the fault on the affected bits may be permanent or transient, thus impacting only the value of the current cycle.
- *Timing precision* - The fault induction may be performed with arbitrary, or accurate timing precision.

### Classification

Fault attacks can generally be assigned to the following categories:

- *Differential fault attacks*
- *Safe-error fault attacks*
- *Algebraic fault attacks*
- *Control flow fault attacks*

### Differential Fault Attacks

Differential Fault Attacks (DFAs) have been described for the first time by Biham et Shamir [BS97] in 1997. DFAs are mostly relevant for implementations of symmetric cryptographic algorithms, however, there are also DFAs on implementations of asymmetric algorithms.

DFA attacks use the differences between correct and faulty outputs of a cryptographic device to derive information about the secret. The output ciphertext and knowledge about the algorithm is used. The plaintext input may remain unknown. However, the attack requires that the same plaintext is used multiple times to be able to gain correct and faulty ciphertext pairs with the same plaintext. Equations are derived from the algorithm and assumptions about the fault model. The ciphertext pairs and parts of the unknown secret key are the unknowns in the equations. The unknown key parts are exhaustively trialled. The correct secret key parts are solutions to the equations under the condition that the assumed fault model is satisfied. Hence, the secret key is revealed.

Published fault attacks mainly differ in the number of required ciphertext pairs and the assumed fault model. The most powerful DFA on implementations of the AES algorithm is from Saha et al. [SMR09]. It is an extension of an earlier attack of Piret et Quisquater [PQ03]. The fault model allows the adversary to be very imprecise in comparison to the model of earlier DFAs. Furthermore, only one pair of correct and faulty ciphertext is required.

### Safe Error Fault Attacks

Safe error attacks were proposed by Yen et Joye [YJ00] in 2000 and observe whether an injected fault actually propagates through the circuitry and influences the result. If this is not the case it is a 'safe' error.

There are two types which are both exceptionally relevant for asymmetric cryptographic algorithms. The Computational (C) safe-error attack tries to exploit dummy operations. This is for instance the case with dummy point additions in the double-and-add-always algorithm [Cor99] for the elliptic curve scalar multiplication. The adversary injects a fault into a conditional operation and observes whether this operation was a dummy operation. Based on this he recovers the value of one bit. This is repeated for every loop iteration to recover the entire secret, e.g., scalar in the case of ECC.

The Memory (M) safe-error attack exploits the fact that some faults are overwritten under certain conditions.

Safe fault attacks only require the information whether there has been an error during the computation or not. Therefore, they can not be gen-

erally prevented through error detection mechanisms. However, algorithmic countermeasures like e.g., scalar randomization for ECSMs serve as attack prevention. Scalar randomization leads to a different scalar in every execution and, thus, prevents an incremental, bit-by-bit recovery of the secret as it is employed in safe-error attacks.

#### Algebraic Fault Attacks

Algebraic fault attacks use algebraic properties of a crypto-system to exploit induced faults and therefore, only apply to asymmetric cryptography. The most popular algebraic attack is the one on RSA implementations which uses the Chinese remainder theorem by Boneh et al. [BDL97].

Another example for algebraic attacks are attacks on implementations of ECC [CJ05] which use faults to move the elliptic curve scalar multiplication from a cryptographically strong elliptic curve to a weak curve where the ECDLP is easier to solve. This can be achieved through changing the base point $P$ (invalid point attack [BMM00]), the curve parameters (twist curve attack or invalid curve attack [CJ05]) or by inducing a fault into the intermediate values during the computation.

#### Control-Flow Fault Attacks

Fault attacks on the control flow of the design try to skip the fault detection or jump to the final state during execution [SH08]. In this way, intermediate results may for instance be read on the output which allow the adversary to recover parts of the secret.

### 2.2.4 Fault Attack Countermeasures

To prevent successful fault attacks, designers may include various countermeasures into cryptographic devices. There are generally three classes of countermeasures:

- Detection or prevention of physical fault injection.
- General detection of errors which follow from fault injection.
- Dedicated, algorithmic fault detection and prevention.

Physical detection of fault injection can be done by using voltage, frequency, or light sensors for instance [KK99]. They target the problem at the time of injection and a device would immediately stop sensitive computations and delete all sensitive data after detecting a fault attack. Similarly, fault attacks

can be prevented or made more difficult by using metal meshes covering to surface of an integrated circuit [KK99].

General error detection measures target the errors which followed the fault injection. Employing duplication on different granularity levels provides a good error detection capability. Such concepts can be applied to software as well as hardware designs to protect the control flow and the data-path. DFAs can be prevented through fault detection mechanisms and subsequently avoiding to output faulty results. One general fault detection mechanism which exploits redundancy is to perform an encryption twice and compare the results before outputting them. Another possible countermeasure, which applies to round-based ciphers like the AES algorithm is to reverse the last $n$ rounds and compare the intermediate value to a previously stored intermediate value. This detects faults introduced in the last rounds.

Redundancy through parity bits, cyclic redundancy checks or error detection codes is similar and provides a better error detection at lower overhead than duplication. Control-flow attacks can be detected through general redundancy in the control flow registers.

Dedicated, algorithmic error detection or prevention mostly applies to asymmetric cryptography such as ECC. Special number theoretic properties are used to employ integrity checks (e.g., point verification in ECC). Also, mathematical properties are used to even prevent successful fault attacks upfront, e.g., scalar randomization in ECC. Section 4.2 provides an overview over countermeasures against fault attacks and side-channel analysis of ECC implementations.

# Chapter 3

# Establishing Localized Electromagnetic Analysis

The electromagnetic field as a side-channel of cryptographic devices has been linked to several advantages in the past. These advantages include a higher information content and the alleged possibility to restrict measurements to a certain spatial extent, thus, high-resolution measurements. The impact of such measurements on implementations of cryptographic algorithms is the topic of this thesis.

A precise view on the qualities and limitations of localized measurements has been lacking. In this chapter, I clear up uncertainties regarding high-resolution measuremens by performing a comprehensive study of the electromagnetic near-field side-channel using high-resolution measurement equipment at close distance to a depackaged integrated circuit die. I use a dedicated design-under-test which is especially designed for the purpose of analyzing localized measurements. It consists of a register with a loop feedback through the AES substitution function and is configured into an FPGA. I employ different high-resolution magnetic probes and discuss important parameters of the measurement setup. The measurements are processed in different ways using statistical analysis to extract the information for the assessment.

From this extensive practical study, I present strong evidence for the feasibility of localized measurements of the electromagnetic field. Furthermore, I demonstrate which measurement setups and parameters are best measurements for side-channel analysis. Several important conclusion about the qualities of localized measurements as well as a comparison to current consumption measurements are presented. The results allow conclusions about localized measurements of electromagnetic fields in general and the side-channel analysis of symmetric cryptography implementations in particular.

As mentioned in the introduction in Chap. 1, this localized property allows for dedicated side-channel attacks exploiting location-based information leakage. This will be presented in Chap. 5. Hence, this chapter establishes the ground for the subsequent ideas based on localized measurements. Parts of this chapter have been published on CARDIS conference in 2012 [HMH+12b].

First, I provide important fundamentals from the field of electrical engineering, explaining the most important facts of electromagnetism in Sect. 3.1. Related work from the field of high-resolution measurements is mentioned in Sect. 3.2. The device-under-test, measurement equipment and analysis methods for this study are described in Sect. 3.3. In the main Sect. 3.4, I present and discuss the measurement results and draw conclusions. It is split into several parts which deal with different important properties. The chapter is summarized in Sect. 3.5.

## 3.1 The Electromagnetic Side-Channel

Measurement of magnetic, or electric fields as physical sources of side-channel leakage have been introduced in Sect. 2.2.1. As an important difference to current measurements which are used in conventional power analysis, electric and magnetic fields are vector fields in three-dimensional space. The shape of such fields, which are a superposed composition of different fields in most cases, is very complex. The signal strengths depend on the distance to the source and the orientation of the measurement equipment. For the same reason, more or different information can be extracted from those fields. For example, the field of only a small part of an integrated circuit can be measured instead of the current consumption of the whole circuit. Such *localized* measurements of magnetic fields and their implication on physical cryptanalysis are the topic of this thesis. Localized means that this observation is restricted to a certain location.

This section is intended to establish the abstraction level in understanding of electromagnetic fields which is required for physical cryptanalysis. It is not intended as an alternative to a textbook about electromagnetism.

The electromagnetic field and electromagnetic radiation of an integrated circuit will not be modeled with a high accuracy. Exact modeling would require complete knowledge of a design and a huge effort. Even then, such a precise model would only apply to one design and is neither required for side-channel analysis, nor does it correspond to a valid adversarial model. Instead, key properties will be discussed and a very abstract model of the electromagnetic field of integrated circuits derived.

The following Sect. 3.1.1 presents the fundamentals from the field of elec-

tromagnetism. For more details, see standard literature. This thesis concentrates on electromagnetic fields in the near-field which is described in Sect. 3.1.2. Sect. 3.1.3 presents the abstract model of an integrated circuit which is required for physical cryptanalysis.

### 3.1.1 Electromagnetic Field

All electronic devices require a voltage supply, hence, separated charges to charge and discharge circuit nodes, thus, producing currents from high to low voltage levels, which represent circuit functionality due to the semi-conductor transistors. Maxwell's equations summarize how such charges and current produce surrounding electromagnetic fields and radiation consisting of a magnetic field part **H** and an electric field part **E**. Electromagnetic fields are generated by all conducting circuit parts within an integrated circuit.

In 1862, Maxwell published a paper, *On the Physical Lines of Force*, where he compiled equations which had been previously stated by Gauss, Faraday, and Ampère. Those equation include:

1. *Gauss's law* which describes that electric fields are generated by charges and which states that the surface integral of the electric field gives the enclosed charge.

2. *Gauss's law for magnetism* describes that there are no magnetic charges, thus, that magnetic field lines have no source. Hence, a surface integral over closed loop magnetic field lines is always zero.

3. *Maxwell-Faraday law* which describes how electric fields are generated by time-varying magnetic fields (law of induction).

4. *Ampère's law with Maxwell's correction* describing that magnetic fields are generated through currents and by time-varying electric-fields. The latter is Maxwell's correction and was the foundation of explaining electromagnetic waves.

Maxwell's equations describe two sources for electric as well as magnetic fields. Those are the direct sources, currents and charges, and the respective other field. The term *electromagnetism* describes the fact that electric and magnetic fields can generate and influence the respective other field.

#### Field or Radiation

The most important property when approaching the measurement of magnetic and electric fields is the distinction between near and far fields. This

distinction is based on the distance to the original source of the fields. The far-field is also called radiative field, or electromagnetic radiation.

In the near-field, the electric and magnetic fields are generated by the direct sources which are charges and currents. At greater distances from the direct sources of fields, the electric and magnetic fields are only caused by the respective other field.

Conductors within an integrated circuit carry the direct sources of electromagnetic fields. They can be seen as antennas. The distinction between near- and far-field is derived based on the wavelength $\lambda$ of the emitted electromagnetic field. The emphasis on the wavelength is important to state the relation between source reversal and propagation within a carrier. For antennas shorter than half of the wavelength of the electromagnetic field, an approximation states that the near-field ends at $r \ll \lambda$, where $r$ is the distance to the antenna, and the far-field begins at approximately $r \gg 2\lambda$. In between those distances, both near- and far-field components are significant.

The wavelength of electromagnetic fields is $\lambda = \frac{c}{f}$, where $c$ is the speed of light $\approx 3 \times 10^8\,\mathrm{m/s}$ and $f$ is the frequency of the generating source.

From published research in side-channel analysis, I learned that side-channel leakage has been exploited in frequencies of about 10 MHz to 1 GHz including higher harmonics. The wavelength at these frequencies are:

- At a frequency $f = 10\,\mathrm{MHz}$, the wavelength is $\lambda \approx 30\,\mathrm{m}$.
- At a frequency $f = 100\,\mathrm{MHz}$, the wavelength is $\lambda \approx 3\,\mathrm{m}$.
- At a frequency $f = 1\,\mathrm{GHz}$, the wavelength is $\lambda \approx 0.3\,\mathrm{m}$.

It is clearly observable, that integrated circuits with sizes of $\approx 5 \times 5\,\mathrm{mm}$ will contain antennas which are significantly shorter than half of the wavelength of the emitted signals. Secondly, even at frequencies as high as 1 GHz, the near-field extends very far from the device. Therefore, it becomes obvious the most of the side-channel measurements observe electromagnetic near-fields.

Nonetheless, I will start with explaining the characteristic properties of the electromagnetic far-field in the following section. After this, I describe the properties of the electromagnetic near-field. This helps to understand the differences between the properties of both fields.

### Far-Field

The far-field is also called radiative field, or electromagnetic radiation, because the magnetic and electric fields are propagating, or, radiating disconnected from the source.

The far-field electromagnetic radiation is generated through the near-field electromagnetic field, thus, generated by the sources indirectly. In the far-field, the changing magnetic field generates a changing electric field which, in turn, generates a magnetic field. This is described through Maxwell's equations, specifically through Faraday's law and Maxwell's correction to Ampère's law. Due to this strict dependence, the two fields have a constant phase relation and either of both can be determined through a description of the other.

In the far-field, the integration over any spherical surface around the source of electromagnetic radiation leads to the same amount of energy. This means that the energy in the far-field radiates from the source instead of transferring the energy back to the source which is the case in the near-field.

The near-field components of Maxwell's equations decrease rapidly with distance to the source. After a certain distance, the far-field components prevail which generally have a $\sim \frac{1}{r^2}$ proportionality to the distance $r$. The $\frac{1}{r^2}$ relation is obvious when considering the formula for the surface of a sphere with $A = 4\pi r^2$. Far-field radiation can therefore be measured from greater distances [AARR03].

One often mentioned aspect of electromagnetic radiation in the side-channel context is that a radiating high-frequency signal can carry leakage signals of lower frequencies through modulation. These leakage signals can be recovered through de-modulation of the high-frequency signal [AARR03, LMM05].

However, the electromagnetic far-field is difficult to exploit for side-channel analysis since signal strengths are very low. Targeted embedded devices operate at clock frequencies of about 30 MHz and I roughly expect the biggest electromagnetic field components in a range below 1 GHz. The transition to the far-field therefore occurs at distances $r \gg 2 \times 0.3\,\mathrm{m}$ where the fields are already weak [Fri79].

### Near-Field

In the near-field, the electric field is caused by charges and the magnetic field is caused by moving charges, or, currents. Hence, the two fields depend on different properties of electrons, one being the total charge in a location, the other being the moving of the charges and, hence, both fields do not necessarily exhibit a constant relation to each other.

According to Gauss's law and Ampère's law, the field strength of electric and magnetic fields is proportional to the inverse of the cubic distance $\sim \frac{1}{r^3}$, where $r$ is the distance. In the near-field, those equation components with a $\sim \frac{1}{r^3}$ proportionality are predominant over the far-field components in the

equations describing electric and magnetic fields. This means that the near-fields' strength decreases quickly.

### 3.1.2 Near-Field Side-Channel Analysis

This thesis concentrates exclusively on the electromagnetic near-field, and specifically, localized aspects of it. Localized means that such observations are restricted to a certain location of the emitting device. Localization of measurements is only exploitable in the near-field because the emitting device, an integrated circuit die, appears as a point source in the far field due to its small dimensions. In this section, I discuss aspects of measuring the electromagnetic near-field.

Electric fields can be measured through capacitive coupling. Capacitive coupling means that an electric current is generated in a secondary circuit through a mutual capacitance due to an electric field. The sensor is an electrode which, when in proximity to a circuit, forms a mutual capacitance with the circuit. This generates a current $I_2$ in the secondary circuit which can be measured and described as $I_2 = C\frac{dV_1}{dt}$, with the mutual capacitance $C$, and the change of electric potential, thus, voltage $V_1$ over time $t$ in the first circuit as $\frac{dV_1}{dt}$.

The electric field is commonly regarded as unsuitable for side-channel analysis. The are no published results exploiting measurements of the near electric field. I confirmed this in Sect. 3.3.2. A possible explanation is that the electric field is shielded by other conductors on the integrated circuit for the most part.

Therefore, I concentrate on measurements of the magnetic near-field in this thesis. In the near-field, we can use Ampère's law without Maxwell's correction to describe the magnetic $\mathbf{H}$ field generated by an electric current $I_1$ in a primary conductor. The integral of the $\mathbf{B} = \mu_0\mathbf{H}$ field, with $\mu_0$ the magnetic constant, over a surface gives the magnetic flux $\Phi$.

Efficient sensors for magnetic fields are wire loops with a certain diameter and number of windings, hence, magnetic coils. The magnetic flux $\Phi$ which is enclosed by a conductor loop induces a voltage $V_2$ into this secondary circuit loop according to Faraday's law of induction. This can be expressed as $V_2 = M\frac{dI_1}{dt}$, where the change of electric current $I_1$ in the primary circuit which generated the magnetic field over time $t$ is $\frac{dI_1}{dt}$. The mutual inductance $M$ describes the inductive coupling of both circuits. This includes how much of the magnetic flux $\Phi$ is traversing the magnetic sensor loop coil while regarding the vector dot-product of the magnetic field vectors $\mathbf{H}$ and the vector orthogonal to the coil plane. This also includes the number of windings if there are multiple loop circuits, thus, an electric conductor coil.

Important properties of sensors for electric and magnetic fields are:

- Direction is important because the fields are vectorized in space.
- Spatial resolution relates to the size of the sensor, e.g., diameter of a magnetic coil.
- Sensitivity and bandwidth e.g., relate to the number of windings, impedance and amplification of the measurement equipment.

A magnetic loop sensor with a smaller diameter will capture less traversing magnetic flux, thus, inducing a lower voltage. This requires greater amplification which introduces more noise into the measurement. However, a smaller diameter of the loop conductor probe allows to target specific parts of the magnetic vector field. I use a magnetic field probe with a coil of a very small diameter which is described in Sect. 3.3.2. This is contrary to previous publications discussed in Sect. 3.2, which describe magnetic sensor coils with larger diameters.

A precise model of the electromagnetic field of integrated circuits and knowledge of the goals of side-channels enables building specific sensors for side-channel analysis. However, this was not investigated in this thesis and I refer to Mulder's thesis for information an designing sensors [Mul10]. Rather than designing sensors, I am using commercially available ones and analyze which produce the best results in terms of side-channel analysis. Within this scope is the selection of sensors according to parameters such as, e.g., the diameter of the coil, number of windings, bandwidth, and amplification when dealing with magnetic loop sensors. The selection of sensors according to those parameters can be reasoned according to the abstract model of the electromagnetic field of an integrated circuit presented in the next section.

### 3.1.3 Abstract Model of an Integrated Circuit

As already indicated in the beginning of this section, I regard it to be computationally infeasible to exactly model the electromagnetic field of an integrated circuit of non-trivial complexity. Fortunately, an abstract understanding, or modeling of the electromagnetic field of an integrated circuit is sufficient for side-channel analysis as well as the design of countermeasures.

Figure 3.1 depicts an abstract drawing symbolizing the layered structure of an integrated circuit. The metal-oxide semiconductor field effect transistors are built from doped areas within the substrate on the lowest layer, including poly-silicon gates on the first layer above the substrate. Standard cells for sequential or logical functions are built from those transistors. The

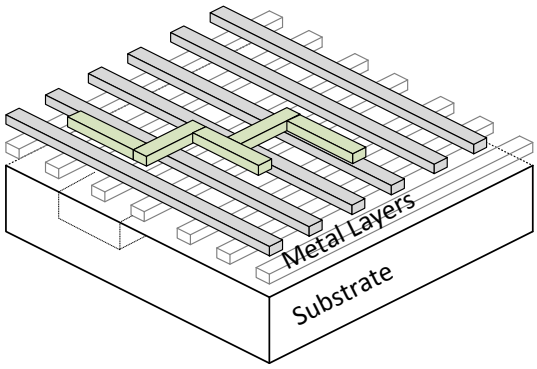



Figure 3.1: Abstract drawing of an integrated circuit with substrate and multiple metal layers

first metal layer usually provides the power supply for the standard cells in rows as well as interconnect on cell level. The upper metal layers are usually used for interconnect and the last metal layer for ground and power supply. The structural sizes within the layers increases from the first to the last one.

All interconnect lines have parasitic capacitances which must be charged and discharged by the output of the cells. This means that a cell which is processing signals will drive those signals onto interconnect lines on different metal layers. Charging and discharging requires short current peaks from the voltage supply or to the ground supply on all metal layers of the circuit.

The logic area of a digital integrated circuit is usually not completely covered by functional cells because the high density cannot be supported by sufficient interconnect in the upper metal layers. The remaining space is filled with filler capacitances. Those capacitances stabilize the voltage supply locally when current is drawn for charging or discharging of parasitic capacitances of interconnect lines. The capacitances as well as parasitic inductances of the signal and power lines act as a low-pass filter for the short current consumption peaks. By comparison to localized measurements of the electric field I demonstrate this low-pass filtering of the current consumption measurements in Sect. 3.4.7.

One important insight is that if a sequential standard cell such as a flip-flop element changes input or output, then this will imply currents in signal lines and supply lines connected to this cell. Those currents will not be limited to the area of the flip-flop standard cell but extend to the surrounding lines.

The drawing in Fig. 3.1 shows metal layers above the substrate in a very abstract way. However, it becomes clear that if charges and currents on different metal layers create electromagnetic fields, they are superposed. At

every point in three-dimensional space, an overlay of electric and magnetic fields from different signal lines from different metal layers can be observed. This means that some of the fields will superpose constructively, and others will cancel each other out.

Usually, design rules restrict the horizontal circuit layout to interconnections in either $x$- or $y$-directions. This supports higher production reliability. Therefore, magnetic fields, which are observed in planes orthogonal to the current direction, are mostly observable in those directions. This means that it makes sense to use a vertical coil in $x$- or $y$-direction as well.

#### Shielding

Electric and magnetic fields within integrated circuits which are generated by one conductor induce voltages and currents into other conductors of the same integrated circuit. This in turn influences the fields and can be interpreted as shielding. For instance electric fields will influence the distribution of electrons in another conductor generating an opposite electric field partly canceling out the original field.

Ferromagnetic conductors within an integrated circuit will guide and confine magnetic fields, or magnetic flux, due to their higher permeability similar to the way that electrical current is flowing through paths of least resistance. However, magnetic flux will also be observed outside of ferromagnetic conductors.

Summarizing, all those facts make it virtually infeasible to approach the measurement of electromagnetic fields for side-channel analysis analytically. Instead I followed a practical approach and analyzed the outcome of using electromagnetic sensors in different configurations as described in Sect. 3.3.2. The results are presented in Sect. 3.4.

## 3.2 Related Work

Precise measurements of the electromagnetic field have been mentioned in past contributions, concentrating mostly on the magnetic near-field, and cartography thereof [QS01] to find locations where side-channel analyses lead to the best results [HdlTR12, SGM09, RVD09].

Agrawal et al. [AARR03] as well as Standaert and Archambeau [SA08] provide evidence for the fact, that magnetic fields are vectored and that different coil directions lead to different information gain in the context of side-channel analysis. Gandolfi et al. [GMO01] state that inductive probes with high spatial resolutions can be used to locally restrict measurements to

specific circuit parts if they are placed close to the surface of an integrated circuit.

A variety of magnetic probes have been used in past contributions. Large, hand-crafted ones are used by Mulder et al. [DMBO+05] for global measurements of a chip. This is mostly dominated by the magnetic field of supply wires. Mulder describes parameters of probe design in her thesis [Mul10].

Peeters et al. [PSQ07] use a custom designed probe with a coil diameter of 0.7 mm at a fixed position and close distance to an integrated circuit after partly removal of the package. However, the probe positions are fixed and they do not investigate local aspects further.

Sauvage et al. [SGM09] use laboratory equipment with a coil diameter of 0.5 mm outside the chip's package. They employ an FPGA as well as an ASIC as device-under-test. As a conclusion, they state that observed areas of signal leakage do not coincide with the placement of the leaking design parts on the floorplan of the FPGA. I provide evidence for the contrary and suggest that their measurement equipment and distance to the die surface were insufficient to observe localized electromagnetic fields. Instead they mainly observe the magnetic field of supply bonding wires.

Kirschbaum and Schmidt [KS11] present first evidence for successfully localizing EM leakage and performed cartographic measurements. However, they use a hand-crafted coil with 0.5 mm diameter, which has a comparably coarse resolution. This lead to coarse localizations and unconvincing results. Contrarily, I provide results which provide clear and precise evidence for localization of electromagnetic fields.

## 3.3 Practically Clarifying Localized EM

This section is the first main section and describes the device-under-test, measurement equipment and analysis methods. The next Sect, 3.4 presents the results of the practical experiment.

### 3.3.1 Device-Under-Test

A *Xilinx Spartan 3A XC3S200A* FPGA in a *VQ100* package, is used as device-under-test. The device is manufactured in a 90 nm technology, uses a 1.2 V supply for the internal logic, and the die measures $4100 \times 4300\,\mu m$. To perform semi-invasive measurements close to the surface of the chip, the FPGA is depackaged from the front-, and backside using fuming nitric acid, i.e., with a concentration of $> 95\%$. Figure 3.2 presents a photograph of the die from the frontside. The FPGA is mounted onto a custom PCB which

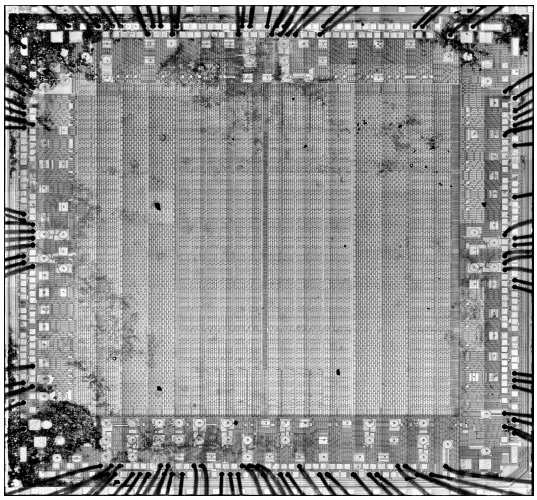


Figure 3.2: Frontside photograph of the *Xilinx Spartan 3A*

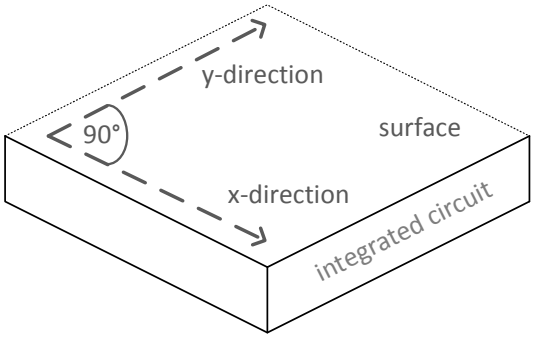
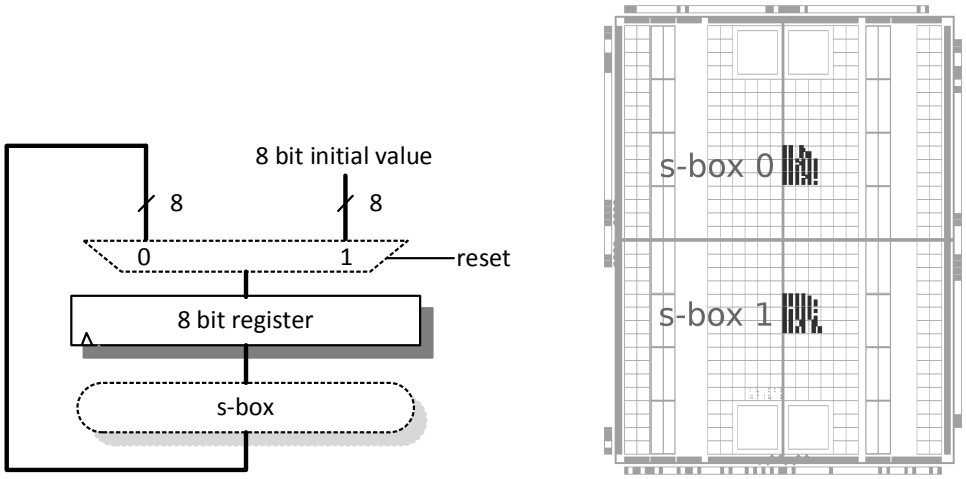



Figure 3.3: Directions $x$ and $y$ for the measurements on the frontside and backside surface of the integrated circuit

supports accessibility for the measurement probes from both sides. Figure 3.3 depicts the x- and y-directions for the movement of measurement probes on the frontside or backside surface of the integrated circuit.

The FPGA is configured with a hardware design-under-test. I derived a simple design to support an easy acquisition of measurements, while being able to draw meaningful conclusions about the side-channel leakage of cryptographic designs. This design is depicted in Fig. 3.4(a) and contains a feedback loop structure including an 8-bit register and an implementation of the AES substitution function, s-box, as published by Canright [Can05]. The 8-bit register is loaded with a fixed initial value at synchronous reset and updates the register with the value's s-box-substitution in every cycle. Therefore, every clock cycle contains a value update, i.e. Hamming distance. The




(a) S-box-structure (b) Placement

Figure 3.4: Design-under-test

design always performs the same operation. Hence, there are no operation-dependencies and the design serves to analyze data-dependent side-channel leakage. This is according to implementations of symmetric cryptographic algorithms which are primarily subject to differential side-channel attacks relying exclusively on data-dependent leakage. According to my opinion, this structure exhibits similar side-channel behavior as implementations of the AES algorithm, because the same non-linear function is used and the amount of combinational logic is representative for such implementations.

To analyze localized, thus, locally restricted aspects of the electromagnetic side-channel leakage, two instantiations of this register-s-box structure are used. I used constraints to place the s-box structures 0 and 1 on the FPGA at a certain distance and to restrict both structures to the same area. The placement on the floorplan of the *Xilinx Spartan 3A* FPGA is depicted in Fig. 3.4(b).

Since both structures are active at the same time, they consume power at the same time and contribute to the electromagnetic field jointly. To analyze the contributions of the two structures separately, they need to be statistically independent. The two instantiations use different initial values for their feedback loop. If values from a limited space are repeatedly replaced by a substitution function projecting into the same space, the initial values are eventually derived since the number space is limited. The number of substitutions, thus, length of the sequence of values depends on the number space, substitution function and the generating initial value. I achieved an

independence, or de-correlation of both structures by using sequences with different initial values and different lengths for both structures.

Hence, the offset between the two sequences is different for every repetition of either one. Alternatively, this could also be achieved by using random numbers for subsequent values of the register updates for both structures. However, this approach simplifies analysis since the value sequences can be reproduced externally and no random values must be stored separately.

*The s-box structure with index* 0 *has an initial value of* `0x1d`*, resulting in a sequence length of* 87*. The s-box structure with index* 1 *has an initial value of* `0x09`*, resulting in a sequence length of* 81*.* Obviously, neither of the two sequences contains values from the respective other sequence.

The design additionally includes a 16-bit counter to generate an external trigger for the oscilloscope and synchronously reset both structures. Every measurement contains $2^{16}$ consecutive clock cycles, thus, 753 repetitions of the sequence with length 87 and 809 repetitions of the sequence with length 81.

### 3.3.2 Measurement Setup

Electromagnetic fields and the measurement principle are explained in Sect. 3.1. It is not obvious which probe, or which magnetic coil direction leads to the best results for side-channel analysis. Therefore, I performed measurements with different coil directions. The following magnetic probes were used to measure the magnetic near-field:

1. *Langer ICR HH 150-6* Magnetic field probe with 150 µm shielded horizontal coil, 6 windings, 100 µm resolution, and 2.5 MHz − 6 GHz frequency span.

2. *Langer ICR HV 150-6* Magnetic field probe with 150 µm shielded vertical coil, 6 windings, 80 µm resolution, and 2.5 MHz − 6 GHz frequency span.

The horizontal coil probe contains a magnetic loop in the horizontal plane and measures the vertical components of the superposed magnetic H-field generated by the circuit. This is depicted in Fig. 3.5. There is no further meaningful degree of freedom in adjusting the horizontal probe.

The vertical probe measures horizontal magnetic H-field components and provides a further choice of direction of the probe. I limited this analysis to two directions, the $x$- and $y$-directions. Figure 3.6 depicts a magnetic coil in the vertical plane in x-direction which captures H-field components in

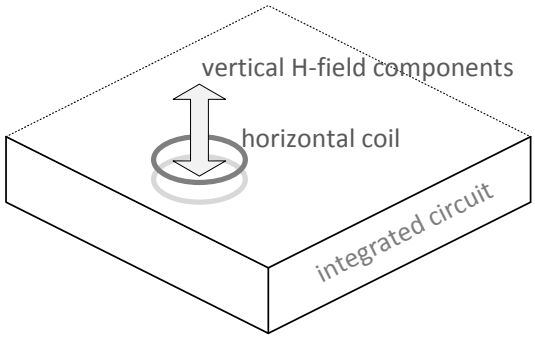



Figure 3.5: Magnetic coil in horizontal plane capturing vertical H-field components

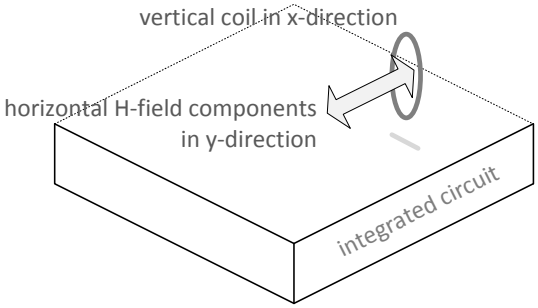



Figure 3.6: Magnetic coil in vertical plane and x-direction capturing horizontal H-field components in y-direction

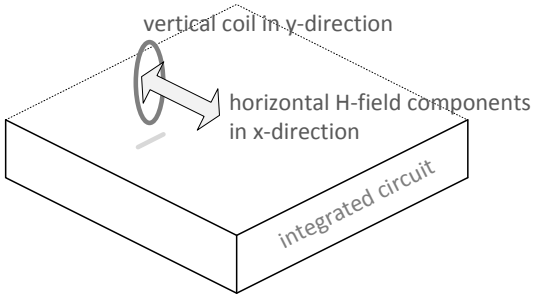
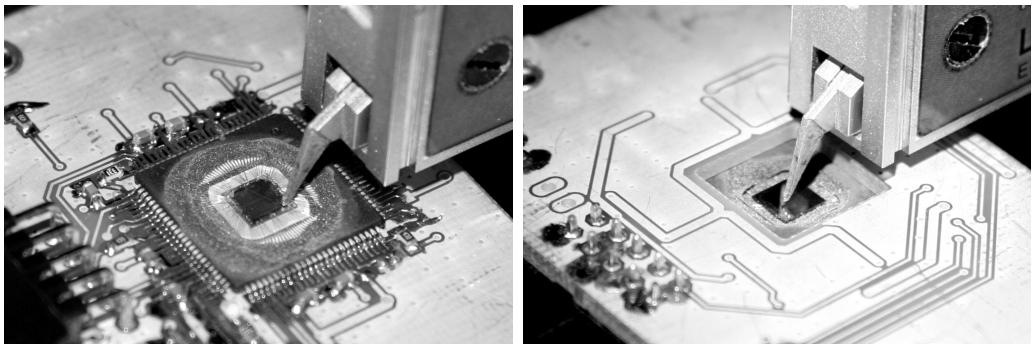



Figure 3.7: Magnetic coil in vertical plane and y-direction capturing horizontal H-field components in x-direction

the horizontal y-direction. Figure 3.7 depicts a magnetic coil in the vertical plane in y-direction which captures H-field components in the horizontal x-direction. The restriction to two directions is reasonable since conductors in integrated circuits are limited in these directions due to manufacturing stability reasons. This can be observed in the abstract illustration in Fig. 3.1.

I also evaluated a high-resolution electric field probe *Langer ICR E 150. However, the measurements did not reveal any detectable signals, thus, I conclude that this probe is unsuitable for side-channel analysis.* The reason might be that the electric field is shielded by the conductors within the circuit. Another difference to the magnetic field probe is, that the coil in the magnetic probe has multiple windings. Therefore, it might be able to captures the field better.



(a) Frontside (b) Backside

Figure 3.8: Probe positioned and moved over the surface of the FPGA die

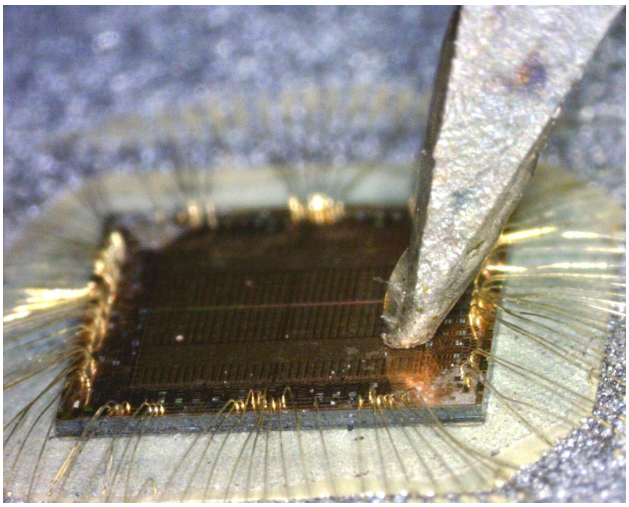


Figure 3.9: Magnetic coil probe above die surface

Electrical conductors within integrated circuit influence magnetic fields. The magnetic fields induce currents in those conductors which generate opposing magnetic fields reducing the strength of the original fields. However,

my results indicate that the fields are still detectable with high signal-to-noise-ratios.

The probes are moved over the front- and backside surface of the FPGA die using a stepping table at a resolution of 100 µm and one measurement is recorded at every position. The backside measurement is depicted in Fig. 3.8(b). The probe touches the surface of the circuit and $43 \times 41$ measurements are recorded. The frontside measurement is depicted in Fig. 3.8(a). To prevent damaging probe and die, the probe to surface distance is $\approx 50$ µm. The bonding wires prevent measurements over the complete surface and $27 \times 27$ measurements are recorded in the area enclosed by the bonding wires.

