

Number 665



**UNIVERSITY OF
CAMBRIDGE**

Computer Laboratory

Security evaluation at design time for cryptographic hardware

Huiyun Li

April 2006

15 JJ Thomson Avenue
Cambridge CB3 0FD
United Kingdom
phone +44 1223 763500
<http://www.cl.cam.ac.uk/>

© 2006 Huiyun Li

This technical report is based on a dissertation submitted December 2005 by the author for the degree of Doctor of Philosophy to the University of Cambridge, Trinity Hall.

Technical reports published by the University of Cambridge Computer Laboratory are freely available via the Internet:

<http://www.cl.cam.ac.uk/TechReports/>

ISSN 1476-2986

Abstract

Consumer security devices are becoming ubiquitous, from pay-TV through mobile phones, PDA, prepayment gas meters to smart cards. There are many ongoing research efforts to keep these devices secure from opponents who try to retrieve key information by observation or manipulation of the chip's components. In common industrial practise, it is after the chip has been manufactured that security evaluation is performed. Due to design time oversights, however, weaknesses are often revealed in fabricated chips. Furthermore, post manufacture security evaluation is time consuming, error prone and very expensive. This evokes the need of *design time security evaluation* techniques in order to identify avoidable mistakes in design.

This thesis proposes a set of *design time security evaluation* methodologies covering the well-known non-invasive side-channel analysis attacks, such as power analysis and electromagnetic analysis attacks. The thesis also covers the recently published semi-invasive optical fault injection attacks. These security evaluation technologies examine the system under test by reproducing attacks through simulation and observing its subsequent response.

The proposed *design time security evaluation* methodologies can be easily implemented into the standard integrated circuit design flow, requiring only commonly used EDA tools. So it adds little non-recurrent engineering (NRE) cost to the chip design but helps identify the security weaknesses at an early stage, avoids costly silicon re-spins, and helps succeed in industrial evaluation for faster time-to-market.

Acknowledgements

This project would not have been possible without the support of many people. I would like to thank my supervisor Dr. Simon Moore for his valuable help and encouragement throughout my research. A. Theodore Markettos and Jacques Fournier provided considerable help with experiments. This work could not have been completed without EDA tools for which I have to thank Robert Mullins. Thanks are also due to Scott Fairbanks for helpful discussions on HSPICE coding, and Sergei Skorobogatov for discussions on optical fault injection. I would like to give thanks to my husband who endured this long process with me, always offering support and love. Engineering and Physical Sciences Research Council (EPSRC) funded this research project.

Contents

1	Introduction	9
1.1	Motivation	9
1.2	Approaches	10
1.3	Outline of the Thesis	10
2	Background	13
2.1	Overview of Smart Card Technologies	13
2.1.1	Types of Smart Card Interface	13
2.1.2	Smart Card Architecture	15
2.1.3	Applications	16
2.2	Smart Card Security Mechanisms	18
2.2.1	Authentication	19
2.2.2	Confidentiality, Integrity and Non-repudiation	22
2.2.3	“Security through Obscurity” vs. “Kerckhoffs’ Principle”	22
2.3	Smart Card Attack Technologies	23
2.3.1	Non-invasive Attacks	23
2.3.2	Invasive Attacks	26
2.3.3	Semi-invasive Attacks	26
2.4	Defence Technologies	27
2.4.1	Countermeasures to non-invasive attacks	27
2.4.2	Countermeasures to semi-invasive and invasive attacks	27
2.5	Summary	28
3	Simulating Power Analysis Attacks	29
3.1	DPA Simulation Methodology	29
3.1.1	Simulation Procedure	30
3.2	Results	34
3.2.1	Simulation Result	34
3.2.2	Measurement Result	36
3.3	Summary	37
4	Simulating EMA Attacks	39
4.1	Background	39
4.1.1	Origin of EM Emissions	39
4.1.2	Near and Far Fields	40
4.1.3	Direct vs Modulated EM emissions	43
4.1.4	EM field Measurement Equipment	43

4.2	Simulation Methodology for EM Analysis	45
4.2.1	System Partitioning	45
4.2.2	Simulation Procedure	45
4.3	Evaluation Results	47
4.3.1	EM Simulation Setup	47
4.3.2	EM Simulation of a Synchronous Processor	48
4.3.3	EM Simulation of an Asynchronous Processor	50
4.4	Summary	52
5	Simulating Optical Fault Injection	53
5.1	Background	54
5.1.1	Ionisation and Charge Collection	54
5.1.2	Metal Shielding Effect	55
5.1.3	Classes of Attackers	55
5.1.4	Modelling Optical Fault Induction	56
5.2	Simulation Methodology	57
5.2.1	Simulation Procedure	58
5.3	Results	61
5.3.1	Optical Attack Simulation Results	61
5.3.2	Experimental Results	63
5.4	Summary	63
	Appendix	64
6	Conclusion and Future Work	69
6.1	Conclusion	69
6.2	Future Work	70
	List of Papers	73

Chapter 1

Introduction

1.1 Motivation

Cryptographic devices, such as secure microcontrollers and smart cards, are widely used in security applications across a wide range of businesses. These devices generally have an embedded cryptographic processor running cryptographic algorithms such as triple DES, AES or RSA, together with a non-volatile memory to store the secure key. Although the algorithms are provably secure, the system can be broken if the keys can be extracted from smart cards or terminals by side-channel analysis attacks, such as timing analysis [35], power consumption analysis [37], or electromagnetic radiation analysis [54] attacks. Timing and power analysis have been used for years to monitor the processes taking place inside microcontrollers and smart cards. It is often possible to figure out what instruction is currently being executed and what number of bits set/reset in an arithmetic operation, as well as the states of carry, zero and negative flags. However, as chips become more and more complex with instruction/data caches and pipelining mechanisms inside their CPUs, it becomes more and more difficult to observe their operation through direct power analysis. A statistical technique has more recently been used to correlate the data being manipulated and the power being consumed. This technique works effectively, and is easily extended from the power side-channel to the electromagnetic side-channel.

With the advancing attack techniques, it is no longer sufficient for the cryptographic processors to withstand the above passive attacks, they should also endure attacks that inject faults into the devices and thus cause exploitable abnormal behaviour. The abnormal behaviour may be a data error setting part of the key to a known value, or a missed conditional jump reducing the number of rounds in a block cipher. Optical fault injection [58] appears to be a powerful and dangerous attack. It involves illumination of a single transistor or a group of adjacent transistors, and causes them to conduct transiently, thereby introducing a transient logic error.

Many designs are contrived to keep cryptographic devices secure against these attacks. To evaluate these designs, it is common industrial practise to test the design post manufacture. This post-manufacture analysis is time consuming, error prone and very expensive. This has driven my study of design-time security evaluation which aims to examine data-dependent characteristics of secure processors, so as to assess their security level against side-channel analysis attacks. Also this design-time security evaluation should cover optical fault injection attacks which have recently aroused interest in the security community.

This design-time security evaluation should be easily employed in the framework of an integrated circuit (IC) design flow. It should be systematic and exhaustive and should be performed in a relatively short time while providing relatively accurate and practical results (compared to commercial post-manufacture test).

1.2 Approaches

This thesis comes up with approaches:

- To simulate differential power analysis (DPA) of secure processors, which includes power simulation of the logic circuitry and low-pass filtering caused by on-chip parasitics and package inductance.
- To simulate electromagnetic analysis (EMA) attacks. This design-time security evaluation methodology first partitions the system under test into two parts: the chip and the package. The package is simulated in an EM simulator and modelled with lumped parameters R, L and C. The chip incorporating the package lumped parameters is then simulated using circuit simulators. This mixed-level simulation obtains current consumption of the system under test accurately and swiftly. Next, the security evaluation methodology involves a procedure of data processing on the current consumption to simulate EM emissions. Different methods of data processing are demanded to target corresponding types of sensors. Furthermore, to simulate modulated EM emissions, demodulation in amplitude or angle is incorporated into the simulation flow.
- To evaluate the security of cryptographic processors against optical fault injection attacks. This simulation methodology involves exhaustively scanning over the layout with any virtual laser spot size according to the attack scenario. The exposed cells for each scan are mapped to their internal nodes. Then the nodes are supplied transient voltage sources via tri-state buffers. These voltage sources temporarily bring down the potential of the selected n-transistor output nodes or raise up the potential for p-transistor output nodes. Finally the circuit behaviour is examined and compared to the normal one without any laser illumination.

The proposed simulation methodologies are easy to employ in the framework of an integrated circuit design flow. They can spot design oversights at an early stage, helping to avoid costly silicon re-spins. With this simulation methodology, we are able to move one step closer to a complete security-aware design flow for cryptographic processors.

1.3 Outline of the Thesis

The rest of the thesis is organised as follows.

Chapter 2 reviews smart card technologies and the associated security issues. Existing attack technologies are surveyed and classified. Some defence technologies that can be used through design, and evaluated by the design-time security evaluation suite, are also discussed.

Chapter 3 introduces the simulation methodology for DPA. Simulation results are demonstrated and compared with measurement results on a test chip.

Chapter 4 introduces the origin of EM emission from IC chips, and the equipment used in EMA attacks. The chapter then presents the simulation methodology that includes system partitioning and current consumption data processing. The chapter also demonstrates the simulation results on the test chip from which data dependent EM characteristics are successfully identified and verified by the measurement results.

Chapter 5 introduces the physical mechanism of laser radiation, ionisation and charge absorption. Then it presents the simulation methodology that includes layout scanning, exposed node list extraction and circuit simulation that incorporates transient voltage supplies to these exposed nodes. Simulation results on the test chip are demonstrated which match the experimental results.

Chapter 6 identifies areas of future work related to this work and provides some concluding remarks.

Chapter 2

Background

2.1 Overview of Smart Card Technologies

Smart cards were first introduced in Europe in 1976 in the form of memory cards, used to store payment information for the purpose of reducing thefts from pay phones. Since then smart cards have been evolving into a much more advanced form to have both microprocessor and memory in a single chip. They are now widely used for secure processing and storage, especially for security applications that use cryptographic algorithms.

The Joint Technical Committee 1 (JTC1) of the International Standards Organisation (ISO) and the International Electrotechnical Commission (IEC) defined an industry standard for smart card technology in 1987. This series of international standards ISO/IEC 7816 [6], started in 1987 with its latest update in 2003, defines various aspects of a smart card, including physical characteristics, physical contacts, electronic signals and transmission protocols, commands, security architecture, application identifiers, and common data elements [2]. ISO/IEC 7816 describes a smart card as an **Integrated Circuit Card (IC card)** which encompasses all those devices where an integrated circuit is contained within an ISO ID1 identification card piece of plastic [6]. The standard card is $85.6mm \times 53.98mm \times 0.76mm$, the same size as a credit card. When used as a Subscriber Identity Module (SIM) card, the plastic card is small, just big enough to fit inside a cellphone.

2.1.1 Types of Smart Card Interface

Smart cards can be contact or contactless. As the name implies, **contact smart cards** work by communicating via physical contact between a card reader and the smart card's 8-pin contact. **Contactless smart cards**, on the other hand, make use of an embedded antenna and electromagnetic signal to create the interaction between cards and card readers. Operating power is supplied to a card by an inductive loop using low-frequency electromagnetic radiation. Signal communications may be transmitted in a similar way or use capacitive coupling. Contactless cards avoid contamination or wear of contacts which are a frequent source of failure for contact cards. Collision needs to be taken into consideration though. Frequency-multiplexing techniques can be used to distinguish individual cards [4]. Figure 2.1 and 2.2 depict contact and contactless smart cards respectively.

Hybrid smart cards are dual-chip cards. Each chip has its respective contact and contactless interface, not connected to each other. When there is only a single chip that has both

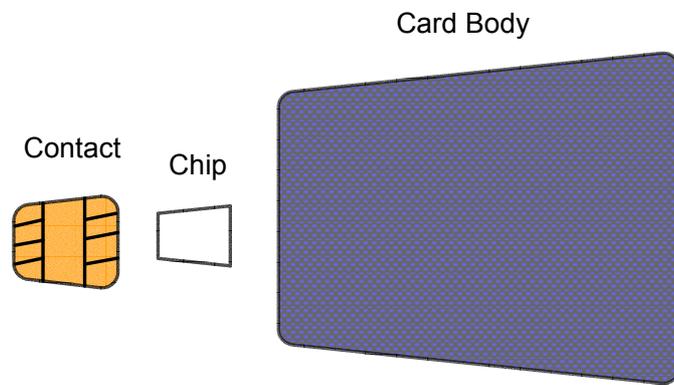


Figure 2.1: Contact Smart Card

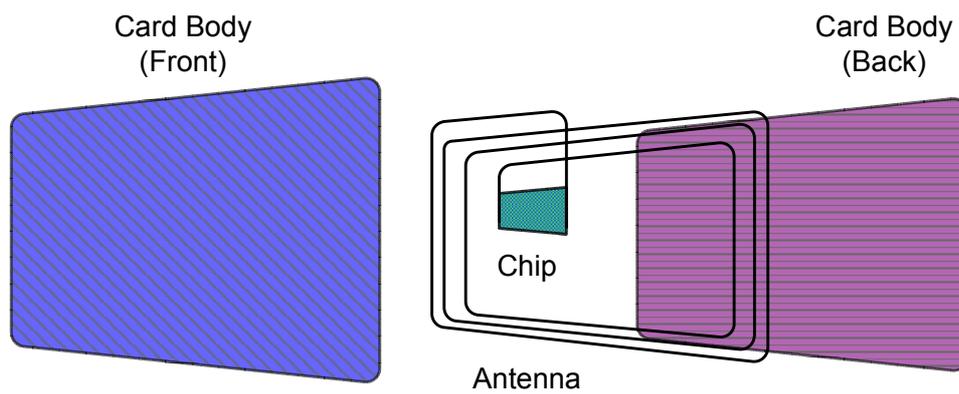


Figure 2.2: Contactless Smart Card

contact and contactless interfaces, the card is referred to as a **combi card**.

2.1.2 Smart Card Architecture

Although some IC cards are just memory cards that merely contain protected non-volatile memory, only those IC cards containing a CPU (Central Processing Unit) are called a smart card, since it is the CPU that justifies the term “smart”. This thesis will refer to a CPU-containing IC card as a “smart card” unless otherwise claimed. As shown in Figure 2.3, a smart card integrated circuit typically consists of [34]:

- a CPU core (e.g. 8-bit Intel 8051, Motorola 68HC05, 16-bit Hitachi H8, or 32-bit ARM 7 processor)
- a hierarchy of 3 classes of memory
 - ROM (read-only memory) – ROM is non-volatile, non-writable. It is used to store operating system routines and diagnostic functions.
 - EEPROM (electronic erasable programmable ROM) or flash memory – They are readable any number of times, but programmed only a limited number of times. They are where data and program code can be read and written under control of the operating system.
 - RAM (random-access memory) – RAM is volatile when power is turned off. It is used to hold transient data during computation.
- a serial I/O interface – It is a single register for data transferring bit by bit, defined by ISO 7816.

In addition to the above basic functional elements. some manufactures offer special coprocessors on the chip to perform cryptographic algorithms.

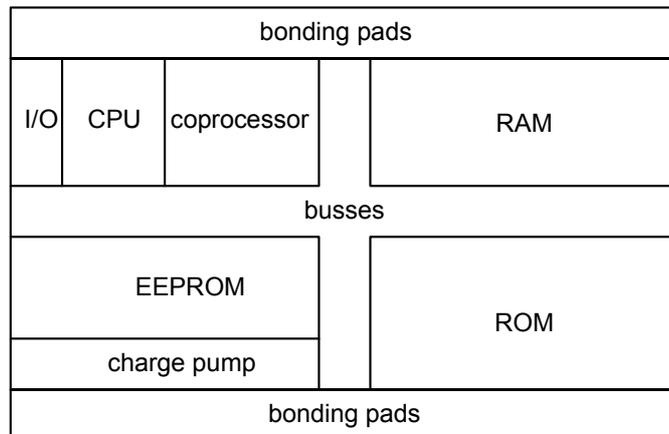


Figure 2.3: Typical arrangement of the functional elements of a smart card microcontroller on a semiconductor die, after [55]

2.1.3 Applications

Smart cards are entering a dramatically growing number of service applications to take the place of money, tickets, documents and files. Credit cards, cash-less pay phones, road toll systems, logical access control devices, health care files and pay TV are just a few of the current examples. Some of the applications will be discussed as follows [5].

1. Transportation

With billions of transport transactions occurring each day, smart cards have found a place in this rapidly growing market. For example, using contactless smart cards allows a passenger to ride several buses and trains during his daily commute to work while not having to worry about complex fare structures or carrying change. In Singapore and London, for example, buses and underground railways use contactless smart cards to collect fares. Each time passengers enter a bus or underground, they pass their card in front of a reader which deducts the fare from the credit stored on the card.

2. Communication

Prepaid Telephone Cards Although various forms of magnetic and optical card have been used for public telephone services, most telephone operators choose smart cards as the most effective card form due to their small overhead. Currently about 80 countries throughout the world use smart cards in public telephone services.

Securing Mobile Phones The *Global System for Mobile communications* (GSM) is a digital cellular communication system widely used in over 90 countries worldwide. A GSM phone uses a SIM card which stores all the personal information of the subscriber. Calls to the subscriber mobile number will be directed accordingly and bills will be charged to the subscriber's personal account. Secure data concerning the GSM subscription is held in the smart card, not in the telephone. A secret code, known as a PIN (Personal Identification Number), is also available to protect the subscriber from misuse and fraud.

3. Electric Utilities

Electric utility companies in the United Kingdom, France and other countries are using smart cards to replace meter reading for prepayment. Customers purchase electricity at authorised payment centers and are issued with a smart card. Customers can also use the card to access information about their account such as amount remaining, amount consumed yesterday or last month, and the amount of remaining credit. An emergency threshold is built in to allow customers to use electricity and pay at a later time. Once the emergency threshold is consumed, electricity is shut off.

4. Computer Security

Boot Integrity Token System (BITS) The *boot integrity token system* (BITS) was developed to protect computer systems from a large number of viruses that affect the booting system, and enforce control of access [18]. BITS is designed so that the computer boots from a boot sector stored on the smart card, bypassing the

boot sector on the computer which can easily be infected by a virus. The card can also be configured to allow access to the computer only by authorised users.

Authentication in Kerberos In an open *distributed computing environment* (DCE), a workstation cannot be trusted to identify its users because the workstation may not be located in a well controlled environment and may be far away from the central server. A user can be an intruder who may try to attack the system or pretend to be someone else to extract information from the system which he/she is not entitled to.

Kerberos [60] is one of the systems which provides trusted third-party authentication services to authenticate users on a distributed network environment. Basically, when a client requests an access to a particular service from the server, the client has to obtain a ticket or credential from the Kerberos *authentication server* (AS). The client then presents that credential to the *ticket granting server* (TGS) and obtains a service ticket. Hence, the user can request the service by submitting the service ticket to the desired server.

Using this protocol, the server can be assured that it is offering services to the client authorised to access them. This is because Kerberos assumes that only the correct user can use the credential as others do not have the password to decrypt it. However, a user can actually request the credential of others, because the user is not authenticated initially.

In this way, an attacker can obtain the credential of another user, and perform an off-line attack using a password guessing approach as the ticket is sealed by password only. This security weakness of Kerberos is identified in [26] and some implementations integrate a smart card into the Kerberos system to overcome this problem. The security of Kerberos is enhanced by authenticating the user via a smart card before granting the initial ticket, so that one user cannot have the ticket of another [26].

5. Medical / Health

Smart cards can also carry medical information such as details of medical insurance coverage, drug sensitivities, medical records, name and phone number of doctors, and other information vital in an emergency.

In the United States, Oklahoma City has a smart card system called MediCard, available since 1994. This smart card is able to selectively control access to a patient's medical history, which is recorded on his/her MediCard. However, essential information, including family physician and close relative to contact, is available to emergency personnel in extreme circumstances. Smart card readers are installed at hospitals, pharmacies, ambulance services, physician's offices and even with the fire department, allowing the MediCard to be used in both ordinary and emergency circumstances [5].

Germany has issued cards to all its citizens that carry their basic health insurance information. In France and Japan, kidney patients have access to cards that contain their dialysis records and treatment prescriptions. These cards are designed with security features to control access to the information for authorised doctors and personnel only.

6. Personal Identification

Several countries including Spain and South Korea have begun trials with smart cards

that provide identification (ID) for their citizens. An ID document in the form of a smart card can hold digitised versions of the holder's signature, photograph and probably his/her biometric information. In an ID system that combines smart card and biometric technologies, a "live" biometric image (e.g., scan of a fingerprint or iris) is captured at the point of interaction and compared to a stored biometric image that was captured when the individual enrolled in the ID system. Smart cards provide the secure, convenient and cost-effective ID technology that stores the enrolled biometric template and compares it to the "live" biometric template. This kind of personal ID system is designed to solve the fundamental problem of verifying that individuals are who they claim to be [9].

7. Payment Card

The payment card has been in existence for many years. It started in the form of a card embossed with details of the card-holder, such as account number, name, expiration date, which could be used at a point of sale to purchase goods or services. The magnetic stripe was soon introduced to cut the cost and errors involved in keying in vouchers for embossed cards. The magnetic stripe also allowed card-holder details to be read electronically in a suitable terminal and allowed automated authorisation. As the criminal fraternity found ways of producing sufficiently good counterfeit cards, magnetic stripe cards have now been developed to the point where there is little or no further scope for introducing more anti-crime measures. An improvement over traditional magnetic strips is Watermark Magnetics technology [39] where a unique watermark pattern is encoded for each card. Watermark encoding relies on the changes in particle orientation. It differs from traditional magnetic stripe encoding which relies on polarity reversals. Together with an active reading technology, the watermark pattern encoded into each card is secure against fraudulent attempts at duplication. However, although possessing the merits of low cost and high security, Watermark Magnetics does not have the memory capacity of the widely publicised smart cards. This has caused the card association of Europay, MasterCard and Visa (EMV) to announce an extensive commitment to include a microprocessing chip on all credit and debit cards distributed worldwide [23].

