

# CORRELATION BASED ELECTROMAGNETIC ANALYSIS ATTACK AND COUNTERMEASURE

by

Suyash Mohan Tamore

A thesis submitted to the faculty of  
The University of North Carolina at Charlotte  
in partial fulfillment of the requirements  
for the degree of Master of Science in  
Electrical Engineering

Charlotte

2019

Approved by:

---

Dr. Fareena Saqib

---

Dr. James M. Conrad

---

Dr. Ronald Sass



## ABSTRACT

SUYASH MOHAN TAMORE. Correlation Based Electromagnetic Analysis Attack and Countermeasure. (Under the direction of DR. FAREENA SAQIB)

Side channel analysis is a non-invasive cryptanalysis attack categorized as an implementation attack. Electronic device leak information during the data processing via covert side channels. Power consumption, electromagnetic (EM) radiations, and process timing information are various side channels that can be exploited to steal the secret information. An EM side channel is a non invasive side channel. With the help of statistical analysis methods, the EM emanation captured from the cryptographic device reveals the secret encryption key. Non invasive side channel attacks are difficult to detect and thus it is necessary to design cryptographic engine/ secure engine resilient against side channel attacks. This thesis demonstrates the EM side channel attack and discusses the state of the art countermeasures against the EM side channel. The attack is demonstrated on the 128-bit Advanced Encryption Standard (AES)[1] encryption that can reveal the secret key and explores the the countermeasure for making the crypto engine resistant against the side channel attack using the key update scheme. The other technique include hiding and masking countermeasures.

## DEDICATION

For all the students and professors who are working in security research area to  
make the technology secure.



## ACKNOWLEDGEMENTS

I am sincerely thankful to my professor Dr. Fareena Saqib who introduced me to the hardware security domain and guided me, supported me with patience. I thank my colleagues, Ali Shuja Siddiqui and Yutian Gui, who have answered so many of my questions and helped me with their words of encouragement. I am thankful to my committee members Dr. Ronald Sass and Dr. Jim Conrad for valuable feedback on my work and taking time out of their busy schedule to help me.

## TABLE OF CONTENTS

LIST OF FIGURES	ix
CHAPTER 1: Introduction	1
1.1. Problem Statement	4
1.2. Thesis Organization	4
CHAPTER 2: Theory	6
2.1. Field Programmable Gate Array	6
2.2. Cryptography	6
2.2.1. Basics of Cryptography	7
2.2.2. Advanced Encryption Standard (AES)	9
2.3. Cryptanalysis of Block Cipher	13
2.3.1. Differential and Linear Cryptanalysis	14
2.4. Side Channel Analysis	16
2.4.1. Power Consumption	17
2.4.2. Electromagnetic Emanation	19
2.5. Side Channel Attacks	21
2.5.1. Adversary Model	21
2.6. Types of Side Channel Analysis Techniques	24
2.6.1. Timing Side Channel Analysis	24
2.6.2. Electromagnetic and Power Analysis	25
2.6.3. Simple Side Channel Analysis	25
2.6.4. Differential Side Channel Analysis	27

2.6.5. Correlation Electromagnetic Analysis	28
CHAPTER 3: Proposed Framework	29
3.1. Basics of Correlation EM Analysis	30
3.2. Principal Component Analysis Based Preprocessing Technique	32
3.2.1. PCA Calculation Procedure	33
CHAPTER 4: Experimental Setup and Description	38
4.1. Experimental Setup	38
4.2. Software Tools and Setup	41
4.3. Hardware Tools and Setup	44
CHAPTER 5: Electromagnetic Side Channel and Countermeasures	46
5.1. Electromagnetic Side Channel Analysis History	46
5.2. Countermeasures Against EM Side Channel Analysis	59
5.2.1. Software Level Countermeasure	60
5.2.2. Hardware Level Countermeasure	64
5.2.3. Sense Amplifier Based Logic (SABL)	66
5.2.4. Wave Dynamic Differential Logic (WDDL)	69
5.3. Proposed Countermeasure	71
5.3.1. Key Update Scheme Countermeasure	72
5.3.2. Key Generation on TPM	74
CHAPTER 6: Experimental Results	76
6.1. Simple Electromagnetic Analytic (SEMA)	77
6.2. Correlation Electromagnetic Analysis (CEMA)	80
6.3. Principal Component Analysis Transformation	85

	viii
6.4. Proposed Key Update Countermeasure Scheme	89
CHAPTER 7: Conclusion and Future Research	92
7.1. Conclusion	92
7.2. Future Research	94
REFERENCES	95

## LIST OF FIGURES

FIGURE 2.1: Key Expansion of 128-bit AES key.	11
FIGURE 2.2: Operations of Round Functions in AES-128 Encryption.	12
FIGURE 2.3: Architecture of 128-bit AES Encryption Algorithm.	12
FIGURE 2.4: Classification of Implementation Attacks on Cryptography.	17
FIGURE 2.5: CMOS Inverter Circuit.	18
FIGURE 2.6: Types of Electromagnetic Coupling.	19
FIGURE 2.7: Hamming Weight Associated Power Consumption.	23
FIGURE 2.8: An Example of EM Emanation Measurement on FPGA.	26
FIGURE 2.9: SEMA Experiment on Arduino UNO microcontroller.	26
FIGURE 3.1: Scatter Plot of PCA Components.	33
FIGURE 3.2: Scree Test Plot.	37
FIGURE 4.1: Experimental Setup Block Diagram.	38
FIGURE 4.2: Experimental Setup in Lab Environment.	40
FIGURE 4.3: EM Probe Positioning.	40
FIGURE 4.4: CEMA Analysis Commands.	41
FIGURE 4.5: LabView Control Panel.	43
FIGURE 5.1: Dual Rail Logic Design.	64
FIGURE 5.2: Design Process Flow for DRP Logic Design.	66
FIGURE 5.3: SABL Gate Operation.	68
FIGURE 5.4: WDDL Basic Logic Gate.	69
FIGURE 5.5: Key Update Scheme Algorithm Flow.	73

FIGURE 5.6: TPM Integration on Sakura-X Board.	74
FIGURE 6.1: EM Emanation from FPGA Implemented AES Encryption.	77
FIGURE 6.2: Magnified Waveform of EM Emanation for AES Encryption.	78
FIGURE 6.3: EM Emanation of Four Different Input Plaintext.	79
FIGURE 6.4: Superimposed Signal of Four Different Input Plaintexts.	79
FIGURE 6.5: CEMA Process Block Diagram.	81
FIGURE 6.6: Correlation Values for Correct and Wrong Key Guess.	82
FIGURE 6.7: Correlation Coefficient Graph for all Encryption Key 1.	83
FIGURE 6.8: Graph for Correlation VS Total Number of EM Traces.	84
FIGURE 6.9: Correlation VS Number Of Traces for Wrong Keyguess.	85
FIGURE 6.10: Scree Plots: PC and Explained Variance.	86
FIGURE 6.10: Scree Plots: PC and Explained Variance.	87
FIGURE 6.11: Correlation Coefficient Graph for all Encryption Key 1.	89
FIGURE 6.12: Graph of Correlation vs Number of Traces for Dataset 4.	90
FIGURE 6.13: Graph of Correlation VS Number of Traces for Dataset 4.	91

## CHAPTER 1: Introduction

Securing sensitive and personal information and data using cryptographic techniques is done by mapping the plaintext or information to the ciphertext using the secret key that is known to the communicating parties. Security of the key is very important as a malicious entity can reverse engineer the ciphertext using mathematical attack models to recover the keys. These keys can be extracted by using various invasive or non-invasive attacks through physical interfaces such as GPIO ports, power/ground terminals, and control or data paths. Confidentiality, integrity, and authenticity are the three core pillars of the security. For every new invention of the cryptographic algorithm, various schemes have also been developed alongside to attack and break the cryptographic algorithm. In the past few decades, there have been vast developments and improvements in computational capabilities of the computer systems. Powerful computers used for implementing cryptographic security are also being used to attack and break the same cryptographic systems.

Various security attacks and breaches have been demonstrated on the software implementation as well as the algorithmic weakness of the cryptographic schemes, for example, Trojan virus, CryptoLocker, MyDoom. There are countermeasures developed against those attacks such as anti-virus software and firewalls. It is important to note that all the computer viruses were developed by humans rather than the computer system itself. In this modern era of broadband internet, wireless communication, and high-speed computer systems cyber-attacks have become a commonly known phenomenon. It has become a tool of cyber warfare between nations and common practice in political and corporate espionage. Malicious or harmful entities/attackers are coming up with novel and more sophisticated techniques than

before. Merely updating security software alone is no more an efficient or viable solution. In the past few decades, the world of cryptography has seen the emergence of attack schemes that target the hardware which is running cryptographic algorithms. This new technique of attack and its study is classified under Hardware Security.

Hardware security is the study and assessment of vulnerabilities present in the hardware device, security policies for cryptography and data protection. Secure hardware design and security policies are developed to provide access control, device authentication, secure key storage, and secure crypto processing environment at hardware level. Meltdown [2] and Spectre [3] are modern examples of attacks on computer processors and have impacted millions of devices that are using this processor chips over last two decades. Majority of this vulnerable chips are being used by governments, military, banking and finance and commercial sectors. The modern market of electronics is flooded with embedded system devices of various size, shape, and functionality. Internet of Things are the resource constraint, communicating embedded devices that are used for automation such as in smart homes or autonomous vehicles. These devices have limited security and are in physical proximity of an attacker who can attack the devices using invasive or non invasive techniques, such as side channel attack.

Hardware security scientist, Dr. Paul Kocher in 1999, [4] showed that mathematically secure cryptographic algorithms can leak information through side channels by demonstrating the attack on Data Encryption Standard (DES) algorithm using the power side channel. Attack successfully extracted the 56-bit DES encryption key. The side channel attack using power side channel came to be known as differential power side channel analysis. Side channel analysis is classified as semi invasive or non-invasive passive attack. Side channel analysis targets various physical aspect of the electronic hardware. The important and common physical aspects are power consumption, EM radiations, timing information, and sound vibrations. By exploiting



these side channel, an attacker can circumvent the mathematically hardened cryptographic algorithm to extract secret information, the cryptographic key. So far, almost all the hardware implementations of the cryptographic algorithm such as DES, AES, RSA, ECC are susceptible to side channel analysis attacks. To perform side channel analysis on power consumption or timing information of the cryptographic hardware device, having physical access of the device is necessary. However, side channel information leakage via EM radiations can be observed, captured, and analyzed at a variable distance from the device. Such non-invasive nature makes the EM side channel attack more sophisticated and threatening.

EM radiations are considered an undesirable noise from Electromagnetic Compatibility (EMC) aspects. Various research efforts are carried out to find a solution to suppress or eliminate the Electromagnetic Interference (EMI). Any electronic device embedded with active or passive electronic components or semiconductor-based microchips radiates EM signals as governed by the laws of physics. Such low levels of EM radiations might protect the device from the dangers of malfunctioning from the EMI. However, experiments have proved that such radiations, if captured and properly studied, can give away or leak information being processed by the device. Gandolfi K., Mourtel C., Olivier F in 2001 and Agarwal, Dakshi et al. in 2002 presented their experimental findings on EM side channel and performed differential EM analysis on DES algorithm implemented in a smartcard chip. This phenomenon is compromising the device's security and ability to protect the data. In technical term, such EM radiations are referred to as compromising EM emanations. Compromising emanations are not limited to EM radiations as it can be a manifestation of EM radiation through different medium such as sound, electrical (voltage or current), mechanical or optical, etc. European products are standardized with EMC Regulation 2004/108/EC [5] and require labeling of the CE mark on all electronics. All the electronic device radiate inherent EM emanations.

## 1.1 Problem Statement

All electronic device inherently radiate the EM emanations and these emanations have correlation with the internal data processing of the computational engine. These correlations can be used by adversarial model to reverse engineer the internal states of the computer engine which can be used to extract the secret key. Cryptographic engine radiate EM emanation which are correlated with the signal transitions and bit flips on control path and data path resulting from cryptographic operations. Evaluation of the cryptographic hardware resilience against side channel attack is important to create necessary countermeasures.

This thesis describes the steps of correlating the EM emanations from cryptographic hardware with the corresponding data processed by the hardware at the same time instant. Use of the preprocessing technique explores the possibility of improving the side channel attack efficiency with practical experiment and results. A proposed countermeasure scheme is tested as a mitigation technique for side channel threat against the AES cipher.

## 1.2 Thesis Organization

The thesis is organized as follows: Chapter 2 covers the introduction of cryptography and background study specifically on the AES engine. Further sections provide an overview of the AES cryptographic algorithm, various cryptanalysis techniques and implementation attacks on cryptographic systems. Side channel attacks and their classification is introduced along with details of different side channel analysis techniques, one of which is used in the experiment. Chapter 3 describes the proposed framework, correlation analysis technique, and preprocessing methods using principal component analysis. Chapter 4 describes the overall experimental setup and various tools and devices used for the experiment. Chapter 5 is an overview of the state of art EM side channel attack and countermeasures along with the proposed key update

countermeasure technique. Chapter 6 covers the graphical results of the experiment and a discussion on the experimental results. The conclusion and future work are provided in chapter 7.

## CHAPTER 2: Theory

### 2.1 Field Programmable Gate Array

A Field Programmable Gate Array (FPGA) is a reconfigurable integrated circuit. It is constructed with various programmable logic cells, Look Up Table (LUT), and block RAMs. FPGAs provide the flexibility of implementing different hardware logic and in field reconfigurability. ASICs are one time programmable in contrast to FPGAs; which can be reprogrammed several times to suit the operational needs based on changing properties of hardware logic. FPGAs are a great prototyping tool as it allows reconfiguration and implementation of logic as small as a single logic gate (AND, XOR, NAND gate, etc.) to complicated combinational circuits such as a microcontroller or multi-core processor. The same FPGA circuit can be used to make improvements to the existing hardware logic by reconfiguration. The hardware design for a FPGA is written in HDL (hardware description language). Verilog and VHDL are the two common and popular HDLs. Written hardware design is synthesized in a bit file which is then used to configure the FPGA.

### 2.2 Cryptography

The security of hardware and software system relies on the resilience of the system from adversaries who intend to exploit and harm the system. In the early days of cryptography, statistical analysis along with intelligent guesswork and pen and paper method were the known techniques to break the cryptographic schemes. Rudimentary cryptographic algorithms were less complicated mathematically. After the invention of the electronic computers, crypto schemes became easier to break and since then many complex cryptographic schemes began to be developed.

Cryptographic algorithms used today aim at providing integrity, authenticity and confidentiality service based on the assumption that the algorithm is computationally hardened [6]. Various cryptographic algorithms available today are public knowledge, but the necessary element of this algorithm, which is the cryptographic key, is unique and user-specific for example, symmetric key cryptographic schemes. Some cryptosystems which use reusable keys are known to be broken by the use of brute force attacks, but the time and resources required to break the cryptosystem depend on the length of the key. Security of any cryptosystem can be defined by the measure of time and work taken to attack and reveal the correct key. An ideal computationally secure cryptosystem will be difficult to break by using any practical known or unknown methods to reveal either the secret message or the key.

In theory, algebraic cryptoanalysis technique can disable the crypto cipher by representing the cryptographic algorithm solvable complex mathematical equations and solutions for the equations can provide information about the secret key. If an attacker is able to find the weakness in the algorithm, or by gaining the knowledge of the intermediate computational values of the implemented cipher;. A low-cost, passive attack called side channel analysis attack, is a unique category of attacks which do not rely solely on the mathematical deduction but also considers the physical characteristics of the electronic systems which are running hardware or software implementations of cryptographic algorithms. Side channel analysis (SCA) is a technique which exploits the physical parameters of the hardware running crypto engine. In this thesis, the target of the side channel attack is the FPGA hardware implementation of the 128-bit Advanced Encryption Standard (AES). Cryptographic algorithms such as DES, RSA , ECC are also vulnerable to SCA.

### 2.2.1 Basics of Cryptography

The encryption process takes plaintext and a secret key as an input and produces encrypted ciphertext as an output. The decryption process takes ciphertext

and the decryption key as an input and produces plaintext as an output. A cryptosystem is the combination of encryption and decryption process. Cryptography is defined in two broad categories: 1. Asymmetric key cryptography, and 2. Symmetric key cryptography. Asymmetric key or also referred to as public-key cryptography algorithm uses a pair of a secret key or a private key and shared key or public key, that are mathematically related. The public key is used for encryption by the end-user and a private key is used to decrypt the message. Security of the system depends on the secrecy of the private key and trust of the users. The different public key crypto algorithms such as RSA (Rivest, Shamir and Adleman 1978), Elliptic Curve Digital Signature Algorithm (ECDSA) are used by the cryptocurrency market, Digital Signature Algorithm (DSA).

The second category of the cryptographic system is a symmetric key cipher. Here, crypto algorithms are public knowledge and only shared secret is the key used for encryption process. The long byte size of the keys used in the symmetric key algorithm makes the cryptosystem resistant to brute force attacks. This category of cryptography is relatively faster than the public-key cryptosystems. Secure distribution of the key is a very important security aspect of the symmetric key cryptography.

#### 2.2.1.1 Block Cipher

One of the types of the symmetric key cipher is a block cipher. Block ciphers operate on a grouped bit of plaintext and produce the ciphertext in the same number of grouped bits. Such grouping of bits in a specific size is called blocks. Block ciphers operate on fixed block sizes at a time. Two commonly used block ciphers are the Data Encryption Standard (DES) and Advanced Encryption Standard (AES). DES has an operating block size of 64-bits and fixed cipher key length of 56-bits plus 8-bit parity, making 64-bit key size. AES offers three different key lengths; 128-bit, 192-bit, and 256-bit. The size of processing block is fixed at 128-bits for all three variants of the AES ciphers.

Block ciphers are round-based algorithms. Each round consist of all or some of the functional operations, for example, S-box (substitution box), bit permutation, arithmetic calculations, and exclusive OR (XOR) operation. S-boxes perform the mapping operation of non-linearly mapping input bits to output bits by substitution. Each iteration of the block cipher performs all the round functions in sequential order. Ciphertext is a final output generated at the last round of encryption and inputs of each rounds are outputs of the previous round.

### 2.2.2 Advanced Encryption Standard (AES)

Data Encryption Standard (DES) is a predecessor of Advance encryption standard, that has key size of 64 bits. The DES was broken, and new standard required a length of the key to be 90-bits or more. Advanced Encryption Standard (AES) was developed as a replacement and improvement to the Data Encryption Standard (DES). A block cipher developed by Vincent Rijmen and Joan Daemen was selected as the AES which is also called as the Rijndael cipher [1]. In October 2000, Rijndael was formalized and since then it is being used in many commercial, government and military applications all over the world.

AES is a symmetric key block cipher which has the processing block size of 128-bits and a basic processing unit of one byte with three different key lengths of 128, 192 and 256 bits [7]. The AES algorithm is performed on the two-dimensional array of 16 bytes called State. The State is a 4x4 matrix of four rows and four columns. Each cell of the matrix is of one-byte size. Depending on the key length being used for the AES, the number of rounds in the AES algorithm operation vary from 10, 12 or 14 rounds for 128, 192- and 256-bit key length respectively. Each round of the AES consists of four sequential operations: [7]:

- AddRoundKey (ARK): The State matrix is XOR-ed with a Round key matrix of the same dimension.

- SubBytes (SB): Each cell byte value from the XOR-ed state matrix is replaced by substituting another byte from the S-box table based on a one-to-one mapping.
- ShiftRows (SR): The last three rows are left-shifted by an offset of 1, 2 and 3 respectively for rows 2, 3 and 4.
- MixColumns (MR): The State matrix columns are mixed in defined order using a linear process.

Out of all the operations mentioned above, only the SubByte operation is a non-linear process. This prevents linear and differential cryptanalysis. Carefully choosing the S-Box is important for achieving the concept of confusion by breaking the linear relationship between plaintext and ciphertext. S-box is derived from a finite Galois Field  $GF\ 2^8$  with a total of 256 elements [1]. The original crypto key is expanded into a total of ten more 128-bit keys for AES-128, and this done by key expansion process which is done before the encryption or decryption begins. Similarly, for AES-192 and AES-256, key expanded into thirteen and fourteen keys respectively. The first round key of any AES operation is the original cipher key.

For a 128-bit key, it is arranged into a 4x4 state matrix array with each cell holding one key byte. Each column of this matrix is considered as a word. Hence, the first column is word  $W_1$ , second  $W_2$ , and so on. This column of four words is further expanded into a total of 44 columns, which are called a key schedule. AES rounds use four words in each round utilizing 11 keys from 44 column key schedule. Figure 2.1 shows the key expansion and arrangement of 44-word key schedule. First round of AES utilize the original key, which is the first four words from key schedule, to perform exclusive OR operation between 128-bit block of plaintext and 128-bit key.



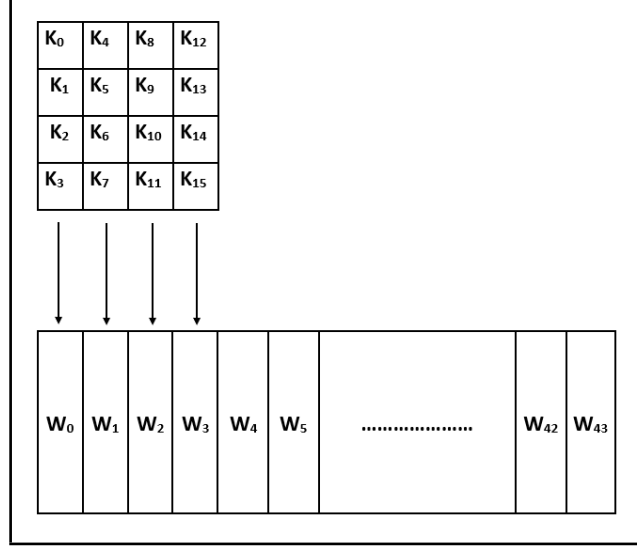


Figure 2.1: Expansion of 4-word AES Key into a 44-word Key Schedule.

The process of key schedule generation contains the following transformation steps:

- SubWord: It is a SubByte operation performed on each byte of the word using the S-box table to produce non-linear four output byte words.
- RotWord: Words from key State matrix are circularly rotated in left direction.
- AddRcon: Pre-defined round constants are exclusive-ORed with the output of RotWord operations. The three rightmost bytes of the round constant are zero.