Figure 3.9 depicts a close-up of the sensor probe above the FPGA die surface. The magnetic measurement coil is integrated into the tip of the probe and invisible in the photograph.

All probes contain a built-in 30 dB amplifier with a noise figure of 4.5 dB. Additionally, a *Langer PA 303* 30 dB amplifier with a bandwidth of 3 GHz and noise figure of 4.5 dB is used.

The *LeCroy WavePro 715Zi* oscilloscope has an analog input bandwidth of 1.5 GHz at 50 Ω impedance. As an approximate upper boundary for the sampling rate, twice the bandwidth of the equipment, thus, 5 GS/s seems reasonable. All measurements have zero offset and a vertical resolution of 50 mV/DIV, so that all measurements stay within scale.

The noise contribution from the measurement setup, i.e., the probes, the two amplifiers, and the oscilloscope was determined by turning of the clock and voltage supply and recording a trace containing noise exclusively. This resulted in noise with a standard deviation of $\approx 22.3$ µV for the horizontal magnetic probe, and noise with a standard deviation of $\approx 20$ µV for the vertical one.

A 20 MHz clock signal is used for the design on the FPGA. Through synchronization of the oscilloscope and the function generator, frequency jitter and drift in the measurements is prevented. Every measurement contains 16384000 byte samples for the $2^{16}$ recorded clock cycles.

I took a current consumption measurement to compare its quality for side-channel analysis to high-resolution EM measurements. A *LeCroy* active differential probe with a bandwidth of 500 MHz was used to measure the voltage drop over a 10 Ω measurement resistor which was inserted into the supply wire of the FPGA's core voltage.

### 3.3.3 Analyses

In every measurement, the two different s-box structures contribute a repeating sequence of value updates of different length. In the following, I describe how the signal parts generated by the two structures are extracted separately from the measurements for the analysis of their signal strength.

The number of sample values in each measurements trace recorded with the oscilloscope in our setup $\mathbf{t}$ is $T_{CLK} * N_{CYCLES}$. $T_{CLK}$ is the number of samples during one cycle of the devices 20 MHz clock and equals 250. $N_{CYCLES}$ is the number of cycles in one measurement trace and equals $2^{16}$. Each measurement trace is denoted as $\mathbf{t} = (t_1, \ldots, t_{(T_{CLK} * N_{CYCLES})})$.

As a first step, the sample mean and standard deviation of all clock cycles is inspected. To achieve this, we cut $\mathbf{t}$ into sub-traces $\mathbf{t}_i^{cycles}$ of length $T_{CLK}$ and compute the mean $\overline{\mathbf{m}}^{cycles}$ and sample standard deviation $\mathbf{s}^{cycles}$ vectors of this set of sub-traces. This is described in Eq. 3.1, 3.2, and 3.3. The sample mean trace $\overline{\mathbf{m}}^{cycles}$ contains the part of the measured signal, which is equal for all clock cycles, hence the data-independent part of the signal. The sample standard deviation $\mathbf{s}^{cycles}$ depicts the part of the signal which varies in every cycle, thus, the data-dependent part and noise components.

$$\mathbf{t}_i^{cycles} = (t_{(1+(i-1)*T_{CLK})}, \ldots, t_{(i*T_{CLK})}),\ 1 \leq i \leq N_{CYCLES} \tag{3.1}$$

$$\overline{\mathbf{m}}^{cycles} = \frac{1}{N_{CYCLES}} \sum_{i=1}^{N_{CYCLES}} \mathbf{t}_i^{cycles} \tag{3.2}$$

$$\mathbf{s}^{cycles} = \sqrt{\frac{1}{N_{CYCLES} - 1} \sum_{i=1}^{N_{CYCLES}} (\mathbf{t}_i^{cycles} - \overline{\mathbf{m}}^{cycles})^2} \tag{3.3}$$

As a next step, the trace is processed to extract the two independent signal components. In this way, the original trace $\mathbf{t}$ is split into sub-traces in two ways according to the different sequence lengths. First, to determine the signal component of s-box structure 0, it is split into $\lfloor \frac{N_{CYCLES}}{87} \rfloor = 753$ sequences $\mathbf{t}_{0,i}$ which contain 87 clock cycles, thus, $87 * T_{CLK}$ samples each. This is described in Eq. 3.4.

$$\mathbf{t}_{0,i} = (t_{(1+(i-1)*87*T_{CLK})}, \ldots, t_{(i*87*T_{CLK})}),\ 1 \leq i \leq \lfloor \frac{N_{CYCLES}}{87} \rfloor \tag{3.4}$$

Second, to determine the signal component of s-box structure 1, it is split into $\lfloor \frac{N_{CYCLES}}{81} \rfloor = 809$ sequences $\mathbf{t}_{1,i}$ which contain 81 clock cycles, thus, $81 * T_{CLK}$ samples each. This is described in Eq. 3.5.

$$\mathbf{t}_{1,j} = (t_{(1+(j-1)*81*T_{CLK})}, \ldots, t_{(j*81*T_{CLK})}),\ 1 \leq j \leq \lfloor \frac{N_{CYCLES}}{81} \rfloor \tag{3.5}$$

As a next step, the sample mean vectors of the set of sequences $\{\mathbf{t}_{0,i} | 1 \leq i \leq \lfloor \frac{N_{CYCLES}}{87} \rfloor\}$, and $\{\mathbf{t}_{1,j} | 1 \leq j \leq \lfloor \frac{N_{CYCLES}}{81} \rfloor\}$ are computed. This is described in Eq. 3.6 and 3.7.

$$\overline{\mathbf{m}}_0 = \frac{1}{753} \sum_{i=1}^{753} \mathbf{t}_{0,i} \tag{3.6}$$

$$\overline{\mathbf{m}}_1 = \frac{1}{809} \sum_{j=1}^{809} \mathbf{t}_{1,j} \tag{3.7}$$

The figures in the next Sect. 3.4 will also depict the sample standard deviation of $\{\mathbf{t}_{0,i} | 1 \leq i \leq \lfloor \frac{N_{CYCLES}}{87} \rfloor\}$, and $\{\mathbf{t}_{1,j} | 1 \leq j \leq \lfloor \frac{N_{CYCLES}}{81} \rfloor\}$. They are computed as described in Eq. 3.8 and 3.9.

$$\mathbf{s}_0 = \sqrt{\frac{1}{753-1} \sum_{i=1}^{753} (\mathbf{t}_{0,i} - \overline{\mathbf{m}}_0)^2} \tag{3.8}$$

$$\mathbf{s}_1 = \sqrt{\frac{1}{809-1} \sum_{j=1}^{809} (\mathbf{t}_{1,j} - \overline{\mathbf{m}}_1)^2} \tag{3.9}$$

This removes the noise in respect to the two different signals at a statistical sample size of 753, and 809 respectively. Therefore, what remains as $\overline{\mathbf{m}}_0$ and $\overline{\mathbf{m}}_1$ are the effective side-channel signals from the two structures. Part of these signals is due to clocking the registers, i.e., the clock tree and common logic parts, and is visible in every cycle, thus, data-independent. This data-independent part can be derived by cutting $\overline{\mathbf{m}}_0$ and $\overline{\mathbf{m}}_1$ into sub-traces $\mathbf{m}_{0,i}^{cycles}$ and $\mathbf{m}_{1,i}^{cycles}$ of the length of a clock cycle $T_{CLK}$ each and computing the sample mean traces $\overline{\mathbf{m}}_{0,cycles}$ and $\overline{\mathbf{m}}_{1,cycles}$ of the two sets again. This is described in Eq. 3.10, 3.11, 3.12, and 3.13.

$$\mathbf{m}_{0,i}^{cycles} = (\overline{m}_{0,(1+(i-1)*T_{CLK})}, \ldots, \overline{m}_{0,(i*T_{CLK})}),\ 1 \leq i \leq 87 \tag{3.10}$$

$$\mathbf{m}_{1,j}^{cycles} = (\overline{m}_{1,(1+(j-1)*T_{CLK})}, \ldots, \overline{m}_{1,(j*T_{CLK})}),\ 1 \leq j \leq 81 \tag{3.11}$$

$$\overline{\mathbf{m}}_0^{cycles} = \frac{1}{87} \sum_{i=1}^{87} \mathbf{m}_{0,i}^{cycles} \tag{3.12}$$

$$\overline{\mathbf{m}}_1^{cycles} = \frac{1}{81} \sum_{j=1}^{81} \mathbf{m}_{1,j}^{cycles} \tag{3.13}$$

The variance between the clock cycles in the two derived signal sequences $\overline{\mathbf{m}}_0$ and $\overline{\mathbf{m}}_1$ is most interesting because it is due to the processed data, hence, the data-dependent part of the signal. This is the target of this side-channel study. To extract this data-dependent part, the sample means $\overline{\mathbf{m}}_0^{cycles}$ and $\overline{\mathbf{m}}_1^{cycles}$ of the clock cycles are subtracted from the cut-out sub-traces $\mathbf{m}_{0,i}^{cycles}$ and $\mathbf{m}_{1,j}^{cycles}$. The data-dependent signals $\mathbf{t}_{0,i}^{data}$ and $\mathbf{t}_{1,j}^{data}$ remain without noise or data-independent signal parts. This is described in Eq. 3.14 and 3.15.

$$\mathbf{t}_{0,i}^{data} = \mathbf{m}_{0,i}^{cycles} - \overline{\mathbf{m}}_0^{cycles},\ 1 \leq i \leq 87 \tag{3.14}$$

$$\mathbf{t}_{1,j}^{data} = \mathbf{m}_{1,j}^{cycles} - \overline{\mathbf{m}}_1^{cycles},\ 1 \leq j \leq 81 \tag{3.15}$$

The data-dependent distribution of measured signals can be assumed Gaussian. To assess the quality of the signal and to compute a signal-to-noise ratio in the following, we compute the sample standard deviation of the set of vectors $\{\mathbf{t}_{0,i}^{data} | 1 \leq i \leq 87\}$ for the signal from structure 0 and the sample standard deviation of the set $\{\mathbf{t}_{1,j}^{data} | 1 \leq j \leq 81\}$ for the signal from structure 1. This is described in Eq. 3.16, 3.17, 3.18, and 3.19.

$$\overline{\mathbf{m}}_0^{data} = \frac{1}{87} \sum_{i=1}^{87} \mathbf{t}_{0,i}^{data} \tag{3.16}$$

$$\overline{\mathbf{m}}_1^{data} = \frac{1}{81} \sum_{j=1}^{81} \mathbf{t}_{1,j}^{data} \tag{3.17}$$

$$\mathbf{s}_0^{data} = \sqrt{\frac{1}{87-1} \sum_{i=1}^{87} (\mathbf{t}_{0,i}^{data} - \overline{\mathbf{m}}_0^{data})^2} \tag{3.18}$$

$$\mathbf{s}_1^{data} = \sqrt{\frac{1}{81-1} \sum_{j=1}^{81} (\mathbf{t}_{1,j}^{data} - \overline{\mathbf{m}}_1^{data})^2} \tag{3.19}$$

To be able to compute signal-to-noise ratio vectors we need to know the noise standard deviation. If we subtract one of the two averaged signal sequences $\overline{\mathbf{m}}_0$ from the trace $\mathbf{t}$ in its cut representation $\{\mathbf{t}_{0,i} |\ 1 \leq i \leq$

$\lfloor \frac{N_{CYCLES}}{87} \rfloor\}$, the noise in respect to this sequence 0 remains as $\{\mathbf{t}_{0,i}^{noise} |\ 1 \leq i \leq \lfloor \frac{N_{CYCLES}}{87} \rfloor\}$. This is described in Eq. 3.20 and3.21.

$$\mathbf{t}_{0,i}^{noise} = \mathbf{t}_{0,i} - \overline{\mathbf{m}}_0,\ 1 \leq i \leq \lfloor \frac{N_{CYCLES}}{87} \rfloor \tag{3.20}$$

$$\mathbf{t}_{1,j}^{noise} = \mathbf{t}_{1,j} - \overline{\mathbf{m}}_1,\ 1 \leq j \leq \lfloor \frac{N_{CYCLES}}{81} \rfloor \tag{3.21}$$

The noise is distributed Gaussian with a zero mean and includes power supply noise, clock supply noise, measurement noise, quantization error and switching, or algorithmic noise from parts of the circuit which did not contribute to the signal. The switching noise includes signals from the counter as well as from the respective other s-box structure, which exhibits an uncorrelated sequence with different length.

In order to derive a noise standard deviation trace over the duration of a clock cycle, the two sets of noise sub-traces $\{\mathbf{t}_{0,i}^{noise} |\ 1 \leq i \leq \lfloor \frac{N_{CYCLES}}{87} \rfloor\}$ and $\{\mathbf{t}_{1,i}^{noise} |\ 1 \leq i \leq \lfloor \frac{N_{CYCLES}}{81} \rfloor\}$ are first transformed into two noise traces $\mathbf{t}_0^{noise}$ and $\mathbf{t}_1^{noise}$ through concatenation. Secondly, they are split into sub-traces $\mathbf{t}_{0,i}^{noise,cycles}$ and $\mathbf{t}_{1,i}^{noise,cycles}$ of length $T_{CLK}$ just like $\mathbf{s}_0^{data}$ and $\mathbf{s}_1^{data}$ . This is described in Eq. 3.22, 3.23, 3.24, and 3.25.

$$\mathbf{t}_0^{noise} = (\mathbf{t}_{0,1}^{noise}, \ldots, \mathbf{t}_{0,N_{CYCLES}}^{noise}) \tag{3.22}$$

$$\mathbf{t}_1^{noise} = (\mathbf{t}_{1,1}^{noise}, \ldots, \mathbf{t}_{1,N_{CYCLES}}^{noise}) \tag{3.23}$$

$$\mathbf{t}_{0,i}^{noise,cycles} = (t_{0,(1+(i-1)*T_{CLK})}^{noise}, \ldots, t_{0,(i*T_{CLK})}^{noise}),\ 1 \leq i \leq N_{CYCLES} \tag{3.24}$$

$$\mathbf{t}_{1,j}^{noise,cycles} = (t_{1,(1+(j-1)*T_{CLK})}^{noise}, \ldots, t_{1,(j*T_{CLK})}^{noise}),\ 1 \leq j \leq N_{CYCLES} \tag{3.25}$$

The sample standard deviation within the two sets of noise traces $\{\mathbf{t}_{0,i}^{noise,cycles} | 1 \leq i \leq \lfloor N_{CYCLES} \}$ and $\{\mathbf{t}_{1,j}^{noise,cycles} | 1 \leq j \leq \lfloor N_{CYCLES} \}$ is computed as described in Eq. 3.26, 3.27, 3.28, and 3.29. These standard deviations $s_0^{noise}$ and $s_1^{noise}$ are traces of length $T_{CLK}$ themselves.

$$\overline{\mathbf{m}}_0^{noise} = \frac{1}{N_{CYCLES}} \sum_{i=1}^{N_{CYCLES}} \mathbf{t}_{0,i}^{noise,cycles} \tag{3.26}$$

$$\overline{\mathbf{m}}_1^{noise} = \frac{1}{N_{CYCLES}} \sum_{j=1}^{N_{CYCLES}} \mathbf{t}_{1,j}^{noise,cycles} \tag{3.27}$$

$$\mathbf{s}_0^{noise} = \sqrt{\frac{1}{N_{CYCLES}-1} \sum_{i=1}^{N_{CYCLES}} (\mathbf{t}_{0,i}^{noise,cycles} - \overline{\mathbf{m}}_0^{noise})^2} \tag{3.28}$$

$$\mathbf{s}_1^{noise} = \sqrt{\frac{1}{N_{CYCLES}-1} \sum_{j=1}^{N_{CYCLES}} (\mathbf{t}_{1,j}^{noise,cycles} - \overline{\mathbf{m}}_1^{noise})^2} \tag{3.29}$$

Finally, the Signal-to-Noise Ratio (SNR) for both independent structures, hence, signals is computed as the quotient between the signal and noise standard deviation traces as described in Eq. 3.30 and 3.31. The SNR is derived for both different signal sources and it depends on the location, which one results in a higher measured SNR. Results presented in the next Sect. 3.4 will prove this.

$$\mathbf{SNR}_0 = 20 * \log(\frac{\mathbf{s}_0^{data}}{\mathbf{s}_0^{noise}})\,\mathrm{dB} \tag{3.30}$$

$$\mathbf{SNR}_1 = 20 * \log(\frac{\mathbf{s}_1^{data}}{\mathbf{s}_1^{noise}})\,\mathrm{dB} \tag{3.31}$$

All up to now derived variables are vectors containing $T_{CLK}$ values each. For the map figures in the next section, I display one value for each position, thus, measurement. This is done by using the maximum value from each vector, e.g., $\max \mathbf{SNR}_0$.

I performed a Correlation Power Analysis (CPA) to evaluate the quality of the measurements for differential power analysis. The Hamming distance leakage model between values from consecutive cycles is employed. The sample size $n$ for the correlation equals the number of recorded clock cycles, $2^{16}$, and a correlation coefficient of 0 results in values of $\pm\frac{4}{\sqrt{n}} = \pm\frac{1}{2^6} = \pm 0.015625$ with a confidence level of 99.99% [MOP07]. Therefore, absolute correlation coefficients below this significance level are disregarded.

## 3.4 Discussion of Measurement Results

In this section, I present measurement results and derive conclusions from their analysis. The analysis methods had been described in Sect. 3.3.3.

The measurements from the frontside with the horizontal coil led to the best results. Therefore, I use those measurements to discuss most aspects and present other setups in comparison to them.

### 3.4.1 Signal and Noise

In this first part, figures for signal, noise and SNR are presented. *For an illustrative example, a measurement at position* $(x, y) = (24, 17)$ *is used and will be called* $P_1$ *subsequently.* This position proved to be approximately above s-box structure 1 during further analysis presented in the subsequent sections.

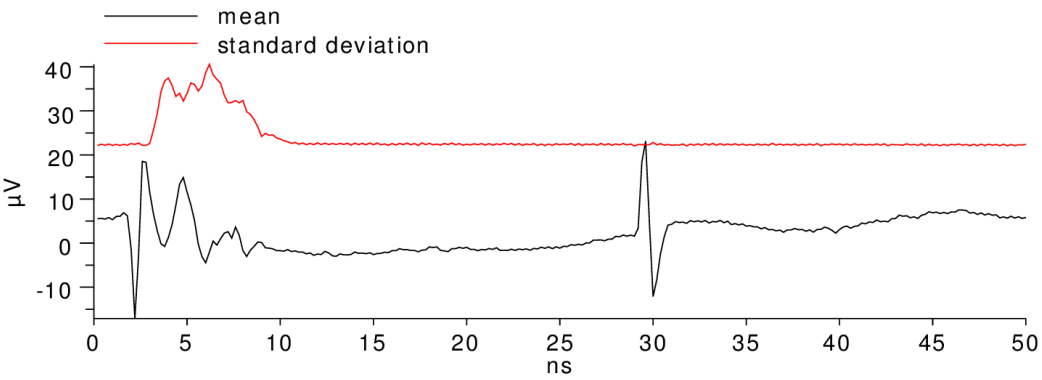



Figure 3.10: Mean $\overline{\mathbf{m}}^{cycles}$ and standard deviation $\mathbf{s}^{cycles}$ of all clock cycles at $P_1$ frontside, horizontal coil)

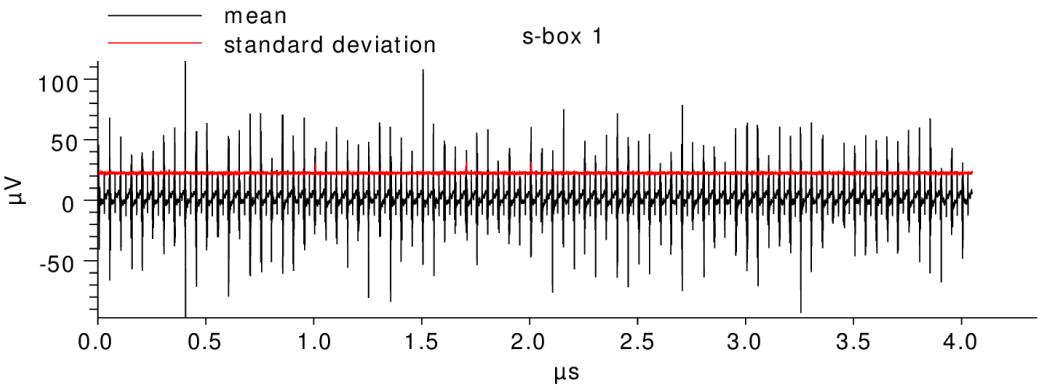



Figure 3.11: Mean $\overline{\mathbf{m}}_1$ and standard deviation $\mathbf{s}_1$ of repeated s-box 1 sequence at $P_1$ (frontside, horizontal coil)

Figure 3.10 depicts the mean $\overline{\mathbf{m}}^{cycles}$ and standard deviation $\mathbf{s}^{cycles}$ of all clock cycles. The computation has been explained in the previous Sect. 3.3.3. The figure spans the time of one clock cycle $T_{CYCLE}$, thus, 50 ns. As discussed in the previous section, the mean represents the data-independent part of the measured signal which is for instance due to clocking the registers. In this mean trace, the active and inactive clock edges can be observed as significant

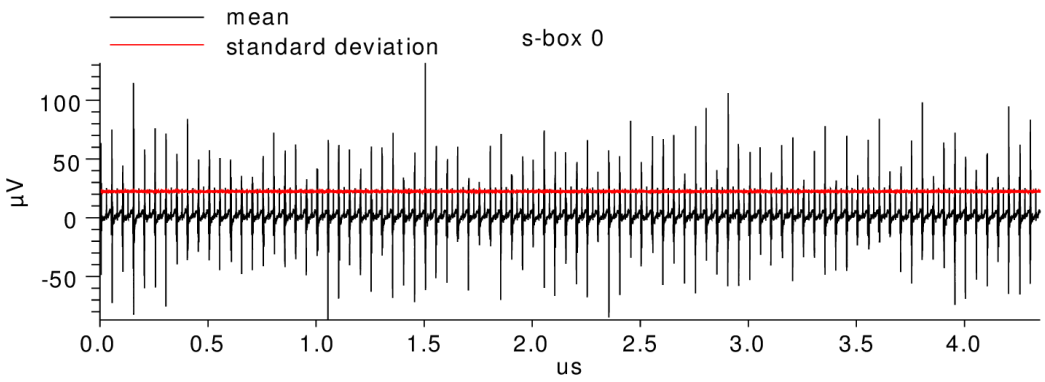



Figure 3.12: Mean $\overline{\mathbf{m}}_0$ and standard deviation $\mathbf{s}_0$ of repeated s-box 0 sequence at $P_0$ (frontside, horizontal coil)

peaks at the times of $\approx 2\,\text{ns}$ and $\approx 30\,\text{ns}$. The active clock edge is the one at $\approx 2\,\text{ns}$ and we can observe significant circuit activity after the first peak. This circuit activity must be data-independent because it is present in all clock cycles.

The standard deviation is constant throughout most of the cycle which can be explained by constant noise factors from the measurement setup and corresponds well to the measured noise floor mentioned in Sect. 3.3.2. The standard deviation is significantly higher during a short time after the active clock edge. This is clearly due to the data-dependent switching activity in the circuit. At this stage, this switching activity cannot be attributed to specific parts of the design, e.g., one or the other s-box structure, and contains data-dependent contributions from all circuit parts.

It can be noted, that the time during which data-dependent signals can be observed is limited to a short duration after the active edge. From this, it can be concluded, that a localized measurement reveals the length of combinational paths which cause the measured signals.

The device seems to be clocked at a frequency significantly lower speed than the critical one, since the data-dependent currents are stabilized long before the next active edge.

### Signal

I determined the data-dependent signal and noise for both structures in every measurement as described in Sect. 3.3.3. Figure 3.11 shows the sample mean $\overline{\mathbf{m}}_1$ and sample standard deviation $\mathbf{s}_1$ of the repeated sequence of values processed by s-box structure 1 which contains 81 clock cycles. Hence, the diagram spans $81 * 50\,\text{ns}$. The constant floor of the standard deviation has

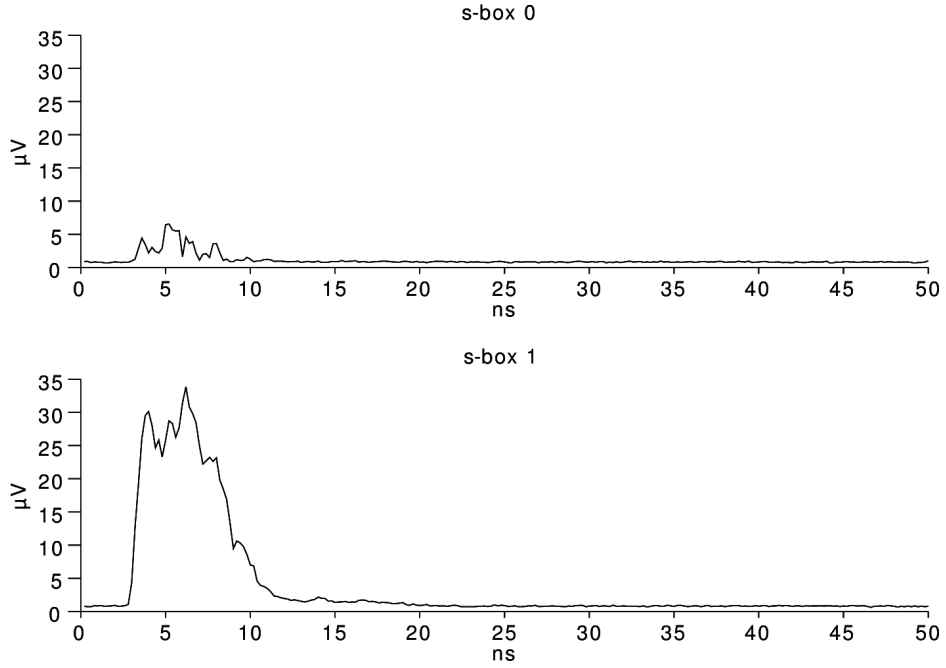



Figure 3.13: Data-dependent signal standard deviations $\mathbf{s}_0^{data}$ and $\mathbf{s}_1^{data}$ over clock cycle for s-box 0 and 1 at $P_1$ (frontside, horizontal coil)

a similar value of $\approx 22\,\mu V$ as the one depicted in Fig. 3.10. However, there are no occurrences of higher standard deviations, as could be observed in Fig. 3.10. This is due to the fact, that the data-dependent signal parts are now incorporated in the mean trace $\overline{\mathbf{m}}_1$. Another proof for this is that the mean exhibits significantly higher peak amplitudes which can be observed in Fig. 3.11. These peaks in the mean trace, and the variance in their values, are the data-dependent signal components.

One might notice that the amplitudes of the data-dependent peaks in Fig. 3.11 exhibit significantly higher values than the peaks that can be observed in the mean depicted in Fig. 3.10. The mean peaks in Fig. 3.10 only contained signals generated by clocking activity and parts of the circuit which are active in every clock cycle, e.g., counters. Contrarily, the mean peaks in Fig. 3.11 additionally contain data-dependent signal parts. The standard deviation in this data-dependent signals of s-box 1, $\mathbf{s}_1^{data}$, is depicted in the lower graphic in Fig. 3.13. It corresponds to the average additional data-dependent amplitude. This results in the fact, that the additional data-dependent amplitude is high in certain cases. And this is exactly what can be observed in Fig. 3.11, where some data-dependent peak amplitudes are quite large compared to the ones in Fig. 3.10.

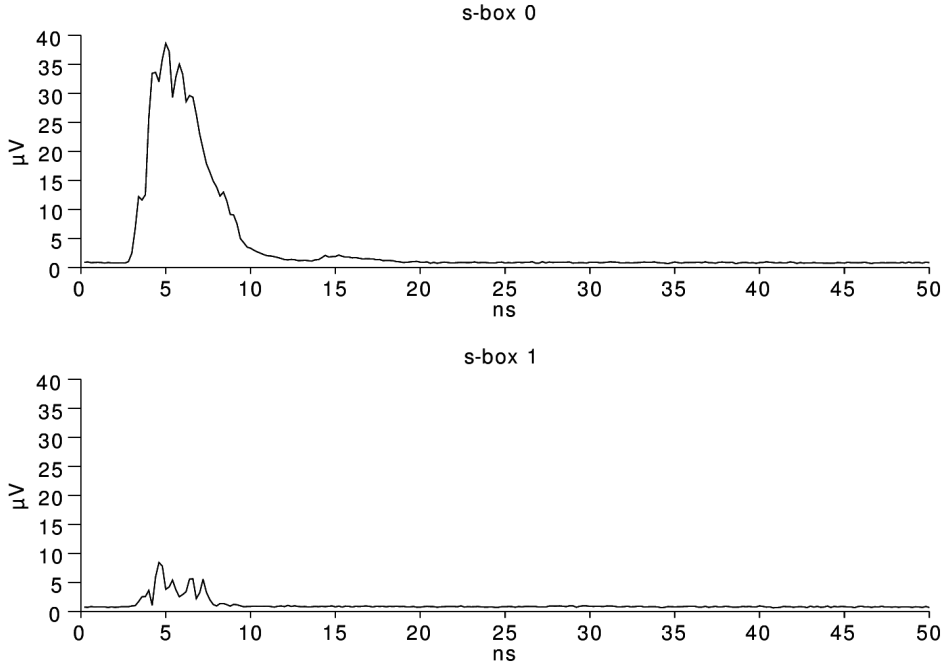



Figure 3.14: Data-dependent signal standard deviations $\mathbf{s}_0^{data}$ and $\mathbf{s}_1^{data}$ over clock cycle for s-box 0 and 1 at $P_0$ (frontside, horizontal coil)

I determined the data-dependent part of the signals as described in Sect. 3.3.3. Figure 3.13 depicts the results, which are the data-dependent signals $\mathbf{s}_0^{data}$ and $\mathbf{s}_1^{data}$ of the two s-box structures over a clock cycle. Hence, the diagram spans 50 ns.

The bottom diagram shows $\mathbf{s}_1^{data}$. The maximum standard deviation amplitude is clearly significant when compared to the noise level mentioned before. The constant floor of $\approx 1\,\mu V$ seems to be due to statistical artifacts.

Significant signal amplitudes are observed within the first 10 ns after the positive, active clock edge. The synthesis tool reported 12.5 ns delay as the longest combinational path of this design. *It is an important observation, that signal leakage is exclusively restricted to a time-span as short as the combinational path after the active clock edge when analyzing local EM measurements close to the source of the leakage.*

The same procedure was performed using the same measurement for s-box structure 0 resulting in the upper diagram in Fig. 3.13. The signal from s-box structure 0 exhibits a significantly low amplitude which is explained by the fact that the measurement position is close to s-box structure 1 and further away from s-box structure 0. *I conclude that the distance to parts of the design in x- and y-directions is important for the detection of leakage*

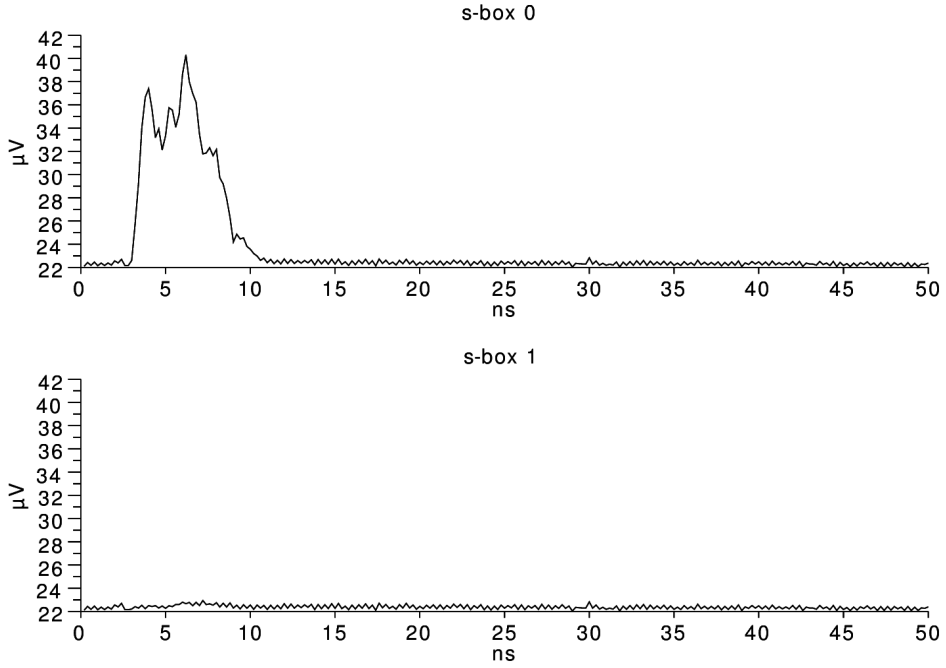



Figure 3.15: Noise standard deviations $\mathbf{s}_0^{noise}$ and $\mathbf{s}_1^{noise}$ over clock cycle for s-box 0 and 1 at $P_1$ (frontside, horizontal coil)

*signals.* This dependence on the measurement distance should theoretically follow a $\sim \frac{1}{r^3}$ as mentioned in Sect. 3.1.1.

Since the signal from s-box 0 is not significant at position $P_1$, I depict the corresponding result for s-box structure 0 for a different measurement position $P_0$ with coordinates $(x, y) = (24, 26)$, which is approximately above s-box structure 0. Fig. 3.12 shows the mean $\overline{\mathbf{m}}_0$ and standard deviation $\mathbf{s}_0$ of the repeated sequence of values processed by s-box structure 0 at position $P_0$. The result exhibits similar features as the previously described one for structure 1. However, the signal sequence is longer, containing 87 clock cycles, instead of 81 clock cycles which can be observed when comparing Fig. 3.12 and Fig. 3.11.

Figure 3.14 presents the data-dependent signals $\mathbf{s}_0^{data}$ and $\mathbf{s}_1^{data}$ within the clock cycle of both s-box structures. It is clearly obvious in the diagrams, how the signal from s-box structure 0 is detectable, and the signal from s-box structure 1 is very low. This state has switched due to the fact, that a different position, the one above the other structure has been used. This corresponds exactly to the localization assumption.

### Noise

Figure 3.15 depicts the noise standard deviations $\mathbf{s}_0^{noise}$ and $\mathbf{s}_1^{noise}$ over a cycle in respect to the other s-box structure at position $P_1$ above s-box structure 1. When comparing the noise figures to two the signals depicted in Fig. 3.13 which is from the same position $P_1$, it is clearly obvious how the signal from s-box structure 1, contributes as algorithmic noise for the s-box structure 0. Therefore, high noise amplitudes can be observed in the upper diagram. The lower diagram depicts the noise in respect to structure 1. From the preceding signal analysis, it is already obvious that there is only a small detectable signal amplitude of the other structure, structure 0, at this position. Hence the depicted noise corresponds almost entirely to the $\approx 22\,\mu V$ measurement noise floor.

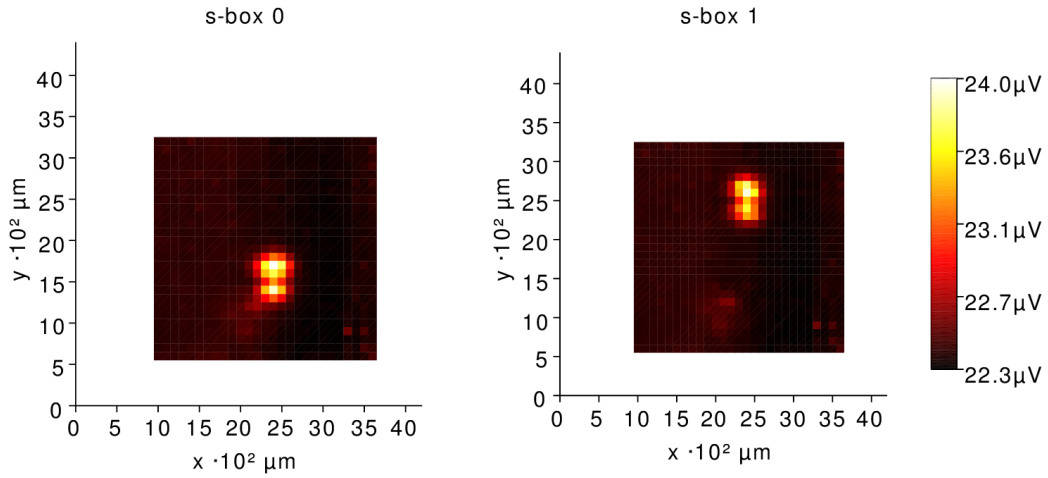



Figure 3.16: Noise standard deviation for each position (frontside, horizontal coil)

Figure 3.16 depicts a map of noise standard deviation values when using the horizontal coil from the frontside and after determining the noise in respect to the two signal generating structures. The values are in a range from 22.3 µV to 24 µV. This corresponds perfectly to the expected noise floor of $\approx$ 22.3 µV which was mentioned for the horizontal probe in Sect. 3.3.2. The regions where the noise is higher than this floor include algorithmic noise from other circuit parts. The noise values in respect to one signal generating structure always contain algorithmic noise from the other signal generating structure. This is obvious from Fig. 3.16, where the signal from s-box 1 introduces noise in the map of s-box 0 on the left-side map and vice versa.

### SNR

The SNR of the signal leakage of an s-box structure is computed as described in Sect. 3.3.3. Strong signal components of an s-box structure add to the algorithmic noise for the respective other s-box structure. The measurements from the frontside using the horizontal coil led to the highest SNRs.