From the anti-crime perspective, there are a number of benefits in adopting the smart card. The card itself (or in conjunction with the terminal) can make decisions about whether or not a transaction can take place. Secret values can be stored on the card which are not accessible to the outside world allowing for example, the card to check the cardholder's PIN without having to go online to the card issuer's host system. Also there is the possibility of modifying the way the card works while it is inserted in a point of sale terminal even to the point of blocking the card from further transactions if it has been reported lost or stolen.

2.2 Smart Card Security Mechanisms

In the previous section, I kept saying smart cards provide security in various kinds of applications. But what is the actual meaning of "security" in the aspect of information technology? Generally speaking, there are four primary properties or requirements that security addresses here:

- **Confidentiality** is the assurance that information is not disclosed to unauthorised individuals or processes.
- **Integrity** is ensuring that information retains its original level of accuracy
- **Authentication** is the process of recognising/verifying valid users or processes and what system resources a user or process is allowed to access
- **Non-repudiation** provides assurance to senders and receivers that a message cannot subsequently be denied by the sender

To fulfil these four basic requirements of security, various security mechanisms are available to the designers of cryptographic devices. The most important mechanisms are based on the use of cryptographic algorithms, which encrypt/decrypt sensitive information using secret keys.

Smart card security mechanisms are based on the use of cryptographic algorithms. Let us consider an application environment to illustrate a typical security mechanism of smart cards. In this environment as shown in Figure 2.4, we have a personal computer with an attached smart card reader (the terminal). The terminal provides the remote interface to allow the smart card to communicate with the authentication center (*e.g.*, via the Internet).

2.2.1 Authentication

Consider the environment illustrated in Figure 2.4. There are actually four entities involved in the act of authentication:

- the card-holder
- the smart card
- the terminal system
- the remote authentication center

Card-holder Authentication

To authenticate the identities involved requires two separate actions [33]. First, the card-holder must authenticate himself to the smart card. This step prevents fraudulence by some person other than the real card-holder. Normally, the mechanism used to authenticate identity is the proof of knowledge of a secret shared between the card and the holder. In this case, the card-holder is usually asked to enter a PIN, typically a four- to eight-digit number that can be entered through a PIN pad or a terminal keyboard. The PIN is passed over to the card, which verifies that it matches a stored PIN value on the card (*e.g.* on the EEPROM). It should be noted that the card-holder must trust the host computer when entering the PIN. If the terminal is not trustworthy, then the PIN could be compromised, and an impostor could use the PIN to authenticate himself to the card and use the card on behalf of someone other than the true card-holder.

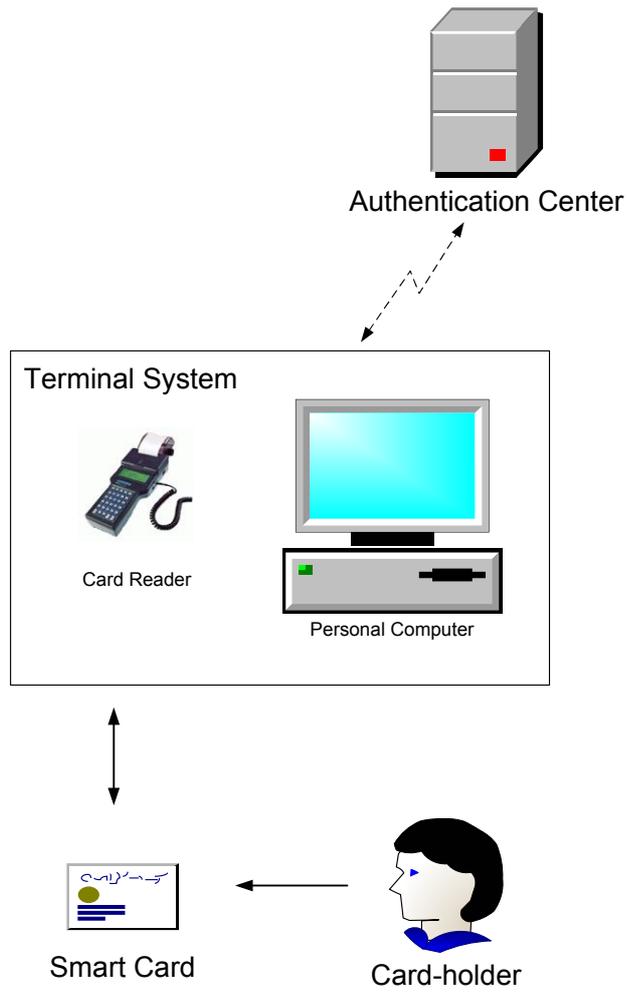


Figure 2.4: An application environment to illustrate the smart card security mechanism

Authentication Between the Card and the Authentication Center

The next process is the mutual authentication of the card with the authentication center (AuC), or in some cases, only the authentication of the card to the AuC. The authentication between the AuC and the card is also based on proving knowledge of a shared secret. However, the secret should not appear on the communication channel linking the card and the terminal. For example, let us consider a naïve protocol illustrated in Figure 2.5. The left part is the operations performed by AuC via the terminal in the middle whilst the right part is the operations performed by the card in response to commands issued from the terminal.

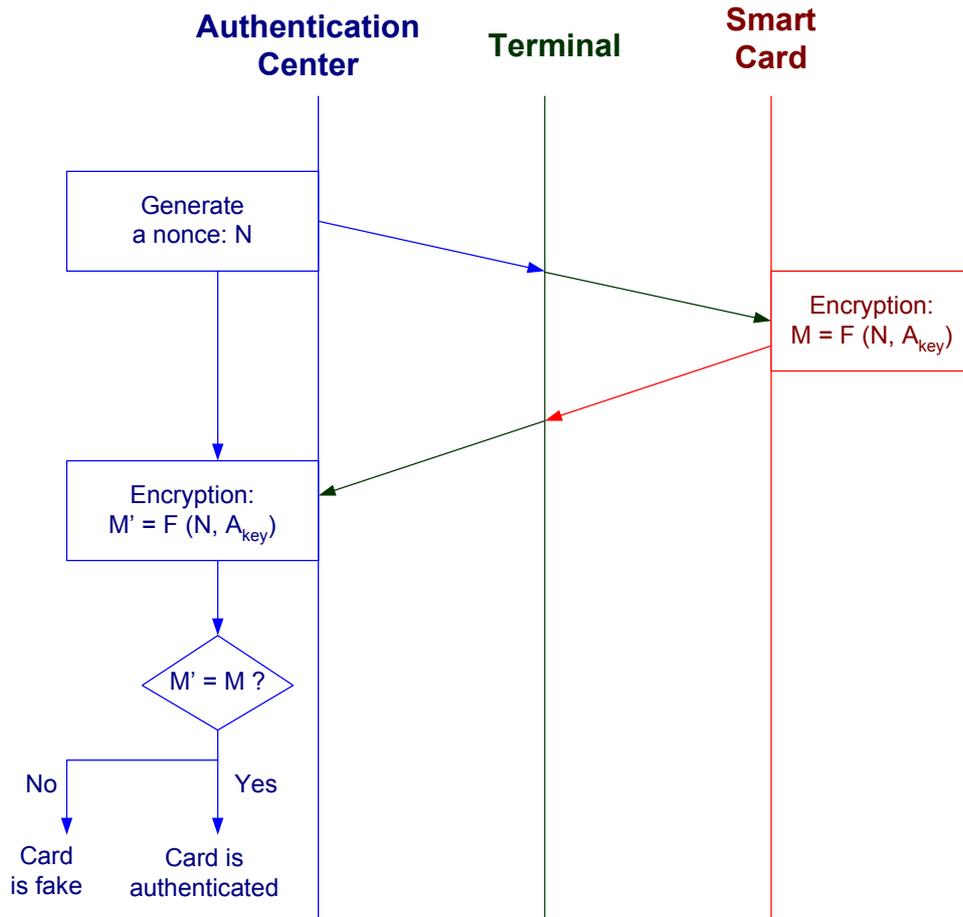


Figure 2.5: The process of the card authenticating itself to the AuC

First, the AuC generates a “number used once”, or *nonce*, N . Then the AuC issues a command via the terminal for the card to authenticate, along with the nonce N . The card encrypts N using the secret key A_{key} , generating M by computing $M = F(N, A_{key})$. M is returned to the AuC which compares the result with its own computation $M' = F(N, A_{key})$ where A_{key} is its copy of the key. If $M' = M$, it means the card knows the true key A_{key} , *i.e.*, the card is authenticated. If $M' \neq M$, then the card is fake and will be rejected. In some protocols, the AuC authenticates itself to the card in a similar manner.

This “challenge-response” authentication method prevents attackers from intercepting the conversation and getting the key A_{key} , since the key A_{key} never passes through the communication channel and the challenge is unique for each transaction. The scheme presented, however, must be refined to prevent “man-in-the-middle”, replay or other attacks [11].

It should be noted that the illustrated authentication process makes use of very important characteristics of smart cards [33]. First, all of the shared secret A_{key} is stored on the smart card in a secure manner. Even when an impostor gains control of a card (e.g., via a rogue terminal), he cannot easily extract this secret information from the card. In fact, it takes great deal of effort to extract information from the card. Attacks on smart cards will be examined later in detail. The second useful characteristic of smart cards is their capability to perform complex cryptographic algorithms, as the cards contain microprocessors and are in essence computer platforms.

2.2.2 Confidentiality, Integrity and Non-repudiation

If a smart card is designed as an identity card or access control card, then the above authentication is all the card security mechanism requires. If the card is designed for applications requiring confidentiality, integrity and/or non-repudiation, such as personal information storage (e.g., medical card) or in a financial services (e.g., credit card), then additional security mechanisms may be required for secure operation. For example, during a secure operation the card may need to encrypt data for confidentiality, hash the data for integrity, or digitally sign the data using a private key for non-repudiation.

2.2.3 “Security through Obscurity” vs. “Kerckhoffs’ Principle”

Smart cards have microprocessors to execute cryptographic algorithms and memories to store secret keys. To keep the algorithm secret is *security through obscurity*, which attempts to use secrecy of design, implementation, etc., to ensure security [66]. A system relying on security through obscurity may have theoretical or actual security vulnerabilities, but its owners or designers believe that the flaws are not known, and that attackers are unlikely to find them. For example, if somebody stores a spare key under the doormat in case they are locked out of the house, then they are relying on security through obscurity. The theoretical security vulnerability is that anybody could break into the house by unlocking the door using the spare key. However, the house owner believes that the location of the key is not known to the public, and that a burglar is unlikely to find it. In this instance, since burglars often know likely hiding places, the house owner would be poorly advised to do so.

Many argue that security through obscurity is flawed for a number of reasons. First, keeping the details of widely-used systems and algorithms secret is difficult. In cryptography, there are a number of examples of proprietary ciphers becoming public knowledge, either by reverse engineering or by a leaked description [66]. Furthermore, keeping algorithms and protocols unpublished means that the ability to review the security is limited only to a few. But many believe that when not keeping a design secret, issues can be found faster and hence can be fixed faster.

The reverse of security by obscurity is *Kerckhoffs’ principle*¹ from the late 1880s [65], which states that system designers should assume that the entire design of a security system is known to all attackers, with the exception of the cryptographic key: "the security of a cypher resides entirely in the key". This principle is widely embraced by cryptographers. In accordance with Kerckhoffs’ principle, the majority of civilian cryptography makes use of

¹not to be confused with Kirchoff’s circuit laws

publicly-known algorithms, although ciphers used to protect classified government or military information are still often kept secret.

Another advantage of keeping the key rather than the algorithm secret is that the disclosure of the cryptographic algorithm would lead to major logistic headaches in developing, testing and distributing implementations of a new algorithm. Whereas if the secrecy of the algorithm were not important, but only that of the keys used with the algorithm, then disclosure of the keys would require a much less arduous process to generate and distribute new keys. Or in other words, the fewer the things one needs to keep secret in order to ensure the security of the system, the easier it is to maintain that security.

2.3 Smart Card Attack Technologies

According to the above description, the security of a smart card system must not depend on keeping the cryptographic algorithm secret, but on keeping the key secret. Attack approaches thus mainly focus on how to retrieve secret keys. Depending on the extent of physical intrusion, and thus on the amount of evidence left on the target device, attacks can be categorised into three types: non-invasive, invasive or semi-invasive attacks.

2.3.1 Non-invasive Attacks

Non-invasive attacks do not physically damage the device under attack and no tamper evidence is left after being applied. An important kind of non-invasive attack is through analysing *side-channel* signals. Every time the smart card performs a computation using the secret data, information may be leaked in the form of timing [36], power dissipation [37] or electromagnetic emission [25, 54] etc. Analysing information like these to extract secret keys is called a *side-channel attack*. These attacks can be performed relatively quickly and easily, while leaving no evidence of tampering, hence they are of particular concern to this project.

Power Analysis Attack

Power dissipation is an important source of side-channel information. For a contact smart card, power is supplied by an external source that can often be directly observed. In smart cards which are mostly made with static CMOS circuits, generally two types of dynamic power are dissipated: switching power and short-circuit power. Switching power is used for charging/discharging parasitic capacitances. Current is only drawn from the power supply when output has a 0-1 transition. During the 1-0 transition, the output capacitor is discharged and energy is dissipated. When there is no data transition (0-0 or a 1-1), no power is used. This asymmetric power consumption provides clues for power analysis attacks. Short-circuit power is due to the short-circuit current drawn when the input of a gate is in transition and both the p- and n- channel transistors are conducting at the same time. Very slow rise and fall times on the input could make this current significant, and must be considered for gates at the end of long wires with large RC delays. But in general the percentage of short-circuit power is smaller than switching power. These two types of dynamic power result in *transition count*² information leakage.

²Transition count is the amount of bits that have changed between two consecutively processed data strings.

On the other hand, the absolute *Hamming weight* information may leak through the data bus. For example, when a precharged bus is used in a design where the data bus is usually precharged to “1”. The number of “0”s driven on to the precharged bus determines the amount of current discharged from a capacitive load C_{load} .

Power analysis attacks exploit these two data-dependent information leakages in an attempt to extract secret keys. Power analysis can be performed in two ways: *Simple Power Analysis* (SPA) and *Differential Power Analysis* (DPA). The former uses pattern matching to identify relevant power fluctuations, while the latter uses statistical analysis to extract information correlated to secret keys [37]. For example, Figure 2.6 demonstrates the first round of Data Encryption Standard (DES) cryptographic algorithm. The 64-bit input block is divided into left and right halves L_0 and R_0 , which are swapped. The left 32-bit half is expanded into 48 bits and then *XORed* with the 48-bit secret key of the first round (K_1). Take K_1 as 8 6-bit subkeys: $K_1 = [K_{1_1} \dots K_{1_8}]$. Then each subkey is *XORed* with 1/8 of the expanded L_0 . For example, K_{1_1} (6 bits) is *XORed* with the first 6 bits of the expanded L_0 , resulting the 6-bit $S1_{input}$ going to the substitution box $S1$. DPA begins by running the DES algorithm N times for N random values of plaintext input. For each run, the power consumption trace is collected. Then the attacker hypothesises all 2^6 possible values of the subkey K_{1_1} . For each guessed subkey, the attacker calculates the corresponding intermediate output $S1_{output}$ (4 bits). Then he divides the power traces into two groups according to one bit (e.g., the least significant bit) of $S1_{output}$. The attacker averages each partition to remove noise, and finally computes a differential trace (the difference between the averages of the two partitions). If the subkey hypothesis is false, then the two partitions are randomly grouped, and the differential trace should be random; If the subkey hypothesis is true, noticeable peaks will occur in the differential trace, indicating points where the subkey was manipulated.

Timing Attack

Smart cards take slightly different amounts of time to perform different operations [36]. Attackers can then garner the leaked information to obtain the secret keys just as they do through power analysis attacks. For example, the cryptographic algorithms based on modular exponentiation, such as Diffie-Hellman and RSA, consist of computing $R = y^x \pmod n$. The goal is to find the w -bit long secret key x . If a particular bit of x_k is 1, then R_k is computed as $R_k = (x_k \cdot y) \pmod n$; if this bit x_k is 0, then R_k is computed as $R_k = x_k$. The slow operation $R_k = (x_k \cdot y) \pmod n$ takes a long time to process, thus leaking the information about x_k .

Masking timing characteristics was suggested as a countermeasure [36]. It could be done either by making all operations take exactly same amount of time or by adding random delay. However, these are difficult for the following reasons:

- Fixed time implementations are slow since the whole system speed will have to depend on the slowest operation.
- Making software run in fixed time is hard, because compiler optimisations and other factors can introduce unexpected timing variations. If a timer is used to delay returning results until a pre-specified time, power consumption may in turn change detectably.
- Random delays can be filtered out by collecting more measurements. The number of samples required increases roughly with the square of the timing noise. For instance, if a modular exponentiator whose timing characteristics have a standard deviation of 10

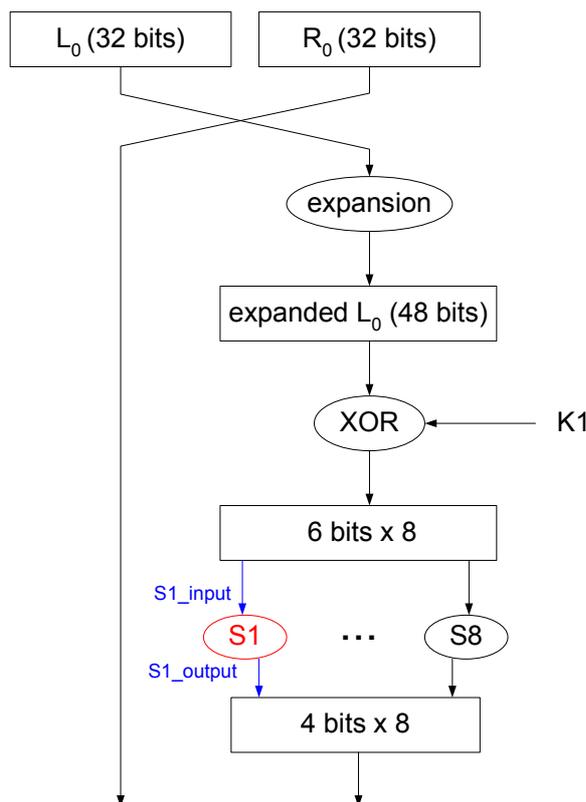


Figure 2.6: The first round of DES computation

ms can be broken successfully with 1000 timing measurements, adding a random normally distributed delay with 1 second standard deviation will make the attack require approximately $(1000ms/10ms)^2(1000) = 10^7$ samples to filter out the noise [36].

Electromagnetic Analysis Attack

Changing electrical current flowing through a conductor results in electromagnetic (EM) emissions [25]. EM energy is closely correlated to power consumption but may be localised into a smaller area. If the global current is like a river, the EM emission is then produced by streams that flow into the river. In some cases when the global power measurement becomes useless, local EM radiation may convey important information [25]. EM emissions are data-dependent just as power consumption or timing is. Attackers may extract secret information through EM analysis (EMA). The EMA attack requires the design of special probes and the development of advanced measurement methods that focus very accurately on selected points of a chip.

Some sophisticated statistical techniques such as differential electromagnetic analysis (DEMA) [25, 54, 8] can detect variations in EM emission so small that individual key bits can be identified. DEMA follows differential power analysis (DPA) becoming an important side-channel cryptanalysis attack on many cryptographic implementations, and constitutes a real threat to smart card security. More details are presented in Chapter 4.

Fault Induction Attack

In addition to simply monitoring the card, circuit activity may be externally influenced by introducing transients (“glitches”) to power or clock lines [59, 10]. This kind of threat to a smart card system is called *fault induction*. These faults may cause the processor to malfunction in a predictable and useful way for attackers. A glitch inserted on the power or clock line was a widely known fault injection technique [10]. It is non-invasive as it does not physically damage the device. There are other fault induction attacks that cause some damage to the chip, falling into the category of semi-invasive attacks which will be introduced soon. Many chips nowadays are designed to resist glitch attacks by having voltage sensors, so glitch attacks are not considered in this thesis.

2.3.2 Invasive Attacks

Unlike non-invasive attacks, an invasive attack requires the smart cards to be depackaged. Picoprobes are needed to read out the signal on buses or elsewhere in the processor [50]. These attacks tend to leave tamper evidence which limits their scope for some applications but they are most dangerous when the same keys are useful for many identical devices such as in pay TV applications. Breaking one card necessitates the revocation of the cards sharing this same revealed key. For smart card applications where each copy contains a unique key, obtaining information from one card may not help to break others, so there is no need to revoke all devices on one card’s secret disclosure. When such a fraud occurs, a solution is to identify the attacked device, cancel it and issue a new one to the true user. A simple invasive attack becomes economically unattractive under this combination of clever hardware and system design, unless it is being used to gather information for a subsequent non-invasive attack.

2.3.3 Semi-invasive Attacks

Semi-invasive attacks require some level of depackaging without going as far as an invasive attack, as it does not involve removing the passivation layer. For example, a smart card chip might be removed from its polymer packaging in order to undertake an *optical probing attack*.

Optical probing uses laser radiation with a sufficiently short wavelength (i.e., sufficient photon energy) and intensity to ionise semiconductor materials [24]. When ionisation occurs in a depletion region the production of additional carriers and the presence of an electric field (built-in field and any reverse bias) causes current to flow. This photocurrent is capable of switching the transistors whose gates are connected to the illuminated junction. This process is a transient one where normal circuit activity resumes once the light source is removed.

This transient process is similar to a glitch attack as it may cause exploitable abnormal behaviour. The abnormal behaviour could be a data error setting part of the key to a known value, or a missed conditional jump reducing the number of rounds in a block cipher. Skrobogotov et al [58] published their study of optical fault injection in 2002, which appears to be a powerful and dangerous threat to cryptographic devices. These attacks are practical as they do not require such expensive equipment as in invasive attacks, nor do they require the detailed knowledge of circuit and program structure that is needed for some non-invasive attacks. More details about optical fault induction attacks are presented in Chapter 5.