Symmetries are avoided by using round constants [1]. Based on the method of implementation, the key schedule is calculated on-the-fly or in advance and stored in the desired memory. For the devices with limited available memory, the key schedule is generated at the runtime of the cryptographic operation, which also overwrites previous round keys. Figure 2.2 and Figure 2.3 shows the AES round function operations and overall structure of the AES-128 encryption process.

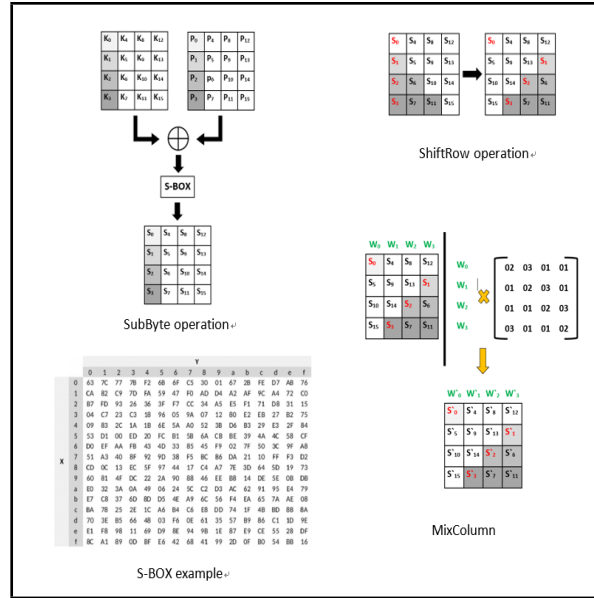


Figure 2.2: Operations of Round Functions in AES-128 Encryption.

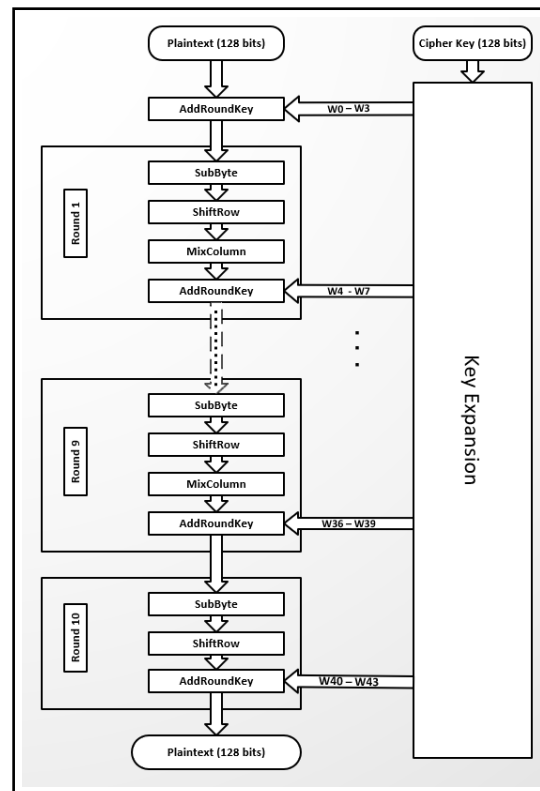


Figure 2.3: Architecture of 128-bit AES Encryption Algorithm.

### 2.2.2.1 AES: Modes of Operations

AES encryption is performed on a fixed data block of 128-bits. To process the large amount of data in separate 128-bit blocks, various modes of operations were created. Mode of operations allows the AES to perform crypto operations on a large chunk of plaintexts. As listed in NIST special publication 800-38A, different recommended modes of operation of the AES are Cipher Block Chaining (CBC), Electronic Codebook (ECB), Output Feedback (OFB), and Counter Mode (CTR) [8].

In an Electronic Code Block mode, large data is divided in to 128-bit chunks to be processed individually. If two different blocks of the message contain identical plaintext, then the encrypted ciphertext output produced will also be identical. This creates a lack of diffusion concept. Studying the pattern of such anomalies can reveal information necessary to extract the original plaintext. However, multiple blocks can be processed in parallel to avoid such anomalies.

In CBC, CGB, and OFB; the output of each encryption operation is used as an initialization vector for the next cycle of the encryption operation. In CTR mode, input blocks, referred to as counters, go through normal AES encryption except the output is again XOR-ed with the plaintext to generate the ciphertext. For the simplicity of analyzing each trace individually, ECB mode is considered as a mode of operation for this thesis to perform side channel analysis.

## 2.3 Cryptanalysis of Block Cipher

Cryptanalysis is a method used to break the ciphers. The prime goal of almost every attacker is to recover the encryption/decryption key since it is the smallest and most useful part of the secret to reveal all of the original plaintexts. An attacker can possess varying resources. The attack model to break the cipher depends on the amount of information available to the attacker. It is always assumed that the mali-

cious entity has a physical control over the encryption engine and since the algorithms are public knowledge, encryption key is the only secret element. Mentioned below is the classification of the possible cryptographic attacks on block ciphers [9].

- Ciphertext-only: The attacker has access to the full ciphertext. Mathematically complex ciphers are resistant to such attacks.
- Known plaintext: The attacker has access to both plaintext and ciphertext. Since it is possible that some similar information such as headers will produce identical ciphertext output, such attacks are realistic.
- Chosen plaintext or ciphertext: The attacker chooses their own set of selected plaintext/ciphertext to encrypt/decrypt and produce ciphertext/plaintext output. Intelligently selecting the plaintext can create a realistic attack model.
- Adaptive chosen-plaintext or ciphertext: Based on the information gathered during the attack, the attacker adaptively chooses different plaintext/ciphertext to perform cryptanalysis.
- Related key: Attacker changes certain key bits or bytes to study the relationship between different outputs. This attack is used along with one or more attacks models mentioned above.

The above mentioned attack models can be used for both algebraic cryptanalysis attack or side channel analysis-based attacks. A known plaintext attack is the most commonly used attack model [9].

### 2.3.1 Differential and Linear Cryptanalysis

These are the two established cryptanalysis attack methods against block ciphers. By statistical analysis, the attacker constructs a pattern over many rounds of crypto operation. The aim is to separate the probabilistic and random permutations and recover the key. As the name suggests, linear cryptanalysis technique looks

for the linear expressions to represent the cipher [10]. Statistical methods are used to find an approximate linearity between S-box inputs and outputs calculating the probability of how much the S-box input and output match with each other linearly. This method only works with a constant key value and it requires large numbers of known input plaintext. If the non-linearity of the S-box is evaluated, the attack can be performed without the knowledge of any other values of functional stages.

Assessment of differences between the pairs of plaintexts and their corresponding ciphertexts is a differential cryptanalysis method. Using such disparities, probabilistic assumptions can be performed on a set of different key candidates to identify the correct key [11]. Differential analysis attacks are most commonly used with chosen-plaintext and in some cases can work with chosen plaintexts [10]. AES was invented as a cryptanalysis resistant algorithm. It is resistant to both linear and differential analysis. Sets of different operations, linear and non-linear, included in each round of the AES ensures the high level of security through diffusion to make it resistant to linear and differential cryptanalysis [1]. Being statistical in nature, cryptanalysis attack makes it necessary to have a huge amount of plaintext/ciphertext data in case of both known or chosen text attack methodology. This makes such attacks impractical for complex crypto algorithms [12]. In contrast, algebraic cryptanalysis attack needs less amount of known plaintext/ciphertext data.

#### 2.3.1.1 Algebraic Cryptanalysis

Here, the crypto algorithm is represented as a set of polynomial equations and solving for those equations leads to exploitation of the algebraic structure. The attacker represents the transformation steps of the encryption process as a large system of a polynomial equations with a low degree and multiple variables. Solving for those equations discloses the information to recover the partial or full key. There are various methods to solve the systems of equations. A well constructed cipher, however, can make the systems of equations unsolvable. In theory, block ciphers are

describable with the systems of polynomial equations using the finite Galois Field. In practice, these polynomial systems are too complicated. However, the algebraic nature of the AES algorithm is prone to algebraic cryptanalysis attacks [13].

## 2.4 Side Channel Analysis

Any electronic computer system is defined as a system which processes and calculates some input data and generates corresponding output. In an ideal system, the output of the system is read or interfaced with another system via predefined output ports. This is the expected behavior of how the system should work. However, in practice, the output of the systems manifests itself through various undesired medium or physical phenomena. These are observable phenomena and are referred to as side channels.

Some of the known side channels observed and analyzed for a long time are power consumption [14], electromagnetic emanations [15], computation time [16], acoustic vibrations, and optical light emission. Most of the practical research has been mainly focused on power consumption, electromagnetic emanation, and timing information. Side channel attacks are cryptanalysis attacks classified under implementation attacks. Figure 2.4 shows the classification of possible implementation attacks on cryptography. Side channel attacks target the hardware implementation of the cryptographic scheme and not the abstract algorithm itself. Implementation attacks are of two main types, passive and active. In an active attack, the attacker tries to make functional changes in cryptographic hardware or software to alter the normal behavior of the system. Side channel attacks are passive attacks wherein attackers observe and measure the side channel leakage to analysis and attack the cryptographic implementation.

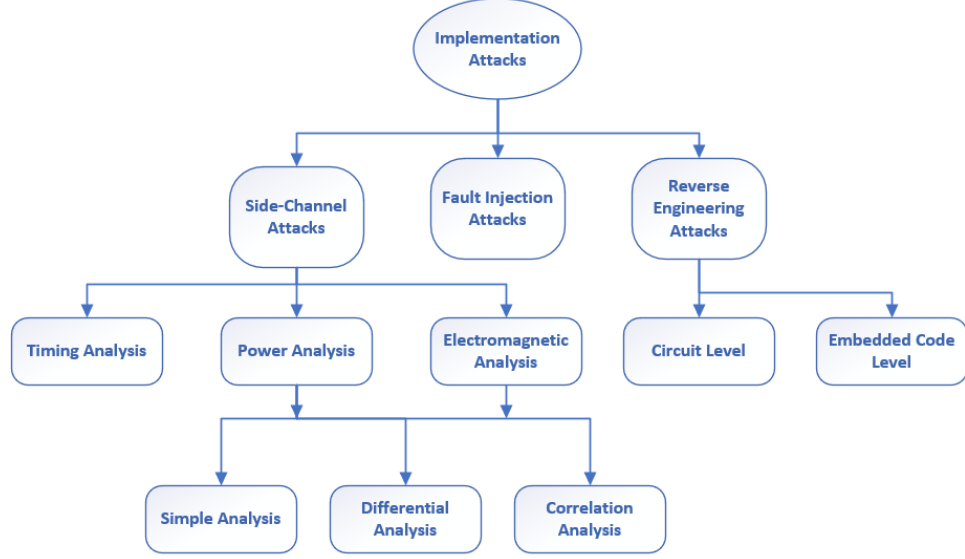


Figure 2.4: Classification of the Implementation Attacks on Cryptography.

#### 2.4.1 Power Consumption

All the components on the embedded electronics platform consume power based on operational needs. Field Programmable Gate Arrays (FPGAs) and almost all modern integrated circuits are fabricated with complementary metal-oxide-semiconductor (CMOS) technology. Figure 2.5 shows the basic CMOS cell NOT gate or inverter. The inverter is composed of two transistors, NMOS, and a PMOS transistor. NMOS transistor acts as a pull-down circuit when active, and PMOS transistor acts as a pull-up circuit when active. When the input to the gates of transistors is static, pull-up PMOS and pull-down NMOS network do not conduct at the same time. For  $V_{in} = V_{DD}$ , NMOS conducts and for  $V_{in} = 0$ , PMOS conducts. There is a very small amount of leakage current present for which static power consumption for static input supply is given as,

$$PC_{constant} = I_{leakage} * V_{DD} [17]$$

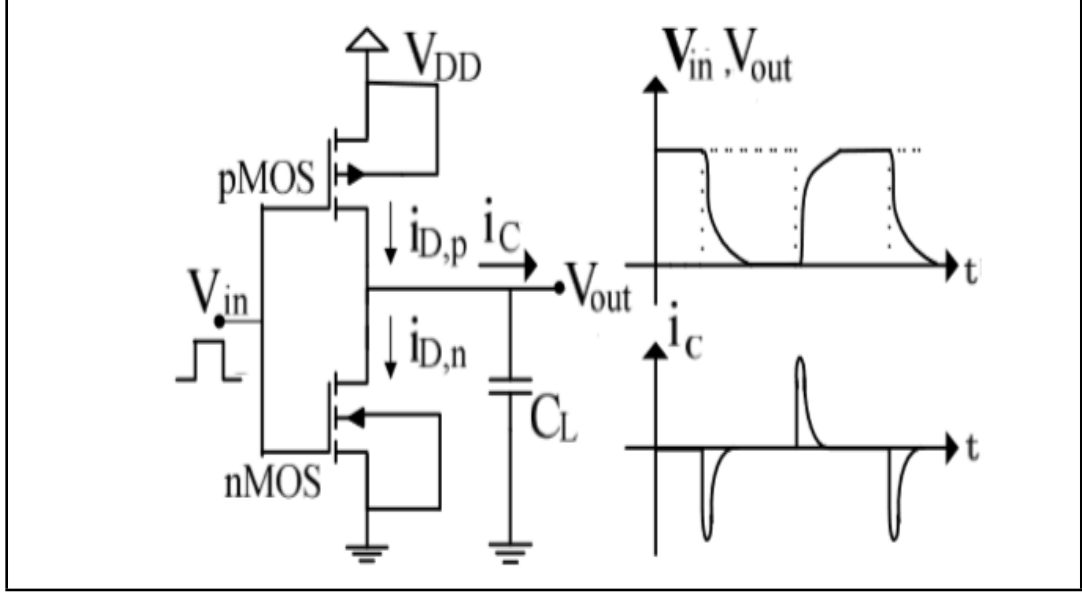


Figure 2.5: CMOS Inverter Circuit and its Input-Output Voltage and Current Waveform.

The main cause of side channel information leakage is dynamic power consumption. For a very small amount of time, transistor output is constant, but transistor output is frequently changing from 0 to 1 or from 1 to 0. Dynamic power consumption is the cause of two factors. First, is due to the amount of current drawn from power supply to charge the output capacitance  $C$  out between changing states of CMOS circuit. Every CMOS cell has an intrinsic capacitance along with the intrinsic capacitance of the wires connecting different CMOS cells. Short circuit in CMOS cells during bit transition, is a second reason for dynamic power consumption. Ideally, NMOS and PMOS should not be active at the same time, however, practical for a very short duration between the input and output state change, NMOS and PMOS both conduct simultaneously contributing towards more power consumption. Change in input causes changes at the output of CMOS; hence, dynamic power consumption is data dependent and is much higher than the static power consumption [17]. Modern integrated circuits operate at high clock frequencies, and that makes the amount of data processed equally high. This speed of operation factors into high dynamic



power consumption. Other factors contributing towards dynamic power consumption are instruction execution, reading or writing in data memory, and register files and type of data being used in all the operations.

#### 2.4.2 Electromagnetic Emanation

Electromagnetic (EM) emanations occur due to various coupling effects in electronic circuits. There are three types of EM emanations: radiative, inductive and conductive. Figure 2.6 shows the EM coupling types in electronic circuits. Time varying current flowing through electric conductors cause coupling effects. Internal transistor states frequently change with respect to clock speed, which in turn makes the flow of current time varying (based on the clock frequency) through CMOS cells and hence causing the coupling effect. Conductive coupling is observable at the physical conductive path between sources and termination points. Power lines, ground lines, onboard routing connections on PCB and connecting wires are areas where the conductive coupling is observed [18].

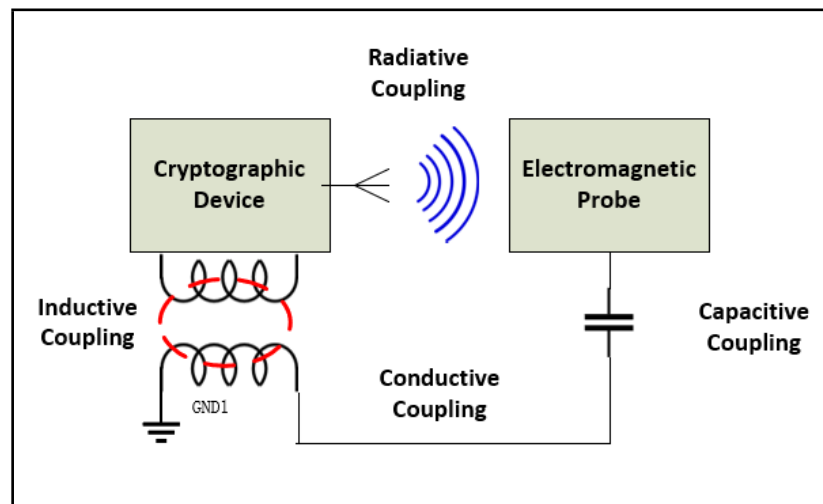


Figure 2.6: Types of Electromagnetic Coupling.

Current carrying wires generate an EM field. When two conductive elements (wires/traces) are placed next to each other within a distance less than the wavelength (near-field region), a coupling effect occurs due to mutual induction. As defined

by the Faraday's law of EM induction; two conductors placed in close proximity induce voltage when one conductor is carrying a time varying current. Capacitive and inductive coupling favours the emissions of the high and low frequency signals respectively. When conductive elements placed more than a wavelength apart from each other; one element acts as an antenna radiating unwanted EM waves [19].

#### 2.4.2.1 Types of EM Emanations

Based on its origin, EM emanations are classified into two categories, 1. Direct/intentional emanations, and 2. Unintentional emanations [18]. Direct emanations are the result of the intentional flow of current. For example, during the state changes in CMOS cell, there is a current burst or current spike for short duration which causes direct EM emanations observable over a wide frequency range. For an attacker, high-frequency components from such emanation are most useful as they are a direct result of the state changes in CMOS circuit. Low-frequency components are mixed with a lot of noise and interference.

Over the past two decades, fabrication technology for CMOS integrated circuits has drastically changed. In adherence to Moore's law, the density of transistors on the same size has multiplied. The submicron/nanometer deep CMOS integrated circuits have resulted in an increase in EM emanations due to electromagnetic coupling. Such coupling effects, so far ignored by designers as insignificant, have become a rich source for capturing data dependent EM leakage. As stated previously in [18], unintentional emanations are manifested due to modulation of the square-wave clock signal, which is present in all the modern circuits.

Unintentional emanations propagate more effectively than direct emanations. This makes it easier for an attacker to probe these emanations without using any invasive techniques and with a significant air gap. To capture direct emanations, invasive techniques such as decapsulation of chip packaging is necessary to get a good result [20].

## 2.5 Side Channel Attacks

Cryptographic algorithms such as AES and DES are developed with an assumption that the hardware they are implemented on is secure. The developers focused on securing the cipher algorithm to be resistant to algebraic cryptanalysis attacks by strengthening the computational complexities of the cipher. Another assumption was that the designed cipher acts as a black box within which cipher operations are safely carried to convert plaintext to ciphertext. However, one type of implementation attacks, i.e. Side Channel Analysis (SCA), target the device hardware running cryptographic implementation. Under SCA, the aim is to capture the information leakage from the available side channels and to acquire knowledge of the data being processed inside the hardware. Common side channel attacks are based on power analysis [14], EM analysis [18][20], and timing analysis [16].

### 2.5.1 Adversary Model

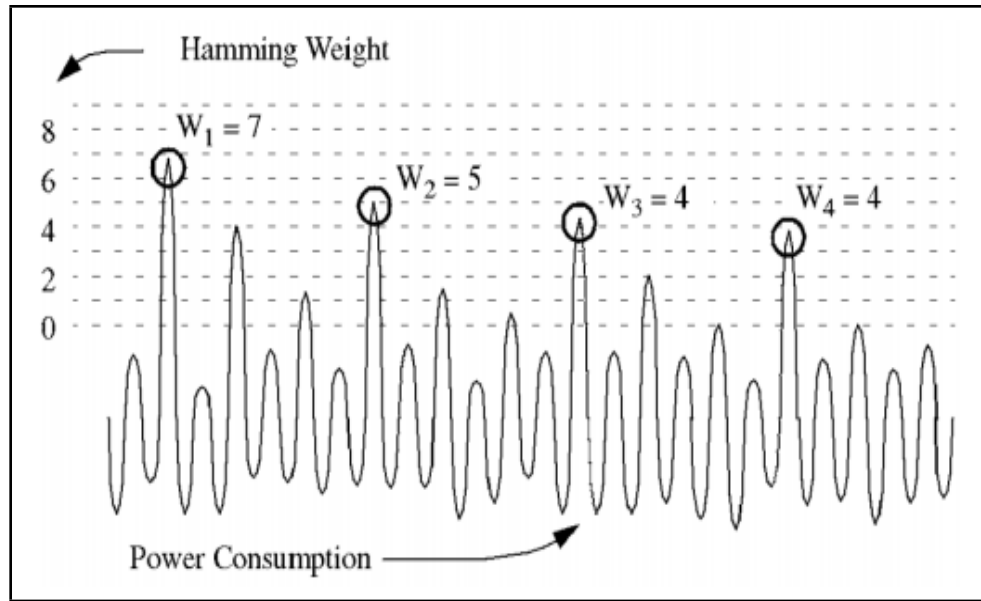
Individual attackers possess varying abilities and knowledge. In theory, strength of the side channel attack is dependent on the strengths of an adversary and availability knowledge and resources. A strong adversary will have full control over the device under threat as opposed to a weak adversary. A powerful attacker is able to find strong attack points by identifying different input and output points, and observe cryptographic operations in various ideal/non-ideal environmental conditions. Based on the availability of the side channels, an attacker may perform different optimizations to improve the quality of the side channel, for example, well timed trigger signals to identify the start or end of the encryption operation or decapsulation of IC package to improve side channel signal strength. In some exclusive cases, an attacker may be able to load its own choice of the cryptographic key into the crypto device. A weak adversary may possess limited knowledge of the system or optimization techniques. Due to less control over the device, measurements obtained by a weak adversary could

produce erroneous results.