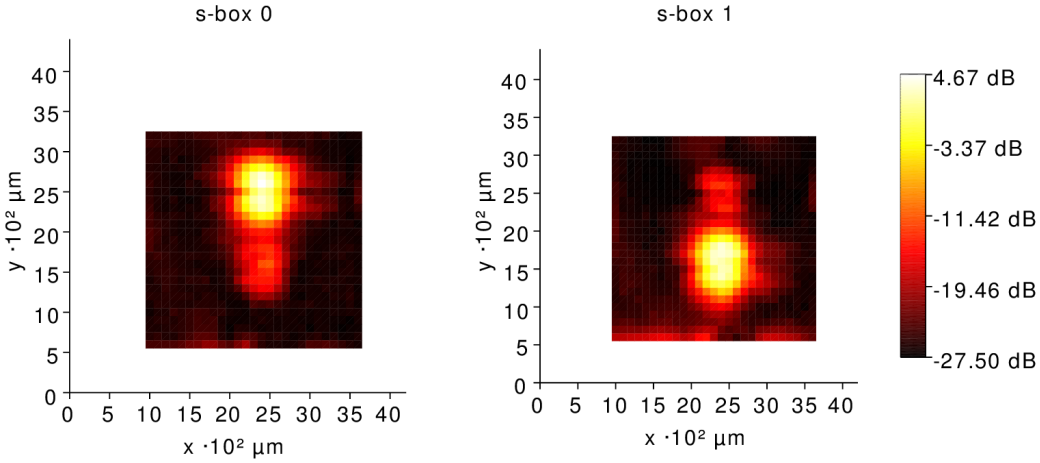



Figure 3.17: Maximum SNRs for both signals at each position (frontside, horizontal coil)

Figure 3.17 depicts a map of maximum SNR values for all positions and both s-box structures. The maximum of $\approx 4.7\,\mathrm{dB}$ represents a significant signal strength. *It is remarkable how the signals from the s-box structures are significant in areas above the placed logic of the structures as depicted in Fig. 3.4(b). This is an important result of this study. As another important conclusion, it is only possible to achieve high SNRs using a high-resolution probe when it is correctly positioned. This requires that adversary is able to find such positions, e.g., through profiling.*

Surprisingly, the SNRs are also above the lowest values of about $\approx -30\,\mathrm{dB}$ close to the respective other structure. I suspect that this is due to a remaining statistical correlation between the two selected sequences of value updates.

### Maximum Amplitudes

The easiest way to find eligible measurement positions is to look for positions with significantly high amplitudes. Figure 3.18 depicts maximum absolute EM measurement values for the measurement on the frontside using the horizontal coil. The values were derived by averaging the 0.05 ‰ highest

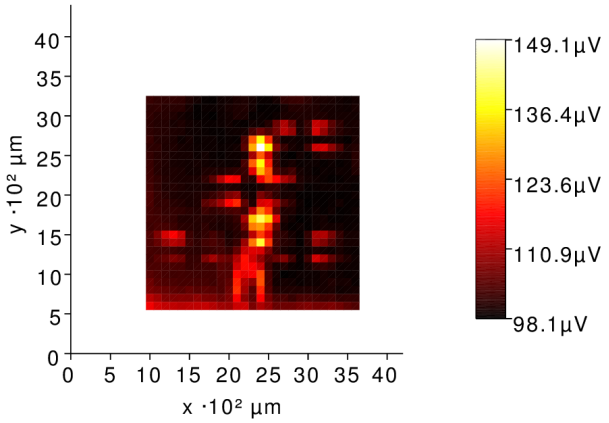



Figure 3.18: Maximum absolute EM values (frontside, horizontal coil)

absolute values from each measurement trace $\mathbf{t}$. When comparing this map of high values to the map of high SNR in Fig. 3.17, it is obvious that the two correlate. However, high SNR is also available at positions with lower amplitudes. And some positions with high amplitudes to not exhibit high SNR values.

*As an important conclusion, it seems partly feasible to locate positions with high localized signal leakage by looking for high measurement amplitudes for the frontside measurement case.* However, there is an important limitation to this observation. This only applies, if the targeted structure, e.g., cryptographic processor, is the only active structure. As soon as there is activity from other design parts, this cannot be used anymore.

As a side note, this confirms that the chosen vertical resolution of the oscilloscope prevents cut-off, since the maximum absolute value at 50 mV/DIV is 200 mV which is higher than the maximum values in both figures.

### 3.4.2 CPA and Localization

I performed CPA as described in Sect. 3.3.3. Figure 3.19 depicts the correlation coefficient over the clock cycle for both s-box structures at position $P_1$. High correlation values, positive as well as negative, can be observed for s-box structure 1 and insignificant correlation values for s-box structure 0. This corresponds well with the results from the previous sections where the signal is depicted in Fig. 3.13 for the same position.

At a sampling rate of 5 GS/s, the observed correlation peaks in Fig. 3.19 are only 1 to 3 samples wide. Therefore, I estimate a minimal required sampling rate of $\approx 2\,GS$. *This cannot be generalized, but I expect, that the*

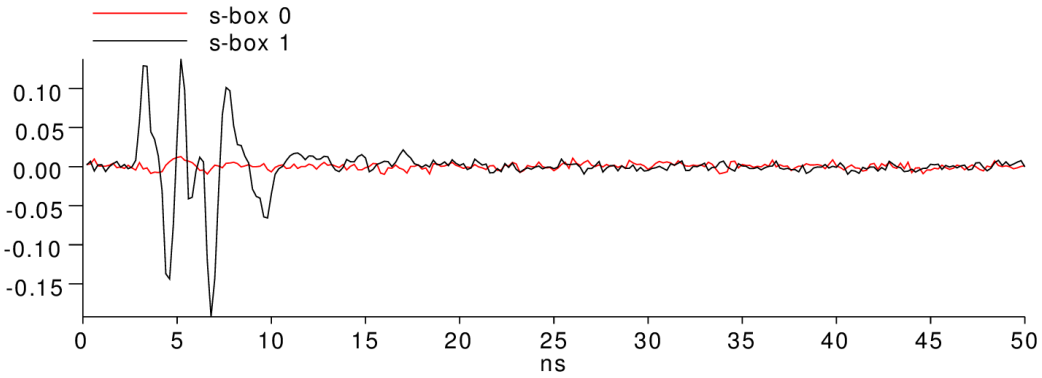



Figure 3.19: CPA over cycle at $P_1$ (frontside, horizontal coil)

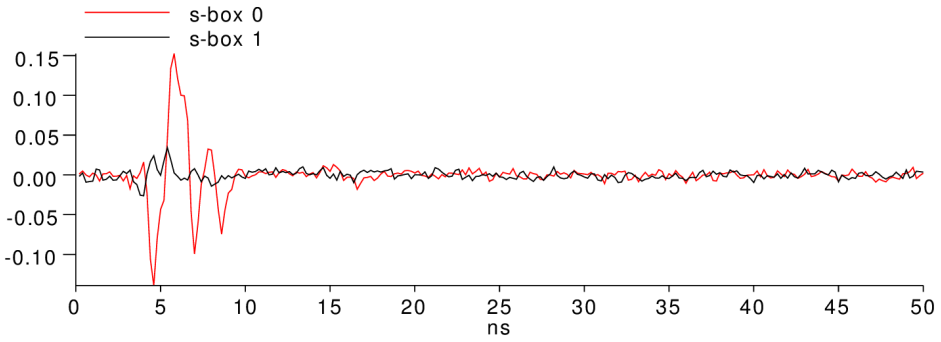



Figure 3.20: CPA over cycle at $P_0$ (frontside, horizontal coil)

*required sampling rate will always be in this range for localized EM measurement.*

Figure 3.20 depicts the correlation coefficient over the clock cycle for both s-box structures at the other position $P_0$. At this position which is above s-box 0, high correlation coefficients for s-box 0 can be observed.

Figure 3.21 depicts a map of maximum absolute correlation coefficients for every measurement on the map. *I would like to strongly emphasize how perfectly distinct the areas with high correlations of the two s-box structures are. This provides the perfect precondition for localized CPA attacks where only a single s-box structure is targeted. However, it must be noted, that an adversary must be able to find those positions.*

Unfortunately, there is no available information about the physical size of the FPGA cells for the FPGA device which is used. During the scanning of the surface, a step size of 100 µm is used. It can be assumed that the distance between the centers of the leakage regions equals the distance of the structure centers in the placement. From this, I derive the distance of

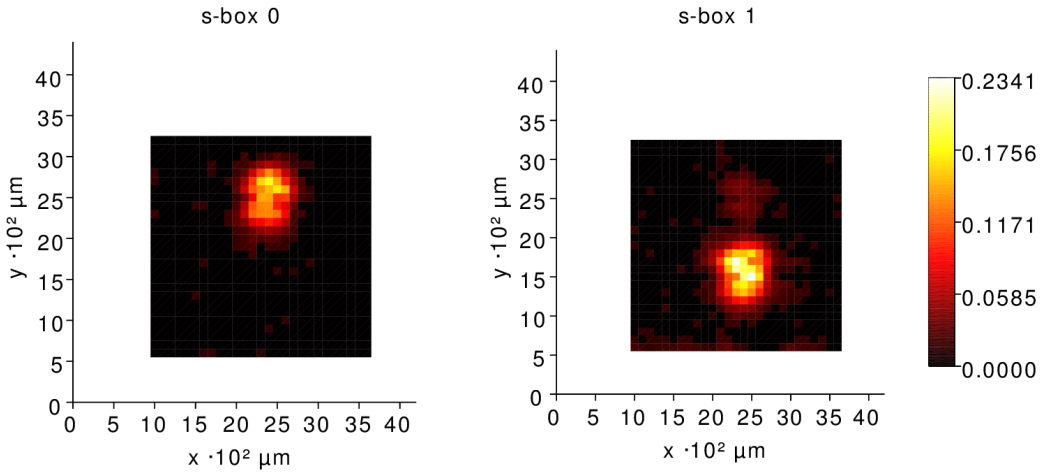



Figure 3.21: CPA coefficients (frontside, horizontal coil)

the s-box structure centers to be $\approx 900\,\mu m$ in the placement which is shown in Fig. 3.4(b). From this, it can be deduced, that the logic area of the two structures is $\approx 250 \times 250\,\mu m$. In Fig. 3.21 and Fig. 3.17, distinct leakage regions corresponding to the two s-box structures can be observed. Those regions with high correlation values of $\approx 0.23$ in Fig. 3.21 span areas of about $\approx 400 \times 600\,\mu m$. Hence, the regions of signal leakage span wider than their actual placement on the die.

In Fig. 3.21, it can be observed, that the regions with significant correlation coefficients do not overlap. *I derive that there are non-overlapping regions of significant correlation coefficients even when the two s-box structures are adjacent to each other, thus, when the centers are only* $\approx 250\,\mu m$ *apart.* This is strongly dependent on the logic structure of the device, hence, I do not suggest generalization. *An interleaved placement of s-box structures will render this significantly more difficult, if not impossible, because of entirely overlapping regions of correlation coefficients. However, measurement equipment with higher resolution may still support localization.*

### 3.4.3 Backside versus Frontside Measurement

Decapsulating a chip from the backside can for some chip packages be achieved with less effort than from the frontside. Hence, it is an important question which preparation leads to better results. Figure 3.22 depicts the SNR map from using the horizontal coil from the backside. In the middle of the device, the localized signal leakage of both structures is clearly visible in distinct, confined regions. The maximum SNR is $\approx 15\,dB$ lower than in

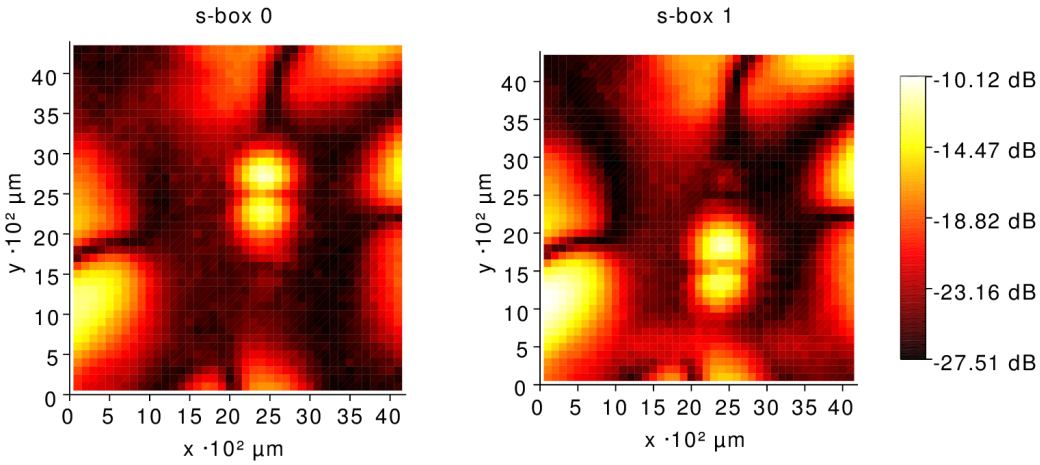



Figure 3.22: SNR is ≈ 15 dB lower on backside (horizontal coil)

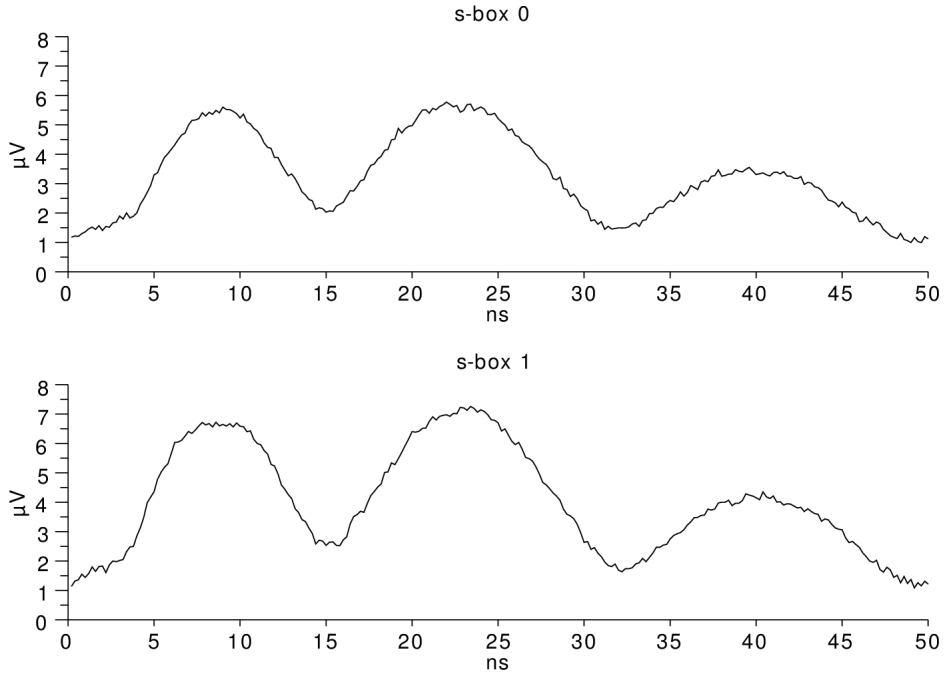



Figure 3.23: Data-dependent signal standard deviations $\mathbf{s}_0^{data}$ and $\mathbf{s}_1^{data}$ over clock cycle for s-box 0 and 1 at position $(1, 15)$ (backside, horizontal coil)

case of frontside measurement. This might be due to the silicone substrate and the fact that the conductors within the integrated circuit, which carry the exploitable signals, are on upper metal layers and therefore, further away when measuring from the backside. *I conclude that backside measurements*

*lead to significantly lower SNRs.*

Additionally, regions with high SNRs are observed on the edges of the die. Those regions have not been covered by the frontside measurements and exhibit signals from *both* s-box structures. Similar to the case of current consumption measurements that will be described in a subsequent Sect. 3.4.7, significant low-pass filtering is expected due to on-chip capacitances and inductances in the supply network. Fig. 3.23 depicts the signal over the clock cycle for one measurement position $(1, 15)$ in such a region on the edge of the map in Fig. 3.22. By comparing this to Fig. 3.13, the low-pass filtering is obvious and it is visible how both signals are detectable at this position. *I deduce that the magnetic field in those regions is caused by bonding wires or parts of the chip supply and that such measurement positions are similar to current consumption measurements. Since those regions contain contributions from both structures, they are useless from a localized perspective.*

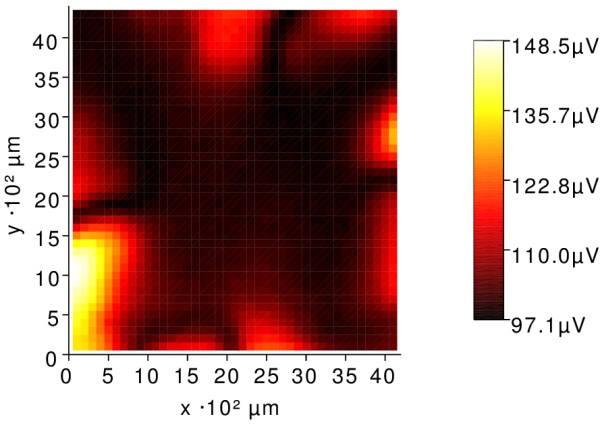



Figure 3.24: Maximum absolute EM values (backside, horizontal coil)

Figure 3.24 depicts the map of maximum absolute EM measurement values for the measurement from the backside using the same coil. It can be observed that the regions of high amplitudes do not show the interesting regions in the middle of the device, where localized leakage is detectable as shown in Fig. 3.22. Hence, if high measurement amplitudes were used as a criterion to select positions, they would not correspond to positions with high localized leakage. Only regions, where contributions from both structures are detectable would be found. *I conclude, that a search for high measurement amplitudes will unlikely lead to positions with high localized leakage in this case.*

### 3.4.4 Probe-to-Chip Distance

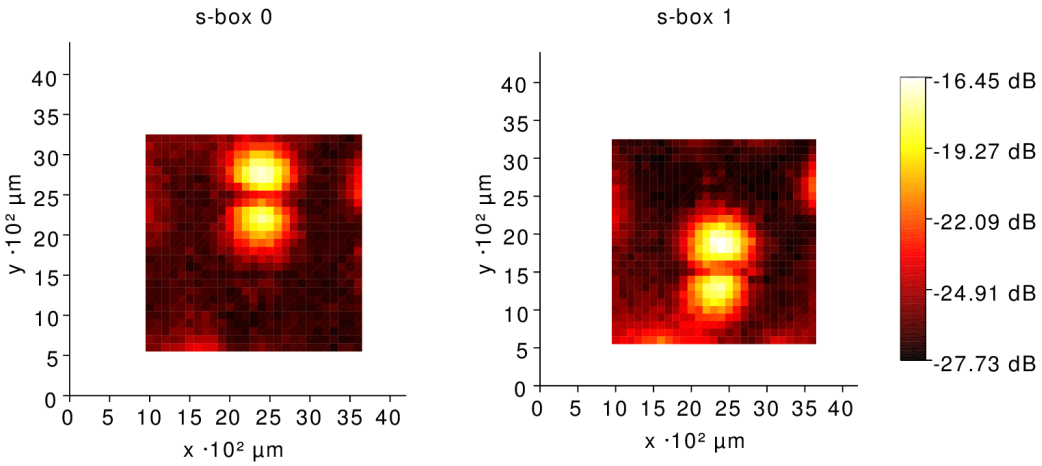



Figure 3.25: SNR at a distance increased by 300 µm (frontside, horizontal coil)

An important question is whether high-resolution EM measurements require semi-invasive depackaging to achieve minimal probe-to-die distances. I repeated the measurement from the frontside using the horizontal coil and increased the distance of the probe to the surface of the chip by 300 µm which roughly equals the package thickness above the die.

Figure 3.25 depicts the resulting SNR values. The measurements lead to a significantly lower maximum SNR of −16.5 dB. This is ≈ 21 dB lower than the maximum SNR observed in the original measurement depicted in Fig. 3.17. Apart from this difference, the highest SNRs of the two s-box structures are still confined in distinct regions. *I conclude that the semi-invasive depackaging is important to achieve high SNRs.*

### 3.4.5 Vertical Coil

As described in Sect. 3.3.2 and depicted in Fig. 3.6 and Fig. 3.7, I evaluated the vertical coil in $x$-, and $y$-direction for measurements. Figure 3.26 depicts the SNR maps for the $x$-direction. A maximum SNR of $\approx -8.9$ dB is achieved which is significantly lower than in case of using the horizontal coil. It can be observed, that the regions with significant SNR corresponding to the s-box structures have different shapes than in case of using the horizontal coil which is depicted in Fig. 3.17. Instead of $\approx 800 \times 800$ µm for the case of the horizontal coil depicted in Fig. 3.17, the significant regions now extend more in the $x$- than $y$-direction and the extent in the $y$-direction is smaller. The

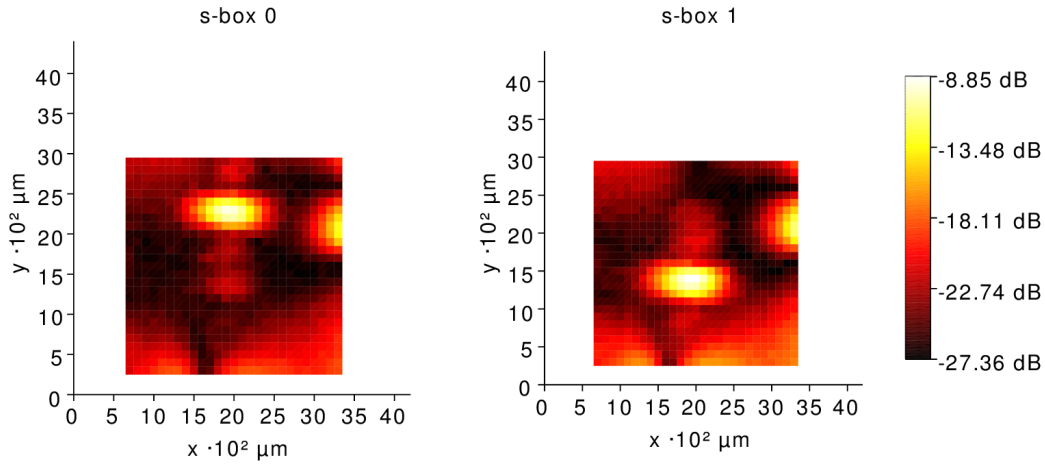



Figure 3.26: SNR using vertical coil in the $x$-direction (frontside)

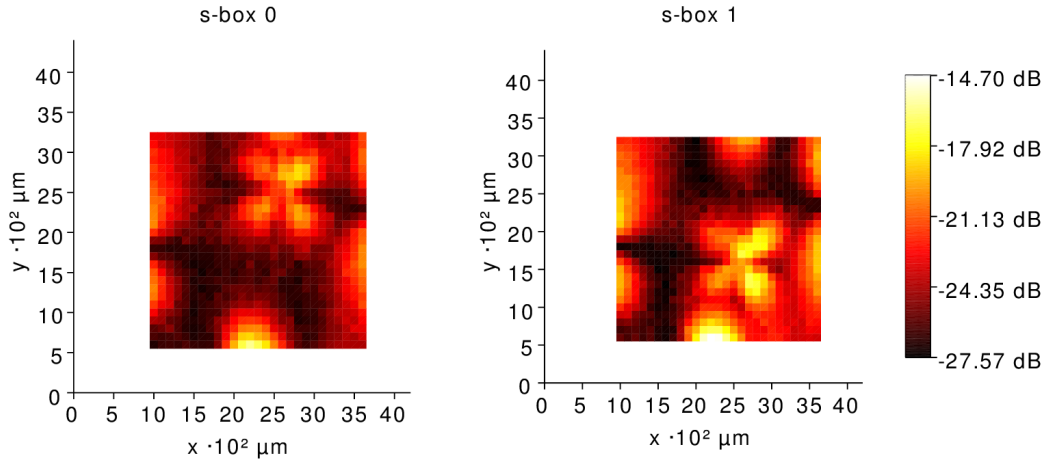



Figure 3.27: SNR using vertical coil in the $y$-direction (frontside)

observed regions have a shape of $\approx 700 \times 400\,\mu\text{m}$. This corresponds to the expectation that different coil orientations 'select' different parts of the field. In this case, the vertical coil in the $x$-direction provides a better selectivity between the two structure's signal regions.

Figure 3.27 depicts the SNR maps for the $y$-direction. It can be observed, that on the edges of the measurement region, there are detectable influences from the supply, similar to the case of the backside measurements from Fig. 3.22. In these regions, the leakage of both structures is detectable equally. In the middle of the device, there is localized leakage from the two s-box structures at approximately the same positions as in the previous figures. However, the SNR is even lower than in the measurements using the

vertical probe in $x$-direction, depicted in Fig. 3.26. The shape of the regions have an interesting appearance which must be due to the selectivity of the vertical probe in the $y$-direction (Fig. 3.7).

*Given those observations, I conclude that the probe with the vertical coil generally leads to lower SNRs compared to horizontal coils.* One reason for this is, that the centroid of the plane enclosed by the coil, hence the area which is traversed by magnetic flux, is further away from the integrated circuit surface in the case of a vertical coil. This is simply due to the fact that the coil is extending in a vertical direction and can be observed when comparing Fig. 3.6 and Fig. 3.5 which display drawings of vertical and horizontal coils.

Another conclusion is that, after profiling, vertical coils, even though providing less SNR, can be useful for better signal selectivity.

### 3.4.6 Trace Compression

Trace compression is popular to reduce data and computational complexity during side-channel analysis. I evaluated four methods which reduce $T_{CLK} = 250$ samples per clock cycle to a single value and repeated the CPA from Sect. 3.4.2 to benchmark the outcome.

During *maximum extraction*, one sample $c_i$ is selected for each clock cycle $\mathbf{t}_i^{cycles}$ which exhibits the maximum mean value over all cycles. This is described in Eq. 3.32 and Eq. 3.33. Figure 3.28 depicts a map of maximum absolute CPA coefficients when using the *absolute maximum* trace compression and the maximum correlation coefficient is 0.086.

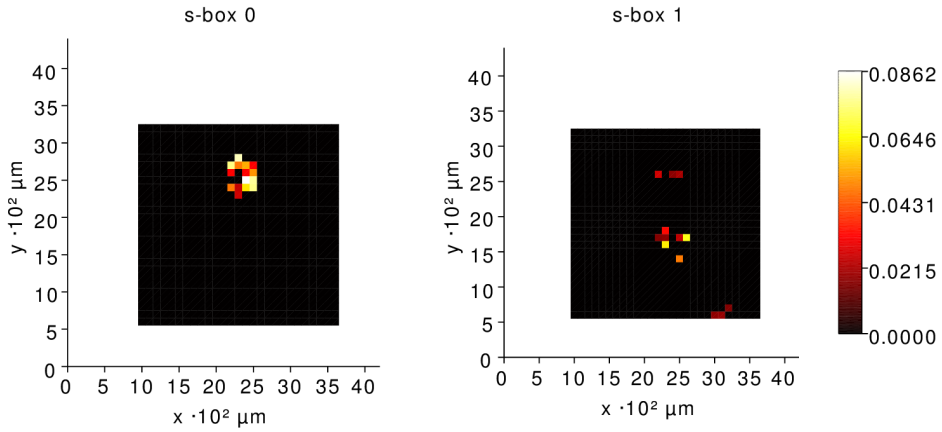



Figure 3.28: CPA using *absolute maximum* compression

$$j_{max} = \arg\max_j \overline{m}_j^{cycles},\ 1 \leq j \leq T_{CLK} \tag{3.32}$$

$$c_i = t_{i,j_{max}}^{cycles},\ 1 \leq i \leq N_{CYCLES} \tag{3.33}$$

*Peak-to-peak extraction* derives the distance between the two values with highest and lowest mean over all cycles to derive the new cycle values $c_i$. This is described in Eq. 3.34 and Eq. 3.35. Figure 3.29 depicts CPA coefficients using the *peak-to-peak* trace compression and the maximum correlation coefficient is 0.030.

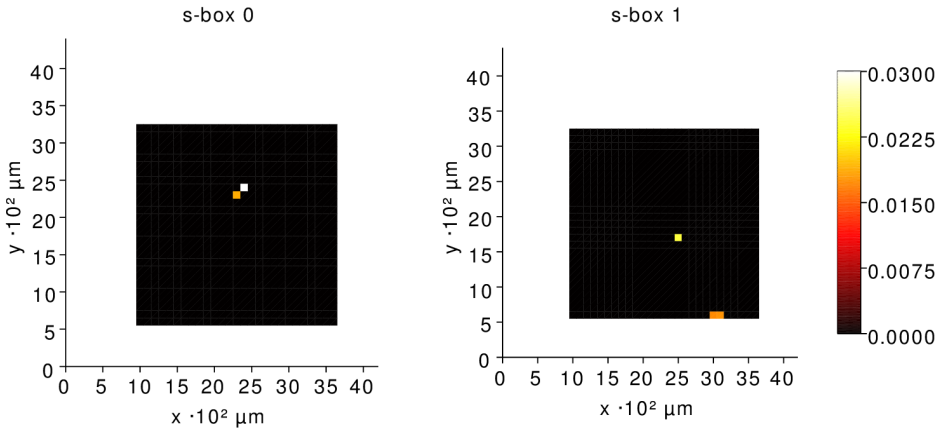



Figure 3.29: CPA using *peak-to-peak* compression

$$j_{min} = \arg\min_j \overline{m}_j^{cycles},\ 1 \leq j \leq T_{CLK} \tag{3.34}$$

$$c_i = (t_{i,j_{max}}^{cycles} - t_{i,j_{min}}^{cycles}),\ 1 \leq i \leq N_{CYCLES} \tag{3.35}$$

*Sum-of-absolutes* integrates absolute values over whole clock cycles to derive the new cycle values $c_i$. This is described in Eq. 3.36. Figure 3.30 depicts CPA coefficients using the *sum-of-absolutes* trace compression and the maximum correlation coefficient is 0.063.

$$c_i = \sum_{j=1}^{T_{CLK}} |t_{i,j}^{cycles}|,\ 1 \leq i \leq N_{CYCLES} \tag{3.36}$$

*Sum-of-squares* integrates squared values over whole clock cycles to derive the new cycle values $c_i$. This is described in Eq. 3.37. Figure 3.31 depicts results from using the *sum-of-squares* trace compression and the maximum correlation coefficient is 0.086

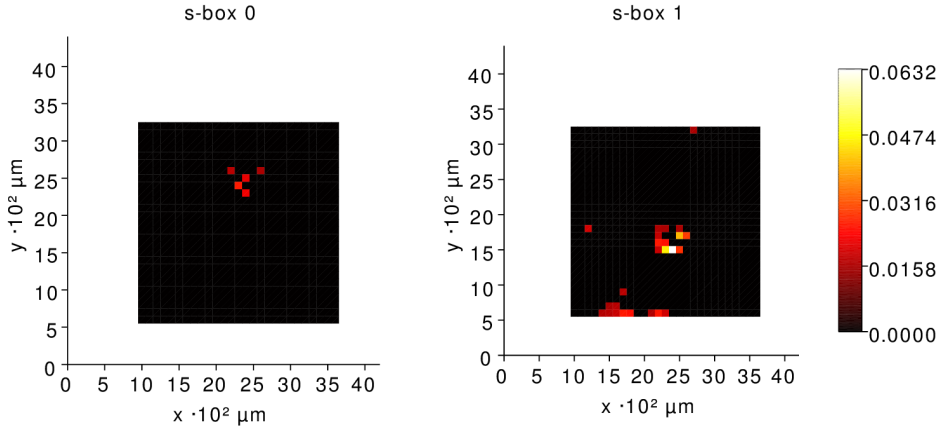



Figure 3.30: CPA using *sum-of-absolutes* compression

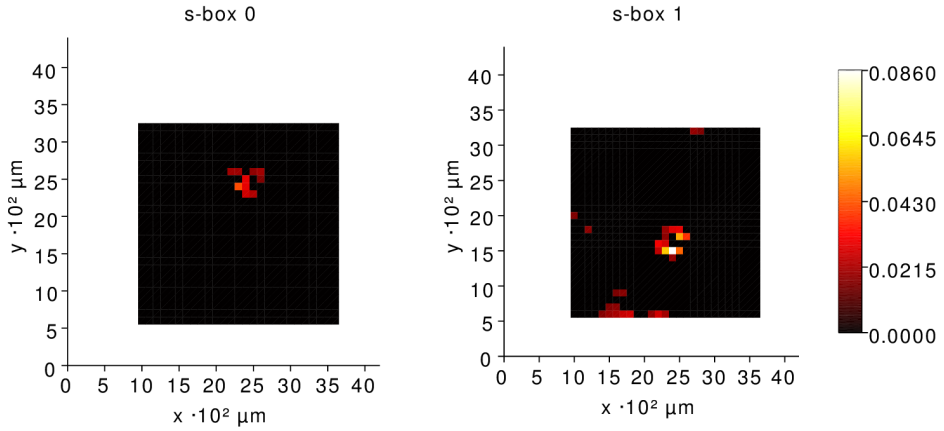



Figure 3.31: CPA using *sum-of-squares* compression

$$c_i = \sum_{j=1}^{T_{CLK}} (t_{i,j}^{cycles})^2, \ 1 \leq i \leq N_{CYCLES} \tag{3.37}$$

The original traces lead to a maximum correlation coefficient of 0.234 in Sect. 3.4.2, Fig. 3.21. Comparing the Figures 3.28 3.29 3.30 3.31 to the results without trace compression in Fig. 3.21, it is clearly obvious that besides the fact that the coefficients are significantly smaller, the regions of occurrence have a strongly altered appearance indicating a difficult use.

*Hence, all compression methods resulted in significantly lower correlation coefficients and I conclude that trace compression is generally inadvisable when analyzing high-resolution EM measurements.* The best one, sum-of-squares and maximum extraction reduced the coefficient by 63%.

For the maximum extraction and peak-to-peak extraction, I argue that this is due to the fact that the exploited signal is not leaked at times where high amplitudes are present. This can be observed when comparing the times with high amplitudes in Fig. 3.10, which are the peaks at $\approx 2\,\text{ns}$ and $\approx 30\,\text{ns}$ and the times when signal is leaking which is *after* the first peak at $\approx 2\,\text{ns}$ as observed in Fig. 3.13. However, a compression method, which simply removes unimportant parts in each clock cycle, e.g., samples between 15 ns and 50 ns in Fig. 3.13, will not influence the outcome. This becomes obvious from Fig. 3.19, where the correlation is detectable in the first part of the cycle only.

Trace compression may, however, be useful in the case of current consumption measurements. Such measurements exhibit a significantly low-pass filtered appearance. Therefore, the described trace compression methods might not reduce the contained information in such cases.

### 3.4.7 Current Consumption versus EM

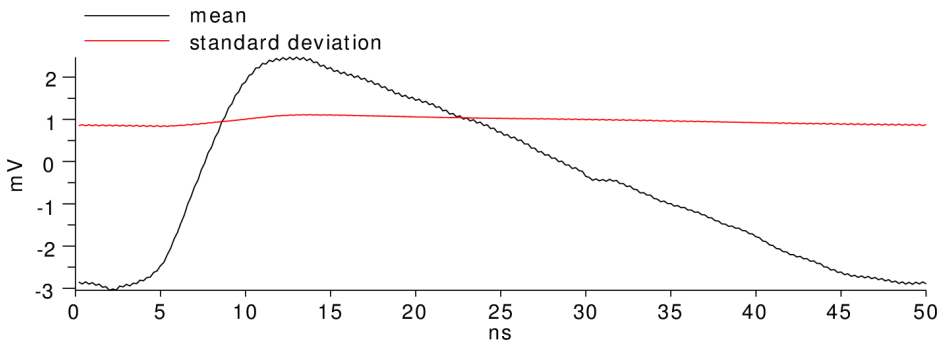



Figure 3.32: Mean $\overline{\mathbf{m}}^{cycles}$ and standard deviation $\mathbf{s}^{cycles}$ of all clock cycles for current consumption measurement

As described in Sect. 3.3.2 I performed a measurement of the current consumption for comparison. Figure 3.32 depicts the sample mean $\overline{\mathbf{m}}^{cycles}$ and sample standard deviation $\mathbf{s}^{cycles}$ of all clock cycles from one current consumption measurement.

Compared to a localized EM measurement depicted in Fig. 3.10, the shape is clearly low-pass filtered. Also the standard deviation, which contains the data-dependent signal at this point of the analysis (Sect. 3.3.3) is almost constant over the cycle. Compared to Fig. 3.10, this indicates that the data-dependent signal leakage only has a small influence and is distributed due to low-pass filtering. This is caused by the low bandwidth in the supply

network containing on- and off-chip capacitances and inductances [MOP07]. Thus, interference with adjacent cycles is observed.

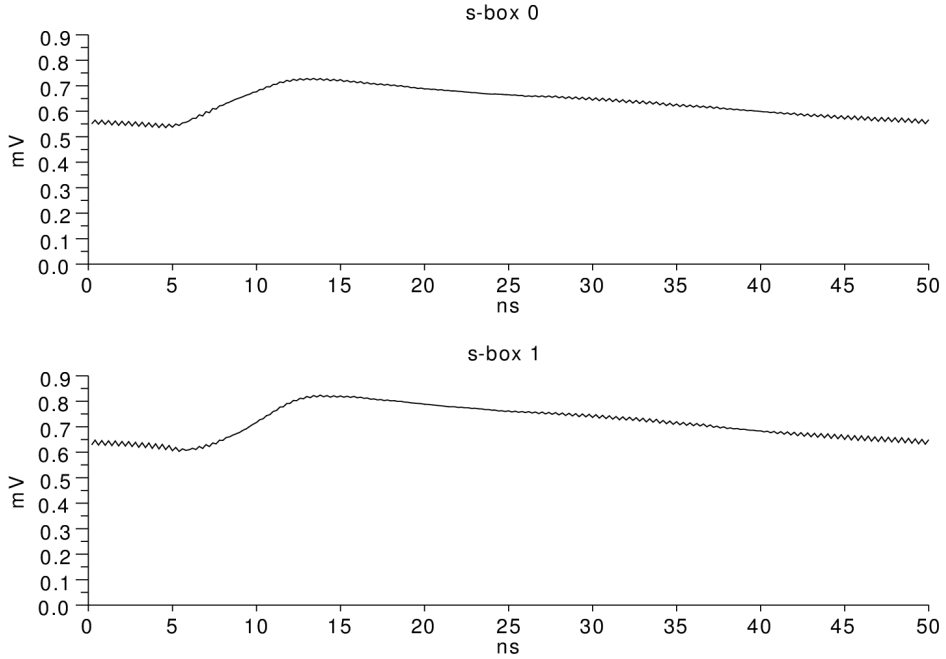



Figure 3.33: Data-dependent signal standard deviations $\mathbf{s}_0^{data}$ and $\mathbf{s}_1^{data}$ over clock cycle for s-box 0 and 1 (current consumption measurement)

Figure 3.33 depicts the data-dependent signal standard deviations $\mathbf{s}_0^{data}$ and $\mathbf{s}_1^{data}$ within the clock cycle of s-box structures 0 and 1. It can be observed, that the leakage of both structures is visible in similar signal strengths. The remaining small difference might be due to the fact, that the two structures are placed and routed differently even though the RTL design is equal. *It can be noted again, that the difference in signal strengths from different parts of the circuit is significantly higher when performing localized measurements.*

It can also be observed, that the signal leakage is detectable over the whole clock cycle almost constantly instead of just during a short time after the active edge. On the one hand, this makes current consumption measurements more robust against misalignment in differential attack settings. However, on the other hand, a maximum SNR of only 0.9 dB and a maximum correlation coefficient of only 0.094 are detected. This is significantly lower than the maximum observed SNR of 4.7 dB and maximum correlation coefficient of 0.234 in the case of high-resolution EM measurement. This is due to the overlap and additional switching noise from other circuit parts.

