Spring 2015

HARDWARE ATTACK DETECTION AND PREVENTION FOR CHIP SECURITY

Jaya Dofe
University of New Hampshire, Durham

HARDWARE ATTACK DETECTION AND PREVENTION FOR CHIP SECURITY

BY

Jaya Dofe

B.E., S.S.G.M.C.E, Shegaon, Maharashtra, India, 2004

M.E.,M.I.T.A.O.E, Pune, Maharashtra, India, 2012

THESIS

Submitted to the University of New Hampshire

in Partial Fulfillment of

the Requirements for the Degree of

Master of Science

in

Electrical Engineering

May, 2015

This thesis has been examined and approved in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering by:

Thesis Director, Qiaoyan Yu, Ph.D.
Assistant Professor
Department of Electrical & Computer Engineering

Thomas Miller, Ph.D.
Professor
Department of Electrical & Computer Engineering

John LaCourse, Ph.D.
Professor
Department of Electrical & Computer Engineering

On February 26, 2015

Original approval signatures are on file with the University of New Hampshire Graduate School.

ACKNOWLEDMENTS

First and foremost, I would like to express my deepest gratitude to my thesis advisor Dr. Qiaoyan Yu whose encouragement, guidance and support from the initial to the final level enable me to develop a profound understanding of the area. I would also like to thank Dr. Thomas Miller and Dr. John LaCourse for their willingness to serve on my thesis committee, their constructive and insightful comments.

I would also like to thank my fellow graduate students in the UNH Reliable VLSI Systems Lab: Hoda Pahlevanzadeh, Jiawei Zhong, Patrick Nsengiyumva, Jonathan Frey and Raashid Ansari. Their advice and help were instrumental in providing such a constructive environment.

Many friends Sanjana Seetharam, Kamini Yadav, Madhuri Jois have helped me stay sane through these difficult years. Their support and care helped me overcome setbacks and stay focused on my graduate study.

Finally, I wish to thank my loving husband Harshal Dofe. He was always there for cheering me up and stood by me through the good times and bad. I would like to thank my family for being my greatest supporter and a source of inspiration throughout this process.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

3D-IC	Three- Dimensional Integrated Circuit
AES	Advanced Encryption Standard
ASIC	Application-Specific Integrated Circuit
ATPG	Automatic Test Pattern Generation
BEOL	Back-End-Of Line
CMOS	Complementary Metal-Oxide Semiconductor
CMP	Chemical Mechanical Polishing
CRP	Challenge-Response Pair
DCVSL	Differential Cascade Voltage Switch Logic
DMR	Double Modular Redundancy
DPA	Differential Power Analysis
DUT	Design Under Test
EPC	Even parity check codes
FEOL	Front-End-Of-Line
FPGA	Field-Programmable Gate Array
FSM	Finite State Machine
HTs	Hardware Trojans
ICs	Integrated Circuits
IP	Intellectual Property
LDPC	Low-Density Parity-Check
NMOS	n-type Metal Oxide Semiconductor

PMOS	p-type Metal Oxide Semiconductor
PUF	Physical Unclonable Function
SCA	Side Channel Analysis
SoC	System on chip
TEAM	ThrEshold Adaptive Memristor model

ABSTRACT

HARDWARE ATTACK DETECTION AND PREVENTION FOR CHIP SECURITY

by

Jaya Dofe

University of New Hampshire, May 2015

Hardware security is a serious emerging concern in chip designs and applications. Due to the globalization of the semiconductor design and fabrication process, integrated circuits (ICs, a.k.a. chips) are becoming increasingly vulnerable to passive and active hardware attacks. Passive attacks on chips result in secret information leaking while active attacks cause IC malfunction and catastrophic system failures. This thesis focuses on detection and prevention methods against active attacks, in particular, hardware Trojan (HT). Existing HT detection methods have limited capability to detect small-scale HTs and are further challenged by the increased process variation. We propose to use differential Cascade Voltage Switch Logic (DCVSL) method to detect small HTs and achieve a success rate of 66% to 98%. This work also presents different fault tolerant methods to handle the active attacks on symmetric-key cipher *SIMON*, which is a recent lightweight cipher. Simulation results show that our Even Parity Code SIMON consumes less area and power than double modular redundancy SIMON and Reversed-SIMON, but yields a higher fault -detection-failure rate as the number of concurrent faults increases. In addition, the emerging technology, memristor, is explored to protect *SIMON* from passive attacks. Simulation results indicate that the memristor-based SIMON has a unique power characteristic that adds new challenges on secrete key extraction.

Chapter 1: Introduction

1.1 Security Challenges on Chip Design

Computers, tablets, cloud computing, social network media (e.g. Facebook, Twitter, and LinkedIn), cars, TV sets, mobile phones, mp3 players, washing machines, microwave ovens, phone cards and many other systems with silicon chips are gradually changing the way we are used to live with. All those fascinating devices and systems fundamentally rely on one thing—integrated circuits (ICs, a.k.a. chips). Silicon chips perform key roles in military, government, commercial products, phone links, banking networks, electronic grids and nuclear power plants. Due to escalating manufacturing costs, increasing semiconductor technologies are often available at offshore foundries. Utilizing these facilities significantly limits the trustworthiness of the corresponding integrated circuits for critical applications. Counterfeit components are most often introduced into the supply chain through non-authorized, gray market sources. Moreover, device scaling is resulting in smaller devices that are increasingly vulnerable to faults due to phenomena such as high energy particle strikes, infant mortality, design defects, wear-out, and fault attacks which is a further challenge for chip's safety.

With constantly growing demand, security of silicon chips has caught increasing attentions in military, government, bank system, and even daily electronic products. Unintended system behavior induced by unsecure chips could have severe consequences, such as heavy

financial damage, loss of human life, threat to national security, etc. Hardware counterfeiting and intellectual property (IP) piracy is another serious issue, for instance costing the U.S. economy more than $200 billion annually [39] on defense and military related chips.

Security design engineers are trying their best for integrated circuit (IC) protection; unfortunately, security challenges remain in practical applications. For examples, although smartcard systems have utilized hardware authentication to strengthen the system security, chips embedded in the smartcard system may still be hacked or cloned by adversaries. High-performance computers significantly reduce the computation time on the key guessing process. Mature side-channel analysis techniques further shorten the period of secret key extraction. As a result, the 'secure' Smartcard system is not truly secure. Moreover, ICs are vulnerable to different types of emerging attacks. A complicated IC can even be compromised by an attack that is realized with a minor silicon cost, which is not noticeable in post-silicon test stages. As the diversity of hardware attacks increases and the attack cost tends to be smaller and smaller, it is imperative to investigate efficient techniques to tackle various hardware attacks.

1.1.1 Diverse Hardware Attacks

Hardware security has become a cause for serious concern in a variety of IC design, fabrication, and testing scenarios. Hardware security can be tampered by various attacks .These attacks relate to malicious modifications of an IC or stealing the hardware details during design or fabrication, in an untrusted design house or foundry. Such modifications can give rise to undesired functional behavior of an IC, or provide covert channels or backdoor through which sensitive information can be leaked. The following attacks are possible in ICs.

(1) Hardware Trojans (HTs): HTs are malicious additions or modifications to the circuit design

that alter the original function or steal the secret.

(2) Side-channel analysis: An attacker can extract the secret information (e.g. secret key for a cipher) by using physical parameters like power, delay, or electromagnetic emission of the IC, which runs a security-critical application.

(3) IP piracy: A malicious silicon foundry may produce excessive ICs than what was quoted originally. Then, the over-produced chips are sold in gray market without the prior permission from the chip designer.

(4) Reverse engineering: An attacker can use reverse engineering techniques on chips to steal design details and then reproduce the chips.

(5) Counterfeiting: An attacker imitates the original chip. Because of counterfeiting, the suppliers of the original components suffer loss. The poor performance of fake products, which are commonly with lower quality or older technology than the original product, adversely influence the overall system performance, reliability, and security.

(6) Fault attacks: Attacker insert faults in the target hardware mainly used for cryptography to leak or steal the secret information. Consequently, cryptographic hardware need to be protected against fault attacks as well as natural phenomena such as radiation-induced soft errors, power supply noise or crosstalk. One-bit flip in cryptographic hardware can cause multiple-bit changes at the output and thus crush the entire security defense line of the system.

Typical countermeasures for hardware attacks are watermarking, fingerprinting, obfuscation, side channel analysis, reverse engineering, split manufacturing, and camouflaging. We discuss the representable techniques in detail in Chapter 2.

1.2 Thesis Contributions

There are varieties of stages in the IC design cycle like specification, design, fabrication, testing, and assembly as shown in Fig. 1.1. Due to the complexity and shorter design cycle, the fabrication process is outsourced offshore. The outsourcing gives third-party fabrication foundries an opportunity to embed unwanted or even malicious hardware into the chips. As seen from the flow mentioned in Fig. 1.1, all the stages except specification and package test are vulnerable to hardware attacks.

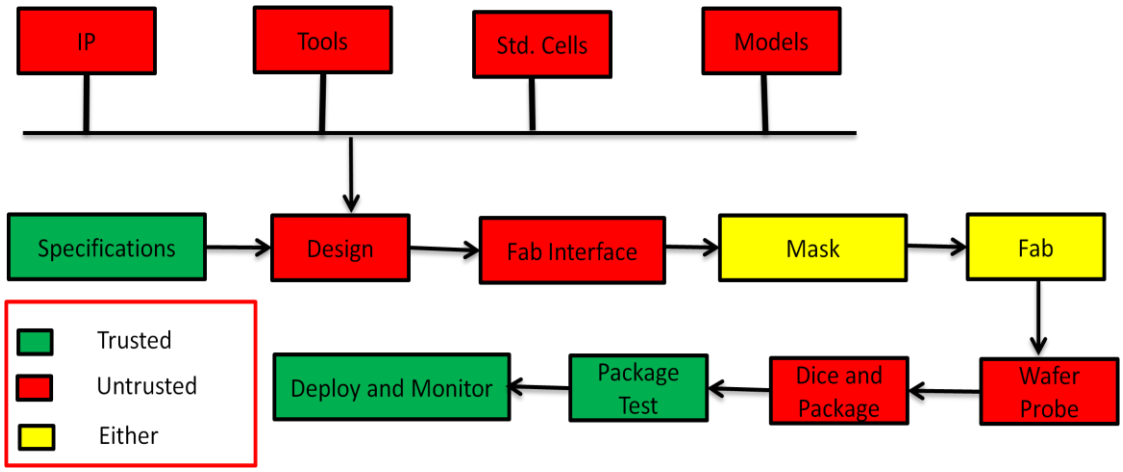


Figure 1.1 Vulnerable steps of modern IC life cycle [Source: R.S. Chakraborty et al. 2010]

Over past years, integrated circuits became increasingly more complex and expensive. The industry began to embrace new design and reuse methodologies that are collectively referred to as system-on-chip (SoC). SoC uses IP cores obtained from third parties, which may include embedded processors, memory blocks, interface blocks, analog blocks, and components that handle application specific processing functions. SoC are increasingly deployed in security and safety critical environment, hence SoC's safety itself is major concern. As SoC consist of many

cores and different components, single solution will not work for different attacks on different modules. We need to think about security assistance that will assure the security of entire SoC.

This work is associated with security against passive and active attack, which can hinder the security of chip. A passive attack attempts to listen or make use of information from the system but does not affect system resources. These attacks are difficult to detect because they do not involve any alteration of the data or hardware. Active attacks involve some modification in hardware or behavior of the system. The work mainly includes three contributions.

1) Exploration of new Hardware Trojan detection method

The first contribution is exploiting the inherent feature of *Differential Cascade voltage switch logic (DCVSL)* logic family to detect HTs in runtime. In normal operation, a system implemented with DCVSL always produces complementary logic values in internal nets and final outputs. Non-complementary values on inputs and internal nets in DCVSL systems potentially result in abnormal power behavior and even system failures. By examining special power characteristics of DCVSL systems upon HT insertion, we can detect HTs, even if the HT size is small.

2) Fault tolerance methods against fault attack designed for new lightweight cipher SIMON

The second contribution is investigating three fault-tolerant techniques - double-modular redundancy (DMR), reverse function, and even parity check code combined with a non-linear compensation function (EPC) for lightweight block cipher SIMON to detect the faults.

3) Exploiting emerging technology memristor for design of block cipher SIMON

The third contribution is exploring emerging devices to model different logic circuit and examine the power characteristics of the memristor itself and memristor-based logic gates. We further study the feasibility of utilizing memristors to implement a new block cipher, SIMON.

1.3 Thesis Organization

The rest of the thesis is organized as follows. Background about hardware attack, its impacts, and solutions are presented in chapter 2. The current fault-tolerant methods for encryption/ decryption algorithms are reviewed. The memristor model is introduced in Chapter 2, as well.

In chapter 3, design of logic circuits using DCVSL logic is presented. The discussion about power characteristics of logic circuit with and without HT is carried out. Further the impact of HT insertion location on circuit power and HT detection rate on DCVSL-based 64 bit full adder and benchmark circuits are described.

In chapter 4, the implementations of three-fault tolerance techniques - DMR, reverse function, and a parity check code combined with a non-linear EPC for the lightweight block cipher SIMON are discussed. Area and power for different fault tolerant implementations are compared. The impact of fault locations on fault detection rate is conferred for three different approaches.

In chapter 5, we present the memristor model and design of memristor-based logic gates. The unique power characteristics of memristor and memristor based logic gates are broadly studied. Further, the feasibility of utilizing memristors to implement a new block cipher SIMON is presented.

Chapter 6 summaries the major contributions of this thesis and limitation and further shed some light on future research directions.

Chapter 2: Background

In this chapter, chip security authentication approaches, diverse hardware attacks and associated countermeasures are presented.

2.1 Current Methods for Chip Security and Authentication

Increasing practice of outsourcing, increases threats to security of chips. Various stages of an IC lifecycle are vulnerable to attacks as discussed in Chapter 1. This section confers different techniques used for chip security and authentication.

2.1.1Watermarking

As defined in [71], a design watermark is an invisible (i.e., imperceptible to human or machine analysis) identification code that is permanently embedded as an integral part within a design. The designer can later reveal the watermark and claim his/her ownership of an IC/IP. Watermarks may include addition of black-hole states to the finite state machine (FSM) [1], addition of secret constraints during high level [3], logic and physical synthesis [13], and field-programmable gate array (FPGA) design [14].

2.1.2 Fingerprinting

Fingerprinting helps the defender to track the source of piracy by embedding the signature of the buyer (for instance, his public key) along with the watermark of the designer

[15]. When challenged, the designer can reveal the watermark to claim the ownership and the buyer's signature to reveal the source of piracy. For example, the power, timing, or thermal fingerprint of an IC is revealed on applying a set of input vectors. Similar to watermarking, fingerprinting can also be applied at high level, logic level, and physical synthesis level [15]. The recent Defense Advanced Research Projects Agency (DARPA) Supply Chain Hardware Integrity for Electronics Defense (SHIELD) program aims at uniquely identifying chips, but the effectiveness of the approach is yet to be seen [41].

2.1.3 Obfuscation

Obfuscation is a technique that transforms an application or a design into one that is functionally equivalent to the original but is significantly more difficult to reverse engineers [74]. Obfuscation hides the functionality and implementation of a design by inserting additional gates into the original design. In one type of obfuscations, xor / xnor gates [75, 76] and memory elements [77] are added. The obfuscated design will function correctly only when the unique input value is applied to these gates and memory elements. In another type of obfuscations, the FSM of the design is obfuscated. An FSM can be obfuscated by adding extra states and/or transitions into it. In software tools, the hardware description source code can be obfuscated with variable renaming, removal of comment and loop unrolling to decrease the comprehensibility of source code without changing the functionality [63].

2.1.4 Split Manufacturing

Leading fabless semiconductor companies such as AMD and research agencies such as Intelligence Advanced Research Projects Agency (IARPA) have proposed split manufacturing to

thwart attacks like reverse engineering, malicious circuit modification and IP piracy [78]. In split manufacturing, the layout of the design is split into the front-end-of-line (FEOL) layers and back-end-offline (BEOL) layers. The FEOL and BROL layers are then fabricated separately in different foundries. Post fabrication, the FEOL and BEOL wafers are aligned and integrated together using either electrical, mechanical, or optical alignment techniques. The final ICs are tested upon integration of the FEOL and BEOL layers [79]. The asymmetric nature of the metal layers facilitates split manufacturing. Split manufacturing is practical [80]. Ideally, an attacker should not be able to retrieve the missing BEOL connections by knowing the FEOL layers [81].

2.1.5 Camouflaging

This is a layout-level technique to hamper image-processing-based extraction of gate-level netlist. In one embodiment of camouflaging, the layouts of standard cells are designed to look alike, resulting in incorrect extraction of the netlist. In traditional ways, the layout of NAND cell and the layout of NOR cell look different and hence their functionality can be extracted by pattern recognition. However, the layout of a camouflaged NAND cell and the layout of camouflaged NOR cell are almost identical; as a result, an attacker cannot unambiguously extract their functionality from layout patterns [82, 83]. One can also use dummy contacts or dummy contact gap in the middle layer to fake a connection between two metal layers for camouflaging [82]. Programmable standard cells can be used to camouflage a design [83].

2.1.6 Physical Unclonable Function (PUF)

A silicon PUF is a special circuit embedded in an electronic device that exploits manufacturing variations in order to generate a unique signature, identifier, or key for its native

device [72, 73]. The inputs and outputs of PUF circuits are called challenges and responses. An applied challenge and its measured response are referred to as a challenge-response pair (CRP). Ideally, manufacturing variations result in unique input/output behavior (i.e., set of CRPs) among all devices. Moreover, since many fabrication variations are random, the unique signature cannot be cloned or replicated even by the manufacturer. These features make PUFs promising for security applications such as authentication [73] and cryptographic key generation [72].

2.2 Emerging Technology

Over this forty-year span, the usage scenario of security technologies has evolved from securing physical premises with mainframe computers to securing lightweight, low-cost, high-performance, and low-power mobile phones, tablets, and sensors. Emerging technologies, acting as alternatives to CMOS logic, have already shown promising features for high performance circuit design. Classical security has created elegant security primitives and protocols. Unfortunately, these solutions are not only slow and consume significant amounts of energy for most modern applications but also vulnerable to physical and side channel attacks (e.g., radiation or exposure to high temperatures). Classical and emerging security requirements and metrics may be addressed in superior ways using emerging technologies [38].Some of emerging technology are mentioned below.

2.2.1 Three-Dimensional Technology

An exciting new development in IC manufacturing is the Three-Dimensional Integrated Circuit (3D-IC), a single circuit built by stacking and integrating separately-built layers. Although the 3D technology is known for its increased density, speed, and power conservation, it

also offers unique security advantages. One can also leverage the 3-D manufacturing technology: security-sensitive components are placed in one layer and manufactured in a trusted low-end foundry, and other components of the design are placed in another layer and manufactured in an untrusted high-end foundry [40].

2.2.2 Graphene Transistor

As MOSFET alternatives, tunneling based transistor technologies [16] are being actively investigated by device scientists. Among these devices is a double-layer graphene transistor often called as SymFET [16]. With extremely low performance overhead and little circuit redesign, SymFET is equipped with unique physical properties which may be leveraged by hardware security approaches to achieve various highly-efficient implementations for IP protection, Trojan detection, and side-channel attack prevention. The newly developed graphene SymFETs have a special property that the source drain current will be cut off if the source-drain voltage is outside a narrow voltage band [29].

2.2.3 Memristor

The memristor is identified as the fourth fundamental circuit element, the complement to resistor, capacitor and inductor. Unlike a resistor which has a fixed resistance value, the amount of the resistance in a memristor depends on the intensity and the direction of the current that has already passed through it [24]. By applying the appropriate electrical bias for the required duration, the device may be repeatedly switched between two resistance states: a high resistance state and a low resistance state. As a result, the memristor's resistance (i.e. memristance) can be rewritable, making it more appealing for the design of computer memory and non-volatile disk

storage. Memristor was first modeled with two adjustable resistors, as shown in Fig. 2.1 [24]. A memristor is simply a variable resistor that changes its resistance based on how much and in which direction of the device the current has flown. When current flows through the device through one terminal, the resistance approaches a resistance R_{on}, and the through the other terminal, a resistance R_{off}. When the device is turned off, it remembers the value of resistance it was prior to shutting off. The value of resistance between these two extremes, R_{on} and R_{off} is considered as a function of the device width. The HP Company successfully produced memristors by using Pt (highly doped) and TiO_2 (highly undoped) [25].