2.4 Defence Technologies

This section discusses defence technologies that can be used to improve smart card security, and how they can be evaluated through simulation.

2.4.1 Countermeasures to non-invasive attacks

Countermeasures to non-invasive attacks involve both hardware and software.

- **Software (algorithmic) methods**

- Random process interrupts [20]
- Random masking of intermediate variables [29]
- Transforming S-boxes (for symmetric cryptoalgorithm) or the curve in Elliptic Curve Cryptography (ECC) (asymmetric cryptoalgorithm) [41, 32]

- **Hardware methods**

- System-level techniques
 - * Inserting random delay
 - * Bus Encryption [12]
 - * Adding noise to obscure power or EM measurement
 - * Random register renaming [43]
 - * Self-timed circuits to remove the clock and 1-of-n encoding with a return-to-zero handshaking protocol to balance power consumption [48]
 - * Geometrically regular structure (e.g. PLA logic) to make EM emissions the same even in a tiny area [24]
- Gate-level techniques (using a standard-cell library)
 - * Balancing the Hamming weights of state transitions [48]
- Transistor-level techniques
 - * Using constant current logic (e.g. differential and dynamic logic families) [64, 62]

All of the above defences can be used in isolation or combination in system design and their effect can be evaluated by the simulation methodologies proposed in Chapters 3 and 4.

2.4.2 Countermeasures to semi-invasive and invasive attacks

Semi-invasive and invasive attacks can disrupt the normal operation of the secure devices, so they require countermeasures to detect and correct errors.

The vulnerability of cryptographic processors to optical fault injection attacks may be countered at the system level or the circuit level. System-level defences include the use of error detection and correction (EDAC) circuitry to monitor and correct errors [21]. This approach requires that extra bits of information be stored with the data to reconstruct the original data in the event of an upset. System overhead can be large, but this is sometimes

the only method available if relatively susceptible parts must be used. Another important technique is triple-modular redundancy (TMR).

Defensive techniques in combinational logic can involve redundant data paths and careful selection of circuit types. An example is the avoidance of all dynamic logic [21], because dynamic logic is highly vulnerable to optical fault injection attacks due to its highly charge-sensitive mode of operation. Security may be further improved by including small optical tamper sensors within each standard cell [24]. They force the generation of an error signal when illuminated. These sensors, constructed from one or two transistors, would normally play no part in normal circuit behaviour (only adding a small amount of capacitance). The number of sensors could be adjusted dependent upon the likelihood of the laser spot size. These defences can be evaluated by the proposed simulation methodology in Chapter 5.

Other defensive approaches include chip coating. For example, top-layer metal shielding can reflect light and help make an optical attack more difficult. Light sensors are also helpful in preventing a decapsulated chip from functioning. The effect of coating defences can not be simulated by the proposed simulation methodology. It should be evaluated by post-manufacturing test.

2.5 Summary

This chapter reviews the smart card technologies, including the structure and the applications of smart cards. The security mechanisms are also discussed, such as authentication, confidentiality, integrity and non-repudiation. Existing attack technologies are surveyed and classified into non-invasive, invasive and semi-invasive attacks, depending on the physical destruction level and temper evident level of the card. Power analysis attacks and electromagnetic analysis attacks in the class of non-invasive attacks, and optical fault induction attacks in the class of semi-invasive attacks are introduced in detail as they are the subject of Chapter 3, 4 and 5 respectively.

Chapter 3

Simulating Power Analysis Attacks

As introduced in Section 2.3.1, CMOS circuits consuming data-dependent power during an operation may leak information in the form of Hamming weight or transition count. Someone analysing this data-dependent power carefully could deduce sensitive information that a cryptographic device such as a smart card strives to protect. There are two kinds of power analysis attack: *Simple Power Analysis* (SPA) and *Differential Power Analysis* (DPA). The former primarily uses pattern matching to identify relevant power fluctuations. It helps attackers to observe macro properties of an algorithm, but it is still very difficult to pinpoint individual instructions let alone individual bits of data. DPA, on the other hand, uses statistical techniques to detect variations in power consumption so small that individual key bits can be identified. Compared to SPA, DPA is more dangerous as it does not require the attacker to know implementation details of the target code.

To keep cryptographic devices secure against power analysis attacks, a huge amount of research has been undertaken to hide or avoid the correlation between the data being manipulated and power being consumed. However, in common industrial practice, design evaluation of secure devices could only be performed after chips are manufactured. This post-manufacture analysis is time consuming, error prone and very expensive. This has driven the study of design-time security evaluation against DPA which aims to examine data-dependent power characteristics of secure processors.

3.1 DPA Simulation Methodology

Commercial power estimation tools are already widely used in integrated circuit (IC) design to provide power consumption details needed to meet power budgets and specifications, to select the proper packaging, to determine cooling requirements and estimate battery life for portable applications. For example, Synopsys® delivers a complete solution to verify power consumption at different levels of the design process. These products include: PrimePower, PowerMill®/NanoSim® and RailMill®.

Synopsys PrimePower is a dynamic, full-chip power analysis tool for complex multimillion-gate ASICs (Application-Specific ICs). PrimePower builds a detailed power profile of the design based on the circuit connectivity, the switching activity, the net capacitance and the cell-level power behaviour data in the Synopsys *.db* library. It then calculates the power behaviour for a circuit at the cell level and reports the power consumption at the chip, block, and cell levels [3].

Synopsys NanoSim is a transistor-level circuit simulation and analysis tool, with simulation speeds orders of magnitude higher than SPICE, NanoSim has the capacity for multi-million transistor designs, and SPICE-like accuracy for designs at 0.13 micron and below. NanoSim uses intelligent partitioning techniques along with a combination of event-based and time-based simulation. A typical SPICE engine treats the entire design as one monolithic block and evaluates all nodes at each time step. NanoSim, on the other hand, uses a “divide and conquer” approach where the design is automatically partitioned into smaller stages based on the channel connectivity, so that any given stage or partition is evaluated only when an input controlling node is triggered.

There are power analysis tools from other EDA (Electronic Design Automation) tool vendors. The list below is not an exhaustive inventory, but may provide an overview for those interested in DPA simulation.

- **Synopsys power solution** (www.synopsys.com)
 - RTL-level: Power Compiler, mainly for power optimisation
 - Gate-level: PrimePower, mainly for power analysis
 - Transistor-level: PowerMill/Nanosim, mainly for power analysis
- **Apache power solution** (www.apache-da.com)
 - From design to verification: RedHawk-SDL, a full-chip physical power analysis tool
- **Sequence power solution** (www.sequencedesign.com)
 - Architectural/RTL/Gate-level: PowerTheater, a comprehensive set of power analysis tools

3.1.1 Simulation Procedure

Using the tools above aids designers to perform power analysis at various levels in the design process. In the DPA simulation approach, I use some of these tools to obtain accurate power consumption of a design under test. But these are not sufficient. In reality, the on-chip capacitance between the power and ground network, (*e.g.* parasitic capacitance [30] and intentionally added decoupling capacitors) and the package inductance play an important role in power consumption waveforms. This so-called *power grid effect* should be taken into account in simulating a power analysis attack in order to make the results realistic.

The procedure to perform a DPA simulation on a chip design was introduced in [14] and is shown in Figure 3.1. The power analysis simulation can be performed at either the gate level or the transistor level; either before layout, or post layout to extract parasitics of the circuit for more accurate simulation. The global core current consumption $I_{dd}(t)$ is collected through the functional/power simulation. Then the data $I_{dd}(t)$ is processed through MATLAB™ programs developed by the author to implement differential power analysis. Unsatisfactory results may mean re-design or re-layout of critical blocks or even the whole system.

Figure 3.2 zooms in the functional/power simulation block and the DPA simulation block. Two sets of current consumption data are collected during the processor under test is computing with different operands. This is to mimic a DPA attack where a number of random

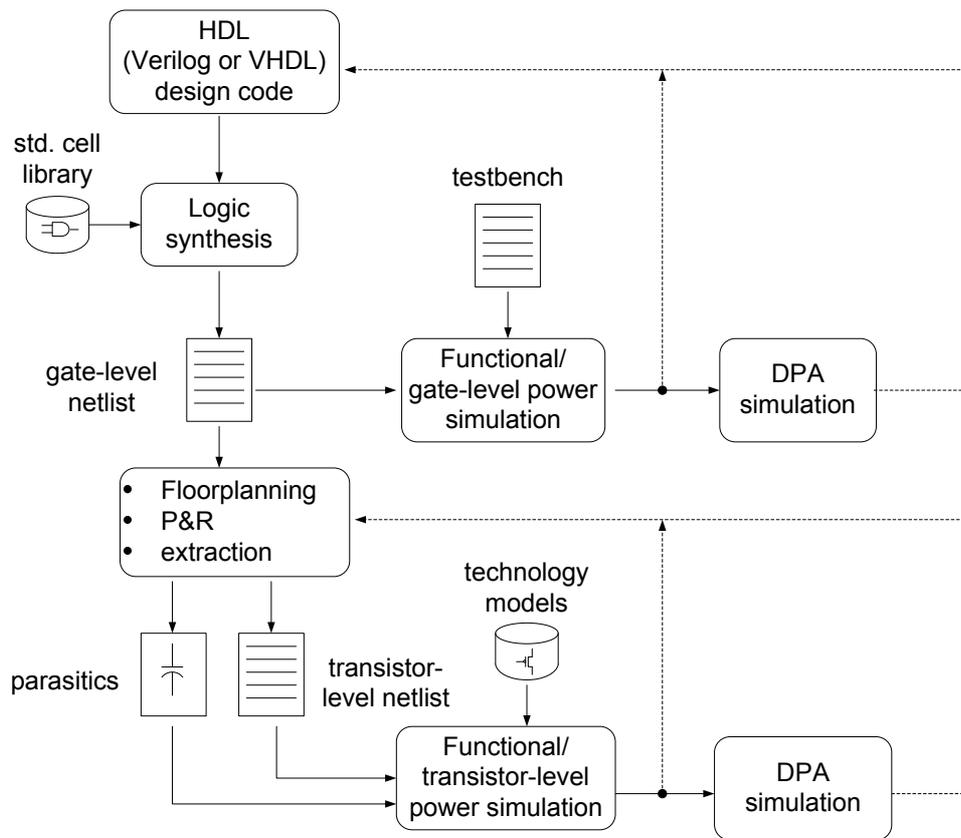


Figure 3.1: Digital IC design flow with DPA simulation

plaintext inputs are encrypted with a key. With a guessed subkey, the power traces are partitioned into two groups and averaged. Only repeatedly executing “1” (or “0”) in some fixed points in time during the computation causes power difference not to be smoothed out. Therefore, the two averaged power traces of each partition ultimately reveal a data dependency of the processor operations. With two runs with different operands, this simulation methodology will be able to examine data-dependent power characteristics of secure processor designs, which are the fundamental weakness a real DPA attack exploits.

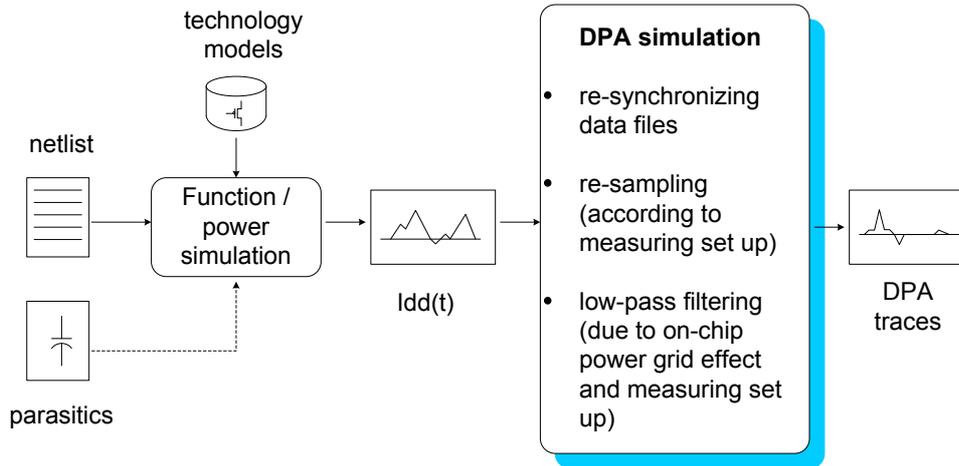


Figure 3.2: Simulating power analysis

Once the two sets of current data $I_{dd}(t)$ are collected, they are passed to MATLAB™ programs to implement the DPA simulation, as illustrated in Figure 3.2. The DPA simulation is mainly processing of $I_{dd}(t)$ data, involving:

- re-synchronising two sets of data for ‘differential’ analysis
- re-sampling the data according to the measurement setup. This step is optional. If the simulation time step is unnecessarily small (for example $1ps$ compared to nanosecond scale of normal measurement sampling frequency), then the data can be decimated for faster simulation speed.
- low-pass filtering the data, considering the load resistance of the measurement instrument and on-chip parasitic capacitance, inductance etc. More detail is presented in the next subsection.

Finally, DPA is performed by subtracting one power trace from another. Security weakness will be manifested as pulses in the DPA trace, revealing data-dependent power characteristics of the design under test.

LC Resonance Effect

In Figure 3.2, the current data $I_{dd}(t)$ obtained from power estimation tools involves only the core circuitry, which is shown in the dashed box in Figure 3.3. It considers neither on-chip parasitics, such as power grid capacitance ($C_{powergrid}$) and on-chip decoupling capacitance ($C_{decoupling}$), nor the package inductance ($L_{package}$). In measurement, they all count and should be considered in the simulation methodology.

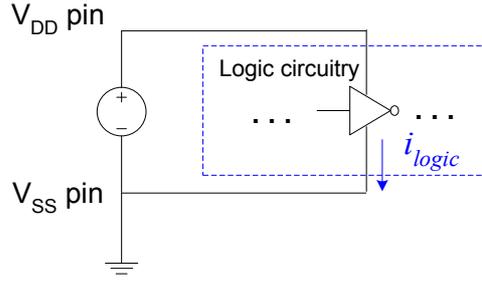


Figure 3.3: Circuit simulation of logic circuitry only

Also, normally in power measurement, a small resistor R_1 (around 20Ω) is added between the ground pin (V_{SS}) and the true ground. Current flowing through R_1 creates a time varying voltage v_{scope} that can be sampled by an oscilloscope. A model including on-chip parasitics, package inductance and measuring resistance is shown in Figure 3.4.

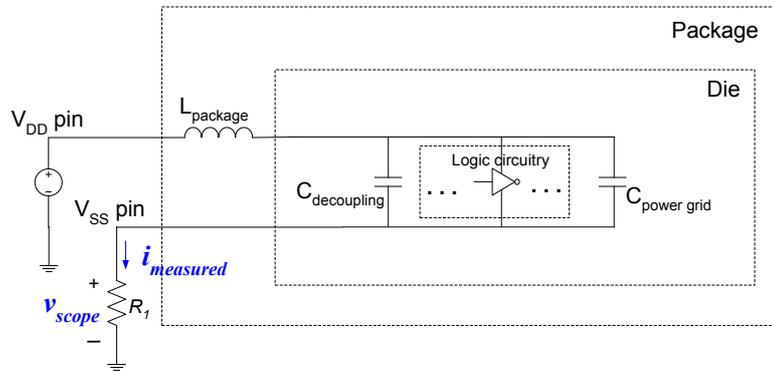


Figure 3.4: Measuring power consumption of a chip with on-chip parasitics and package inductance

Transforming the circuit into a Norton equivalent structure and replacing the current source with i_{logic} obtained from logic circuitry power simulation (such as shown in Figure 3.3), we get Figure 3.5 where the on-chip capacitance $C_{onchip} = C_{powergrid} + C_{decoupling}$, and $C_{powergrid}$ is derived from [30] as the lumped capacitor between the power and ground network.

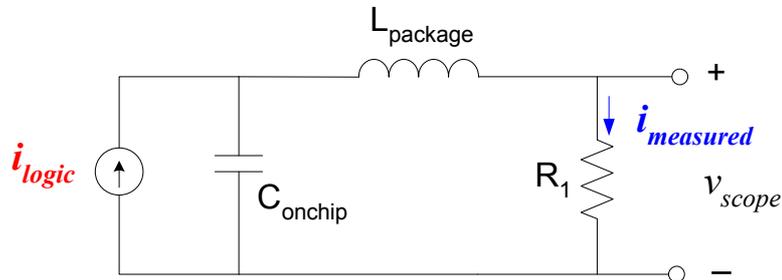


Figure 3.5: RLC low-pass filter for input current i_{logic} obtained from logic circuitry power simulation

The RLC circuit in Figure 3.5 forms a low-pass filter for input current i_{logic} , with the 3dB cutoff frequency¹ of output current $i_{measured}$ at $f_{cutoff} = 1/2\pi\sqrt{LC}$.

¹the frequency at which the output current is 70.7% of the input current.

Take the Springbank test chip as an example. This chip was fabricated in the UMC 0.18 μm 6-layer metal process as part of the G3Card project [19, 24]. The chip is packaged in PGA120 (Pin Grid Array 120 pin) and mounted in a ZIF (Zero-insertion Force) socket on the evaluation board. The package inductance ($L_{package}$), here including bond wire inductance, trace inductance, pin inductance and socket inductance, is about 10nH. Power-grid capacitance and on-chip capacitance is about 400pF. The 3dB cutoff frequency f_{cutoff} is calculated to be 79.6MHz, and this is used for the simulation later.

3.2 Results

DPA simulation has been carried out on the Springbank test chip. Figure 3.6 shows a picture of the test chip which contains five 16-bit microprocessors with different design styles. This experiment addresses the dual-rail asynchronous processor (DR-XAP) only (in the middle of the chip).

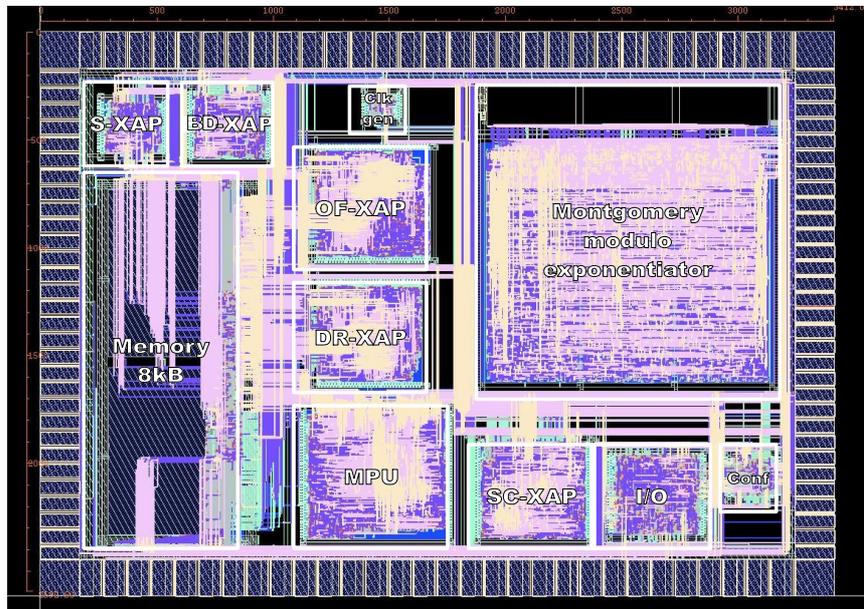


Figure 3.6: Springbank test chip showing the microprocessor (DR-XAP) in the middle is under DPA test

I target simple instructions (*e.g.* XOR (exclusive OR), shift, load, store etc) which can give a good indication of how the hardware reacts to operations of cryptographic algorithms. A short instruction program runs twice with operands of different Hamming weight. The first run computes $\#H'11$ XOR $\#H'22$, while the second computes $\#H'33$ XOR $\#H'55$. Figure 3.7 shows a fragment of the instruction program.

3.2.1 Simulation Result

Synopsys PrimePower is used to collect the current data, and the preliminary result (without considering filtering effect from power-grid and package inductance) is presented in Figure 3.8: the upper curves are the two superposed power consumption traces and the lower one is their differential trace.

```

main:  ld      x, #H'FFF0          ; initialise stack
      ld      al, #H'0011       ; load the 2 operands for first run
      st      al, @(0,x)
      ld      al, #H'0022

loop:  nop                      ; 5 'no-operation' constructions to
      nop                      ; ease synchronisation in measurement
      nop
      nop
      nop
      xor     al, @(0,x)        ; construction to be analysed
                                   ; On first run: #H'11 xor #H'22
                                   ; On second run: #H'33 xor #H'55

      nop
      nop
      nop
      nop
      nop
      ld     al, #H'0033       ; load the 2 operands for second run
      st     al, @(0,x)
      ld     al, #H'0055

      nop
      nop
      bra    loop             ; loop for calculation with the 2nd
                                   ; set of operands

```

Figure 3.7: Fragment of the instruction program used for the DPA evaluation

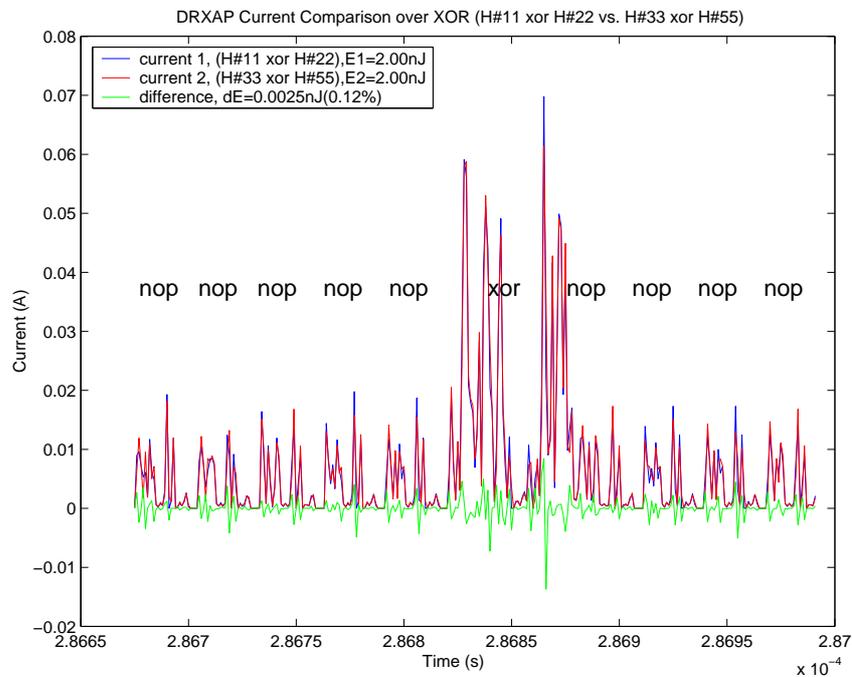


Figure 3.8: Power Simulation: DR-XAP executing XOR

Then I perform a second order low-pass filtering on the original power curves, as described in the previous section. Figure 3.9 demonstrates the filtered power traces and their differential trace.