*I conclude, that localized EM measurements prevent interference of signals across multiple cycles at high clock frequencies of the design under test and provides significantly higher SNRs.* However, this requires to be able to position the probe correctly.

*The case of performing measurements of the magnetic field generated by the supply network, hence, for instance at the edge regions of the backside measurement in Fig. 3.22 leads to the exact same observations.* Hence, the benefits of localized EM measurements can only be taken advantage of at measurement positions close to the circuit surface and at positions close to the leaking structures.

## 3.5 Summary

The findings which have been presented in this chapter clearly prove the feasibility of localized measurements of electromagnetic fields. If the correct position for the measurement is found, this leads to higher SNRs than current consumption, or non-localized electromagnetic measurements. Such localized measurements will allow the exploitation of location-based leakage to attack, e.g., exponentiation algorithms. This will be described in Chap. 5. The security of masked implementations of symmetric cryptographic algorithms may also suffer from these findings.

Localized measurements are only superior in terms of SNR, if correct positions for measurement are known. In this case, signals of circuit parts can be recorded selectively and with minimal low-pass filtering from the supply network. Other positions even result in SNRs below detectability. Measurements from the die frontside lead to better results, and I generally recommend semi-invasive depackaging to achieve minimal probe-to-die distances. The horizontal probe provided higher SNRs while the vertical coil could be used to increase selectivity. High sampling rates, e.g., at least $> 1\,\mathrm{GS/s}$ will be required in most cases and compression of traces is generally not recommended.

I suggest that some of the presented conclusions about high-resolution EM measurements can be generalized to other integrated circuits such as FPGAs or ASICs.

# Chapter 4

# ECC Hardware Design

The feasibility of localized measurements of electromagnetic fields, which has been proven in the previous Chap. 3, enables side-channel attacks which exploit location-based leakage. I will present such an attack on implementations of exponentiation algorithms in the next Chap. 5. Exponentiation algorithms are important in asymmetric cryptography like for instance in Elliptic Curve Cryptograpy (ECC). To demonstrate the attack, an implementation of such an exponentiation is required. In this chapter, I describe the digital hardware design of a processing unit which is able to perform the elliptic curve operations. It is capable of performing Elliptic Curve Scalar point Multiplications (ECSMs), so-called exponentiations, and additions of elliptic curve points. Hence, the design serves as a design-under-test for the practical evaluations in the following Chapters 5, 6, and 7. Parts of this chapter have been published on HOST conference in 2010 [HS10].

I start this chapter by explaining the background of ECC along with high-level aspects of ECC implementations in Sect. 4.1. I then go on to explain the physical implementation security of ECC implementations in greater detail in Sect. 4.2. Section 4.3 presents related work in the area of hardware implementations of elliptic curve algorithms. The main Sect. 4.4 describes my ECC implementation and Sect. 4.5 summarizes this chapter.

## 4.1 Elliptic Curve Cryptography Background

This section introduces Elliptic Curve Cryptography (ECC). My aim is to establish a level of understanding which is required for applied cryptography, which includes implementations, physical cryptanalysis, and protection against physical cryptanalysis.

I give a historical perspective of how elliptic curves were introduced into cryptography and I put together explanations for the most important aspects of ECC. The presented information is extracted and re-combined from Hankerson et al. [HMV03], Asha et Gross [AG12], Werner [Wer02], and Paar et Pelzl [PP10]. For more details I refer to these sources.

In order to understand ECC, it is crucial to understand the general discrete logarithm problem along with a minimal amount of algebra which I explain in Sect. 4.1.1. In Sect. 4.1.2, I proceed to explain elliptic curves in great detail which leads to the elliptic curve discrete logarithm problem in Sect. 4.1.3. This section is completed with Sect. 4.1.4 about parameters and standardization and Sect. 4.1.5 giving an overview about the abstraction layers in ECC.

### 4.1.1 The General Discrete Logarithm Problem

One of the two most important mathematical problems which asymmetric cryptography is based on is the Discrete Logarithm Problem (DLP). The DLP in cyclic groups was discovered for cryptography in 1976 by Diffie and Hellman for key agreement [DH76] and ElGamal for signatures and public key cryptography in 1985 [ElG85]. The Digital Signature Algorithm (DSA) [NIS94, NIS09] is also based on the discrete logarithm problem. Understanding the DLP requires a few basic theorems from algebra.

#### Minimal Algebra

A group $G$ is a set of elements e.g., set of integers $\mathbb{Z}$, or finite set modulo the prime $p$, $\mathbb{Z}_p := \mathbb{Z}/n\mathbb{Z}$, together with a binary operation $\circ$, e.g., addition $+$, or multiplication $*$, which combines elements of $G$. Accordingly groups are called additive or multiplicative groups. The following properties must be satisfied:

1. The group operation is closed, i.e., $a \circ b = c \; \forall \; a, b, c \in G$.
2. The operation is associative, i.e., $a \circ (b \circ c) = (a \circ b) \circ c \; \forall \; a, b, c \in G$.
3. There is an identity element $1 \in G$ for which $a \circ 1 = 1 \circ a = a \; \forall \; a \in G$.
4. There is an inverse element $a^{-1} \in G \; \forall \; a \in G$ for which $a \circ a^{-1} = a^{-1} \circ a = 1$.
5. A group is Abelian if the operation $\circ$ is commutative, i.e., $a \circ b = b \circ a \; \forall \; a, b \in G$.

The order $\text{ord}(G)$ equals the cardinality $|G|$ of a group $G$ and is the number of elements in the set of the group.

The order of one element $a \in G$, $\text{ord}(a)$ is the smallest positive integer $k$ such that $a^k = a \circ a \circ a \circ ... \circ a = 0$. Or, equally, $a \in G$, $\text{ord}(a) := |\langle a \rangle|$ where $\langle a \rangle$ is the set of all multiples of $a$ regarding the group operation.

The basis for the definition of a hard DLP are finite cyclic groups. A group is cyclic if it has at least one element $\exists\, a \in G$ with $\text{ord}(a) = |G|$ which means that $G = \langle a \rangle$. This element $a$ is called a generator or primitive element because it can generate all other elements $a^i$ with $i \in [1, (|G| - 1)]$ of the group $G$ through repeated application of the group operation $\circ$. All elements $a^i$ can be seen as powers of $a$. Equally, $G$ is cyclic if there is one $a \in G$ with $G = \langle a \rangle$, which means that $a$ is a generator of $G$. According to Fermat's little theorem, $a^{|G|} = a$ if $|G|$ is prime which is the case in a cyclic group $G$.

However, not all elements $a$ of a cyclic group $G$ are necessarily generators of $G$. Elements $a$ which are not generators of the original group $G$, generate sub-groups $H$ of the first group. Lagrange's theorem states that the order of each such sub-group, $\text{ord}(H)$ divides $\text{ord}(G)$ if $\text{ord}(G)$ is finite. Hence, all elements $a \in G$ have orders which are factors of the group order $\text{ord}(G)$ and each sub-group of a cyclic group is cyclic.

From this it follows, that if the order $|G|$ of a group $G$ is prime $p$, the group is cyclic and all elements $a \neq 1 \in G$ have the same order $p$ and are all generators except for the identity element which has order 1.

A field $F$ is defined as the combination of two Abelian groups $G$ with additive and multiplicative operations $\circ$, $*$ and different identity elements over a set of elements where the multiplicative inverse does not apply to the additive identity element. Additionally the field fulfills:

- The distributivity law, $a * (b \circ c) = (a * b) \circ (a * c)\ \forall\ a, b, c \in F$.

### The Generalized Discrete Logarithm Problem

The Discrete Logarithm Problem (DLP) can be defined in any cyclic group, however, is a hard problem only in certain cyclic groups such as the multiplicative group of integers modulo a prime $p$, $(\mathbb{Z}_p, *)$. This group is used in Diffie-Hellman key agreement and ElGamal cryptosystems. The generalized DLP is the problem, given a cyclic group, or sub-group $G$ generated by a generator $a$, hence, $G = \langle a \rangle$, with large order and an element $b \in \langle a \rangle$, of finding $k \in [1, (|G| - 1)]$ such that $a^k = b$. It is called the discrete logarithm problem since the logarithm returns the power which the base has been raised to and the number space is discrete and finite. For finite groups such as the

*multiplicative group over a prime set* $\mathbb{Z}_p$, this problem is computationally hard.

The DLP is often referred to as a one-way function since, contrary to the logarithm, the computation of the $k$-th power of a generating element is computationally easy.

One general rule is that the order of the cyclic group which is used as a base for the DLP should generally be chosen as a prime to prevent the Pohlig-Hellman attack which exploits a possible factorization of the group order and, subsequently, the Chinese remainder theorem. In practice, e.g., in the case of ECC, this means that the DLP is often defined in sub-groups of prime order of groups with non-prime order.

All cyclic groups of the same order have essentially the same structure, however, with their elements labeled differently. The different representation of group elements results in different algorithms of different speeds for the group operations and for solving the DLP. [HMV03]

### 4.1.2 Elliptic Curves

Algebraic curves are curves that can be defined by a polynomial equation. The solutions to this equation, or zeros respectively roots of this equation represent a curve. The term *elliptic curve* refers to non-singular algebraic curves in two variables in the two-dimensional projective plane of homogeneous degree three. It does not simply refer to ellipses. Elliptic curves have been studied in number theory and algebraic geometry for many years before their application for cryptography was found [HMV03].

#### History

Newton formulated the chord-and-tangent addition rule for points on elliptic curves over rational numbers in the late 17th century [Kna92, pp. 9-12]. He observed that each new point on an elliptic curve which is defined over rational numbers either lies on the intersection of a tangent to the curve in a known point or the intersection of a chord through two already known points with the curve. Historically, this chord-and-tangent method was used to construct an additional point on a curve where two points were already known while using the algebraic operations in the underlying field. [Bro09]

These findings were subsequently improved by Jacobi, Weierstrass, and Poincaré. Essentially an operation to add points on elliptic curves, different or same points, which is called doubling, using the chord-and-tangent method was established. The chord-and-tangent point addition and doubling is derived from geometric reasoning, but can be expressed through algebraic

equations. An Abelian group was constructed using the set of solutions to the curve. [Wer02]

A general algebraic curve of degree three will be intersected by a line in at most, but not always, three points [AG12]. It required additional properties to be able to derive this group structure.

The curve must be non-singular, hence, not include singular points. In a singular point, the partial derivatives of the curve equation vanish simultaneously, which prevents the derivation of a tangent line in this point. However, the addition of two points which are the same point require the tangent. The intersection of the tangent with the curve is counted twice.

Instead of the two-dimensional affine plane, the two-dimensional projective plane had to be used. In the projective plane, there is one additional solution to the curve which is the point-at-infinity. This was necessary to make lines intersect the curve at this point-at-infinity which would otherwise only intersect the curve twice. This also means that this point-at-infinity can serve as an identity element for the Abelian group structure. [Kna92, Bro09, PP10]

These properties allowed a well-defined Abelian group structure where every line which intersects the curve in two points, possibly the same ones, intersects the curve in a third one.

### Elliptic Curve Equation

The above described findings lead to the following definition of elliptic curves. Elliptic curves are non-singular polynomial equations over two variables $f(x, y)$ of degree three with $x, y$ from a field $F$ which allow, in the two-dimensional projective plane, to define a group structure over $E(F)$.

The algebraic formulae for point adding and doubling require two operations along with their inverse in an underlying field $F$, e.g., addition, subtraction, multiplication and division, and are derived from the geometric description. Hence, an elliptic curve is defined over an underlying field which means that the variables and coefficients are elements of the field $F$.

The affine form of such an elliptic curve polynomial is called Weierstrass equation and the most general form is depicted in Eq. 4.1 with variables $x, y$ and coefficients $a_1, \ldots, a_6 \in F$.

$$f(x, y) = y^2 + a_1xy + a_3y - x^3 - a_2x^2 - a_4x - a_6 \tag{4.1}$$

A curve equation is given with a discriminant which has to be $\neq 0$ to determine whether the curve is non-singular under given parameters. This discriminant can be derived from the partial derivatives of the curve and depends on the underlying field because all parameters are elements of the

field and the operations are defined by the field. Non-singularity ensures, that there are no points at which the tangent cannot be determined [HMV03].

The set of points on the curve is the set of solutions, or zeros of the curve polynomial, $\mathrm{E}_f(F) = \{(x, y) \in F \times F : f(x, y) = 0\}$. The solutions $(x, y)$ to Eq. 4.1 are elements of the two-dimensional affine $\mathbb{A}^2(F)$ plane over the field $F$, hence, $(x, y) \in \mathbb{A}^2(F)$. The point-at-infinity $\mathcal{O}$ is not a solution to this equation. However, the group definition requires this element as an identity element for the group structure. This leads to the two-dimensional projective plane.

### Projective Plane

In a two-dimensional projective plane $\mathbb{P}^2$, two-dimensional points are represented by equivalence classes in three variables. The equivalence relation describes, that every member of this equivalence class is a valid representation of the same point in the two-dimensional affine plane. The equivalence class is described in Eq. 4.2.

$$(X, Y, Z) \sim (tX, tY, tZ) \in F \times F \times F \backslash \{(0, 0, 0)\} \; \forall \; t \in F \backslash \{0\} \tag{4.2}$$

Hence, the projective plane is defined as $\mathbb{P}^2(F) = F \times F \times F \backslash \{(0, 0, 0)\}$ obeying the equivalence relation from above. The tuple $(0, 0, 0)$ is not part of the plane, hence, $(0, 0, 0) \notin \mathbb{P}^2(F)$.

If we move from an affine plane $\mathbb{A}^2(F)$ to a projective plane $\mathbb{P}^2(F)$, solutions to the curve equation are represented as $(X, Y, Z)$ and it is required that all equivalent representations of each point are solutions to the curve as well. This can be achieved by substituting $(x, y) = (\frac{X}{Z}, \frac{Y}{Z})$ in the elliptic curve equation to derive an equation over three variables $X, Y, Z$. A multiplication by $Z^3$ leads to a homogeneous elliptic curve equation over three variables in the projective plane. A polynomial is homogeneous if all its monomial terms have the same degree. Homogeneous polynomials fulfill the requirement that all equivalent representations from the projective equivalence relation are solutions to it. Therefore, solutions from the set $\mathrm{E}_g(F) = \{(X, Y, Z) \in F \times F \times F \backslash \{(0, 0, 0)\} : g(X, Y, Z) = 0\}$ are representative for all $(tX, tY, tZ) \in F \times F \times F \backslash \{(0, 0, 0)\} \; \forall \; t \in F \backslash \{0\}$ which are also solutions. [Wer02, AG12]

The homogeneous, generalized Weierstrass elliptic curve equation $g(X, Y, Z)$ in the two-dimensional projective plane $\mathbb{P}^2(F)$ is depicted in Eq. 4.3. The equation is well-defined in the projective plane due to the homogeneity and can be transformed into the affine version by setting $Z = 1$, hence, $g(X, Y, 1) = f(x, y)$.

$$g(X,Y,Z) = Y^2Z + a_1XYZ + a_3yZ^2 - X^3 - a_2X^2Z - a_4XZ^2 - a_6Z^3 \quad (4.3)$$

The only reason for this seemingly complicated step is that the elliptic curve equation has an additional solution in the projective plane. The set of solutions of the elliptic curve equation in the projective plane equals the set of solutions in the affine plane including an additional point-at-infinity $\mathcal{O} = (X,Y,Z) = (0,1,0)$.

All solutions on the projective plane, except for $\mathcal{O}$, have an image on the affine plane by using the equivalent representative $(\frac{X}{Z}, \frac{Y}{Z}, 1)$ and setting $(x,y) = (\frac{X}{Z}, \frac{Y}{Z})$. All solutions in the affine plane have an image on the projective plane $(X,Y,Z) = (x,y,1)$. The definition of the group structure required the projective plane. However, since all solutions except for $\mathcal{O}$ have an image in the affine form, computations can also be performed on the affine form. [Wer02]

### Using Projective Coordinates for Computations

As already mentioned, all group operations can be performed in either the affine, or the projective plane as long as the point-at-infinity is not used. However, for instance during an elliptic curve scalar multiplication, the point-at-infinity is reached if the scalar is not reduced by the base point order $n$. This is deliberately the case when performing scalar blinding as a side-channel countermeasure. Therefore, in such cases, computations must be performed in the projective plane to avoid divisions by zero.

A convenient property of the group operations in the projective plane is that they do not require the calculation of field inverse. This makes projective representations the preferred choice for most implementations.

The above explanations refer to the so-called standard projective plane. There are other projective coordinate systems such as the López-Dahab projective coordinates [LD99b] which support further computational advantages for point adding and doubling.

### Simplifying the Curve Equation

The generalized Weierstrass equation can be simplified depending on the field $F$ that the curve is defined on. There are two reasons for this. One is the requirement for non-singularity, which is expressed by the determinant, will lead to the disappearance of certain terms.

The second reason is that the Weierstrass equation can be transformed into an isomorphic shorter version through a so-called admissible change of

variables [HMV03]. An isomorphism is a bijective transformation $f : G \rightarrow H$ between two groups with $f(a \circ b) = f(a) \circ f(b)\ \forall\ a, b \in G$. This means that it is essentially unimportant in which group a certain operation is performed since there exists a bijective transformation into the other group. Hence, there is no reason to use a more complicated curve equation because the structure will provide the same complexity.

Equation 4.4 depicts the simplified Weierstrass elliptic curve equation in the projective plane for the case that the curve is defined over a binary field $\mathbb{F}_{2^m}$.

$$g(X, Y, Z) = Y^2Z + XYZ - X^3 - a_2X^2Z - a_6Z^3 \tag{4.4}$$

Equation 4.5 depicts the curve equation over a binary field $\mathbb{F}_{2^m}$ in the affine plane.

$$f(x, y) = y^2 + xy - x^3 - a_2x^2 - a_6 \tag{4.5}$$

The Abelian group properties of elliptic curves are independent of the underlying field which the curve is defined on [PP10]. For more details on the algebraic equations for point adding and doubling and further topics such as supersingular curves see Hankerson et al. [HMV03], Werner [Wer02], or Paar et Pelzl [PP10].

### Underlying Fields

The group operations on elliptic curves require arithmetic operations in the underlying field. Algebraic fields have been defined in Sect. 4.1.1. The efficiency of those operations in the underlying field is an important factor for its choice.

In cryptography, finite fields, called Galois fields GF, are used which are defined using a finite set of elements. The most common kinds of fields used in ECC are prime fields GF($p$) with $p$ being a large prime characteristic and characteristic-two extension fields, or binary fields GF($2^m$).

Prime fields are popular because they have been used in RSA and DL-based cryptosystems before.

Binary finite fields GF($2^m$) in polynomial representation are especially convenient for digital processing. All coefficients are modulo 2, thus, digital, and the addition operation can for instance be implemented through a simple bit-wise XOR without carry. Fields with binary characteristic GF($2^m$) require a *prime* extension $m$ due to the Weil descent attack [GHS02]. A different application of binary field arithmetic are the substitution-box and mix-columns round transformations during the AES algorithm [NIS01].

Elements of binary fields GF($2^m$) in polynomial representation are polynomials over the variable $x$ of maximum degree $m-1$ with coefficients $a_i \in$ GF(2) as in Eq. 4.6.

$$\text{GF}(2^m) = \{a_{m-1}x^{m-1} + a_{m-2}x^{m-2} + ... + a_2x^2 + a_1x + a_0 : a_i \in \{0,1\}\} \quad (4.6)$$

Every such field can be represented as GF($2^m$)$[x]/p$ with an irreducible reduction polynomial $p$ of degree $m$. The field elements, especially the results of multiplications of elements, are seen as the unique remainder of degree less than $m$ after polynomial division through the reduction polynomial $p$. This is called reduction.

Due to the isomorphism of groups, it is essentially unimportant which of the possible reduction polynomials is chosen to construct the binary field in polynomial representation since the difference is only in labeling of the elements [HMV03]. However, a polynomial of degree $m$ with a minimum number of coefficients $\neq 1$ is convenient for implementations.

While binary finite fields GF($2^m$) are especially suited for hardware implementations, prime fields GF($p$) are better suited for software implementations and use similar algorithms like implementations of RSA.

Most protocols require regular modular integer operations in addition to the field operations. A hardware implementation for prime fields naturally supports such operations with minimal modifications. Hardware implementations for binary fields do not support those and, hence, require additional dedicated functionality. Nonetheless, Wenger et Hutter [WH12a] state that even with the integration of additional regular modular integer arithmetic, GF($2^m$) ECDSA [ANS05] hardware implementations outperform GF($p$) ECDSA hardware implementations by a factor of $\approx 2$ in terms of computation time, or, equally, energy consumption.

### 4.1.3 The Elliptic Curve Discrete Logarithm Problem

In 1985, Lenstra [Len87] proposed an integer factorization algorithm based on elliptic curves.

This independently inspired both Miller [Mil86] and Koblitz [Kob87] to apply the discrete logarithm problem to groups of points on an elliptic curve over a finite field. The fact that the points on a homogeneous projective Weierstrass elliptic curve including the point-at-infinity and the chord-and-tangent addition operation form an Abelian group was already known at the time. The novelty was to select an elliptic curve equation and underlying field in a way as to make the DLP on the group structure harder than on the multiplicative prime groups which had been used previously.

The Elliptic Curve Discrete Logarithm Problem (ECDLP) is the problem of determining the integer $k \in [1, n]$ given an elliptic curve $E$ defined over a finite field GF($q$), the generating point, or base point $P \in E(\mathrm{GF}(q))$ of order $n$ and the point $Q = kP$ from the cyclic sub-group $\langle P \rangle$ generated by $P$.

Repeated addition is called scalar point multiplication where the number of additions is the scalar. This is the analogue to exponentiation in multiplicative groups and an elliptic curve scalar point multiplication can be implemented through similar algorithms like exponentiations.

The main reason for the fact that the ECDLP is harder than the DLP on multiplicative groups is that the so-called index-calculus algorithm [HMV03], which can be used to solve the DLP on multiplicative groups in sub-exponential time, does not work on cyclic groups over elliptic curve points. Only generic algorithms to solve the DLP can be used.

Such generic algorith only use the group operations [PP10]. The best known generic algorithm to solve the discrete logarithm problem is Pollard's rho method. This algorithm also applies to groups over elliptic curve points and has a run-time which is exponential in the parameter size of the group order. The run-time is $2^{n/2}$ where $n$ is the bit-size of the group [PP10]. This means that in order to achieve a security level of 80-bits, a group size of $2^{160}$ must be used. Therefore, the advantage of a DLP defined over groups on elliptic curves is that the bit-size of parameters is shorter than for DLPs over multiplicative groups [HMV03].

To set up a hard ECDLP, it is important to know the order of the group because the hardness of the ECDLP depends on the order of the cyclic sub-group which is generated by the base point. The order of a group of points on an elliptic curve can be estimated using Hasse's theorem. The theorem states that the order of the group is roughly in the range of the order of the underlying field. The order of the group can then be factored in polynomial time to determine the orders of the sub-groups. The generator of the cyclic subgroup with the largest available prime order is selected as a base point for the ECDLP. Standardized curves usually have a group order as a product of a small, e.g., 2, and a large prime. [PP10]

The selection of ECC domain parameters is done by choosing an underlying field and then deriving the simplified Weierstrass equation for this field. Finally curve parameters can be chosen randomly and it must be checked, whether the generated curve has a large cyclic sub-group.

The ECDLP is hard, if there exists no known attack which is faster than exponential time in the group size. This requires that mathematical properties which lead to specific weaknesses of specific curves are prevented. Since this requires profound knowledge in mathematics and cryptanalysis, it is convenient to use standardized parameters and curves which considered such

properties as described in Sect. 4.1.4. [PP10]

### 4.1.4 Parameters and Standardization

The German Federal Office for Information Security [Bun11] regards both popular underlying fields, prime fields GF($p$) with $p$ being a large prime number and binary fields GF($2^m$) with $m$ prime, as valid choices for EC-based electronic signature schemes and provides standardized parameters in their 'Algorithmenkatalog'.

The same holds for the American National Standards Institute in their publication ANSI ANS X9.62-2005 [ANS05] as well as for the National Institute of Standards and Technology in their publication FIPS 196 [NIS97]).

The parameters which are standardized are, the elliptic curve parameters $a$, $b$, the elliptic curve base point $P = (x_P, y_P)$, base point order $n = \text{ord}(P) = |\langle P \rangle|$, co-factor $h$ of the curve order, and the reduction polynomial in case of GF($2^m$).

The German Federal Office for Information Security [Bun11], in their Algorithmenkatalog, additionally provides recommendations for the parameter sizes to support a reasonable security level. The latest recommendation for elliptic curve-based digital signatures is a secret key length, thus field size, of at least 224-bits until 2015. A field size of 224-bits corresponds to a security level of 112-bits [ANS05, p. 34].

### 4.1.5 Layers of ECC

An elliptic curve cryptosystem can be seen as a layered structure [PP10]:

1. The bottom layer is an underlying finite field.

2. The second layer is the algebraic cyclic group structure. It is defined through a point adding and doubling operation on a set of points which is generated by one point on the curve. This requires algorithms for adding and doubling on the elliptic curve in a projective or affine plane on the underlying field.

3. The third layer is the ECDLP which is based on the elliptic curve scalar point multiplication one-way function.

4. The top layer is the protocol or cryptographic scheme which is built on top of the ECDLP. Examples are Diffie-Hellman key agreement, the ElGamal cryptosystem, or the elliptic curve version of the DSA, ECDSA [ANS05].

*Here, it becomes obvious that ECC includes two different finite algebraic structures, an underlying finite field and a cyclic group of points on an elliptic curve.*

**Abstraction for Practitioners**

For practitioners, implementers, as well as researchers in physical cryptanalysis an abstract understanding of elliptic curves is sufficient. This understanding should cover:

1. Understanding the ECDLP generally.
2. Being able to choose an underlying field based on: recommendations, the application, or possible re-use of implemented routines.
3. Employing only standardized curve parameters, which guarantee a certain level of security.
4. Using the given elliptic curve equation and algebraic equations for the group operations which can be found in literature for the chosen underlying field.

Building on these basics, the choice of implemented algorithms and protection mechanisms against physical cryptanalysis remains as a field of action.

## 4.2 Physical Security for ECC

Elliptic curve cryptography was explained in Sect. 4.1. The main operation in all ECC-based protocols is the Elliptic Curve Scalar Multiplication (ECSM). This operation is the one that the security is based on and which has to be protected against physical cryptanalysis. The state-of-the-art in securing ECSM computations against side-channel analysis and fault attacks was presented by Fan et al. [FGDM+10] in 2010.

As introduced in Section 2.2, attacks have different requirements. Fan et al. [FGDM+10] introduced the following terms to describe those most important properties of attacks:

SE - Single-Execution. This means that the attack requires only a single observed execution.

ME - Multiple Execution. This means that the attack requires multiple observed executions.

CI - Chosen Input. The adversary must be able to chose the input, e.g., ciphertext, to the device-under-attack. Other attacks require that the adversary deliberately choses the secret scalar for a prior profiling of the same or an equal device.

The number of possibly observed executions with a constant secret for an adversary depends on the protocol. The ability to deliberately chose inputs to the design-under-test depends on the application.

| Passive attacks | SE | ME | CI | Recommended countermeasures |
|---|---|---|---|---|
| Timing analysis [Koc96] | x | x | | * Montgomery powering ladder [JY03, LD99a]<br>* Double-and-add-always [Cor99]<br>* Indistinguishable point addition and doubling |
| Simple SCA [KJJ99] | x | | | * Montgomery powering ladder [JY03, LD99a]<br>* Double-and-add-always [Cor99]<br>* Indistinguishable point addition and doubling<br>* Window exponentiation |
| Template attack [MO09] | x | | x | * Randomized projective coordinates [Cor99] |
| Differential SCA [KJJ99] | | x | | * Randomized projective coordinates [Cor99]<br>* Base point blinding [Cor99]<br>* Scalar randomization [Cor99]<br>* Random scalar splitting [FV03]<br>* Random field representation [CJ03] |
| Refined power analysis [Gou02]<br>Zero Value Analysis [AT03] | | x | x | * Scalar randomization [Cor99]<br>* Random scalar splitting [FV03]<br>* Base point blinding [Cor99] |
| Comparative SCA [HMA$^+$08] | | x | x | * Scalar randomization and base point blinding [Cor99] |

Table 4.1: Passive attacks and countermeasures for ECSMs according to Fan et al. [FGDM$^+$10]

Table 4.1 and Table 4.2 are extracted from Figure 1 and Table I in Fan et al. [FGDM$^+$10] and summarize attacks and countermeasures for ECSMs. The tables list possible attacks including references in the left column. In the three columns to the right, the required properties of the attack as described above are marked. Finally, the rightmost column lists possible countermeasures for each attack including references.

### 4.2.1 ECC Protocols

There are two major classes of ECC protocols:

1. Protocols that use a different random number as ephemeral secret scalar and constant base point in every ECSM. Examples include the ECDSA [ANS05], ECGDSA [ISO02], ECKDSA [ISO02], EC-El-Gamal signatures [HPM94] and EC-Schnorr identification [Sch90].

| Active attacks | SE | ME | CI | Recommended countermeasures |
|---|---|---|---|---|
| M-safe-error attack [YJ00] | | x | | * Montgomery powering ladder [JY03, LD99a]<br>* Unified memory access pattern [YJ00]<br>* Scalar randomization [Cor99] |
| C-safe-error attack [YJ00] | | x | | * Montgomery powering ladder [JY03, LD99a]<br>* Eliminate dummy operations [YJ00]<br>* Scalar randomization [Cor99] |
| Invalid point attack [BMM00] | | x | | * Point validity check at input and output [BMM00] |
| Invalid curve attack [CJ05] | | x | | * Curve integrity check [CJ05] |
| Twist-curve-based attack [FLRV08] | | x | | * Montgomery ladder with Y-coordinate [JY03, LD99a]<br>* Twist-strong curves [FLRV08]<br>* Point validity check with Y-coordinate [BMM00]<br>* Random scalar splitting [FV03] |
| Differential fault attack [BMM00] | | x | | * Point coherence check [DO08]<br>* Repeated point validity check [BMM00]<br>* Scalar randomization [Cor99] |
| Sign-change fault attack [BOS06] | | x | | * Montgomery ladder without Y-coordinate [JY03, LD99a]<br>* Point coherence check [DO08]<br>* Use combined curve to detect faults [BOS06] |

Table 4.2: Active attacks and countermeasures for ECSMs according to Fan et al. [FGDM+10]

2. Protocols that use a constant secret scalar and varying base point in the ECSM during multiple executions. Examples include El-Gamal encryption, the Elliptic Curve Integrated Encryption Scheme (ECIES) and a basic variant of EC-DH [DH76] key exchange without random numbers.

In the following sections, the two different classes are discussed with respect to required countermeasures against physical attacks.

### 4.2.2 Protecting ECSMs against Single Observation Attacks

The ECDSA algorithm is an example for the class of protocols which use an ephemeral secret scalar. In those applications, the ECSM must only be protected against single-execution attacks.

The generation of an ECDSA signature consists of two computations where $k$ is the randomly chosen ephemeral secret scalar, $P$ is the EC base point, $(x_Q, y_Q)$ the coordinates of the resulting point $Q$ after the ECSM, $n$ is the base point order, $h(m)$ is the hash value of the message $m$, and $d$ is the long-term secret key. For details about ECC, see Sect. 4.1.

$$(x_R, y_r) = k * P \tag{4.7}$$

$$s = k^{-1} * (h(m) + d * x_R) \bmod n \tag{4.8}$$

To recover the long-term secret key $d$, an adversary can either target the secret key $d$ itself or the ephemeral secret $k$ from which the secret key can be computed. This means that both, the ECSM in Equation 4.7, and the general modular arithmetic in Equation 4.8 have to be protected against physical attacks. In this thesis, I concentrate on ECSMs since the ECSM is the core operation of every ECC implementation and the most important target for adversaries.

The single-execution attacks on ECSMs are highlighted in Table 4.1 and Table 4.2. It is obvious, that SPA, SEMA, timing attacks, and template attacks have to be considered. To protect an implementation against SPA, SEMA, and timing attacks it is recommended to use the Montgomery powering ladder [Mon87, JY03, LD99a] which exhibits a constant processing time and constant sequence of operations, independent of the processed scalar.

Projective coordinate randomization as proposed by Coron et al. [Cor99] can be used to protect against template attacks to prevent profiling. There is no published fault attack on an ECSM in a single-execution setting.

### 4.2.3 Protecting ECSMs against Multiple Observation Attacks

If the same secret scalar is used for every protocol execution, the ECSM must be protected against multiple-execution attacks additionally. In Table 4.1 and Table 4.2, it can be observed, that many attacks have to be considered in this case. These include differential side-channel attacks as well as fault attacks.

To protect an implementation against SPA, SEMA, and timing attacks [Koc96] it is recommended to use the Montgomery powering ladder [Mon87, JY03, LD99a] which exhibits a constant processing time and constant sequence of operations, independent of the processed scalar. The Montgomery powering ladder also protects against M-safe and C-safe error attacks [JY03].

Projective coordinate randomization [Cor99] can be used to protect against template attacks to prevent profiling.

Projective coordinate randomization [Cor99] as well as scalar randomization (exponent blinding) [Cor99], or base point blinding [Cor99] protect against differential side-channel [KJJ99], and differential fault attacks [BMM00].

To protect against other fault attacks, the integrity of the base point and curve parameters must be verified [CJ05] and it must be verified, whether

input, and output points are in fact on the specified elliptic curve [BMM00].

## 4.3 Related Work on ECC Hardware Designs

Several hardware designs for elliptic curve processing units have been published in literature. Most published designs use elliptic curves defined over binary fields GF($2^m$) with polynomial base representation. This supports efficient hardware designs because the addition of two elements is a bit-wise *xor* operation.

Wenger et Hutter [WH12b] provide a practical study which confirms the advantages of binary field-based ECC for hardware designs in terms of runtime and implementation complexity, especially when the Elliptic Curve Scalar Multiplication (ECSM) operation is considered stand-alone.

The López-Dahab Montgomery multiplication algorithms [LD99a] are based on the Montgomery powering ladder [Mon87, JY03] and are the most popular algorithms for ECSM implementations. They are most efficient in terms of required field operations as well as storage since the $y$-coordinate is not required throughout the Montgomery powering ladder.

Most designs use standardized EC parameters from NIST [NIS99] under the denominator *Curve B-163*.

Kumar et Paar [KP06] described an EC processor design in 2006 which supports field size between 113-bit and 193-bit. In addition to the full-precision binary field multiplication and addition, they include a dedicated squarer unit and perform the binary field inversion based on the Itoh-Tsujii multiplicative inverse algorithm [IT88, RHSDPK06]. A binary field squaring can be performed in a single cycle. Thus, they are able to efficiently compute the inversion using the Itoh-Tsujii method. The field polynomial is hard-wired in the ALU. Kumar et Paar use a modified version of the López-Dahab Montgomery multiplication algorithms [LD99a] based on affine coordinates. This modification reduces the two required field inversions to a single one in each step of the ladder.

Later contributions use the López-Dahab Montgomery multiplication algorithms [LD99a] with projective coordinates to circumvent the binary field inversion during the Montgomery powering ladder. Only the final conversion to affine coordinates requires one inversion.

Lee et al. [LSBV08] as well as Bock et al. [BBD+08] proposed EC processors which employ a 163-bit field in 2008. Their designs include a full-precision arithmetic unit with hard-wired field polynomial. In the latest version of their processor design, Lee et al. [LBSV10] include a dedicated binary field squarer and the computation of affine output coordinates. This

is contrary to the design by Bock et al. [BBD+08] which do not support this.

Wenger et Hutter [WH11, WH12a] present a similar architecture which uses a custom 16-bit controller and datapath to achieve a lower implementation complexity. They also use a dedicated squarer in their 16-bit ALU and the Itoh-Tsujii method for binary field inversion to be able to transform the projective coordinates to affine coordinates at the end of the computation.

The design presented in this section is based on López-Dahab Montgomery multiplication algorithms [LD99a] with projective coordinates similar to previous contributions. I employ a full-precision arithmetic unit and provide configuration features to select 2- or 4-bit digit-wise binary field multiplications. I included a dedicated squarer unit as described by Kumar et Paar [KP06]. The affine $x$ and $y$ output coordinates are computed using Itoh-Tsujii method similar to Kumar et Paar [KP06]. The processing unit additionally includes additions of arbitrary EC points.