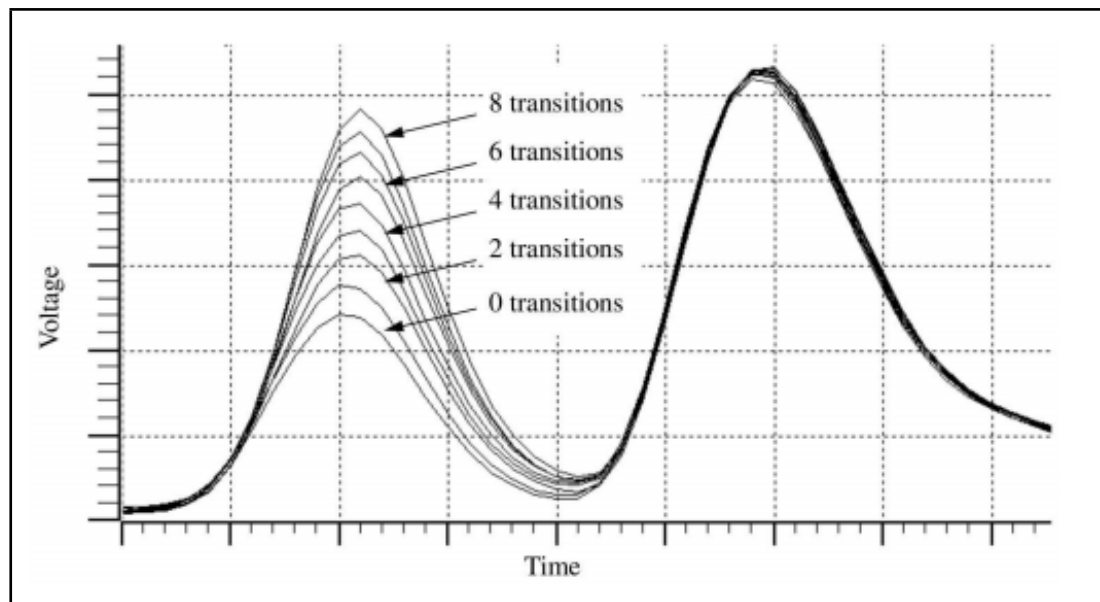
#### 2.5.1.1 Hamming Weight Model

Hamming Weight (HW) corresponds to the total of the number of bits which are equal to binary digit 1. The HW model, proposed and proved by P. Kocher [21], assumes that the dynamic power consumption of the CMOS device is correlated with the number of bits set to 1. At any instance of time, the power consumption is dependent on the number of 1s processed at the same instance when the power values are measured, and independent of any values processed before or after the instantaneous process. By the physics of the CMOS power consumption theory, which states that CMOS power consumption is a direct result of the transition of the 1 to 1 and vice versa, the HW model is contradictory. The HW focuses on the number of 1 set in an (8-bit) word at a specific instance. For performing the electromagnetic side channel analysis on the first round of AES-128 implementation, the HW model is the most efficient approach. For example, consider that the output of the operation is being stored on an 8-bit register initialized to value zero. Power consumption readings, as a direct result of storing values on the register, will be a figure of the number of 1 in the 8-bit register. Figure 2.7 (a) shows how the high HW corresponds to high power consumption and for lower HW power consumption is low.

The HW model is best suited when there is very little or no knowledge available of the underlying hardware cryptographic implementation under attack. It is advantageous over the Hamming Distance (HD) model as the HW model does not necessitate having data available from the encryption operations, such as ciphertext necessary for the HD model, while attacking tenth round of AES encryption.



(a) Hamming Weight and its corresponding power consumption



(b) Hamming Distance and its corresponding power consumption

Figure 2.7: Hamming Weight and Hamming Distance Power Consumption [22].

### 2.5.1.2 Hamming Distance model

Eric Brier et al., proposed a Correlation Power Analysis attack using the Hamming Distance (HD) model in 2004 [23]. Based on works mentioned in [21] and [24], the HD power model makes the following assumptions:

- Power consumption relates the transitions of bits 0 to 1 and 1 to 0 in CMOS circuit.
- Power consumption is constant when there is no bit transition occurring.
- Equal amount power is consumed for both, bit transitions 1 to 0 and 0 to 1.

The HD model is applied when extra information about the hardware cryptographic implementation is available. In the case of AES, output ciphertext is used to generate the HD power model to attack tenth round of the AES operation. Consider two values  $X_0$  and  $X_1$  and the HD of these two values is calculated by performing XOR operation and measuring the HW i.e.  $HD_{(X_0, X_1)} = HW(X_0 \oplus X_1)$  [17]. Figure 2.7 (b) represents the superimposed waveforms of varying the HD values and their corresponding power consumption. High power consumption relates to the greater number of bit transitions.

## 2.6 Types of Side Channel Analysis Techniques

### 2.6.1 Timing Side Channel Analysis

The concept of the timing attack was proposed by P. Kocher in 1996 [16]. Time taken by the cryptographic algorithm to perform all the operations could be dependent on the encryption key. In his experiment, he performed encryption operations on a set of different data using the same encryption key. Timing analysis was performed on the crypto algorithms like RSA, Diffie-Hellman, and DSS. In modern instruction set architectures with pipelining and caches, individual programs have different execution speed and completion time. Hence, such variation in program

execution time can leak information about the underlying process under execution. In 2005, Daniel J. Bernstein [25] demonstrated a practical timing attack on OpenSSL AES implementation to successfully guess the secret key.

### 2.6.2 Electromagnetic and Power Analysis

Gandolfi et al. and Quisquater et al. introduced EM analysis in 2001 [20][26]. Since then, side channel attacks and countermeasures have been vastly researched. Measurement of the side channel leakage signal over a particular run time of any cryptographic operation is referred to as a trace. Secret information from the device can be obtained by correlating the observable behaviors such as power consumption, EM signal, and timing information with the internally processed data of the device.

### 2.6.3 Simple Side Channel Analysis

In some cases, details about cryptographic operations can be analyzed by Simple Side Channel Analysis (SSCA). Under Simple Power Analysis (SPA) [14] and Simple EM Analysis (SEMA) [20][26] techniques for power and EM side channel respectively, EM or power traces can be directly interpreted to gain information about operations executed by the device to infer the crypto key. Weaknesses in the algorithms such as conditional jump operations dependent on key bits can reveal unique information [14]. The source of measurement for the power and EM traces is different, although the technique to perform SSCA is similar for both. Figure 2.8 is a diagrammatic representation of measuring EM emanations from a microcontroller or an FPGA using an electromagnetic magnetic field (H-field) sensor, for example, a simple loop antenna. Figure 2.9 (a) represents an experimental setup for SEMA using H-field probe and Arduino UNO microcontroller running a simple square wave generation program. Figure 2.9 (b) Shows the single trace capture of the square waveform and its corresponding EM emanation waveform.

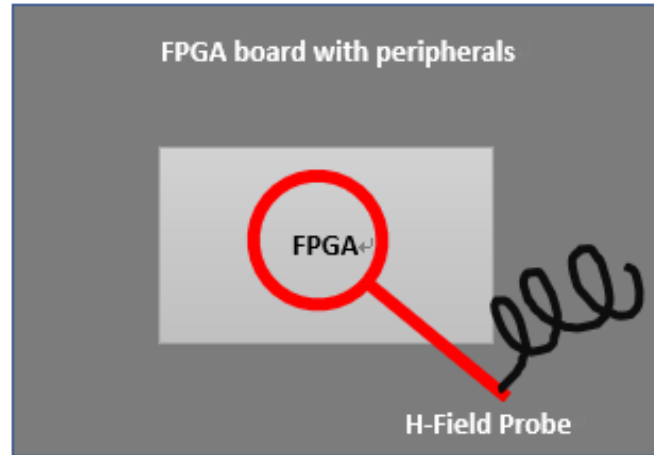


Figure 2.8: An Example of EM Emanation Measurement on FPGA.

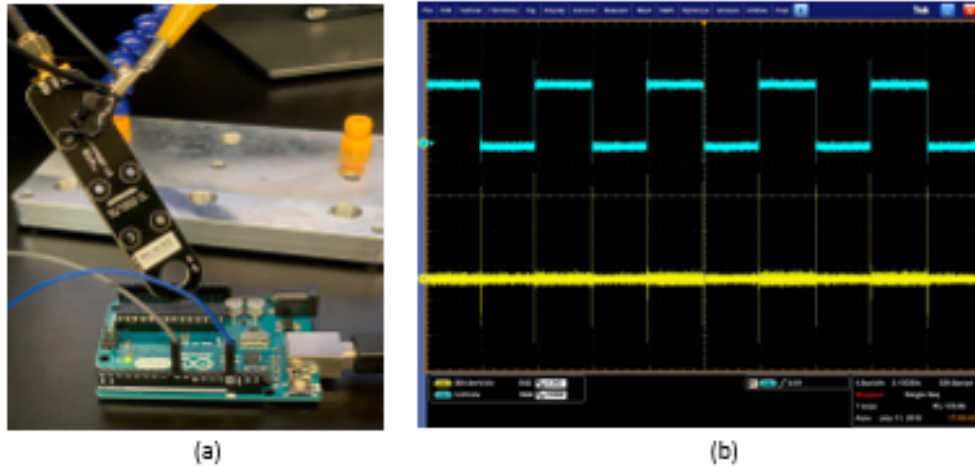


Figure 2.9: (a) SEMA on Arduino UNO Microcontroller Running Square Wave Generator. (b) Captured EM Emanation for Square Wave (Blue) EM Emanation (Yellow).

By doing a simple observation of the EM trace visible on an oscilloscope, an attacker can identify specific details about the crypto operations. From Figure 2.9 (b) it is evident that the rising edge of the square wave appears as large voltage spikes in the EM trace and the falling edge of the square wave appears as the small voltage spike. SEMA technique requires little to no preprocessing operations of the captured traces. In most cases, a voltage or current amplifier, for H-field probe or B-field probe respectively, is required to capture EM emanation in a noisy environment. It



is important for an attacker to possess detailed knowledge of the target encryption algorithm to perform successful SEMA. SEMA benefits in a scenario where an attacker gains access to the crypto device for a very short duration and can capture a few EM traces. Based on the amount of traces captured, there are two types of SEMA; Single-shot SEMA and Multi-shot SEMA. Single-shot SEMA is performed with using only single EM traces while Multi-shot SEMA uses multiple EM traces which could help to extract more information of the encryption process.

#### 2.6.4 Differential Side Channel Analysis

Differential analysis is an efficient side channel analysis technique because in contrast to SEMA, here an attacker does not need to possess knowledge about the target crypto device. Differential analysis works effectively for the target hardware, which produces a lot of noise. In comparison to SEMA, a large number of EM traces are necessary to perform successful Differential Electromagnetic Analysis (DEMA). Hence, it is important for an attacker to have full physical access to the cryptographic hardware for a longer duration than SEMA.

SEMA is more of a visual observation technique of analysis on one EM trace at a time. DEMA uses statistical methods to extract key information. The concept of DEMA is an extension of the Differential Power Analysis (DPA) [14]. The definition of DPA is extended to DEMA as it also exploits the data dependency of the EM emanations of the cryptographic hardware similar to power consumption. A large number of EM traces are required for analyzing the EM emanations of the cryptographic operation within fixed points on the time axis. An early practical EM side channel attack on AES using differential analysis technique is shown in [27]. The attack was conducted on a small Personal Digital Assistance (PDA) device with Java-based AES software implementation. It is important to note that the side channel analysis on the software implementation of the AES can be performed using few EM traces when compared to attacking hardware implementation of the AES.

### 2.6.5 Correlation Electromagnetic Analysis

Correlation analysis is considered the generalization of the differential analysis. Eric Brie et al. first introduced Correlation Power Analysis (CPA) in 2004 [23]. Correlation analysis technique uses Pearson Correlation Coefficient (PCC) value as an identifier to find a relation between captured traces and a hypothetical model of traces, which is a hypothetical model based on all possible key guesses based on the certain plaintext byte and key byte. The Xilinx Kintex-7 FPGA chip is used as the device under attack in this thesis experiments. Research work presented in [28], [29], [30], and [31] show some of the initial experiments performed on the hardware implementation of AES encryption on FPGA chips demonstrating the correlation based EM side channel attack. All the steps of Correlation Power Analysis are applicable for Correlation Electromagnetic Analysis (CEMA).

For CEMA side channel attack, EM leakage signal traces are captured at runtime when the FPGA chip is running hardware implementation of AES encryption. Afterward, the statistical analysis method is used to calculate PCC, which provides enough information about the correlation between EM traces and key bytes. Nowadays, the correlation analysis method has become an efficient and commonly used method to perform side channel attacks. The PCC method is used to derive a linear relationship between the two sets of data. In CEMA, one data set is the actual captured EM signal traces, and the second data set is the hypothetical guessed values for the EM leakage signal for individual key byte. CEMA can work with either the HW model or the HD model. Steps to perform CEMA are described in the next chapter.

## CHAPTER 3: Proposed Framework

Power side channel attacks based on differential and correlation based analysis have received more research attention as compared to EM side channel attacks. EM side channel attacks have been proven to be more efficient than the power side channel as they are of non-invasive nature and do not require physical tampering with hardware. Since the physics behind the origination of the power consumption or EM leakage signal is almost the same in CMOS devices, it is assumed that the countermeasures developed for the power side channel will work with the same efficiency for EM side channel. In most of the recent literature on the CEMA attacks, largely the HD model has been used to focus the attack point on the last round of the AES encryption. However, using the HD models makes the CEMA attack computationally more complex and lengthier. The HD model for CEMA requires it to store ciphertext values from the AES operations, and these ciphertext values are used as known data sets while performing CEMA attacks. In the aspect of the time required to perform the CEMA attack after capturing the EM leakage, the HD based model takes more than twice the amount of time required to perform CEMA using the HW model. Although there are several preprocessing techniques used on captured traces to reduce the attack time for the HD model, this only adds up to more time to perform the attack. Despite being time efficient, the HW model is less efficient compared to the HD model because the later provides a close approximation of the EM emanation signal leakage model. In this thesis, I propose the HW model for CEMA to perform the EM side channel attack on the first round AES-128 encryption. The proposed framework will also employ preprocessing techniques using the Principle Component Analysis (PCA) method mentioned in [32] to improve the performance and efficiency

of the proposed attack. This chapter discusses steps to perform CEMA using the HW model, a preprocessing technique for noise reduction, and PCA transformation for dimensionality reduction and noise reduction method as a performance improvement method for CEMA.

### 3.1 Basics of Correlation EM Analysis

Based on the Hamming Weight model, it is determined that there is a linear relationship between the EM leakage signal from cryptographic hardware and the corresponding data processed by the same hardware. Hence, the underlying processed data can be considered to be the secret encryption key being guessed, and the possible magnitude of the correlation between the captured signal and the data can be estimated by using the Pearson's correlation coefficient [33] method. The sample point in the EM trace exhibiting high correlation magnitude indicates the correct key byte guess. Pearson's correlation coefficient  $C$  is calculated between reference value  $R$  and hypothesized value  $H$ . PCC values range between -1 to 1 where a correlation values close to 1 indicates high correlation, a value close to -1 indicates low correlation, and value 0 indicates no correlation. The equation for calculating the correlation coefficient is given as follows:

$$C_{R,H} = \frac{E(R \cdot H) - E(R)E(H)}{\sqrt{Var(R)Var(H)}}$$

here,  $E$ = Mean and  $Var$ = Variance

Before performing CEMA, the first important step is to determine the Intermediate Value  $I_{val}$  of AES algorithm as a target point. This value has to be the function of the part secret key and the random input plaintext data or ciphertext data. This selection function of key  $k$  and plaintext data  $d$  is represented as follows:

$$I_{val} = f(d, k)$$

Intermediate values for CEMA attacks are chosen from the output of the S-box operation from the first round of encryption. More specifically, the Hamming Weights

of the intermediate values.

In the next step, a different set of 128-bit fixed size plaintexts are provided as input to the device for encryption. While the encryption process is running, EM signal traces are captured and saved. Each captured EM trace will correspond to the one 128-bit input plaintext. The input plaintext data can be represented as a vector  $d = \{d_1, d_2, d_3, d_4, \dots, d_D\}$ . Captured EM traces for individual plaintext data can be denoted as  $t_i = \{t_{i1}, t_{i2}, t_{i3}, t_{i4}, \dots, t_{iT}\}$ . Next, the captured traces are stored in matrix  $T$  of dimension  $M$  by  $N$ , where  $M$  rows are the total number of EM traces captured and  $N$  columns are the total number of sample points in individual EM trace. Since calculating the correlation coefficient requires two quantities to correlate, a hypothetical Intermediate Value matrix  $H$  is generated as a reference data to be correlated with the EM trace matrix  $T$ . Matrix  $H$  will contain all the possible key values representing the single byte of the secret key. Key hypothesis vector  $k$  is given as  $k = \{k_1, k_2, k_3, k_4, \dots, k_K\}$ . Hypothetical value matrix  $H$  is computed with data vector  $d$  and key vector  $k$ .  $H$  is given as  $H = h_{(i,j)}$ . Here,  $h_{(i,j)} = f(d_i, k_j)$ . In calculation steps for the hypothetical value matrix  $H$ , the key value for the selected key byte is varied from hexadecimal  $\text{hx00}$  to  $\text{hxFF}$ . If considering  $k_j = \text{hx00}$  as one possible key byte, then the AddRoundKey operation is performed by completing the XOR operation between  $k_j$  and the corresponding plaintext byte from each individual plaintext input  $d$ . The output of the XOR operation is passed to the S-box input for generating non-linearly transformed the S-box output. The Hamming Weight of the S-box output is calculated and stored in hypothetical value matrix  $H$ . This process is repeated for all possible 256 key guess values and for each individual plaintext input. Hence, the dimensions matrix  $H$  will be 256 rows and the number of columns equal to a total number of input plaintext/EM traces.

Elements stored in matrix  $H$  are considered as EM leakage signal, and the value from matrix  $H$  are correlated with actual EM signal traces matrix  $T$ . It is

important that either one of the dimensions of matrix  $H$  and  $T$  should be similar. The correlation coefficient is calculated between each column of matrix  $H$  with each column of matrix  $T$ . The result is stored in a new matrix  $P$ . The correct key guess value is obtained from matrix  $P$  by finding the index value for the array element having the largest correlation coefficient. The row index will indicate the correct key, and the column index will indicate the time.

### 3.2 Principal Component Analysis Based Preprocessing Technique

The Principal Component Analysis (PCA) technique is widely used to extract only the most relevant data points from a large chunk of data and to reduce the dimensionality of the data. The PCA also helps to reduce noise when considering a data set of power/EM trace signal sample points. The inventor of the PCA method is the same person who invented the correlation coefficient method, Dr. Karl Pearson [34], and it was improved by Dr. Harold Hotelling, to the modern day PCA technique [35]. PCA evaluates data points with large variance and finds linear combination of such high variance data points. These combinations are then split into Principal Components (PC) where data points with higher indexes contain more relevant information from the data. This method can be used in side channel attacks as a preprocessing technique to eliminate noise components from the captured power/EM traces. PCA helps to reduce the data dimensionality to decrease the time to perform CEMA/CPA.

To understand the process and statistics of PCA, consider an example of data with two dimensions (a,b) with fifty observation points [36]. Figure 3.1 (a) represents the plot graph for the given data. The first PC has a large variance in data points. The second component has the second largest variance and is orthogonal to the first PC. After the original data is transformed using the two principal components, the data set is presented in Figure 3.1 (b).

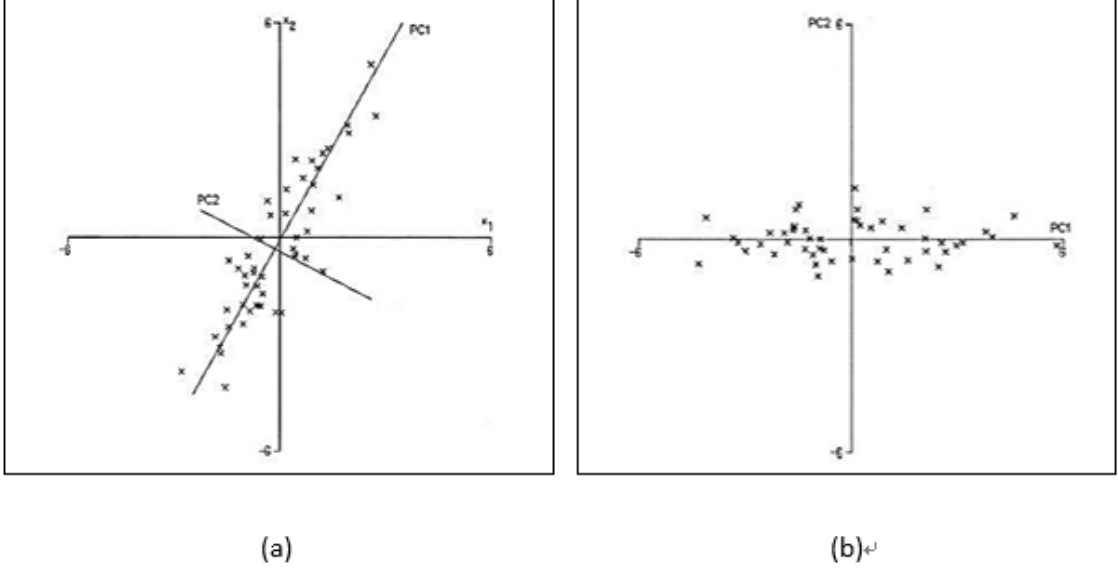


Figure 3.1: (a) Dataset Plot with the First Two Principal Components (b) PCA Transformed Data Using the First Two Principal Components.

It can be observed from the plots that the data points in the direction of PC1 have large variances than the data points in the direction of PC2. Applying the same PCA method to the EM signal measurements, in which the data set is composed of dimensionality i.e., a number of columns match the number of sample points, and the number of rows matches with a number of total EM traces captured. The total number of PC generated will be equal to the number of sample points. This section describes PCA transformation and selection of an appropriate PC.