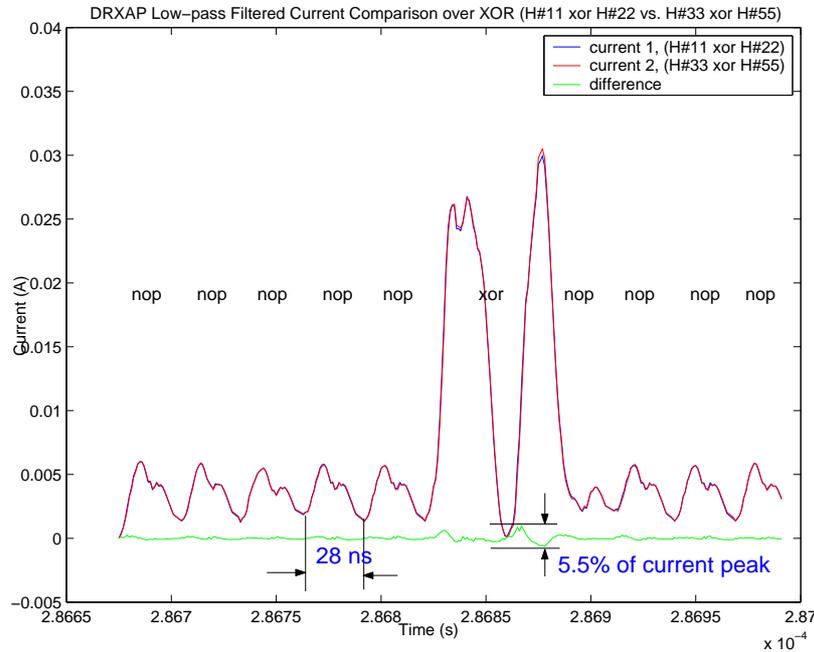


Figure 3.9: Power Simulation: DR-XAP executing XOR, low pass filter applied

It takes about 3 minutes to run the power simulation with Synopsys PrimePower over the 10,000 gates of the processor DR_XAP. The data processing with MATLAB takes about 2 minutes. All the simulation work is done on a 1.6 GHz AMD Athlon processor with 2 GB memory.

3.2.2 Measurement Result

To achieve a side-by-side comparison, the processor DR-XAP is measured by NDS® against DPA with the same instruction program². The same instruction program runs twice, computing #H'11 XOR #H'22 in the first run, and #H'33 XOR #H'55 in the second. Figure 3.10 shows the results of collecting power traces for each operation, averaging the traces over 4000 runs, and then subtracting one averaged trace from the other. The upper curves are two superposed power traces; the centre curve represents their difference, of which the small disturbance at left of centre is the result of data-dependent differences for the two XOR operations. The lowest curve is an I/O signal used to trigger the oscilloscope.

Comparing Figure 3.9 to Figure 3.10, we see how the filtered simulated power traces match with the measurement. In Figure 3.9, the NOP operation timing is 28ns, close to the measured 25ns, verifying the LC resonance effect and its calculated 3dB cutoff frequency f_{cutoff} used in the simulation. The differential peak is 5.5% of the XOR operation in the simulated result, while in the measurement it is 1.25%. This is expected as our simulation does not

²This evaluation is also part of the G3Card project [19, 24].

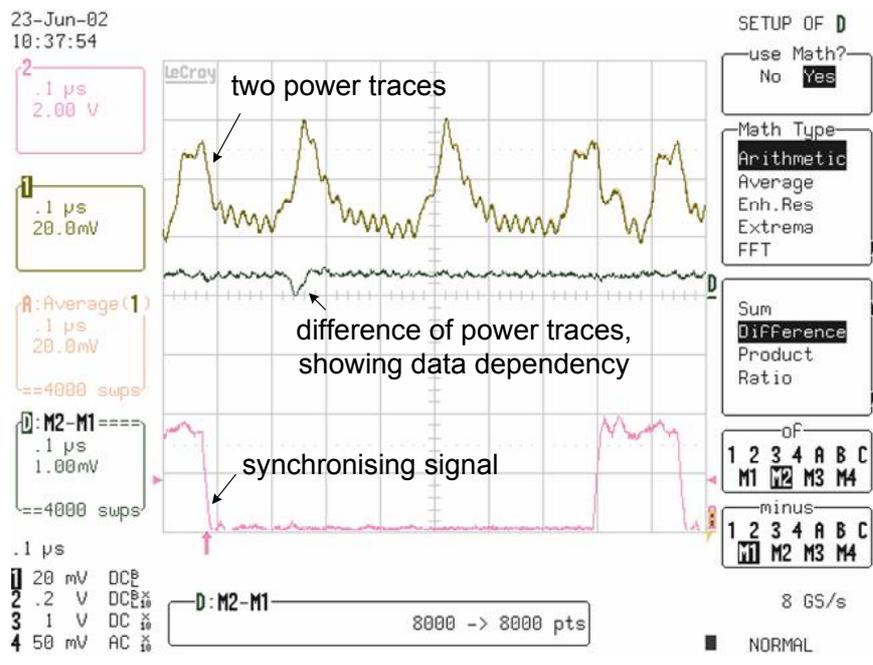


Figure 3.10: Differential Power Analysis of the DR-XAP processor on the Springbank chip (experimental graph)

cover the power used by memory accesses – we had no memory power model available. This in turn raises the ratio of differential power to operation power. The upper power curves for the XOR operations differ in shape from those measured also caused by no memory accessing power. This produces a significant drop in power simulation at the point where one operand of the XOR operations is fetched from memory.

Using caches can reduce the number of power-hungry memory fetches. However, frequent cache misses, *e.g.*, when there are many *different* data referenced by the S-box in a cypher, take longer encryption time [63]. Obtaining key differences by observing the encryption time can reduce the key search space. This so-called cache attack requires careful use of caches in more complex processors.

3.3 Summary

This chapter presents a simulation methodology for differential power analysis (DPA) of secure processors. This simulation methodology includes power simulation of the logic circuitry and low-pass filtering to mimic the effects caused by on-chip parasitics and package inductance. Comparison between the simulation result and measurement result on our Springbank test chip has demonstrated reasonable agreement, thus indicating the validity of the proposed DPA simulation methodology.

Chapter 4

Simulating EMA Attacks

As introduced in Chapter 2, cryptographic devices could be broken through analysing electromagnetic radiation [54, 8, 25] during computation so as to extract information about the secret key. Like the DPA attacks described in Chapter 3, differential electromagnetic analysis (DEMA) attacks deploy similar sophisticated statistical techniques in order to detect variations in EM emission so small that individual key bits can be identified.

DEMA followed DPA in posing a real threat to smart card security. A serious research effort has been made to counter the DEMA attacks. These countermeasures generally endeavour to hide or avoid the correlation between the data being manipulated and the EM side-channel information. To evaluate these techniques, I propose design-time security evaluation of their effectiveness against EMA attack. This aims to examine data-dependent EM characteristics of secure processors, so as to assess their security level against EM side-channel analysis attacks.

4.1 Background

4.1.1 Origin of EM Emissions

To comprehend the origin of electromagnetic (EM) emissions, we must know Maxwell's Equations. The four equations form a complete description of electric and magnetic fields and their interaction. I give only a brief description here. The first equation (4.1) is Gauss's law for electricity, which says that electric field diverges from electric charge. The second (4.2) is Gauss' law for magnetism, which says there are no isolated magnetic poles. The third equation (4.3) is Faraday's law of induction, which says that electric fields are produced by changing magnetic fields. The last one (4.4) is Ampere's law, which says that circulating magnetic fields are produced by changing electric fields and by displacement currents in the dielectric.

$$\epsilon_0 \oint \mathbf{E} \cdot d\mathbf{S} = q \quad (4.1)$$

$$\oint \mathbf{B} \cdot d\mathbf{S} = 0 \quad (4.2)$$

$$\oint \mathbf{E} \cdot d\mathbf{l} = -\frac{d\Phi_B}{dt} \quad (4.3)$$

$$\oint \mathbf{B} \cdot d\mathbf{l} = \mu_0 \left(\epsilon_0 \epsilon_r \frac{d\Phi_E}{dt} + i \right) \quad (4.4)$$

Where,

\mathbf{E} = Electric Field Strength, V/m^2

\mathbf{B} = Magnetic Flux Density, Tesla or $N/A \cdot m$

$\epsilon_0 = 8.85418782 \times 10^{-12} \frac{F}{m}$, Permittivity of a vacuum

ϵ_r , Relative permittivity, the ratio of permittivity of a dielectric relative to that of a vacuum

$\mu_0 = 4\pi \times 10^{-7} \frac{H}{m}$, Permeability of a vacuum

Maxwell's Equations explain the origin of EM radiation: waves of interrelated changing electric and magnetic fields propagate through space. Referring to the third and fourth equations, we know that in an integrated circuit, it is the changing current flowing in a closed loop that produces a changing magnetic field which in turn produces a changing electric field.

4.1.2 Near and Far Fields

Circuits that cause fields can be sorted into four basic classes [61]:

- Electrostatic
- Magnetostatic
- Electric, time-variant
- Magnetic, time-variant

Electrostatic circuits are simply fixed distribution of charges. A simple case is the charge dipole, where two equal and opposite charges are spaced some distance apart. There is an electric field which does not vary with time (*i.e.*, \mathbf{E} is constant in time), but no magnetic field (*i.e.*, \mathbf{H} is zero). Magnetostatic circuits consist of DC current loops. This is the dual of the electrostatic case. There is a constant magnetic field \mathbf{H} which falls off with the cube of distance, but no electric field (*i.e.*, \mathbf{E} is zero). For both the electrostatic and magnetostatic cases, there is no wave, so field information does not propagate.

Time-variant Electric Circuit

A time-variant electric circuit, for example a dipole driven by an AC (Alternating Current) voltage source, has positive and negative charge at the open ends which reverse harmonically. The movement of electric charge q forms a displacement current I ($I = dq/dt$), which generates an electric and magnetic field. In spherical coordinates, as shown in Figure 4.1, the magnetic field generated by the displacement current I is:

$$\mathbf{H}_\Phi = \frac{Il}{4\pi r^2} (1 + j\beta r) e^{-j\beta r} \sin \theta \vec{\Phi} \quad (4.5)$$

where $\vec{\Phi}$ represents the vector direction, β denotes a constant of $2\pi/\lambda$ where λ is the wavelength, r denotes the distance from the source.

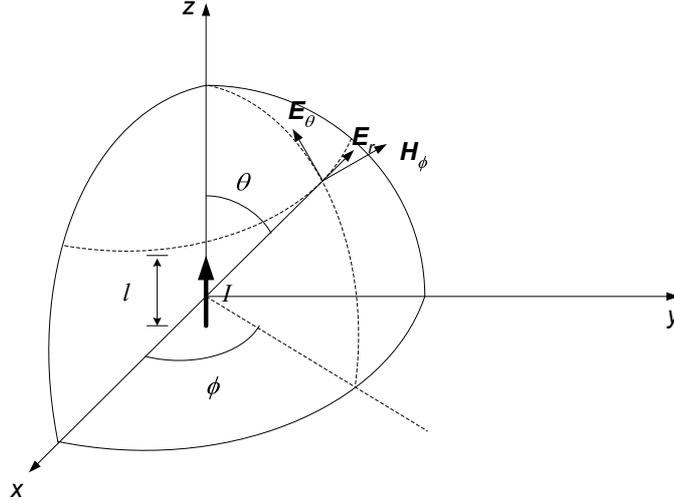


Figure 4.1: A dipole produces electric and magnetic fields.

The electric field is derived using Maxwell's equations as:

$$\mathbf{E} = \frac{1}{j\omega\epsilon} \nabla \times \mathbf{H} \quad (4.6)$$

$$= \frac{Ile^{-j\beta r}}{j\omega\epsilon 4\pi r^3} [2 \cos\theta(1 + j\beta r)\vec{r} + \sin\theta(1 + j\beta r - \beta^2 r^2)\vec{\theta}] \quad (4.7)$$

where \vec{r} and $\vec{\theta}$ represent that the electric field \mathbf{E} has two components along the r and θ direction in spherical coordinates.

Let us consider approximations for the electric and magnetic fields in near and far fields as βr changes:

- **Case I. Near Field** When $\beta r \ll 1$, i.e. $r \ll \lambda/2\pi$,

$$\mathbf{E} \cong \frac{Ile^{-j\beta r}}{j\omega\epsilon 4\pi r^3} [2 \cos\theta\vec{r} + \sin\theta\vec{\theta}] \quad (4.8)$$

$$\mathbf{H} \cong \frac{Ile^{-j\beta r}}{4\pi r^2} \sin\theta\vec{\Phi} \quad (4.9)$$

$\mathbf{H} \ll \mathbf{E}$, electric field dominates.

- **Case II. Far Field** When $\beta r \gg 1$, i.e. $r \gg \lambda/2\pi$,

$$\mathbf{E} \cong j\omega\mu \frac{Ile^{-j\beta r}}{4\pi r} \sin\theta\vec{\theta} \quad (4.10)$$

$$\mathbf{H} \cong j\beta \frac{Ile^{-j\beta r}}{4\pi r} \sin\theta\vec{\Phi} \quad (4.11)$$

Note that $\frac{E_\theta}{H_\Phi} = \frac{\omega\mu}{\beta} = \sqrt{\frac{\mu}{\epsilon}}$. \mathbf{E} and \mathbf{H} are orthogonal to each other and are both orthogonal to the direction of propagation. The relative strength of the electric and magnetic field is fixed, which is defined as the wave impedance. Electric and magnetic fields are jointly referred to as electromagnetic field in far field.

Time-variant Magnetic Circuit

Circuits can also generate time-variant magnetic emissions which are the dual of circuits generating time-variant electric emissions. A current loop excited by an AC source carrying current I generates electric and magnetic fields. In spherical coordinates as shown in Figure 4.2, the magnetic and electric fields generated by the current loop mirror those for the dipole:

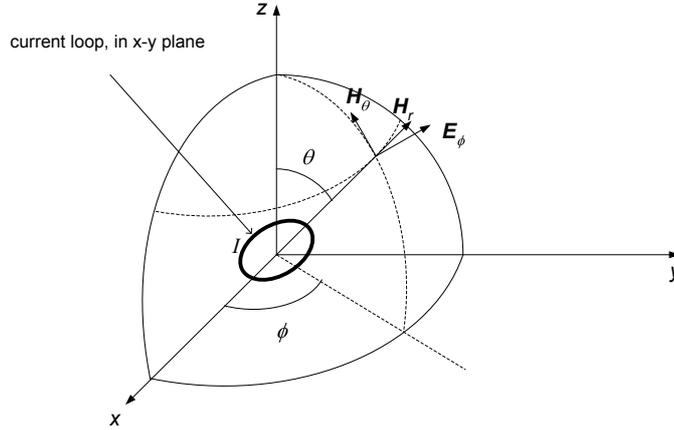


Figure 4.2: A current loop produces magnetic and electric fields.

$$\mathbf{H} = \frac{IAe^{-j\beta r}}{4\pi r^3} [2 \cos \theta (1 + j\beta r) \vec{r} + \sin \theta (1 + j\beta r - \beta^2 r^2) \vec{\theta}] \quad (4.12)$$

$$\mathbf{E} = \frac{IAe^{-j\beta r} \beta}{j\omega \mu 4\pi r^2} (1 + j\beta r) \sin \theta \vec{\Phi} \quad (4.13)$$

Let us consider approximations for the electric and magnetic fields in near and far fields as βr changes:

- **Case I. Near Field** When $\beta r \ll 1$, i.e. $r \ll \lambda/2\pi$,

$$\mathbf{H} \cong \frac{IAe^{-j\beta r}}{4\pi r^3} [2 \cos \theta \vec{r} + \sin \theta \vec{\theta}] \quad (4.14)$$

$$\mathbf{E} \cong \frac{IAe^{-j\beta r} \beta}{j\omega \mu 4\pi r} \sin \theta \vec{\Phi} \quad (4.15)$$

$\mathbf{H} \gg \mathbf{E}$, magnetic field dominates.

- **Case II. Far Field** When $\beta r \gg 1$, i.e. $r \gg \lambda/2\pi$,

$$\mathbf{H} \cong -\frac{IAe^{-j\beta r} \beta^2}{4\pi r} \sin \theta \vec{\theta} \quad (4.16)$$

$$\mathbf{E} \cong \frac{IAe^{-j\beta r} \beta^2}{\omega \mu 4\pi r} \sin \theta \vec{\Phi} \quad (4.17)$$

\mathbf{E} and \mathbf{H} are orthogonal to each other and are both orthogonal to the direction of propagation. They are now together referred to as an electromagnetic field.

From the above description, EM radiation is determined by two things:

- The source – whether it is open ended (dipole) or closed (current loop). If the source is a current loop, which is applied in an IC circuit, measuring \mathbf{H} in near field is more efficient than measuring \mathbf{E} .
- The measurement distance – in the near field or far field.

However in each case, the measured element (\mathbf{E} or \mathbf{H}) is proportional to current I . This is the fundamental reason why current is used to represent EM field (in some cases, the rate of change of current is used and the reason will be explained in next section).

4.1.3 Direct vs Modulated EM emissions

Section 4.1.2 discussed \mathbf{E} or \mathbf{H} elements which are referred to as **direct emissions** in that the emissions are caused directly by current flow with sharp rising/falling edges. To measure direct emissions from a signal source isolated from interference from other signal sources, one uses tiny field probes positioned very close to the signal source and uses special filters to minimise interference. To get good results may require decapsulating the chip.

Modulated emissions occur when a data signal modulates carrier signals which then generate EM emissions propagating into space. A strong source of carrier signals is a harmonic-rich square-wave signal such as a clock, which may then be modulated in amplitude, phase or some other manner. The recovery of the data signals requires a receiver tuned to the carrier frequency with a corresponding demodulator.

Exploiting modulated emissions can be easier and more effective than working with direct emission [8]. Some modulated carriers could have substantially better propagation than direct emission, which may sometimes be overwhelmed by noise. The modulated emission sensing may not require any intrusive/invasive techniques or fine grained positioning of probes.

4.1.4 EM field Measurement Equipment

A number of sensors can be used to detect the EM signals in EMA attacks. They are divided into those detecting electric and those detecting magnetic fields in near-field¹ or far-field. In EM analysis attacks on small devices with weak EM emissions such as a smart card, near-field sensors are more appropriate.

Near-field Electric Field Sensors

An example of a **near-field electric field sensor** is a monopole antenna. It generally measures the near-field electric component around current carrying conductors where $\mathbf{E} \propto I$.

Near-field Magnetic Field Sensors

Near-field magnetic field sensors generally measure the near-field magnetic component around current carrying conductors where $\mathbf{B} \propto I$.

¹Near-field refers to a distance within one sixth of the wavelength from the source ($r < \lambda/2\pi$), while far-field refers to a distance beyond it ($r > \lambda/2\pi$).

- Magnetic loop (also referred to as inductive loop)

The simplest magnetic field sensor is a loop of wire. Hard disk drive write heads are mainly inductive loops too. An EM field is induced in the loop due to a change in magnetic flux through the loop caused by a changing magnetic field produced by an AC current-carrying conductor. This is the transformer effect. The induced voltage is:

$$V = - \int_S \frac{\partial \mathbf{B}}{\partial t} \cdot d\mathbf{s} \quad (4.18)$$

over surface S using area element $d\mathbf{s}$. Let us rewrite it into the following equation, which says the measurement output is proportional to the rate of change of the current which causes the magnetic field.

$$V = M \frac{dI}{dt} \quad (4.19)$$

where M denotes the mutual inductance between the sensor and the concerned circuit.

Inductive sensors sense the change of magnetic flux, so I use the rate of change of the current dI/dt to track EM emission. Simulation for this type of sensor involves differential calculation on current consumption data.

- Magnetoresistive sensors

These are used in hard disk drives for reading and are made of materials that have resistance linear to the magnetic field (\mathbf{H}) [53]. The magnetoresistive probe output is proportional to the magnitude of the field, rather than the rate of change of the magnetic field such as in inductive probes.

- Hall probe

A Hall probe works by way of the Hall effect. Any charged particle moving perpendicular to a magnetic field will have a Lorentz force upon it, given by $F = q(\mathbf{v} \times \mathbf{B})$. However the moving electrons accumulate an electric field which gives the electrons an electric force in the other direction by $F = q\mathbf{E}$, where $\mathbf{E} = V_{measured}/d$. Thus, $V_{measured} \propto B$. The detectable field range of Hall-effect sensors are above 10 gauss [17], too large to discern EM emanation from a chip through ambient noise.