Comparing the implementation complexity and runtime of EC processor designs is challenging because there is always a trade-off. Additionally, different functional features, e.g., whether the computation of affine output coordinates is supported, and synthesis technologies, or different finite field sizes make it difficult. Nonetheless, Wenger et Hutter [WH12b, WH12a] recently provided a comparison of published hardware designs according to those parameters. In addition to binary field implementations, they also compare implementations based on prime fields.

The run-times of this design for different configurations are reported in Sect. 4.4.4 and are comparable to the published state-of-the-art.

## 4.4 Hardware Architecture

This section described the hardware architecture of the elliptic curve processing unit which is able to perform Elliptic Curve Scalar Multiplications (ECSMs) and elliptic curve point additions. For an application and implementation of ECC, decisions on the following abstraction layers have to be made:

1. *Finite field*
   A field characteristic, extension and, optionally, representation must be chosen along with the algorithms to perform the field operations. I chose a binary field with extension of 163 in polynomial representation due to the fact, that operations in binary fields can be implemented efficiently in digital hardware.

2. *Elliptic curve parameters*

Elliptic curve parameters such as field polynomial, $a$, $b$, base point $(x_P, y_P)$ and base point order $n$ are chosen, e.g., from a standard. I chose to use a NIST [NIS99] curve under the denominator *Curve B-163* for the chosen binary field. The field polynomial is hard-coded into the ALU of the design. The curve parameter $a$ is hard coded in the control part of the design as zero. The curve parameter $a$, $b$ and the base point $(x_P, y_P)$ are configurable.

3. *Coordinate system*
   Points on elliptic curves can be represented through, e.g., affine or projective coordinates. The implementation performance of point addition and doubling operations depends on the chosen finite field and coordinate system. I chose polynomial representation of binary field elements.

4. *Elliptic curve operations*
   The central elliptic curve scalar point multiplication consists of successive adding and doubling of points for which an algorithm must be chosen. I chose the algorithms from López and Dahab [LD99a].

5. *Protocol*
   An application requires the choice of cryptographic schemes and protocols based on ECC, e.g., ECDSA signatures, Diffie-Hellman key agreement. This processing unit supports various protocols.

The processing unit requires affine $x$- and $y$-coordinates of the base point $P$ and the scalar $d$ as inputs for performing an ECSM. It returns affine $x$- and $y$-coordinates of the resulting point $d \cdot P$ as a result. It provides several design-time configuration options in the source code which influence the area against computation time trade-off which are detailed in Sect. 4.4.4. As a first configuration option, the computation of the resulting point's $y$-coordinate can be omitted. This saves hardware for the storage of the base point's $y$-coordinate and computation time for the inversion. The next section describes the employed algorithms in greater detail.

### 4.4.1 Algorithms

The design implements the Montgomery powering ladder [Mon87, JY03] ECSM algorithm presented by López and Dahab [LD99a]. The algorithm is depicted in Alg. 1 and processes the binary-represented secret scalar $d \in \{0,1\}^l$-bitwise in a uniform operation sequence. It uses standard projective coordinates $X$ and $Z$ during the Montgomery powering ladder and no

**Algorithm 1** López-Dahab elliptic curve scalar multiplication algorithm [LD99a] using the Montgomery powering ladder [Mon87, JY03]

**Input:** Scalar $d = d_D d_{D-1} ... d_2 d_1$ with $d_i \in \{0, 1\}$, Point $P = (x_P, y_P) \in E$, Curve Parameter $b$

**Output:** Point $Q = d \cdot P = (x_Q, y_Q)$

1: $X_0 \leftarrow 1$, $Z_0 \leftarrow 0$, $X_1 \leftarrow x_P$, $Z_1 \leftarrow 1$
2: **for** $i = D$ downto 1 **do**
3: $\quad T \leftarrow Z_{1-d_i}$
4: $\quad Z_{1-d_i} \leftarrow (X_{1-d_i} \cdot Z_{d_i} + X_{d_i} \cdot Z_{1-d_i})^2$
5: $\quad X_{1-d_i} \leftarrow x_P \cdot Z_{1-d_i} + X_{1-d_i} \cdot X_{d_i} \cdot T \cdot Z_{d_i}$
6: $\quad T \leftarrow X_{d_i}$
7: $\quad X_{d_i} \leftarrow X_{d_i}^4 + b \cdot Z_{d_i}^4$
8: $\quad Z_{d_i} \leftarrow T^2 \cdot Z_{d_i}^2$
9: **end for**
10: $(x_Q, y_Q) \leftarrow \mathrm{Mxy}(X_0, Z_0, X_1, Z_1, x_P, y_P)$ **return** $(x_Q, y_Q)$

costly inverse in the binary field GF($2^m$) of extension size $m$ must be computed during the ladder algorithm. Depending on the scalar bit $d_i$, a fixed sequence of operations is executed with different addressing of the working registers ($X_1, Z_1$ and $X_2, Z_2$). The registers $T, x_P$ and $b$ are used in the same way, independent of the bit value. The $y$-coordinate is not used throughout this first part of the computation.

The original algorithm by López and Dahab presented in Algorithm 1 and the appendix of their paper [LD99a] was modified in two aspects. First, the algorithm handles a fixed size scalar. This can be observed in Line 2 when comparing it to [LD99a, Algorithm 1]. This means that leading zeros of the scalar are no longer ignored for computation time reductions. This is important in order to guarantee a constant run-time to protect against side-channel attacks. In order to support this, $\mathcal{O}$ and $P$ are used instead of $P$ and $2 \cdot P$ as starting points for the Montgomery ladder.

Second, *Mdouble* in the appendix of [LD99a] uses $\sqrt{b}$ as input. I modified the algorithm in order to use $b$ instead. This is useful for configuring the curve parameter $b$ from an external interface and did not have an influence on the algorithm runtime.

The affine $x$- and $y$-coordinates of the output point are computed from the projective coordinates in a routine which López and Dahab [LD99a] denoted as *Mxy*. This can be observed in Alg. 1. This coordinate transformation requires an inversion.

The inversion is performed using the Itoh-Tsujii Multiplicative Inverse Algorithm (ITMIA) [IT88, RHSDPK06] which can be efficiently mapped on

the available hardware. This mapping is more efficient than implementing the extended Euclidean algorithm. The ITMIA algorithm is an effective recursive re-arrangement of the required field operations through addition chains while calculating the inverse through the exponentiation $a(x)^{2^m-2}$ according to Fermat's little theorem, where $m$ is the constant extension, or bit-size, of the binary characteristic field GF($2^m$). Kumar et Paar [KP06] describe a similar implementation of this inversion method. It extensively uses squaring operations and is especially efficient in normal basis representation since squarings can be implemented as cyclic shifts. However, the same applies in the case of this implementation, since I included a squaring circuit which supports single-cycle squaring operations. With the ITMIA, the required operations are $(\lfloor(\log_2(m-1))\rfloor + HammingWeight(m-1) - 1)$ multiplications and $m-1$ squarings [IT88] where $m$ is the extension of the binary field GF($2^m$). A simple square and multiply method of computing $a(x)^{2^m-2}$ would require $m-1$ squarings and $m-2$ multiplications for comparison. This is a significantly higher effort.

### 4.4.2 Architecture

The processing unit consists of an ALU for operations in GF($2^{163}$), a control module to perform the algorithms and a storage module to store elements of GF($2^{163}$) as depicted in Fig. 4.1.

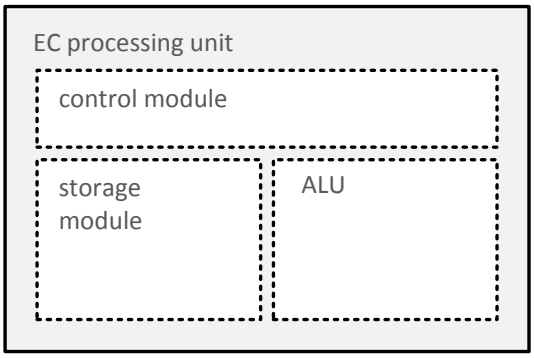



Figure 4.1: Architecture of the EC processing unit

The processor's ALU for the basic operations in GF($2^{163}$) features XOR-based additions, digit-serial multiplications [SP98] and a dedicated single cycle squarer circuit. The dedicated squarer circuit is optional and the inclusion can be configured during design time. The impact on the area and computation time will be detailed later. The digit-serial multiplier can be configured to either use 2-bit digits, or 4-bit digits. Obviously this has a sig-

nificant impact on the computation time and area requirements which will be detailed in Sect. 4.4.4.

The control part of the processing unit performs the following steps:

- Transformation from affine $x_P$ to projective coordinates $(X_P, Z_P)$.
- Optional initial randomization of projective coordinates $(X_P, Z_P)$.
- López and Dahab [LD99a], Montgomery powering ladder [Mon87, JY03].
- Optional computation of affine coordinates $(x_Q, y_Q)$ using Mxy and inversion.
- Optional check, if output point is on the curve using curve equation.

The storage module stores the $x$- and $y$-coordinates of the input base point $P$, the scalar $d$, and six further temporary elements of GF($2^{163}$) which are used in Alg. 1.

The design does in the current version, which is used for research purposes, not include a true random number generator. It uses a simple linear feedback shift register for this purpose instead.

### 4.4.3 Countermeasures

The protection of ECSM implementations against physical cryptanalysis is discussed in details in Sect. 4.2.

This design uses the Montgomery powering ladder [Mon87, JY03] ECSM algorithm presented by López and Dahab [LD99a]. This binary exponentiation algorithm processes the scalar bitwise with a uniform operation sequence which is independent of the data. This also applies to the ITMIA inversion in the end. The uniform operation sequence prevents most kinds of single-execution leakage, hence, SPA and timing attacks, and C-safe fault attacks [FGDM$^+$10].

However, in Sect. 5.2 I show that the algorithm still exhibits single-execution leakage, specifically location-based information leakage because it uses its working registers $X_i$ and $Z_i$ with $i \in \{0, 1\}$ differently, depending on the value of the processed scalar bit. This leakage, which I describe in Chap. 5, is not prevented by state-of-the-art protection mechanisms. Therefore, I present a countermeasure in Sect. 5.5.

The projective coordinates of the input point can be randomized [Cor99] as a countermeasure against DPA and against profiling for template attacks. After the Montgomery ladder computation using projective coordi-

nates, the resulting point in projective coordinates, conveys marginal information about the scalar [NSS+04]. However, after a transformation into affine $x$-coordinates this information within the projective coordinates is lost.

As a countermeasure against differential fault attacks [BMM00] which alter the base point, or any intermediate value, the design verifies whether the output point in affine coordinates is in fact on the curve (point verification) [BMM00]. However, this method does not protect the curve parameters, or the control flow.

### 4.4.4 Run-Time and Implementation Complexity

Different applications of an EC processing unit have different functional requirements. If the unit is used as an extension in a device which *generates* ECDSA signatures, an ECSM has to be performed, however, without computing the affine $y$-coordinate. In this case, protection against physical cryptanalysis is crucial, since an ephemeral secret scalar is handled.

If the unit is used as extension of a device which *verifies* ECDSA signatures, the ECSM computation must return affine $x$-, and $y$-coordinates of the output point. Furthermore, the processing unit must be able to perform an addition of EC points. Contrary to the signature generation case, the EC processor does not have to be protected against physical cryptanalysis since only public parameters are used.

Every functional feature of a processing unit increases its hardware complexity. The basic functionality of the design is an ECSM without computation of the output point's $y$-coordinate and without support for the addition of arbitrary EC points. Furthermore, the basic version includes an ALU which only supports 2-bit digit-serial multiplications in GF($2^{163}$) and no dedicated squarer unit. This basic version of the EC processor requires $\approx 176\,\mathrm{k}$ clock cycles to perform an ECSM. An FPGA synthesis for a *Xilinx XC3S1200E-5FG320* FPGA with high effort on minimal area led to the area requirement of $\approx 1,400$ flip-flops and $2,100$ four-input Look-Up Tables (LUTs).

The additional features which have been mentioned in the sections above can additionally be configured for the design:

- *Compute affine y-coordinate*
  Specifies whether the affine $y$-coordinate is computed in addition to the affine $x$-coordinate of the resulting point from the ECSM. This is for instance required to verify ECDSA signatures. This requires that the base point's $y$-coordinate is loaded into the device.

This feature increases the computation time in cycles by +0.5 % compared to the basic version. Furthermore, $2 \cdot 163$ additional flip-flops, and +20.1 % LUTs are required.

- *Include addition of EC points*
  Specifies whether the addition of arbitrary points on the elliptic curve is supported in addition to the ECSM. This also requires the above listed *compute affine y-coordinate* feature because the same amount of additional storage space is required. This feature is for instance necessary when verifying ECDSA signatures.

  The *compute affine y-coordinate* feature increases the computation time in cycles by +0.5 % compared to the basic version. Furthermore, $2 \cdot 163$ additional flip-flops, and +20.1 % LUTs are required. The addition of two arbitrary EC points requires $\approx 1,700$ cycles.

- *Verify output point*
  A verification whether the resulting point is on the specified elliptic curve is included as a countermeasure against fault attacks. This also requires the above listed *compute affine y-coordinate* feature to be able to use the curve equation.

  Together, the two features increase the computation time in cycles by +0.8 % compared to the basic version. Furthermore, $2 \cdot 163$ additional flip-flops, and +22.2 % LUTs are required.

- *Activate coordinate randomization*
  Specifies whether the projective coordinates are randomized as a countermeasure against DPA.

  This feature has an insignificant impact on computation time and implementation complexity.

- *Register location randomization*
  Specifies whether the storage location of certain variables is randomized during the ECSM computation. This countermeasure addresses location-based information leakage which is described in one of the main chapters, Chap. 5.

  The feature increases the computation time by +2.2 % compared to the basic version and has no impact on the implementation complexity.

- *Include dedicated squarer*
  Specifies whether the GF($2^{163}$) ALU includes a dedicated squarer circuit instead of squaring through the regular digit-serial multiplication.

This reduces the run-time of multiplications at the cost of additional circuitry.

This feature significantly decreases the total computation time by −45.6 % compared to the basic version. No additional flip-flops, but +12.4 % additional LUTs are required.

- *Four bit digit-serial multiplication*
  Specifies whether the GF($2^{163}$) ALU includes a 4-bit digit-serial multiplication instead of a 2-bit digit-serial multiplication. This reduces the run-time of multiplications at the cost of additional circuitry.

  The feature significantly decreases the total computation time by −45.9 % compared to the basic version. No additional flip-flops, but +23.3 % additional LUTs are required.

- *Load scalar at beginning*
  The scalar is processed sequentially. Therefore, the basic configuration loads the scalar byte-wise during the ECSM. This property specifies that the scalar is loaded into the design before the start of the computation at once.

  This feature does not alter the computation time compared to the basic version. However, 163 additional flip-flops, and +4.7 % LUTs are required.

- *Configure variable curve parameter b*
  Specifies whether the internal hard-wired curve parameter $b$ from NIST [NIS99] *Curve B-163* is used, or the parameter is loaded before the ECSM.

  This feature does not alter the computation time compared to the basic version. However, 163 additional flip-flops, and +3.7 % LUTs are required.

The above features can be selected according to individual assessments. The selection of some of them only depends on the desired application. The dedicated squarer unit as well as the four bit digit-serial multiplication feature support a significant decrease in computation time for reasonable additional hardware complexity. The countermeasures against physical cryptanalysis only have a slight impact on the computation time and hardware complexity. Therefore, their activation seems reasonable. Table 4.3 summarizes the configuration options along with the impact on computation time and implementation complexity.

| Hardware Features | ECSM Computation time | Implementation complexity |
|---|---|---|
| Basic functionality (reference for below) | $\approx 176$ k cycles | $\approx 1,400$ FFs, $2,100$ LUTs |
| + Affine $y$-coordinate | +0.5 % cycles | +2·163 FFs, +20.1 % LUTs |
| + Addition of EC points (requ. Affine $y$-coordinate) | Pt. Add. $\approx 1.7$ k cycles +0.5 % cycles | . +2·163 FFs, +20.1 % LUTs |
| + Verify output point (requ. Affine $y$-coordinate) | +0.8 % cycles | +2·163 FFs, +22.2 % LUTs |
| + Coord. randomization | $\ll$ | $\ll$ |
| + Reg. location rand. | +2.2 % cycles | - |
| + Dedicated squarer | −45.6 % cycles | +12.4 % LUTs |
| + 4-bit multiplication | −45.9 % cycles | +23.3 % LUTs |
| + Load scalar at beginning | - | +163 FFs, +4.7 % LUTs |
| + Variable $b$ | - | +163 FFs, +3.7 % LUTs |

Table 4.3: EC processing unit hardware configuration features. Influence on computation time in clock cycles and implementation complexity in Flip-Flops (FFs) and four-input Look-Up Tables (LUTs) compared to the basic functionality version as a reference.

## 4.5 Summary

This chapter describes the digital hardware design of a processing unit for elliptic curves defined over binary fields which includes state-of-the-art countermeasures against physical cryptanalysis. Most features of this processing unit are configurable at design-time to allow for flexible use. An FPGA-based implementation of the design is used for the work on location-based information leakage of exponentiation algorithms which I present in the following Chapters 5, 6, and 7.

# Chapter 5

# Localized EM Analysis of Exponentiation Algorithms

In this chapter I describe, how localized measurements of electromagnetic fields can be used to attack implementations of exponentiation algorithms. Localized means that the measurement is restricted to a certain spatial extent. I demonstrated the feasibility of performing localized measurements in Chap. 3. This has an important impact on implementations of exponentiation algorithms which are used in asymmetric cryptography. Localized measurements allow to exploit location-based information leakage to break implementations which are otherwise secure against side-channel analysis. Parts of this chapter have been published on CT-RSA conference in 2012 [HMH+12a].

Functional components such as registers within integrated circuits, or hardware implementations in particular, are distributed over the circuit area. A precise, localized measurement allows to distinguish the activity of registers on the circuit which are located at different distances to the probe. I found, that for certain algorithms, this location-dependent information leakage can be exploited in a dedicated side-channel attack. In particular, all cryptographic algorithms where the usage of registers depends on secret information are affected by side-channel attacks using *localized electromagnetic analysis*.

The main computation in DSA, RSA, and ECC-based cryptosystems is the modular exponentiation using a secret exponent or the elliptic curve scalar multiplication using a secret scalar. Binary exponentiation algorithms which are used in modular exponentiations for RSA and in Elliptic Curve Scalar Multiplications (ECSM) are examples of algorithms that are particularly susceptible to location-based side-channel attacks. In DSA or ECDSA, this exponent is different for every execution, e.g., chosen randomly

as ephemeral secret. (RSA uses the same exponent multiple times.) If the same exponent is used repeatedly, exponent blinding [Koc96] is often used as a countermeasure which also leads to a different exponent for every execution. Hence, an adversary employing side-channel attacks can only exploit single-executions to recover a secret exponent. To prevent SPA and timing attacks [Koc96], protected implementations either perform equal operations for different exponent bits or different operations, e.g., square and multiply, are implemented so that they appear equal.

In this chapter, I describe how fine-grained, localized EM measurements can be used to attack such implementations. The exploited side-channel leakage is the location-information during the computation which is recovered through localized measurements. This is contrary to conventional side-channel attacks which use data-, or operation-dependent leakage. I performed a practical evaluation of the idea using an FPGA-based implementation of the ECC hardware design described in Chap. 4. The practical results show how, after profiling to find a valid measurement position, localized electromagnetic analysis leaks sufficient information about the secret to recover it using a single trace.

Conventional countermeasures against side-channel attacks are ineffective against location-dependent side-channel leakage. As a general countermeasure for affected algorithms, the assignment of certain registers to physical locations should be randomized by swapping their locations at random times.

I start by presenting related work in attacks on exponentiations in Sect. 5.1. The main idea behind using location-dependent information for a dedicated attack is presented in Sect. 5.2 and the application to binary exponentiation algorithms is described in Sect. 5.3. I present the results of the practical study in Sect. 5.4 and how to protect the evaluated implementation using randomization of locations in Sect. 5.5. I summarize this chapter in Sect. 5.6.

## 5.1 Related Work

In this section I mention some other approaches to breaking exponentiation using side-channel analysis which are connected to the work presented in this chapter. Retrieving a secret exponent through the sequence of register addressing was introduced by Messerges et al. [MDS99b, MDS99a] and extended to ECC by Itoh et al. [IIT03a]. Many power traces are averaged in order to remove the data dependency of the power consumption. The remaining differences stem from the different power consumption of the addressing registers. The attack uses power measurements and relies on the

detectable and *different* power consumption of the addressing bits. This can be prevented by simple measures in the implementation. Contrarily, the work described in this chapter exploits differences which may be undetectable in power consumption measurements because they are due to the positioning of a probe. In the practical study, I demonstrate a successful attack without averaging to remove the data-dependent switching noise.

Another attack on register sequences was presented by Witteman et al. [WvWM11]. They present an SPA attack on the square-and-multiply-always RSA algorithm by finding consecutive operations which share the same input values which can be seen as the first application of side-channel based collision attacks to public key cryptography. To the best of my knowledge, the attack does for instance not apply to Montgomery ladder algorithms since no consecutive mathematic operations share the same input which is crucial for the attack. Therefore, when regarding the Montgomery ladder as a protection mechanism, the attack described in this chapter even works on protected implementations.

## 5.2 Location-Based Information Leakage

The fundamental cause for side-channel leakage, value changes in CMOS gates which lead to dynamic power consumption, is described in Sect. 2.2.1. The corresponding currents produce concentric magnetic fields around conductors which is explained in Sect. 3.1. Variations in the superposed magnetic field of multiple conductors can be measured using inductive sense coils. The field strength is proportional to current changes and decreases drastically with distance to the conductors. In order to measure the magnetic field of small regions of a device, a high spatial resolution, near-field sense coil can be placed close to the surface of an integrated circuit die. This is argued in Sect. 3.1.2.

In the following, I describe the main idea of exploiting location-dependent information leakage specifically for a side-channel attack. It is based on the hypothesis of being able to perform localized measurements of electromagnetic fields. This hypothesis was mentioned by other authors in the past [GMO01, AARR03] and was proven by my empirical findings in Chap. 3.

Cryptographic algorithms use registers to hold data values. Those registers are distributed over the integrated circuit. Figure 5.1 depicts a simplified magnetic near-field probe close to the surface of an integrated circuit die with three implemented registers $a$, $b$ and $c$. A register is written to by changing control lines and supplying it with a clock signal to update its internal value. All the involved logic cells, e.g., multiplexers in the datapath, processing the

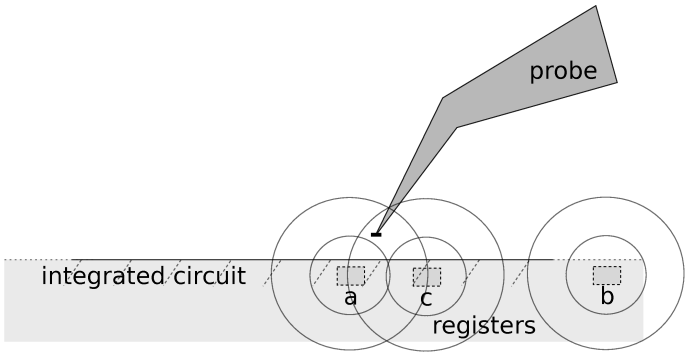



Figure 5.1: The distance to the current consuming circuit elements influences the measurement

value change consume dynamic power from the supply network which can be measured. A register which is not updated keeps the current value in a feedback mode or is even clock-gated, thus, not consuming dynamic power.

The probe in Fig. 5.1 is closer to register $a$ than to register $b$. Therefore, activity in register $a$ will lead to greater measured values than activity in register $b$. However, note that typically, the single bit cells which belong to one multi-bit register will not be located within confined areas. They will be located interspersed. Nevertheless, there are locations likely on the surface of the die, where the *accumulated distance* of the probe to the power consuming elements of one multi-bit register is shorter than the distance to the elements of another multi-bit register. At those locations it is possible to distinguish, which of the registers has been used based on different signal strengths. Hence, cryptographic algorithms which use their registers differently, depending on the value of the secret, are prone to location-dependent side-channel leakage because the recovery of the fact which registers are used leaks information about the secret.

*Localized EM analysis* includes attacks which are based on exploiting location-dependent information leakage. Many concepts which exploit data- or operation-dependent side-channel leakage such as SPA attacks in general, side-channel-based collision attacks [SWP03, SLFP04], template attacks [CRR03], correlation-based attacks and DPA attacks can be adapted to exploit location-dependent side-channel leakage instead of data-dependent leakage.

In the conventional case of exploiting data-dependent leakage, electronic noise and algorithmic noise from other parts of the circuit present as an unfavorable influence and reduce the signal-to-noise ratio of the data-dependent leakage. These noise components can be reduced through averaging multiple

traces with the *same* processed data. When exploiting location-dependent leakage, *all* data-dependent leakage signals present as an unfavorable noise influence and could be reduced through averaging over *different* processed data.

## 5.3 Attacking Binary Exponentiations

**Algorithm 2** Main loop of an abstract pseudo-algorithm. Computation sequence and timing are uniform while register usage depends on secret $d$.

**Input:** Secret $d = d_D d_{D-1} ... d_2 d_1$ with $d_i \in \{0, 1\}$

1: **for** $i = D$ downto 1 **do** ▷ Main loop of the algorithm
2: **if** $d_i = 1$ **then** ▷ loop iterations
3: $c \leftarrow a$
4: $c \leftarrow c^2$
5: $a \leftarrow c$
6: **else**
7: $c \leftarrow b$
8: $c \leftarrow c^2$
9: $b \leftarrow c$
10: **end if**
11: **end for**

Exponentiation algorithms are used for modular exponentiations in multiplicative groups in RSA and for Elliptic Curve Scalar Multiplications (ECSMs) on additive group structures. For more information about this duality see Sect. 4.1. The double-and-add-always algorithm (ECC), the square-and-multiply-always algorithm (RSA) as well as the Montgomery ladder algorithm (RSA and ECC) are examples for binary versions of this kind. Binary exponentiation algorithms typically consist of a main loop and process only one secret bit in each loop iteration. Secure implementations of such algorithms typically contain uniform operation sequences in each loop iteration, which are independent of the secret, as a countermeasure against simple side-channel analyses. However, the operations are performed on a different set of registers depending on the value of the currently processed secret bit. Hence, while their constant-time processing sequence protects them against other attacks, they are a perfect target for side-channel attacks based on *localized EM analysis* because the use of registers depends on the processed secret. An adversary can use localized EM analysis to detect the usage sequence of those registers to derive the secret.

Algorithm 2 presents an abstract pseudo-example showing the relevant properties of such algorithms. The depicted pseudo-algorithm has a uniform operation sequence and contains two operations in the loop iteration which are prone to location-dependent leakage because the register usage depends on the currently processed secret bit. In Lines 3 and 7, either register $a$ or $b$ are read depending on the secret bit's value. In Lines 5 and 9, a result is either written to register $a$ or $b$. As described in Sect. 5.2, the registers are placed on an integrated circuit and an adversary can, under certain circumstances, use location-dependent leakage to detect which of two registers is used in every iteration to recover the secret. The usage of the register $c$ is independent of the secret bit and therefore not relevant for an adversary.

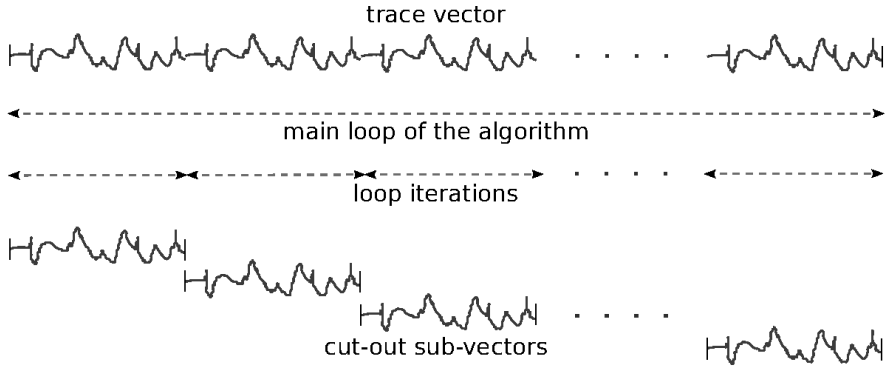



Figure 5.2: Segmentation of trace vector $\mathbf{t}$ into sub-vectors $\mathbf{t}_i$

Figure 5.2 depicts a simplified example of a recorded EM trace vector $\mathbf{t} = (t_1, \ldots, t_T)$ with $T$ samples. It contains the main loop part of a binary exponentiation algorithm (cf. Alg. 2). This trace vector is split into sub-vectors each containing one loop iteration, $\mathbf{t}_i = (t_{(1+(i-1)\frac{T}{D})}, \ldots, t_{(i\frac{T}{D})})$, $1 \leq i \leq D$ and $D$ the number of loop iterations. This segmentation is usually derived from visual inspection but can for instance be refined through cross-correlation of sub-vectors with the trace. All sub-vectors $\mathbf{t}_i$ are samples of the same operation sequence while processing different secret bits $d_i$. Thus, the values within the different sub-vectors $\mathbf{t}_i$ belong to the same operations from the loop iterations (cf. Alg. 2).

The measured power consumption of digital hardware processing uniformly distributed data consists of operation- and data-dependent parts and electronic noise and is typically distributed according to a Gaussian function. In certain operations, thus, at certain relative cycles, or operations within the sub-vectors, two different registers are used. This has been described for the abstract pseudo-algorithm depicted in Alg. 2. Since such different registers are placed at different locations on an integrated circuit, power is

also consumed at different locations on the circuit. Each location leads to a Gaussian distribution of the consumption. The goal of an adversary is to distinguish the distributions and to partition the sub-vectors $\mathbf{t}_i$ into two sub-sets. One set contains the sub-vectors where one register was used, the other one the sub-vectors where the other was used. This equals recovering the secret. In order to do this, an adversary must find similarities among the sub-vectors $\mathbf{t}_i$ which are due to the location-dependence.

### 5.3.1 Exploiting Location-Based Leakage

This location-dependent leakage can be exploited using different side-channel attacks. One likely class of side-channel attacks are so-called side-channel-based collision attacks [SWP03, SLFP04]. These can be used to find similar sub-vectors $\mathbf{t}_i$ and classify them into two groups, hence, recovering the secret. Collisions can for instance be detected through least-square tests [SLFP04] or correlation [MME10, WvWM11]. A collision attack can be performed using a single recorded trace, where the secret exponent or scalar is processed. Data-dependent influence of the power consumption might make correct classifications more difficult. If the cryptographic protocol allows multiple executions with a constant secret and different processed data, those can be averaged to reduce this.

I describe an important improvement to those methods in Chap. 6. The presented approach uses unsupervised cluster classification algorithms to partition the sub-vectors into two sets.

Another class of side-channel attacks which can be used to exploit location dependent leakage are template attacks. Template attacks require a profiling phase using a known exponent or scalar. Fortunately, when exploiting location-based information, a public exponentiation can be used for profiling even if it uses a different base (e.g., base point in the ECSM). This is different to the case of exploiting data-dependent leakage where templates are built to characterize certain intermediate data values which requires using the same base-point for the ECSM during profiling. Such template attacks on binary exponentiation algorithms [CRR03, MO09] require profiling with a chosen exponent or scalar and multiple templates. In case of exploiting location-based information, only two templates are required to classify all sub-vectors $\mathbf{t}_i$ into two groups.

The sub-vectors from a profiling trace are grouped according to the bit values of the known exponent and the two groups' mean vectors and covariance matrices are used as templates. Since different data is processed in each sub-vector, data-dependencies, or switching noise, must be modeled in this way. In the attack, the sub-vectors $\mathbf{t}_i$ of a recorded trace with a

secret exponent or scalar are matched against the templates to recover the exponent.

### 5.3.2 Finding Locations

If an adversary can observe executions with a known secret scalar, he can employ a difference-of-means test to find eligible measurement positions on the surface of an integrated circuit die for an attack. At eligible locations, e.g., one of two registers of the algorithm is physically closer to the probe than the other. Therefore, when location-based information is leaked, the sample values are distributed according to a Gaussian mixture of e.g. two superposed normal distributions instead of one because the power is consumed at two different locations.

This test is performed for every measurement location of the integrated circuit to find location-based leakage. As already stated, the secret exponent $d$ is *known* during this test and I describe the case of a binary exponentiation.

The sub-vectors $\mathbf{t}_i$ are divided into two sets according to the value of the corresponding bit $d_i$ which is processed in the respective loop iteration $i$. The first set $T_0$ contains sub-vectors where the value of the corresponding bit $d_i$ is zero, thus, $T_0 = \{\mathbf{t}_i | 1 \leq i \leq D, d_i = 0\}$. The second set $T_1$ contains sub-vectors where the value of the corresponding bit $d_i$ is one, thus, $T_1 = \{\mathbf{t}_i | 1 \leq i \leq D, d_i = 1\}$. The cardinality of the two sets $T_0$ and $T_1$ of sub-vectors $\mathbf{t}_i$ is roughly equal because the bits' values are distributed uniformly, thus, $p(d_i = 1) = 0.5$. The mean vectors $\overline{\mathbf{m}}_0$ and $\overline{\mathbf{m}}_1$ of the two sets $T_0$ and $T_1$ of sub-vectors $\mathbf{t}_i$ are estimated using the averages as shown in Eq.5.1 with $n_1$ and $n_0$ the number of sub-vectors in the sets as described in Eq. 5.2. The average vectors $\overline{\mathbf{m}}_0$ and $\overline{\mathbf{m}}_1$ contain $\frac{T}{D}$ measurement values just like the sub-vectors.

$$\overline{\mathbf{m}}_0 = \frac{1}{n_0} \sum_{i=1}^{D} \mathbf{t}_i \cdot (1 - d_i)\,,\ \overline{\mathbf{m}}_1 = \frac{1}{n_1} \sum_{i=1}^{D} \mathbf{t}_i \cdot d_i \tag{5.1}$$

$$n_1 = \sum_{i=1}^{D} d_i\,,\ n_0 = \sum_{i=1}^{D} (1 - d_i) \tag{5.2}$$

The null-hypothesis $H_0$ assumes that the two means vectors are different at some index, or cycle. The alternative hypothesis $H_1$ states that both means are equal.

$$H_0 : \overline{\mathbf{m}}_0 - \overline{\mathbf{m}}_1 \neq 0\,,\ H_1 : \overline{\mathbf{m}}_0 - \overline{\mathbf{m}}_1 = 0. \tag{5.3}$$

A confidence interval is used to test the null hypothesis $H_0$. As described by Mangard et al. [MOP07, p. 94], the confidence interval of the difference-of-means $\overline{\mathbf{m}}_0 - \overline{\mathbf{m}}_1$ is shown in Eq. 5.4.

$$[\overline{\mathbf{m}}_0 - \overline{\mathbf{m}}_1 - \mathbf{s_{0-1}} \cdot z_{1-\alpha/2}\,,\ \overline{\mathbf{m}}_0 - \overline{\mathbf{m}}_1 + \mathbf{s_{0-1}} \cdot z_{1-\alpha/2}]\,. \tag{5.4}$$

The quantile $z_{1-\alpha/2}$ is derived from the chosen confidence level [MOP07, p. 88]. For example, a confidence level of 99.9 % makes $\alpha = 0.1\,\%$ and results in $z_{1-\alpha/2} = 3.29$. The empirical standard deviations $\mathbf{s_0}$ and $\mathbf{s_1}$ of the two sets of vector are computed as shown in Eq. 5.5.

$$\mathbf{s_0} = \sqrt{\frac{1}{n_0 - 1}\sum_{i=1}^{D}(\mathbf{t}_i - \overline{\mathbf{m}}_0)^2 \cdot (1 - d_i)}\,,\ \mathbf{s_1} = \sqrt{\frac{1}{n_1 - 1}\sum_{i=1}^{D}(\mathbf{t}_i - \overline{\mathbf{m}}_1)^2 \cdot d_i} \tag{5.5}$$

The empirical standard deviation $\mathbf{s_{0-1}}$ of the distribution of the difference-of-means is shown in Eq. 5.6 as described in Mangard et al. [MOP07, p. 94].

$$\mathbf{s_{0-1}} = \sqrt{\frac{(n_0 - 1) \cdot \mathbf{s_0}^2 + (n_1 - 1) \cdot \mathbf{s_1}^2}{n_0 + n_1 - 2}} \cdot \sqrt{\frac{n_0 + n_1}{n_0 \cdot n_1}}\,. \tag{5.6}$$

At cycles where zero is *not included in this confidence interval (Eq. 5.4), location-based information about the secret is leaked* at the respective location on the surface of the die. Designers can generally use traces with a known exponent or scalar to perform difference-of-means tests to look for leaks of location-dependent information.

If an adversary may not perform profiling to find eligible locations, a chi-square test or simple search for high variances may lead to good results because the variance of samples will generally be higher at such locations.

## 5.4 Case Study

In this case study, I performed a reduced template attack [MOP07] exploiting location-dependent leakage to break a protected FPGA implementation of the Elliptic Curve Scalar Multiplication (ECSM).

### 5.4.1 Design-under-Attack and Measurement Setup

The implementation of the ECSM is described in Chap. 4. It employs a constant runtime Montgomery ladder exponentiation and is secure against conventional simple power analysis and timing analysis.

The implemented Montgomery ladder exponentiation algorithm is depicted in Alg. 1. From the description, it can be observed, that the registers $T, x_P$ and $b$ are used equally, independent of the scalar bit values. However, the working registers $X_0, Z_0$ and $X_1, Z_1$ are used differently, depending on the scalar bits $d_i$. This is similar to the abstract pseudo algorithm Alg. 2 from the previous section and makes the implementation prone to localized EM analysis.

The registers' design is completely equal on the RTL level. No further manual effort or constraints were employed during FPGA synthesis and placement on the die. Therefore this case study well-represents realistic circumstances.