### 3.2.1 PCA Calculation Procedure

PCA is used to isolate only the relevant information data points from large data sets. Captured EM traces contain large sample points, and it will help to extract only the data points which are highly correlated with the actual data processed by cryptographic hardware. PCA can help extract such sample points by removing the sample points associated with the noise signal. Steps to perform PCA [37] as a preprocessing technique on an EM trace dataset is as follows:

- Calculate a row wise mean of  $n$  dimensions (sample points) over all the captured traces  $T_i$  and calculate the mean vector  $M_n$ .

$$M_n = \frac{\sum_{i=1}^T T_{i,n}}{n}$$

Next, subtract mean  $M_n$  from every column dimension  $n$  for all traces  $T_i$ .

$$T_{i,n} = T_{i,n} - M_n$$

- Calculate the covariance matrix  $Cov$  which is a square matrix  $n \times n$  with dimensions equal to  $n$  dimensions (sample points).  $(i, j)^{th}$  element in the matrix will correspond to the covariance between  $i^{th}$  and  $j^{th}$  dimensions from individual EM trace. The main disadvantage of PCA is, a large number of sample points increases the calculation time to generate covariance. The covariance matrix for two dimensional data  $A$  and  $B$  is given as follows:

$$Cov(A, B) = \frac{\sum_{i=1}^n (A_i - A')(B_i - B')}{n - 1}$$

Here,  $A'$  and  $B'$  are the row-wise means of the  $A$  and  $B$  respectively.

Applying the same formula, the covariance matrix for mean centered data  $T_{(i,n)}$  is given as follows:

$$\sum_{i,j}^{n \times n} = C_{i,j}, C_{i,j} = Cov(Dim_i, Dim_j)$$

Here,  $Dim_x$  is the  $x^{th}$  dimension.



- Next, extract the eigenvectors and eigenvalues from the covariance matrix.

$$\sum^{n \times n} = \mathbf{U} \times \mathbf{\Lambda} \times \mathbf{U}^{-1}$$

Here, the eigenvalue matrix  $\mathbf{\Lambda}$  is made of all the diagonal elements from the covariance matrix, and  $\mathbf{U}$  is the eigenvector matrix with rows equal to total sample points. These two matrices provide relevant information about the main data set. The eigenvalue matrix is arranged in descending order of elements. The eigenvector corresponding to the largest eigenvalue is the first principal component.

- Select the necessary number of eigenvectors  $PC$  from matrix  $\mathbf{U}$  to generate a feature vector matrix  $F$  in which the eigenvectors are arranged in columns.

The feature vector matrix can be used in two ways. First, the original data can be transformed using  $PC$  to reduce the dimensionality. Second, data points contributing to noise readings from the original data set can be reduced by using some eigenvectors while keeping the original dimensionality of the data. The original data matrix  $T$  and the feature vector  $F$  can be used to reduce the dimensionality of the original data. The transposed featured vector is multiplied with the transposed mean centered original data matrix, which generates PCA transformed original data matrix  $T'$ . And  $T'$  is the original data represented in the form of PCA transformation with reducing dimensions.

$$T' = F^t \times T^t = (T \times F)^t$$

Here,  $t$  stands for the transpose operation.

For reducing the noise from the original data while retaining all the data dimensions, similar steps can be followed as mentioned for PCA transformation. For noise reduction, multiplication of the transposed feature vector and the original data matrix is multiplied again with the non transposed feature vector. Next, the result is transposed to get back original data  $T$  and by adding the mean which was subtracted

in the first step to noise reduction and same dimensionality. This process is called reverse PCA transformation. The equation is given as follows:

$$T_{noisereduceddata} = ((T^t \times F^t) \times F)^t - M_n$$

### 3.2.1.1 Selecting the Appropriate Number of Principal Components

It is important to decide how many PCs to use to transform the original data without using unnecessary principal components. There are several techniques mentioned in [36], including one method, the Scree test. This method provides a visual means by plotting the graph of eigenvalues to define a threshold to select the necessary number of PCs. Figure 3.2 is an example of such a Scree test plot. As the name suggests, the Scree plot visually looks similar to the gradient of the mountain slope. At a point in the plot, where the sloping line turns into a flat line, it is taken as a threshold point, and all the PCs falling before this point are deemed as most relevant PCs containing important information.

Scree plot contains two quantities which help to select right number of principal components. In Figure 3.2 the blue line depicts the scree plot and the orange line depicts the variance ratio. Scree plot is a plot of eigenvalue vector  $\Lambda$  of principal components where x-axis represents the number of principal components and the y-axis represents the eigenvalues. Eigenvalues are displayed in descending order and the highest value of eigenvalue represents the first principal component. Any eigenvalue greater than zero can be chosen as principal component (PC). First principal component PC1 with high eigenvalue captures most of the original information. And the next high value of eigenvalue corresponding to second principal component PC2, and captures less of original information than PC1. In given scree plot, eigenvalues reach to zero after first fifty principal components.

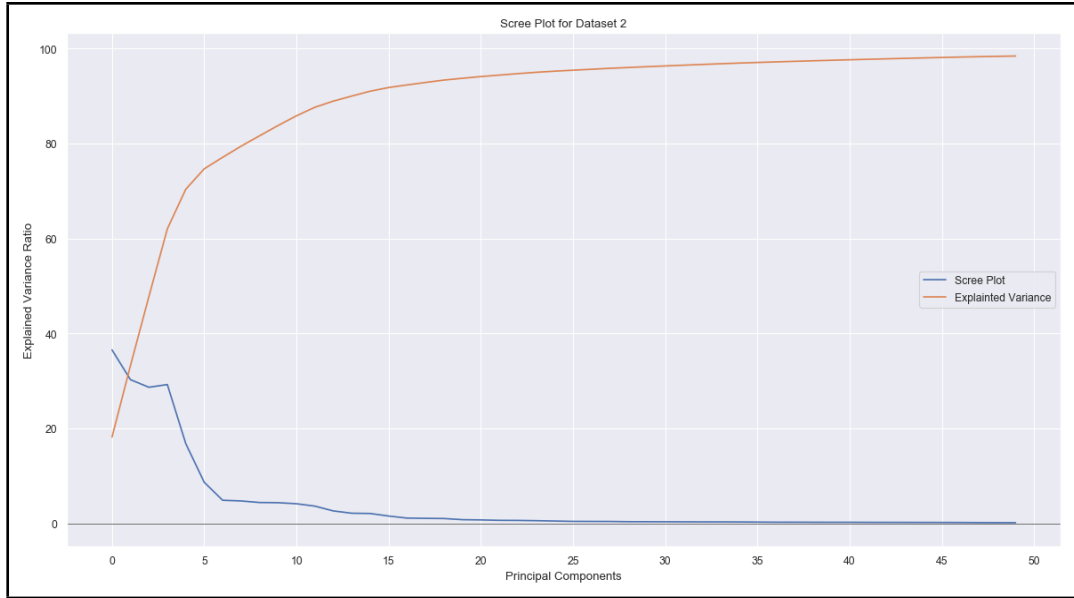


Figure 3.2: Scree Plot for PCA Transformation of EM Traces Captured for Key 2.

Explained variance ratio provides the information of how much percentage of original information is captured by different numbers of PC. Variance ratio is plot a plot of PC on x-axis and the ratio of the total variance divided by individual variance of each PC in percentage. Orange line of explained variance ratio in the figure shows that the fifty PC captures/explains almost 99% of the original information.

In the EM side channel analysis experiment, each captured EM trace contained two hundred sample points. Principal component analysis technique reduced the sample points between nineteen and fifty for different plaintext and encryption key datasets used. Reduction in the size of dataset improved the analysis time of CEMA attack. Some non extracted key bytes, when performing CEMA attack without PCA for noise reduction, were extracted after using the PCA for noise reduction.

## CHAPTER 4: Experimental Setup and Description

### 4.1 Experimental Setup

This chapter provides a description of the experiment setup and correlation analysis of EM emanation side channel. Figure 4.1 shows the block diagram representation of the overall experimental setup. All the main components and their functions are mentioned below.

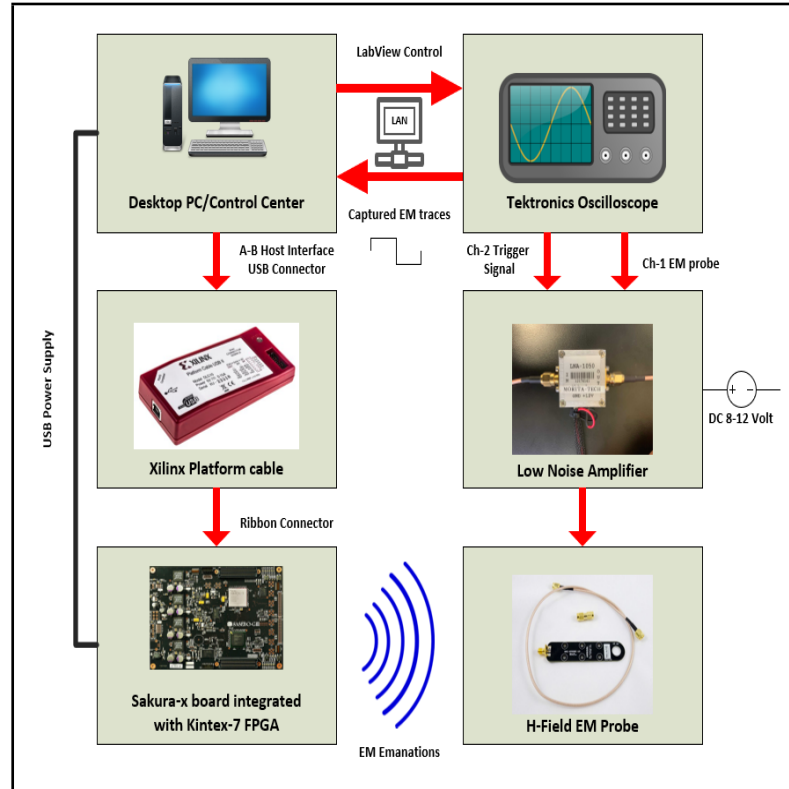


Figure 4.1: Block Diagram of Experimental Setup for CEMA.

- Desktop PC is installed with necessary software tools such as Python 3, LabView, Virtual Box Ubuntu Operating System, and Xilinx Vivado.
- LabView software is used to interface the oscilloscope with the PC to control

the operation of capturing EM leakage signal traces and stores it in a necessary format.

- Sakura-x is the main side channel attack experimental board equipped with the FPGA chip and pins to acquire the trigger signal for oscilloscope.
- Xilinx Vivado is a VHDL/Verilog programming tool, and it is used to program the FPGA board. A bitstream file for AES 128-bit encryption is used for the hardware implementation on FPGA.
- Xilinx platform cable is used to connect the Kintex-7 FPGA and the Xilinx Vivado programming tool.
- CEMA attack and preprocessing scripts are written in Python 3 programming language.
- Ubuntu Operating System installed on the Virtual Box is installed with the side channel analysis driver application.
- The EM probe is a 15mm loop antenna PCB which is used as a sensor element to capture the EM emanations.
- The low noise amplifier is interfaced between the oscilloscope and the EM probe to amplify the output of EM probe.
- The oscilloscope captures the EM traces with appropriate sampling rate settings.

Figure 4.2 shows the actual lab environment setup for the experiment to perform the EM side channel attack. Figure 4.3 shows the EM probe positioning on the FPGA chip.

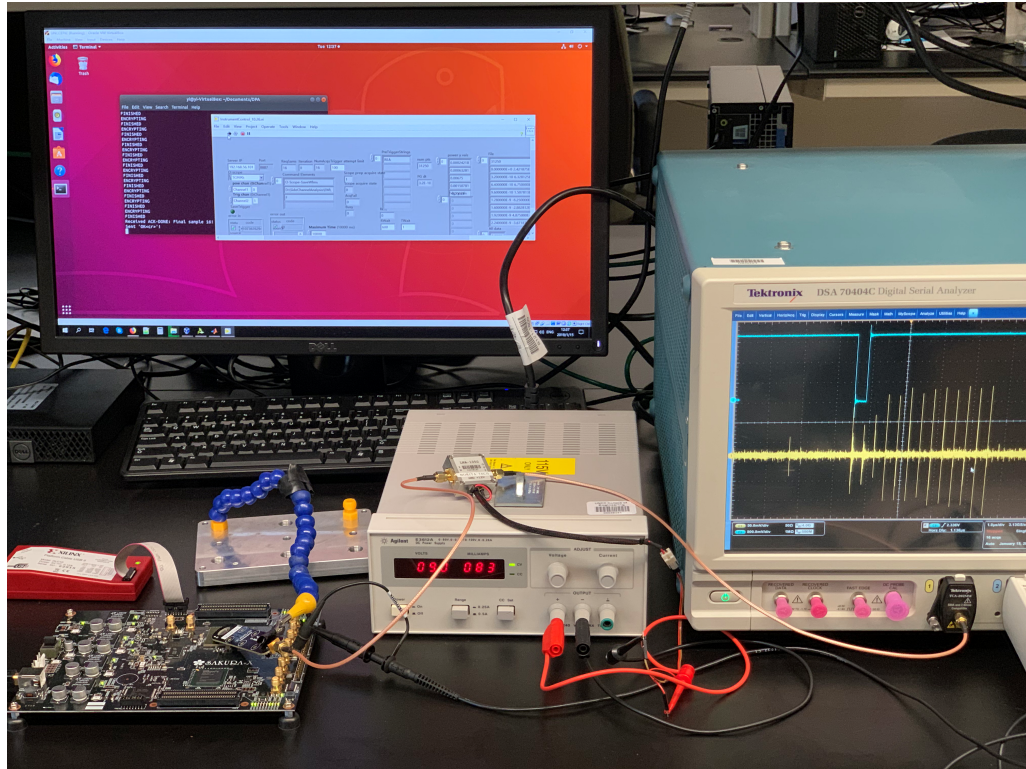


Figure 4.2: Experimental Setup in Lab Environment.

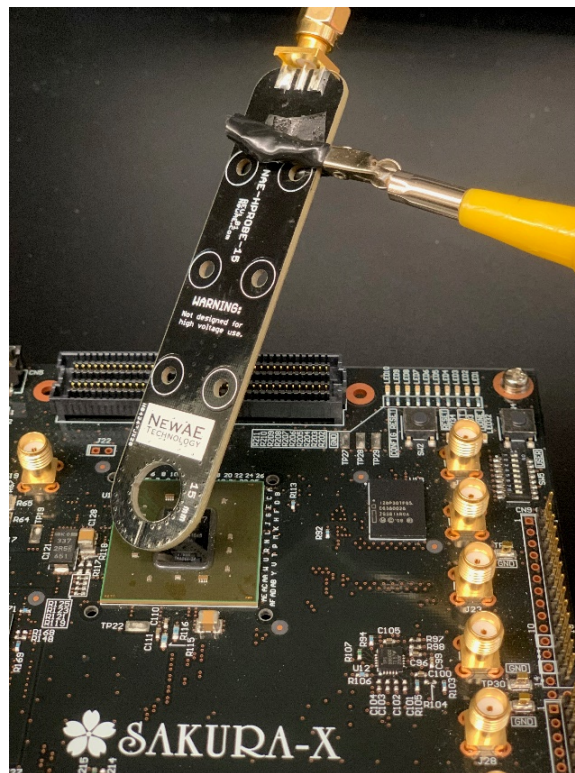
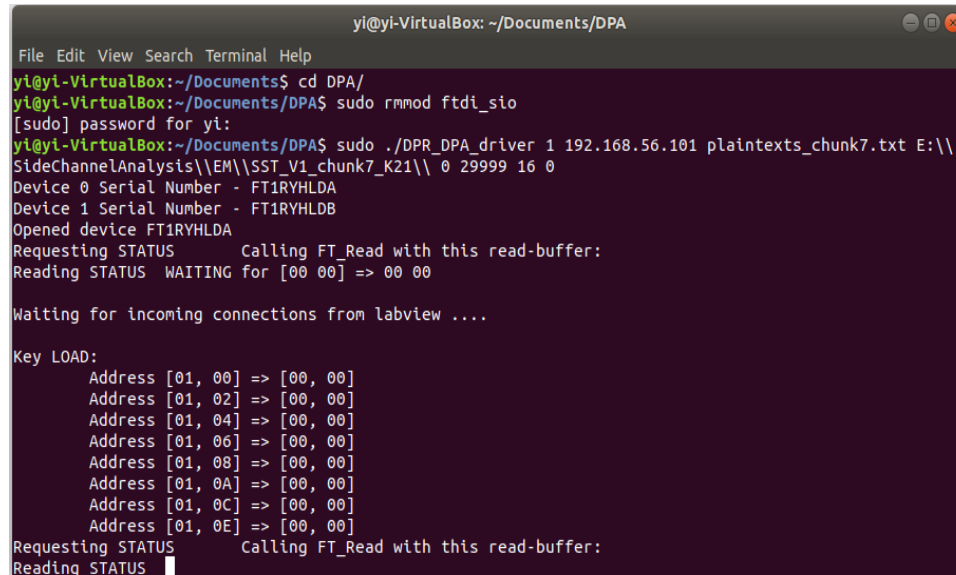


Figure 4.3: EM Probe Positioning on FPGA.

## 4.2 Software Tools and Setup

All the software tools used in the experiment serve the purpose of control and data acquisition. The side channel analysis drivers installed on the virtual Ubuntu operating system environment is the core of the control environment. The main script is written in C++ programming language, which is used to establish a connection between the PC and the Sakura-x experimental board. More specifically, the drivers connect the Host PC with the Kintex-7 FPGA cryptographic chip's flash ROM via JTAG-2 programming connector. The main script has two specific memory address locations for reading and writing operations to and from FPGA's flash ROM. The input plaintext and encryption key are stored in the write memory address in the block size of 128-bits.

Driver application is initialized with a specific command format which accepts values such as LabView status bit, LabView server IP, input plaintext file name and location, the storage location for the captured EM trace, and the total number of plaintext bytes (16 bytes for AES-128). Figure 4.4 is an example screenshot of the commands in the Ubuntu command prompt.



```

yi@yi-VirtualBox: ~/Documents/DPA
File Edit View Search Terminal Help
yi@yi-VirtualBox:~/Documents$ cd DPA/
yi@yi-VirtualBox:~/Documents/DPA$ sudo rmmod ftdi_sio
[sudo] password for yi:
yi@yi-VirtualBox:~/Documents/DPA$ sudo ./DPR_DPA_driver 1 192.168.56.101 plaintexts_chunk7.txt E:\\
SideChannelAnalysis\\EM\\SST_V1_chunk7_K21\\ 0 29999 16 0
Device 0 Serial Number - FT1RYHLDA
Device 1 Serial Number - FT1RYHLDB
Opened device FT1RYHLDA
Requesting STATUS          Calling FT_Read with this read-buffer:
Reading STATUS WAITING for [00 00] => 00 00

Waiting for incoming connections from labview ....

Key LOAD:
Address [01, 00] => [00, 00]
Address [01, 02] => [00, 00]
Address [01, 04] => [00, 00]
Address [01, 06] => [00, 00]
Address [01, 08] => [00, 00]
Address [01, 0A] => [00, 00]
Address [01, 0C] => [00, 00]
Address [01, 0E] => [00, 00]
Requesting STATUS          Calling FT_Read with this read-buffer:
Reading STATUS

```

Figure 4.4: Commands for Initiating CEMA Analysis Script.

The control flow of the program while initializing the CEMA attack script is as follows:

- The drivers first check the availability of the USB port which is used to connect the FPGA with the Host PC via USB platform cable.
- After the USB port is initialized, the drivers check for any programmed IP (Intellectual Property) programmed on the FPGA chip. It also checks for input/output buffers and registers specifying AES mode of operation for programmed IP.
- The drivers establish a connection with the LabView application based on the server IP and LabView port number. LabView application is installed on the Windows operating system.
- Before starting with the actual encryption operation, some test encryptions are performed to check to see if the AES working.
- Next, the encryption operation begins by selecting the address locations for loading 16 bytes of the encryption key and plaintext. Once the load operation is complete, encryption begins, and the application waits till all encryption rounds are complete.
- For each plaintext and key pair, encryption is performed sixteen times. Then the average of these sixteen trace signals is calculated, and the averaged signal is finally stored at the specified file location mentioned in the command.
- Encryption operation continues in steps, taking each 16 bytes of plaintext at a time.

The LabView application is used as an instrument control for the oscilloscope. It is interfaced with the oscilloscope over an Ethernet LAN connection. Figure 4.5 is a screenshot of the LabView control panel. The oscilloscope channels are specified



for the EM signal and trigger signal. The server IP is the same as the LabView IP used in driver application commands.

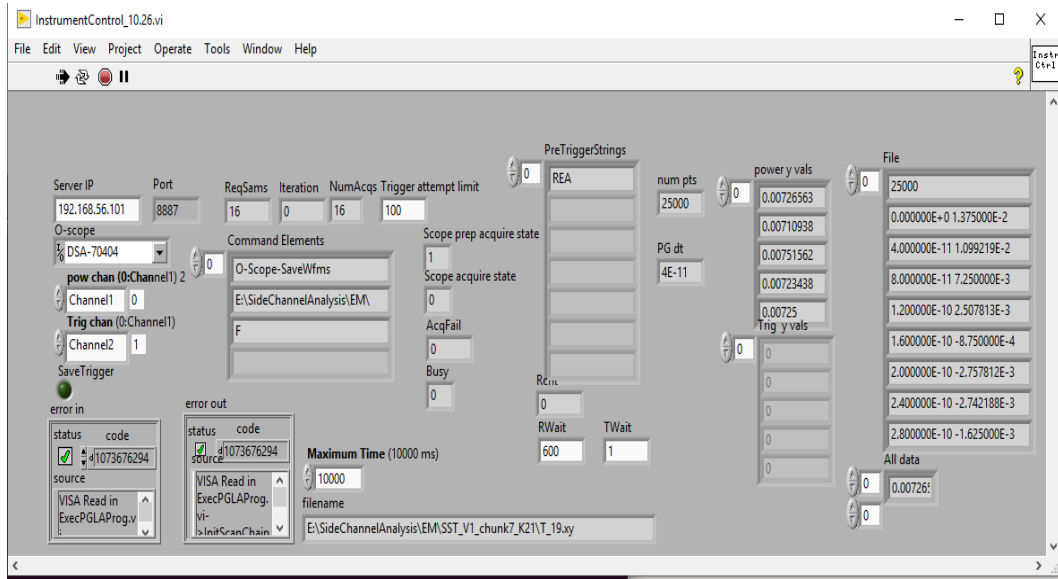


Figure 4.5: LabView Control Panel for Capturing EM Trace.

The Xilinx Vivado design suite, installed on the Microsoft Windows operating system, is used to write the AES encryption IP. The entire AES encryption code is written in Verilog programming language used for circuit design at the register transfer level. A trigger signal is programmed along with the AES code. This trigger signal indicates the start of AES encryption operation. The trigger signal is a square wave signal, and the rising edge of the signal is marked as the start of the AES encryption operation. All the input and output ports of the AES IP design are specified in the user constraint file.