There are also far-field electromagnetic field sensors such as log-periodic antennas. They generally measure far-field electromagnetic field and often work with other equipment to harness modulated emissions. For example, an AM receiver tuned to a clock harmonic can perform amplitude demodulation and extract useful information leakage from electronic devices [8].

This is not an exhaustive list of field sensors, but illustrates that different types of sensors measure different types of field, so different approaches are required to conduct EM simulations.

4.2 Simulation Methodology for EM Analysis

4.2.1 System Partitioning

The most straightforward way to simulate EM waves propagating in a circuit is to use a 3D or planar EM simulator, which involves solving Maxwell's equations for the electric and magnetic vector fields in either the frequency or time domain. However a full-wave 3D simulator incorporating characterised nonlinear² semiconductor devices is too time consuming to be practical for chip-level analysis.

Our simulation approach is to partition an electronic system into two parts. The first part is the chip, simulated in **circuit simulators** like SPICE, which is fundamentally flawed because wave coupling is not accurately represented even if transmission lines are used for the interconnects. However, the chip dimensions are small enough (compared to the wavelength) to tolerate the errors³. The second part is the package and even the printed circuit board (PCB), which can be accurately simulated by a (3D or planar) **EM simulator** and be modelled with lumped components (R, L and C). The lumped elements will then be incorporated into the same circuit simulator to achieve the response of the entire system.

4.2.2 Simulation Procedure

The procedure to perform an EMA simulation on a chip design is shown in Figure 4.3. The EM analysis simulation flow is similar to that of power analysis which measures the global current of a device [14] (see Chapter 3). However EM analysis may focus on a smaller block such as the ALU or the memory. In this case, a Verilog/SPICE co-simulation can be used where the partitioning function provides an easy means to select the desired block(s) to test. With Verilog/SPICE co-simulation, various instructions are easily executed and modified through testbench files written in Verilog. Accurate simulation of current consumption is achieved in the SPICE-like simulation. Once the current data $I(t)$ for the desired block(s) or a whole processor is collected, it is passed to MATLABTM and is processed to implement DEMA according to the sensor types and emission types.

The data processing procedure for EM analysis is shown in the shadowed box in Figure 4.4. It includes synchronising two sets of current consumption data when the processor under test is computing with different operands. I perform signal processing on each set of current consumption data, for example, using differential calculation, if wish to simulate emission sensed by an inductive sensor, or using amplitude demodulation to simulate amplitude modulated EM emissions.

²Some examples of nonlinear components are Diodes, BJTs and MOSFETs.

³The velocity of electromagnetic propagation is limited by the laws of nature, and in silicon-dioxide it is approximately 1.5×10^8 m/s . The rule of thumb is that we usually need to consider the transmission-line effect when the edge length is shorter than three times the longest dimension of a device. Fast signal edges in smart card chips with an edge rate of under 1ns have to be considered as “high speed” only when the longest chip dimension is beyond 50mm. Smart card chips are typically $< 5mm$, so wires are never longer than 10mm, but even this is unlikely.

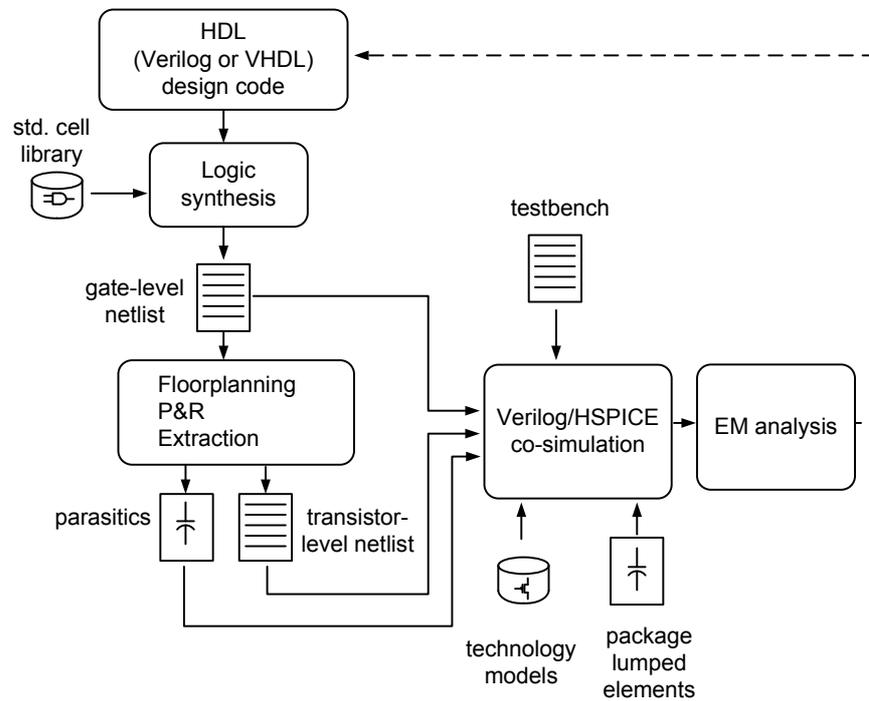


Figure 4.3: Digital design flow with EM analysis

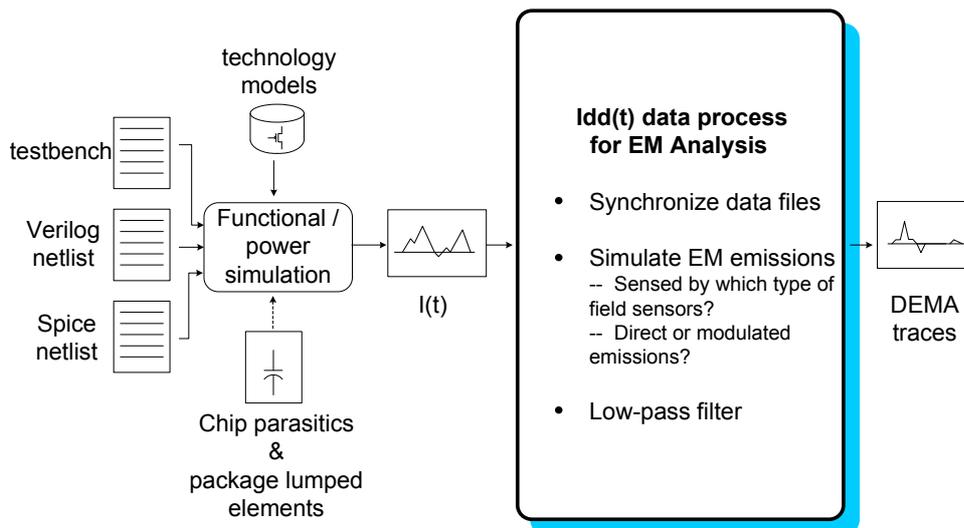


Figure 4.4: EM analysis simulation procedure

Low-pass Filtering Effect of EM Sensors

Since the EM sensors low-pass filter the EMA traces, the two sets of processed current consumption data have to be low-pass filtered at the end of the EMA data processing procedure. Considering the inductance in inductive sensors, and the load resistance from connected instruments (e.g. an amplifier or an oscilloscope), an RL low-pass filter is formed as shown in Figure 4.5. Its 3dB cutoff⁴ frequency is $f_{cutoff} = R/2\pi L$.

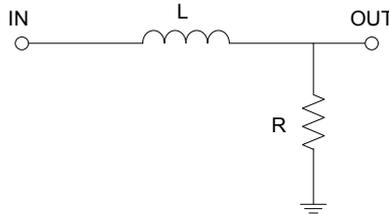


Figure 4.5: RL low-pass filter

Finally, DEMA is performed by subtracting one EMA trace from another. Security weaknesses will manifest as pulses in the DEMA trace, revealing data-dependent EM characteristics of the design under test. The term DEMA here refers to the variation (difference) in the EM emissions, instead of statistical treatment correlating the variation to hypothetical data being manipulated as in a real DEMA attack [54]. This is because the proposed methodology is to evaluate data-dependent EM characteristics of secure processor designs, which are the fundamental weakness a real DEMA attack exploits and can be identified with deterministic data.

4.3 Evaluation Results

4.3.1 EM Simulation Setup

DEMA simulation has been carried out on the Springbank test chip as shown in Figure 3.6. This evaluation addresses the synchronous processor (S-XAP) on the top left corner and the dual-rail asynchronous processor (DR-XAP) in the middle.

The aim of the test is to examine the data-dependent EM characteristics of the processors. I target simple instructions (e.g. XOR, shift, load, store etc) which can give a good indication of how the hardware reacts to the operations used in cryptographic algorithms. A short program runs twice with operands of different Hamming weight. The first run sets the I/O trigger port high by storing '1' into memory, computes '00 XOR 55', and sets the I/O trigger port low by storing '0' into memory, while the second run is identical except the computation is '55 XOR 55'.

The current collected in the simulation is the globe current $I_{dd}(t)$, since it is aimed to compare with the measurement result demonstrated later, where a sensor with large enough size covering the entire processor is used. Using the globe current $I_{dd}(t)$ implicates the approximation that the magnetic field produced by individual current paths within the processor is represented by that produced by the combined current. This approximation assumes the

⁴The frequency at which the output voltage is 70.7% of the input voltage.

distances between individual current paths are much shorter than the distance from the circuit to the sensor. The approximation also neglects the effect of different orientations of branch currents, assuming they are flowing in parallel and the produced field are added as scalars rather than vectors. This approximation may result in quantitative magnitude difference from the real emission, but it is effective in simulating differential analysis where the qualitative difference is crucial.

4.3.2 EM Simulation of a Synchronous Processor

Figure 4.6 shows the EMA simulation over the S-XAP processor. I simulate direct EM emission picked up by an inductive sensor. On the graph I plot the EM traces of the processor for ‘00 XOR 55’ and ‘55 XOR 55’, as well as the differential EM plot of EMA1 - EMA2 (DEMA). The EM traces (EMA1 and EMA2) are superposed and appear as the top trace in Figure 4.6. The differential EM trace (DEMA) is shifted down from the centre by 6×10^5 unit to clearly show its relative magnitude. The EM emission magnitude is computed through dI/dt as discussed in Section 2.3, thus has units of A/s.

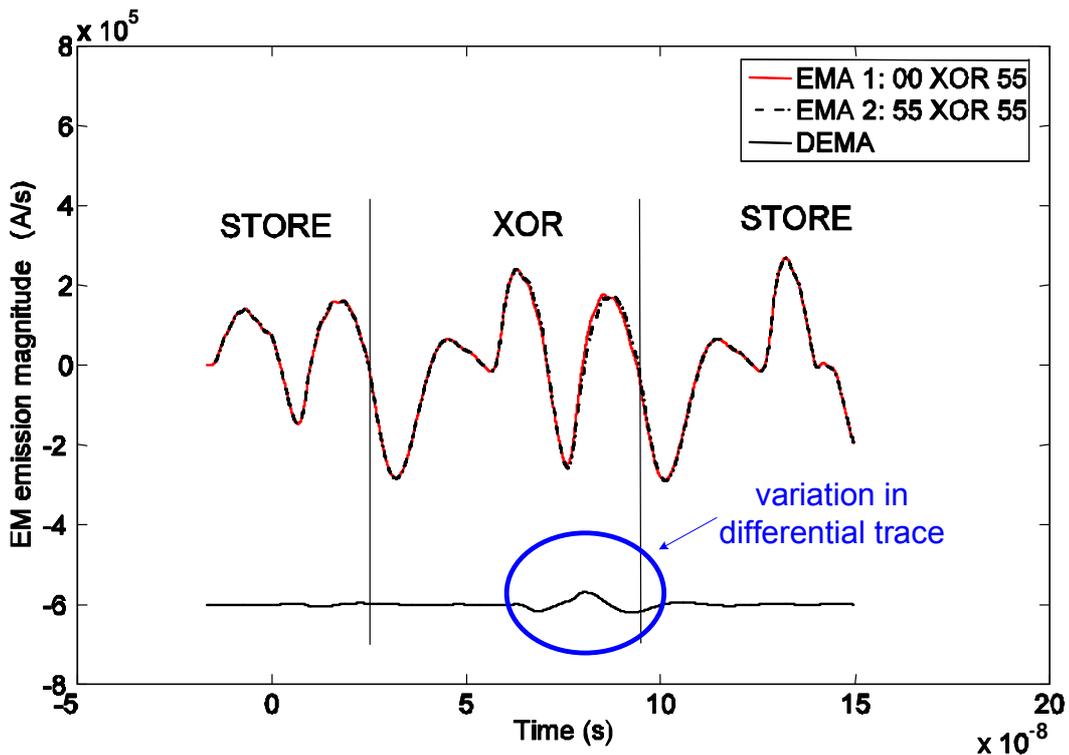


Figure 4.6: EMA simulation over the S-XAP processor executing XOR with different operands

It takes about 5 minutes to run the HDL/SPICE co-simulation to collect current consumption data, with 14,000 transistors simulated in Synopsys NanoSim™ and the rest tens of logic gates simulated in Synopsys VCS™. The small number of logic gates are mainly the interface to the memory shared by the 5 processors. In the VCS/NanoSim co-simulation these logic gates act as the required top-level module in Verilog. The data processing with MATLAB takes about 2 hours, mainly to align two sets of data through interpolation. All the simulation work is done on a 1.6 GHz AMD Athlon processor with 2 GB memory.

The measurement (done by Theodore Marketos in Computer Laboratory) of EM emissions on the same processor performing the same code is shown in Figure 4.7. The EM emissions are picked up by an inductive sensor over 5000 runs to average out the ambient noise (although 200 runs are enough), then monitored on an oscilloscope. The inductive head in use has resistance $R = 5.42\Omega$, inductance $L = 9.16\mu\text{H}$. When delivering power into a $4\text{K}\Omega$ load, the 3dB cutoff is calculated as 70MHz. The measurement results demonstrate the EM traces are around 50MHz, complying to the explanation of the RL low-pass filtering effect in Section 3.2, and the parameters have been used in the EMA simulation shown in Figure 4.6.

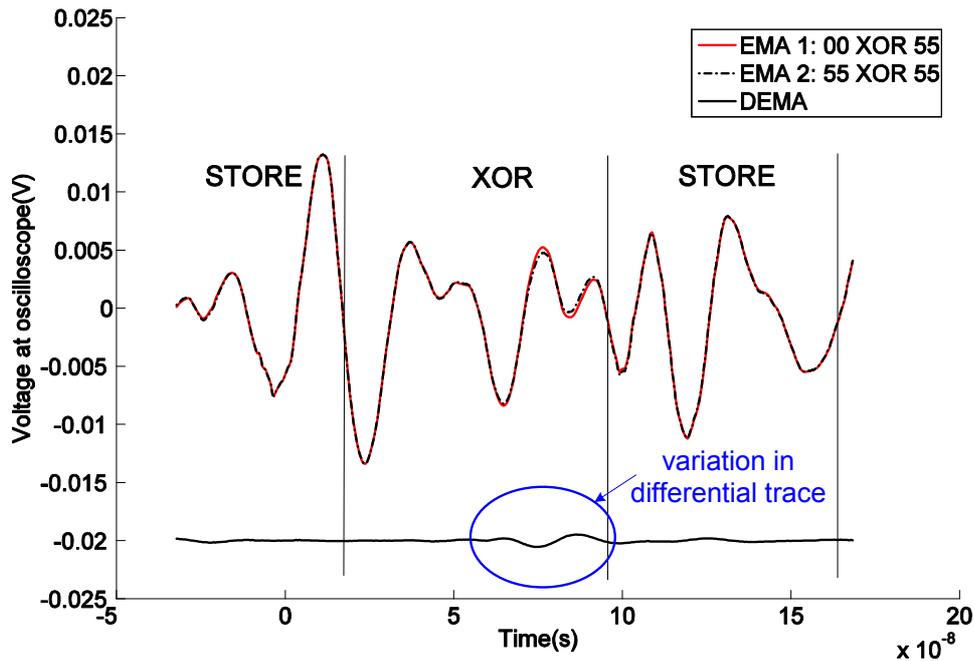


Figure 4.7: EMA measurement over the S-XAP processor executing XOR with different operands (experimental graph)

Both the measurement and the simulation results observe the differential trace peaks when the processor is executing XOR logic operations. This means data dependent EM emission is leaking information related to key bits then, which means vulnerability to EMA attacks. The agreement between the measurement and the simulation results confirms the validity of the proposed EMA simulation approach. The simulated EM traces in Figure 4.6 are lower in shape compared to those measured around the circled places, as the simulation includes no power contribution from memory accesses.

To compare the DPA attack and the DEMA attack, Figure 4.8 demonstrates DPA measurement over S-XAP processor performing the same code. Although we did only 4 measurement runs to average out noise, data dependent power consumption can clearly identify when the processor is executing XOR logic operations. The peak-to-peak in the differential trace (DPA) is about 6% of the peak-to-peak of the original signals (Power Analysis 1 and Power Analysis 2). As a comparison, the peak-to-peak DEMA is about the same level of the peak-to-peak of the original signals (EMA 1 and EMA 2) in Figures 4.6 and 4.7, indicating the same level of information leakage in the EM side-channel and in the power channel.

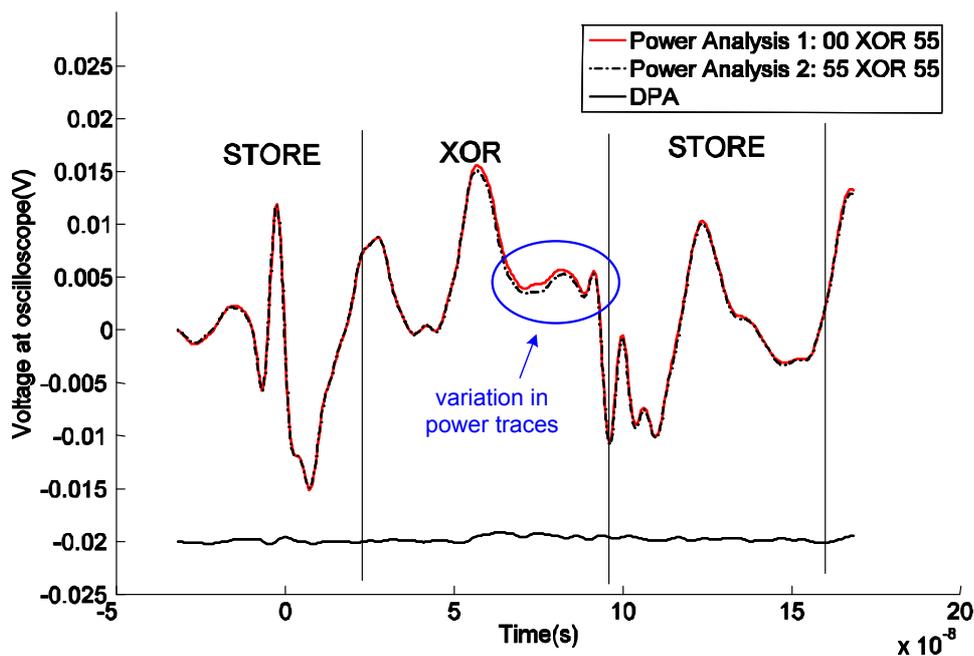


Figure 4.8: DPA measurement over the S-XAP processor executing XOR with different operands (experimental graph)

4.3.3 EM Simulation of an Asynchronous Processor

I then performed an EMA simulation on the DR-XAP processor which is designed in a dual-rail asynchronous style with a return-to-zero handshaking protocol. This balanced asynchronous circuitry was believed to be secure since power consumption should be data independent [24]. Figure 4.9 shows the EMA simulation result. On the graph I superpose the EM traces of the processor for ‘00 XOR 55’ and ‘55 XOR 55’, and put the DEMA trace at the bottom. The DEMA trace exhibits a wobble at only about 1% magnitude of that of the original traces (EMA1 and EMA2). This matches the projection that asynchronous design with dual-rail coding and return-to-zero handshaking is more secure against side-channel analysis attacks.

The measurement result in Figure 4.10 also indicates no information leakage during the logic operation. Comparing Figure 4.9 and 4.10, we can observe again lower magnitude in shape around the circled places in simulation, resulted from no memory accesses power consumption in simulation.

Performing EMA simulation on *modulated emissions* on the asynchronous processor, I achieved more intriguing results. I collected the current consumption data as I did in direct emission simulation, then I processed the data with amplitude demodulation. From the simulation results shown in Figure 4.11, a greater level of differential signals is observed compared to Figure 4.9. The peak-to-peak of the differential trace (DEMA) is about 32% of the peak-to-peak of the original signals (EMA 1 and EMA 2). The reason why the amplitude demodulated EMA reveals stronger differential signals is suspected to be data-dependent time shift in the program execution.