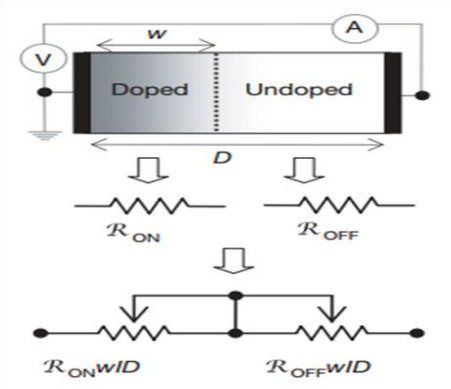


Figure 2.1 HP Memristor model [24].

2.3 Hardware Trojan Detection Approaches

The growing number of ICs manufactured offshore increases threats to the chip security [17-19, 26, 27]. Research has exposed an increase in existence of hardware Trojans (HTs), which are malicious additions or modifications to the original circuit design. The simple structure of HT is shown in Fig. 2.2. This insertion can occur at any stage in a production cycle, and could have devastating effects on the final design. Malicious inclusions of hardware have the potential

to degrade system performance, surreptitiously delete data, leave a backdoor for secret key leaking, or eventually destroy the chip [20, 21]. The threat from such maliciously inserted hardware is of increasing concern [22] to government and military agencies [2]. Recently, Skorobogatov et al. [4] demonstrated the presence of a backdoor in a military grade FPGA manufactured by Actel.

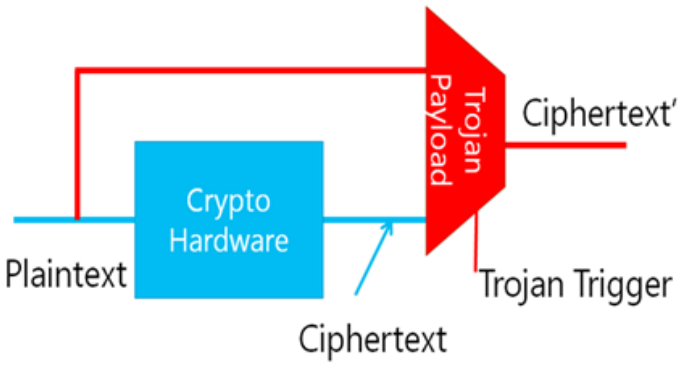


Figure 2.2 Simple example of Hardware Trojan

2.3.1 Power-Based Side Channel Analysis

Agrawal et al. were the first to use side-channel information to detect hardwareTrojan contributions to circuit power consumption [37]. The power signatures of golden version of ICs are obtained by applying random patterns. The related data for each input pattern of each power measurement consists of several elements like (1) power consumption of the circuit after applying inputs that are the same in all Trojan-free ICs, (2) measurement of noise which can be removed by several measurements, (3) process variations which are random and cannot be removed, and (4) Trojan contributions to the measured power consumption. HTs are detected if the power signature of chip under testing is different from the golden reference. The general side channel based HT detection method is shown in Fig. 2.3. If the HT size is very small its effect may be restrained by process variation.

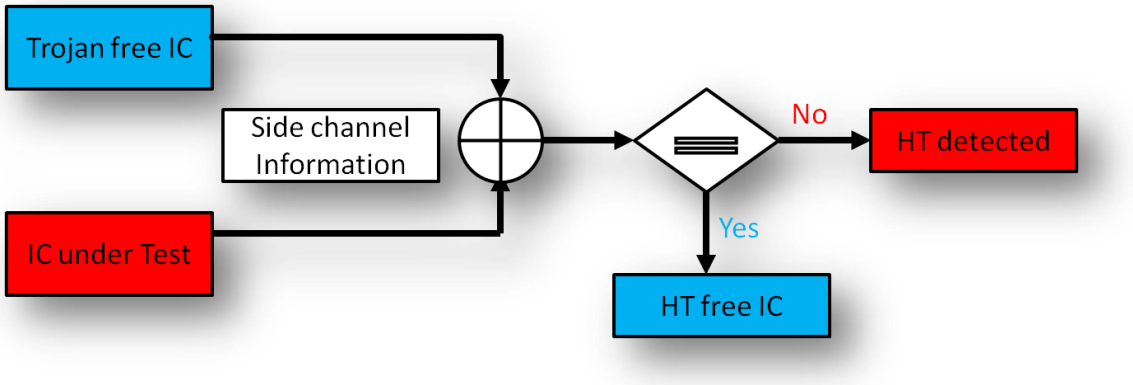


Figure 2.3 Side channel based HT detection approach

2.3.2 Delay-Based PUF

Li and Lach proposed a delay-based physical unclonable function (PUF) for hardware Trojan detection [34]. This method uses a sweeping-clock-delay measurement technique to measure selected register-to-register path delays, as shown in Fig. 2.4. The components inside the dotted box are part of main circuit and the parts outside the box are extra components for HT detection. The registers on the main circuit path are triggered by the main system clock (CLK1). The shadow register takes the same input as the destination register in the main circuit but is triggered by the shadow clock (CLK2), which runs at the same frequency as CLK1 but at a controlled phase offset. HTs can be detected when one or a group of path delays are extended beyond the threshold determined by the process variations level.

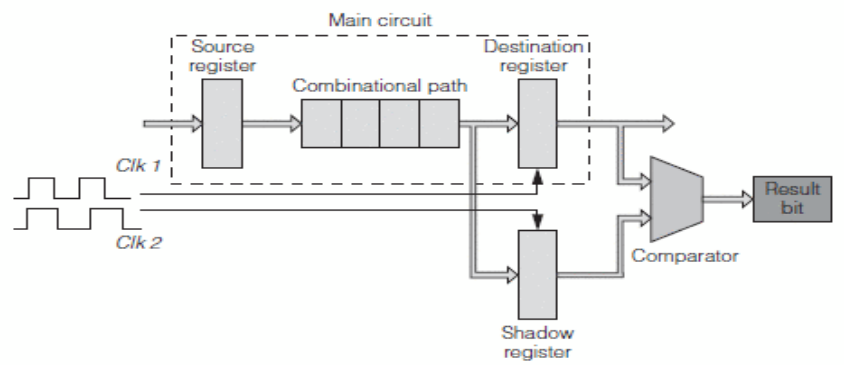


Figure 2.4 Path delay circuit using shadow register [34]

2.3.3 Automatic Test Pattern Generation

Automatic Test Pattern Generation (ATPG) methods are used for HT detection, as well [36]. The digital stimulus is applied on the circuit under test and the digital output is verified with the golden version. The digital stimulus is derived using the netlist of the chip. ATPG approaches work well for a functional unit with a small set of inputs, as the probability of rare events is relatively high. When the circuit complexity increases, the number of test vectors for ATPG will significantly increase to an unaffordable degree.

2.3.4 Destructive Reverse Engineering

HTs can be detected by destructive approaches which involves depackage an IC and obtain microscopic images of each layer to reconstruct the design for trust validation of the end product. One of the approaches is a chemical mechanical polishing (CMP) method [28]. In addition to being expensive, this type of technique is also time-consuming (takes several months) and loses its efficiency when the transistor density increases. Although this method can detect with 100% probability but the IC will be no more of use and only applicable to one IC.

2.4 Cryptography

The information shared over the internet is one of the valuable assets, which need to be protected from the unauthorized access. Cryptography, the science of encryption, plays a central role in various applications in our daily lives, such as mobile phone communications, pay-TV, e-commerce, and security of ATM cards. Encryption is an effective and popular technique for discouraging illegal copying and information distribution. With the rapid development of Internet technology, communication using multimedia implement secure communication. On the other hand, some data transmitted are characterized in terms of privacy, integrity and authenticity thought public.

In cryptography, the AES, also known as Rijndael, is a block cipher adopted as an encryption standard by the US government, which specifies an encryption algorithm capable of protecting sensitive information [35]. Encryption converts data to an unintelligible form called ciphertext. Decryption of the ciphertext converts the data back into its original form, which is called plaintext, shown in Fig. 2.5. Existing cryptographic algorithms were, for the most part, designed to meet the needs of the desktop computing era. Such cryptography tends not to be particularly well suited to the emerging era of embedded systems, where the overhead of hardware and software-based cryptography are highly constrained.

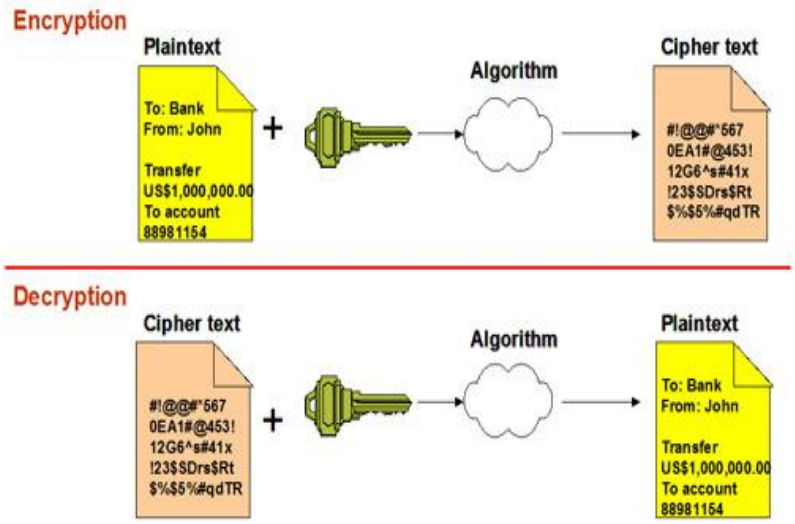


Figure 2.5 Symmetric key cryptography

2.5 Fault-Tolerance Methods

Although the security primitive's hardware is designed to withstand the linear and differential attacks, the security of encrypted messages is not guaranteed. Bit flips occurring during the encryption due to soft error or purposely invoked by an attacker are a major security concern. It can significantly endanger integrity, privacy, and confidentiality and hence the security of the system. Therefore, techniques to increase the reliability (fault-tolerance) and security of cryptographic systems are necessary.

Traditional fault-tolerant methods have been widely investigated for the implementation of cipher codes, such as Advanced Encryption Standard (AES) [43, 46, 48, 51]. Reverse functions are used to recover the original input for the encryption/decryption process; the recovered input is compared to the original one to detect faults in AES [42]. These hardware-redundant-based concurrent error detection schemes [47, 49] can be applied to the internal function, round, or entire encryption process levels. Another category of fault detection methods

for AES is based on parity codes [43, 45, 48-51]. The parity check codes can be applied to each byte of the state matrix, each transformation, after each round, or at the end of encryption process. Because of the wide input and key width (typical value is 64, 128, or 256), simple parity check codes are not sufficient to detect multiple faults. To increase fault detection capability, multiple groups of simple parity check codes or complicated parity codes are needed, at the cost of dramatic increase on hardware overhead. Low-density parity-check (LDPC) is incorporated in sequential circuits to mitigate the fault attacks [44]. The common challenge in the application of coding for error correction is the non-linear transformation in cipher codes. Fault attacks are serious threat to cryptographic systems, which can use the faulty ciphertext to retrieve the key [85].

2.6 Chapter Summary

In this chapter, the essentials about the hardware security, authentication, fault tolerance and cryptography are discussed. The representable HT detection approaches are presented as well. We identify the limitation of the existing HT detection methods as follows.

1. The sensitivity of SCA-based methods is challenged by the increasing process variation.
2. False detection on small HTs can happen when process variation effects exceed the signal threshold (e.g. power) for side-channel analysis.
3. Most of the existing approaches mentioned above are time consuming and expensive.
4. Reverse engineering techniques and SCA techniques requires golden model for comparison.
5. PUF based schemes require extra circuitry to be embedded into the actual hardware, which increases the size of IC.

6. Due to HTs furtive nature, activating arbitrary Trojan instances and observing their effects can be extremely difficult.

Next, the discussion about potential use of emerging technology for security applications is carried out. Lastly, the chapter mentions about the fault-tolerance method to protect cryptographic system against fault attack. Deliberate fault injection into a cryptographic device and the observation of the corresponding erroneous outputs can lead to the retrieval of secret key by use of cryptanalysis.

Chapter 3: Hardware Trojan detection using Differential Cascade Voltage Switch Logic (DCVSL)

In this chapter, the proposed HT detection method is presented. It highlights the basis for abnormal power consumption in DCVSL, and introduces the proposed HT detection method. Later the thorough evaluation of the area, power and HT detection rate of proposed method in full adders and ISCAS benchmark circuits is presented.

3.1 Method Overview

HT detection is typically carried out during test stages, when numerous test vectors are simultaneously applied to both the device under test (DUT) and a golden reference. As it is difficult to obtain a golden version, the behavioral model is often used as a reference. Because of the effects of process variation and imperfect device libraries for computer-aided design tools, a behavioral model based golden version is not precise enough to detect small and ultra-small HTs. Moreover, due to the demand of short time-to-market, the verification and testing period has been reduced significantly. Although it is always desired, thorough testing is not economically feasible. It is imperative to develop a HT detection method that is not limited by the HT size and does not take very long time to perform chip testing and authorization.

We propose a HT detection method that allows users to detect potential HTs *at runtime* and *without a golden reference*. The proposed method exploits the abnormal instantaneous power for DUT to detect small HTs. Figure 3.1 shows the overview of the proposed method. Given a stable supply voltage, we examine the current through the current monitor for abnormal power behavior. A notable difference with offline power-based side channel analysis methods is that we are not interested in a particular power value; instead, *the current monitor detects the current (we can interpret it to power consumption) staying at a constant high value for a relatively long duration*. As shown in Fig. 3.1, the current monitor will trigger an alarm circuit, when the power value falls in and remains in the blue shadow region for a relatively long period. This duration is comparable with the duration of an input vector, rather than input rising and fall times. The triggered HT in the DUT causes the abnormal power period. We propose to implement DUTs with DCVSL, which always produces Out and Out bar, a pair of complementary outputs. Such complementary outputs will be used as inputs for next stage. *In DCVSL, non-complementary inputs (invalid inputs) result in short-circuit power remaining for a long period of time until the non-complementary inputs disappear*. The proposed method exploits this inherent feature of DCVSL to detect the presence of HTs. Besides power detection, the proposed method further examines the complementary characteristic of the output pair, Out and $\overline{Out}$.

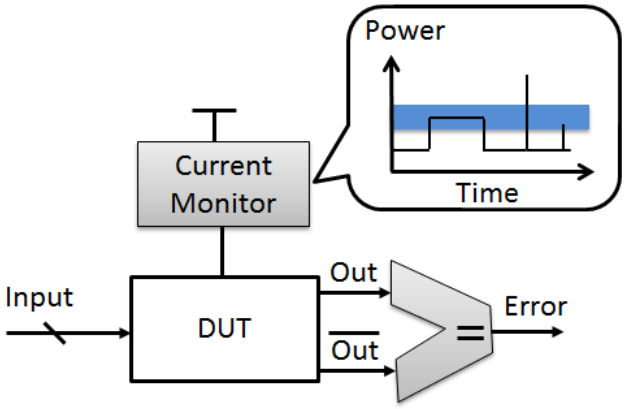


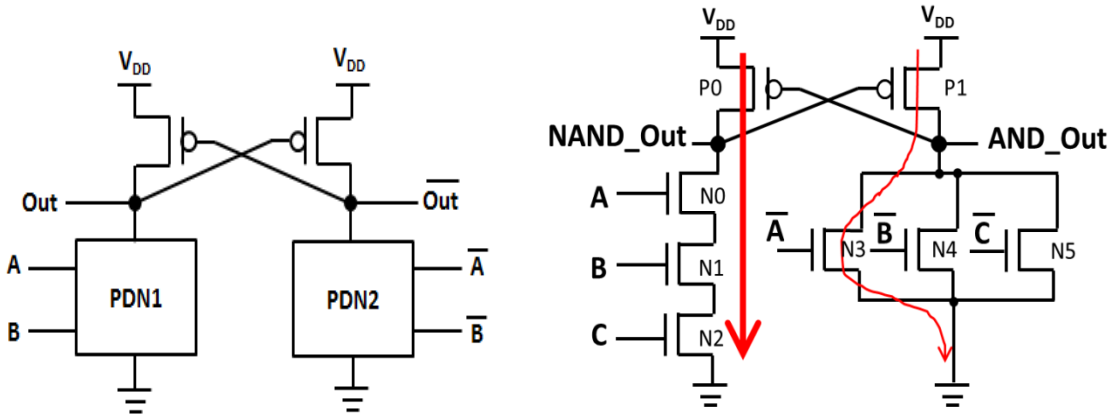
Figure 3.1 Proposed HT detection system.

The non-complementary output pair indicates a potential hardware Trojan insertion in the DUT. These non-complementary outputs can be utilized for HT detection when no abnormal power values appear due to HT being triggered. The current monitor is connected with the DUT on a separate platform at the user end. If the current monitor is integrated on the same chip with the DUT, potentially leaving opportunity for an attacker to tamper or remove the HT detection mechanism. A current sensor is needed to convert the transient current of the DUT and produce an analog voltage that is proportional to the measured DUT current. A programmed microcontroller can sample the analog voltage signal at specific intervals using interrupts. When the voltage value stays approximately constant for multiple interrupts, it indicates an abnormal short-circuit power due to a HT creating a short-circuit path from supply voltage V_{DD} to ground. The microcontroller can be further configured to set off an alarm or trigger a light-emitting diode to indicate HT detection to the user.

3.2 Circuit Power Based HT Detection

3.2.1 Unique Short-Circuit Power in DCVSL

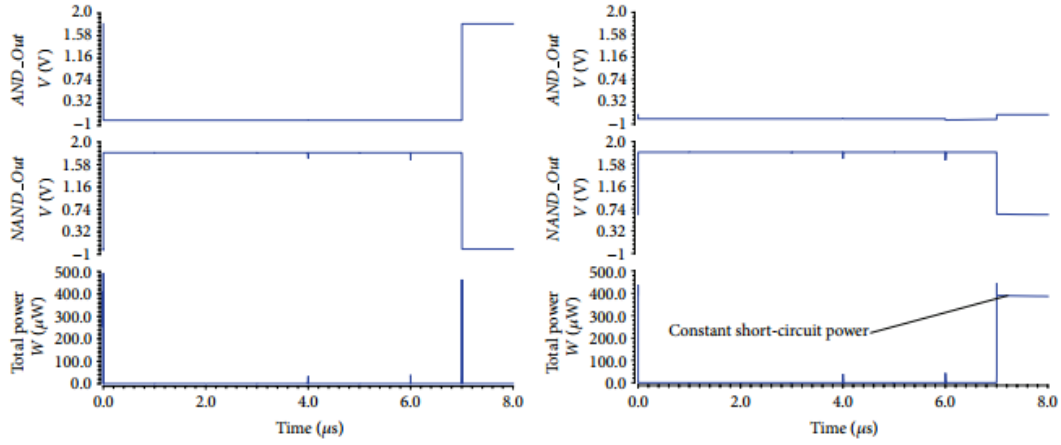
Each DCVSL gate needs complementary inputs and produces complementary outputs [30], as shown in Fig. 3.2(a). In normal operation, short-circuit power consumption of DCVSL gate is close to that of CMOS logic gate, as the time period for the direct current path from V_{DD} to ground is extremely small compared with that in switching and steady state conditions. When the input pair is non-complementary (both inputs being either logic 0 or logic 1), a DCVSL gate loses its complementary nature of output. More specifically, the output pair may be non-complementary, resulting in the short-circuit power consumption lasting for a significantly longer time than the case with complementary inputs. Take a 3-input NAND-AND gate as an example.



(a) (b)

Figure 3.2 DCVSL logic gates (a) General gate structure, and (b) circuit schematic of NAND3-AND3. Current track highlighted in the figure is for non-complementary inputs on A and $\bar{A}$.

The circuit schematic is shown in Fig. 3.2(b). In normal operation conditions, we give the input vector of A=B=C=1 and $\bar{A}$.= $\bar{B}$=$\bar{C}$=0. The NAND_Out port is pulled down to logic low through NMOS transistors N0, N1 and N2; this in turn activates PMOS transistor P1. As P1 is turned on, the AND_Out node is pulled to logic high and thus P0 is turned off. The time period when both PMOS and NMOS transistors are on is extremely small. Let us reconsider the 3-input NAND-AND gate with the same input vector, except that we make A=$\bar{A}$=1. Now, there exist two paths from VDD to the ground terminal: one is through N0, N1, N2 and another one is through N3. The path through N0, N1 and N2 pulls the NAND_Out port low as before, which turns on P1. P1 then tries to pull the AND_Out port high. At the same time, the path to ground through N3 tries to pull the AND_Out node low. If N3 is stronger than P1 (which is typically the case), the AND_Out port is pulled low and this activates P0. Therefore, a path from VDD to ground is created through P0, N0, N1, and N2, resulting in a high and constant short-circuit power. The constant short-circuit power remains as long as the duration of the input vector. Figures 3(a) and (b) show the power waveforms with complementary and non-complementary inputs, respectively. As shown in Fig. 3.3 (b), in the duration of the input vector A =$\bar{A}$ =B = C = 1(from 7 to 8 μs on the time axis), the peak power has a constant high value. This is because the non-complementary input pair (A=A$\bar{}$) makes NAND_Out and AND_Out both stay at logic low The time from 7-8 us represents the high time of the shortest input pulse A. As a result, the two PMOS transistors, P0 and P1, are both turned on; thus the two current paths from VDD to ground (highlighted in Fig. 3.2 (b)) exist until the input vector is changed.