Python programming language is used to write a program for correlation analysis, principal component analysis, and preprocessing related tasks. All the captured EM traces are stored in individual '<tracefile>.XY' file extension format. For example, if a total of 1000 input plaintext of 16 bytes each are used then, there will be 1000 EM trace files. The bash script in windows is used to change all the '<tracefile>.XY' file extension to '<tracefile>.txt' general text file format. Each trace file

consists of two columns, each one for x-axis and y-axis readings. To reduce the size of the data, all the trace files are consolidated in one single file with Python NumPy format. Only y-axis data column, which is voltage readings, is extracted from each trace file. Plaintext file is also converted into Python NumPy file format. Graphical representation of the CEMA results is generated using a Python script.

### 4.3 Hardware Tools and Setup

The host desktop PC, oscilloscope, target FPGA for side channel attack and the EM probe with amplifier are the main hardware components of the CEMA attack. All the hardware component work in synchronization to capture EM traces. Each individual component is configured specifically to achieve optimal results in a noisy lab environment. Here noise is referred to as electronic noise resulting through all the lab equipment in the vicinity along with the electronic noise generated from the Sakura-x board.

The Sakura-x experimental board is used to perform experiments on side channel attack and countermeasures. It consists of two FPGA chips, one is the Xilinx Kintex-7 FPGA, which is used as a target FPGA for the side channel attack, and the second is the Spartan-6 FPGA. Each FPGA has its own configurable flash ROM with dedicated interfacing ports. The Kintex-7 FPGA is programmed with target AES hardware implementation. The trigger signal is connected to one of the output pins available on the board. The pin is connected to the oscilloscope via a voltage probe. The USB connector on the board is used as both power supply and as a communication port between the Xilinx Vivado tool and experimental board. The target FPGA is programmed via an onboard USB port.

The CW505 Planar H-field EM probe is an inexpensive EM probe available in the market. It is a loop antenna designed on a PCB. The loop diameter is 15mm, and the trace width on the PCB is 1mm. The loop antenna alone is not sufficient to capture the low-frequency EM emanations, hence it is important to use a signal

amplifier to boost the EM signal and get a clear visible signal view on an oscilloscope. The amplifier used in this experiment has a frequency range of 10-1000 MHz. The amplifier requires an external power supply with a maximum voltage requirement of 12 volts. For this experiment the amplifier was supplied with only 8 volts of power supply, which was sufficient to get the desirable readings. Increasing the power supply to the maximum will increase the EM probe's sensitivity, and the voltage values on the y-axis will increase. Since the oscilloscope can capture only those signal components which are visually present on the oscilloscope screen, keeping the amplifier supply voltage at 8 volts helps to keep the voltage spikes within the screen limits.

A good high resolution oscilloscope is essential to perform a successful side channel attack. A signal sampled at an optimum sampling rate will provide enough data points to perform correlation analysis. For this experiment, I am using the Tektronix DSA 70404C oscilloscope. It has a maximum signal bandwidth of 4GHz and a maximum sampling rate of 2 GS/s (Gigasample per second). The sampling rate of 12.5 GS/s is used for this experiment, which produces 1250 sample points in the captured EM trace. It is equipped with 4 individual channels. Channel-1 and channel-2 is connected with the EM probe, and channel-1 is connected to the trigger signal pin on Sakura-x board. Since the side channel attack is focused on the first round of AES encryption, the EM signal for AES encryption is zoomed in to focus on only signal waveform for the first round with the help of the horizontal positioning knob along with time delay knob. The signal averaging function available in the oscilloscope settings can be used to reduce the noise present in the EM signal. However, it affects the speed of signal the capturing operation.

## CHAPTER 5: Electromagnetic Side Channel and Countermeasures

### 5.1 Electromagnetic Side Channel Analysis History

The EM side channel analysis technique was introduced in 2001. Afterward, a plethora of research papers on EM analysis were published in the years to follow. The following section describes the chronological details of some important publications on electromagnetic side channel analysis attack techniques.

- ElectroMagnetic Analysis (EMA): Measures and Counter-Measures for Smart Cards, 2001 [26]: This was one of the first published papers on EM side channel analysis presented by Jacques Quisquater and David Samyed. They study electromagnetic emanations from a smart card chip, that leaked information. The timing side channel is limited to single dimensional data processing with only timing information, and the power side channel used two-dimensional data representation for finding the correlation using timing and power information. However, EM radiations constitute information on three dimensions which include timing, power and frequency information. Different smart card chips were studied for EM radiation, and it was concluded that the frequency spectrum of EM radiations from each individual chip was unique. They graphically plotted the 3D signature of EM radiations using a motorized XYZ-axis rotation table. The paper also proposed some possible countermeasures against the EM side channel. Using a Faraday cage to reduce the EM emission was one of the suggestions. Another idea posited was to design the circuit in a way that the power consumption will be very low. Designing asynchronous systems, employing dual-rail logic design, using parallel architecture for chip designing, etc.,

were some other proposed countermeasures. Authors of this paper were the first ones to bring in use the terms "Simple EM Analysis" (SEMA) and "Differential EM Analysis" (DEMA).

- Electromagnetic Analysis: Concrete Results, 2001 [20]: K. Gandolfi, C. Mourtel, and F. Olivier published their work on EM side channel analysis at the same time the previous paper was published. This work also focused on EM radiations emanating from a smart card chip. For the first time, this paper presented a practical EM side channel attack on three individual smart card chips fabricated with CMOS technology. They also paid attention to the different EM probe designs and efficient ways of probe positioning to capture the best possible EM emanations. The probe used in their experiment was an ad hoc coil of copper wire. The differential EM analysis method was performed on three different encryption algorithms, namely COMP128, DES, and RSA. The results for the EM analysis were compared with the power analysis to reveal that the EM measurements provided better results than the power measurements. They stated that the global current consumption of the chip was much like a big river, which is the sum of all the tributary currents carrying different information.
- The EM side channel(s), 2002 [18]: The work presented by D. Agrawal et al. was based on the previous papers. However, they explained different aspects of the electromagnetic side channel. A detailed description was given on different sources of the EM radiations along with different types of EM wave propagation. The EM emanations were categorized into two types, direct/intentional emanations result of the intentional current flow and indirect/unintentional emanations, which were the result of the electromagnetic and electric coupling effect between the various electric components on circuit. Unintentional emanations revealed themselves through amplitude or angle modulations in communication circuits. All the categorization of EM emanations were based on the experiments

performed on the smart card chips. They also studied the EM emanation signal in the frequency domain to show how EM emanation signals were composed of different frequency signals carrying different information. A comparison between the EM and the power side channel showed that some part of the EM leakage signal was not significant, and it was a result of bad instruction. An experiment performed on different smart cards showed how a 'bad' instruction presented itself in the EM emanation signal, but not in power consumption signal. Some of the proposed countermeasures suggested were the reduction of signal strength by adding some unintentional emissions on purpose and using shielding to reduce signal strength.

- Multi-channel Attacks, 2003 [38]: Agrawal et al. introduced a concept of multi-channel side channel attack at the CHES conference in 2003. The idea was to use the EM side channel along with the power side channel. The techniques were introduced to select the best possible leakage signal from two different side channels. It was a generalization of a template attack and the paper focused on experimentally proving how multi-channel template attacks were more efficient than using a single channel template attack. Signals from multiple channels were concatenated. The experiment, however, did not yield the exact expected results as there was an apparent drawback of mismatch in the timing instance of information between power and the EM signal. Which lead to some erroneous results.
- Advances in side channel Cryptanalysis Electromagnetic Analysis and Template Attacks, 2003 [39]: Agrawal et al. presented another paper in 2003 on the EM side channel attack. This experiment was performed by placing the antenna source over the distance of 15 meters. The target hardware of this attack was a Linux server running on the Intel chip, and the target cryptographic algorithm was RSA. The EM sensor, an antenna, was placed in a different room and the

EM leakage signal was used to mount various differential EM analysis attacks based on timing information. The EM signal was captured at variable distances of 1.5 meter, between 3 to 5 meters, etc. The increase in distance increased the noise in the captured EM signal. However, this did not eliminate the possibility of a successful attack.

- Exploiting Radiated Emissions - EM Attacks on Cryptographic ICs, 2003 [40]: In all the initial EM side channel attacks performed on smart cards, the EM sensors were placed close (in near-field EM region) to the target chip. S. Mangard presented a method to perform an EM analysis attack on a smart card chip using an EM sensor and a biconical antenna which was placed over 5 meters distance (in far-field EM region). However, this experiment was performed in a room which provided shielding from the EM interference and other electronic noise sources. The experiment showed that by using such a shielded environment, the EM attack was successful after using only a few hundred EM signal traces. The same experiment used more measurements when performed in an unshielded noisy environment. A physical probing of a trigger signal was necessary for the latter experiment.
- Electromagnetic Analysis Attack on an FPGA Implementation of an Elliptic Curve Cryptosystem, 2005 [41]: All previous EM analysis attacks were targeted at software implementations of different cryptographic algorithms. In May 2005, P. Buysschaert, E. De Mulder et al. published their experimental work on one of the first EM side channel attacks on a hardware implementation of Elliptical Curve Cryptography (ECC). They used a correlation analysis technique to calculate Pearson's correlation coefficient between captured EM traces and the hypothetical EM power model. Their target hardware platform was the Virtex FPGA, and the target point of the ECC algorithm was 160-bit EC point multiplication. The simple EM analysis revealed the difference in EM signal

response of multiplication and addition operations. The EM sensor used in this experiment was an ad hoc loop antenna of a single loop of copper wire.

- Generalizing Square Attack using side channels of an AES Implementation on an FPGA, 2005 [42]: Another EM analysis attack on FPGA implementation of cryptography was presented by Vincent Carlier, Herve Chabanne, Emmanuelle Dottax, and Hervé Pelletier on August 2005. Target hardware was ALTERA Cyclone FPGA operating at a clock frequency of 50 MHz. This experiment targeted the hardware implementation AES encryption for the first time.
- EM Analysis of Rijndael and ECC on a Wireless Java-Based PDA, 2005 [27]: Gebotys et al. published their research on the first ever EM side channel attack on an actual embedded device. The attack was mounted on a wireless PDA device running on a JAVA platform. The target algorithm was a software implementation of a 128-bit AES encryption written in JAVA programming language. The technique they used was different from any EM analysis technique used before. They performed a differential analysis method on signals in the frequency domain rather than in a time domain. This was a new Differential Frequency Analysis (DFA) method. The EM emanations were captured using a 1cm loop antenna EM probe. The AES encryption was performed over the same input data 1000 times to capture 1000 EM signal traces. The captured traces were transformed from the time domain to the frequency domain by using a Fourier transform method. The main advantage of this method was that it removed all uncorrelated misalignment in the time domain. However, the fast Fourier transform method also does not contain timing information, and hence operations which were data dependent could not be distinguished. Also, in the frequency domain, information which generated short frequency spikes was not distinguishable. To overcome these drawbacks of frequency domain analysis, the authors used a spectrogram to study the spectral density of a signal over a



time axis. The same year they published another paper providing countermeasures for EM attack on a wireless PDA. The proposed idea was to mask the various S-boxes randomly. Each different S-box would have unique masking.

- Security Evaluation Against Electromagnetic Analysis at Design Time, 2005 [43]: This paper was a discussion on the design methods of a smart card chip to evaluate the EM leakage at the design time. The experiment was mostly a simulation of the different aspects of the electronic chip. The experiment studied the flow of the current, parasitic components of the chip layout, and direct and indirect EM radiation/emission. After simulating the results, they compared the outcomes with the measurements acquired from a processor running in synchronous and asynchronous mode operations. Data dependent emissions were most significant on the synchronous processor and the asynchronous processor exhibited time shifts with respect to the processed data.
- Security Limits for Compromising Emanations, 2005 [44]: Markus G. Kuhn wrote this paper to shed some light on compromising emanations in consumer electronic devices. He stated that most of the security standards for compromising emanations only existed for military application devices and consumer products were not subjected to similar standards and tests. While there are global standards available for consumer electronics to protect against radio frequency/electromagnetic interference, however, none of the standards consider the compromising emanation which leaks information. The aim of this article was to open a discussion to include security standards in consumer electronics for protection against EM sided channels. He provided an example of an eavesdropping attack on a digital display of the laptop. The attack was performed using an omnidirectional antenna to capture compromising emanations from the laptop over a distance of 10 meters. The attack successfully shows the reconstruction of the captured signal to recreate the text message that was

being displayed on the laptop. One possible solution given by Khun was to lower the signal power in communication devices by a million times to achieve signal attenuation and shielding. However, such a method is far from reality.

- **High-Resolution Side Channel Attack Using Phase-Based Waveform Matching, 2006 [45]:** Homma et al. proposed a technique to align the captured EM trace signal to improve the efficiency of the DEMA attack. The EM emanations captured without proper trigger signal cause time displacement within individual captured traces. It is important for statistical analysis to be successful that all the traces must be aligned accurately in the time domain to capture events occurring at a similar time instance. This paper proposed a phase correction preprocessing technique to correct such misalignments in captured signals. A reference waveform is used to align all the other traces based on a phase of a reference signal at a particular time instance. An Inverse Discrete Fourier Transform (IDFT) method is used to calculate the displacement measure between the two traces. Based on the reference waveform, the second waveform's phase is rotated in the frequency domain.
- **Remote Password Extraction from RFID Tags, 2007[46]:** Yossef Oren and Adi Shamir performed a new type of side channel attack on an EPC class 1 RFID tags. They observed the backscattered signal from the RFID tags, which shows that the signal transmitted from the RFID reader was modulated by RFID tags. Also, the modulation of the signal was dependent on the number of computations performed by the RFID tag chip. RFID tags were equipped with a directional antenna, which acted as a transceiver. The antenna reflected the signals received from the RFID reader with the addition of some high amplitude modulations. The intensity of such amplitude modulation was related to the volume of computations and hence, the power consumption of the RFID tags. To recover the password from RFID tags, wrong password or 'kill password'

was sent to the tag constantly and the reflected modulated signal was captured. The tag behaved in a certain way when processing 1 and 0 inputs, this behavior was exploited to recover the actual password.

- **Power and Electromagnetic Analysis: Improved Model, Consequences and Comparisons**, 2007 [47]: E. Peeters et al. presented a new and more accurate side channel attack leakage model referred to as "switching distance." This model proved to be more efficient to distinguish between transitions from 1 to 0 and 0 to 1. This model was efficient for the EM side channel attack. However, the model required an accurate positioning of the EM probe on the exact spot on top of the target chip.
- **A Phase Substitution Technique for DEMA of Embedded Cryptographic Systems**, 2007 [48]: This preprocessing technique was presented by Gebotys et al. as an improvement over the phase matching technique presented by Homma et al. in [62]. Phases of all the captured EM trace signals are changed with a reference phase measurement. According to the authors, this method works better for course temporal alignment errors. This technique is also suited under the conditions of random operations and delays.
- **Enhancing Correlation ElectroMagnetic Attack Using Planar Near-Field Cartography**, 2009 [31]: D. Real et al. proposed a new sophisticated technique to find a spot on a cryptographic chip which will find the best location to capture most correlated EM emanations for successful CEMA attack on the AES encryption. Their experiment was conducted using a table with x and y-axis movement. Two sets of plaintexts P1 and P2 were used as an input to the AES engine. The EM signal was captured over each spatial location of the chip while providing P1 once and P2 twice to the AES encryption engine. A ratio of the difference between the EM traces corresponding to P1 and P2 and the difference between EM traces of two P2 measurements at the same location was

calculated. The location, which provided the maximum difference in ratio, was chosen as the most favorable spot to capture the best EM emanation signal. This experiment reduced the number of traces required to successfully perform the CEMA attack.

- **Successful Attack of an FPGA-Based WDDL DES Cryptoprocessor without Place and Route Constraints, 2009 [49]:** This paper presented the first successful EM analysis attack on the FPGA implementation of the DES algorithm protected with dual rail logic design. Sauvage et al. used a cartography method to find the hot spots on the FPGA chip, which were giving out intense EM radiations. They showed that the leakage in WDDL was caused by the imbalance between two opposite logic placements. Though, WDDL brings a certain level of protection against EM analysis by significantly increasing the number of EM traces for CEMA. They concluded that a differential placement and routing for WDDL can make it more robust against EM side channel attacks.
- **Evaluation on FPGA of Triple Rail Logic Robustness against DPA and DEMA, 2009 [50]:** This paper studied the robustness of the triple rail logic countermeasure against EM side channel. Investigation of the dual-rail precharge logic countermeasure showed that it is not robust against the EM side channel analysis. The authors found that DRP logic was inferior to the triple rail logic. They also point out that merely balancing the flow of current and signal timing is not enough for the success of the design technique proposed. To produce a balanced EM emanation devoid of information leakage, it is important to balance the ground and power rails and proper cell placement. Their work also points to the important aspect of the EM side channel, i.e., the EM attack is highly dependent on the attack distance and the EM sensor accuracy.
- **Compromising Electromagnetic Emanations of Wired and Wireless Keyboards, 2009 [51]:** Martin V. and Sylvain P. introduced a method of capturing the com-

promising EM emanations from the keyboards. They attacked three different standard protocols of keyboards such as PS/2, USB, and wireless keyboards. Different environments with varying EM interference was considered for this experiment. The EM emanations were captured using a biconical antenna without using any filtering methods. Analog to digital converters coupled directly with antennas were able to capture raw sampled signal. A Short Time Fourier Transform method was used to process the captured signal and acquire information about time, amplitude, and frequency of the signal. This attack was performed inside a busy office, from outside of the office building, and in an anechoic chamber. A template of captured EM emanations corresponding to various keys was created and stored in a database. Later while performing the actual attacks, those templates were compared with the captured EM signal to find the unknown key pressed on the keyboard. Some of the countermeasures later introduced for such attacks included using signal jammers and unique fonts which reduce EM emanations.

- Far Correlation-Based EMA with a Pre-Characterized Leakage Model, 2010 [52]: The authors of this paper investigate the feasibility of the CEMA attack in a far-field region of up to 50 cm. All the previous experiments attack the cryptographic hardware in a near-field region of the chip or by focusing on localized EM emanations. This experiment attacks the FPGA hardware implementation of the AES encryption. They used the Hamming Distance leakage model to attack the last AES round. To obtain precharacterized data, the output signal from S-box for 256 input values was captured and later used as a template for attack over a distance. An important point this study reveals is that distance plays a very important role in a successful EM analysis attack. The CEMA attack was successful in the near field region and was best suited for EM analysis on microchips.

- Analysis of Electromagnetic Information Leakage from Cryptographic Devices With Different Physical Structures, 2013 [53]: This paper explores how the structure of the device, its size, and its shape, effects the EM radiation. The proposed scheme is based on the electromagnetic interference theory. They test the three factors which are part of the cause for the EM radiations. The first one is the source, i.e. the voltage potential between the power and ground line. Second is the path, which represents the coupling elements, which cause EM signal propagations. Lastly, the antenna itself which is a physical structure radiating EM signal. Tests are conducted on a cryptographic device model constructed with a two-layer PCB board. The supply current for the model is extracted from the actual cryptographic device running AES encryption. The board size and length of the wire between the model and the cryptographic board is changed. The experiment shows that as the length of the wire increases, the captured EM signal becomes more distinct, showing 11 peaks of AES encryption operation. However, for the smaller board size, the captured signal is stronger than the one captured with increased board size. The experiment also performed a CEMA attack on the EM signal captured from the cryptographic model device, which proved that small board size and longer wire length were causing more prominent EM emanations.
- Localized electromagnetic analysis of RO PUFs, 2013 [54]: One of the proposed countermeasures against EM side channel attacks were to implement a Ring Oscillator (RO) which are Physically Unclonable Functions (PUF). RO PUFs were proposed as a technique to add a true random number generator alongside a cryptographic implementation to add random noise. Dominik et al. showed in their paper that RO PUF implementation is still vulnerable to EM analysis attack using localized EM emissions. They used the cartography method to find the hot spot on the chip to reveal the locations of the ROs implemented

on the chip. The main objective of this attack was to obtain the frequencies related to different RO implementations on the chip. The paper also proposed two countermeasures; the first one was to randomize the placement and path of the implemented RO and the second, interleaved routing and placement of components.

- Experimental Demonstration of Electromagnetic Information Leakage from Modern Processor-Memory Systems, 2014 [55]: Previous work on EM analysis is focused on very specific cryptographic devices and systems. This paper has focused on the investigation of compromising EM radiation through modern day commercial computers, desktops, and laptops. Desktop microprocessors from manufacturers such as Intel and AMD were put to the test to experimentally show how those processors were causing both intentional and unintentional EM emanations. One test case shows that a program code specifically performing certain memory operations can generate EM emanations, which can be used to perform EM analysis attacks. Or similar programs software can intentionally transmit secret information through EM radiations. Other results also show that modern computer systems naturally cause compromising EM emanations through some unintentional repetitive operations. The experiment also showed that emanations were captured at a far-field region. Since this experiment was performed purely on EM emanations caused by different programming language constructs, proposed countermeasures in this paper suggest that making changes in the programming constructs such as if-then-else statements, for example, can mitigate the intensity of the EM emanations.
- Comparison of Electromagnetic Side Channel Energy Available to the Attacker from Different Computer Systems, 2015 [56]: As explored in the previous paper, this paper also investigates the compromising EM signal energy from modern commercial computer systems. However, this paper focuses on different specific

instruction execution and measures the corresponding EM emanations. FPGA, laptop, and desktop are the three devices tested under this experiment as these three systems show a similar trend on how they each radiate EM emanations. Different machine code instructions are executed on all three platforms to capture and measure the frequency of the radiations. They used a 16 turn loop antenna placed at 10 cm distance from the source to capture the EM emanations. Eight different memory and register instructions were executed on each system, and the corresponding EM signal energy was tabulated for each instruction. The experiment result shows that the frequencies of the operation directly affects the energy of the EM radiation. Switching between frequency causes stronger emanations and consequently, more EM radiation energy.