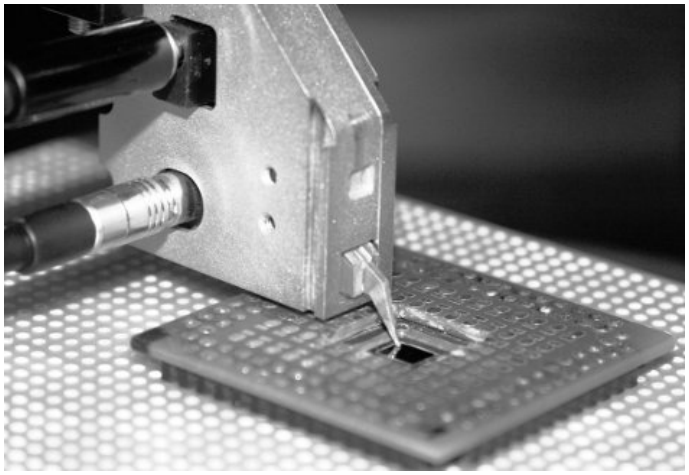


Figure 5.3: Near-field probe close to the surface of the die

I employed a *Xilinx Spartan-3 (XC3S200)* FPGA device in a $VQ100$ package for design configuration. I depackaged the FPGA from the backside as suggested by Skorobogatov et al. [Sko10]. Backside depackaging is less complex for an adversary than frontside depackaging for smartcards and many plastic packages since it can be achieved purely mechanical instead of using chemicals. The plastic packaging is ground away down to the copper plate of the package lead-frame using a drill. Then, the copper plate is removed using a carpet cutter and remaining glue scraped away using a scalpel.

I performed localized electromagnetic field measurements from the backside. Results presented in Chap. 3 indicated that frontside measurements lead to even better signal-to-noise ratios. The attack presented in this chapter works nonetheless which underlines its capabilities.

The measurement setup with the measurement probe and FPGA die is

depicted in Fig. 5.3. It is similar to the setup from Chap. 3. The same magnetic near-field EM probe with a horizontal coil, 100 µm resolution, and a $2 * 30$ dB amplification as described in Chap. 3 was used at a close distance to the surface of the die.

The near-field probe was moved over the surface of the $\approx 5000 * 4000$ µm die by an x-y-table with a step size of 50 µm. At every location, one trace was recorded at a sampling rate of 5 GS/s. Through synchronization of the oscilloscope and the function generator, frequency jitter and drift in the measurements is prevented.

I compressed the trace to one sample per clock cycle, extracting the difference of the maximum peak to the minimum peak EM value in every cycle to reduce the amount of data and computation complexity. In this way, 250 samples per clock cycle, using a 20 MHz clock frequency and 5 GS/s sample rate, are reduced to a single value per cycle. The results from Chap. 3 showed that trace compression reduces the signal-to-noise ratios of leakage signals.

Because of the disadvantageous measurement from the backside and employment of trace compression, the presented practical study has a worst-case character. The successful demonstration of the attack nonetheless emphasizes its capabilities.

### 5.4.2 Template Attack

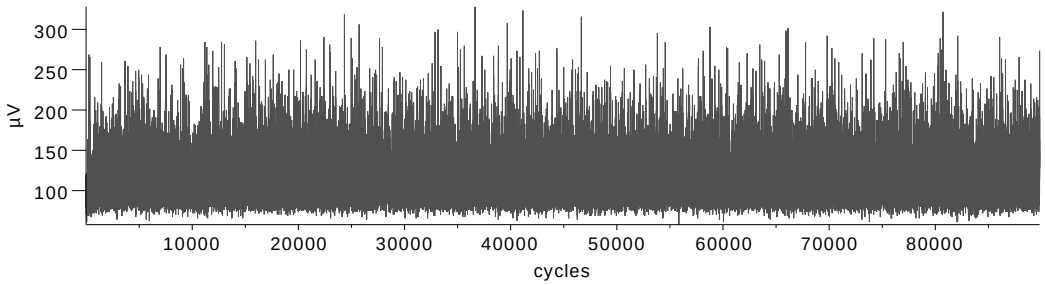
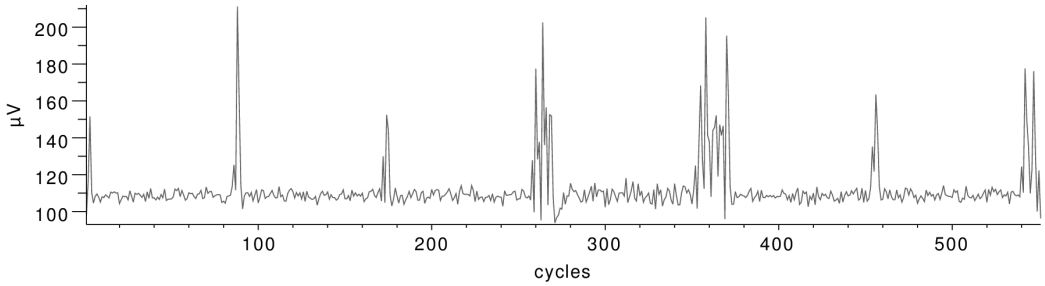
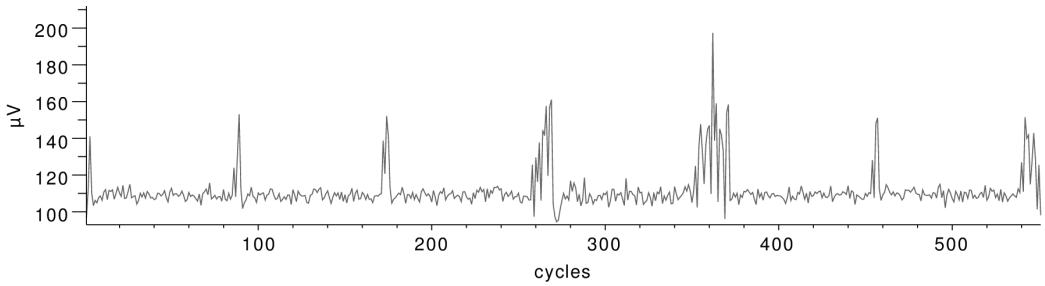



Figure 5.4: Recorded EM trace $\mathbf{t}$ at location $(x, y) = (37, 42)$

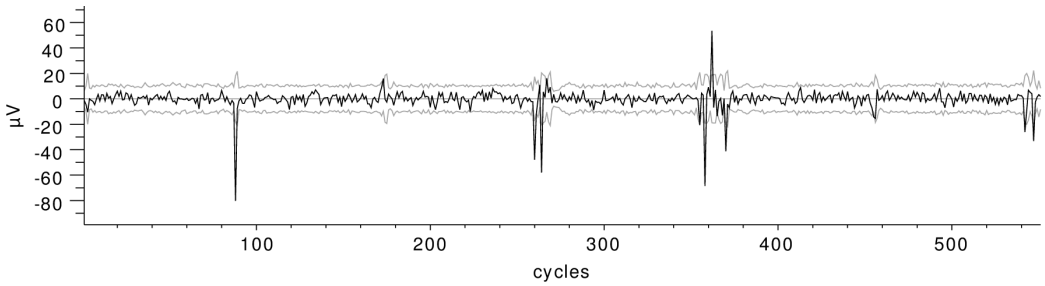
In this case study, I assume that an adversary can observe public operations using a public and known scalar on the device he is attacking. This knowledge is used to find eligible locations for the attack and to build templates. In ECC, the public operation can for instance be a signature verification using a known scalar for the ECSM.




(a) Estimated mean $\overline{\mathbf{m}}_0$ of sub-set where $d_i$ is zero.




(b) Estimated mean $\overline{\mathbf{m}}_1$ of sub-set where $d_i$ is one.

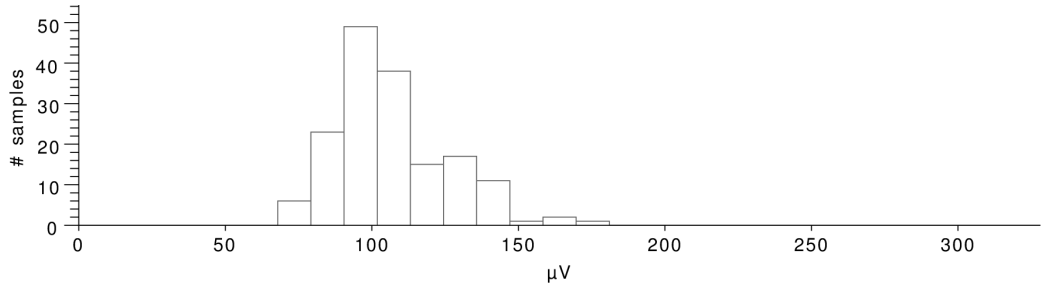



(c) Difference-of-means $\overline{\mathbf{m}}_0 - \overline{\mathbf{m}}_1$ as a black graph including a confidence interval depicted by two grey graphs around a grey zero line.

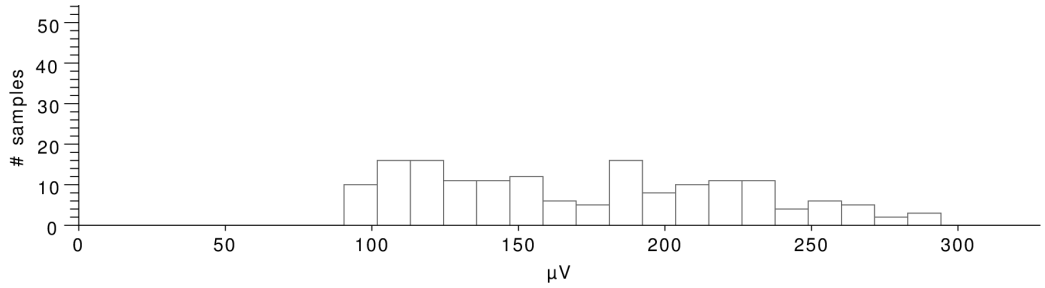
Figure 5.5: Sub-vector means and difference-of-means at location $(x, y) = (37, 42)$

### Profiling

Figure 5.4 depicts an EM trace vector as an example which was recorded using the described measurement setup and a known scalar. The depicted trace was recorded at the map position $(x, y) = (37, 42)$ (cf. map in Fig. 5.7) and serves as an example to illustrate the explanations from the previous sections. Every point represents one clock cycle and contains the value which was derived from the peak to peak pre-processing in µV. The trace includes $\approx 90\,k$ cycles, covering the main loop of Alg. 1, Lines 3 to 8.




(a) Regular data-dependent distribution of samples in cycle 110.




(b) Location-influenced and data-dependent distribution of samples in cycle 88.

Figure 5.6: Histograms of the samples from different sub-vectors of one trace at location $(x, y) = (37, 42)$

The recorded trace vector $\mathbf{t}$ was split into 163 sub-vectors $\mathbf{t}_i$ corresponding to 163 scalar bits as described in Sect. 5.3. All sub-vectors correspond to a uniform operation sequence which is depicted in Alg. 1, Lines 3 to 8, and include $\approx$ 550 measurement values each. The register access depends on the value of the corresponding scalar bit which will be exploited during the attack.

As described in Sect. 5.3.2, the sub-vectors were assigned to two sets according to the known scalar bits. Figure 5.5(a) and Fig. 5.5(b) depict the mean vectors $\overline{\mathbf{m}}_0$ and $\overline{\mathbf{m}}_1$ of those two sets. Figure 5.5(c) shows the difference-of-means as a black graph. The figure also depicts two grey lines corresponding to the confidence interval around zero at a confidence level of 99.9 %. Cycles, where the black difference-of-means graph exceeds the zero-region, which is confined through the grey graphs, are clearly visible, showing that this design leaks location-dependent information.

I used the mean vectors $\overline{\mathbf{m}}_0$ and $\overline{\mathbf{m}}_1$ as reduced templates for the attack and employed a least-square matching. This means that the covariance matrices are disregarded from the template matching step to simplify the computational complexity. This corresponds to assuming a uniform variance and is justified as long as it leads to reasonable results.

To visualize the location-dependent leakage in the measurements, two histograms are provided in Fig. 5.6. The same trace which was analysed in Fig. 5.5, is used. In Fig. 5.5(c) it could be observed, that there are cycles with a significant difference-of-means. Those are the cycles which contain the location-dependent leakage. To visualize this I show the histograms in two cycles. First cycle 88 with a high difference-of-means, and second cycle 110 with no observable difference-of-means as can be seen in Fig. 5.5(c). The histogram in Fig. 5.6(a) depicts the distribution of values from the 163 sub-vectors in cycle number 110. The histogram in Fig. 5.6(b) depicts the distribution in cycle number 88. The distribution of samples from different sub-vectors is significantly broader in cycle 88. This an indicator proving the fact, that two overlayed normal distributions are observed corresponding to two locations which are active alternately. The histogram for cycle 110 is narrower, indicating that only one Gaussian distribution is observed. While the two distributions for cycle 88 are not clearly distinct, the histograms clearly show the difference which is due to the location-dependent influence.

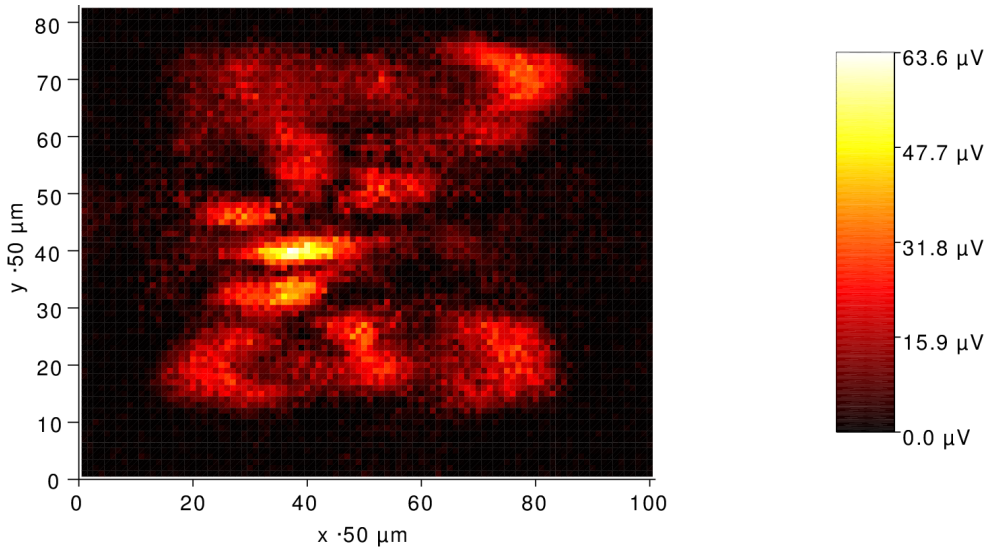



Figure 5.7: Greatest *absolute* difference-of-means for all locations

I performed the described difference-of-means test on every location on the surface of the die using a single recorded trace at every location. Figure 5.7 presents an $(x, y)$ map of the greatest *absolute* difference-of-means in µV for each location on the die. The maximum of about 63 µV is clearly significant compared to the amplitudes from Fig. 5.4. *A pattern of locations with high information leakage* can be observed. For the actual attack, the location with the greatest difference-of-means, $(x, y) = (37, 42)$, was used.

The same location was used for the previous Fig. 5.4, 5.5, and 5.6.

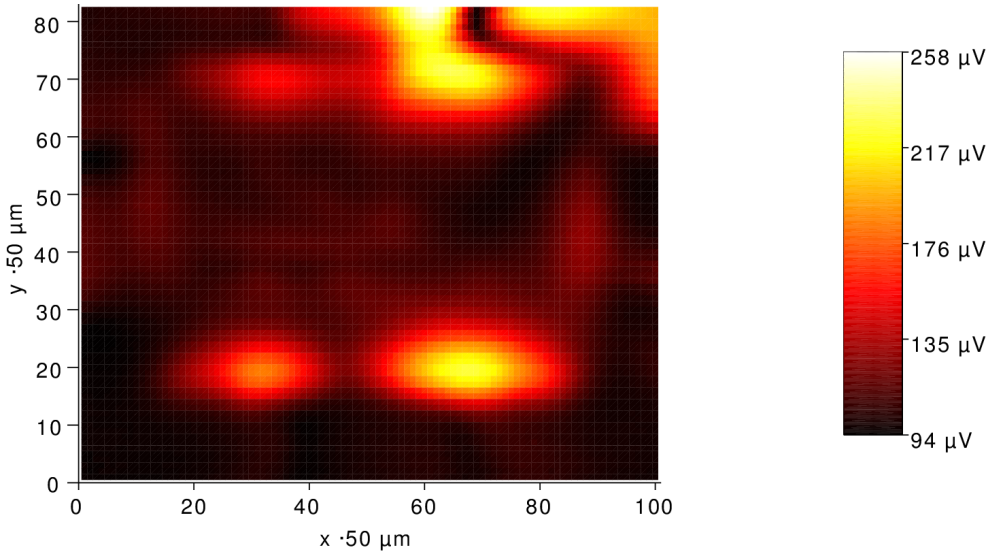



Figure 5.8: Average amplitude for all locations

Figure 5.8 depicts a map with average EM amplitudes of the recorded traces. It can be observed that regions with *high EM amplitudes are not congruent to regions leaking most location-based information.* This is contrary to the perception in many contributions where regions with high signal strengths are automatically regarded as suitable for side-channel analysis.

Figure 5.9 depicts an overlay of the map of location-dependent information leakage over a die photograph from the frontside. It can be observed how the regions of high information leakage are spread over the middle area of the FPGA where the configurable logic is located.

### Significance of the Location Dependence

This section shall provide an additional illustrative example for the significance of the location dependence. The previous Fig. 5.7 presents an $(x, y)$ map of the greatest *absolute* difference-of-means for each location on the die. At every measurement position, the greatest absolute value is chosen from a vector of difference-of-means values $\mathbf{m}_0 - \mathbf{m}_1$. Contrarily, in the following map in Fig. 5.10, I depict the signed difference-of-means for a specific vector index. Similar to when showing the histogram in Fig. 5.6(b), I chose cycle index 88 within the trace sub-vectors because it exhibits a significant information leakage. This can be observed in Fig. 5.5(c). The significant

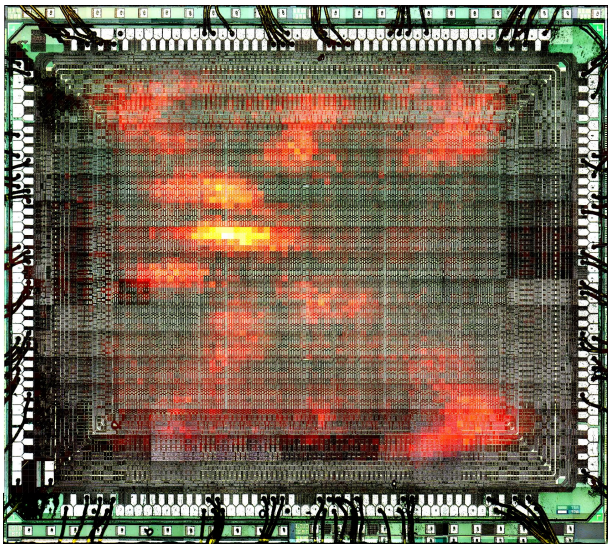


Figure 5.9: Overlay of the location-based leakage over a die photo of the *Xilinx Spartan-3 (XC3S200)* FPGA from the frontside

location-based information leakage stems from the alleged store or read operation which is performed in this cycle.

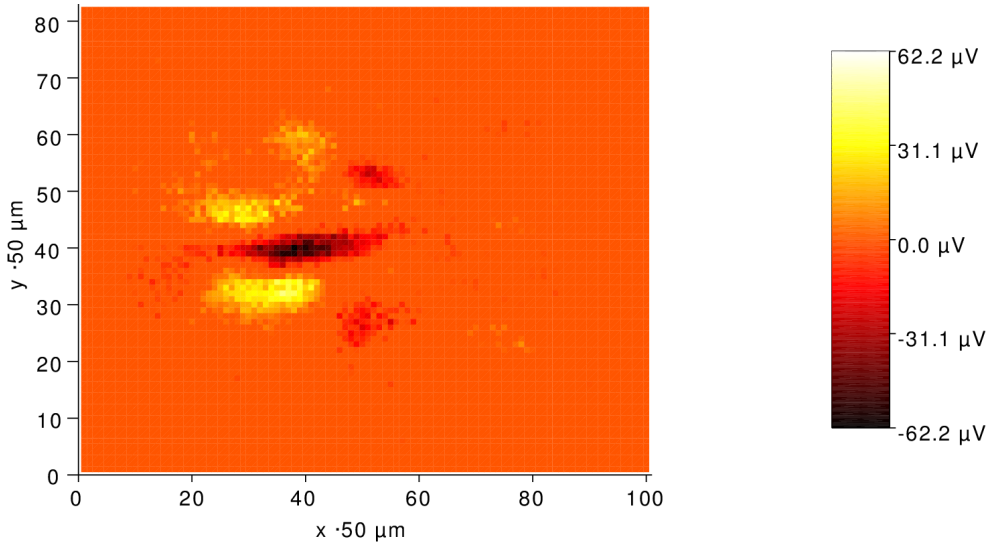



Figure 5.10: *Signed* difference-of-means for cycle 88 at all locations

Figure 5.10 shows a map where each value corresponds to the *signed* difference-of-means when looking at this single operation. Most of the map is colored in orange indicating a 0 µV difference-of-means, thus, no information

leakage. However, when the probe is moved over the die, it gets closer to cells belonging to one of two registers. Note that multi-bit registers in synthesized digital hardware get distributed over the circuit area. Therefore, it is not expected that there are two confined register regions, but rather that the cells belonging to the registers are distributed interspersed. In regions, which are colored in yellow to white, cells from the first register lead to higher EM values than the other one. In other regions, colored in red to black, the probe gets closer to cells belonging to the other register. In those regions, the other register leads to higher EM values and, therefore, the difference-of-means changes sign. This shows that there are significant location-dependences of the information leakage. *The oppositely signed differences would likely cancel out each other when analyzing the power consumption of the entire device which emphasizes the relevance of localized measurements.*

Generally, it is hard to balance the physical implementation of registers to achieve identical power consumptions. Such differences in the registers' power consumption might be partly detectable in overall power consumption measurements. However, such differences are usually small and hard to exploit in single-observation attacks. *The localized measurement provides a significant improvement in this respect, allowing single-observation attacks.*

### Attack

Due to reasons explained in Sect. 4.2, an adversary only has a single trace to recover the secret scalar. Retrieving the scalar is equivalent to a total break of the system since the secret key can be computed easily from it.

The segmented sub-vectors are compared to both templates using a least-square distance test. *In this practical experiment* 161 *of* 163 *secret scalar bits could be classified correctly, leaving* 2 *erroneous bits.* This *proves that location-dependent leakage can be successfully exploited in practice.* A measurement setup with frontside instead of backside measurements and no trace compression would most likely further improve the attack (cf. Chap. 3).

The remaining erroneous bits are most likely due to electric and data-dependent noise. The error probability is highest for those recovered bits, where the classification is least decisive. In this manner, an adversary incrementally brute-forces bits where the difference between the two least-square matching likelihoods is smallest until the recovered secret is correct. In this practical experiment, the correct scalar could be recovered after brute-forcing 14 bits at maximum. Hence, even if the classification does not provide a full recovery of the scalar, it *reduces the search space to a practical level.*

I tested recording multiple traces with a constant scalar and different input data to reduce the data-dependent influence through averaging. The

success rate increased to 100 % using only 3 averaged traces. Even though this is not realistic for many adversarial scenarios as described in Sect. 4.2, this result is significant.

## 5.5 Countermeasures

Countermeasures against power or EM analyses such as e.g., employing the Montgomery ladder, randomization of projective coordinates [Cor99], base point blinding or exponent blinding *do not prevent location-dependent information leakage*. Random delay insertion and similar countermeasures, however, complicate *localized EM analysis*.

Exponent blinding only prevents template attacks based on location-dependent leakage if it is also employed for the public operation. However, this introduces a significant computational overhead and does for example not prevent collision attacks based on location-dependent leakage. Another countermeasure against template attacks is to not allow the public as well as private operation be performed on the same device.

*Randomizing the assignment of registers to physical locations* on an integrated circuit *generally prevents location-dependent information leakage because the relation between location-based information and the secret is eliminated.* This prevents the only remaining single-observation side-channel leakage of exponentiation algorithms such as Montgomery ladder exponentiation. The concept of randomized locations was presented by Itoh et al. [IIT03b] in a different context.

**Algorithm 3** Countermeasure for Alg. 1

9: $r \leftarrow random \in [0, 1]$
10: $c \leftarrow swap\_state \oplus r$
11: $T \leftarrow X_0 + X_1$
12: $X_0 \leftarrow T - X_{1-c}$, $X_1 \leftarrow T - X_c$ ▷ *swap* $X_0$ *and* $X_1$ *if* $c = 1$
13: $T \leftarrow Z_0 + Z_1$
14: $Z_0 \leftarrow T - Z_{1-c}$, $Z_1 \leftarrow T - Z_c$ ▷ *swap* $Z_0$ *and* $Z_1$ *if* $c = 1$
15: $swap\_state \leftarrow r$

Accordingly, I introduce a countermeasure for implementations of the López-Dahab Montgomery ECSM which is depicted in Alg. 3. The additional operations are integrated into the ECSM Alg. 1 within the loop beyond Line 8 and swap the register contents of $X_0, X_1$ and $Z_0, Z_1$ according to a chosen random number $r$. The value of the scalar bit $d_i$, thus, the addressing of the registers within the loop, is inverted according to whether the registers

are swapped (*swap_state*). This *swap_state* is initialized with 0 before the computation. It requires 2 field additions and 4 field subtractions, which are equal to additions in $GF(2^m)$, in every loop iteration which resulted in a computational overhead of about 4 % for the implementation described in Chap. 4.

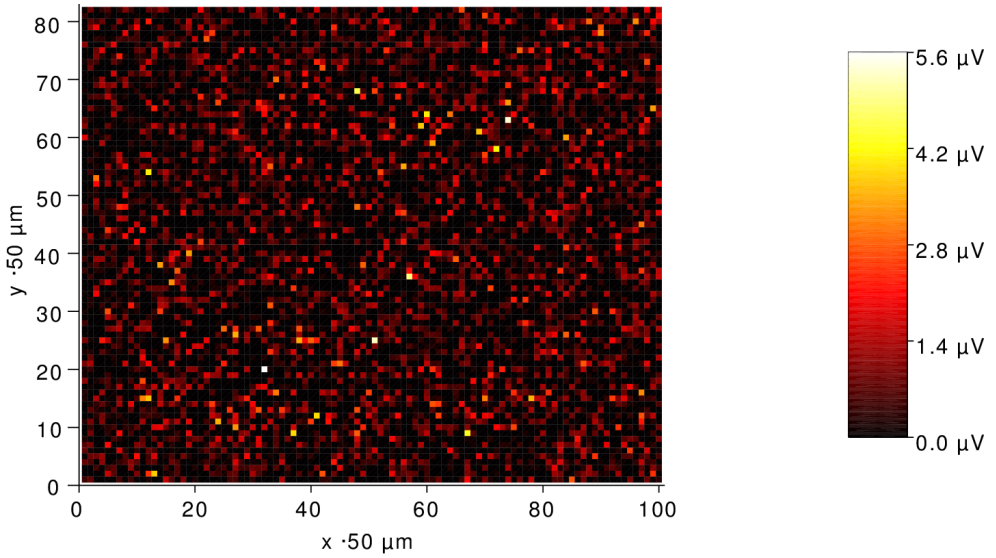



Figure 5.11: Greatest absolute difference-of-means when employing the countermeasure

I repeated the known scalar difference-of-means test. Figure 5.11 depicts a map of the resulting greatest absolute difference-of-mean values. A comparison to the results in Fig. 5.7 from the unprotected implementation confirms that the random swapping of register locations effectively destroys the relation between location-based information and the secret.

The countermeasure is uniform in its operation sequence, hence, not detectable through simple analyses. However, during the steps listed in Alg. 3, the same working registers which are attacked in the original attack, are used in a way which is depending on the *swap_state*. In this way, a second order attack is feasible where an adversary recovers the information whether the registers are swapped or not by separately attacking the operations from Alg. 3. Therefore, the presented countermeasure provides security against first order attacks. It can be assumed that the success probability would be significantly lower for a second order attack since twice as many classifications have to be accomplished.

Poucheret et al- [PBB+10] propose to create multiple equivalent datapaths for the computation of intermediate values in implementations of sym-

metric cryptographic algorithms which can be exchanged during operation. This is similar to the above described approach, besides the fact, that the above described approach does not increase datapath complexity and represents a solution on the algorithmic level.

A general countermeasures against localized EM analysis is interleaved placement [HdlTR12] of logic circuitry on a very low abstraction level of the design flow. However, randomization on an algorithmic level renders it unnecessary to interleave or carefully balance the physical implementation of those registers.

## 5.6 Summary

I demonstrate how location-dependent leakage instead of data-dependent leakage of cryptographic implementations can be used for side-channel attacks on exponentiation algorithms. This requires the feasibility of localized measurements of electromagnetic fields of integrated circuits which has been described in the previous Chap. 3. I demonstrated a successful template attack exploiting location-dependent information leakage on an FPGA implementation of the elliptic curve implementation which was described in Chap. 4. As a conclusion I promote that in all implementations of affected cryptographic algorithms, the assignment of concerned registers to physical locations should be repeatedly randomized by swapping their locations at random times during execution.

# Chapter 6

# Using Unsupervised Clustering for Non-Profiled Single Execution Attacks on Exponentiation Algorithms

I provided an extensive practical analysis about the feasibility of localized measurements of electromagnetic fields in Chap. 3. In Chap. 5, I described, how such spatially restricted measurements can be used to exploit location-based information leakage in dedicated side-channel attacks. Using an FPGA-based digital hardware implementation of an elliptic curve scalar multiplication, I demonstrated how a template attack led to a successful recovery of the scalar using only a single measurement trace. This required profiling using a known scalar to find an eligible measurement location and to build templates.

In this chapter I describe, how unsupervised clustering algorithms can be used to exploit such location-based information leakage of exponentiation algorithms in a *non-profiled* attack. Contrary to the profiled template attack from Chap. 5, this single execution attack does not require profiling to build templates. Furthermore, the concept of using such unsupervised cluster classification algorithms also applies to any single-execution side-channel leakage of exponentiation algorithms. Parts of this chapter are planned for publication [HIM+13].

When attacking exponentiation algorithms used in asymmetric cryptography, adversaries can only exploit single-executions to recover a secret exponent because the exponent is different for every execution. To prevent SPA and timing attacks [Koc96], protected implementations of such exponentiation algorithms either perform equal operations for different exponent

bits or different operations, e.g., square and multiply, are implemented so that they appear equal. Algorithms like the square-and-multiply(-always), double-and-add(-always) or the Montgomery ladder algorithm are examples for algorithms with constant operation sequences. However, several contributions indicate, that a certain level of side-channel leakage about independently processed bits or digits during a single exponentiation cannot be prevented [Bau12, PTBM12, SI11, IIT03a, Wal01], i.e., single-execution leakage.

This may for instance be location-based leakage such as I presented in Chap. 5, address bit leakage [IIT03a], or operation-dependent leakage, e.g., when square and multiply operations can be distinguished [Bau12, PTBM12].

In this chapter, I propose a new class of algorithms to attack exponentiation algorithms by exploiting single-execution leakages of independently processed bits or digits during exponentiations. Specifically, I suggest to apply unsupervised cluster classification algorithms [DHS01]. Unsupervised clustering is generally useful in side-channel analysis when either supervision information is not available from prior profiling, or an exhaustive partitioning is computationally infeasible.

I demonstrate the proposed attack on the FPGA-based implementation of an elliptic curve scalar multiplication which is described in Chap. 4 and has also been used in the previous Chap. 5. As presented in the previous Chap. 5, this implementation exhibits location-based leakage. A complete recovery of the secret scalar was accomplished using the presented method and one single measurement of the localized EM field at a position which has been found through prior analysis of the leakage of the device.

The success probability of this attack generally depends on the Signal-to-Noise Ratio (SNR) of the exploited single-execution leakage. If the SNR in one single measurement is insufficient, it can be improved by combining the information from simultaneous measurements. Following this idea, I present how the attack described in this chapter can be improved in the subsequent Chap. 7.

I start the chapter with related work in Sect. 6.1. The main idea is described in Sect. 6.2 and in the subsequent Sect. 6.3, I provide practical results confirming the findings. In a short Sect. 6.4, I present further side-channel attacks, which could benefit from the application of unsupervised clustering algorithms. A summary of this chapter is provided in Sect. 6.5.

## 6.1 Related Work

This related work section covers two aspects. First, I list relevant contributions from the area of single-execution side-channel attacks against exponentiation algorithms. Second, I list other articles which mentioned cluster analysis in the context of side-channel analysis.

### Single-Execution SCAs against Exponentiations

I start with related work regarding attacks on exponentiations using single executions.

The most recent attack against exponentiation algorithms was presented by Schindler and Itoh [SI11]. While the attack targets blinded exponentiations it still uses multiple executions to recover the secret. In its basic version, executions with equal blinding factors are found and exploited. A general single-execution leakage of each exponent bit is assumed, however, its exploitation not discussed in details.

My contribution presents an extension rather than an alternative to Schindler and Itoh's attack since I propose cluster classification algorithms as a measure to improve the exploitation of such single-execution leakages. If the exponent can be recovered from a single-execution, or by means of combining simultaneous measurements of a single execution as proposed in Chap. 7, the method of Schindler and Itoh is not necessary.

Schindler and Itoh propose a grouping of measurements with equal blinding factors. This could be improved through using clustering algorithms.

Walter [Wal01] describes a single-execution side-channel attack on $m$-ary ($m > 2$) sliding window exponentiation algorithms. He recognizes the repeated use of the same pre-computed multiplier values depending on exponent digits in different segments of digit-wise multiplications. He uses his own individual algorithm to partition segments into buckets according to their pair-wise similarity.

While the core idea of this contribution is similar to the one described by Walter, I propose to use well-established cluster classification algorithms for unsupervised classification which are known to be optimal under the respective assumptions about the distribution of samples. Such algorithms can be applied to a wide range of exponentiation algorithms and are able to exploit arbitrary multivariate single-execution leakages of independent exponent bits or digits.

Messerges et al. [MDS99a] first mention the possibility to match segments of an exponentiation measurement by using cross-correlation, however, found that this was not leading to meaningful results. They proceeded to describe

differential methods to recover exponents which include averaging and do not provide automated interpretation of matching results. However, they use multiple executions with constant exponents. Amiel et al. [AFV07] use correlation of a heuristic leakage model of fixed multiplier values to recover the exponent. However, they also use multiple traces, hence this is not relevant for the single-execution context I am targeting.

The idea of using correlation to distinguish segments of an exponentiation was later successfully used by Clavier et al. [CFG+10]. Similar to Walter, they concentrate on the leakage of a specific implementation. Using a heuristic power model, they derive hypothetical power values which they correlate with observed measurements. However, Clavier et al. do not describe an algorithm for unsupervised interpretation of the correlation results.

I argue that the method described in this chapter should provide significantly improved results since it is not restricted by the use of heuristic leakage models. Furthermore, I argue that there is generally no reason to use the correlation coefficient instead of the Euclidean distance as a measure of similarity to compare samples from the same measurement setup. The correlation coefficient must be used if the two samples which are compared only share linear dependencies while their absolute ranges are different. This is for instance the case during DPA, where values derived from an abstract leakage model are compared to actual measurements. When comparing samples from the same measurement setup, the absolute values are within the same range. Hence, more information from the values is incorporated by comparing them with, e.g., a least-square matching.

Another attack on exponentiations which uses cross-correlation was presented by Witteman et al. [WvWM11]. They present an SPA attack on the square-and-multiply-always RSA algorithm by finding consecutive operations which share the same input values which can be seen as the first application of side-channel based collision attacks to public key cryptography. The outcome of the cross-correlation-based matching is interpreted manually.

Perin et al. [PTBM12] exploit differences between square and multiply, or dummy multiply operations, and between memory accesses in exponentiation algorithms using the electromagnetic side-channel. They present the latest results in preprocessing and filtering of EM traces to increase the SNR of single-execution leakage which is due to the different operations. However, they require averaging of multiple measurements in their practical results and simply compare the segments of the exponentiation by subtraction to recover the exponent

Summarizing, the method I describe in this chapter enables exploitation of *arbitrary* single-execution leakages and employs well-researched unsupervised classification algorithms to derive the exponent *automatically*. This is

contrary to previous contributions which exploit *specific* leakages, use *unnecessary restrictions*, e.g., correlation coefficient instead of Euclidean distance as a measure for similarity, and require *supervised interpretation* of outcomes, or an individual, not well-established algorithm [Wal01].

### Use of cluster analysis in side-channel analysis

There are other published contributions which mention methods from cluster analysis. Batina et al. [BGLR09] proposed Differential Cluster Analysis (DCA) as an extension to DPA. Similar to DPA, the proposed attack uses known input values and key guesses to derive partitions of multiple observed executions. Classic DPA employs a difference-of-means test to evaluate partitions into two bins based, e.g., on a single bit of an intermediate value. In the case of DCA, a correct key is detected by using a cluster criterion to reveal a statistically significant cluster separation. Batina et al. do not discuss unsupervised cluster classification but only use cluster criteria as a statistical distinguisher. In [BHW12, MBLM12], this work is extended by considering PCA.

Lemke-Rust and Paar [LRP07] propose a profiled multi-execution attack against masked implementations using the expectation-maximization algorithm. They estimate multivariate Gaussian mixture densities of masked implementations during a profiling phase. In a profiled setting, they estimate mixture densities of clusters for known key values and unknown mask values using multiple executions. Because of the unknown input values, multiple separate distributions are expected for every known key value. The EM algorithm is used to estimate the corresponding parameters. During multiple executions of an attack, they compare the observed Gaussian mixtures from an unknown key to the Gaussian mixtures templates. The most likely model leads to the correct key.

Contrary to my proposal, this is a profiled attack with a training set for the estimation of the clusters. Thus, it is a supervised, rather than unsupervised setting.

## 6.2 Using Cluster Analysis to Attack Exponentiations

In this section, I describe how cluster analysis can be used to exploit arbitrary single-execution side-channel leakage from exponentiation algorithms.