(a) (b)

Figure 3.3 Voltage and power waveforms for DCVSL NAND3-AND3 gate. (a) Complementary inputs, and (b) non-complementary input $A = \bar{A}$.

The amplitude of short-circuit power is typically three orders of magnitude higher than the leakage power. This significant power difference between the cases using complementary and non-complementary inputs is large enough for a monitoring device to indicate the presence of HT. We examine the average power for complementary and non-complementary inputs for basic DCVSL gates using a typical IBM7RF technology library. As shown in Table 3.1, the increase on the average power (averaging power for all possible input patterns) caused by non-complementary inputs is over three orders of magnitude. This is the basis for choosing DCVSL to implement functional units that facilitate HT detection. If the triggered HT flips the internal node of a functional unit, it will create a non-complementary signal in the middle of that functional unit. Consequently, the power consumption will stay high for a long time, which is different from normal switching power.

Table 3.1 Power increase caused by non-complementary inputs

Avg. Power / Logic Gates	Power for Complementary Inputs	Power for Non-Complementary Inputs
Inverter	20.51 nW	205.25 µW
NAND2-AND2	12.76 nW	92.84 µW
NOR2-OR2	11.85 nW	92.61 µW
XNOR2-XOR2	19.97 nW	171.2 µW
NAND3-AND3	7.501 nW	40.36 µW
NOR3-OR3	6.673 nW	39.81 µW
XNOR3-XOR3	16.49 nW	84.90 µW
D-Flip-Flop	17.23 nW	181.8 µW

3.2.2 Probability of Abnormal Short-Circuit Power

The key reason for DCVSL gate having abnormal short-circuit power is the non-complementary output nodes turning on the two PMOS transistors simultaneously. We assume the HT impact on DCVSL functional units is flipping one of the input logic. This is similar to HT insertion in other technologies, i.e. triggered HT is used to change the logic value of a logic gate or memory element. Because of electrical and logical masking, the non-complementary inputs (caused by HTs) do not always yield abnormal short-circuit power. As the logic gate

topology varies between gates, it is difficult to obtain a closed-form expression for the probability of abnormal power occurrence. We summarize the general procedure for how to analyze the HT detection probability in DCVSL systems through abnormal power observation. Figure 3.4 is the flowchart for the analysis procedure. In order to create an erroneous output in DCVSL, a HT has to make one or more inputs non-complementary. This may result in an erroneous output if the effect of the non-complementary input is propagated through the output port. An important point to note is that not all erroneous outputs are accompanied by abnormal power peaks. Only if the erroneous output creates at least one path from VDD to ground, will we observe the abnormal short-circuit power.

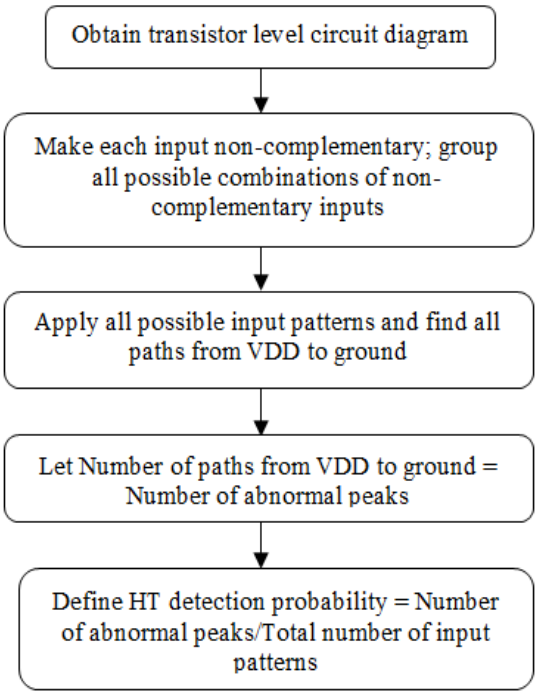


Figure 3.4 Flowchart for analyzing the HT detection probability for a DCVSL gate.

We examine the probability of abnormal power and output error occurrence for all input patterns. Table 3.2 shows the ratio of the total number of abnormal power peaks over the total number of all input patterns for various basic DCVSL gates.

Table 3.2 Probability of abnormal power and output error rate over all possible input patterns for DCVSL logic gates

DCVSL Gates	Percentage of abnormal power over all input patterns	Percentage of output error over all input patterns
Inverter	50.00%	100%
XOR2	41.66%	100%
XOR3	33.92%	100%
AND2	25.00%	25.00%
AND3	12.50%	12.50%
OR2	25.00%	50.00%
OR3	12.50%	25.00%
OAI21	23.21%	28.57%
AOI21	26.78%	50.00%
AOI22	25.89%	45.08%
OA22	25.89%	35.27%
MUX21	30.35%	55.00%
Average	27.72%	52.00%

The average probability for power exception and output mismatch are 27.7% and 52%, respectively. This means our HT detection method has over 50% chance to detect HTs, even if the HT trigger circuit is implemented with a single gate. This is a significant advantage over other power-based side channel analysis methods, which have a lower bound on the size of detectable HTs.

Moreover, we observe that abnormal power occurs more often on the input pattern that produces the rare output value. For example, an AND3 gate produces high output only when all three inputs are high; the abnormal power appears at the exact input pattern if one of the inputs is not in the complementary form. To hide a HT, hackers often utilize the rare case to trigger the HT. As discussed above, our approach inherently achieves a higher detection rate for the HT triggered by rare cases. This means a system equipped by our method will pose a greater challenge to attackers in order to obscure HTs.

3.3 Experimental Results

3.3.1 Experimental Setup

We evaluated the proposed method on the 64-bit ripple carry adder, ISCAS'85 and ISCAS'89 benchmark circuits. The schematic and layout of the 64-bit adder were implemented in Cadence Virtuoso with the IBM CMOS7RF technology. We set all transistor lengths to 220nm (minimum length in the CMOS7RF technology), and set the PMOS and NMOS transistor widths to 500nm and 600nm, respectively. The average power, leakage power and peak dynamic power were obtained from schematic-level simulations by examining all possible input patterns. The area for DCVSL modules was obtained from customized layout in Virtuoso. Five metal layers were used in layout design. The fastest switching period for input is 1μs. We synthesized

the Verilog codes of ISCAS benchmark circuits in Synopsys Design Compiler with IBM. CMOS7RF technology. The synthesized netlist is modified with an in-house python-based netlist generator, which converts CMOS netlist to DCVSL netlist. The behavior model of CMOS library is modified according to the gate output and power performance obtained from simulation in Cadence Virtuoso. HT detection rate is evaluated through gate-level simulation in Cadence NCVerilog.

Table 3.3 Number of transistors for DUTs and HTs in this work

Circuit	CMOS 64-bit adder	HT-1	HT-2	HT-3
Transistor Number	2560	8	28	100
Circuits	DCVSL 64-bit Adder	C432	C1908	C3540
Transistor Number	1644	2070	5516	9874
Circuits	S526	S832	S1196	S1488
Transistor Number	1682	1408	3056	2824

To observe the accumulated HT-induced effects through the system, we inserted the HTs payload on the inputs of DUTs deliberately to model the propagation of HT effect in a large-scale system. To compare the area and power consumption of DUT and HTs, we designed three HTs. HT-1 is OR3 trigger circuit with XOR2 payload. HT-2 is OR(XOR(AND(x,y),z),w) trigger circuit with XOR2 payload. HT-3 is AND4 plus modulo-8 counter trigger circuit with XOR2

payload. The complexity of the DUTs and HTs in this work is listed in Table 3.3. As can be seen, the HTs are significantly smaller than the target design.

3.3.2 Case Study on a 64-bit Full Adder

We implemented a 64-bit full adder using CMOS and DCVSL in Cadence Virtuoso. The layout area for these two adders is shown in Table 3.3. Because less PMOS transistors are needed in DCVSL, the area of DCVSL-based full adder is less than that of CMOS full adder when optimization is applied on both implementations. HTs are rarely triggered and the leakage power for HTs is a few orders of magnitude less than the adder switching power, as shown in Table 3.4.

Table 3.4 Power consumption for two 64-bit full adders and HT insertions

Unit under Test		Dynamic Power (mW)	Leakage Power (nW)
CMOS-based 64-bit full adder	Adder	24.6	65.78
	HT-1	0.444	0.419
	HT-2	0.942	1.243
	HT-3	1.028	2.583
DCVSL-based 64-bit full adder	Adder	8.002	47.50
	HT-1	0.566	0.328
	HT-2	0.892	0.993
	HT-3	1.544	2.465

All possible input patterns were applied to the 64-bit ripple carry adder. We placed a HT circuit to alter one complementary input pin in the adder. The power over time waveform is shown in Fig. 3.5. As can be seen in Fig. 3.5(a), when no HT is triggered, the switching power has instantaneous peaks whereas the leakage power remains flat (close to zero). Fig. 3.5(b) shows the power for the adder with one HT inserted at the 49^{th} 1-bit full adder. As can be seen, the power has an extra periodical increase, which is noticeably higher than the leakage power. This is the short-circuit power (discussed in Section 3.2) induced by the non-complementary inputs from HT insertion. We placed the HT payload circuit to the 2nd 1-bit full adder and observed different power behavior. As shown in Fig. 3.5(c), the increased short-circuit power appears in almost all input patterns. This is because the 2nd 1-bit full adder with non-complementary inputs yields non-complementary outputs, and those outputs are further propagated to other 1-bit full adders. Because of the propagation of HT effects, the power consumption is exceptionally higher than that in normal cases.

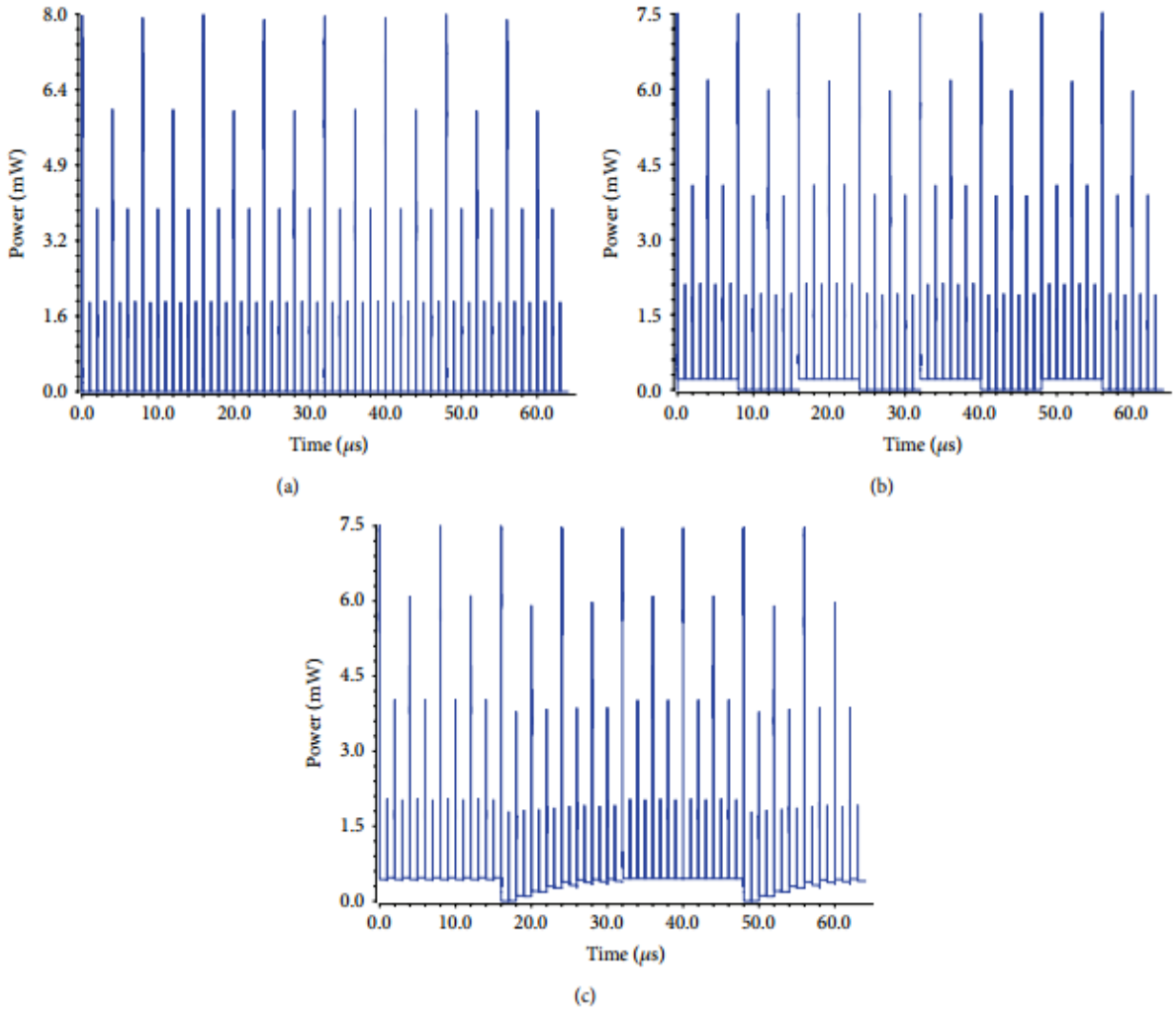


Figure 3.5 Power consumption for a 64-bit DCVSL full adder. (a) No HT, (b) HT on the 49th 1-bit full adder carry in port, and (c) HT on the 2nd 1-bit full adder carry in port.

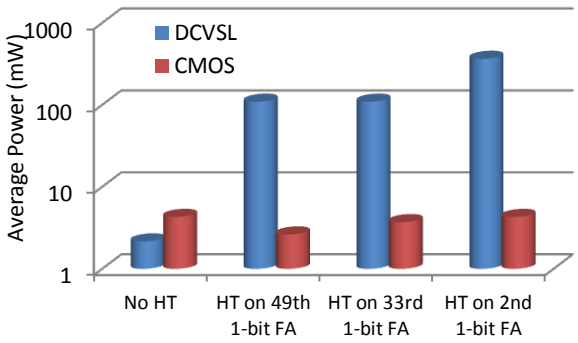


Figure 3.6. Impact of HT location on average power of 64-bit DCVSL adder.

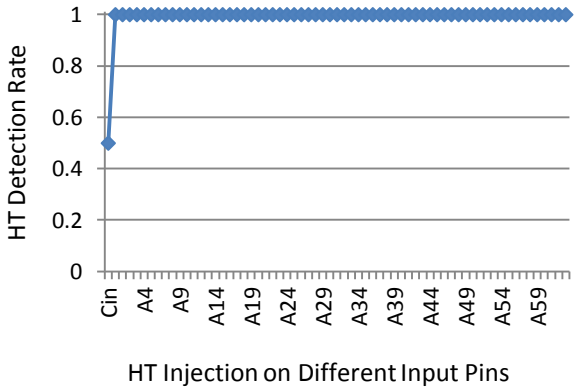


Figure 3.7 Impact of HT insertion locations on HT detection rate.

CMOS circuits have more PMOS transistors than the DCVSL version. Consequently, the dynamic power consumption of CMOS is higher than that of DCVSL. As shown in Fig.3.6, DCVSL has less average power consumption than CMOS. However, when the HT is triggered to change the non-complementary inputs for the DCVSL-based full adder, the increased short-circuit power results in a dramatic increase on the average power. Fig. 3.6 also show that the average power difference between original and HT affected version is over 50X. If the HT is inserted at the early stage in the functional block, the average power difference increases to over two orders of magnitude. This is favorable for power-based side-channel analysis HT detection

methods. To assess the HT detection rate, we assume that HTs are inserted to change the complementary inputs. As input vectors A and B for a 64-bit full adder are equivalent, we select 64-bit input A to receive the potential impact from HTs. Besides half of the inputs, A, the carry-in bit for the first 1-bit full adder is another potential location for HT insertion. As the proposed method is independent of the particular HT trigger circuit, we flipped one of the complementary inputs to model the effect of HT insertion. As shown in Fig. 3.7, for the HTs on A, the HT detection rate reaches to 1. Given a HT area over chip area ratio below 1%, the HT detection rate is higher than the one reported in [31]. Such high HT detection rate is mainly contributed by the non-complementary inputs, which lead to internal non-complementary outputs. Those outputs are further propagated to the remaining gates. Consequently, one HT injection possibly leads to more gate failures. Fig. 3.7 also shows that the HT inserted on the carry-in (Cin) input can be detected with a HT detection rate of 0.5, which can be compensated by comparing outputs. Our simulation results show, after the output comparison, the HT detection rate can be enhanced close to 1.

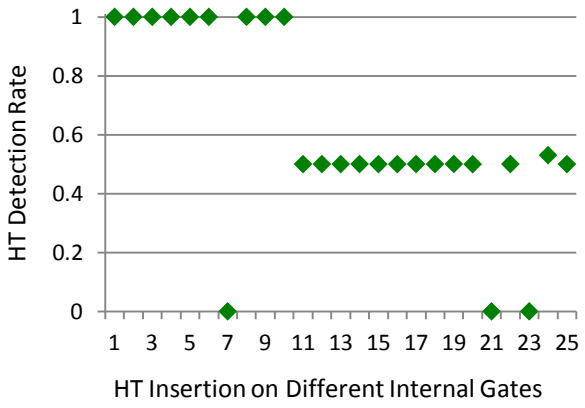
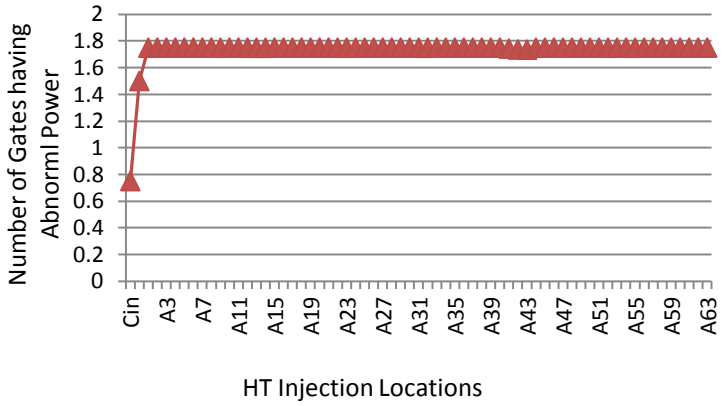
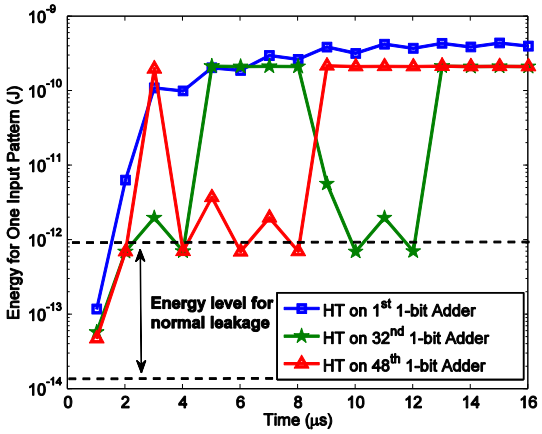


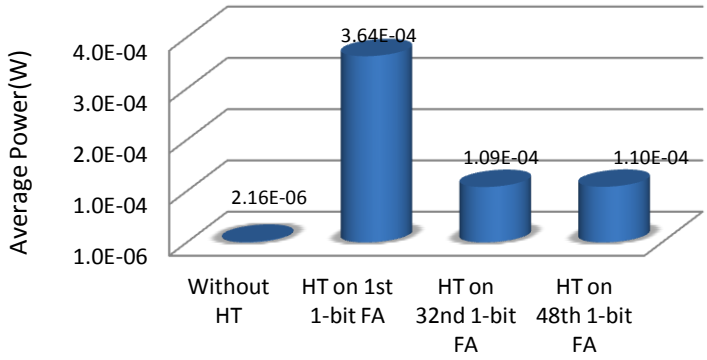
Figure 3.8 Impact of HT insertion locations on HT detection rate.