- Electromagnetic Analysis Method for Ultra-Low Power Cipher Midori, 2017 [57]: A low power consuming cipher Midori was developed for small IoT devices to maintain overall low device power consumption. The Midori cipher is so far the lowest power consuming cipher available and was released in 2015. This paper investigates the resistance of this low power cipher against EM side channel analysis by performing a CEMA attack. The Midori cipher operates on a 64-bit key and 17 rounds of encryption and decryption. Structure of the cipher is almost similar to the AES cipher with initial round performing XOR operations on the plaintext and key block. Each round of cipher has three stages; Substitution, Shuffling cell, and MixColumn. The cipher key schedule generates only two keys, and each key is alternated between rounds. The EM emanations were captured in two stages, in the first stage the CEMA attack was performed on the last round of cipher focusing on the SubCell operation and in the second stage attack was focused on the second round of the cipher. The Hamming Distance power model was used to perform CEMA. In both stages of CEMA attacks, all 64 key bits were recovered using up to 20000 EM traces, proving



that this new low power consuming cipher is also vulnerable to CEMA attacks.

## 5.2 Countermeasures Against EM Side Channel Analysis

After the discovery of various side channel analysis techniques, researchers started devising countermeasures to make the cryptographic device resistant to side channel attacks. Some of the countermeasures proposed for EM side channel analysis date back to the year 1940. Then, Bell Labs suggested three countermeasures, which are also valid today, and they are as follows:

- Shielding of electronic components, circuits, and wires to reduce EM radiations.
- Using appropriate Signal Filters for power lines to avoid propagation of information signals through power lines.
- Using masking techniques, which is a signal encoding technique.

Although these countermeasures are basic and only considered the physical aspect of electronic equipment, many countermeasures proposed for cryptographic systems find their root in those fundamental ideas.

Countermeasures against EM side channel can be developed considering two points; first, trying to reduce unintentionally emanations by redesigning the circuit and second, increasing the amount of noise signature in EM emanation to decrease the signal to noise ratio (SNR). Faraday cage structure is considered very efficient at blocking EM radiations. Use of the Faraday cage like design on Integrated Circuit (IC) packaging or on the entire electronic device was one of the proposed methods to counteract EM side channel attack. Such a method could not be a practical solution since removing the Faraday cage by tampering with the device was one simple possibility. Also, it was not a viable solution when considering the size and the cost of the electronic product. Modern efforts to design countermeasures are focused on making changes at the algorithmic level or at the transistor level. Consideration of area and

cost for additional logic implementation is still a major bottleneck in the practicality of including countermeasures. It is important to note that no countermeasure developed so far has been able to completely eliminate the EM/power side channel. In the best case scenario, a good countermeasure is able to increase the cost, time, and efforts of mounting a side channel analysis attack.

As mentioned in Chapter 2, the power consumption of the device as well as the EM emanation is dependent on the physical structure of the CMOS gates and input and output of the CMOS gates. There is a correlation between data under process and the corresponding power consumption pattern and EM emanation. Considering correlation analysis side channel attacks on the hardware implementation of AES cipher, an attacker targets the weak points such as the calculated intermediate values in the structure of AES execution, for example, the output of the first round and the last round of AES encryption. The attack focuses on the mathematical calculations and its corresponding power consumption or EM emanations. Masking and hiding are two countermeasure techniques widely implemented today which focus on the weak target points of ciphers and use obfuscation to counteract the correlation or differential side channel analysis attacks.

### 5.2.1 Software Level Countermeasure

#### 5.2.1.1 Masking

The objective of masking [58] is to bring randomization in the calculation of Intermediate Value (*IV*). A cryptographic algorithm is executed normally, but masking makes the *IV*s independent from the encryption key. Masking does not create any changes in the EM emanation/power consumption pattern resulting from the processing of *IV*. EM emanation is not reduced by using the masking technique; it only masks the actual *IV* by some other value independent of key. Masking can be done by performing any linear combinations of logical or arithmetic operations on *IV*. The combination of such logical operations is referred to as a mask. This

mask is reversible when performing the decryption algorithm. Exclusive-OR (XOR) operation is one of the examples of commonly used masks. XOR masking can be given as  $V_m = V \oplus m$ , where  $V$  is the intermediate value and  $m$  is a randomly selected value and the XOR operation is a selected mask operation. The XOR mask can also be replaced with arithmetic operations such as addition, multiplication, or modulo operation. Power consumption related to the processing of  $IV$  remains uncorrelated since  $V_m$  and  $V$  are uncorrelated. Selecting an appropriate mask is very important as the mask is an additional operation in the cipher algorithm. A wrongly selected mask can impact the speed of execution and efficiency of the cryptographic algorithm. It is advisable to use two masks instead of just one since masking two masked values will ensure that the final output is masked.

#### 5.2.1.2 Hiding

The primary objective of hiding [58] is to make power consumption/ EM emanations uncorrelated with the processed intermediate values and any other processed data. Further, the signal to noise ratio can be minimized by changing the hardware implementation of the cryptographic algorithm using hiding methods. The hiding countermeasure is implemented at the algorithmic level, which included randomization of the intermediate values. Some of the examples of hiding techniques are adding duplicate instructions, invoking random interrupts, changing clock duty cycle or frequency, and applying different clock sources with different frequencies. By using the hiding technique, timing instance of the operation is also randomized since additional hiding operations will utilize a few clock cycles to execute. The hiding technique does not eliminate the EM emanations; it only increases the cost and time of side channel attacks similar to the masking countermeasure. One way algorithmic hiding can be achieved is by shuffling the order of execution of the cryptographic algorithm. In AES encryption, the algorithm can refer to more than one S-box table, and some random S-box lookups can be inserted with the actual S-box lookup to obfuscate the power

consumption pattern. In practice, using a combination of hiding methods can work efficiently for the custom implementation of the cryptographic algorithm. Shuffling the sequence of the algorithm does not affect the final throughput. The countermeasure techniques discussed above are implemented through making changes in the cryptographic algorithm. These changes are at the software level and hence the countermeasure implementation is software level too. Also, the hiding and the masking techniques are not ideal countermeasures against side channel analysis attacks. Intermediate values are not changed completely, but only randomized through masking, and this process is still reversible. By measuring more power or EM signal traces, hiding and masking countermeasures are defeatable using CEMA/CPA or DPA attacks. As investigated in [59] and [60] the masking technique is not efficient against DPA attacks. First, the characteristics of the power consumption of the CMOS circuit and the EM emanation are both dependent on the processed data. And secondly, correlation based attacks are the improved generalization of the DPA attacks; it is safe to assume that the masking and hiding countermeasures can be defeated using CEMA/CPA attacks. Based on software or hardware implementation of the cryptographic cipher, particular countermeasures are devised respectively. For software level countermeasures the categorizations are an assessment of the leakage, static code transformation, and code morphing. One of the methods to assess the side channel leakage is called Test Vector Leakage Assessment (TVLA) [61] [62].

#### 5.2.1.3 Test Vector Leakage Assessment

TVLA [61] is a statistical tool which can detect the side channel leakage of the device under test without having to perform the side channel analysis attack on the cryptographic implementation. Many hardware testing tools are used in the development phase of the electronic device to check for the functionality of the device. The test is conducted by generating all possible input test vector using the architectures of the algorithm to be implemented on the hardware. The generated test vectors are

then applied to the cryptographic device as inputs, and their output response is assessed for the correctness of the functionality of the device. For example, test vectors generated for the AES cipher will contain sets of plaintexts and keys. TVLA tries to check for the impact of the sensitive intermediates on the test vector input data.

Generated test vectors are separated into two different sets. One set contains a fixed set of plaintexts and one key for those sets of plaintexts. The second set included the same fixed key and the random sets of plaintexts. A hypothesis test is carried out, where the null hypothesis assumed that the mean and variance of the two TVLA sets is similar. To prove the null hypothesis false, Welch's T-test [62] is conducted on the two sets. This test determines whether the two sets reveal enough information to prove the null hypothesis wrong. The reason for having two different vector sets in TVLA is to determine whether the leakage is dependent on the plaintext or the key. Having a fixed set of plaintexts undergo encryption with a fixed key, collected EM or power traces will be a function of the fixed plaintext set. The result of the TVLA analysis only reveals the side channel attack possibility by showing that the emanations or power consumption pattern is still correlated with the internally processed data [63]. Assessment of the possible information leakage from the cryptographic device can help the designers to make changes in the algorithm or the hardware design to minimize the leakage and build the side channel attack resistant device.

The equation for Welch's T-test is given below:

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{S_1^2}{N_1} + \frac{S_2^2}{N_2}}}$$

Welch's T-test calculation between two sets of data points determines the possible similarity of the mean between two sets. In a different perspective, this test checks for the magnitude of the difference between two data sets. In the given equation,  $\bar{X}_1$  and  $\bar{X}_2$  are the means of two data sets.  $S^2$  is the variance of the data

set, which is a square of the standard deviation. The variance is divided by the  $N$ , which is the number of data samples in the set, and the difference between two data sets is given as  $t$ . A value of less than 4.5 for  $t$  indicates that there is no discernible information leakage. A value greater than 4.5 means that discernible information leakage is present; however, extraction of the key is practically not feasible. A value of  $t$  greater than 80 indicates that a sufficient amount of leakage information is available and practical extraction of the key is possible.

### 5.2.2 Hardware Level Countermeasure

As the name suggests, hardware level countermeasures are implemented alongside the hardware implementation of the cryptographic algorithm. Previously mentioned software countermeasures such as hiding, and masking techniques are implemented at hardware or at CMOS cell level. The objective of the countermeasure is unchanged. The hiding technique aims to achieve balanced power consumption to eliminate the correlation between power consumption and the processed data at any time instance. Overall, the power consumption is tied to the maximum value in the individual clock cycle. Many of the hardware countermeasures are based on the concept of dual-rail precharge (DRP) logic design methods [64] [65]. DRP logic design consists of duplicating the original logic cell design by matching gate cells and wiring. In the duplicate design scheme, each structure will be an inversion of the other. Figure 5.1 is an example of the SR latch and its DRP logic cell.

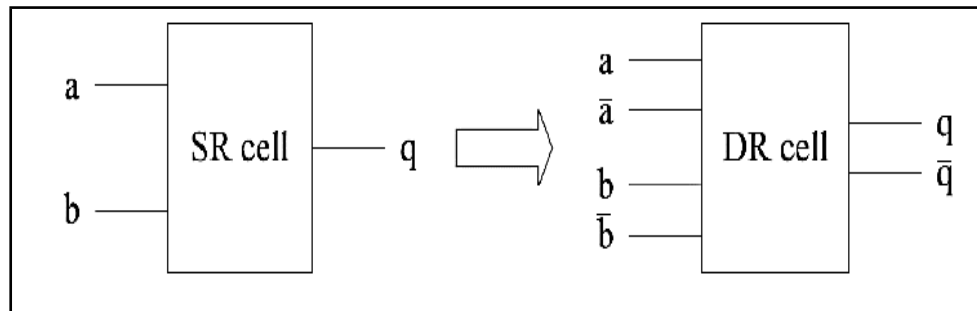


Figure 5.1: 2-input Single Rail Logic Cell and a 5-input Dual Rail Logic Cell.

Each input and output of the single-rail cell structure is paired with its inverted dual-rail input-outputs to create a DRP logic cell. DRP logic cell has two phases of operation. First is the precharge phase, and the second is the evaluation phase. Each phase is executed in a single clock cycle. In the precharge phase, all the input signals are set to a single value, such as either a 1 or 0. In the next evaluation phase, one of the inverted signals changes its value from 1 to 0 or 0 to 1 while the other retains the precharge value. Such duplication of the design ensures that only signal transition occurs in one clock cycle. Another important design consideration for the DRP logic cell design is having a balanced self and load capacitance values. To make sure that the power consumption of the DRP cell is constant, load capacitance values for the inverted signals should be the same.

Existing standard VLSI circuit design tools are not capable of working with the DRP design style. Hence, the designed circuit is synthesized using regular single cell design and later layout of the design is updated by changing the standard gate cells with their corresponding DRP logic cells. There are custom DRP standard cell libraries available, for example, SecLab (Secured Library) [66]. In the design stages of the CMOS circuit, an extra stage is included for logic style conversion. This stage uses defined rules for converting standard cell logic into a DRP logic either by using a standard cell library or custom cell libraries. Load capacitance balancing is carried out in the placement and routing stages of the design process. Figure 5.2 shows the design flow of the standard cell circuit design with additional stages for DRP design style.

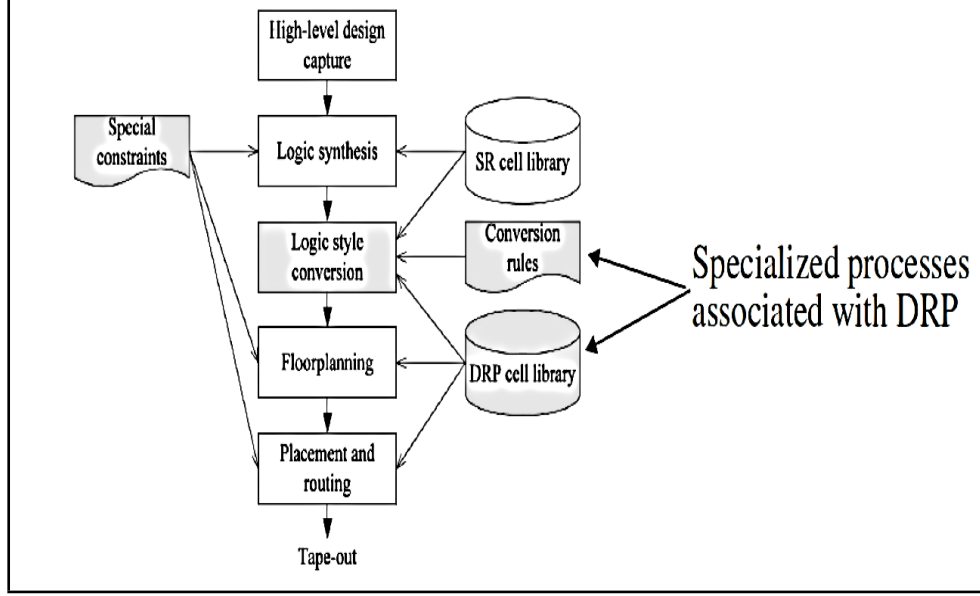


Figure 5.2: Semiconductor Circuit Design Flow for DRP cell Logic Design.

Designing circuits with perfectly balanced load capacitance is particularly difficult. Since the routing of wires is at the micrometer scale, the spacing between wires creates coupling capacitances. Wiring is also affected by the process variations, which adds certain imperfections in the design; for example, over the length of wire, its width will not be exactly the same. To avoid such problems, a differential routing method is used where wires of inverted signals are routed parallel to each other. These different logic design style method use the DRP logic cell structure. Examples of some DRP logic styles, which are used for implementing the hiding technique, are Sense Amplifier Based Logic (SABL) [67], and Wave Dynamic Differential Logic [68].

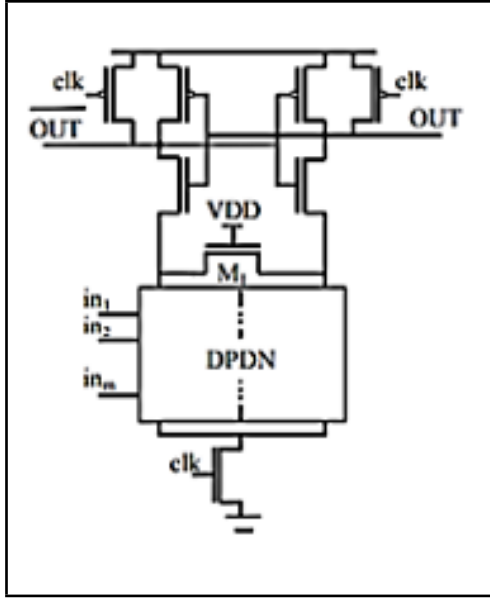
### 5.2.3 Sense Amplifier Based Logic (SABL)

SABL logic cell design makes the time of evaluation phase data independent. The evaluation phase is completed only after all the inputs are settled. All the SABL cells are connected to a clock signal for simultaneous precharge operation. To achieve constant power consumption, load capacitance should charge and discharge with a constant rate in every clock cycle. This is achieved in SABL design where

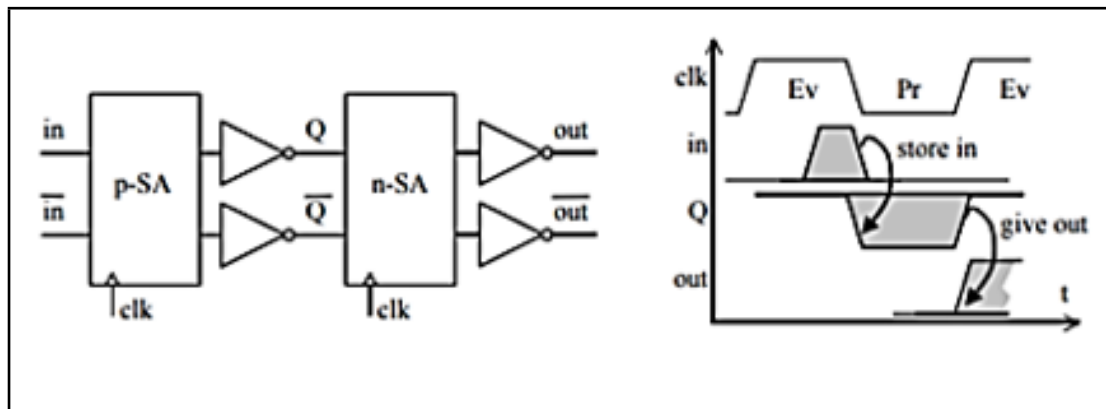


output transition is independent of the changes in signals at input. SABL design also considers load capacitance balancing by setting the constant load capacitance value as a combination of all internal nodes and one of the balanced output loads.

SABL logic cells are combinations of dynamic and differential logic design [67]. This combination charges the load capacitance for all input-output transitions i.e., 0-0, 0-1, 1-1, 1-0. Differential logic design discharges the load capacitance at a constant rate. Therefore, all the output transitions appear the same. However, 1-0 or 0-1 transitions consumes power and 0-0, or 1-1 transitions does not consume power, which makes the difference of power consumption between these two pair discernible. Dynamic logic solves this problem by making power consumption constant at the charging time of load capacitance and keeping it independent of the input switching pattern. In dynamic logic, only 0-1 and 1-1 transition consumes power. Figure 5.3 (a) shows a generic SABL gate. All the subthreshold currents are passed through transistor M1. All the internal nodes of the cell are connected to M1 and the nodes discharge their capacitance together through M1. Differential input-output signals are connected to each other, and all signals paths have the same resistance, and all signals travel the same distance. Precharge phase starts at a low clock cycle, and all the node capacitances are charged, and in the next clock cycle, all of the same capacitances are discharged. This behavior of balanced charging and discharging of capacitances make the overall power consumption balanced and independent of input signal changes.



(a) SABL Generic n-Gate Structure



(b) SABL Flip-Flop Operation

Figure 5.3: SABL Logic Cell Operation.

Figure 5.3 (b) is the representation of the SABL flip-flop implementation. This flip-flop operates in master-slave mode. At the clock falling edge, p-SA is evaluated, and the value is held until next rising clock edge. At the rising clock edge, n-SA is evaluated, and the value is held until next falling edge of the clock cycle. So, values are stored at every clock cycle.

### 5.2.4 Wave Dynamic Differential Logic (WDDL)

To design SABL logic cells, designers use a special gate library containing predesigned DRP gate cells. WDDL [68] [69] cells are designed using single rail logic standard cell designs. The objective of the WDDL design is the same as the SABL design i.e., to balance power consumption. WDDL is simpler to implement than SABL but not as secure or resistant to side channel analysis as SABL. WDDL design occupies less space on a semiconductor fabric as opposed to SABL. Unlike SABL, power consumption and time of evaluation are dependent on the input-output data transitions. WDDL cells do not have clock signals for differential/dynamic operations.

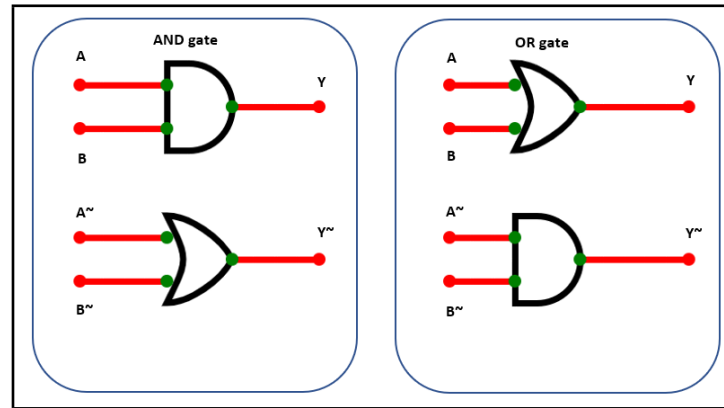


Figure 5.4: WDDL AND-gate (left) and WDDL OR-gate (right).

Logic gates represented in Figure 5.4 are basic WDDL structure gates. There are two pairs of complementary gates; the first pair is AND gate constructed with AND-OR gate pairing, the second pair is OR gate constructed with OR-AND gate pairing. The actual logic input is provided to one of the gates, and inverted inputs or false inputs are provided to the complementary gates. AND gate logic can be proved by using De Morgan's equations for logic gates. In WDDL AND gate, actual inputs A and B are connected to AND gate and false or complementary inputs  $A'$  and  $B'$  are connected to the OR gate. Y is the primary output, and  $Y'$  is the false complementary output.

For AND gate, the logical equation is given as:

$$Y = A \cdot B$$

$$Y = (A \cdot B)'$$

$$Y' = A' + B'$$

$$Y = Y'$$

In the same manner, WDDL OR gate is realized for the OR-AND gate pair. Hence, using single rail standard logic AND and OR gates along with inverter can be used to design any WDDL cell gate design. Many such WDDL gates can be cascaded together to design combinational circuit designs.

In the WDDL design based AND gate's operation, the precharge phase all the input signals are set to 0 or 1. Therefore, in precharge both outputs of WDDL AND gate will be 0. During the evaluation phase, only one of the  $Y$  or  $Y'$  output will transition. In [70], Tiri K. et al. present a prototype IC design with WDDL design for DPA resistance analysis. They designed a WDDL based hardware implementation for AES 128-bit cipher. The aim of their experiment was to check the resistance of the WDDL design against differential power analysis. They called their WDDL implementation a secured implementation and compared this with regular hardware implementation of AES. Their experiment showed that the WDDL design was successful at resisting DPA attack since almost 1.5 million power traces captured for WDDL were not sufficient to disclose the encryption key. This proved the vulnerability of the WDDL design countermeasure against localized EM side channel attacks.