We can see higher peaks in Figure 4.11 around the second STORE operation, as a result of the time shift accumulated in previous operation. This EM information leakage caused by data-dependent timing is much higher in the tested asynchronous design than the synchronous

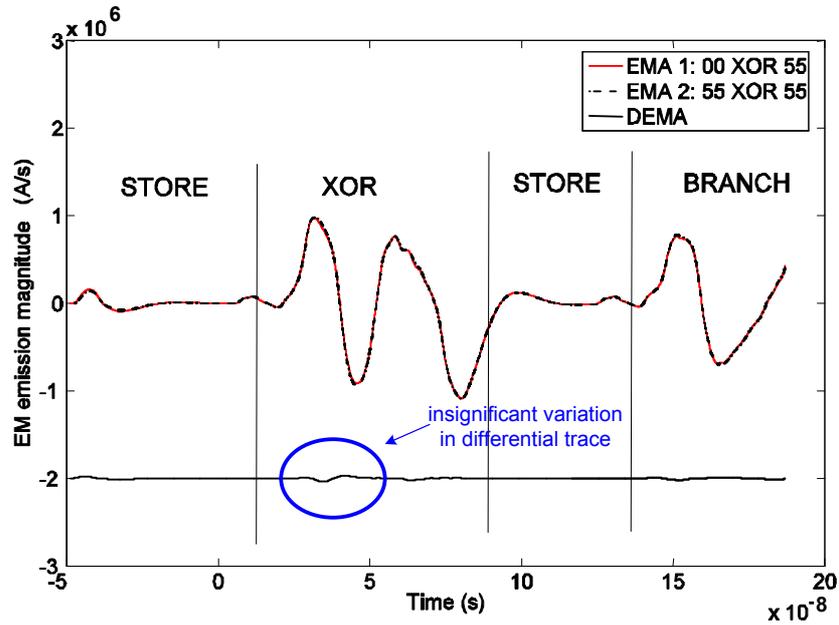


Figure 4.9: EMA simulation over the DR-XAP (asynchronous dual-rail) processor executing XOR with different operands

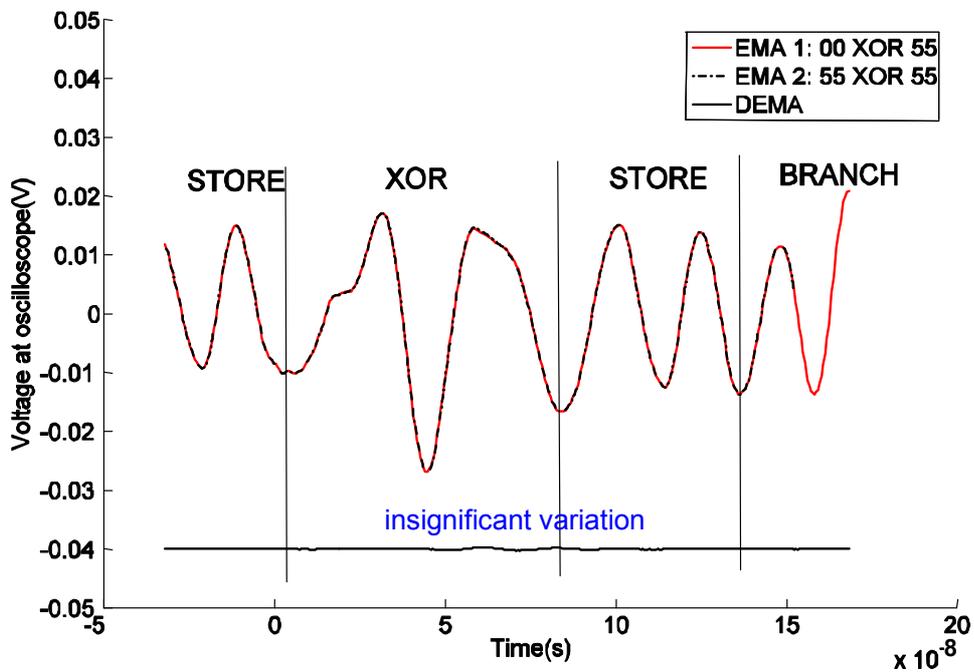


Figure 4.10: EMA measurement over the DR-XAP (asynchronous dual-rail) processor executing XOR with different operands (experimental graph)

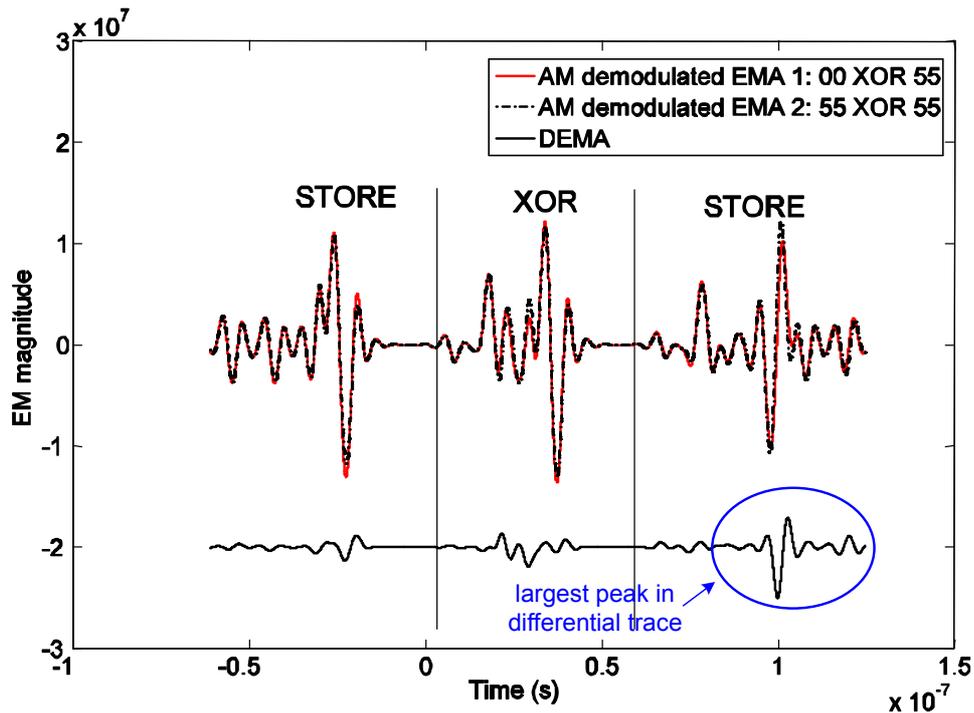


Figure 4.11: EMA simulation over the DR-XAP (asynchronous dual-rail) processor executing XOR with different operands, examining modulated emissions

design, as a result of the lack of clock synchronisation. The amplitude demodulated EMA simulation reveals an unexpected weakness in the asynchronous design against EM side-channel attacks, which provides a good example of the usefulness of design-time evaluation in a secure processor design flow.

4.4 Summary

A simulation methodology for EMA has been proposed on the basis of an analytical investigation of EM emissions in CMOS circuits. This simulation methodology involves simulation of current consumption with circuit simulators and extraction of IC layout parasitics with extraction tools. Once collected, the data of current consumption is processed with MATLAB to simulate EMA. The proposed simulation methodology can be easily employed in the framework of an integrated circuit design flow.

Testing has been performed on synchronous and asynchronous processors and the results have demonstrated that DPA and DEMA of direct emissions reveal about the same level of leakage. While DEMA of amplitude demodulated emissions reveals greater leakage, suggesting better chances of success in differential EM analysis attacks. The comparison between the EMA on synchronous and asynchronous processors indicates that the synchronous processor has data dependent EM emissions, while the asynchronous processor has data dependent timing which is visible in DEMA.

Chapter 5

Simulating Optical Fault Injection

As introduced in Chapter 2, secure microcontrollers and smart cards are cryptographic devices widely used for applications demanding confidentiality and integrity of sensitive information. They are also used for services requiring mutual authentication and non-repudiation of the transactions. These devices generally have an embedded cryptographic processor running cryptographic algorithms such as triple DES, AES or RSA. The algorithms encrypt data using secret keys, which should be kept safe in the devices so that attackers can not directly read out the key value or deduce it from side-channels [35, 37, 54].

However, it is not sufficient for the cryptographic processors to withstand the above passive attacks. They should also endure attacks that inject faults into the devices and thus cause exploitable abnormal behaviour. The abnormal behaviour may be a data error setting part of the key to a known value, or a missed conditional jump reducing the number of rounds in a block cipher. A glitch inserted on the power or clock line was the most widely known fault injection technique [10], but many chips nowadays are designed to detect glitch attacks. Optical fault injection introduced by Skorobogatov [58] in 2002 appears to be a more powerful and dangerous attack. It involves illumination of a target transistor which causes the transistor to conduct transiently, thereby introducing a transient logic error. Such attacks are practical as they do not require the expensive equipment that is needed in invasive attacks¹. This threat has become increasingly relevant as transistor dimensions and supply voltages are constantly scaling down. In deep submicron technologies², it is easier to introduce and propagate transient voltage disturbances as the capacitance associated with individual circuit nodes is very small, and large voltage disturbances can be produced from relatively small amounts of ionised charge. Also, due to the high speed of deep submicron circuits, the voltage disturbances can propagate more easily.

To keep cryptographic devices secure against optical fault induction attacks, various ideas have been proposed for the design of cryptographic devices. To evaluate this research effort, a *design-time security evaluation* methodology is proposed to exhaustively examine the response of secure processors under optical illumination by simulation, so as to assess their security level against optical fault injection attacks at design time.

¹Invasive attacks require decapsulation and deprocessing to get direct access to the internal components of the device.

²Gate lengths below $0.35 \mu\text{m}$ are considered to be in the deep submicron region.

5.1 Background

Optical fault injection is not entirely new. After semiconductor devices were invented, they were found to be sensitive to ionising radiation in space, caused by protons, neutrons, alpha particles or other heavy ions [13]. Pulsed lasers were then used to simulate the effects of ionising radiation on semiconductors [15]. Depending on several factors, laser illumination may cause: no observable effect, a transient disruption of circuit operation, a change of logic state, or even permanent damage to the device under test [21].

5.1.1 Ionisation and Charge Collection

It has long been known that laser ionisation and absorption is a fundamental band-to-band absorption process, where a pulsed laser with photon energy greater than the band gap of the semiconductor material excites carriers from the valence to the conduction band [31], and produces electron-hole pairs within semiconductor material such as Si and GaAs. In more detail, each absorbed photon is assumed to produce a single electron-hole pair, and the light is absorbed exponentially with depth x . Beer's Law describes the laser intensity function as: $I = I_0 e^{-\alpha x}$, where the absorption coefficient α is strongly dependent on the wavelength of the laser light λ and has been assumed to be constant for old device technologies. However, the assumption of linear absorption is no longer valid for new silicon-based technologies and most GaAs technologies for a number of reasons. First, the absorption coefficient α varies with temperature. For silicon, α approximately doubles at $125^\circ C$ compared to its value at room temperature. Secondly, at high doping levels, the presence of a large number of impurities reduces the energy gap and hence increases the absorption coefficient [31]. Thirdly, when pulsed lasers are focused to small spots, the resulting high power densities may cause additional absorption mechanisms such as two-photon absorption, which involves simultaneous absorption of two photons and thus a highly nonlinear increase in the absorption [31]. Furthermore, free-carrier absorption may occur, which does not produce ionisation but increases the energy of carriers within conduction or valence bands.

The laser intensity in the semiconductor sample is thus described by the following equation [51]:

$$\frac{dI_1(x,t)}{dx} = -\alpha(x,t)I_1(x,t) \quad (5.1)$$

where I_1 is the laser intensity, W/cm^2 ; x is the distance from the point of interest to the chip surface illuminated point; t is time; α is the total absorption coefficient, cm^{-1} . In silicon, α can be estimated as [52]:

$$\alpha = \alpha_{iz} + \alpha_n \cdot n + \alpha_p \cdot p \quad (5.2)$$

where α_n and α_p are the laser radiation interaction with electrons and holes cross sections, cm^2 ; n and p are the concentration of free carriers, cm^{-3} . $\alpha_{iz} (= \alpha_{iz}^0 + \alpha_{bn} \cdot N_d)$ is the laser radiation interzoned absorption factor of semiconductor, cm^{-1} ; α_{iz}^0 is the laser radiation interzoned absorption factor in lightly doped semiconductor, cm^{-1} ; α_{bn} is the band narrowing effect factor caused by high doping concentration N_d , α_{bn} is in cm^2 and N_d is in cm^{-3} .

Equation (5.1) can give us the free carriers generation rate [49] as:

$$G(x) = \eta \cdot \alpha_{iz} \cdot \frac{I_1}{h\nu} \cdot (1 - R) \cdot e^{-\alpha x} \quad (5.3)$$

where η is the photo-ionisation quantum efficiency (the free carriers pairs quantity, generated by an absorbed quantum), with value at about 1 near the main absorption band edge; $h\nu$ is the laser quantum energy, in Joules; R is the reflection coefficient (0.3 for silicon substrates when radiation performed from the back side; $0.1 \sim 0.3$ for various oxide thickness when radiation performed from top-side).

When the excited charge amount reaches the critical charge Q_{crit} , the charge necessary to flip a binary "1" to a "0" or vice-versa, a single event upset (SEU) occurs. Device immunity is determined by its threshold linear energy transfer (LET). The threshold LET (LET_{th}) is defined as the minimum LET required to produce a voltage change (ΔV) sufficient for an SEU, then mathematically:

$$LET_{th} \propto \Delta V (= \frac{Q_{crit}}{C}) \quad (5.4)$$

Where C is the capacitance of the struck node.

5.1.2 Metal Shielding Effect

The previous subsection introduces the physical mechanism of laser ionisation and charge collection in a semiconductor. However, metal on top of the sensitive junctions prevents the light from penetrating these regions directly, so that has to be taken into consideration for fault injection. The metal shielding reduces the average incident energy in proportion to the surface metallisation [49]:

$$P_e^m(x) = P_e(x)(1 - K_m) \quad (5.5)$$

where $P_e(x)$ is the incident energy without metal shielding effect; $P_e^m(x)$ is the incident energy with metal shielding effect; $K_m = S_m/S$; S is the total top surface area under illumination, while S_m is the metallisation area within.

A way to bypass metal shielding is to attack the chip from the back, if the target device allows this.

5.1.3 Classes of Attackers

Abraham et al defined attackers of IBM cryptographic products into three classes according to their expected abilities and attack strengths [7]. Following this classification, and porting it to optical fault induction attacks, we categorise those attackers into three types according to their knowledge about the system and the resolution that their laser scan equipment allows:

Type I (not knowing layout, targeting many transistors):

They are outsiders with moderately sophisticated tools. They do not have detailed knowledge of the layout, and can only perform moderately low resolution scans of the chip, targeting a group of neighbouring transistors.

Type II (not knowing layout, targeting a single transistor):

They are outsiders with sophisticated tools. They do not have detailed knowledge of the layout, but can perform high resolution scans of the chip targeting individual transistors in order to determine what faults can be injected.

Type III (knowing layout, targeting a single transistor):

They are knowledgeable insiders, having detailed information of the layout of the chip under attack, and information about the program code. They also have access to highly sophisticated tools such as a probing-station with a high resolution focused laser allowing any single transistor to be targeted.

Type I attackers are especially dangerous, since the entry costs for training, intelligence and equipment are relative low. Therefore they represent the largest group of potential attackers. The ease of Type I attacks indicates that they are the most dangerous, so are the focus of this work.

Type II and III attackers on the other hand can conduct an attack on any transistor node during a cryptographic program execution, knowing or not knowing its specified functionality. The demanding large capital investment and detailed internal knowledge prevent most attackers falling into these categories. However, they are still of interest. Such attackers have higher capability to manipulate the circuit so more defensive effort is required from chip designers. Type II differs from Type III in that Type II attackers have no detailed knowledge of the layout of the chip. This is often the case for attackers targeting a design implemented in a “glue-logic” approach, which is widely used in smart cards [67]. Glue logic involves a layout optimisation of the whole non-memory part of the chip – so the instruction decoder, register file, ALU and I/O are no longer visible to the attacker as separate functional units, but become indistinguishable from each other in a sea of gates. This design style makes reverse engineering and microprobing much more tiresome. Exhaustive laser scans can still identify vulnerabilities where they exist, but now the attackers need significant automation. In effect, glue logic results in a significant separation of the costs and capabilities of Type II versus Type III attackers, and creates a strong incentive for chip layouts to be kept confidential.

5.1.4 Modelling Optical Fault Induction

Numerical device modelling for radiation effects has long been in existence. It can be made at a number of different levels, from physical device models through to digital abstractions.

Device modelling

The earliest work for device simulation consisted of one-dimensional drift-diffusion models [28]. In a drift-diffusion (DD) model, current equations are derived from the Boltzmann transport equation considering a steady state situation and some numerical approximations for a 1-D geometry. These equations are discretized and solved on a mesh using finite-difference or finite-element techniques [57].

The alternative device modelling strategy is based on hydrodynamic and energy balance (EB). It has fewer assumptions [40], but is more computationally intensive, based on five or six equations of state rather than the three used in the drift-diffusion method.

The 1-D device models based on drift-diffusion equations for carrier densities and models based on hydrodynamic and energy balance have evolved to 2-D and 3-D device modelling approaches. Many charge collection and SEU studies have been performed using these models. An early comparison of 2-D and 3-D charge-collection simulations showed that while the transient responses were qualitatively similar, quantitative differences existed in both the magnitude of the current response and the time scale over which collection was observed [38].

The comparison implies that 2-D simulations can provide basic insight whilst 3-D simulations become necessary when truly predictive results are to be obtained.

Circuit simulations

Although fully 3-D device simulators were first reported in the literature in the early 1980s [16], only in the last few years have fully 3-D device simulators become commercially available [21]. Even optimised for high-end workstations, a fairly large 3-D device simulation can still take a few hours. Even 2-D device modelling is too computationally expensive for simulating response of a large circuit to optical fault injection. Therefore, in order to exhaustively examine the effect of optical fault injection on a large circuit, we need to relate the collection of charge in individual device junctions to the changes in the circuit currents and voltages. A common circuit model for charge collection at a junction due to direct funnelling or diffusion is a double-exponential, time-dependent current pulse [46], with a typical rise time on the order of tens of picoseconds and a fall time on the order of 200 to 300 ps [42]. The actual magnitude and time profile of the current model depends on material parameters, the ion species, the ion energy, device dimensions, and the hit location relative to the junction. If the time profile (or the shape) of the collection current pulse is not important to the circuit response to the hit, then analytical current models can usually adequately describe the induced current pulse. If, however, the time profile is critical to the circuit response, more accurate models for the current pulse are necessary, such as those derived from a device simulation. In an optical fault injection attack introduced in [58], the shape of the collection current is not important to the circuit response to the attack, and a piece-wise linear (PWL) pulse can even be used to represent the induced current pulse for the purpose of simplicity.

Mixed device/circuit simulations

Recently, the simultaneous solution of device and circuit equations has been increasingly used. With this technique, known as mixed device/circuit simulation of an SEU, the struck device is modelled in the “device domain” (using multi-dimensional device simulation), while the rest of the circuit is represented by SPICE-like compact circuit models. The two domains are tied together by the boundary conditions at contacts, and the solution to both sets of equations is rolled into one matrix solution [56, 44]. The advantage is that only the struck device is modelled in multiple dimensions, while the rest of the circuit consists of computationally efficient SPICE models. This decreases simulation times and greatly increases the complexity of the external circuitry that can be modelled.

However, as circuits grow exponentially in density and complexity, comprehensive mixed device/circuit simulation is impractical. Therefore in our approach, we stick to circuit level simulation with analytic current models to perform a systematic and exhaustive laser scanning examination, as described in Section 5.1.4.

5.2 Simulation Methodology

The flow of designing and evaluating a test chip against optical fault injection attacks is outlined in Figure 5.1. A major concern with this traditional approach is that security evaluation occurs too late in the design cycle to allow for efficient repair. The deficiencies in the design

often result in costly and frequent design re-spins. As a comparison, the procedure with evaluation incorporated in the design flow is demonstrated in Figure 5.2. This design flow can spot design oversights or errors at an early stage to avoid costly silicon re-spins.

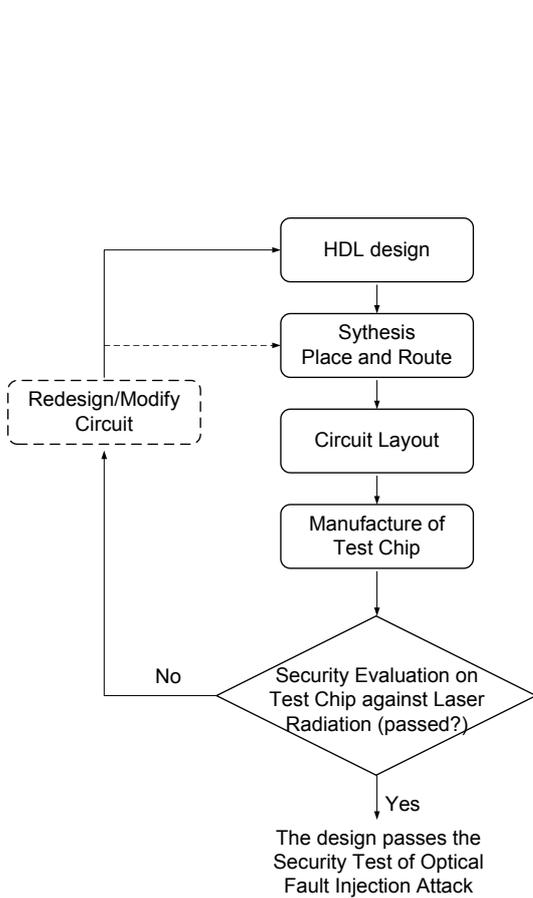


Figure 5.1: Flow chart exhibiting the traditional iterative process to design and evaluate a test chip against optical fault injection attacks, after [45]

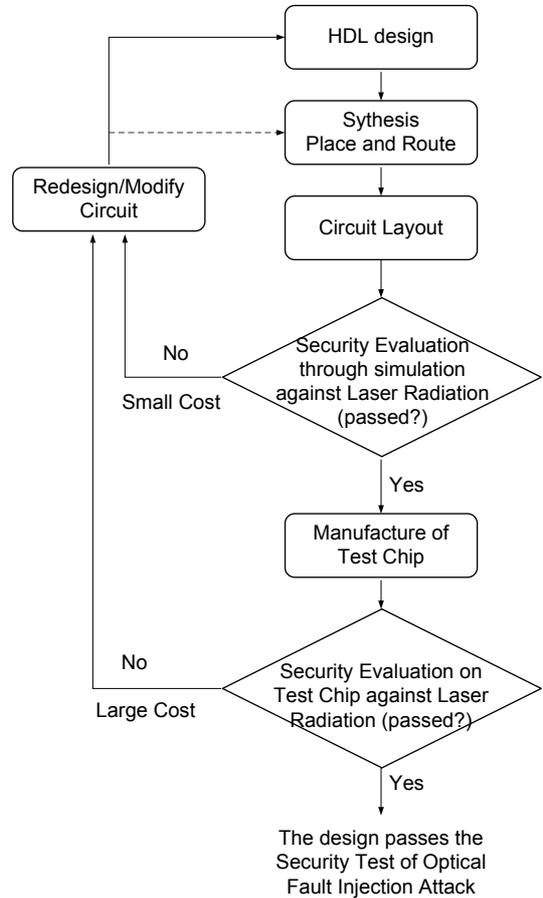


Figure 5.2: Flow chart exhibiting the iterative process to design and evaluate a test chip against optical fault injection attacks with the aid of design-time security evaluation

5.2.1 Simulation Procedure

The procedure for simulating optical fault injection attacks is illustrated in Figure 5.3. A co-simulator is used to combine a Verilog simulator (or simulators supporting other hardware description languages (HDLs)) and a SPICE-like simulator. The modules of interest in the Verilog netlist are swapped out with the full transistor-level netlist. Within the transistor-level netlist, the cells under attack are instantiated into transient stimuli according to the layout scanning process. The stimuli are in essence voltage pulses supplied via tri-state buffers to the nodes under attack. The HDL/SPICE integration allows the simulation to have gate-level speed and transistor-level accuracy. The scanning process in this paper is performed with Cadence Silicon Ensemble™, and the HDL/SPICE co-simulator is chosen to be Synopsys NanoSim™ integrated with the Synopsys Verilog simulator, VCS™. Other similar and

commercially available simulation environments include Cadence AMS™, Mentor Graphic ADVance MS™, Dolphin Integration SMASH™, etc.