(a)



(b)



(c)

Figure 3.9 Results for HT-induced abnormal power assessment. (a) Average number of gates experiencing high short-circuit power per HT inserted case. (b) Abnormal energy caused by HT insertion over regular leakage energy. (c) Average power for three different HT injection locations. The simulated HT detection rate was obtained from 200,000 random input patterns.

HTs placed on input pins at earlier stages in the design have higher potential to be detected, because of the propagation of non-complementary outputs. We examine the impact of HT insertion locations on the HT detection rate. As shown in Fig. 3.8, as the HT insertion location shifts towards the final output, the HT detection rate decreases to around 0.5. The earlier the HT is inserted, the higher the probability of obtaining abnormal power behavior which can be used to determine the presence of HTs.

For HT injection on the very early inputs, each HT detected case will have about 1.7 gates experiencing high short-circuit power, as shown in Fig. 3.9(a). According to the Tables 3.1 and 3.4, the short-circuit power for one gate is one order of magnitude higher than the leakage power of a full adder. Therefore, the power difference is high enough for use in HT detection. As shown in Fig. 3.9(b), the HT inserted in the early 1-bit full adder stage yields an abnormal energy that is up to three orders of magnitude higher than normal leakage energy. HT insertion location approaching the final output yields less abnormal power, in terms of absolute energy value and the frequency of abnormal energy. As explained before, the latter HT injection location has a higher probability to demonstrate errors on the final outputs.

3.3.3 Evaluation on Benchmark Circuits

The proposed method is further evaluated with ISCAS benchmark circuits, which are composed with various logic gates listed in Table 3.1.. In the experiments below, we assume single HT is inserted in the benchmark circuit. More HT insertions in the target circuit lead to a higher HT detection rate, as more gates experience abnormal short-circuit power.The HT detection rate is defined as the number of cases experiencing abnormal short-circuit power over the total number of test cases. Three combinational benchmark circuits, c432, c1908 and c3540,

are used to assess the HT detection rate of our method. 500,000 random input patterns were applied to the evaluation of c432 and c1908 circuits. Because of larger scale, c3540 was evaluated with 1,000,000 random input patterns.

As shown in Fig. 3.10, our method achieves the HT detection rate up to 1 in the c432 circuit. The lowest HT detection rate is 0.7333. The majority logic gates in c432 are Inverter and AND2; thus the HT rates are centered around two particular regions, 1 and 0.73.

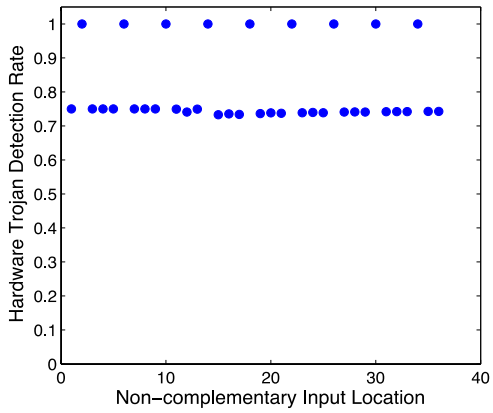


Figure 3.10 HT detection rate in c432.

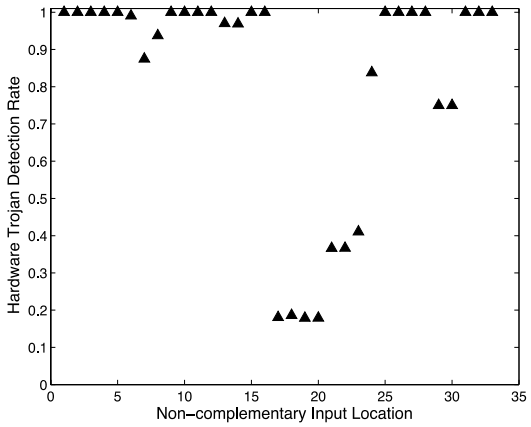


Figure 3.11 HT detection rate in c1908.

The scales of c1908 and c3540 are larger than c432; the kind of logic gates in c1908 and c3540 is more diverse than c432. These two factors affect the HT detection rate. Figures 3.11 and 3.12 show that the HT detection rate is distributed over the whole range, but the HT detection rate stay mostly above 0.7. We averaged the HT detection rate over all test cases in Fig. 3.13. As can be seen, our method achieves a HT detection rate over 0.8 in c432 and c1908. The HT detection rate for c3540 is slightly low; however, our HT detection rate is still significant, as our method is not limited by the size of HTs and can be used to detect extremely small HTs.

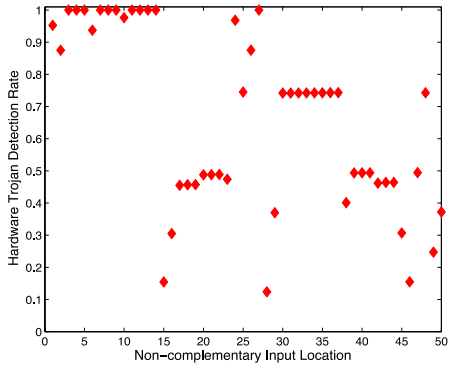


Figure 3.12 HT detection rate in c3540.

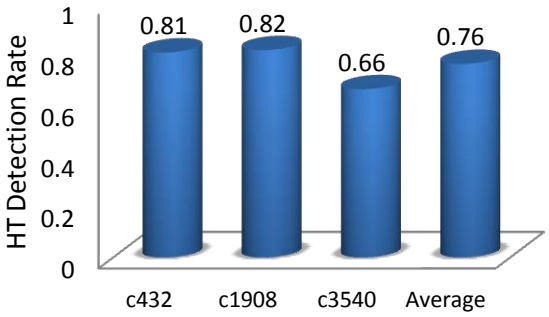


Figure 3.13 Average HT detection rate.

The average HT detection rate for the examined ISCAS'85 benchmark circuits is 0.76. To examine the amount of power increased by HT insertion, we further investigated the number of gates experiencing high short-circuit power per HT insertion. Fig. 3.15 shows the average number of gates that are affected by one HT insertion. As the abnormal short-circuit power also depends on input patterns of the target gate, the results reported in Fig. 3.15 is not always integer valued. As shown in Fig. 3.15, in most cases, each inserted HT does not only affect the gate directly controlled by the HT payload.

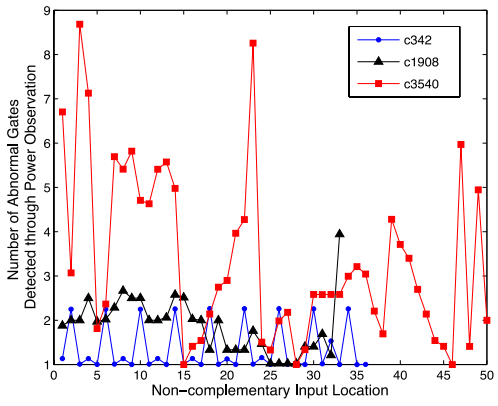


Figure 3.14 The number of gates experiencing abnormal power during each HT insertion.

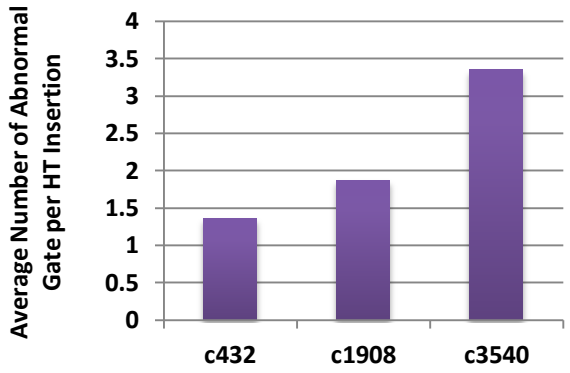


Figure 3.15 Average number of gates with abnormal power per each non-complementary input pair.

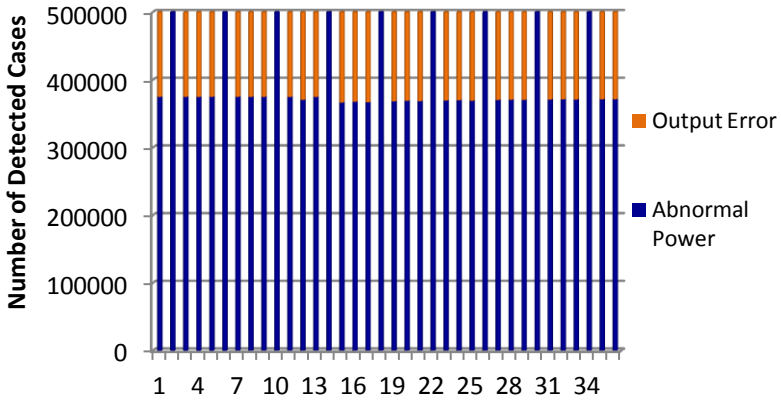


Figure 3.16 HT detection rate improvement by comparing complementary outputs in c432 circuit.

Instead, the impact of propagation of HT effect is more common in large-scale circuits (e.g. c1908 and c3540) than in small circuits. As shown in Fig. 3.15, the average number of gates affected by each HT insertion in c3540 reaches more than three. The higher number means more significant power increase for power-based HT detection. Detecting the non-complementary final output of DUT helps to improve the HT detection rate. As shown in Fig. 3.16, not all test cases have abnormal power behavior. We collected the number of cases that have non-

complementary outputs (i.e. output error) and observed that the cases for non-complementary DUT final output can achieve a HT detection rate of 1. This outstanding performance depends on circuit topology and the employed logic gates. Sometimes, the output error occurs at the same moment when abnormal short-circuit power is observed. Sequential circuits are more likely to be affected by HT effect propagation, as latches and flip-flops have a higher probability to remain high with short-circuit power than combinational logic gates.

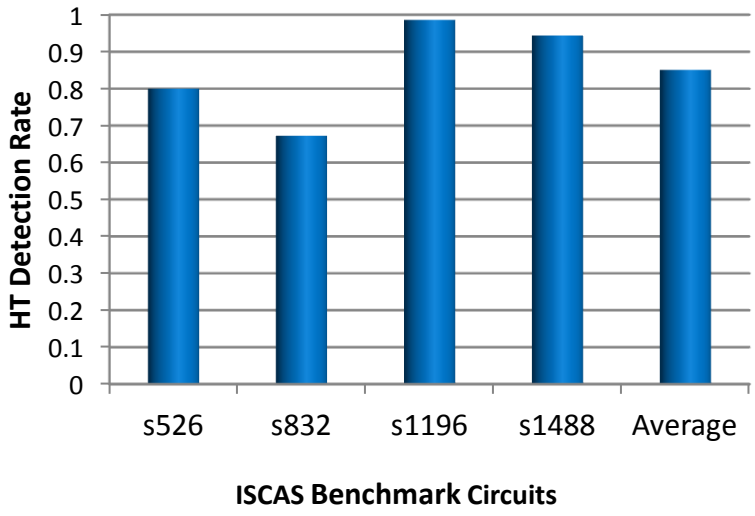


Figure 3.17 Average HT detection rate of different sequential benchmark circuits.

We injected single HT on the inputs of benchmark circuits, s526, s832, s1196 and s1488, to model the impact of HT on circuits. As shown in Fig. 3.17, on average, the HT detection rate on sequential circuit is higher than that in combinational circuits. The HT detection of s1488 and s1196 is close to 1. The average HT detection rate for the examined ISCAS'89 benchmark circuits is 0.85.

3.4 Conclusion

Hardware Trojans (HTs) challenge the chip security because of the increasing number of chips being fabricated, assembled, and packaged offshore. To enforce the confidence of chip security, efficient HT detection is imperative. HT detection can be performed during chip testing stage, although it requires large numbers of test vectors and long verification times. As argued by many researchers, testing approaches may not be practical to identify the rare events caused by HTs in a short period of time. Chip fingerprint is examined in IC authorization stages through side-channel analysis. Existing side-channel analysis approaches are challenged by process variation, lack of a perfect golden chip for comparison, and the presence of small-scale HTs. To address this need, the method proposed in this, use the inherent characteristic of DCVSL to detect HTs at runtime, without requiring a golden chip and a large number of test vectors. Proposed method is low-cost, convenient for user and complementary to existing power-based side channel analysis methods.

In this work, the DCVSL's complementary feature on both inputs and outputs to detect hardware Trojans at runtime, rather than offline is exploited. Non-complementary inputs in DCVSL-based systems lead to constant and abnormal short-circuit power peaks, which remain until the non-complementary inputs disappear. A case study on a 64-bit ripple carry adder shows that the proposed method achieves from 50X to two orders of magnitude higher average power difference than CMOS-based power analysis. Such high power difference between normal operation and HT triggered conditions is desirable for power-base side-channel analysis. Evaluation on a 64-bit adder shows that our method achieves a HT detection rate approaching 100%, if HTs are inserted to flip one of the adder inputs logic value. As HT payload circuits are

placed close to the final outputs, our abnormal power-based HT detection slightly loses its efficiency. The examination on the complementary characteristic of the outputs can improve the HT detection rate. Assessment on ISCAS'85 and ISCAS'95 benchmark circuits show that the HT detection rate is in the range of 66% to 98%. On average, our method can detect 76% and 85% HTs inserted in ISCAS'85 and ISCAS'89 benchmark circuits, respectively. By examining the complementary nature of the final output, we further improve the HT detection rate. Simulation on ISCAS'85 c432 circuit shows that the HT detection rate can be achieved to 100%.

Chapter 4: Fault-Tolerant Methods for A New Lightweight Cipher SIMON

In this chapter, first we investigated three low-cost fault-tolerant methods – reversed SIMON, double modular redundant (DMR) SIMON and even parity code (EPC) SIMON and then assessed their hardware cost and fault detection efficiency. The impact of fault-detection failure rate is studied further for three implemented methods.

4.1 Introduction

As the technology keeps shrinking and the supply voltage scales down, there is an increasing integration of more and more devices on a single die However, the reliability of the integrated circuit (IC) designs becomes severely challenged. The probability of occurrence of soft error [23] in circuits is rapidly increasing [8]. These soft errors are already making an impact in industry. According to Robert Baumann in an IEEE 2002 Reliability Physics Symposium tutorial, Sun Microsystems acknowledged in 2000 that cosmic ray strikes on unprotected cache memories had caused random crashes at major customer sites in its flagship enterprise server line, losing a major customer IBM [5]. In 1996, Eugene Normand reported numerous incidents of cosmic ray strikes after studying the error logs of several large computer systems [7]. The fear of cosmic ray strikes prompted Fujitsu to protect 80 percent of the 200,000 latches in its recent Sparc processor with some form of error detection [6]. If these soft errors are deliberately introduced by attackers on integrated circuits like cryptographic algorithm, statistical analysis can be performed on the correct and faulty outputs to retrieve the key, called as fault attack.

4.1.1 Need for Lightweight Block Cipher

AES [9] has been suggested for lightweight use, and given its stature; it should be used whenever appropriate. However, for the most constrained environments, AES is not the right choice: in hardware, for example, the emerging consensus in the academic literature is that area should not exceed 2000 gate equivalents [10], while the smallest available implementation of AES requires 2400 [11,16].

Among the block ciphers intended for use on constrained devices, some have been designed specifically to perform well on dedicated Application-Specific Integrated Circuits (ASICs), and thus can be realized by small circuits with minimal power requirements. Others are meant to perform well on low-cost microcontrollers with limited flash, SRAM, and/or power availability. Unfortunately, design choices meant to optimize performance on one platform often adversely affect performance on another [12]. Flexibility extends in another direction as well: since applications and devices vary, a variety of plaintext and key sizes is useful. For instance, block sizes of 64 and 128 bits are prevalent in the world of desktop computing, but atypical block sizes of 48 or 96 bits are optimal for some electronic product code applications. Key sizes, on the other hand, are related to the desired level of security, a very low-cost device may achieve adequate security using just 64 bits of key, while more sensitive applications (running on suitably higher-cost devices) may require as many as 256 bits of key.

4.1.2 Light-weight Block Cipher- SIMON

NSA published SIMON cipher in June 2013 [12], SIMON is a balanced Feistel cipher, which consumes 70% smaller area than the standardized low-cost AES alternative PRESENT

[11]. SIMON is composed of round and key schedule functions, and it can be implemented with shifting functions (SHIFT-L and SHIFT-R), ANDs and exclusively ORs which is shown in Fig. 4.1 and Fig. 4.2 respectively. Varieties of application use different block and key length, the algorithm which support different message and key parameters is useful. SIMON is available in different configuration to provide this flexibility. Table 4.1 lists the different block size, key size and no. of rounds sizes for SIMON.

Table 4.1SIMON parameters

Block Size (Bits)	Key Size (Bits)	No. of round
32	64	32
48	72	36
48	96	36
64	96	42
64	128	44
96	96	52
96	144	54
128	128	68
128	192	69
128	256	72

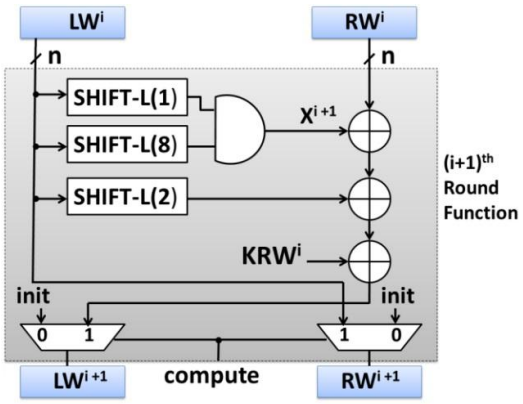


Figure 4.1 Round function in SIMON

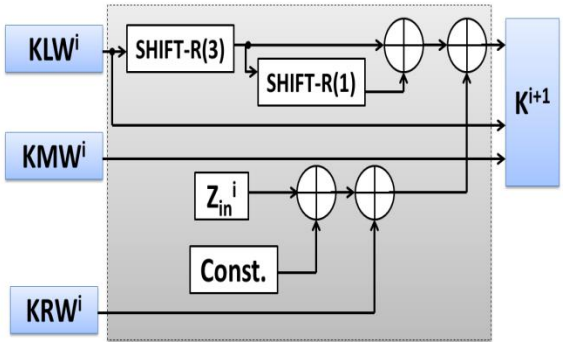


Figure 4.2 Key schedule function in SIMON

The round function for SIMON is described in equations (1)-(3). RW^i and LW^i stand for the right and left halves of the plaintext, respectively. The superscript i is the round number.

$$RW^i = LW^{i-1} \tag{1}$$

$$LW^i = RW^{i-1} \oplus F\left(LW^{i-1}\right) \oplus K^{i-1} \tag{2}$$

Where F(·) is defined in (3).

$$F(X) = \left((X \ll 1) \& (X \ll 8)\right) \oplus (X \ll 2) \tag{3}$$

To increase the difficulty for side-channel analysis attack, we assume that the initial key has three words. The algorithm for the key schedule function is expressed in (4)-(7). KRW^i, KMW^i, and KLW^i stand for the right, middle and left word of the key K^i, respectively.

$$KRW^i = KMW^{i-1} \tag{4}$$

$$KMW^i = KLW^{i-1} \tag{5}$$

$$KLW^i = G\left(KLW^{i-1}\right) \oplus \left(\sim KRW^{i-1}\right) \oplus Zin^i \oplus Const. \tag{6}$$

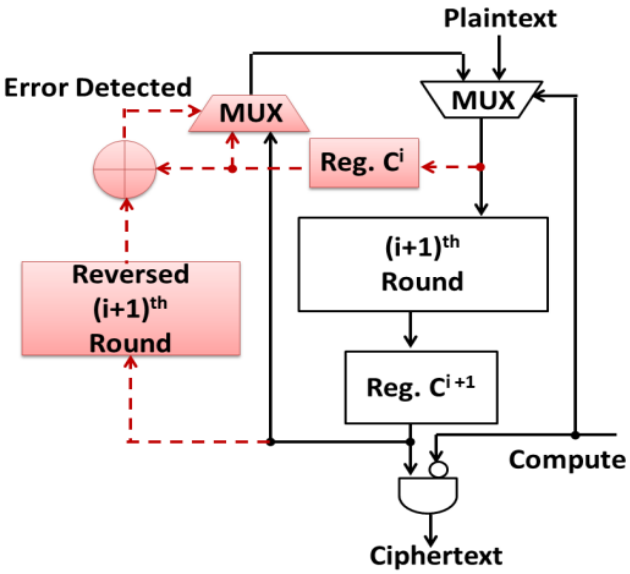
In which, G(·) is defined in (7), Zin^i a random number, and Const. is a constant value for the given SIMON.

$$G(X) = \left(X >> 4\right) \oplus \left(X >> 3\right) \tag{7}$$

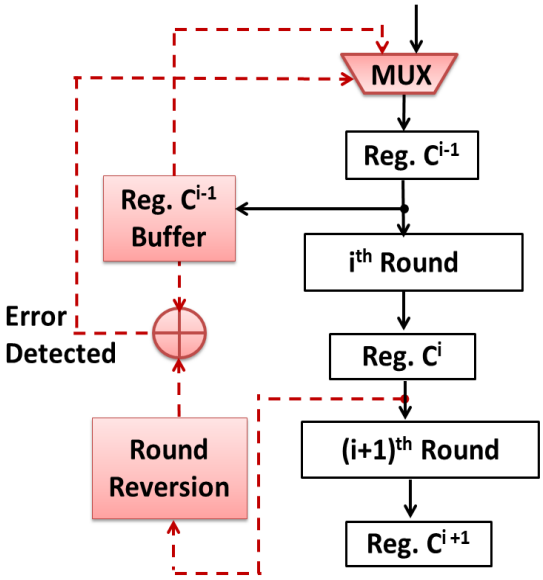
4.2 Proposed Fault Tolerance Methods

4.2.1 Round-Level Reversed-SIMON

Similar to the method proposed in [42], we develop a reverse function for SIMON to recover the original input and then compare with the saved input to detect errors. Figure 4.3 shows the proposed schematic for the SIMON equipped with a reverse function combined with a retry mechanism. The proposed reversed-SIMON can be implemented either in iterative or pipelined fashion. The reverse-function-based fault detection is performed on every round. If an error is detected by using reverse function, the warning signal activates the data path to reload the previous input saved in the register.