However, WDDL design is not resistant towards DEMA or CEMA attacks. In [71] Sauvage L. et al. performed an EM analysis attack on the secured WDDL implementation of the AES 128-bit cipher. The AES cipher used for the experiment had eight different S-box implementations. Using an EM probe and a moving table, the entire chip surfaced was scanned to find a hot spot for maximum localized EM

emanation point. Using a Hamming Distance power model, a CEMA attack was performed over 100,000 captured EM emanations for last round of AES encryption. Experiment results for CEMA proved that the attack was able to recover encryption key sub-byte with as low as 5000 EM traces.

Countermeasures for the unsecured implementation of various cryptographic algorithms, either at software or hardware level, face various difficulties. Manufacturers who provide cryptographic security module in commercial devices do not wish to spend more on implementing expensive countermeasures. Apart from the pecuniary implementation cost of the countermeasure, cost accrued semiconductor chip size and performance is also an important aspect. Expensive countermeasure implementations can be included for the high-security devices used in military operations or sensitive government operations. Unfortunately, the same is not possible for commercial electronics. Algorithm level masking and hiding techniques are used commonly today in various cryptographic implementation. Such countermeasures are not enough and still vulnerable to side channel analysis attacks. An efficient and reasonable countermeasure is a necessity for two categories of cryptographic devices. First, are the devices which are already developed and in use and second, all the new devices which will be developed with cryptographic security operations.

### 5.3 Proposed Countermeasure

Electromagnetic side channel and power consumption side channel are the most common sources of information leakage in various electronic devices. Countermeasures developed to resist side channel analysis attacks are not an optimum solution. The fundamental reason for the manifestation of information leakage through power or EM side channel is the same i.e., physical aspect of CMOS gate design. Implementation of one countermeasure technique might be efficient against a power analysis attack. However, the same technique might still be vulnerable to EM side channel analysis attacks. One such example is a hardware-level countermeasure, WDDL de-

sign. As presented in [71] WDDL efficiently resisted a differential power analysis attack, but it failed to protect against localized EM side channel analysis attack. A holistic approach is necessary when devising a side channel analysis of resistant countermeasure. It should be able to resist all types of attacks, for example, DPA, DEMA, CPA, CEMA, etc. Software level countermeasures are still breakable, and hardware-level countermeasures are expensive to implement. One advantage of a software countermeasure is its implementation is possible on the existing cryptographic systems.

### 5.3.1 Key Update Scheme Countermeasure

The countermeasure I am proposing in this thesis is a software/algorithm level countermeasure designed for the hardware implementation of the AES 128-bit cipher. The fundamental concept behind this countermeasure is to update the cryptographic key after a defined interval. This technique can be implemented on existing software-based or reconfigurable hardware-based (FPGA, partially reconfigurable ASIC) cryptographic implementations. Any symmetric key cryptographic algorithm can be protected with this technique. The proposed scheme is based on the similar countermeasures proposed in [72] and [73]. The main difference between the previous presented countermeasure scheme and the proposed scheme in this thesis is the generation and storage of the new keys. Fresh keys generated in [72] make use of Physically Unclonable Functions (PUF) implementation to generate new keys on the device. PUF implementation requires extra overhead through cost and additional logic design. The basic idea behind the scheme is described below:

Figure 5.5 depicts the algorithm flow chart for the proposed key update scheme. The first step is to assess the cryptographic hardware for power/EM side channel analysis vulnerability. Performing the side channel attack (CEMA/CPA) on the hardware will provide an estimate of the minimum number of leakage signal traces utilized for successful key extraction. The minimum number of power/EM traces will be referred

to as the Least Needed Traces (LNT). After finalizing the LNT, key must be updated on all the nodes synchronously before any of the key sub-byte can be revealed.

- Obtain the LNT value for Single key (LNTS) used on target hardware.
- Generate a list of random keys on a Trusted Platform Module (TPM) for encryption/decryption and share the keys with the receiver.
- Based on the obtained LNTS value, set the Update Frequency (UF) to be less than the obtained LNTS, and share the same with the receiver.
- Begin the encryption operation with the first key and switch the encryption key with the new key from the key list circularly. Key switching will occur after the counter value reaches the UF value on both the sender and the receiver side.

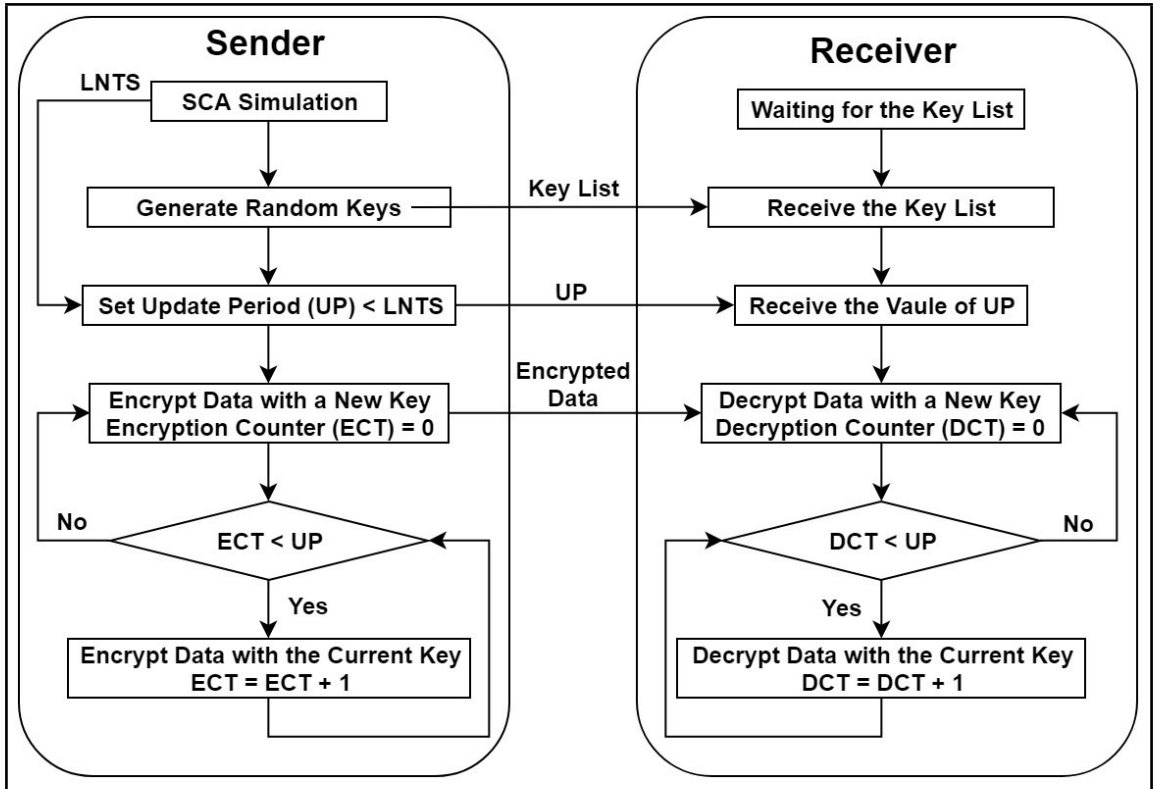


Figure 5.5: Flow of Algorithm for Key Update scheme.

Here, UF corresponds to the total number of encryption operations after which the key is updated. For the side channel analysis attack, each captured EM/power

trace corresponds to the one full encryption operation. LNT value determines the total number of executions of encryption operations before updating the encryption key. Initial side channel analysis simulation provides a threshold value of LNT before which CEMA/CPA attack is particularly ineffective to reveal the secret key. Implementation and strength of the proposed key update scheme are flexible and depends on 1. Determining the length of the key list, 2. Frequency of updating/switching the key.

### 5.3.2 Key Generation on TPM

The TPM chip used in the proposed scheme is Infineon OPTIGATM TPM 2.0 SLB9670 which is encapsulated in Iridium 9670 Evaluation Boards [74]. For its many advantages, a TPM device is used for key generation and key storage purposes. To generate the random key list, a True Random Number Generator (TRNG) is used which is available on the TPM chip. It is able to create a high quality of random bit sequence. TPM integration with the Sakura-X board was achieved by using the MicroBlaze soft processor core which was implemented on the Kintex-7 FPGA chip. Integration between the TPM and the MicroBlaze core is supported with separate software drivers. Figure 5.6 shows the TPM-Sakura-X interface and TPM key generation process. For the proposed scheme, a total of eight random keys are generated inside the TPM, and the average process time is 0.014 seconds.

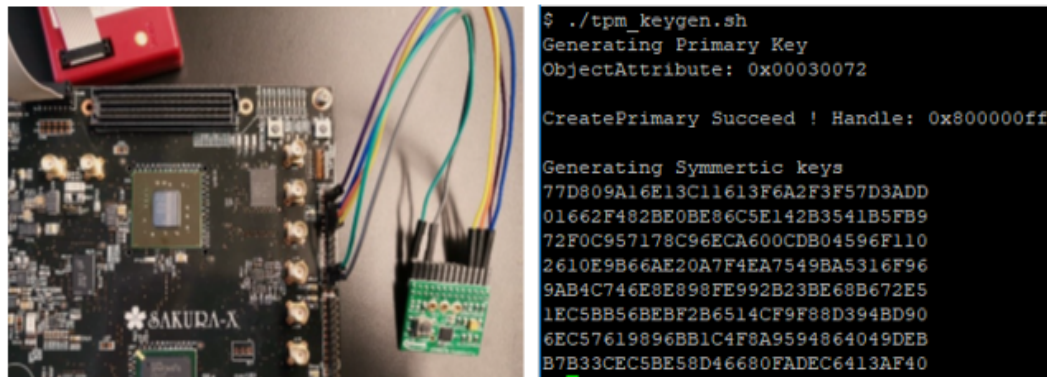


Figure 5.6: TPM - Sakura-X Interface (left) and TPM Key Generation process (right).



Before implementing the countermeasure, the hardware AES implementation on the Kintex-7 FPGA chip was evaluated with side channel analysis to determine LNTS. The result of CPA and CEMA determined the LNTS value at total 3000 traces. A CEMA/CPA attack was performed using 30000 of EM/power traces. The minimum number of LNT for CPA is 3000 traces, and for CEMA it is higher at around 10000 traces. LNT value is used as a UF counter value which is an incremental counter. As soon as the number of encryption operations reach the UF count, the algorithm updates the key. For the new key, the UF counter is again reset to zero, and the count limit is set to the LNTS value. This process continues for all eight encryption keys.

There are two overheads associated with the proposed key update scheme: 1. Area overhead for additional logic level hardware required for changes in algorithm, 2. The time delay associated with key generation and communication. Area overhead also constitutes the storage memory requirement for key storage on TPM. The TPM has integrated non-volatile memory chip of 6962 Bytes [74], which can save a total of 435 AES 128-bit keys. Each 128-bit AES key occupies 16 Bytes of memory space. Another overhead is the integration of MicroBlaze softcore and implementation of communication protocol. TPM can support two communication protocol, 1. Serial Peripheral Interface (SPI), 2. I2C. In the proposed scheme, TPM is integrated using SPI communication protocol. The results of the conducted experiment with the key update scheme are presented in experimental result Chapter 6.

## CHAPTER 6: Experimental Results

This section describes the experimental results for CEMA attack, preprocessing techniques, and the proposed countermeasure. Results of the CEMA attack show the extraction of 128-bit AES cipher key. These results are compared with the results obtained after applying the preprocessing method using PCA. This method is applied for 1. Reduction in data size, 2. Reduction of noise components. Result of the proposed key update countermeasure technique is compared with the result of the CEMA attack.

The CEMA attack is performed using five different data sets. Each dataset contains 30000 rows of plaintext. Each plaintext is of 128 bits arranged as 16-byte hexadecimal values. Each dataset has an individual encryption key. EM emanations are captured for each dataset by performing 30000 encryption operations to collect 30000 EM traces corresponding to each row of plaintext. Following is the list of five keys associated with five plaintext datasets:

- Dataset 1 key 1: 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C
- Dataset 2 key 2: 1D 22 BF 01 AC 77 D9 21 EA 34 15 F5 36 89 10 A2
- Dataset 3 key 3: F0 1E D2 3C B4 5A 96 78 09 AF 81 EB 27 CD 1F A9
- Dataset 4 key 4: 97 45 C3 73 1D AD 77 B1 17 B5 76 F4 5B 4C 1E E0
- Dataset 5 key 5: 5B C3 F3 DD 34 E9 7D 15 F6 FF BE 13 FB FA F6 02

## 6.1 Simple Electromagnetic Analytic (SEMA)

Figure 6.1 shows the captured EM emanations at runtime from Kintex-7 FPGA implementation of AES encryption. Observation of the waveform following the SEMA technique reveals the initial proof of the existence of the EM emanations resulting from the FPGA. The blue square wave signal is a trigger signal that marks the start of the AES encryption process. The encryption operation begins after the rising edge of the square wave signal. The yellow signal with periodic pulses is the captured EM emanation signal for the whole run of the AES-128 encryption operation. The AES-128 process constitutes a total of ten rounds within one encryption operation. Ten rounds of encryption are visible in the figure along with the initial round presenting the XOR operation between the encryption key and the input plaintext.

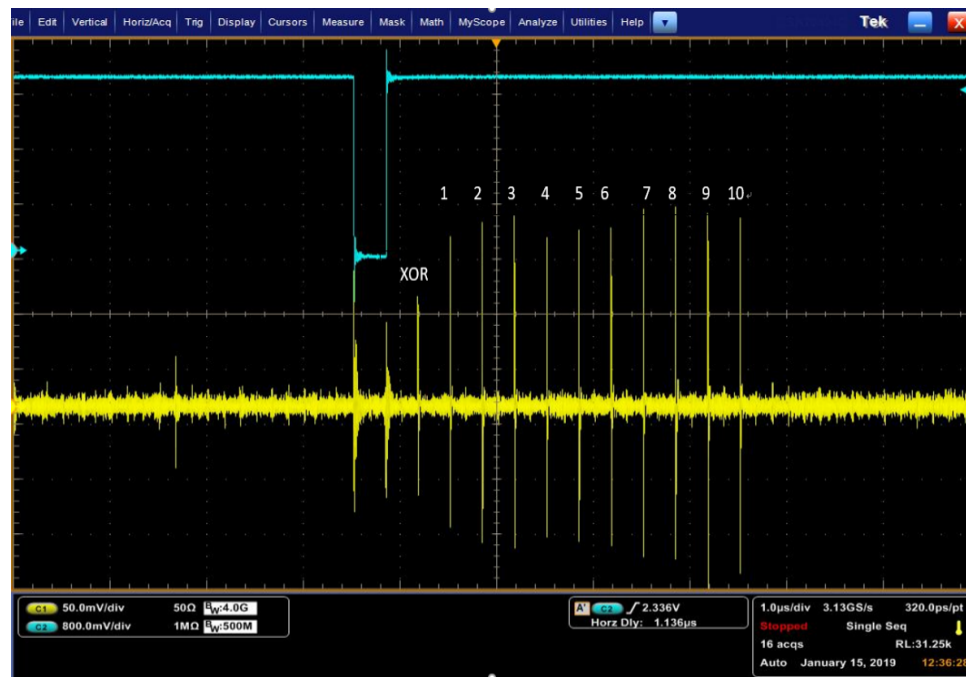


Figure 6.1: Waveform of the EM Emanations for the AES Encryption Showing all Ten Rounds of Operation.

In the presented CEMA attack, the first round of the AES-128 encryption is the target round. The output of the S-box operation generated from the first encryption round is selected as the intermediate value for the Hamming Weight power model. Figure 6.2 shows the zoomed EM emanation waveform to focus on the first three rounds of encryption.

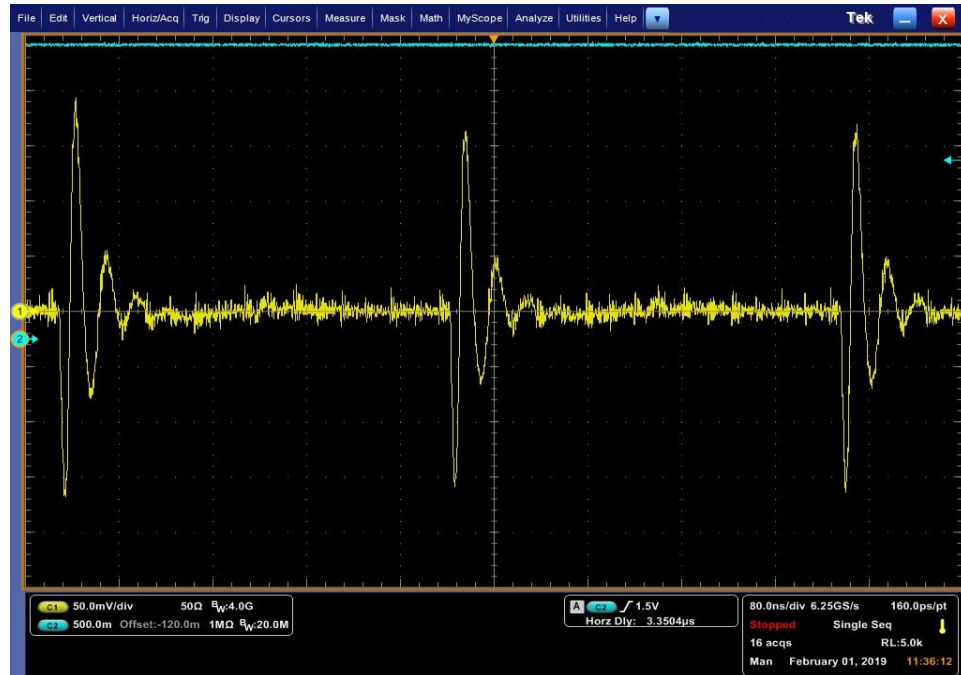


Figure 6.2: Magnified Version of the EM Emanation Waveform Focusing on the First Three Rounds of Encryption.

Figure 6.3 shows an example of the captured EM traces for the four different plaintext values from dataset 1. Figure 6.4 shows the superimposed waveform of four signals for the same four plaintexts. It is visible from the observation that the peak voltage of the EM emanation is different for all four traces. Therefore, proving the theory that the EM emanations are dependent on the data bits processed within the FPGA.



Figure 6.3: EM Emanations of the Different Four Encryption Operations for Four Distinct Plaintext.

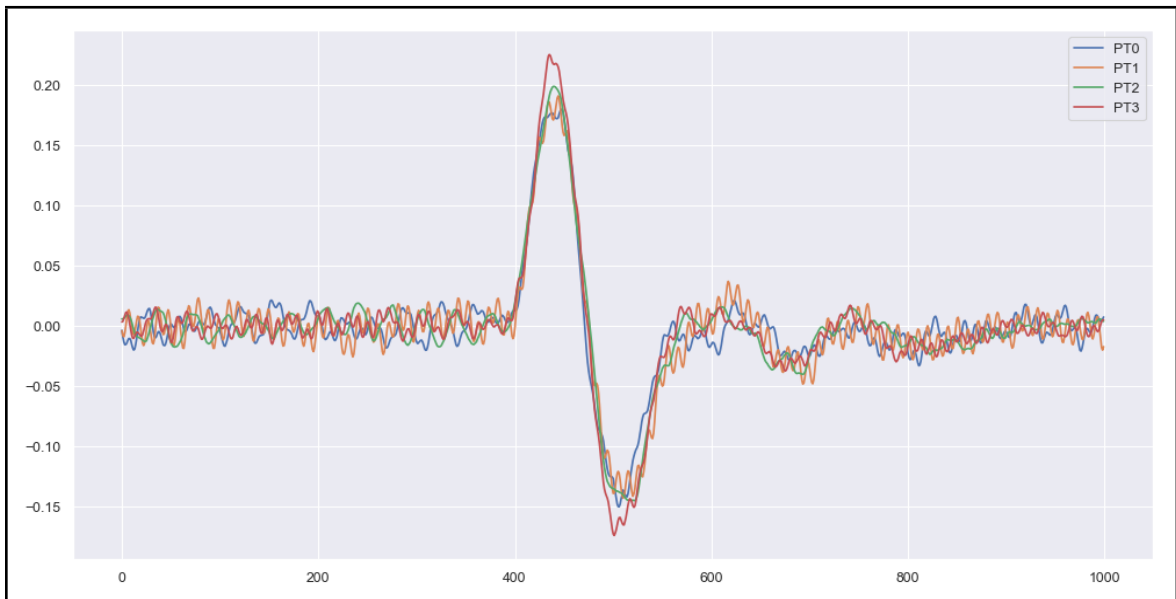


Figure 6.4: Superimposed Waveform for Four Encryption Operations for Four Distinct Plaintexts.

## 6.2 Correlation Electromagnetic Analysis (CEMA)

The CEMA attack was carried out on all five datasets. The EM emanation waveform is adjusted to focus on the first AES round using horizontal positioning and the time scale settings on the oscilloscope. The sampling rate is adjusted at 12.5 GS/s (Gigasamples per seconds), capturing 1250 sample points in each captured EM trace file. All the captured EM trace files are saved in text format, and later all 30000 trace files were consolidated in one file saved in the Python NumPy format for easier processing. Since CEMA requires only the voltage readings, sample points from x-axis time scale readings are not necessary and can be removed. The text files containing plaintext are also converted to Python NumPy format.

The positioning of the EM probe on the FPGA as well as the input voltage supply for the low noise amplifier affects the process of capturing the traces. The amplifier supply voltage can be a minimum of eight volts and a maximum of twelve volts. High voltage increases the sensitivity of the EM probe resulting in higher voltage amplitude signals. Position of the EM is kept fixed, and it is placed 1 centimeter away from the FPGA chip. Increasing the distance of the measurement will result in noisy readings due to the noise signals emanating from the nearby electronics in the lab environment.

CEMA analysis is conducted between collected EM traces and the hypothesized EM power model generated based on the Hamming Distance of 8-bits of the S-box output in the first AES round. Figure 6.5 is the block diagram of the CEMA analysis process.