The common property of all exponentiation algorithms, e.g., binary, $m$-ary, or sliding window exponentiations is that the computation is segmented

into parts where operations are repeated and depend only on single bits or digits of the exponent. The square-and-multiply-always exponentiation algorithm for instance consecutively either performs a square-and-multiply operation, or a square-and-dummy-multiply, depending on each bit. Those repeated parts share similarities for equal processed exponent bits or digits which are detectable through side-channels. *Side-channel information that can be collected from a single execution of the exponentiation is referred to as single-execution leakage.*

An implementation may exhibit single-execution leakage about the independently processed bits on the timing, power consumption, or electromagnetic field side-channel. I describe the use of different variables in Montgomery exponentiation algorithms depending on the processed exponent bit as a source of *location-based single-execution leakage* in Chap. 5.

Walter [Wal01] describes the use of pre-computed multiplier values in digit-wise multiplications during $m$-ary or sliding window exponentiation algorithms as a source of single-execution leakage. In case of conventional square-and-multiply exponentiation algorithms, single-execution leakage must be prevented by making the two different operations as indistinguishable as possible. However, as described by Bauer [Bau12], a certain amount of leakage in the power or time domain remains in most cases.

Most single-execution leakages can be assumed to be normally distributed and are affected by normally distributed measurement- and switching noise from not concerned circuit parts. Therefore, the adversary only has a certain amount of signal-to-noise ratio of this leakage that he can exploit. The next section will present a way to exploit such single-execution leakage.

### 6.2.1 Unsupervised Clustering

In this section, I describe the application of unsupervised clustering to exploit general single-execution leakage.

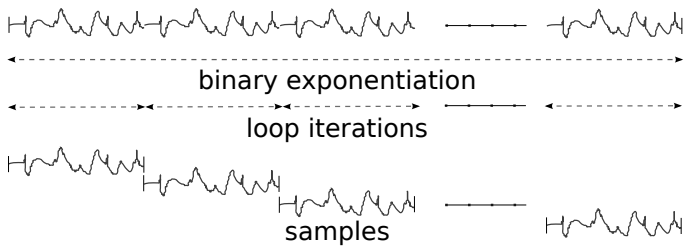



Figure 6.1: Segmenting a side-channel measurement of an exponentiation into samples

In the case of timing-safe binary exponentiation algorithms, e.g., square-and-multiply-always, double-and-add-always, and Montgomery ladder algorithms, the secret exponent is processed bitwise during a loop in segments of constant timing.

Figure 6.1 abstractly depicts a side-channel measurement of such an exponentiation. For the sake of simplicity, I assume it is a binary exponentiation algorithm. Hence, there are two different cases for each iteration during the exponentiation loop algorithm. The measurement trace vector $\vec{t} = (t_1, ..., t_l)$ contains $l$ measurement values $t$ and covers the whole exponentiation. To exploit the single-execution leakage of different secret exponent bits, the trace must be cut into samples. Hence, it is segmented into $n$ samples $\vec{t_i} = (t_{(1+(i-1)\frac{l}{n})}, ..., t_{(i\frac{l}{n})}),\ 1 \leq i \leq n$ of equal length $\frac{l}{n}$ where each sample corresponds to the side-channel leakage of one exponent bit. This segmentation can for example be derived from visual inspection.

Due to the fact, that they contain the leakage of independent *bits*, the samples $\vec{t_i}$ belong to *one of two* classes $\omega_j$ and are assumed normally distributed with $p(\vec{t_i}|\omega_j) \sim \mathcal{N}(\vec{\mu}_j, \vec{\Sigma}_j),\ j \in \{A, B\}$. When attacking $m$-ary, or sliding window exponentiation algorithms, more classes are expected.

Samples within classes $\omega_j$ cluster around mean vectors $\vec{\mu}_j$. The distance between the mean vectors $\vec{\mu}_j$ is caused by the exploited single-execution leakage. The covariance $\vec{\Sigma}_j$ within classes $\omega_j$ is due to measurement noise and switching noise from uncorrelated leakage information. This intra-class variance might for instance be due to data-dependent signal leakage, while exploiting location-based information leakage.

The distribution of samples $\vec{t_i}$ in the two classes $\omega_A$, and $\omega_B$ can be described as $p(\vec{t_i}|\omega_A) \sim \mathcal{N}(\vec{\mu}_A, \vec{\Sigma}_A)$, and $p(\vec{t_i}|\omega_B) \sim \mathcal{N}(\vec{\mu}_B, \vec{\Sigma}_B)$. This assignment to the correct classes of each sample, is unknown and the joint set of samples results in a multivariate Gaussian mixture. The correct assignment to classes of all samples equals recovering the values of the exponent bits.

It is computationally infeasible to test all possible partitions of samples for the one which extremizes a criterion which would describe the quality of cluster separation, e.g., sum of squared distances. In the case of attacking binary exponentiations, the number of possible partitions equals $2^n$ with $n$ the number of exponent bits.

Fortunately, unsupervised clustering algorithms optimize a criterion and find partitions iteratively [DHS01]. I propose unsupervised cluster classification algorithms [DHS01] such as *k-means clustering*, or *expectation-maximization clustering* to obtain this classification. A correct classification equals the recovery of the exponent because there remain only two possibilities to assign the bit values 0 and 1 as class labels to two classes.

*Hence, I propose to use clustering algorithms for a single-execution side-channel attack on exponentiation algorithms without prior profiling or heuristic leakage model.*

An unsupervised classification, or partition of the samples into sets is equivalent to estimating the parameters of the densities of the two classes. It depends on the model, hence assumed shape of the clusters, how many parameters have to be estimated and which unsupervised cluster classification algorithm is optimal. In the context of side-channel measurements, *centroid-based clustering*, or *distribution-based clustering* algorithms seem most appropriate.

Measurement noise can generally be assumed to be distributed independently in all variables with equal variance. However, this is not exactly the case for switching noise. Variables might be partly dependent because, e.g., subsequent samples in measurements of the power consumption possibly contain the same data-dependent leakage information.

Unequal variance in the samples leads to hyperellipsoidal clusters. This requires modeling using an arbitrary covariance matrix $\vec{\Sigma}_j$. This covariance matrix is only diagonal if all variables are independent $\vec{\Sigma}_j = \text{diag}(\sigma^2_{j,1}, ..., \sigma^2_{j,n})$. If both clusters are distributed identically, a single covariance matrix can be used $\vec{\Sigma}_j = \vec{\Sigma}$. In all cases which require the modeling of clusters using arbitrary covariance matrices $\vec{\Sigma}_j$, the *expectation-maximization clustering algorithm* is optimal.

If the optimal model and algorithm cannot be used due to a high computational effort or non-existent numerical convergence, simpler models and algorithms which employ a reduced parameter set can still lead to reasonable results [DHS01].

**Algorithm 4** Unsupervised k-means clustering algorithm [DHS01]

**input:** samples $\vec{t_i},\ 1 \leq i \leq n$, number of clusters $k$
**output:** cluster means $\vec{\mu}_j$ and classification $c_i \in [1..k]$

1: initialize by picking $k$ random samples $\vec{t_i}$ as $\vec{\mu}_j,\ 1 \leq j \leq k$
2: **repeat**
3: derive $c_i$ by classifying all samples $\vec{t_i}$ into $\omega_j$ from minimal Euclidean distance to $\vec{\mu}_j$
4: compute new $\vec{\mu}_j$ as mean of all samples $\vec{t_i}$ in $\omega_j$
5: **until** new $c_i$ = old $c_i\ \forall\ i$

I chose to use and discuss a simple model of cluster distributions and corresponding classification algorithm to simplify computations. Using a

simple model is valid as long as it provides reasonable results and I show in Sect. 6.3, that this leads to a successful attack in the practical evaluation. In this way, I assume that the variables within the sample vectors $\vec{t_i}$ are independent and exhibit the same variance $\sigma^2$. This leads to a simplified covariance matrix $\vec{\Sigma} = \sigma^2 \vec{I}$. Furthermore, I assume that both classes $\omega_j$ exhibit approximately the same variance leading to distributions $p(\vec{t_i}|\omega_j) \sim \mathcal{N}(\vec{\mu}_j, \sigma^2\vec{I}),\ j \in \{A, B\}$. The optimal classification algorithm under this simplified assumption is the *k-means clustering algorithm* which is depicted in Alg. 4. It uses the *Euclidean distance* as a similarity metric and essentially estimates $k$ cluster mean vectors $\vec{\mu}_j$ where $1 \leq j \leq k$ by minimizing the *sum-of-squared-error* criterion to derive the unsupervised classification.

### 6.2.2 Signal-to-Noise Ratio and Bit-Error-Rate for Clustering

This chapter describes the exploitation of single-execution leakage for a non-profiled single-execution side-channel attack. The success probability of side-channel attacks always depends on the Signal-to-Noise-Ratio (SNR) of the exploited leakage signal.

In Chap. 3, I used a method of assessing the SNR which is described in Eq. 3.30. In the context of applying cluster analysis as a tool for attack, an SNR which describes the cluster separation is reasonable. In this way, I assess the SNR as the proportion of the exploited single-execution leakage to the sum of data-dependent, or algorithmic noise and measurement noise. More specifically, and for the case of two classes, the SNR is defined as the logarithm of the quotient of the squared difference of estimated cluster means $\vec{\mu}_A$, and $\vec{\mu}_B$ and the sum of the variances $\sigma_A^2$, and $\sigma_B^2$ of the two clusters. This is formalized in Eq. 6.1.

$$\mathrm{SNR}(\vec{\mu}_A, \vec{\mu}_B, \sigma_A^2, \sigma_B^2) = 10 * \log\left(\frac{(\vec{\mu}_A - \vec{\mu}_B)^2}{(\sigma_A^2 + \sigma_B^2)}\right) \text{ dB} \tag{6.1}$$

The estimated means and variances from above could also be used to compute an expected, theoretic Bit Error Rate (BER) from known distribution parameters. *The BER describes the estimated number of wrongly classified bits in percent from the known parameters.* The fact that the two classes $\omega_A$ and $\omega_B$ have to be labeled correctly with 0 and 1 bit values is ignored since it is a simple one out of two trial. First the two *normally* distributed clusters are projected on a line through the two means. The expected BER then equals the sum of the two tail probabilities (the sum of *both* distributions equals 1) for a distance $d = \frac{|\vec{\mu}_A - \vec{\mu}_B|}{2}$ which is half the distance of the two

means of the equally distributed classes $\omega_j$. (The equation can be simplified because equal distributions are assumed and two times half makes one.) The simplified equation is given in Eq. 6.2.

$$\mathrm{BER}(d, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{-d} e^{-\frac{t^2}{2\sigma^2}} dx \tag{6.2}$$

It is obvious that the expected BER can be decreased by either increasing $d$ or decreasing $\sigma^2$. The same measures increase the SNR. Therefore, it is crucial for single-execution attacks on exponentiation algorithms to increase the SNR of the exploited single-execution leakage to achieve low BERs.

I propose to achieve this through the combination of simultaneous side-channel measurements and describe this the following main Chap. 7.

## 6.3 Practical Evaluation

I performed a practical evaluation of the method described in the previous section to attack an implementation of the Elliptic Curve Scalar Multiplication (ECSM).

### 6.3.1 Design-Under-Attack and Measurement Setup

For this practical evaluation, I used the *Xilinx Spartan-3 (XC3S200)* FPGA-based device under test which has been used in the previous Chap. 5 and described in Sect. 5.4. The FPGA is configured with the implementation of the ECSM as described in Chap. 4. As demonstrated in the previous Chap. 5, this design exhibits location-based information leakage which enabled a successful recovery of the secret exponent in a profiled attack. I will exploit the same location-based leakage in this chapter, but employ a non-profiled attack instead of the profiled template attack.

The same measurement setup as in the previous Chap. 5 is used. Trace compression is employed to reduce the amount of data and computation complexity for the clustering. Different to the measurement setup from the previous chapter, I compressed the trace by computing the sum-of-squared values instead of the peak-to-peak distance in every cycle. Chapter 3 concluded stating that frontside measurement without trace compression are superior in terms of side-channel quality. In this way, the setup can be seen as a kind of worst-case approach. Nonetheless, the practical results achieved with the setup, which are described in the following, are convincing and demonstrate a successful attack.

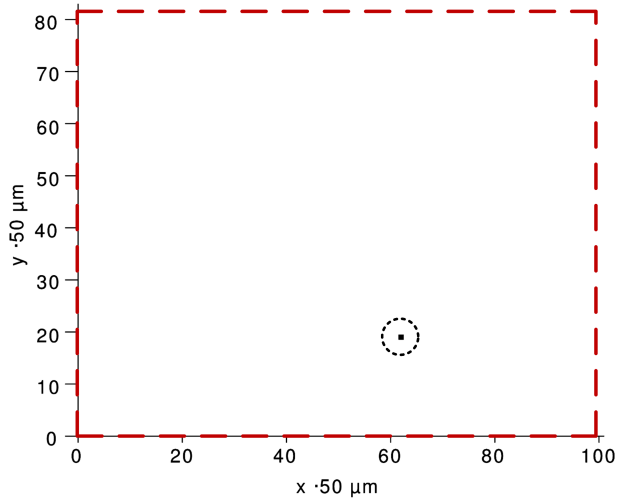



Figure 6.2: FPGA die surface area as dashed rectangle with marked measurement position which exhibits the most single-execution leakage

I performed a difference-of-means test as described in Chap. 5 to find measurement positions with high single-execution information leakage. In this way, I selected the best measurement position which is depicted in Fig. 6.2.

This single-execution leakage is mainly due to location-based information leakage. However, partly, the single-execution leakage may also be due to unbalanced placement of the involved logic cells. This issue has been mentioned in Chap. 5.

It is important, to separate between finding a way of measuring single-execution leakage and exploiting it. The attack described in this chapter does not require profiling. It only requires a sufficient SNR of the single execution leakage. In this practical evaluation, I used a profiling method to find a measurement position with high location-based information leakage.

Hence, I use location-based information leakage as an example to demonstrate how unsupervised clustering can be used to exploit single-execution leakage. However, this could be replaced with any other single-execution side-channel leakage. The next Chap. 7 describes a method to increase the SNR by combining measurements. In this case, measurement positions with lower SNRs can be used, hence, the profiled search for good positions is not necessary anymore.

### 6.3.2 Results of the Practical Clustering Attack

I performed the attack on the measurement at the selected position with the highest SNR of the single-execution leakage. According to Sect. 6.2 and

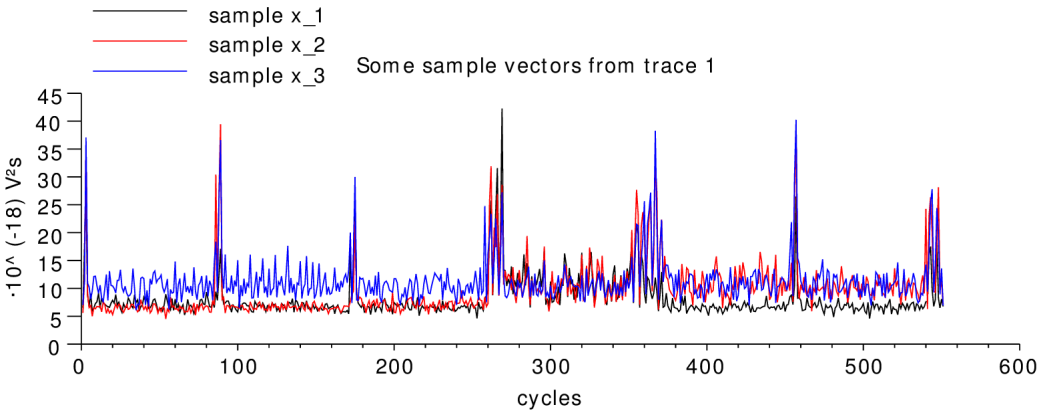



Figure 6.3: Three samples $\vec{t_i}$ from the measurement trace at best position (trace 1)

Fig. 6.1, I segmented the electromagnetic field measurement into multivariate samples $\vec{t_i}$. Figure 6.3 depicts three samples from the measurement as an illustrative example to give the reader an idea. Each multivariate sample contains 551 compressed values which were recorded during 551 cycles while processing one exponent bit.

*I used the unsupervised* k-means *classification algorithm to classify the samples in two clusters as described in Sect. 6.2.*

*As an important result I report, that every exponent bit was classified correctly which proofs that the application of unsupervised cluster classification as a non-profiled single-execution attack on exponentiation algorithms is perfectly meaningful.*

The measurement exhibited an SNR of 13.8 dB according to the metric defined in Eq. 6.1.

In a different practical attack setting, the SNR of the exploited leakage may be insufficient to correctly recover the complete exponent. In the subsequent Chap. 7, I describe, how simultaneous side-channel measurements can be combined in such clustering-based attacks to improve the SNR.

Apart from this improvement, this situation can be dealt with by selectively trialling single bits. During clustering, the densities of the classes of samples are estimated. Given those estimated densities, the posterior class-membership probabilities of samples $\vec{t_i}$ can be computed [DHS01] and used to selectively trial bits. Therefore, even if the exponent is not completely recovered correctly, the search space is reduced significantly. E.g., samples from class $\omega_A$ which are close to the separating plane between $\omega_A$ and $\omega_B$ have a relatively low posterior probability of belonging to class $\omega_A$. Hence, the strategy is to exhaustively try different values for those exponent bits

first.

### 6.3.3 Countermeasures

Regarding the location-based information leakage, countermeasures such as exponent blinding are useless, since the attack uses a single-execution. Therefore, I strongly advise to randomize variable locations as described in Sect. 5.5.

Alternatively, all methods which reduce the SNR of single-execution leakage help to make such attacks more difficult. Location-based information leakage can for instance be reduced by balancing registers and their signal paths or by locating them in an interleaved way that they cannot be distinguished, as proposed by He et al. [HdlTR12]. However, such approaches fail to generally prevent such attacks.

Poucheret et al- [PBB+10] propose to create multiple equivalent datapaths for the computation of intermediate values in cryptographic algorithms which can be exchanged during operation. This seems to introduce a lot of logic overhead for duplication and multiplexing.

## 6.4 Other Applications of Clustering

There are other side-channel attacks, which could be improved through employing unsupervised clustering algorithms.

Schramm et al. [SWP03] introduced side-channel-based collision attacks and use cross-correlation of averaged power consumption measurements to find similarities. Side-channel-based, internal collision attacks [SLFP04, SWP03, MME10, Bog08] are based on distinguishing equal intermediate values during cryptographic computations without leakage modeling or profiling. These collisions depend on secret keys and known input values and, together with analytical properties of the attacked algorithm, allow to recover the secret. Schramm et al. [SLFP04] also applied collision attacks to AES. Bogdanov [Bog07] presents an improvement to a side-channel-based collision attack on the AES algorithm.

I suggest to use unsupervised cluster classification algorithms to detect collisions. The class labels, thus, recovery of the actual processed values remain unknown and are not required for collision attacks. Colliding samples correspond to the side-channel appearance of processed values. The expected number of different classes is $2^n$ for, e.g., $n$ bit values if all values can be distinguished. If only the Hamming weight of the value can be distinguished, $n + 1$ different classes are expected. This is similar to the idea presented

by Lerman et al. [LMV+12] who use clustering of Hamming weights of key bytes to perform a simple attack. However, they restrict this to a univariate analysis at one known time-instant and employ the *k-medoid* clustering algorithm.

## 6.5 Summary

I demonstrated that unsupervised clustering algorithms are powerful for attacking exponentiation algorithms in non-profiled single-execution settings without heuristic leakage modeling. They allow to exploit arbitrary single-execution side-channel leakages of implementations of a wide range of exponentiation algorithms. In the practical evaluation I successfully recovered the secret exponent from a single EM measurement.

# Chapter 7

# Improving the Clustering-Based Attack using Simultaneous EM Measurements

I presented a clustering-based non-profiled attack against exponentiation algorithms in the previous Chap. 6. This attack is theoretically able to exploit arbitrary single-execution side-channel leakage. I demonstrated the attack by exploiting location-based side-channel leakage which can be recovered using localized measurements of electromagnetic fields. The success probability of such an attack depends on the Signal-to-Noise-Ratio (SNR) of the exploited single-execution side-channel leakage. In this chapter, I describe how simultaneous side-channel measurements can be combined in such non-profiled single-execution attacks to increase the SNR, hence, success probability. Parts of this chapter are planned for publication [HIM+13].

I provide practical results using multiple localized measurement positions of the electromagnetic field to combine spatially diverse information. These positions are selected without knowledge, as to where high SNR values would be expected. While the single measurement positions chosen in this way exhibit insufficient SNR for a successful attack, the combination of simultaneous measurements lead to a full recovery of the secret exponent.

I start with presenting related work regarding the combination of simultaneous side-channel measurements in Sect. 7.1. In Sect. 7.2, I describe how the clustering-based attack presented in the previous Chap. 6 can be extended to combine simultaneous measurements. The results from the practical evaluation are presented in Sect. 7.3 and Sect. 7.4 contains a summary of this chapter.

## 7.1 Related Work

The combination of simultaneous side-channel measurements can generally improve the success probabilities of side-channel attacks. However, it depends on the employed statistical tools of the attacks, how measurements can be combined.

Agrawal et al. [ARR03] combine simultaneous measurements of the power consumption and electromagnetic field for profiled template attacks by concatenating them for profiling as well as for template matching. This can easily be extended to other profiled attacks like mutual information analysis and the stochastic approach. Standaert and Archambeau [SA08] extend this and apply Principal Component Analysis (PCA) and Fisher's Linear Discriminant Analysis (LDA) to reduce the data dimensionality for template attacks.

Agrawal et al. [ARR03] present a simple approach to combine simultaneous measurements for classic DPA by treating measurements from different channels jointly. However, this requires that the different channels have similar leakage characteristics at the same cycles. Souissi et al. [SBG+12] and Elaabid et al. [EMGD11] extend Correlation-based differential Power Analysis (CPA) [BCO04] to combine simultaneous measurements by combining the correlation coefficients using a product [EMGD11] or sum [SBG+12]. The proposed methods require either that different channels leak information at the same cycles [SBG+12] or that the leaking cycles for combination can be recovered in a profiling step [EMGD11].

Contrary to previous contributions, I propose the combination of simultaneous side-channel measurements for non-profiled, single-execution attacks without heuristic leakage model.

## 7.2 Using Cluster Analysis to Combine Side-Channel Measurements

Single-execution attacks on exponentiation algorithms generally suffer from low SNRs of the exploited leakage [SI11, Bau12]. Low SNRs reduce the probability of success. In the previous Chap. 6, I described a single-execution attack on exponentiation algorithms which uses unsupervised cluster classification algorithms. In the practical evaluation, I demonstrated, that an adversary may be successful recovering the secret exponent by using a single measurement. This was due to the fact that this single measurement exhibited enough SNR for an attack.

Many countermeasures against SCA aim at reducing the SNR by introducing superficial noise or reducing the leakage signal, e.g., using differential logic styles. Low SNRs can generally be coped with by using more measurements. Averaging of repeated measurements with equal input values is a simple example for increasing the SNR. However, this is not feasible in many cases because the secret changes in every execution of the cryptographic algorithm. I address this by proposing to increase the SNR through combining the contained information from multiple, simultaneous side-channel measurements generally, and specifically suggest to combine spatially diverse measurements of electromagnetic fields.

The combination of measurements for non-profiled single-execution attacks without heuristic leakage model is possible through the application of cluster analysis. Precisely, I propose to combine measurements by extending multivariate samples using values from different measurements. As an example where samples $\mathbf{x}_i^A$ from measurement position $A$ are combined, or joined with samples $\mathbf{x}_i^B$ from measurement position $B$, this can be expressed as $\mathbf{x}_i^{joined} = (\mathbf{x}_i^A, \mathbf{x}_i^B)$. This improves cluster separation, thus classification, if the new measurements contain additional information.

The electromagnetic field side-channel allows spatially diverse measurements containing different information. This has been discussed and practically demonstrated in Chap. 3. In this context, Souissi et al. [SBG+12] suggest to use multiple antennae simultaneously to measure the electromagnetic field.

**Reducing Dimensionality** If the amount of variables in the samples $\vec{t}_i$ impedes computation, it can be reduced using feature extraction methods such as Principal Component Analysis (PCA) [DHS01]. PCA performs a projection into a lower dimensional variable space by maximizing the variance in the samples. However, this happens without regard of clusters or cluster discriminants and, therefore, PCA possibly impairs discrimination.

## 7.3 Practical Evaluation

I performed a practical evaluation of the method described in the previous section. I use spatially diverse measurements of the electromagnetic field and show that the combination of measurements improves the success probability of the attack.

Such measurements can for instance be achieved through an array of electromagnetic probes at different positions over the chip surface. Our measurement setup reflects this idea.

### 7.3.1 Design-Under-test and Measurement Setup

I used exactly the same FPGA-based device-under-test and configured ECSM design as described in the previous Chap. 6. I also use the same measurement equipment. However instead of using one measurement position which is found by profiling the leakage of the device, I use multiple measurement positions. Those positions may have lower qualities in terms of detectable signal leakage, and are chosen by geometrical means instead of profiling. Such measurements could be recorded with an array of electromagnetic probes. The idea of this chapter is the improvement of single-execution attacks by combining simultaneous side-channel measurements. In this way, multiple positions with lower signal qualities are combined to accumulate enough signal for a successful attack.

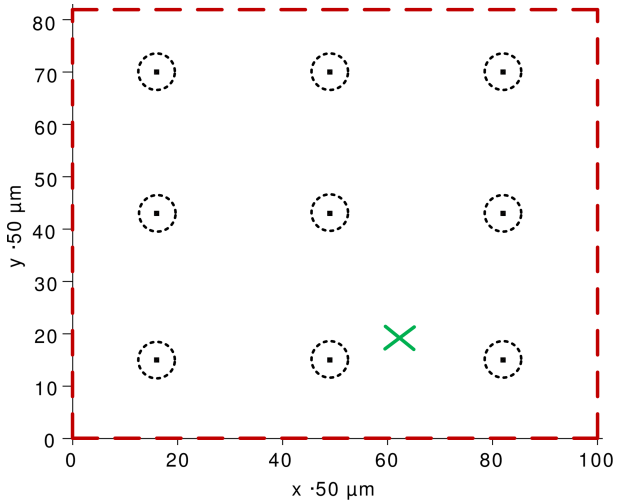



Figure 7.1: FPGA die surface area as dashed rectangle with regular grid of marked measurement positions (dashed circles around dot) and measurement position from previous Chap. 6 as green cross

For the practical evaluation, I used 9 measurement positions which are organized in an equidistant array layout of $3*3$ positions on the surface of the FPGA. The distance between measurement positions in $x$, and $y$ directions is 1.5 mm. This regular grid of measurement positions is depicted in Fig. 7.1. The dashed rectangle depicts the surface of the FPGA die which measures $\approx 5000 * 4000\,\mu m$. The measurement positions are marked with dashed circles. The positions are chosen only according to geometric reasoning regarding a possible array probe. No profiling or heuristic was used to determine the measurement positions. This is contrary to the previous chapter, where I used a difference-of-means test to find the position with most signal leakage.

This measurement position is marked with a green cross.

Since I only had one measurement probe of the same kind available in our laboratory, I repeated the measurement by exactly repeating the computation and moved this one probe over the surface by an $x$-$y$-table to gain measurements from the described positions. In actual devices, the exponent is different every time, preventing repeated measurements. However, I repeated the same computation for several times in a laboratory setup.

While this simplification is not exactly the same as concurrently using multiple probes, I am convinced that the results from this setup are representative. First, I would not expect significant interference between separate measurement chains in an array probe. Second, measurement noise is mainly generated by the probes, amplifiers and the oscilloscope. Hence, it makes no difference from the noise perspective, whether one measurement chain is reused at different times, or several different ones are used simultaneously. The measurement noise is uncorrelated in both cases.

This underlines the assumption that we can provide representative results by repeating measurements.

### 7.3.2 Clustering Combined Measurements

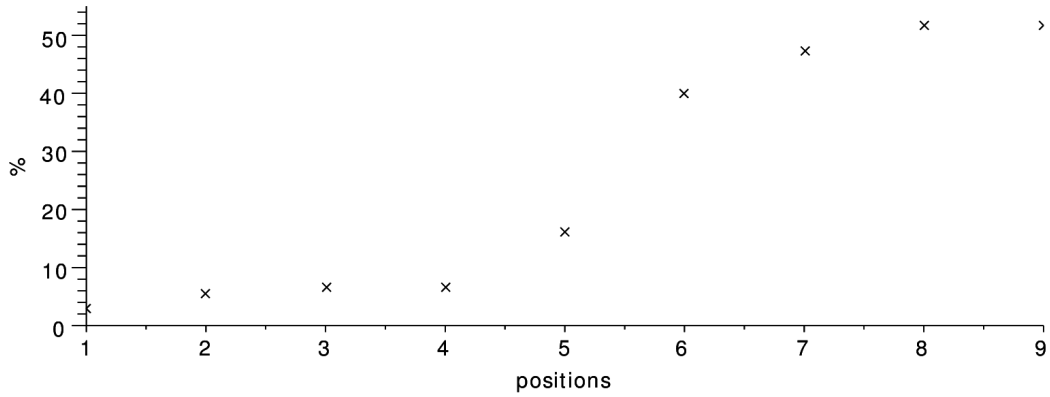



Figure 7.2: BER after clustering for *individual measurements* at different positions

As a first stept I analyzed the measurements separately. Hence, I processed each of the 9 measurements in the same way as I did with the single one from the previous Chap. 6. This is described in Sect. 6.3.2. Hence, I applied the unsupervised *k-means clustering algorithm* to the 9 individual electromagnetic field side-channel measurements that have been recorded at

different positions. Figure 7.2 depicts the Bit-Error-Rate (BER) after applying clustering to the *individual* measurements. The BER denotes the number of bits which are classified incorrectly by the clustering algorithm in percent through comparison of the classification result with the correct exponent.

*The BER of clustering individual measurements shows that no individual measurement could provide enough SNR for a correct classification, thus, complete recovery of the secret scalar.* Some individual measurements even exhibit a very high BER, which means that the SNR of the leaked information, i.e., location-based leakage, is very low in those measurements.

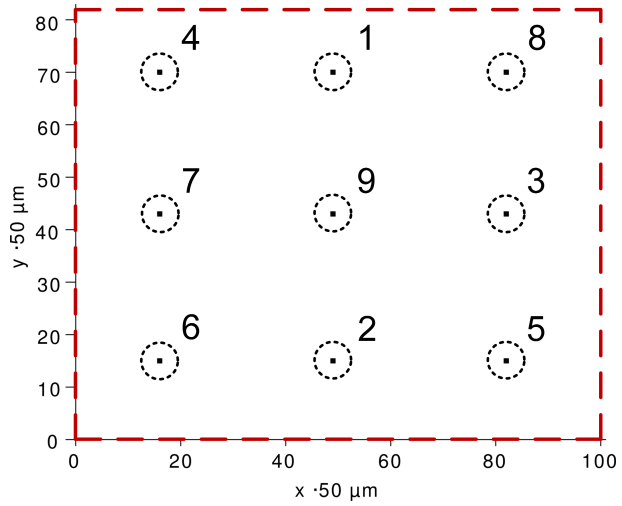



Figure 7.3: FPGA die surface area as dashed rectangle with marked *and numbered* measurement positions

Clear evidence for an advantage of the combination of spatially diverse electromagnetic field measurements from multiple positions is proven, if the best measurement leading to the lowest BER can be further improved by adding more measurements. Therefore, we sorted the measurements from the lowest BER to the highest before incrementally combining them for joint unsupervised cluster classification.

An adversary cannot perform sorting based on low BERs. However, this is not necessary for the adversary because the classification results from joint measurements are independent of the order of joining them.

We performed ordering and incremental joining of measurements to demonstrate the effects of combining measurements step-by-step. Figure 7.3 depicts the regular array of measurement positions including the numbers describing the order after sorting according to BER.

Figure 7.4 depicts the SNR after unsupervised *k-means clustering* of in-

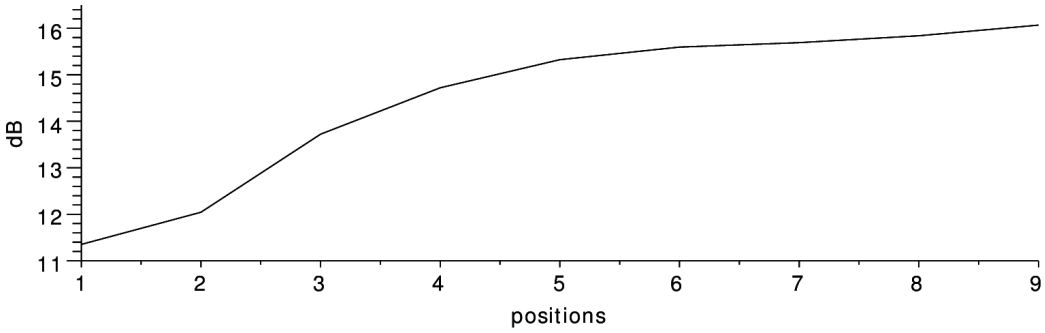


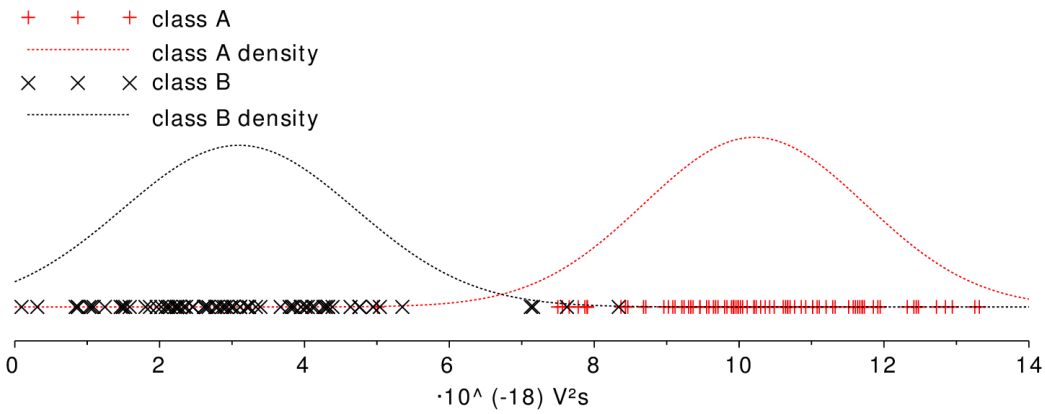

Figure 7.4: SNR after unsupervised clustering of *incrementally joint measurements*

crementally joint measurement positions. It can be observed that the SNR increases with every added measurement and a total increase of $\approx +4.7\,\mathrm{dB}$ is achieved after joining 9 measurements. *This clearly shows the improvement in SNR from combining spatially diverse measurements of the electromagnetic field.*

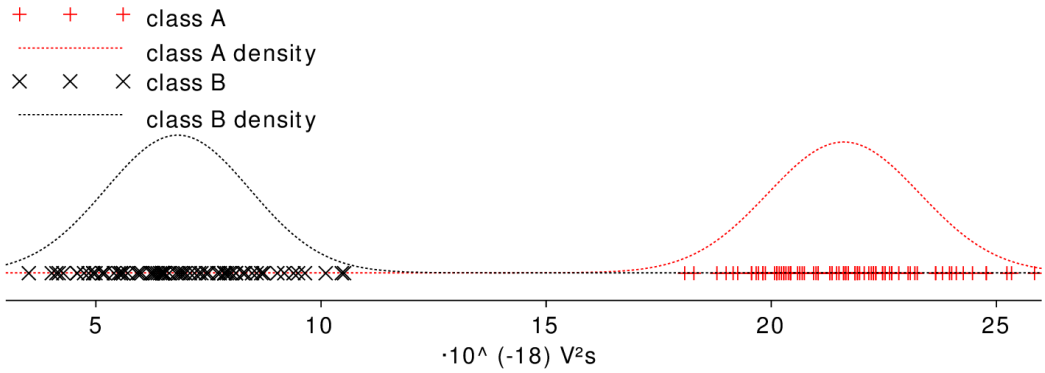
Figure 7.5(a) and Fig. 7.5(b) depict this increase in SNR in a more demonstrative way. Figure 7.5(a) presents a projection of the multivariate samples $\vec{t_i}$ *for the single measurement at position* 1 on the line between the two cluster means $\vec{\mu}_A$ and $\vec{\mu}_B$ derived from unsupervised *k-means clustering*. I used the correct classification to mark the samples accordingly. Additionally, and also using the correct classification, I computed the parameters mean and variance from the two Gaussian distributions. The figure includes two Gaussian curves, denoted as *class A/B density* in Fig. 7.5(a), generated using those parameters to illustrate their distribution. It is obvious, how the two distributions overlap. Samples are across the wrong side of the half distance between the two distributions. This causes errors in the recovery of bits through clustering.

Figure 7.5(b) depicts the same linear projection of samples $\vec{t_i}$ and their Gaussian distribution curves *for* 9 *joint measurements. It can be observed, that the separation of the two classes is significantly improved by combining measurements and no sample appears on the wrong side of the half distance between the two distributions.*

Figure 7.6 depicts the BER after unsupervised classification using the *k-means clustering algorithm* denoted as *BER achieved from unsupervised clustering. This BER of incrementally joint measurements improves and equals zero after* 3 *joint measurements.* This clearly demonstrates the advantage of combining measurements for attacking exponentiation algorithms using unsupervised clustering algorithms.




(a) Distribution of samples $\vec{t}_i$ in measurement at position 1




(b) Distribution of samples $\vec{t}_i$ in 9 joint measurements

Figure 7.5: SNR gain in cluster separation through joint measurements

As a second graph, Fig. 7.6 also depicts the estimated BER when given the parameters from the two distributions. This estimated BER is computed according to Eq. 6.2 from the previous Chap. 6. It presents as an estimation which is based on the simplified model of cluster parameters described in Sect. 6.2 of the previous Chap. 6. The depicted graph matches the empirical result after clustering very closely which is an indicator for the fact that the simplified model fits in a reasonable manner.