(a)



(b)

Figure 4.3 Schematics for proposed reversed-SIMON with (a) iterative, and (b) pipelined designs. Shadowed components and dashed lines for proposed additional functions over original design.

The reverse functions for round and key schedule functions are expressed in equations (8)-(9) and (10)-(12), respectively. We assume that the current round is *i*. The superscript *i-1* means the input from the previous round.

$$LW^{i-1} = RW^{i} \tag{8}$$

$$RW^{i-1} = LW^{i} \oplus F\left(RW^{i}\right) \oplus K^{i-1} \tag{9}$$

$$KLW^{i-1} = KMW^{i} \tag{10}$$

$$KMW^{i-1} = KRW^{i} \tag{11}$$

$$KRW^{i-1} = \sim \left(G\left(KMW^{i}\right) \oplus KLW^{i} \oplus Zin^{i} \oplus Const.\right) \tag{12}$$

4.2.2 Proposed EPC-SIMON

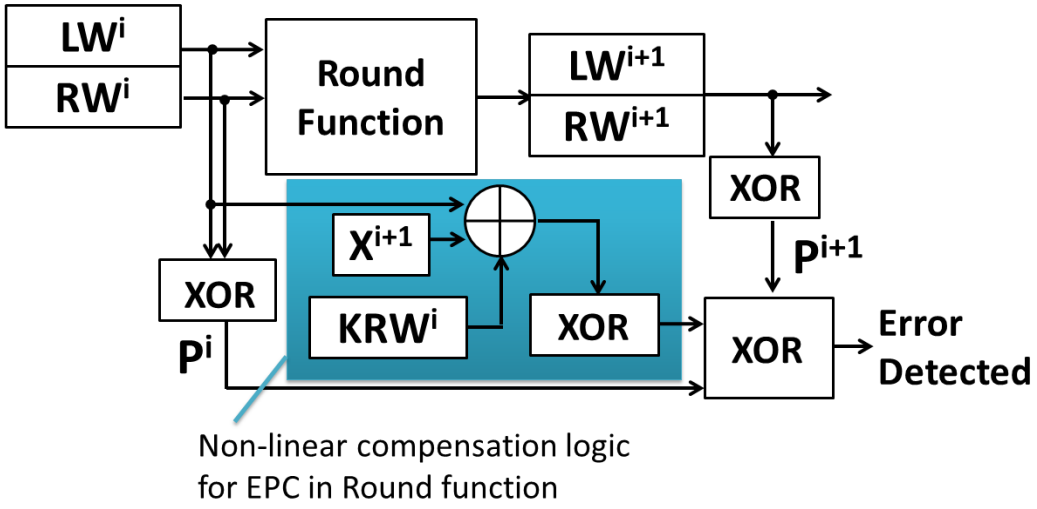
Either the round function or key schedule function in SIMON is not linear. It is not suitable to directly apply linear error detection codes to SIMON without special non-linear compensation function. For simplicity, we use an even-parity check code (EPC) as an example to explore the application of parity check codes to detect faults in SIMON. The parity check bit for the intermediate ciphertext of round *i* is expressed in equation (13).

$$P^{i} = \Theta\left(\left\{LW^{i}, RW^{i}\right\}\right) \tag{13}$$

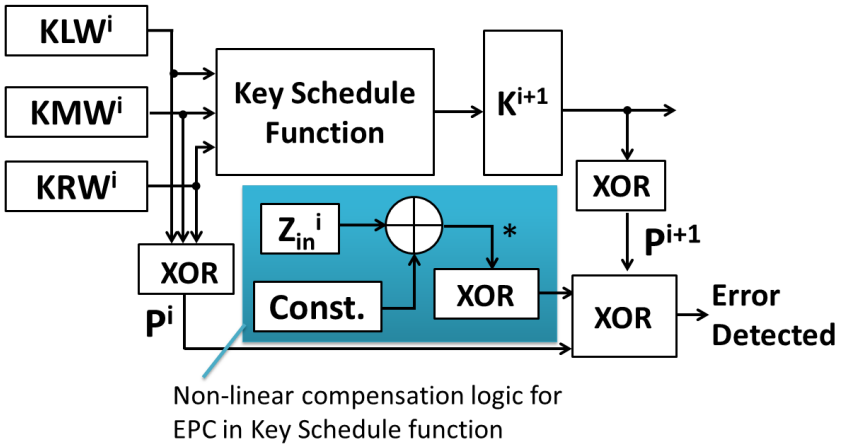
Where $\Theta(\cdot)$ is a reduction operator to exclusive-OR all input bits, and P^{i} is the check bit for the even-parity check code. The check bit for the next round is shown in (14).

$$\begin{aligned} P^{i+1} &= \Theta\left(\left\{X^{i+1} \oplus RW^{i} \oplus LW^{i} \oplus KRW^{i}\right\}, LW^{i}\right) \\ &= \Theta\left(X^{i+1} \oplus RW^{i} \oplus KRW^{i}\right) \end{aligned} \tag{14}$$

$$R.CompensationFunc^{i} = \Theta\left(X^{i+1^{i}} \oplus KRW^{i} \oplus LW^{i}\right) \tag{15}$$



(a)



(b)

Figure 4.4 Schematic for proposed EPC-SIMON (a) round function and (b) key schedule function. XOR is exclusive-OR all input bits. Symbol ⊕ is bit-wise exclusive-OR.

Because of the feature of bitwise exclusive-OR and reduction operator, we can derive the difference between P^i and P^{i+1} and obtain the non-linear compensation function for round function (i.e. *R.CompensationFunc* in (15)) required utilizing the linear parity check code for fault tolerance. Figure 4.4 (a) shows the schematic of the proposed EPC method for the fault

detection in SIMON round function.

We use the similar approach and obtain the non-linear compensation logic for the key schedule function. The corresponding schematic is in the Fig. 4.4(b). To save the hardware cost, one can obtain the intermediate output * from the original key schedule function. The limitation will be the fault in the calculation of the signal * cannot be detected.

4.3 Simulation Results

4.3.1 Experimental Setup

We used a SIMON with 64-bit plaintext and 96-bit key as an example to demonstrate the hardware cost and fault-tolerant capability of three proposed methods. Three fault-tolerant methods were applied to both iterative and pipelined SIMON architectures. With a TSMC 65nm CMOS technology, we synthesized the Verilog HDL codes for baseline SIMON (*baseline*), SIMON with double modular redundancy (*DMR-SIMON*) at each round and key schedule function, SIMON with even parity check code for error detection (*EPC-SIMON*), and SIMON with round-level reverse function for error detection (*Reversed-SIMON*). In the pipelined design, each pipeline stage contains one round and one key schedule function, which is as same as the design in iterative style. Therefore, we set same clock frequency for iterative and pipelined design to 500 MHz (as the worst-case delay is less than 2ns).

In the experiments for fault detection assessment, we adopt the method in [43] to inject faults at the input of one or multiple internal transformations of a round. Unlike [43], we also consider the fault injection in the key schedule function. We believe both round function and key schedule function are vulnerable to faults with an equal probability.

4.3.2 Area Comparison

We compared the area cost of eight different SIMON implementations in Table 4.2. In the iterative style, the area of Reversed-SIMON is close to that of DMR-SIMON. In contrast, in the pipeline style, the former one consumes less area than the latter one because of less registers. As the reversed round and key functions are used in the decryption process, one can manage to re-use those functions from decryption module to further reduce the cost for fault detection.

The area cost of EPC-SIMON is less than that of DMR-SIMON and Reversed-SIMON; however, the parity check code based approach may increase hardware cost as the number of parity check bits increases. We used the baseline as an example to break the total area of SIMON into three portions—round function, key schedule function, control logic and registers. As shown in Fig. 4.5, the SIMON area is dominant by the sum of round and key schedule functions. The control logic and registers consumes about 1/3 of the total area. We also examined the variation on the ratio of round and key schedule area over the total area among different SIMON implementations. As shown in Fig.4.6, the application of fault tolerance results in the decrease on the ratio of round and key schedule function area over the total SIMON area, no matter iterative or pipelined style. In EPC-SIMON, the ratio of round and key schedule drops to around 40%. If all faults are evenly distributed on the fault-tolerant SIMON, the increased area because of the control logic and registers for fault tolerance indicates that we need pay increasing attention to the faults in the fault-detection circuit itself. The results in Section 4.3.4 will show the consequence of faults in detection circuits.

Table 4.2 Area Cost Comparison.

Cost / Design	Area (um^2)			
	Iterative		Pipelined	
Baseline	2297.52	(100%)	74744.27	(100%)
DMR	4885.20	(213%)	234180.36	(313%)
EPC	3800.52	(165%)	175787.99	(235%)
Reversed	4824.72	(210%)	193973.75	(260%)

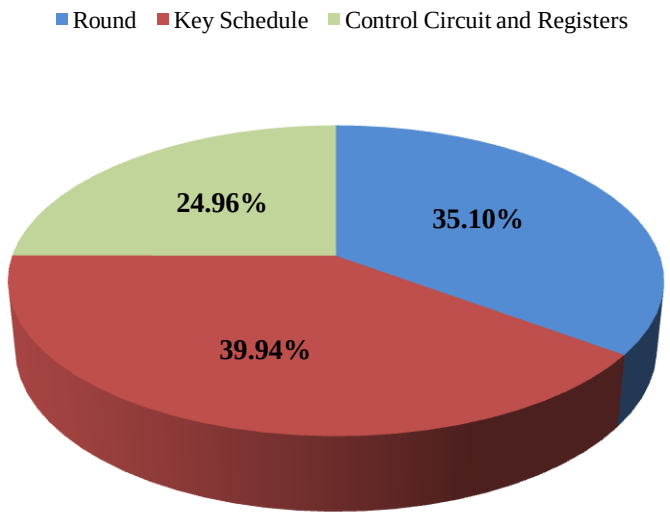


Figure 4.5 Area pie chart for baseline SIMON.

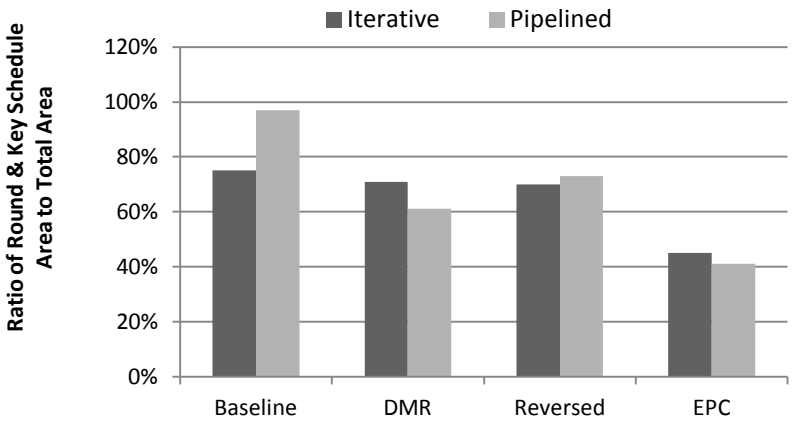


Figure 4.6 Dominant area portions of different fault-tolerant SIMONs.

4.3.3 Power Consumption

The power consumption for different fault-tolerant SIMONs is compared in Table 4.3. As shown, the power consumption of the reversed-SIMON is the highest one, compared with other methods. The three methods for fault detection and correction by retry result in the power increase by more than twice over the baseline design (except EPC).

Table 4.3 Power Consumption Comparison.

Cost / Design	Dynamic Power (mW)		Leakage Power (mW)	
	Iterative	Pipelined	Iterative	Pipelined
Baseline	0.9043	34.8190	0.0107	0.3424
DMR	1.8609	95.4959	0.0228	1.0919
EPC	0.9634	68.2444	0.1061	0.7894
Reversed	1.7650	96.4859	0.0225	0.8844

4.3.4 Fault-Detection Failure Rate

We examined the impact of increasing fault number on the fault-detection failure rate, which is defined as the ratio of the number of undetected faults over the total fault injection cases. We injected faults to the iterative SIMONs as an example. As shown in Fig.4.7, EPC-SIMON can detect all single-bit faults even if the fault is injected on the fault detection circuit; but, as the number of faults per test case increases, the fault-detection failure rate of EPC increases. Because we are using even-parity check code in EPC-SIMON, most of double faults cannot be detected. That is why the failure rate of EPC is bouncing between even and odd number of faults. DMR-SIMON achieves a lower fault-detection failure rate than Reversed-SIMON, as the number of faults injected increases. This is because the portion of control logic and registers in Reversed-SIMON is greater than that of DMR-SIMON. More faults in the control logic lead to more fault detection failures. Because of the small scale of iterative SIMON, we ran 10,000 random fault injection cases for each data point in Fig. 4.7.

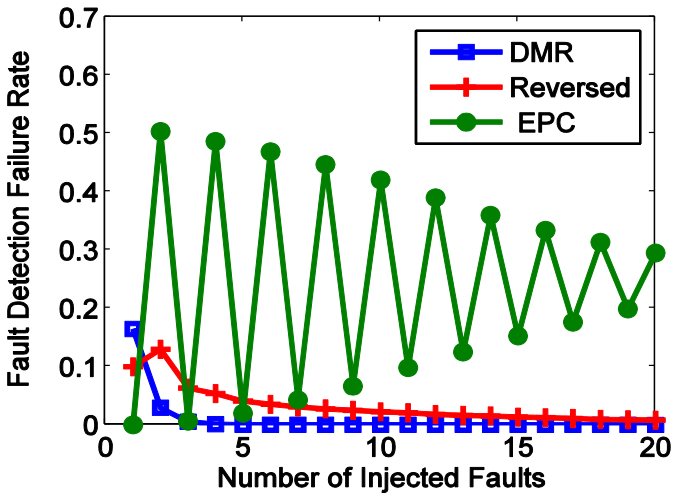


Figure 4.7 Impact of fault location in intermediate location on the fault detection capability of different approaches.

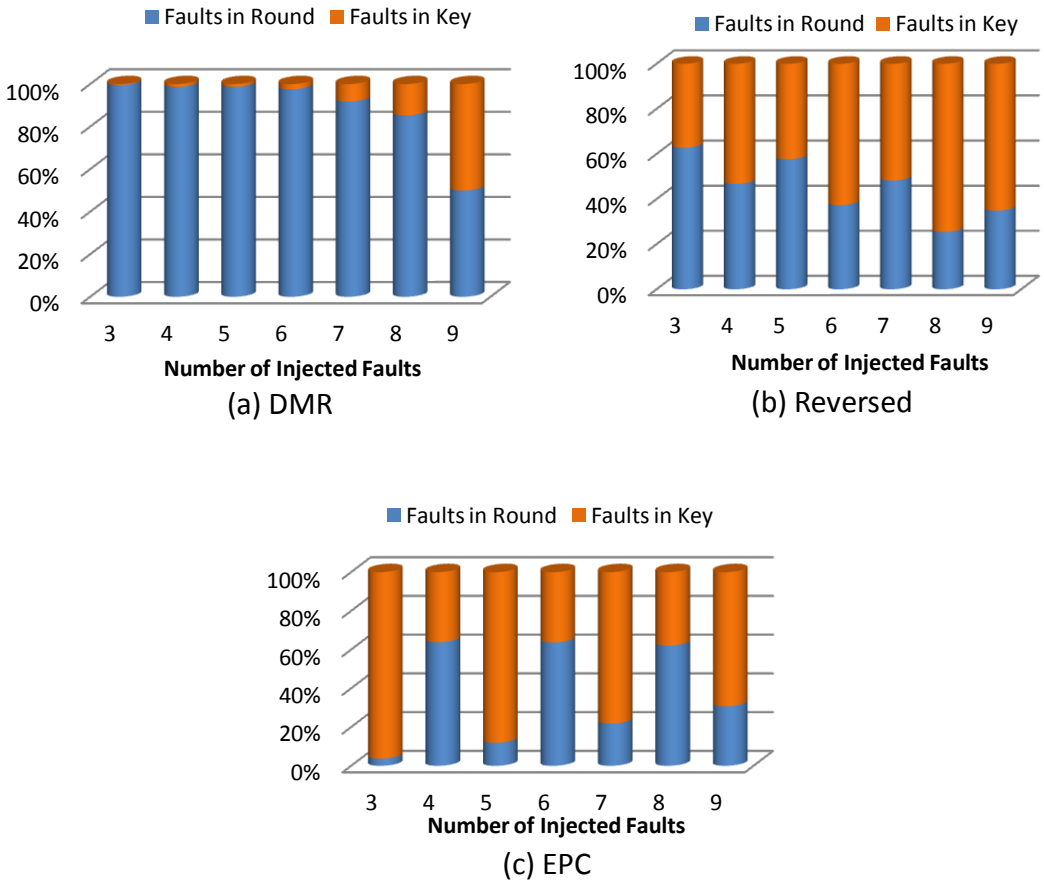


Figure 4.8: The contribution of faults in round and key schedule functions to the undetectable faults.

We further examined the distribution of undetected faults in different fault-tolerant SIMONs. In this set experiment, we only inject faults on the round and key schedule. functions. As shown in Fig.4.8(a), in DMR-SIMON, the undetected faults are mainly from the faults injected on the round function when the number of injected faults is less than nine. As shown in Fig. 4.8(b), the majority of undetected faults in the reversed-SIMON are from the key schedule function. This is because the reverse function uses key as an input to recover the previous plaintext; thus, the wrong key will be propagated to the reverse function and lead to more undetectable faults. For EPC-SIMON, the undetectable faults are mostly from key function in the

cases of odd-number faults; round function errors contribute more to the undetected faults for the cases of even-number faults.

4.4 Conclusion

The application of cipher is attractive to prevent secret leaking in embedded systems. As embedded systems typically have tight hardware-budget, we investigate the newly released lightweight cipher SIMON. Unlike regular integrated circuit, single fault injection in the implementation of cipher could lead to the ciphertext and decrypted plaintext completely wrong. Consequently, it is important to detect faults in cryptographic systems especially when the faults are induced intentionally for stealing the secret information.

Although the principle of our fault-tolerant methods is same to traditional methods, we derive the closed-form expressions of the reverse functions for reversed-SIMON, and propose the non-linear compensation functions for EPC-SIMON. Area cost and power consumption for eight different SIMON implementations are compared in this work. The parity check based fault detection method consumes slightly less area than DMR and reversed function based methods; the latter two double the area cost over the baseline design. By zooming in the area cost, we observed that the control logic and registers introduced by fault-tolerant methods tend to get close to the total area of the round and key schedule functions. This suggests us to pay a close attention to the faults injected to the fault detection circuit itself. Furthermore, our simulation results show that DMR-SIMON achieves better fault-detection failure rate than other methods. By comparing the undetectable faults in different fault-tolerant SIMONs, we found that faults in round function of DMR-SIMON leads to more undetected faults than faults in key function. However, this conclusion changes in the reversed-SIMON and EPC-SIMON

Chapter 5: Investigating Power Characteristics of Memristor-based Logic Gates and Their Applications in a Security Primitive SIMON

This chapter describes the memristor models used in this work and the preliminaries for memristor gate design. Further, we introduce the power characteristic for the memristor gates, and show the power dependent factors. The application of memristor for SIMON implementation and conclusion are discussed in subsequent section.