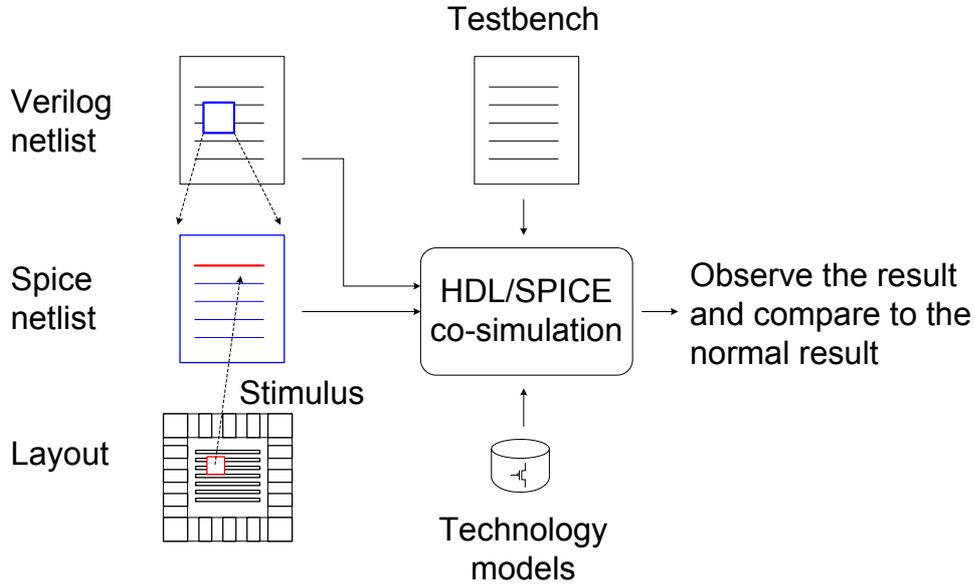


Figure 5.3: Simulation procedure for optical fault injection attack

The layout can be scanned with any size of laser illumination spot, which can target from a single transistor to hundreds of transistors, depending on the equipment used by the attackers as described in Section 5.1.3. The scans can be performed over a particular area such as the ALU, register file, or even the whole processor. Figure 5.4 illustrate scanning in simulation, where each scan (S_{11} , S_{12} ... S_{mn}) generates a list of logic cells under attack. For example, in a particular scan, exposed cells are listed as follows:

```
m/datapath/U355      m/datapath/fi_reg_4   m/U1490   m/U1506
m/datapath/alu/U33  m/U1458               FC_299    m/U1223
```

Among the selected cells, FC_299 is a filler cell and the rest are logic cell instances. We first discard the filler cells, then check the standard cell library, mapping the logic cells to their internal nodes, especially the nodes connected to n-type transistors³. In addition to what may be considered a useful attack mechanism, negative effects are also possible. These include the possibility that latch-up may be induced by the generation of photocurrents in the bulk (the substrate and well). Of less concern when using readily available infra-red and visible laser light sources is the ionisation of gate- and field-oxides due to the large band gap energy of silicon dioxide (which would require a laser with a wavelength in the UV-C range). Ionisation of this type is common when higher energy forms of radiation are absorbed. The subsequent accumulation of positive charge results in a long term shift in transistor characteristics.

Based on the fact that optical attack is substantially more effective at turning on n-type

³Or connect the nodes to p-type transistors depending on the process technologies, especially the substrate type and the well type, see Appendix for details.

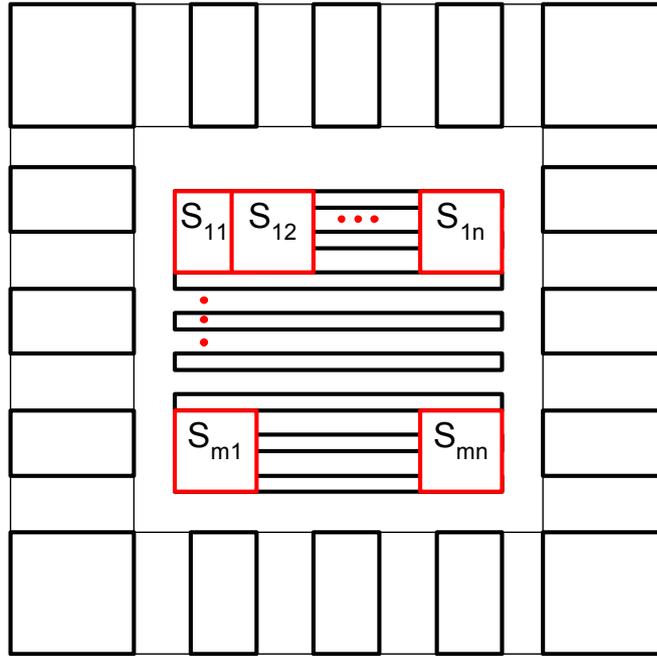


Figure 5.4: Layout scanning to extract groups of exposed cells

transistors than their p-type counterparts⁴, the laser radiation will result in one of three behaviours in a given logic gate:

- The laser radiation is not strong enough to cause either the n-type or the p-type CMOS transistors to conduct, so no state change occurs at the logic cell output.
- The laser radiation switches on the n-type but not p-type transistors, so abnormal behaviour may occur.
- The laser radiation is strong enough to cause both n-type and p-type CMOS transistors to conduct in a logic gate. This results in large leakage current or even a strong VDD-to-GND short circuit, which may damage the circuit eventually if no current limit protection is provided

Of the three behaviours, only the second is considered as a successful attack as opposed to sabotage, and is therefore the focus of this simulation methodology. This allows us to simply focus on n-type transistors in the simulation of security evaluation targeting Type I attackers. Apparently, in the case where the laser can target a single p-type transistor and successfully switch it on, the attacker is able to manipulate the circuit more capably. This situation falls into the category of Type II and III attacks. The corresponding simulation requires layout scans over every single transistor.

After obtaining the list of exposed cells for each scan, we then supply the internal nodes with transient voltage pulses via tri-state buffers. The enable signals of the tri-state buffers are synchronised with the target instruction execution during a cryptographic program operation. The co-simulation shown in Figure 5.3 integrates the voltage pulses and illuminated

⁴Or more effective at turning on p-type transistors than n-type, depending on the process technologies, especially the substrate type and the well type, see Appendix for details.

cells in SPICE, whilst the rest of the circuit remains in Verilog. Analysing the response and comparing it to that of the normal operation, we can evaluate the security of the circuit design against optical fault injection attacks. If it fails, modification or even redesign of the circuit is required as demonstrated in Figure 5.2. If it passes, then designers can continue to have the chip manufactured.

5.3 Results

5.3.1 Optical Attack Simulation Results

Simulation of optical fault injection attacks has been carried out on the Springbank test chip. This simulation addresses the synchronous processor (S-XAP) on the top left corner of the chip as shown in Figure 3.6. The substrate/well formation is a p-substrate with twin-well. According to the Appendix, n-type transistors are easier to switch on, so are simulated.

The aim of the test is to exhaustively examine the ALU and decoder of processor S-XAP to determine if it is susceptible to optical fault injection attacks. We target simple instructions (*e.g.* XOR, shift, load, store etc) again as we did for DPA and DEMA in Chapter 3 and 4, which can give a good indication of how the hardware reacts to operations of cryptographic algorithms. The fragment of a program, shown in Figure 5.5, is used for the evaluation, where the processor loads the first argument to register AH, XOR it with the second argument from memory, then saves the result back to memory. The laser attack is synchronised with the XOR operation, meaning the transient voltage sources will be activated at this moment in simulation.

```

...
ld      ah,@(1,x)    ; load first argument
nop
nop
nop
xor     ah,@(2,x)    ; XOR operation
nop
nop
nop
st      ah,@(3,x)    ; save result
...

```

Figure 5.5: Fragment of the instruction program used for the evaluation

The simulation procedure is implemented as introduced in Section 5.2.1. I scan the ALU and decoder with a scanning square size of about $300 \mu\text{m}^2$, to cover 10 ~15 logic cells. Figure 5.6 shows the screen-shot of the scanning procedure. The spot is moved within the area horizontally each time by one cell width (about $4 \mu\text{m}$ or more), then vertically by one cell height (about $6 \mu\text{m}$). The scanning produces 120 lists of cells, mimicking 120 optical fault injection attacks. For each list, we connect the internal nodes to transient voltage sources and incorporate the stimuli into the SPICE netlist. Then the Verilog/HSPICE co-simulation

running the above simple instruction program is performed to examine the circuit response during each optical attack.

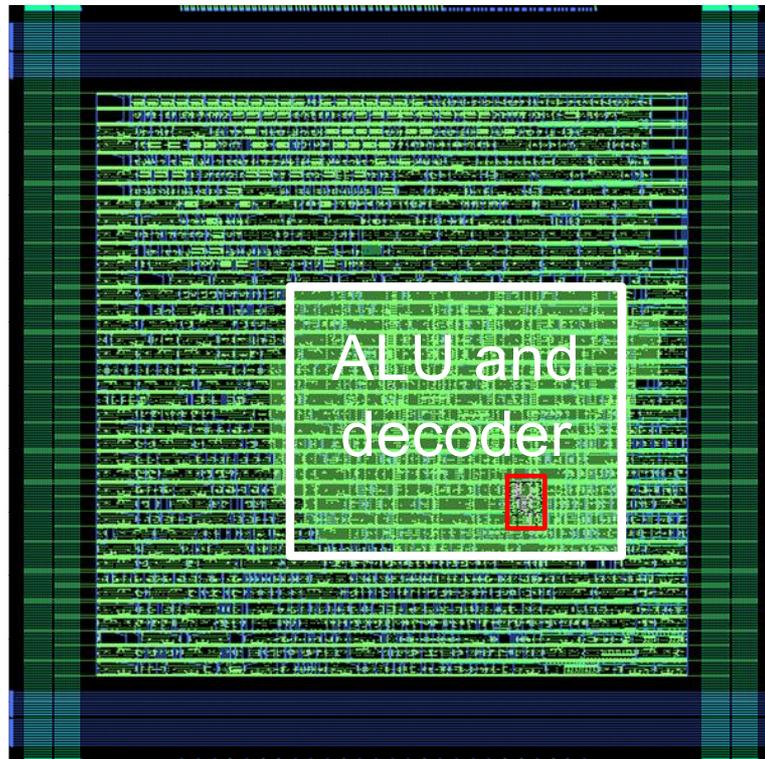


Figure 5.6: Screen shot of scanning procedure over the layout of ALU and decoder of processor S-XAP: the region within the little square being illuminated

The exhaustive examination of the 120 simulation runs shows different results:

1. The processor results in *deadlock* in many cases, which is desirable in terms of security, provided this does not leak secret information.
2. Some other cases show *normal* program execution. This implies the introduced fault may be part of the “don’t care” state of the subsequent operation of the circuit [21].
3. Two *failures* are also revealed:
 - (a) The first disrupts the XOR operation by changing the value in the AH register.
 - (b) The second failure causes a memory dump. Instead of executing a data write to memory, the processor keeps reading the contents of the whole memory. We suspect the memory dump occurs when the decoder was struck in the test, which resulted the opcode being modified from “1101” (standing for XOR) to “0001” (standing for LOAD).

Modifying register values implies that setting part of the key to a known value becomes feasible to the attackers. Dumping memory can be dangerous to designs implemented with an architecture where a single storage structure is used to hold both instructions and data. If the memory contains passwords and decryption keys, then by carefully analysing the dumped

memory, one can break the cryptographic device. In contrast, a design implemented with Harvard architecture [1] could offer better protection against microprobing attacks, as it uses physically separate storage for instructions and data. The same trick applied to a Harvard microcontroller would reveal only the program code, whereas the data memory containing sensitive information will not be available.

It takes about 10 minutes to run the scanning process (containing 120 scans) with Cadence Silicon Ensemble™. Then it takes about 4 hours to complete the 120 runs of HDL/SPICE co-simulation, with each run to have 14,000 transistors simulated in Synopsys NanoSim™ and the rest tens of logic gates simulated in Synopsys VCS™. All the simulation work is done on a 1.6 GHz AMD Athlon processor with 2GB memory.

5.3.2 Experimental Results

A laser fault injection experiment was conducted by Gemplus® on the same test chip to provide a side-by-side comparison [24]. The test chip was mounted in a ZIF (zero-insertion force) socket, which was mounted on the bottom side of the test board, thus easing access for the laser attack. The laser is synchronised with the executed program (same as the code used in the simulation) via an interrupt signal from a particular I/O pin. The experiments reveal that not every portion of the processor is sensitive to the laser. When there is an actual effect, the processor goes into a failure state in most cases, and the chip has to be reset in order to reload the program. By shooting the laser at the ALU of the processor, we finally obtained effects like modification of the result of a XOR operation, which agrees with the first type of failure discovered through simulation. Also we succeeded in dumping the data memory in the processor S-XAP by shooting the laser at a place within the region of the ALU and registers. This result is similar to the simulation except that in the experiment the assembly code contained a subroutine to display the two operands and the result of the XOR operation. The disrupted execution had the effect of outputting consecutive values from data memory.

5.4 Summary

A simulation methodology has been proposed to evaluate the security of cryptographic processors against optical fault injection attacks at design time. This simulation methodology involves exhaustive scans over the layout with any laser spot size according to the attack scenario. Cells under illumination are identified and simulated in SPICE with additional voltage spikes at appropriate nodes which mimic the attack. This SPICE model is co-simulated with the rest of the system represented in Verilog.

Simulation performed on our test chip has demonstrated that the optical fault injection could modify the value stored in registers, so that setting part of the key to a known value becomes feasible to the attackers. Attacks on other area also caused a data memory dump, which can be extremely dangerous if the memory contains passwords and decryption keys. Experimental results revealed the same kind of weaknesses, which gives us the confidence in the proposed simulation methodology.

Appendix

Charge collection in different processes considering types of the substrate and well

When laser illumination strikes a microelectronic device, the most sensitive regions are usually the reverse-biased p/n junctions. The high field present in a reverse-biased junction depletion region can very efficiently collect the ionised charge through drift processes, leading to a transient current at the junction contact. An important consideration for charge collection is whether the junction is located inside a well or in the substrate. Figure 5.7 shows a cross-section of a CMOS inverter in a p-substrate with n-well process. There are other substrate and well types, including:

- p-substrate(n-MOS) + n-well(p-MOS)
- p-substrate + twin-well (p-MOS in n-well and n-MOS in p-well)
- n-substrate(p-MOS) + p-well(n-MOS)
- n-substrate + twin-well (p-MOS in n-well and n-MOS in p-well)

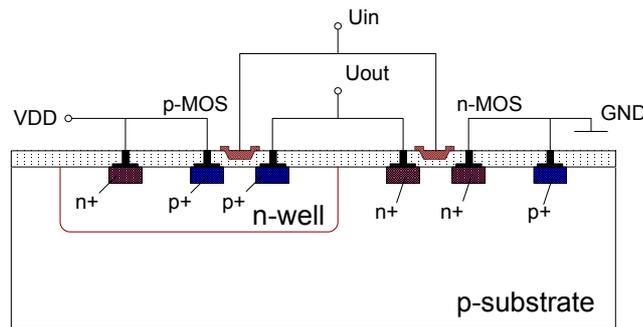


Figure 5.7: Cross-section of a CMOS inverter in a p-substrate + n-well process

As technologies are constantly scaling down, inside-the-well strikes are particularly interesting because of shunt and bipolar effects that can occur in multilayer structures [22]. Figure 5.8 demonstrates a n-MOS transistor implemented in a p-substrate with p-epitaxial and p-well process. As a SEU transient proceeds, holes deposited in the p-well are collected at the p-substrate contact, raising the well potential and leading to injection of electrons by the source. This results in the turn-on of the horizontal n-source/p-well/n-drain (emitter/base/collector) parasitic bipolar. The movement of the carriers is illustrated in Figure 5.9 [22].

Dodd et al [22] studies the gate-length scaling trend in inside-the-well strikes for both p- and n-substrate technologies (Sandia 2 μm , 1 μm and 0.5 μm). Figure 5.10 shows the simulated SEU threshold scaling trend of OFF transistors fabricated on a n-substrate. The upset threshold of the inside-the-well strikes decreases at a much faster rate than that of outside-the-well strikes. Figure 5.11 displays the scaling trend of OFF transistors fabricated on a p-substrate. A similar trend exists except that the inside-the-well (p-MOS) strike starts out much harder (much higher LET threshold than the n-MOS counterpart). The weaker

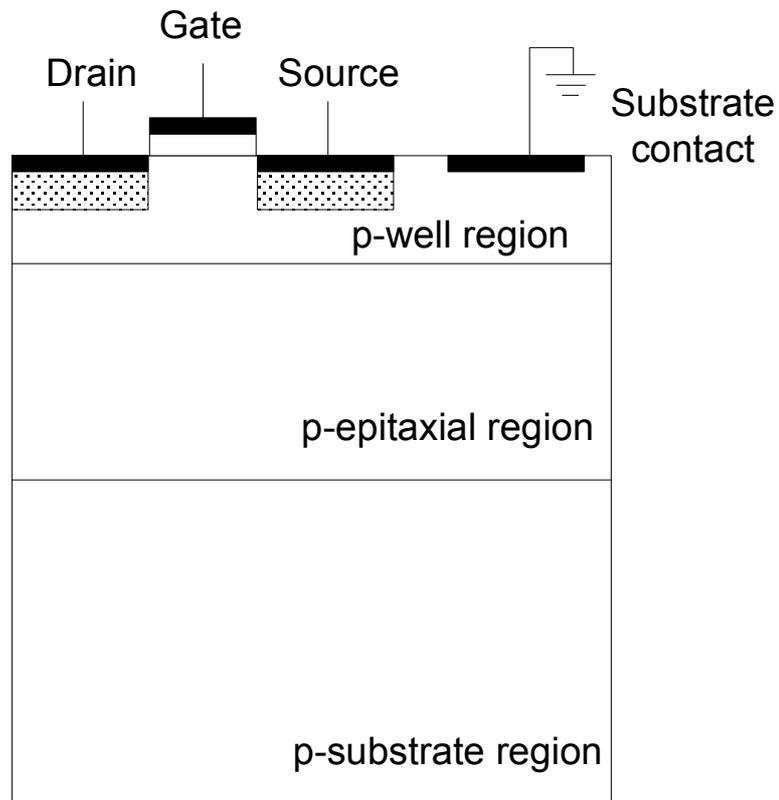


Figure 5.8: Cross-section of a n-MOS in p-well + p-epitaxial + p-substrate process[22]

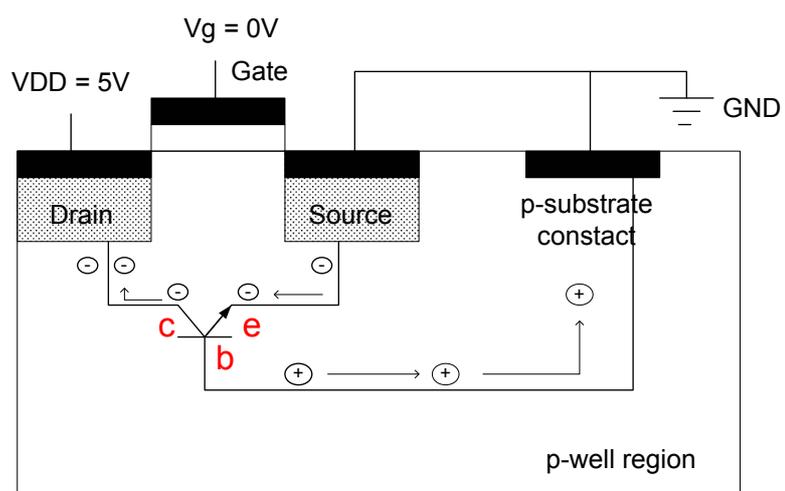


Figure 5.9: The movement of carriers in the parasitic bipolar [22]

bipolar effect for the p-well case is simply because in p-well, the parasitic bipolar is pnp rather than npn. For identical structures, a pnp bipolar will have lower current gain ($\sim 1/3$) than an equivalent npn due to the lower mobility of holes compared to electrons.