In Fig. 7.4, it can be observed, that after 3 joint measurements, the SNR reaches $\approx 14\,\mathrm{dB}$. In Chap. 6, the attack was performed using only a single measurement with high SNR. This single measurement exhibited an SNR of 13.8 dB which is comparable to the SNR of 3 joint measurements in this evaluation.

We learn, that even if an adversary may not be able to analyze a device to find a measurement position with the highest SNR as described in Chap. 6,

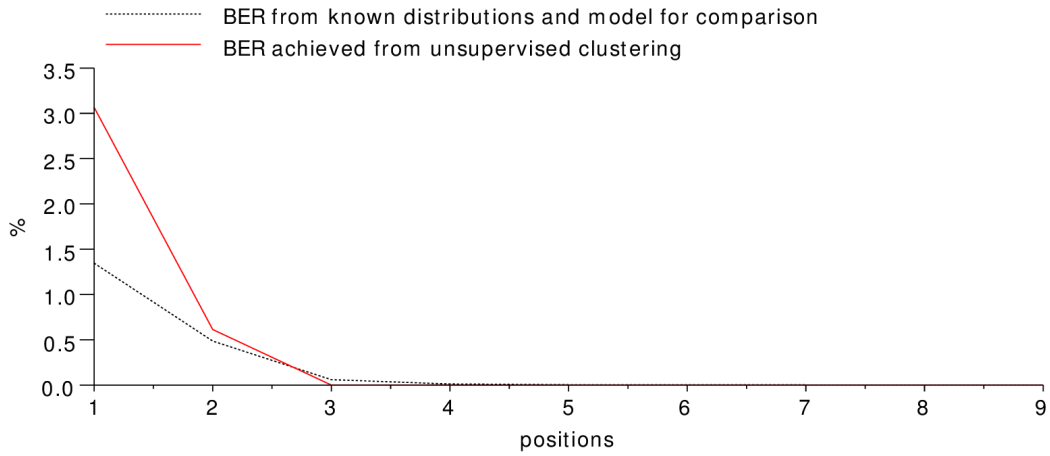



Figure 7.6: BER after unsupervised classification of *incrementally joint measurements*

he can break a device by combining the information leakage of multiple, simultaneous measurements at positions, which do not necessarily exploit the highest SNRs.

## 7.4 Summary

In this chapter I show that the combination of measurements of the electromagnetic field improves single-execution attacks. In the practical evaluation, three measurement positions were sufficient to recover the secret exponent.

An important aspect is that an adversary does not have to find the positions with the highest information leakage. He can instead employ several measurement positions concurrently and retrieve the contained leakage in all positions to achieve a sufficient SNR. This means that an adversary could gain a significant advantage by employing more than one electromagnetic probe.

This strongly emphasizes the requirement to keep the single-execution side-channel leakage of implementations as low as possible in order to counteract such attacks.

# Chapter 8

# Conclusions

In this thesis, I provide results about the most important success factors for localized measurements of electromagnetic fields for side-channel analysis, i.e., direction of the measurement coil, small distance to the frontside integrated circuit surface, and high sampling rates. Such high-resolution measurements of electromagnetic fields have a significant impact on cryptographic implementations for two reasons. First, conventional side-channel attacks can be improved through better signal-to-noise ratios. Second, dedicated side-channel attacks exploiting location-based leakage become feasible.

At eligible positions, such localized measurements lead to significantly higher signal-to-noise-ratios of the exploited leakage information. The reasons for this are less parasitic influences and noise from other circuit parts. However, in most cases, finding such eligible positions will require profiling using a known secret. If this is not possible, measurements at positions which are selected e.g., only because of high signal amplitudes, will provide lower signal-to-noise ratios. This can be seen as a significant drawback of high-resolution electromagnetic field measurements.

The feasibility of localized measurements enables attacks that specifically exploit location-based information leakage. The possible exploitation of such location-based leakage requires countermeasures beyond the current state-of-the-art. This means that whenever a cryptographic algorithm uses different variables depending on the secret, which are stored in different locations, the assignment of such variables to physical storage locations must be randomized. Previous guidelines for the side-channel-secure implementation of cryptographic algorithms require that the operation sequences are independent of the secret, or uniform. In future, the access of variables must either be independent of the secret, or masked in addition to this.

Side-channel attacks exploiting single execution leakage such as location-based leakage can be significantly improved by employing unsupervised clus-

ter classification algorithms. My practical results show, that such an attack can be successful using a single measurement at an eligible location. Furthermore, the case study showed that the combination of multiple measurements, which can be recorded simultaneously, lead to a successful recovery of the scalar even though every single measurement is insufficient to do so. This leads to the important conclusion that an adversary may use an array of measurement probes to record simultaneous measurements of a single execution without any previous profiling of a device. The adversary then has a good chance to recover a secret scalar by combining the measurements without prior profiling. This emphasizes the need for the mentioned novel design guideline.

**Future Work**

There are several topics which could be pursued as a follow-up to my thesis. For example, it remains as a subject for further investigation, to what extent the conclusions about the qualities of high-resolution measurements and about the most appropriate measurement setup from Chap. 3 apply to other FPGAs or ASICs. It is also unknown whether software implementations may be prone to location-based side-channel leakage.

More measurement probes for magnetic fields with different properties could be analyzed and benchmarked. For example, it is an open question, whether probes with even smaller coil diameters will lead to better results in terms of side-channel analysis, or whether the measurement resolution could be enhanced by other means such as signal processing.

A topic for future work is the modeling of the electromagnetic fields of integrated circuits. Open questions are, on which metal layers the predominant fields are created and to what extent the electric and magnetic fields are shielded by other layers within the integrated circuit.

In Chap. 5, I describe randomization of storage locations as a countermeasure against location-based leakage on the algorithm level. This countermeasure may be prone to second-order attacks, which could be investigated.

Another countermeasure against localized electromagnetic field measurements is interleaved placement of digital logic. Such an interleaved placement could be done automatically by design-automation scripts. It should be analyzed, to what extent such interleaved placement can reduce or prevent location-based information leakage. An important aspect in this analysis will be to compare feasible measurement resolutions against the granularity of interleaved placement methods.

In Chap. 6, I use a simplified model for the cluster distributions to be able to employ the k-means clustering algorithm for my practical evaluation.

The expectation-maximization clustering algorithm which has more free parameters could be used instead to improve the clustering results. However, this also increases the computational effort and the trade-off remains as an unanswered question.

It could be investigated in future work, if side-channel-based collision attacks can be improved through applying unsupervised clustering algorithms like I suggest in Chap. 6.

The demonstrated improvement of side-channel attacks through spatially diverse electromagnetic field measurements from Chap. 7 can be adapted to other side-channel attacks. In this way, a differential attack on symmetric ciphers using multiple EM probes could be investigated.

Power measurements of different supply pins from an FPGA device which supply different parts of the circuit and, thus, carry different information, are another eligible candidate for simultaneous measurement. This could be investigated in future efforts.

# Bibliography


[AARR03] Dakshi Agrawal, Bruce Archambeault, Josyula Rao, and Pankaj Rohatgi. The EM sidechannel(s). In Burton Kaliski, Cetin Ko, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2002*, volume 2523 of *Lecture Notes in Computer Science*, pages 29–45. Springer Berlin / Heidelberg, 2003.

[AFV07] Frederic Amiel, Benoit Feix, and Karine Villegas. Power analysis for secret recovering and reverse engineering of public key algorithms. In Carlisle Adams, Ali Miri, and Michael Wiener, editors, *Selected Areas in Cryptography*, volume 4876 of *Lecture Notes in Computer Science*, pages 110–125. Springer Berlin Heidelberg, 2007.

[AG12] Avner Ash and Robert Gross. *Elliptic Tales: Curves, Counting, and Number Theory.* Princeton University Press, March 2012.

[ANS05] ANSI. *ANS X9.622005. Public Key Cryptography for the Financial Services Industry. The Elliptic Curve Digital Signature Algorithm (ECDSA).* American National Standards Institute, 2005.

[ARR03] Dakshi Agrawal, Josyula Rao, and Pankaj Rohatgi. Multi-channel attacks. In Colin Walter, Cetin Ko, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2003*, volume 2779 of *Lecture Notes in Computer Science*, pages 2–16. Springer Berlin / Heidelberg, 2003.

[AT03] Toru Akishita and Tsuyoshi Takagi. Zero-value point attacks on elliptic curve cryptosystem. In Colin Boyd and Wenbo Mao, editors, *Information Security*, volume 2851 of *Lecture*

*Notes in Computer Science*, pages 218–233. Springer Berlin / Heidelberg, 2003.

[Bau12] Sven Bauer. Attacking exponent blinding in rsa without crt. In Werner Schindler and Sorin Huss, editors, *Constructive Side-Channel Analysis and Secure Design*, volume 7275 of *Lecture Notes in Computer Science*, pages 82–88. Springer Berlin / Heidelberg, 2012.

[BBD+08] Holger Bock, Michael Braun, Markus Dichtl, Erwin Hess, Johann Heyszl, Walter Kargl, Helmut Koroschetz, Bernd Meyer, and Hermann Seuschek. A milestone towards rfid products offering asymmetric authentication based on elliptic curve cryptography. In *Workshop on RFID Security 2008*, 2008.

[BCO04] Eric Brier, Christophe Clavier, and Francis Olivier. Correlation power analysis with a leakage model. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004*, volume 3156 of *Lecture Notes in Computer Science*, pages 135–152. Springer Berlin / Heidelberg, 2004.

[BDL97] Dan Boneh, Richard A. DeMillo, and Richard J. Lipton. On the importance of checking cryptographic protocols for faults. In *Proceedings of the 16th annual international conference on Theory and application of cryptographic techniques*, EUROCRYPT'97, pages 37–51, Berlin, Heidelberg, 1997. Springer-Verlag.

[BGLR09] Lejla Batina, Benedikt Gierlichs, and Kerstin Lemke-Rust. Differential cluster analysis. In Christophe Clavier and Kris Gaj, editors, *Cryptographic Hardware and Embedded Systems - CHES 2009*, volume 5747 of *Lecture Notes in Computer Science*, pages 112–127. Springer Berlin / Heidelberg, 2009.

[BGP+11] Lejla Batina, Benedikt Gierlichs, Emmanuel Prouff, Matthieu Rivain, François-Xavier Standaert, and Nicolas Veyrat-Charvillon. Mutual information analysis: a comprehensive study. *J. Cryptology*, 24(2):269–291, 2011.

[BHW12] Lejla Batina, Jip Hogenboom, and JasperG.J. Woudenberg. Getting more from pca: First results of using principal component analysis for extensive power analysis. In Orr Dunkelman,

editor, *Topics in Cryptology CT-RSA 2012*, volume 7178 of *Lecture Notes in Computer Science*, pages 383–397. Springer Berlin Heidelberg, 2012.

[BM06] Joseph Bonneau and Ilya Mironov. Cache-collision timing attacks against aes. In Louis Goubin and Mitsuru Matsui, editors, *Cryptographic Hardware and Embedded Systems - CHES 2006*, volume 4249 of *Lecture Notes in Computer Science*, pages 201–215. Springer Berlin / Heidelberg, 2006.

[BMM00] Ingrid Biehl, Bernd Meyer, and Volker Müller. Differential fault attacks on elliptic curve cryptosystems. In *Advances in CryptologyCRYPTO 2000*, pages 131–146. Springer, 2000.

[Bog07] Andrey Bogdanov. Improved side-channel collision attacks on aes. In *Proceedings of the 14th international conference on Selected areas in cryptography*, SAC'07, pages 84–95, Berlin, Heidelberg, 2007. Springer-Verlag.

[Bog08] Andrey Bogdanov. Multiple-differential side-channel collision attacks on aes. In Elisabeth Oswald and Pankaj Rohatgi, editors, *Cryptographic Hardware and Embedded Systems CHES 2008*, volume 5154 of *Lecture Notes in Computer Science*, pages 30–44. Springer Berlin / Heidelberg, 2008.

[BOS06] Johannes Blömer, Martin Otto, and Jean-Pierre Seifert. Sign change fault attacks on elliptic curve cryptosystems. In Luca Breveglieri, Israel Koren, David Naccache, and Jean-Pierre Seifert, editors, *Fault Diagnosis and Tolerance in Cryptography*, volume 4236 of *Lecture Notes in Computer Science*, pages 36–52. Springer Berlin / Heidelberg, 2006.

[Bro09] Ezra Brown. Three fermat trails to elliptic curves. *Biscuits of Number Theory*, 34:273, 2009.

[BS97] Eli Biham and Adi Shamir. Differential fault analysis of secret key cryptosystems. In *Advances in Cryptology CRYPTO '97*, volume 1294 of *Lecture Notes in Computer Science*, pages 513–525. Springer Berlin / Heidelberg, 1997.

[Bun11] Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen. Algorithmenkatalog 2011. Bundesanzeiger Nr. 85 vom 07. Juni 2011, S. 2034, May 2011.

[Can05] D. Canright. A very compact s-box for aes. In Josyula Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems CHES 2005*, volume 3659 of *Lecture Notes in Computer Science*, pages 441–455. Springer Berlin / Heidelberg, 2005.

[CFG+10] Christophe Clavier, Benoit Feix, Georges Gagnerot, Mylne Roussellet, and Vincent Verneuil. Horizontal correlation analysis on exponentiation. In Miguel Soriano, Sihan Qing, and Javier López, editors, *Information and Communications Security*, volume 6476 of *Lecture Notes in Computer Science*, pages 46–61. Springer Berlin Heidelberg, 2010.

[CJ03] Mathieu Ciet and Marc Joye. (virtually) free randomization techniques for elliptic curve cryptography. In *Information and Communications Security*, volume 2836 of *Lecture Notes in Computer Science*, pages 348–359. Springer Berlin / Heidelberg, 2003.

[CJ05] Mathieu Ciet and Marc Joye. Elliptic curve cryptosystems in the presence of permanent and transient faults. *Designs, Codes and Cryptography*, 36:33–43, 2005.

[Cor99] Jean-Sébastien Coron. Resistance against differential power analysis for elliptic curve cryptosystems. In *CHES '99: Proceedings of the First International Workshop on Cryptographic Hardware and Embedded Systems*, pages 292–302, London, UK, 1999. Springer-Verlag.

[CRR03] Suresh Chari, Josyula Rao, and Pankaj Rohatgi. Template attacks. In Burton Kaliski, Cetin Ko, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2002*, volume 2523 of *Lecture Notes in Computer Science*, pages 51–62. Springer Berlin / Heidelberg, 2003.

[DH76] W. Diffie and M. Hellman. New directions in cryptography. *Information Theory, IEEE Transactions on*, 22(6):644 – 654, nov 1976.

[DHS01] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2 edition, November 2001.

[DMBO+05] E. De Mulder, P. Buysschaert, S.B. Ors, P. Delmotte, B. Preneel, G. Vandenbosch, and I. Verbauwhede. Electromagnetic analysis attack on an fpga implementation of an elliptic curve cryptosystem. In *Computer as a Tool, 2005. EUROCON 2005.The International Conference on*, volume 2, pages 1879 –1882, nov. 2005.

[DO08] Agustin Dominguez Oviedo. *On fault-based attacks and countermeasures for elliptic curve cryptosystems*. PhD thesis, University of Waterloo, Waterloo, Ont., Canada, Canada, 2008.

[DPRS11] Julien Doget, Emmanuel Prouff, Matthieu Rivain, and François-Xavier Standaert. Univariate side channel attacks and leakage modeling. *Journal of Cryptographic Engineering*, 1:123–144, 2011. 10.1007/s13389-011-0010-2.

[ElG85] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In George Blakley and David Chaum, editors, *Advances in Cryptology*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18. Springer Berlin / Heidelberg, 1985.

[EMGD11] M. Elaabid, Olivier Meynard, Sylvain Guilley, and Jean-Luc Danger. Combined side-channel attacks. In Yongwha Chung and Moti Yung, editors, *Information Security Applications*, volume 6513 of *Lecture Notes in Computer Science*, pages 175–190. Springer Berlin / Heidelberg, 2011.

[FGDM+10] Junfeng Fan, Xu Guo, E. De Mulder, P. Schaumont, B. Preneel, and I. Verbauwhede. State-of-the-art of secure ECC implementations: a survey on known side-channel attacks and countermeasures. In *Hardware-Oriented Security and Trust (HOST), 2010 IEEE InternationalSymposium on*, 2010.

[FLRV08] Pierre-Alain Fouque, Reynald Lercier, Denis Réal, and Frédéric Valette. Fault attack on elliptic curve montgomery ladder implementation. In *Proceedings of the 2008 5th Workshop on Fault Diagnosis and Tolerance in Cryptography*, FDTC '08, pages 92–98, Washington, DC, USA, 2008. IEEE Computer Society.

[Fri79] Hans Fricke. *Leitfaden der Elektrotechnik: Grundlagen der elektrischen Nachrichtenübertragung*. Teubner Verlag, 1979.

[FV03] Pierre-Alain Fouque and Frederic Valette. The doubling attack why upwards is better than downwards. In Colin Walter, etin Ko, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2003*, volume 2779 of *Lecture Notes in Computer Science*, pages 269–280. Springer Berlin / Heidelberg, 2003.

[GBTP08] Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel. Mutual information analysis - a generic side-channel distinguisher. In Elisabeth Oswald and Pankaj Rohatgi, editors, *Cryptographic Hardware and Embedded Systems - CHES 2008*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442, Washington DC,US, 2008. Springer-Verlag.

[GHS02] P. Gaudry, F. Hess, and N. Smart. Constructive and destructive facets of weil descent on elliptic curves. *Journal of Cryptology*, 15:19–46, 2002.

[GMO01] Karine Gandolfi, Christophe Mourtel, and Francis Olivier. Electromagnetic analysis: Concrete results. In Cetin Ko, David Naccache, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems CHES 2001*, volume 2162 of *Lecture Notes in Computer Science*, pages 251–261. Springer Berlin / Heidelberg, 2001.

[Gou02] Louis Goubin. A refined power-analysis attack on elliptic curve cryptosystems. In Yvo Desmedt, editor, *Public Key Cryptography PKC 2003*, volume 2567 of *Lecture Notes in Computer Science*, pages 199–211. Springer Berlin / Heidelberg, 2002.

[HdlTR12] Wei He, Eduardo de la Torre, and Teresa Riesgo. An interleaved epe-immune pa-dpl structure for resisting concentrated em side channel attacks on fpga implementation. In Werner Schindler and Sorin Huss, editors, *Constructive Side-Channel Analysis and Secure Design*, volume 7275 of *Lecture Notes in Computer Science*, pages 39–53. Springer Berlin / Heidelberg, 2012.

[HIM+13] Johann Heyszl, Andreas Ibing, Stefan Mangard, Fabrizio De Santis, and Georg Sigl. Clustering algorithms for non-profiled single-execution attacks on exponentiations. Cryptology ePrint Archive, Report 2013/438, 2013. http://eprint.iacr.org/.

[HMA+08] Naofumi Homma, Atsushi Miyamoto, Takafumi Aoki, Akashi Satoh, and Adi Shamir. Collision-based power analysis of modular exponentiation using chosen-message pairs. In Elisabeth Oswald and Pankaj Rohatgi, editors, *Cryptographic Hardware and Embedded Systems CHES 2008*, volume 5154 of *Lecture Notes in Computer Science*, pages 15–29. Springer Berlin / Heidelberg, 2008.

[HMH+12a] Johann Heyszl, Stefan Mangard, Benedikt Heinz, Frederic Stumpf, and Georg Sigl. Localized electromagnetic analysis of cryptographic implementations. In Orr Dunkelman, editor, *Topics in Cryptology CT-RSA 2012*, volume 7178 of *Lecture Notes in Computer Science*, pages 231–244. Springer Berlin / Heidelberg, 2012.

[HMH+12b] Johann Heyszl, Dominik Merli, Benedikt Heinz, Fabrizio De Santis, and Georg Sigl. Strengths and limitations of high-resolution electromagnetic field measurements for side-channel analysis. In Stefan Mangard, editor, *Smart Card Research and Advanced Applications*, Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012.

[HMV03] Darrel Hankerson, Alfred J. Menezes, and Scott Vanstone. *Guide to Elliptic Curve Cryptography*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.

[HPM94] Patrick Horster, Holger Petersen, and Markus Michels. Meta-ElGamal signature schemes. In *CCS '94: Proceedings of the 2nd ACM Conference on Computer and communications security*, pages 96–107, New York, NY, USA, 1994. ACM.

[HS10] Johann Heyszl and Frederic Stumpf. Efficient one-pass entity authentication protocol based on ECC forconstrained devices. In *IEEE Int. Symposium on Hardware-Oriented Security and Trust (HOST 2010)*, Anaheim, USA, June 2010. IEEE Computer Society.

[IIT03a] Kouichi Itoh, Tetsuya Izu, and Masahiko Takenaka. Address-bit differential power analysis of cryptographic schemes OK-ECDH and OK-ECDSA. In *Cryptographic Hardware and Embedded Systems - CHES 2002*, volume 2523 of *Lecture Notes in Computer Science*, pages 399–412. Springer Berlin / Heidelberg, 2003.

[IIT03b] Kouichi Itoh, Tetsuya Izu, and Masahiko Takenaka. A practical countermeasure against address-bit differential power analysis. In *Cryptographic Hardware and Embedded Systems - CHES 2003*, volume 2779 of *Lecture Notes in Computer Science*, pages 382–396. Springer Berlin / Heidelberg, 2003.

[ISO02] ISO. *ISO/IEC 15946-2: Information technology – Security techniques – Cryptographic techniques based on elliptic curves – Part 1: Digital Signatures*. International Organization for Standardization, 2002.

[IT88] Toshiya Itoh and Shigeo Tsujii. A fast algorithm for computing multiplicative inverses in GF(2m) using normal bases. *Inf. Comput.*, 78(3):171–177, 1988.

[JY03] Marc Joye and Sung-Ming Yen. The montgomery powering ladder. In Burton Kaliski, Cetin Ko, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2002*, volume 2523 of *Lecture Notes in Computer Science*, pages 1–11. Springer Berlin / Heidelberg, 2003.

[KD02] Louis Kruh and Cipher Deavours. The commercial enigma: beginnings of machine cryptography. *Cryptologia*, 26(1):1–16, January 2002.

[Ker83] Auguste Kerckhoffs. La cryptographie militaire. *Journal des sciences militaires*, IX:5–83, January 1883.

[Kir11] Mario Kirschbaum. *Power Analysis Resistant Logic Styles - Design, Implementation, and Evaluation*. PhD thesis, Technische Universität Graz, 2011.

[KJJ99] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '99, pages 388–397, London, UK, 1999. Springer-Verlag.

[KK99] Oliver Kömmerling and Markus G. Kuhn. Design principles for tamper-resistant smartcard processors. In *Proceedings of the USENIX Workshop on Smartcard Technology on USENIX Workshop on Smartcard Technology*, pages 2–2, Berkeley, CA, USA, 1999. USENIX Association.

[Kna92] A. W. Knapp. *Elliptic Curves*. Mathematical Notes Series. Prentice Hall, 1992.

[Kob87] Neal Koblitz. Elliptic Curve Cryptosystems. *Mathematics of Computation*, 48(177):203–209, 1987.

[Koc96] Paul C. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In *Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '96, pages 104–113, London, UK, 1996. Springer-Verlag.

[Koc03] Paul C. Kocher. Leak-resistant cryptographic indexed key update, 2003. United States Patent 6,539,092 filed at San Francisco, CA.

[KOP10] Timo Kasper, David Oswald, and Christof Paar. A versatile framework for implementation attacks on cryptographic RFIDs and embedded devices. In Marina L. Gavrilova, Chih Jeng Kenneth Tan, and Edward D. Moreno, editors, *Transactions on Computational Science X - Special Issue on Security in Computing*, volume 10 of *Lecture Notes in Computer Science*, pages 100–130. Springer, 2010.

[KP06] Sandeep Kumar and Christof Paar. Are standards compliant elliptic curve cryptosystems feasible on rfid. In *Workshop on RFID Security*, pages 12–14, 2006.

[KQQ99] Francois Koeune, Jean-Jacques Quisquater, and Jean-jacques Quisquater. A timing attack against rijndael. Technical report, Université catholique de Louvain, 1999.

[KS11] Mario Kirschbaum and Jörn-Marc Schmidt. Learning from electromagnetic emanations - a case study for iMDPL. In *Workshop Proceedings COSADE 2011*, pages 50 – 55, 2011.

[LBSV10] Yong Ki Lee, Lejla Batina, Dave Singelée, and Ingrid Verbauwhede. Low-cost untraceable authentication protocols for rfid. In *Proceedings of the third ACM conference on Wireless network security*, WiSec '10, pages 55–64, New York, NY, USA, 2010. ACM.

[LD99a] Julio López and Ricardo Dahab. Fast multiplication on elliptic curves over GF(2m) without precomputation. In *CHES*

*'99: Proceedings of the First International Workshop on Cryptographic Hardware and Embedded Systems*, pages 316–327, London, UK, 1999. Springer-Verlag.

[LD99b] Julio López and Ricardo Dahab. Improved algorithms for elliptic curve arithmetic in GF(2n). In *SAC '98: Proceedings of the Selected Areas in Cryptography*, pages 201–212, London, UK, 1999. Springer-Verlag.

[Len87] Hendrik W. Lenstra. Factoring integers with elliptic curves. *The Annals of Mathematics*, 126(3):649–673, November 1987.

[LMM05] Huiyun Li, A. Markettos, and Simon Moore. Security evaluation against electromagnetic analysis at design time. In Josyula Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems CHES 2005*, volume 3659 of *Lecture Notes in Computer Science*, pages 280–292. Springer Berlin / Heidelberg, 2005.

[LMV+12] Liran Lerman, Stephane Fernandes Medeiros, Nikita Veshchikov, Cedric Meuter, Gianluca Bontempi, and Olivier Markowitch. Semi-supervised template attack. Cryptology ePrint Archive, Report 2012/082, 2012. http://eprint.iacr.org/.

[LRP07] Kerstin Lemke-Rust and Christof Paar. Gaussian mixture models for higher-order side channel analysis. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pages 14–27. Springer Berlin / Heidelberg, 2007.

[LSBV08] Yong Ki Lee, Kazuo Sakiyama, Lejla Batina, and Ingrid Verbauwhede. Elliptic curve based security processor for rfid. *IEEE Trans. Computers*, 57(11):1514–1527, 2008.

[MBLM12] Dimitrios Mavroeidis, Lejla Batina, Twan Laarhoven, and Elena Marchiori. Pca, eigenvector localization and clustering for side-channel attacks on cryptographic hardware devices. In PeterA. Flach, Tijl Bie, and Nello Cristianini, editors, *Machine Learning and Knowledge Discovery in Databases*, volume 7523 of *Lecture Notes in Computer Science*, pages 253–268. Springer Berlin Heidelberg, 2012.

[MDS99a] Thomas Messerges, Ezzy Dabbish, and Robert Sloan. Power analysis attacks of modular exponentiation in smartcards. In *Cryptographic Hardware and Embedded Systems*, volume 1717 of *Lecture Notes in Computer Science*, pages 724–724. Springer Berlin / Heidelberg, 1999.

[MDS99b] Thomas S. Messerges, Ezzy A. Dabbish, and Robert H. Sloan. Investigations of power analysis attacks on smartcards. In *Proceedings of the USENIX Workshop on Smartcard Technology on USENIX Workshop on Smartcard Technology*, pages 17–17, Berkeley, CA, USA, 1999. USENIX Association.

[Mil86] Victor Miller. Use of elliptic curves in cryptography. In Hugh Williams, editor, *Advances in Cryptology CRYPTO 85 Proceedings*, volume 218 of *Lecture Notes in Computer Science*, pages 417–426. Springer Berlin / Heidelberg, 1986.

[MME10] Amir Moradi, Oliver Mischke, and Thomas Eisenbarth. Correlation-enhanced power analysis collision attack. In Stefan Mangard and Franois-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems, CHES 2010*, volume 6225 of *Lecture Notes in Computer Science*, pages 125–139. Springer Berlin / Heidelberg, 2010.

[MO09] Marcel Medwed and Maria Elisabeth Oswald. Template attacks on ECDSA. In *9th International Workshop, WISA 2008, Jeju Island, Korea, September 23-25, 2008, Revised Selected Papers*, Lecture Notes in Computer Science, pages 14 – 27. Springer, 2009.

[Mon87] Peter L. Montgomery. Speeding the Pollard and Elliptic Curve Methods of Factorization. *Mathematics of Computation*, 48(177):243–264, 1987.

[MOP07] Stefan Mangard, Elisabeth Oswald, and Thomas Popp. *Power Analysis Attacks: Revealing the Secrets of Smart Cards (Advances in Information Security)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

[MOS09] Stefan Mangard, Elisabeth Oswald, and Francois-Xavier Standaert. One for all - all for one: Unifying standard dpa attacks. Cryptology ePrint Archive, Report 2009/449, 2009. http://eprint.iacr.org/.

[MOVR01] Alfred J. Menezes, Paul C. Van Oorschot, Scott A. Vanstone, and R. L. Rivest. *Handbook of Applied Cryptography*. CRC Press, Inc., Boca Raton, FL, USA, 2001.

[Mul10] Elke De Mulder. *Electromagnetic Techniques and Probes for Side-Channel Analysis on Cryptographic Devices*. PhD thesis, Katholieke Universiteit Leuven, 2010. Bart Preneel and Ingrid Verbauwhede (promotors).

[NIS94] NIST. *Digital Signature Standard (DSS) (FIPS PUB 186)*. U.S. National Institute of Standards and Technology, 1994.

[NIS97] NIST. *Entity Authentication using Public Key Cryptography (FIPS PUB 196)*. U.S. National Institute of Standards and Technology, 1997.

[NIS99] NIST. *Recommended Elliptic Curves For Federal Government Use (FIPS PUB 186-3)*. U.S. National Institute of Standards and Technology, July 1999.

[NIS01] NIST. *Advanced Encryption Standard (AES) (FIPS PUB 197)*. U.S. National Institute of Standards and Technology, November 2001.

[NIS09] NIST. *Digital Signature Standard (DSS) (FIPS PUB 186-3)*. U.S. National Institute of Standards and Technology, 2009.

[NSS+04] David Naccache Nigel, Nigel P. Smart, Jacques Stern, Gemplus Card International, and Rue Guynemer. Projective coordinates leak. In *Advances in Cryptology - EUROCRYPT 2004, volume 3027 of LNCS*, pages 257–267. Springer, 2004.

[PBB+10] Francois Poucheret, Lyonel Barthe, Pascal Benoit, Lionel Torres, Philippe Maurine, and Michel Robert. Spatial em jamming: A countermeasure against em analysis? In *VLSI-SoC*, pages 105–110. IEEE, 2010.

[PHS03] Josef Pieprzyk, Thomas Hardjono, and Jennifer Seberry. *Fundamentals of computer security*. Springer, 2003.

[PP10] Christof Paar and Jan Pelzl. *Understanding Cryptography - A Textbook for Students and Practitioners*. Springer, 2010.

[PQ03] Gilles Piret and Jean-Jacques Quisquater. A differential fault attack technique against SPN structures, with application to the AES and Khazad. In Colin Walter, Cetin Ko, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2003*, volume 2779 of *Lecture Notes in Computer Science*, pages 77–88. Springer Berlin / Heidelberg, 2003.

[PSQ07] Eric Peeters, François-Xavier Standaert, and Jean-Jacques Quisquater. Power and electromagnetic analysis: improved model, consequences and comparisons. *Integr. VLSI J.*, 40(1):52–60, January 2007.

[PTBM12] G. Perin, L. Torres, P. Benoit, and P. Maurine. Amplitude demodulation-based em analysis of different rsa implementations. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2012*, pages 1167 –1172, march 2012.

[QS01] Jean-Jacques Quisquater and David Samyde. Electromagnetic analysis (EMA): Measures and counter-measures for smart cards. In Isabelle Attali and Thomas Jensen, editors, *Smart Card Programming and Security*, volume 2140 of *Lecture Notes in Computer Science*, pages 200–210. Springer Berlin / Heidelberg, 2001.

[RHSDPK06] Francisco Rodríguez-Henríquez, N. A. Saqib, A. Díaz-Pèrez, and Cetin Ko. *Cryptographic Algorithms on Reconfigurable Hardware (Signals and Communication Technology)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[RS09] Mathieu Renauld and Francois-Xavier Standaert. Algebraic side-channel attacks. Cryptology ePrint Archive, Report 2009/279, 2009. http://eprint.iacr.org/.

[RSA78] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, February 1978.

[RVD09] D. Real, F. Valette, and M. Drissi. Enhancing correlation electromagnetic attack using planar near-field cartography. In *Design, Automation Test in Europe Conference Exhibition, 2009. DATE '09.*, pages 628 –633, April 2009.

[SA08] Franois-Xavier Standaert and Cedric Archambeau. Using subspace-based template attacks to compare and combine

power and electromagnetic information leakages. In Elisabeth Oswald and Pankaj Rohatgi, editors, *Cryptographic Hardware and Embedded Systems CHES 2008*, volume 5154 of *Lecture Notes in Computer Science*, pages 411–425. Springer Berlin / Heidelberg, 2008.

[SBG+12] Youssef Souissi, Shivam Bhasin, Sylvain Guilley, Maxime Nassar, and Jean-Luc Danger. Towards different flavors of combined side channel attacks. In Orr Dunkelman, editor, *Topics in Cryptology CT-RSA 2012*, volume 7178 of *Lecture Notes in Computer Science*, pages 245–259. Springer Berlin / Heidelberg, 2012.

[Sch90] Claus-Peter Schnorr. Efficient identification and signatures for smart cards. In *CRYPTO '89: Proceedings of the 9th Annual International Cryptology Conference on Advances in Cryptology*, pages 239–252, London, UK, 1990. Springer-Verlag.

[SGM09] Laurent Sauvage, Sylvain Guilley, and Yves Mathieu. Electromagnetic radiations of fpgas: High spatial resolution cartography and attack on a cryptographic module. *ACM Trans. Reconfigurable Technol. Syst.*, 2:4:1–4:24, March 2009.

[SH08] J.-M. Schmidt and C. Herbst. A practical fault attack on square and multiply. In *Fault Diagnosis and Tolerance in Cryptography, 2008. FDTC '08. 5th Workshop on*, pages 53 –58, August 2008.

[SI11] Werner Schindler and Kouichi Itoh. Exponent blinding does not always lift (partial) SPA resistance to higher-level security. In Javier López and Gene Tsudik, editors, *Applied Cryptography and Network Security*, volume 6715 of *Lecture Notes in Computer Science*, pages 73–90. Springer Berlin / Heidelberg, 2011.

[Sko10] S. Skorobogatov. Optical fault masking attacks. In *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2010 Workshop on*, pages 23 –29, August 2010.

[SLFP04] Kai Schramm, Gregor Leander, Patrick Felke, and Christof Paar. A collision-attack on AES. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004*, volume 3156 of *Lecture Notes*

*in Computer Science*, pages 146–169. Springer Berlin / Heidelberg, 2004.

[SLP05] Werner Schindler, Kerstin Lemke, and Christof Paar. A stochastic model for differential side channel cryptanalysis. In Josyula Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2005*, volume 3659 of *Lecture Notes in Computer Science*, pages 30–46. Springer Berlin / Heidelberg, 2005.

[SMR09] Dhiman Saha, Debdeep Mukhopadhyay, and Dipanwita RoyChowdhury. A diagonal fault attack on the advanced encryption standard. Cryptology ePrint Archive, Report 2009/581, 2009. http://eprint.iacr.org/.

[SP98] Leilei Song and Keshab K. Parhi. Low-energy digit-serial/parallel finite field multipliers. *J. VLSI Signal Process. Syst.*, 19(2):149–166, 1998.

[SWP03] Kai Schramm, Thomas Wollinger, and Christof Paar. A new class of collision attacks and its application to DES. In Thomas Johansson, editor, *Fast Software Encryption*, volume 2887 of *Lecture Notes in Computer Science*, pages 206–222. Springer Berlin / Heidelberg, 2003.

[Wag12] Mathias Wagner. 700+ attacks published on smart cards: The need for a systematic counter strategy. In *COSADE*, pages 33–38, 2012.

[Wal01] C. Walter. Sliding windows succumbs to big mac attack. In Cetin Ko, David Naccache, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems CHES 2001*, volume 2162 of *Lecture Notes in Computer Science*, pages 286–299. Springer Berlin / Heidelberg, 2001.

[Wer02] Annette Werner. *Elliptische Kurven in der Kryptographie*. Springer-Verlag Berlin Heidelberg New York, 2002.

[WH11] Erich Wenger and Michael Hutter. A hardware processor supporting elliptic curve cryptography for less than 9 kges. In Emmanuel Prouff, editor, *Smart Card Research and Advanced Applications*, volume 7079 of *Lecture Notes in Computer Science*, pages 182–198. Springer Berlin Heidelberg, 2011.

[WH12a] Erich Wenger and Michael Hutter. Exploring the design space of prime field vs. binary field ecc-hardware implementations. In Peeter Laud, editor, *Information Security Technology for Applications*, volume 7161 of *Lecture Notes in Computer Science*, pages 256–271. Springer Berlin / Heidelberg, 2012.

[WH12b] Erich Wenger and Michael Hutter. Exploring the design space of prime field vs. binary field ecc-hardware implementations. In Peeter Laud, editor, *Information Security Technology for Applications*, volume 7161 of *Lecture Notes in Computer Science*, pages 256–271. Springer Berlin Heidelberg, 2012.

[WvWM11] Marc Witteman, Jasper van Woudenberg, and Federico Menarini. Defeating RSA multiply-always and message blinding countermeasures. In Aggelos Kiayias, editor, *Topics in Cryptology CT-RSA 2011*, volume 6558 of *Lecture Notes in Computer Science*, pages 77–88. Springer Berlin / Heidelberg, 2011.

[YJ00] Sung-Ming Yen and M. Joye. Checking before output may not be enough against fault-based cryptanalysis. *Computers, IEEE Transactions on*, 49(9):967 –970, September 2000.