5.1 Memristor for security primitive

The emerging technologies are discussed in chapter2. There are various reasons why to choose memristor as emerging technology. Memristor is identified as the fourth fundamental circuit element, complement to resistor, capacitor and inductor. In 2008, Hewlett Packard (HP) Laboratories successfully fabricated memristors by using Pt (highly doped) and TiO2 (highly undoped) [54] .

The prevailing CMOS technology does not support security applications naturally. Therefore, developing CMOS based hardware security solutions, despite circuit optimizations and improved design techniques, still faces many challenges particularly in terms of circuit complexity, performance, and power consumption. Fortunately, the development of emerging technologies provides hardware security researchers opportunities to change the passive role that

CMOS technology plays in security application. The performance of current devices and machines is greatly hindered because of performance gap between CPU and memory. We need to find out opportunities, which dealt with speed and area. Memristor has special characteristics like non-volatility, non-linearity, low-power, and good scalability which make it great candidate for exploration. [33].Researchers have exploited the non-linear I-V characteristics of memristors to design the functional blocks for control systems [68], storage elements [52, 64, 67], analog circuits [64] and computing [55, 69], as well as, computational logic gates [53, 66]. Recently, the possibility of applying memristors to security primitive designs, such as physically unclonable function (PUF) blocks, has also been examined [56, 62]. In this work, we studied the feasibility to utilize memristors to a new security primitive, SIMON cipher, by exploiting the unique power characteristics of memristors and memristor-based logic gate.

5.2 Memristor Modeling

5.2.1 Verilog-A Modeling for Circuit Simulation

Memristor was first modeled with two adjustable resistors. A memristor is simply a variable resistor that changes its resistance based on how much current has flown through the device, and which direction of the device the current has flown. When current flows through the device through one terminal, the resistance approaches a resistance R_{on}, and the through the other terminal, a resistance R_{off}. When the device is turned off, it remembers the value of resistance it was prior to shutting off. We define the value of resistance between these two extremes, R_{on} and R_{off} as a function of the device width.

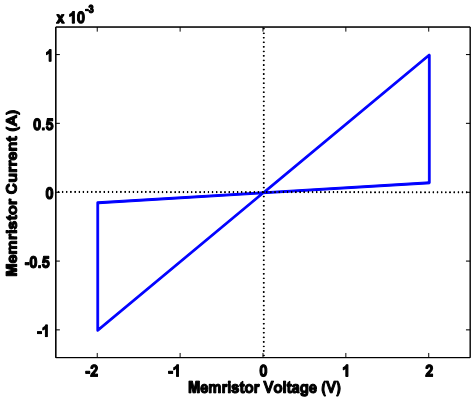


Figure 5.1. Memristor measured I-V curve for square-wave input voltages.

The HP Labs first modeled the unique current-voltage (I-V) characteristic of memristor with the Pickett model [58]. After that, Joglekar and Wolf [59], Biolek et. al [60], Prodomakis et al [61], and Corinto and Ascoli [65]. The ThrEshold Adaptive Memristor Model (TEAM) is the one that has been proved to be suitable for circuit simulations [57]. In this work, we use the TEAM model [57] to implement the memristor, which is further used for building memristor based basic logic gates. The TEAM model was chosen as it reflects Simmons' physical model and achieves a higher accuracy in terms of reproducing memristor dynamic and periodic behaviors than other models [32]. The I-V curve for the memristor modeled by the TEAM model (described by Eq.(1)) is shown in Fig. 5.1. The change in slope of the I-V characteristic demonstrates a switching between different resistance states; where the resistance is positive when the applied voltage increases and the negative when decreases. At a respective voltage, the curve shows the minimum and maximum current that can go through the memristor, which corresponds to R_{on} and R_{off}. The shape of the I-V curve is due to the resistance changing, which is controlled by the effective memristor width *x*.

$$v(t) = \left(R_{ON} + \frac{R_{OFF} - R_{ON}}{x_{off} - x_{on}} (x - x_{on}) \right) * i(t) \quad (1)$$

To facilitate the circuit design and obtain the convergence in Cadence-based simulations, Kvatinsky et al [8] proposed an exponential TEAM-based I-V relationship illustrated in the Eqs. (2)-(4) .

$$v(t) = R_{on} * e^{\frac{\lambda * (x - x_{on})}{(x_{off} - x_{on})}} * i(t) \quad (2)$$

where x_{on}, x_{off}, and λ are initially 0, 3*10^{-9} meter, and ln(R_{off}/R_{on}) respectively.

$$\frac{dx(t)}{dt} = \begin{cases} k_{off} \cdot \left(\frac{i(t)}{i_{off}} - 1 \right)^{\propto_{off}} \cdot f_{off}(x), 0 < i_{off} < i \\ k_{on} \cdot \left(\frac{i(t)}{i_{on}} - 1 \right)^{\propto_{on}} \cdot f_{on}(x), 0 < i_{on} < i \\ 0, \quad otherwise \end{cases} \quad (3)$$

where K_{on}=-8e-13, K_{off} =8e-13, α_{on}=3, α_{off} =3, i_{on} and i_{off} are on- and off-current, respectively, and

$$f_{on,off}(x) = \exp[-\exp(| x - x_{on,off} | / w_c)] \quad (4)$$

in which, w_C is the normalized length for Simmons tunnel barrier. In our model, we used

107*10^{-11} meter.

5.2.2 Our Device Symbol

We used Verilog-A language to model memristors and generated its symbol using Cadence Virtuoso. Our symbol is shown in Fig.5.2. Positive (*p*) and negative (*n*) represent two electrical nodes; a positive current enters the memristor through the *p* terminal and exits the memristor through the *n* terminal. In addition, *w_pos* corresponds to the relative width *x* in Eq. (1). *w_pos* is the measure of effective resistance which is function of R_{on} and R_{off}. As the

effective resistance approaches R_{on} or R_{off}, the device acts more and more like either a short or an open load.

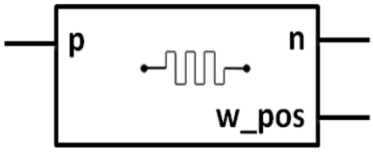


Figure 5.2. Our memristor symbol in Cadence Virtuoso.

5.2.3 Power Characteristic for Memristor

We measure the power consumption for memristor through the equation v(t)*i(t). Given an input period of 0.25ms, we obtain the power time diagram shown in Fig. 5.3. As can be seen, the power consumption changes over the time. This is because the memristor width is gradually controlled by the input voltage pulse. In the input duration time of 0.25 ms, the power decreases by 20%. If the input pulse remains a longer time, more power drops can be observed. This is different with the power consumption of a CMOS transistor.

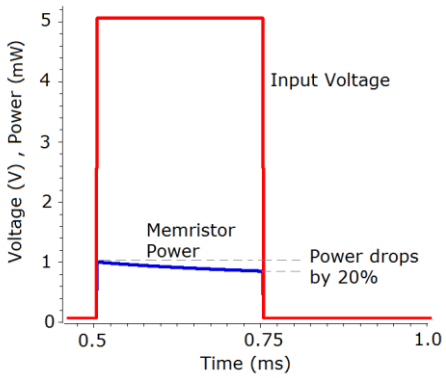


Figure 5.3. Power and input voltage versus time, highlight the power slowly change with input voltage.

5.3 Memristor-Based Gate Design

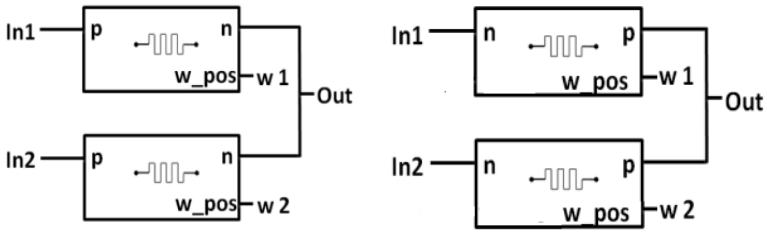
5.3.1 Memristor Logic

We constructed basic logic gates using a pair of memristors. Figures 5.4(a) and (b) show the functionality of a two-input memristor-based AND (*M-AND2*) gate and OR (*M-OR2*) gate, respectively. The difference between M-AND2 and M-OR2 is which terminal, n or p, the input voltage is connected. We use M-AND2 as an example to explain how this device works. When both input voltages are identical, there is no voltage across either memristor and the output voltage equals the input voltage. However, when one of the inputs is high and the other input is low, the current flows through the memristor with logic high as an input. This is because the high input increases the memristor's resistance. When the current flows from the p-node through the n-node, the current effectively pushes the memristor width towards R_{off}. That makes the memristor look more like an open circuit due to its high resistance. The other input of logic low is treated as ground, and the current continues to flow through the second memristor from the n-node through the p-node. This current lowers second memristor effective resistance and pushes the memristor width towards the lower resistance, Ron. This current effectively makes this memristor look more like a short, giving a direct path from ground to the output. The output is taken between the two memristors, which is a voltage division between the two memristors. The output voltage is close to 0 and it is defined in Eq. (5).

$$V_{out} = \frac{R_{on}}{R_{on}+R_{off}} V_{high} \cong 0 \quad (5)$$

Figure 5.5 shows the functionality of a two-input memristor-based AND (M-AND2). It can be seen that the gate follows the truth table for the AND function. In the third case in the

waveform, it takes more time to reach the output voltage value of logic low, as the memristor widths were further apart compared to the transition of the first and second cases.



(a) (b)

Figure 5.4. (a) M-AND2 circuit, (b) M-OR2 circuit.

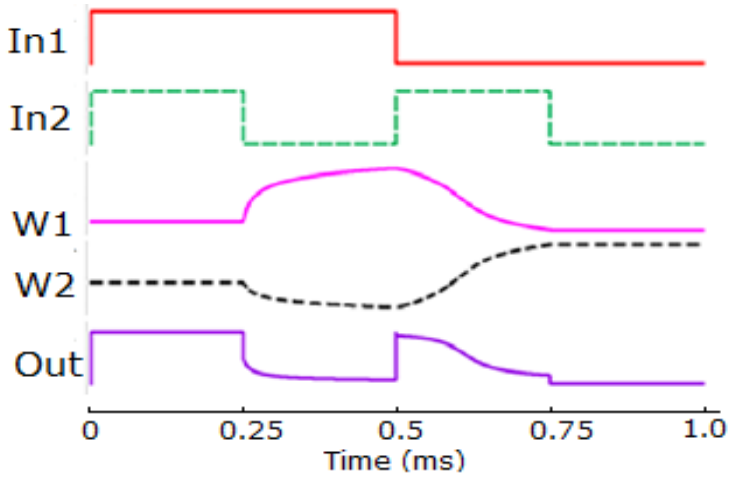


Figure 5.5. Simulated output waveform for M-AND2.

5.3.2 Hybrid Memristor-CMOS Logic Gate

Hybrid memristor-CMOS memory elements have been investigated in literatures [52]. In computational logic gate design, we also need hybrid design due to the resistive nature of the memristor—the inverter logic gate cannot be created using pure memristance. Consequently, we

used a CMOS-based NOT gate to implement a hybrid memristor-CMOS NAND gate (MC-NAND2) from M-AND2 shown in Fig. 5.6. The hybrid MC-NOR2 can be built with the same technique. As an example, we show the simulated output waveform for MC-NAN2 in Fig. 5.7. Note, the output waveform for the third input pattern, i.e. In1 is low and In2 is high, is not usual, as it takes time to change the memristor resistance to realize the correct logic function.

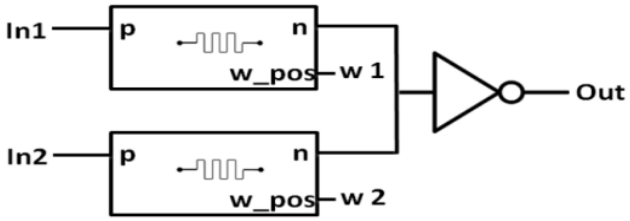


Figure 5.6. MC-NAND2 circuit.

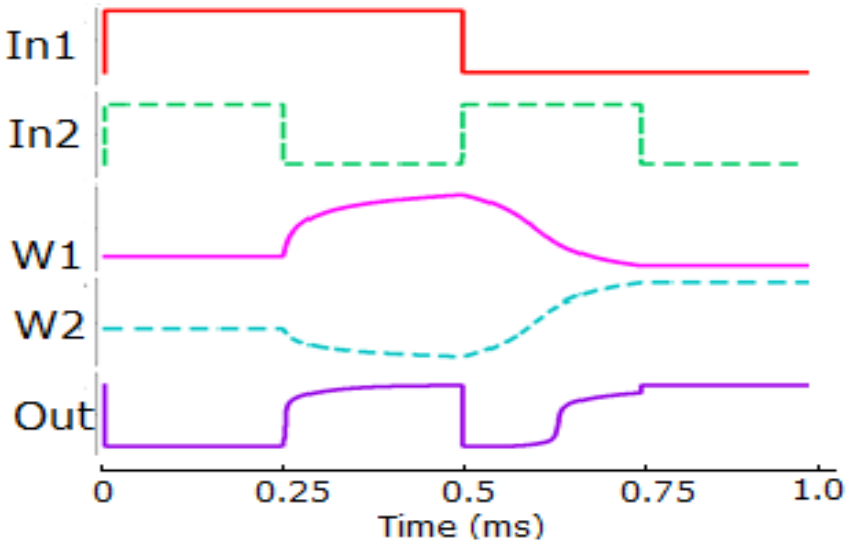


Figure 5.7. Simulated output waveform for MC-NAND2.

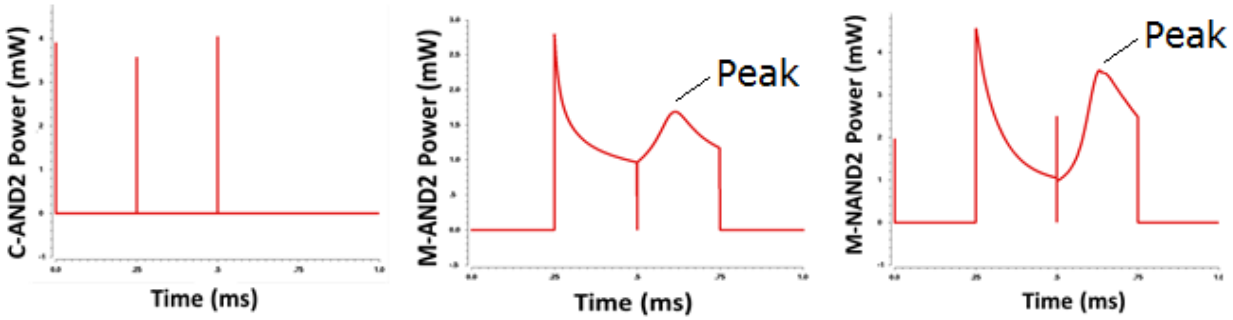
5.4 Power Characteristic for Memristor Logics

5.4.1 Experimental Setup

We used Verilog-A language to describe the TEAM model for memristor and constructed the logic gates in the same way introduced in Section 2.2. The hybrid logic gate is based on memristors and an 180nm IBM CMOS technology. All the simulations were performed in Cadence IC 6.1.5. The maximum voltage for inputs is 5 volt. The input switching period is 0.25ms.

5.4.2 Gradually Changed Power Characteristic

In Fig. 5.8, we compare the power consumptions of the purely memristive AND (M-AND2) and hybrid NAND (MC-NAND2) to that of the respective CMOS gates. The applied inputs for all the gates here are same. As shown in Fig. 5.8(a), the power for the CMOS-based AND2 has a sharp transition close to the edge of the input switching moment. This peak power is reached right after the input voltage switches. In contrast, the power for the memristor-based gate M-AND2 gradually changes with the input, as shown in Fig. 5.8(b). This slow-switching characteristic on power is due to the applied voltage gradually changing the memristor's resistance. The peak power is obtained at the moment when the two memristors in the M-AND2 achieve the same relative width (we define it as crossing point) and the most current is drawn on each memristor in the gate. This phenomenon is similar to the situation in CMOS—when both transistors in a CMOS gate are ON, that gate reaches its peak power consumption



(a) (b) (c)

Figure 5.8. Power comparison of (a) C-AND2, (b)M-AND2, and (c) MC-NAND2.

After the crossing point, the relative width of one memristor in the gate increases and that of another memristor decreases. As a result, the total power consumption for the gate decreases with time.

The power characteristic for the MC-NAND2 is shown in Fig. 5.8(c). As the hybrid NAND gate is the purely memristive AND followed with a CMOS NOT gate, the power for MC-NAND2 is the sum of the power for the M-AND2 and the CMOS NOT gate. Therefore, the power of memoristor-based logic gates gradually changes, rather than a sharp peak.

5.5 Dependent Factors for the Power Characteristic of Memristor-based Gates

5.5.1 Period Time of Logic Gate Input

We applied the input with period time varying from 50μs to 400μs on the M-AND2 and M-OR2 gates, and measured the peak power for each case. As shown in Fig. 5.9, the peak power for both gates increases with the input period time till the input period is beyond 150μs. After the input period exceeds 150μs, the memristor in the gate can reach the crossing point and thus the peak power approximately remains same. The increase on the power before the input period below 150μs is because the relative memristor width increases with the input period time.

Next, we examined the impact of input period on changing the relative width of memristors in the M-AND2 gate. As shown in Fig. 5.10, the relative widths for the two memristors in the M-AND2 gate (i.e. w1 and w2 shown in Fig. 5.4(a)) do not vary linearly with the input duration. When the input period time increases, w1 increases to 578mm at the input period length of 150µs and then remains approximately flat. On the other hand opposite behavior is observed for width w2. We re-examined this dependence in M-OR2 gate as well. As shown in Fig. 5.11, w1 decreases with the increasing input period time till the period exceeds 150µs. Although the w1 and w2 in the M-OR2 have opposite trends to those in the M-AND2, the dependence of input period on the relative width of memristor retains in same fashion.

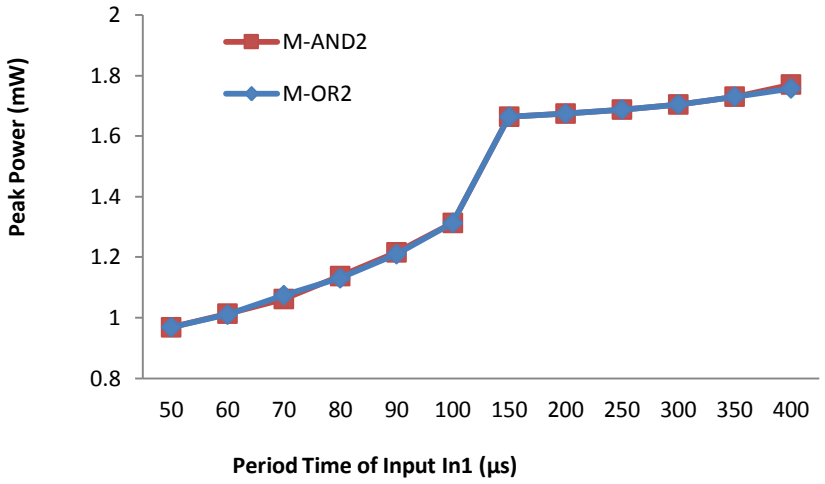


Figure 5.9. Impact of input pulse width on memristor peak power of M-AND2 and M-OR2.

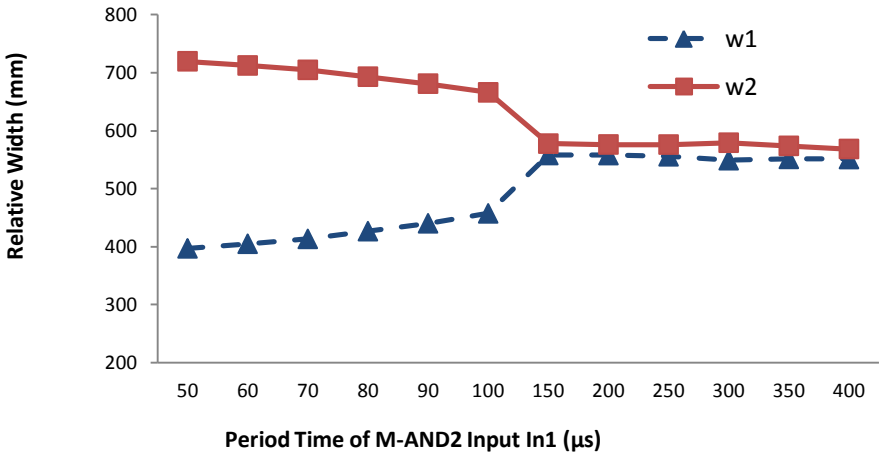


Figure 5.10 Impact of input preiod on memristor relative width in M-AND2.