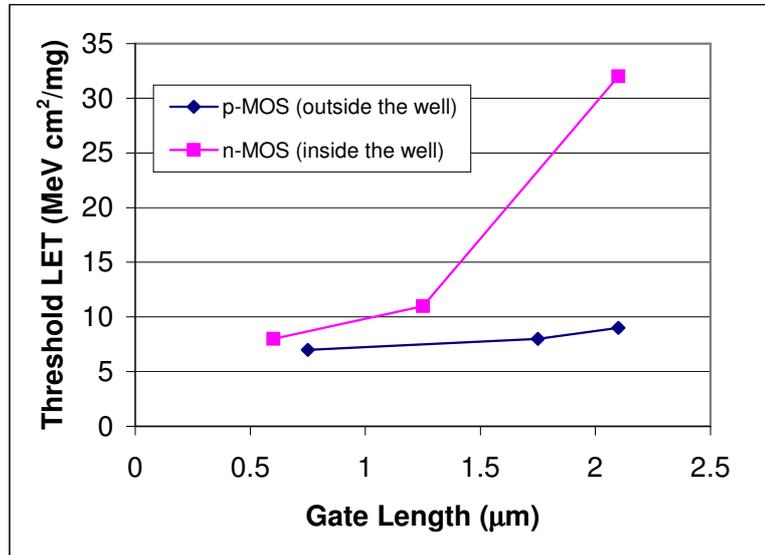


Figure 5.10: Simulated threshold LET vs. gate length in n-substrate technologies, after [22]

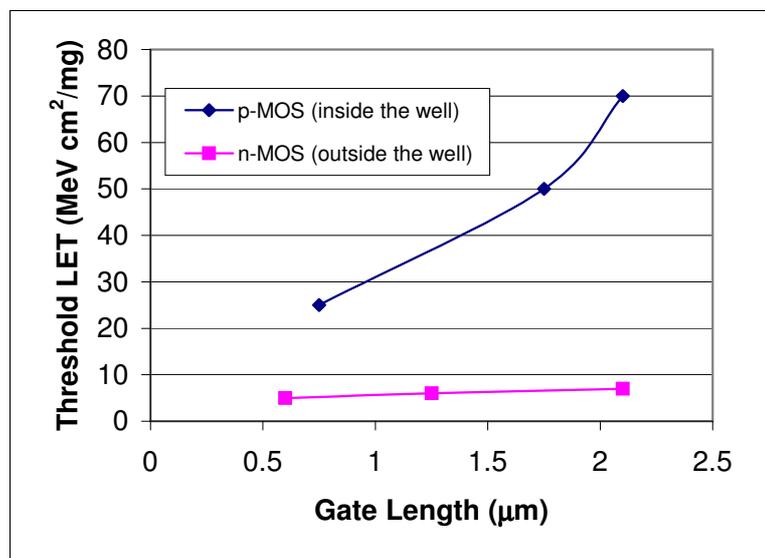


Figure 5.11: Simulated threshold LET vs. gate length in p-substrate technologies, after [22]

According to the trend shown in Figure 5.10 and 5.11, a rule of thumb is

- for p-substrate, either p-substrate + n-well or p-substrate + twin-well: n-MOS is easier to switch on
- for n-substrate, either n-substrate + p-well or n-substrate + twin-well: above 1 μm technology node, p-MOS is easier to switch on, below 1 μm technology, device simulation or experiment is required to determine the minimum upset LET for

n- and p-MOS respectively, before the proposed simulation methodology is applied on a large scale IC.

With silicon-on-insulator, the situation will be different but this is not discussed here as all microcontrollers and smart cards nowadays use a bulk silicon design style.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

This thesis has introduced the security hazards for consumer devices like smart cards. Traditional industrial practise has been to evaluate the security of hardware post manufacture. This is an expensive and error prone process. Therefore I proposed a set of design-time security evaluation methodologies which provide systematic and exhaustive simulation at design time to evaluate the security of the design under test against various attacks.

The main contribution of this thesis is the design-time security evaluation methodology against differential power analysis (DPA) attacks, electromagnetic analysis (EMA) attacks and optical fault injection attacks.

- The simulation methodology for DPA of secure processors includes power simulation of the logic circuitry and low-pass filtering caused by on-chip parasitics and package inductance.
- The simulation methodology for EMA involves simulation of current consumption with circuit simulators and extraction of IC layout parasitics with extraction tools. Once collected, the current consumption data is processed with MATLAB to implement Differential EMA (DEMA) according to various sensor types and emission types.
- The simulation methodology for optical fault injection attacks involves exhaustive scans over the layout with a laser spot size chosen according to the attack scenario. Once the exposed cells for each scan are identified, they are mapped to their internal nodes, especially the n-transistor output nodes or the p-transistor output nodes, depending on the process technologies. Then these nodes are driven by transient voltage sources via tri-state buffers, to mimic the effect of transistor conduction caused by laser illumination. Finally, the response of the circuit is examined and compared to that of the normal circuit without a laser attack.

These simulation methodologies have covered side-channel analysis attacks that have been threatening the smart card industry. Although the simulation examples demonstrated in the thesis are on simple microprocessors, the simulation methodologies are applicable for evaluating more complex processors including multiple pipelines, multiple cores and multi-threading implementations. The simulation methodologies are also applicable for evaluating

advanced defence techniques, such as out-of-order execution, random-delay insertion and cryptographic algorithm transforming, by writing proper test benches to verify these countermeasures. The DPA and DEMA simulation methodologies can be easily extended to other variants of side-channel analysis attacks, such as second-order differential power analysis suggested by Messerges [47] to defeat random masking [29]. Second-order differential power analysis requires the attacker to know the time before and after the random masking operation, and compute the difference of the power consumption between the two time instances within the same power trace. This process can be easily performed in the proposed simulation flow.

Comparison with post-manufacture test

Compared to post-manufacture test, these simulation methodologies can spot design oversights at an early stage to avoid expensive silicon re-spins, and they can be performed in a relatively short time and yield relatively accurate and practical results.

The simulation methodologies have the potential to extend for more advanced attacks. For example, in the EMA attack, the sensor may be further miniaturised in the future and focused on more local emissions. The simulation methodology can cope with this easily by collecting the desired branch current data. For optical fault injection attacks, the simulation examples demonstrated in the thesis are only for “*one place at a time*” attacks. In an advanced form, attackers may simultaneously hit two or more distinct places for better control or even rapidly move the laser spot(s) over a certain area. The simulation methodology can cope with this by incorporating extra transient voltage sources in those places.

Final comments

The proposed simulation methodologies have laid the cornerstones for building a complete suite of design-time security evaluation tools. Our design-time evaluation methodology is able to simulate all known circuit-level attacks and defence techniques. Such techniques are often complemented by barrier technologies, such as refractory chip coatings or top-level defence grids; these must still be evaluated by post-manufacture test. However, our techniques can replace the most tedious and expensive part of the security test process.

6.2 Future Work

Finally we suggest some directions for further research into design-time security evaluation.

Differential EM analysis in the frequency domain

There is an extension of the existing differential side channel attack, where instead of performing analysis in the time domain, the frequency domain is used [27]. Analysing signals captured in the frequency domain solves the problem of misalignment (or time-shifts) in traces since fast Fourier transform (FFT) analysis is time-shift invariant. Additionally, frequency analysis may reveal loops and other repeating structures in an algorithm that is not possible with time domain analysis. However, there are two problems with using frequency domain signals in differential analysis. First, it reveals no information about when

data-dependant operations occur. This timing information is very useful as it helps an adversary focus the signal analysis on these data-dependant operations. Secondly, any peaks in frequency domain due to an event that occurs in a short duration may be discernible if the acquisition duration is a lot longer. The solution to these problems is to use a spectrogram, i.e., time dependant frequency analysis. The main component when creating a spectrogram is an FFT which coverts a time domain signal to a frequency domain signal, with the appropriate width of Hamming windows which are used to suppress the Gibbs phenomena in spectral windowing.

Device modelling before OFI simulation

With shrinking technology size and an increasing number of metal layers, optical fault induction (OFI) attacks that previously focused on a single transistor will necessarily affect several devices. This pushes attacks toward Type I in our taxonomy in Chapter 5. The proposed simulation methodology maps the illuminated area on the physical layout to the nodes of the logic cells in the netlist, especially the output nodes of the n-type or p-type transistors. Which type to choose depends on the semiconductor process technology including the substrate and well topology, dopant concentration, as well as the laser intensity. To constitute a generic simulation methodology against optical fault injection, a device modelling is suggested prior to the circuit-level simulation to compare the upset threshold LET for n- and p- type transistors. The lower upset threshold LET means that type of transistor is easier to switch on using a laser. Closing upset threshold LET for n- and p- type transistors, however, can be regarded as an effective defence technique for CMOS IC circuits, since simultaneous conduction of n-type and p-type CMOS transistors in a logic gate causes a large leakage current or even a strong VDD-to-GND short circuit which can be easily detected.

List of Papers

The research work in this thesis was presented and published in the official proceedings of rigorously refereed conferences through the following research papers:

- Huiyun Li and Simon Moore, “Security Evaluation at Design Time Against Optical Fault Injection Attacks”, accepted by IEE Proc. Information Security.
- Huiyun Li, A. Theodore Markettos and Simon Moore, “A Security Evaluation Methodology for Smart Cards Against Electromagnetic Analysis”, in proceedings of 39th IEEE International Carnahan Conference on Security Technology, Pages 208- 211, 2005
- Huiyun Li, A. Theodore Markettos and Simon Moore, “Security Evaluation Against Electromagnetic Analysis at Design Time”, in proceedings of Workshop on Cryptographic Hardware and Embedded Systems (CHES2005), LNCS Volume 3659, Pages 280 - 292, 2005
- J. Fournier, H. Li, S.W. Moore, R.D. Mullins and G.S. Taylor, “Security Evaluation of Asynchronous Circuits”, in proceedings of Workshop on Cryptographic Hardware and Embedded Systems (CHES2003), LNCS Volume 2779, Pages 137 - 151, 2003

The following informal presentation was also given:

- Huiyun Li, Simon Moore and A. Theodore Markettos, “Towards Security by design: A security evaluation methodology for Electromagnetic analysis”, in proceedings of Post-graduate Research in Electronics, Photonics, Communications and Software (PREP2005), March 2005, UK

Bibliography

- [1] The free dictionary encyclopedia: Harvard architecture. <http://encyclopedia.thefreedictionary.com/harvard%20architecture>.
- [2] An introduction to java card technology. <http://developers.sun.com/techtopics/mobility/javacard/articles/javacard1/>.
- [3] PrimePower Manual.
- [4] RFID Technologies. <http://www.tandis.com/rfid.htm>.
- [5] Smart card applications. http://www.javacard.org/others/sc_applications.htm.
- [6] Smart card tutorial. <http://www.smartcard.co.uk/tutorials/sct-itsc.pdf>.
- [7] D.G. Abraham, G.M. Dolan, G.P. Double, and J.V. Stevens. Transaction security system. In *IBM Systems Booktitle*, volume 30, pages 206–229, 1991.
- [8] D. Agrawal, B. Archambeault, J. Rao, and P. Rohatgi. The EM side-channel(s). In *Proceedings of Cryptographic Hardware and Embedded Systems - CHES2002*, pages 29–45, 2002.
- [9] Smart Card Alliance. Smart cards and biometrics report. http://www.smartcardalliance.org/about_alliance/Smart_Card_Biometric_report.cfm, 2002.
- [10] R. Anderson and M. Kuhn. Tamper resistance - a cautionary note. In *The Second USENIX Workshop on Electronic Commerce Proceedings*, pages 1–11, 1996.
- [11] Ross Anderson. *Security Engineering – A Guide to Building Dependable Distributed Systems*. Wiley, 2001.
- [12] R.M. Best. Microprocessor for executing enciphered programs. In *US Patent 4168396*, 1979.
- [13] D. Binder, E.C. Smith, and A.B. Holman. Satellite anomalies from galactic cosmic rays. In *IEEE Transactions on Nuclear Science*, volume 22, pages 2675–2680, 1975.
- [14] M. Bucci, M. Guglielmo, R. Luzzi, and A. Trifiletti. A Power Consumption Randomization Countermeasure for DPA-Resistant Cryptographic Processors. In *PATMOS*, pages 481–490, 2004.

- [15] S. Buchner. Laser simulation of single-event upsets. In *IEEE Transactions on Nuclear Science*, volume 34, 1987.
- [16] E.M. Buturla, P.E. Cottrell, B.M. Grossman, and K.A. Salsburg. Finite-element analysis of semiconductor devices: The fielday program. In *IBM J. Res. Develop.*, volume 25, pages 218–231, 1981.
- [17] Michael J. Caruso, Tamara Bratland, Carl H. Smith, and Robert Schneider. A new perspective on magnetic field sensing. In *Sensors*, December 1998.
- [18] P.C. Clark and L.J. Hoffman. BITS: a smartcard protected operating system. In *Communications of the ACM*, volume 37, pages 66–70, New York, NY, USA, 1994. ACM Press.
- [19] G3Card Consortium. 3rd generation smart card project. <http://www.g3card.org/>.
- [20] J. Daemen and V. Rijmen. Resistance against implementation attacks: A comparative study of the AES proposals. Proc. Second AES Candidate Conf., available at <http://csrc.nisc.gov/encryption/aes/round1/conf2/aes2conf.htm>, 1999.
- [21] P.E. Dodd and L.W. Massengill. Basic mechanisms and modeling of single-event upset in digital microelectronics. In *IEEE Transactions on Nuclear Science*, volume 50, pages 583–602, 2003.
- [22] P.E. Dodd, F.W. Sexton, G.L. Hash, M.R. Shaneyfelt, B.L. Draper, A.J. Farino, and R.S. Flores. Impact of technology trends on SEU in CMOS SRAMs. In *IEEE Transactions on Nuclear Science*, volume 43, pages 2797–2804, 1996.
- [23] Thales e Security. White paper: Smart cards for payment systems. http://www.thales-ecurity.com/Whitepapers/documents/Smart_cards_for_payment_systems.pdf.
- [24] J. Fournier, S. Moore, H. Li, R. Mullins, and G. Taylor. Security evaluation of asynchronous circuits. In *Proceedings of Cryptographic Hardware and Embedded Systems - CHES2003*, pages 137–151, 2003.
- [25] K. Gandolfi, C. Mourtel, and F. Olivier. Electromagnetic analysis: Concrete results. In *Proceedings of Cryptographic Hardware and Embedded Systems - CHES2001*, pages 251–261, 2001.
- [26] G. Gaskell and M. Looi. Integrating smart cards into authentication systems. In *Proc. of the 1st International Conference on Cryptography: Policy and Algorithms*, pages 270–281, 1995.
- [27] C. Gebotys, S. Ho, and C. Tiu. EM Analysis of Rijndael and ECC on a Wireless Java-based PDA. In *Proceedings of Cryptographic Hardware and Embedded Systems - CHES2005*, 2005.
- [28] C.W. Gwyn, D. L. Scharfetter, and J. L. Wirth. The analysis of radiation effects in semiconductor junction devices. In *IEEE Transactions on Nuclear Science*, volume 15, pages 153–169, 1967.

- [29] M. Hasan. Power analysis attacks and algorithmic approaches to their countermeasures for koblitz curve cryptosystems. In *IEEE Transactions on Computers*, volume 50, pages 1071–1083, October 2001.
- [30] S. Hayashi and M. Yamada. EMI-noise analysis under ASIC design environment. In *IEEE Trans. Computer-aided Design of Integrated Circuits and Systems*, volume 19, pages 862–867, 2000.
- [31] A.H. Johnston. Charge generation and collection in p-n junctions excited with pulsed infrared lasers. In *IEEE Transactions on Nuclear Science*, volume 40, pages 1694–1702, 1993.
- [32] M. Joye and C. Tymen. Protections against differential analysis for elliptic curve cryptography. In *Proceedings of Cryptographic Hardware and Embedded Systems - CHES2001*, pages 377–390, 2001.
- [33] T.M. Jurgensen, S.B. Guthery, T. Jurgensen, and S. Guthery. *Smart Cards: The Developer's Toolkit*. Prentice Hall PTR, 2002.
- [34] W. Kinsner. Smart cards. <http://www.ee.umanitoba.ca/~kinsner/whatsnew/tutorials/tu1999/smcards.html>.
- [35] P. Kocher. Cryptanalysis of Diffie-Hellman, RSA, DSS, and other cryptosystems using timing attacks. In *Proceedings of 15th International Advances in Cryptology Conference – CRYPTO '95*, pages 171–183, 1995.
- [36] P. Kocher, J. Jaffe, and B. Jun. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In *Proceedings of 16th International Advances in Cryptology Conference – CRYPTO '96*, pages 104–113, 1996.
- [37] P. Kocher, J. Jaffe, and B. Jun. Differential Power Analysis. In *Proceedings of 19th International Advances in Cryptology Conference – CRYPTO '99*, pages 388–397, 1999.
- [38] J.P. Kreskovsky and H.L. Grubin. Numerical simulation of charge collection in two- and three-dimensional silicon diodes – a comparison. In *Solid-State Electron.*, volume 29, pages 505–518, 1986.
- [39] Security Magnetics Pty Ltd. Watermark magnetics card technology. <http://www.securitymagnetics.com.au/content/techwatermark.html>.
- [40] M.S. Lundstrom. *Fundamentals of Carrier Transport*. Addison-Wesley Publishing Company, 1990.
- [41] C. Giraud M. Akkar. An implementation of DES and AES, secure against some attacks. In *Proceedings of Cryptographic Hardware and Embedded Systems - CHES2001*, pages 309–318, 2001.
- [42] L. W. Massengill. SEU modeling and prediction techniques. In *IEEE NSREC Short Course*, pages III–1 – III–93, 1993.

- [43] D. May, H.L. Muller, and N.P. Smart. Random register renaming to foil dpa. In *Proceedings of Cryptographic Hardware and Embedded Systems - CHES2001*, 2001.
- [44] K. Mayaram, J. H. Chern, and P. Yang. Algorithms for transient three dimensional mixed-level circuit and device simulation. In *IEEE. Trans. Computer-Aided Design*, volume 12, pages 1726–1733, 1993.
- [45] D. McMorrow, J. S. Melinger, and S. Buchner. Application of a pulsed laser for evaluation and optimization of SEU-Hard designs. In *IEEE Transactions on Nuclear Science*, volume 47, pages 559–565, 2000.
- [46] G.C. Messenger. Collection of charge on junction nodes from ion tracks. In *IEEE Transactions on Nuclear Science*, volume 29, pages 2024–2031, 1982.
- [47] T.S. Messerges, E.A. Dabbish, and R.H. Sloan. Examining smart-card security under the threat of power analysis attacks. In *IEEE Transactions on Computers*, 2002.
- [48] S. Moore, R. Anderson, P. Cunningham, R. Mullins, and G. Taylor. Improving smart card security using self-timed circuits. In *Eighth IEEE International Symposium on Asynchronous Circuits and Systems*, 2002.
- [49] A.Y. Nikiforov, A.I. Chumakov, and P.K. Skorobogatov. CMOS IC’s transient radiation effects investigations, models verification and parameter extraction with the test structures laser simulation tests. In *Proceedings of the 1996 IEEE International Conference on Microelectronic Test Structures*, pages 253–258, 1996.
- [50] M. Kuhn O. Kömmerling. Design principles for tamper-resistant smart-card processors. In *USENIX Workshop on on Smartcard Technology*, pages 1–11, May 1999.
- [51] J.L. Pankove. *Optical Processes in Semiconductors*. Prentice-Hall, New Jersey, 1971.
- [52] S.T. Pantelides, A. Selloni, and R. Car. Energy gap reduction in heavily doped silicon: causes and consequences. In *Solid-State Electronics*, volume 28, pages 17–24, 1985.
- [53] M.M. Parish and P.B. Littlewood. Non-saturating magnetoresistance in heavily disordered semiconductors. In *Nature*, volume 426, pages 162–165, 2003.
- [54] J-J. Quisquater and D. Samyde. ElectroMagnetic Analysis (EMA): Measures and counter-measures for smart cards. In *E-smart*, pages 200–210, 2001.
- [55] W. Rankl and W. Effing. *Smart Card Handbook, 2nd ed.* John Wiley & Sons, 2000.
- [56] J.G. Rollins and Jr. J. Choma. Mixed-mode pisces-spice coupled circuit and device solver. In *IEEE. Trans. Computer-Aided Design*, volume 7, pages 862–867, 1988.
- [57] S. Selberherr. *Analysis and Simulation of Semiconductor Devices*. Vienna, Austria: Springer-Verlag, 1984.
- [58] S. Skorobogatov and R. Anderson. Optical fault induction attacks. In *Proceedings of Cryptographic Hardware and Embedded Systems - CHES2002*, pages 2–12, 2002.

- [59] E. Sprunk. Clock frequency modulation for secure microprocessors. In *US Patent 5404402*, filed December 1993.
- [60] Jennifer G. Steiner, B. Clifford Neuman, and Jeffrey I. Schiller. Kerberos: An authentication service for open network systems. In *USENIX Winter*, pages 191–202, 1988.
- [61] I. Straus. Near and far fields – from statics to radiation. <http://www.conformity.com/0102reflections.html>.
- [62] K. Tiri, M. Akmal, and I. Verbauwhede. A dynamic and differential CMOS logic with signal independent power consumption to withstand differential power analysis on smart cards. In *Proc. IEEE 28th European Solid-state Circuit Conf. (ESSCIRC'02)*, 2002.
- [63] Y. Tsunoo, T. Saito, T. Suzaki, M. Shigeri, and H. Miyauchi. Cryptanalysis of DES implemented on computers with cache. In *Proceedings of Cryptographic Hardware and Embedded Systems - CHES2003*, 2003.
- [64] N. Weste and K. Eshraghian. *Principle of CMOS VLSI Design, 2nd ed.* Addison Wesley, 1994.
- [65] The free encyclopedia Wikipedia. Kerckhoffs' law. http://en.wikipedia.org/wiki/Kerckhoffs'_law.
- [66] The free encyclopedia Wikipedia. Security through obscurity. http://en.wikipedia.org/wiki/Security_by_obscurity.
- [67] Philips Semiconductors Leads Industry with Smart Card Security Benchmark. Product news from Philips semiconductors. http://www.semiconductors.philips.com/news/content/file_354.html, October, 1998.