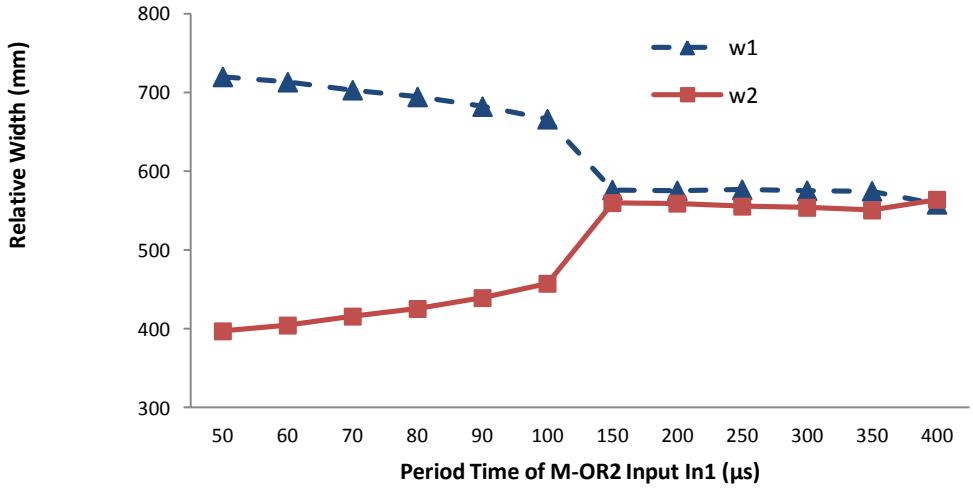


Figure 5.11. Impact of input period on memristor relative width in M-OR2.

5.5.2 Voltage Amplitude of Logic Gate Input

Another dependent factor for the peak power is the voltage amplitude of logic gate inputs. As shown in Fig. 5.12, the peak power does *not* monotonically increase with the input voltage. This is different with CMOS technology. Because the effective memristor width not

only depends on the applied voltage but also relies on how long the input voltage is provided, the maximum peak power is obtained as a combination of input period and input voltage amplitude. For instance, the maximum peak power for the input period of 100µs is achieved at 5V; in contrast, the M-AND2 with an input period of 150µs consumes the least peak power at the input voltage amplitude of 5V.

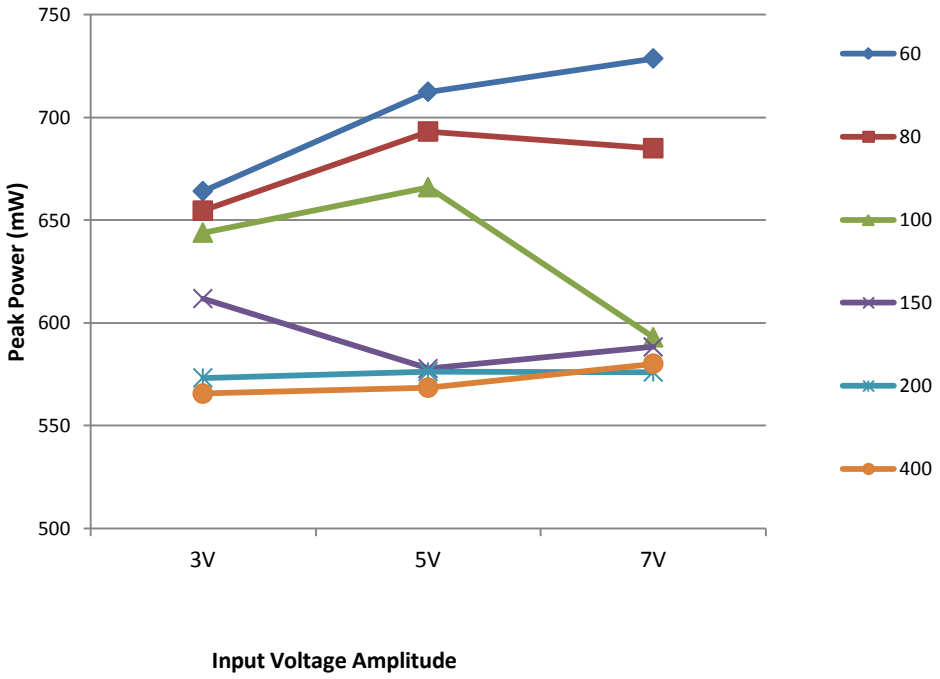


Figure 5.12. Impact of input voltage amplitude on the peak power of M-AND2 with different input periods.

5.6 Case Study of Memristor-Based Block Cipher *Simon*

5.6.1 Instantaneous Power

Using memristor-based logic gates (including hybrid ones) described in Section 5.2, we implemented a lightweight block cipher SIMON [12], as shown in Fig. 5.13. Due to the simplicity of hardware implementation, SIMON has potential to be used in hardware-restricted systems. SIMON consists of two functions: round function and key scheduling function. In this case study, we investigated the power of SIMON round function for 64-bit message and 96-bit key to examine the impact of special power characteristics on SIMON cipher. The logic gates M-AND2 and MC-XOR2 are the fundamental gates we used in SIMON implementation. The power waveforms for the CMOS and memristor-CMOS based SIMON round functions are shown in Figs. 5.14(a) and 5.14(b), respectively. We obtained the power characteristic for SIMON from Cadence Virtuoso. The input period was set to 5μs and the voltage amplitude was 5 volt. As can be seen, the CMOS-based SIMON (C-SIMON) has clear periodical peak power to indicate the power consumption for a given key.

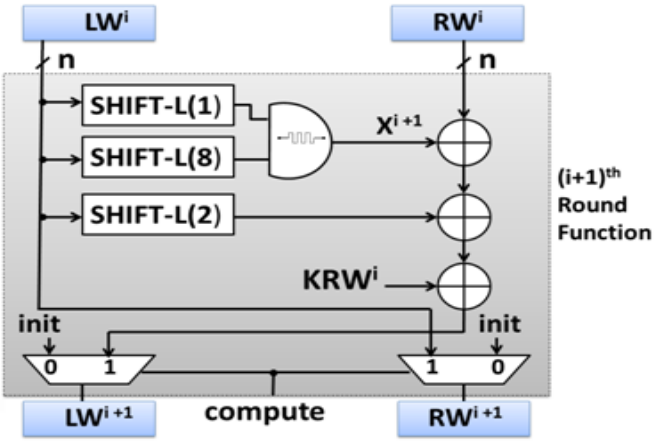
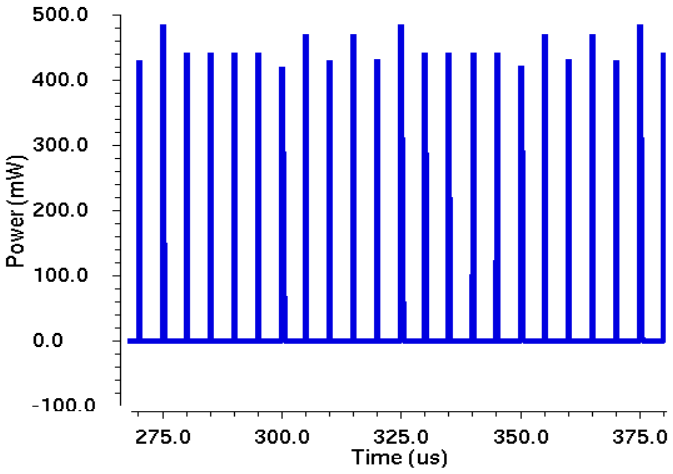
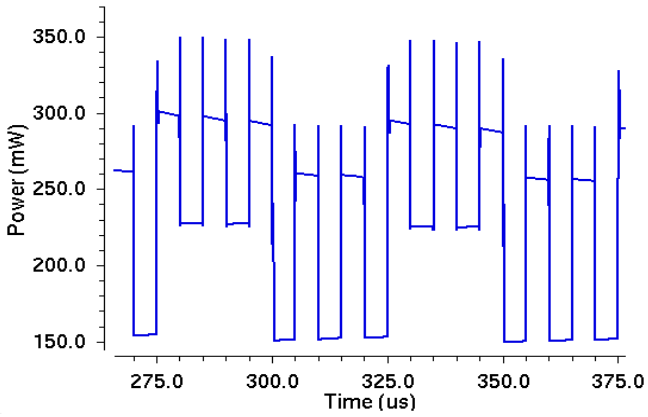


Figure 5.13. Memristor-based round function.



(a)



(b)

Figure 5.14. Power comparison between (a) CMOS and (b) hybrid memristor-CMOS round function in the SIMON block cipher.

In contrast, the power of memristor-CMOS-based SIMON (MC-SIMON) has different profile of peak power sequence. Moreover, after the peak power, the power for MC-SIMON gradually decreases until a new input voltage arrives. The CMOS inverters induce the sharp power peaks, and the ramp powers are caused by the nature of the memristor-based gates. We zoomed in the

power peaks of MC-SIMON in Fig. 5.15 and observed that the occurrences of power peaks are not predictable. The power variation near the edge of input switching moments is not regular.

5.6.2 Peak Power versus Sampled Power

In this section, we examine the effect of non-regular peak power. First, we investigate the power on a single gate, where we provide the same inputs to CMOS and memristor-CMOS gates. C-AND2 and C-NAND2 reach their peak power at the 0.5ms. In contrast, M-AND2 and M-NAND2 achieve their peak power at 0.25ms. The peak power comparison is shown in Fig. 5.16(a). If we use CMOS gates to determine when to sample peak power for power analysis, we obtain the sampled power shown in Fig. 5.16(b). As can be seen, the power sampled with the CMOS reference is only 32% (54%) of the real peak power of M-AND2 (MC-NAND2). If attacker attempts to use power as a side-channel signal to perform well known security attack—Differential Power Analysis attack] [70], the wrong sampling power could lead to wrong secret key for the cipher.

Now, we study the difference between peak power and sampled power on the SIMON round function. We define the power deviation as the ratio of the power difference between the peak power and stable power in the middle of input period over the peak power. We reported the power deviations for 64 cycles in Fig. 5.17. As shown, the power deviation varies from 6.43% to 56.94%. We examined the power deviation on the MC-SIMON with three different keys. As shown in Table 5.1, the average deviation is over 25% and the maximum deviation is up to 94.11%. This large deviation on the sampled power is significant enough to force attackers to collect extra power traces to guess the key bits, if attacker attempts to extract the secret key from a given SIMON implementation.

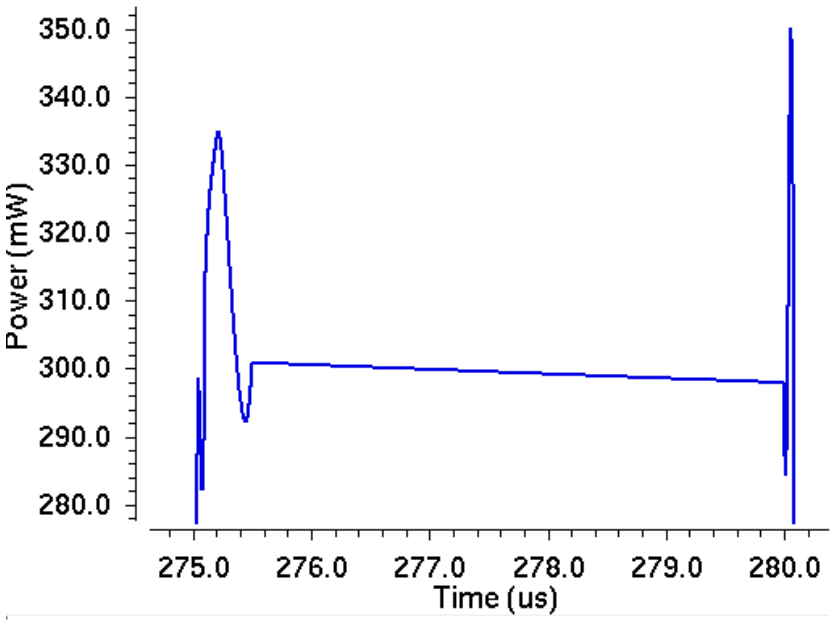
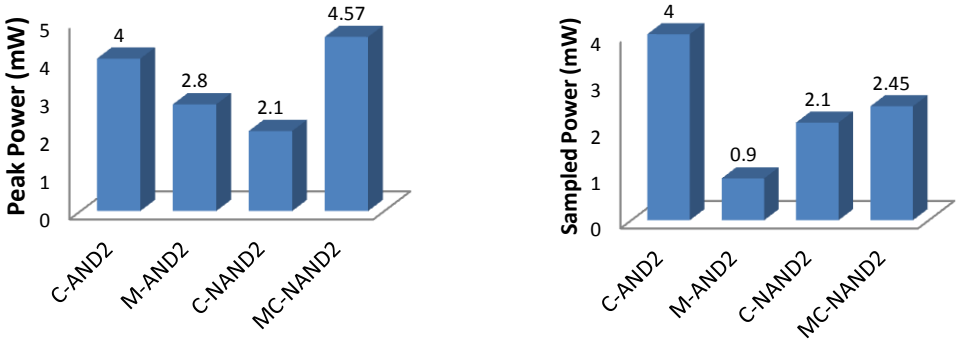


Figure 5.15. Power peaks of MC-SIMON at the edge of input switching moments.



(a) (b)

Figure 5.16. Power consumption comparison (a) peak power, and (b) sampled power.

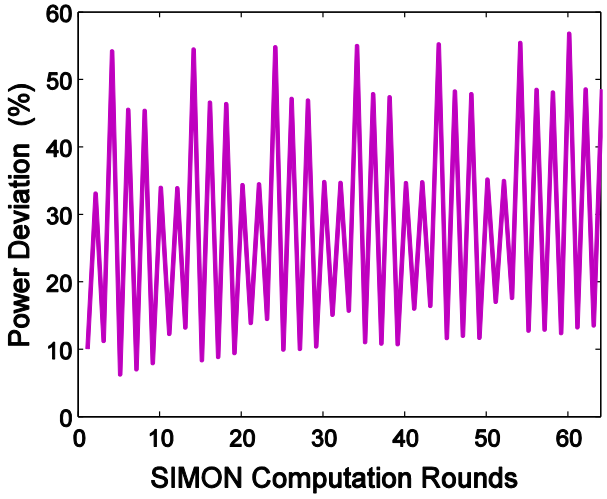


Figure 5.17 Power deviation due to sampling error in MC-SIMON.

Table 5.1. Average power deviation for three random keys applied to MC-SIMON

	Key 1	Key 2	Key 3
Average Deviation	28.46%	26.40%	25.39%
Maximum Deviation	56.94%	94.11%	49.99%

5.7 Conclusion

Memristor is identified as the fourth fundamental circuit element, complement to resistor, capacitor and inductor. The current-voltage relation of memristors has been extensively studied and modeled; however, the power characteristic of memristors has not been widely investigated. In this work, we use the threshold adaptive memristor model to implement memristor-based logic gates and study the power characteristic of memristors and memristor-based gates in a

circuit design environment. Our power analysis for different gates shows that the power characteristics for memristor-based logic gates gradually changes with the effective resistance formed in memristor, rather than a sharp power switching observed in CMOS logic gates. More interestingly, we observed that the occurrence of peak power depends on a combination of input period length and voltage amplitude.

As a case study, we examine the impact of memristor's unique power characteristic on the implementation of a block cipher SIMON. Because of the nature of memristor, it is difficult to design an NOT function with memristor. In our case study, we utilize hybrid memristor-CMOS logic gates to implement the SIMON (MC-SIMON). Compared with CMOS-based SIMON, MC-SIMON demonstrates a sharp power peak followed by a ramp power. This special power characteristic adds extra challenge on the power-based side-channel attacks on the cipher, as the memristor's power feature may introduce up to 94% power deviation in the process of power sampling. This may cost the attackers a longer time to extract secret key.

The limitation of the work is that the used TEAM model for memristor design works well at 5V.

Chapter 6: Discussion and Future work

6.1 Discussion

Due to escalating manufacturing costs, the advanced semiconductor technologies are often available at offshore foundries. Utilizing offshore manufacturing facilities significantly limits the trustworthiness of the chips that are fabricated and assembled by third-parties. The outsourcing business model leaves a hole for the adversary to embed unwanted or malicious hardware into the chips. As the chip scale continues to increase with new technologies, exhaustive testing and authentication processes are not affordable for commercial chip design companies. Hence, hardware security has emerged as a serious concern in chip designs and applications with increasing hardware attacks. These attacks attempt to either harm the security of chips or extract secret information for the original chip design. This thesis is devoted for developing detection and prevention techniques for passive and active hardware attacks. A passive attack attempts to listen or make use of information from the system but does not affect system resources. These attacks are difficult to detect because they do not involve any alteration of the data. Active attacks involve some modification in hardware or behavior of the system.

A new hardware disruptive threat has surfaced over the past few years, called as Hardware Trojan. As mentioned earlier in the thesis, Hardware Trojan is a backdoor that can be inserted in IC design cycle in any stage. A hardware Trojan may be able to defeat any security

mechanism to gain access to secret data like encryption key on the chips. Due to the complexity of today's integrated circuit design , detecting a small Hardware Trojan is extremely difficult. Due to technology scaling, detection by physical inspection and destructive reverse engineering is very challenging and costly. HTs are typically activated under very specific conditions, which makes them unlikely to be activated and detected using random stimuli. Lastly, due to process variation the impact of HT can be misleading. Considering these problem, the proposed HT detection method is feasible solution to detect small scale HTs.

In Chapter 3, we proposed HT detection method that utilizes the differential cascade voltage switch logic's (DCVSL) complementary feature on both inputs and outputs to detect small-scale HTs at runtime. This method is fundamentally different with existing offline approaches. The proposed method's area and test overhead are negligible compared to existing methods. The HT detection rate is in the range of 68% to 98%.

DCVSL logic family generates two logic output true and complementary. The complementary logic can be produced using less number of transistors than CMOS logic, indeed causing less hardware cost as discussed in chapter 3. Due to absence of pull up network different from CMOS, DCVSL generates faster response. The complementary feature of DCVSL logic makes it special for HT detection, as it may be difficult to mute complementary output simultaneously.

In particular, future investigations will extend by exploring the implementation of proposed method for large-scale circuits.

Various hardware attacks challenge the chip security and reliability. In the context of cipher hardware implementation, fault attack is another type of attack, aiming to extract secret

keys applied to the protected embedded systems. Fault attacks have been used to break robust cryptographic systems which are used for embedded system applications. Embedded system is becoming a main solution to most specific tasks because of its high stability, economic power consumption, portability, and usefulness. As a result, embedded systems typically have tight hardware-budget. Embedded systems are used in important application like navigation tools like global positioning system (GPS), intelligent cruise control applications in radar, and military equipment. Hence, the security of embedded system is crucial. Due to hardware constraints, lightweight security measures like lightweight ciphers are required. We explored the newly released lightweight cipher SIMON which can performs exceptionally well in hardware platforms.

Chapter 4 presents three different fault tolerance methods for block cipher SIMON to protect the SIMON cipher from the attack causing faults (intentional as well as natural). In this work, diverse faults are inserted in three implementations of SIMON i.e double modular redundancy (DMR), even parity check (EPC) and Reverse function. The results showed that DMR leads the smallest fault failure detection but cost more area. EPC implementation cost comparatively smaller area than other two implementation but lead to more fault failure detection rate. Compared with other works, we specifically studied the impact of faults on the key scheduling module, as key module is equally responsible for generating the faulty ciphertext.

The proposed methods cannot address the identical faults placed in symmetric module, in case of DMR based approaches. The further investigation can be carried out to extend this work to address the special cases of fault attack. We did not study the impact of fault propagation from key to round module, as key schedule output is used for round function calculations.

Last contribution of the thesis is application of emerging technology memristor for protection of hardware cryptosystem from side channel attack. Side channel attacks are based on side channel information leaked while performing the cryptographic operation. Attacker silently listens to the leakage and find out the secret key using differential power analysis. Side channel attacks are threats to chip's security. CMOS technology based hardware design shows sharp power peak when there is input transition, which make power analysis uncomplicated. Hence the technology which supports security by itself is worth exploring.

In chapter 5, we explored the feasibility of applying an emerging technology, memristor, for SIMON implementation. It introduced the power characteristic for the memristor gates, and showed the power dependent factors of memristor gates. Unfortunately power dependant factor for memristor have not been studied widely. We found that the special power characteristic of memristor can add extra challenge on the power-based side-channel attacks on the cipher, as the memristor's power feature may introduce up to 94% power deviation in the process of power sampling. This may cost the attackers longer time to extract secret key.

The limitation of work is, the memristor model used for experimentation requires 5V for efficient operation. The future work can be done to improve memristor model. Side channel analysis have not been performed to validate the proposed method.

Classical security has created elegant security primitives, unfortunately, these solutions are vulnerable to physical and side channel attacks. It has been revealed that nanoelectronic (memristors, graphene, plasmonics, and quantum dots) based security primitives are potentially more robust than conventional CMOS device-based security primitives.

6.2 Future Work

6.2.1 Investigation of Integrity Test for Possible Attack Detection in SoC

System on chip (SoC). As SoC consist of many cores and different components, one solution will not work for different attacks on different modules. We need to think about security assistance, which will assure the security of entire SoC.

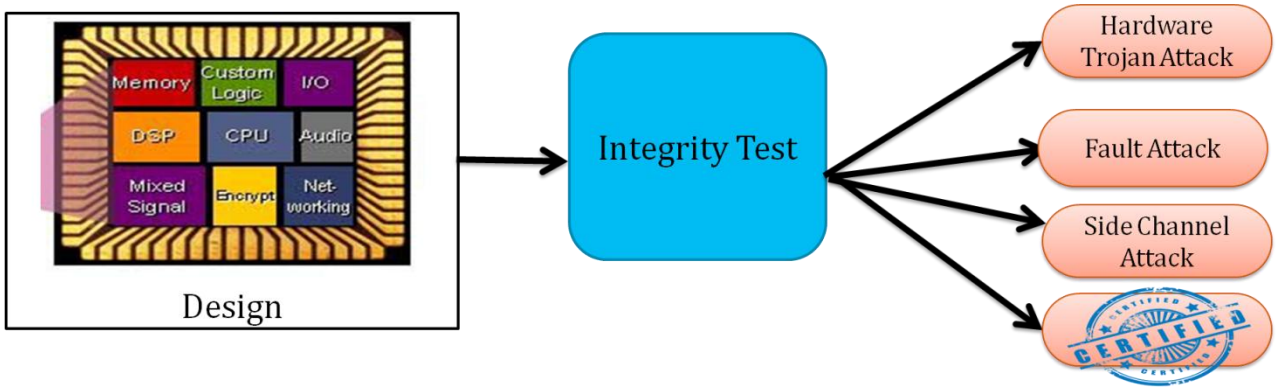


Figure 6.1 Uniform framework for detection of different attacks

The background knowledge can be used to build uniform framework to detect different types of attack in SoC.

6.2.2 Investigation of the Impact of Fault Tolerance Techniques on Cipher Power

In this thesis, we have examined the error detection rate of applying different fault tolerance techniques to SIMON. In addition to fault attacks, power-based side-channel attacks are another way to facilitate the extraction of secret keys used in the cipher. One interesting topic one can purse is to investigate the impact of fault tolerance techniques on the power characteristic of the hardware implementation of ciphers. The findings from this research will help future cipher implementation to achieve high reliability and security simultaneously.

6.2.3 Investigation of Methods to Address Multiple Hardware Attacks in One Framework

Most existing approaches for hardware attacks are customized for a single hardware attack. However, it will be more practical to develop chips that can tolerant multiple hardware attacks concurrently. Detection and prevention solutions for passive and active attacks are different. If one can develop a uniform framework to integrate solutions for different attacks in a hardware-efficient fashion, that solution will be more suitable for practical usages.

6.2.4 Investigation of Power Unifying Techniques to Thwart Side-Channel Attacks

The application of memristor to cipher implementation is one way to increase the difficulty of power-based side-channel attacks, as memristor's power characteristic is unique compared with traditional CMOS technologies. One interesting research topic along this line can be unifying the power characteristic over time, so that any changes made on the original chip design will be noticed easily without using a reference model.

REFERENCES

[1] Y. Alkabani and F. Koushanfar, "Active hardware metering for intellectual property protection and security," in Proc. 16th USENIX Security Symp., 2007, pp. 291–306.

[2] ADEE, S. The hunt for the kill switch. IEEE Spectrum 45, 34 –39, 5(may 2008).

[3] F. Koushanfar, I. Hong, and M. Potkonjak, "Behavioral synthesis techniques for intellectual property protection," ACM Trans. Design Autom. Electron. Syst., vol. 10, no. 3, pp. 523–545, 2005.

[4] SKOROBOGATOV, S., AND WOODS, C. Breakthrough silicon scanning discovers backdoor in military chip. Cryptographic Hardware and Embedded Systems–CHES (2012), 23–40.

[5] R. Baumann, "Soft Errors in Commercial Semiconductor Technology: Overview and Scaling Trends," IEEE 2002 Reliability Physics Symp. Tutorial Notes, Reliability Fundamentals, IEEE Press, 2002, pp. 121-01.1–121-01.14

[6] H. Ando et al., "A 1.3GHz Fifth Generation SPARC64 Microprocessor," Proc. IEEE Int'l Solid-State Circuits Conf. (ISSCC 03), IEEE Press, 2003, pp. 246-247.

[7] E. Normand, "Single-Event Upset at Ground Level," IEEE Trans. Nuclear Science, vol. 43, no. 6, Dec. 1996, pp. 2742-2750.

[8] P. Maistri and R. Leveugle, "Double-data-rate computation as a countermeasure against fault analysis," IEEE Transactions on Computers, vol. 57, no. 11, pp. 1528–1539, Sept. 2008.

[9] J. Daemen and V. Rijmen. The Design of Rijndael. Springer, Berlin, 2002.

[10] Juels and S. A. Weis. Authenticating Pervasive Devices with Human Protocols. In Advances in Cryptology—CRYPTO '05, Lecture Notes in Computer Science, No. 3126, pages 293–308. SpringerVerlag, 2005.

[11] Moradi, A. Poschmann, S. Ling, C. Paar, and H. Wang, "Pushing the Limits: A Very Compact and a Threshold Implementation of AES," In Advances in Cryptology—EUROCRYPT , Lecture Notes in Computer Science, No. 6632, pages 69–88. Springer-Verlag, 2011.

[12] R. Beaulieu, et al., "The SIMON and SPECK Families of Lightweight Block Ciphers," Cryptology ePrint Archive, Report 2013/404, 2013, http://eprint.iacr.org/2013/404.

[13] A. Kahng et al., ''Robust IP watermarking methodologies for physical design,'' in Proc. IEEE/ACM Design Autom. Conf., 1998, pp. 782–787.

[14] J. Lach, W. Mangione-Smith, and M. Potkonjak, ''FPGA fingerprinting techniques for protecting intellectual property,'' in Proc. IEEE Custom Integr. Circuits Conf., 1998, pp. 299–302.

[15] Y. Alkabani and F. Koushanfar, ''Active hardware metering for intellectual property protection and security,'' in Proc. 16th USENIX Security Symp., 2007, pp. 291–306.

[16] Pei Zhao, R.M. Feenstra, Gong Gu, and D. Jena, "Symfet: A proposed symmetric graphene tunneling field-effect transistor," Electron Devices, IEEE Transactions on, vol. 60, no. 3, pp. 951–957, March 2013.

[17] J. Markoff, "Old Trick Threatens the Newest Weapons," Oct. 26, 2009. URL:http://www.nytimes.com/2009/10/27/science/27trojan.html?pagewanted=all&_r=0

[18] John Ellis, "Trojan integrated circuits," URL: http://chipsecurity.org/2012/02/trojan-circuit

[19] S. Johnson, "Fake chips threaten military", San Jose Mercury News, Sept. 2010, http://www.mercurynews.com/breaking-news/ci_15990184.

[20] M. Banga and M. S. Hsiao, "Region Based Approach for the Identification of Hardware Trojans," in Proc. IEEE Intl. Workshop on Hardware-Oriented Security and Trust (HOST'08), pp. 40–47, 2008.

[21] D. Mukhopadhyay and R. S. Chakraborty, "Testability of Cryptographic Hardware and Detection of Hardware Trojans," in Proc. *ATS'11* , pp.517–524, Nov. 2011.

[22] Bhunia S., Hsiao M.S., Banga M., and Narasimhan S., "Hardware Trojan Attacks: Threat Analysis and Countermeasures," In Proc. of the IEEE, Vol.102, no. 8, July 2014.

[23] R. Baumann, "Radiation-induced soft errors in advanced semiconductor technologies," IEEE Transactions on Device and Materials Reliability, Vol. 5, no. 3, pp.305-316, Sept. 2005.

[24] Chua, L. O., "Memristor-the missing circuit element," IEEE Transactions on Circuit Theory, Vol.18, no. 5, pp.507-519, 1971.

[25] Strukov, D. B., Snider, G. S., Stewart, D. R., and Williams, R. S., "The missing memristor found," Nature, Vol.453, no.7191,pp. 80-83, 2008.

[26] F. Wolff, et al., "Towards Trojan-Free Trusted ICs: Problem Analysis and Detection Scheme," In Proc. Conf. Design, Automation, and Test in Europe, pp. 1362–1365, 2008.

[27] M. Banga and M. S. Hsiao, "Region Based Approach for the Identification of Hardware Trojans," In Proc. *HOST'08*, pp. 40–47, 2008.

[28] M. Tehranipoor and F. Koushanfar, "A Survey of Hardware Trojan Taxonomy and Detection," IEEE Design & Test of Computers, vol. 27, no. 1, pp.10–25, Jan.-Feb. 2010.

[29] Yu Bi, Gaillardon, P.-E. ; Hu, X.S., Niemier, M., Jiann-Shiun Yuan , Yier Jin , "Leveraging Emerging Technology for Hardware Security - Case Study on Silicon Nanowire FETs and Graphene SymFETs," , In proc. ATS, pp. 342-347, 2014.

[30] D. Rennels and H. Kim, "Concurrent Error Detection in Self-Timed VLSI," in Proc. 24th Intl. Symp. Fault-Tolerant Computing, pp.96–105, 1994.

[31] K. Hu, A. N. Nowroz, S. Reda and F. Koushanfar, "High-Sensitivity Hardware Trojan Detection Using Multimodal Characterization," in Proc. *DATE'13*, pp. 1271 – 1276, 2013.

[32] A. Ascoli, F. Corinto, V. Senger and R. Tetzlaff, “Memristor model comparison,” IEEE Circuits and Systems Magazine, vol. 13, no. 2, pp. 89-105, 2013.

[33] D. Niu, Y. Chen, C. Xu, and Y. Xie. Impact of process variations on emerging memristor. In Design Automation Conference (DAC), pages 877–882, 2010.

[34] J. Li and J. Lach, ‘‘At-Speed Delay Characterization for IC Authentication and Trojan Horse Detection,’’ in Proc. *HOST‘08*, IEEE CS Press, 2008, pp. 8-14

[35] National Inst. Of Standards and Technology, "Federal Information Processing Standard Publication 197, the Advanced Encryption Standard (AES)," Nov. 2001.

[36] X. Wang, M. Tehranipoor, and J. Plusquellic, “Detecting malicious inclusions in secure hardware: Challenges and solutions,” in Proc. IEEE Intl. Workshop on Hardware-Oriented Security and Trust (HOST' 08), pp. 15–19, 2008.

[37] Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, "Trojan detection using IC fingerprinting," in proc. IEEE Symposium on Security and Privacy, pp. 296-310, 2007.

[38] J. Rajendran et al. "Nanoelectronic Solutions for Hardware Security." In proc. IACR, 2012.

[39] Frontier Economics Ltd, London., “Estimating the global economic and social impacts of counterfeiting and piracy,” 2011.

[40] J. Valamehr “A 3-D split manufacturing approach to trustworthy system development” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 32, no. 4, pp. 611-615, Apr., 2013.

[41] Defense Advanced Research Projects Agency (DARPA), ‘‘Supply Chain Hardware Integrity for Electronics Defense (SHIELD),’’ Microsystems Technology Office/MTO Broad Agency Announcement, 2014

[42] R. Karri, W. Kaijie, P. Mishra, and K. Yongkook, “Fault-Based Side-Channel Cryptanalysis Tolerant Rijndael Symmetric Block Cipher Architecture,” in Proc. DFT’01, pp. 418–426, 2001.

[43] Guido Bertoni, et al., “Error Analysis and Detection Procedures for a Hardware Implementation of the Advanced Encryption Standard,” IEEE Trans. Computers, 52(4):492–505, 2003.

[44] Y. X. Su., J. Mathew, J. Singh, and D. K. Pradhan, “Pseudo parallel architecture for AES with error correction,” In Proc. IEEE Intl. SOC Conf., pp. 187–190, 2008.

[45] J. Mathew, et al., “On the design of different concurrent EDC schemes for s-box and gf(p),” In Proc. ISQED, pp. 211-218, 2010.

[46] L. Breveglieri, I. Koren, and P. Maistri, “Incorporating Error Detection and Online Reconfiguration into a Regular Architecture for the AES,” In Proc. DFT ’05, pp. 72–80, Oct. 2005.

[47] R. Karri and X. Guo “Invariance-based concurrent error detection for advanced encryption standard,” In Proc. DAC, pp. 573–578, 2012.

[48] Ting An, Lirida Alves de Barros Naviner and Philippe Matherat “Evaluation of Fault-tolerant Composite Field AES S-Boxes under Multiple Transient Faults,” In Proc. NEWCAS’13, pp 1–4, 2013.

[49] M. Mozaffari-Kermani and A. Reyhani-Masoleh, “Concurrent Structure-Independent Fault Detection Schemes for the Advanced Encryption Standard,” IEEE Trans. Computers, vol. 59, no. 5, pp. 608–622, May 2010.

[50] M. Mozaffari-Kermani and A. Reyhani-Masoleh, “ A Lightweight High-Performance Fault Detection Scheme for the Advanced Encryption Standard Using Composite Field,” IEEE Trans. on VLSI, vol. 19, no. 1, pp. 85–91, 2011.

[51] Barenghi, L. Breveglieri, I. Koren, and D. Naccache e, “Fault injection attacks on cryptographic devices: Theory, practice, and countermeasures,” In Proc. IEEE, vol. 100, no. 11, pp. 3056–3076, Nov. 2012.

[52] Mohammad, D. Homouz, O. AlRayahi, H. Elgabra, and A. AlHosani, “Hybrid Memristor-CMOS memory cell: Modeling and design,” in Proc. IEEE Int. Conf. Microelectronics, pp. 1–6, Dec. 2011.

[53] S. Kvatinsky, A.Kolodny, U.C. Weiser, and E.G. Friedman, “Memristor-based IMPLY logic design procedure,” in Computer Design (ICCD), 2011 IEEE 29th International Conference (pp. 142-147).

[54] Strukov D. B. , Snider G. S., Stewart D. R., and Williams, R. S. (2008), “The missing memristor found,” Nature, 453(7191), 80-83.

[55] J.Wang, Y. Tim, W. Wong, and H. Li, “A practical low-power memristor-based analog neural branch predictor,” in Proceedings of ISLPED 2013.

[56] Rajendran, J., Rose, G. S., Karri, R., & Potkonjak, M., “Nano-PPUF: A memristor-based security primitive,” IEEE Computer Society Annual Symposium on VLSI, (pp. 84-87) 2012.

[57] Kvatinsky, S., Friedman, E. G., Kolodny, A., & Weiser, U. C. “TEAM: threshold adaptive memristor model,” IEEE Transactions on Circuits and Systems I: Regular Papers, 60(1), 211-221, 2013.

[58] H. Abdalla and M. D. Pickett, “SPICE modeling of memristors,” in Proc. IEEE Int. Symp. Circuits and Systems, pp. 1832–1835. May 2011.

[59] Y. N. Joglekar and S. T. Wolf, “The elusive memristor: Properties of basic electrical circuits,” Eur. J. Phys., vol. 30, pp. 661–675, 2009.

[60] Z. Biolek, D. Biolek, and B. Biolkova, “Spice model of memristor with nonlinear dopant drift,” Radio Eng., vol. 18, no. 2, pp. 210–214, 2009.

[61] T. Prodromakis, B. P. Peh, C. Papavassiliou, and C. Toumazou, “A versatile memristor model with non-linear dopant kinetics,” In IEEE Trans. Electron Devices, vol. 58, no. 9. 2011.

[62] P. Koeberl, U. Kocabas, and A-R. Sadeghi, "Memristor PUFs: a new generation of memory-based physically unclonable functions," In Proc. Design, Automation and Test in Europe (DATE'13)., pp. 428-431, March 2013.

[63] Chakraborty R.S., and Bhunia S., "HARPOON: an obfuscation-based SoC design methodology for hardware protection,". IEEE Trans CAD Vol.28, no.10, pp.1493–1502, 2009.

[64] Niu, Y. Chen, and Y. Xie, "Low-power Dual-element Memristor Based Memory Design," In Proc. ISLPED'10, pp. 25-30, 2010.

[65] F. Corinto and A. Ascoli, "A boundary condition-based approach to the modeling of memristor nanostructures" In IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 59, no. 11, pp. 2713-2726, Nov. 2012.

[66] J. Rajendran, H. Manem, R. Karri, G. S. Rose, "An Energy-Efficient Memristive Threshold Logic Circuit", IEEE Transactions on Computers, vol. 61, no. 4, pp. 474-487, 2012.

[67] Niu, Y. Xiao, and Y. Xie, "Low Power Memristor-Based ReRAM Design with Error Correcting Code," In Proc. 17th Asia and South Pacific Design Automation Conference (ASP-DAC) ,, pp. 79-84, 2012.

[68] X. Wang, Y. Zhao, and Y. Liao, "Dynamic Performance Analysis of PID Controller with one Memristor," In Proc. Intl. Conference on Information Science and Technology, pp. 1234-1237, March 26-28, 2011.

[69] Jo, S. H., Chang, T., Ebong, I., Bhadviya, B. B., Mazumder, P., & Lu, W., "Nanoscale memristor device as synapse in neuromorphic systems," Nano letters, vol.10, no 4, 1297-1301 (2010).

[70] Kocher, P., et al., "Introduction to Differential Power and Related Attacks," Cryptography Research, Inc., [online]: http://www.cryptography.com/resources/whitepapers/DPA-technical.html, pp. 1-5 (1998).

[71] Kahng et al., ‘‘Watermarking techniques for intellectual property protection,’’ in Proc. IEEE/ACM Design Autom. Conf., 1998, pp. 776–781.

[72] G. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret keygeneration," Proc. DAC, pp. 9-14, 2007

[73] J. Guajardo , S. Kumar , G. Schrijen and P. Tuyls, "FPGA intrinsic PUFs and their use for IP protection," Proc. CHES, pp. 63-80, 2007

[74] R. S. Chakraborty and S. Bhunia. Hardware protection and authentication through netlist level obfuscation. In Proceedings of IEEE/ACM ICCAD, pages 674–677. 2008.

[75] J. Roy, F. Koushanfar, and I. Markov, ‘‘EPIC: Ending piracy of integrated circuits,’’ IEEE Computer, vol. 43, no. 10, pp. 30–38, Oct. 2010.

[76] J. Rajendran, Y. Pino, O. Sinanoglu, and R. Karri, ‘‘Security analysis of logic obfuscation,’’ in Proc. IEEE/ACM Design Autom. Conf., 2012, pp. 83–89.

[77] Baumgarten, A. Tyagi, and J. Zambreno, ‘‘Preventing IC piracy using reconfigurable logic barriers,’’ IEEE Design Test Comput.,vol. 27, no. 1, pp. 66–75, Jan./Feb. 2010.

[78] Intelligence Advanced Research Projects Activityy (IARPA), ‘‘Trusted integrated circuits program,’’2011.[Online].

Available:https://www.fbo.gov/utils/view?id=b8be3d2c5d5babbdffc6975c370247a6.

[79] R. Jarvis and M. G. McIntyre, ‘‘Split manufacturing method for advanced semiconductor circuits,’’ U.S. Patent 7 195 931, 2004.

[80] B. Hill, R. Karmazin, C. Otero, J. Tse, and R. Manohar, ‘‘A split-foundry asynchronous FPGA,’’ in Proc. IEEE Custom Integr. Circuits Conf., 2013, DOI: 10.1109/CICC.2013.6658536.

[81] J. Rajendran, O. Sinanoglu, and R. Karri, ‘‘Is split manufacturing secure?’’ in Proc. IEEE Design Autom. Test Eur. Conf. Exhibit., 2013, pp. 1259–1264.

[82] SypherMedia, ‘‘Syphermedia library circuit camouflage technology.’’ [Online]. Available: http://www.smi.tv/solutions.html

[83] J. P. Baukus, L. W. Chow, R. P. Cocchi, P. Ouyang, and B. J. Wang, ‘‘Building block for a secure CMOS logic cell library,’’ U.S. Patent 8 111 089, 2012.

[84] S. P. Skorobogatov. Semi-invasive attacks - a new approach to hardware security analysis. In Technical Report UCAM-CL-TR-630. University of Cambridge Computer Laboratory, April 2005.

[85] J. Blomer and J. P. Seifert, “Fault Based Cryptanalysis of the Advanced Encryption Standard (AES),” FC 2003,LNCS 2742,pp.162-181, 2003