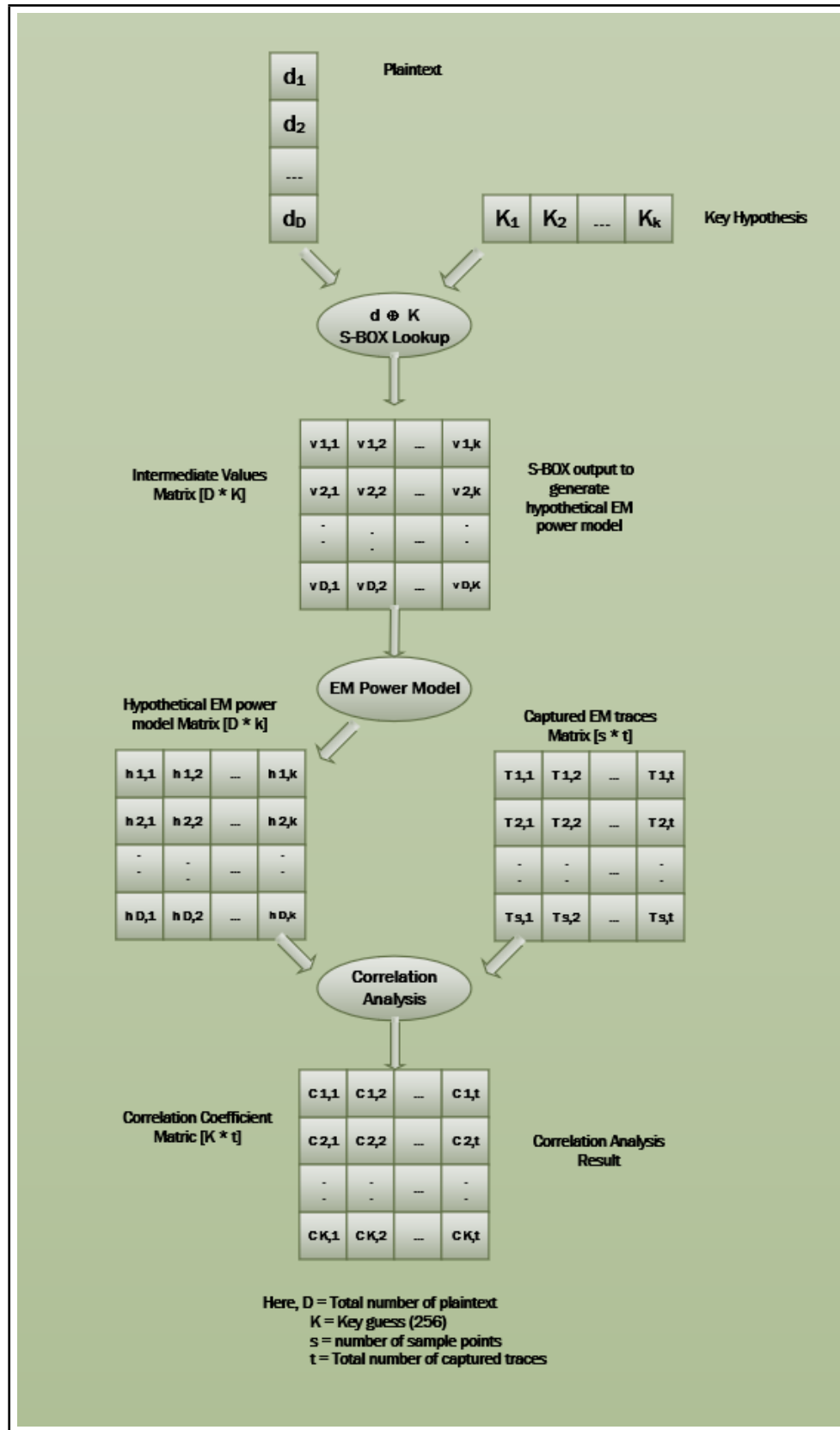
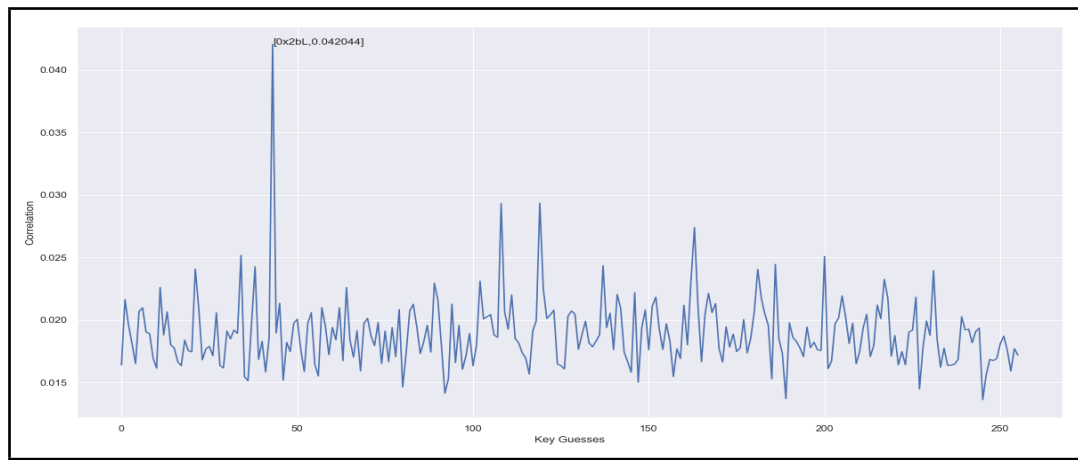


Figure 6.5: Block Diagram of CEMA Analysis Process to Calculate Correlation Coefficient.

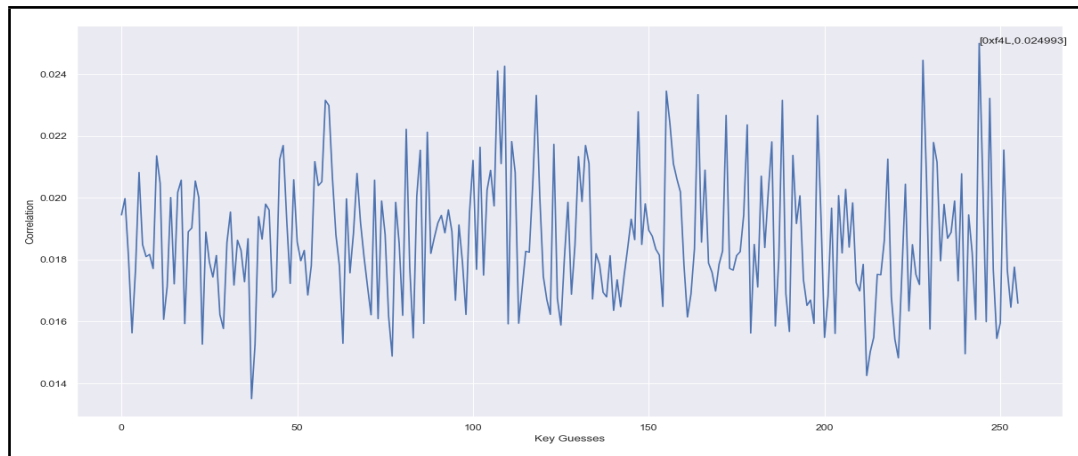
### 6.2.0.1 Results for CEMA Attack on Dataset 1

Encryption Key 1: 2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09 CF 4F 3C.

The CEMA attack script extracts one key byte at a time. The key guess byte, which has the highest correlation coefficient value, is determined as the possible correct key. Figures 6.6 (a) and (b) shows a graph of the correlation coefficient value against 256 key guess values for the correct key guess for key byte 0x2B and incorrect key guess 0xF4 for fourth key byte 0x16.



(a) High Correlation Coefficient Value for the Correct Keyguess 0x2B



(b) High Correlation Coefficient Value for Wrong Keyguess 0xF4 instead of 0x16

Figure 6.6: Correlation Coefficient VS Possible 256 Keyguess.



Figure 6.7 shows the correlation of all 16 bytes from the AES key. From the first CEMA attack, correlation analysis successfully revealed 15 key bytes. The CEMA attack is using all 30000 EM traces for correlation analysis; however, some keys do not require all 30000 traces to yield successful key guess. Figure 6.8 is a graph of correlation coefficient measurement against a total number of traces for correctly guessed first (2Bh) and third (15h) key byte.

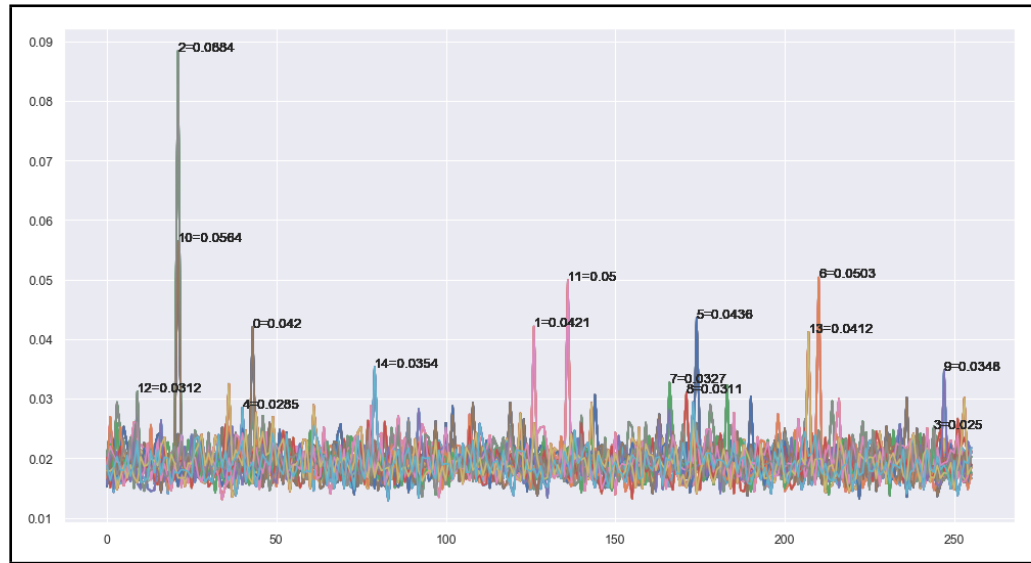
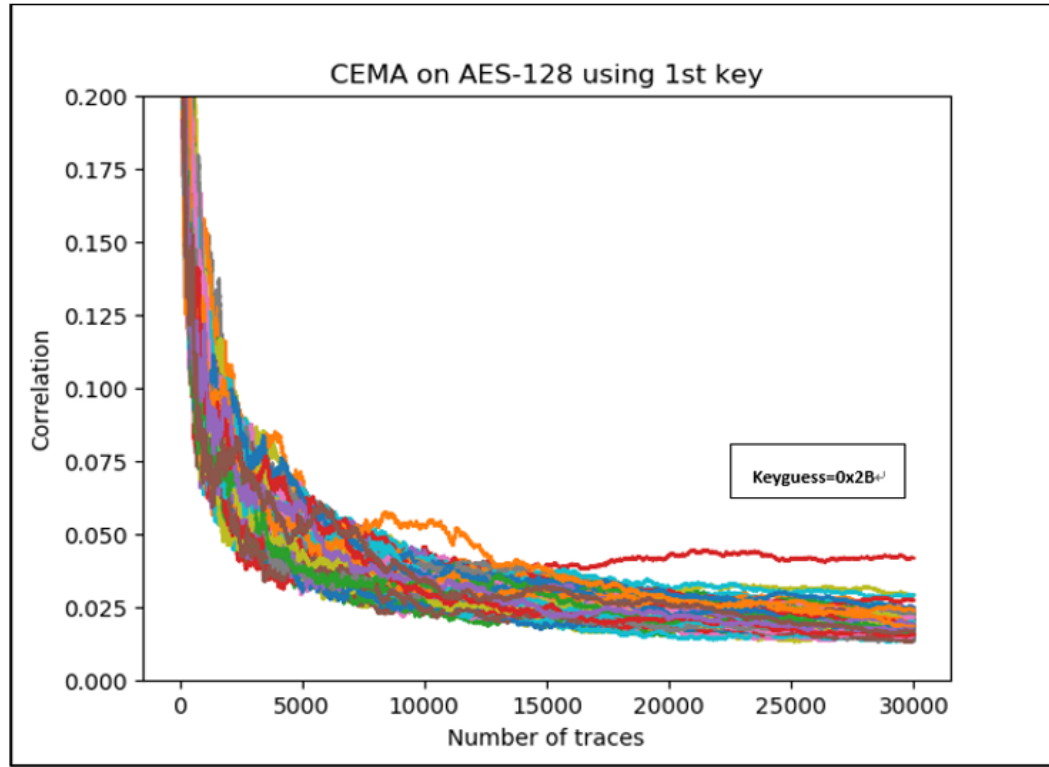
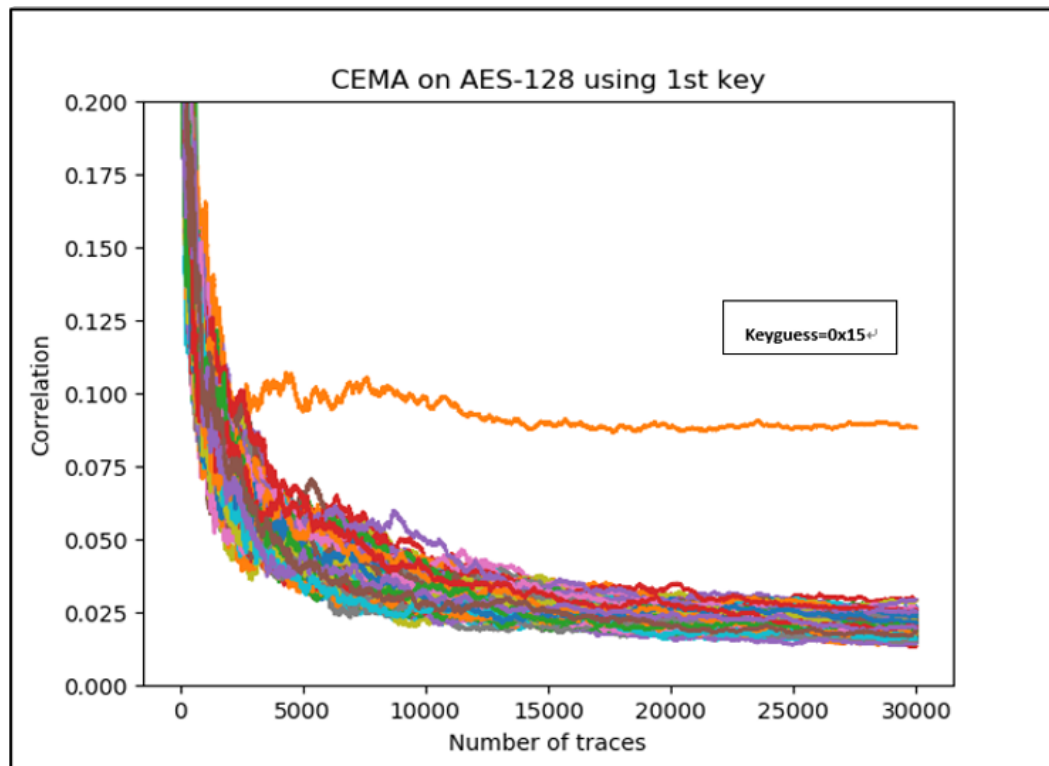


Figure 6.7: Maximum Correlation Coefficient Graph for all Sixteen Encryption Key Bytes.

In Figure 6.8 (a), it is evident that the correlation coefficient value for the first correct key byte starts to increase after 15000 number of EM traces. In figure 6.8 (b) correlation coefficient value for the second correct key guess starts to increase after almost 3000 EM traces. However, for the incorrect key guess, the correlation coefficient value does not increase as the number of EM trace data is not sufficient to yield the correct guess. Figure 6.9 shows the correlation coefficient graph for incorrect key guess (0xF4) for the fourth key byte (0x16).



(a) Correlation Coefficient Starts Increasing after 15000 EM Traces for the Correct Keyguess.



(b) Correlation Coefficient Starts Increasing after 3000 EM Traces for the Correct Keyguess.

Figure 6.8: Graph of Correlation Coefficient VS Number of EM Traces.

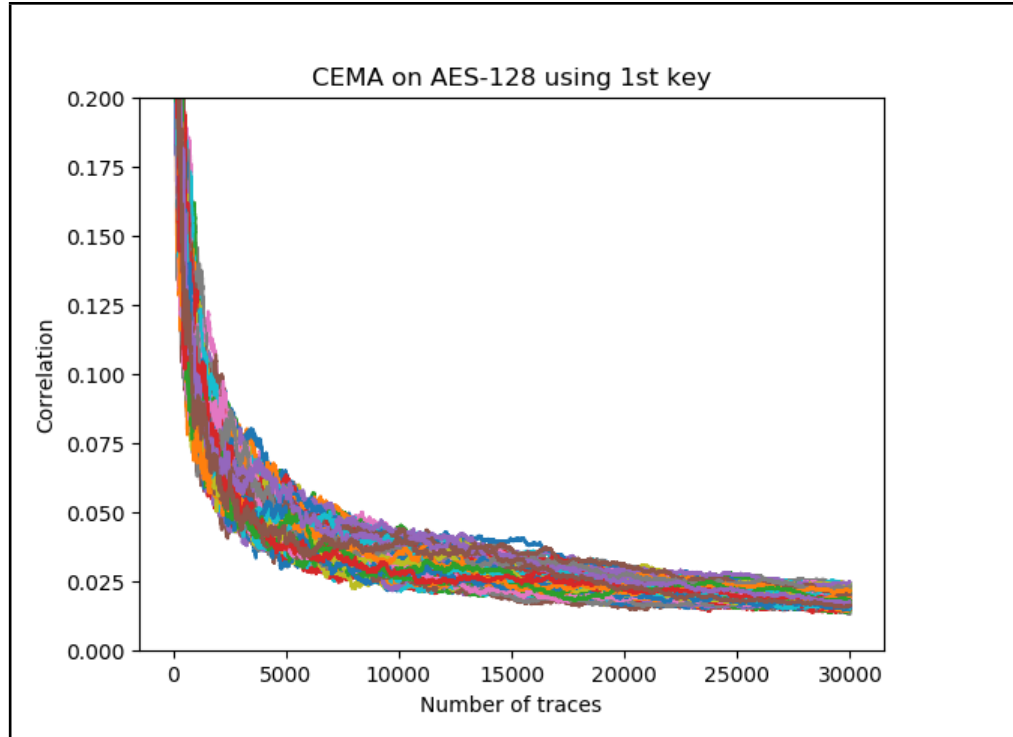
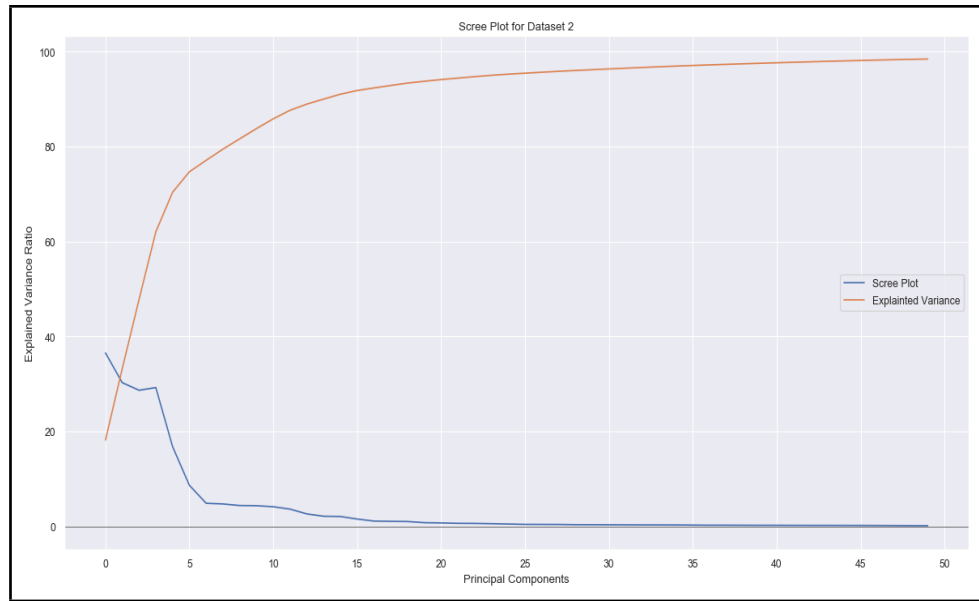


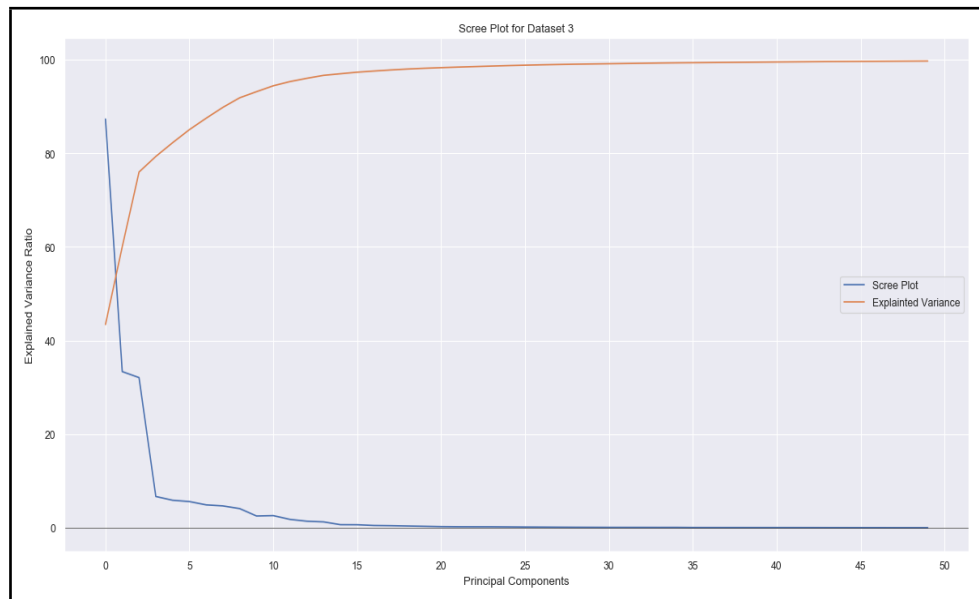
Figure 6.9: Graph for Correlation Coefficient VS Number of Traces for Wrong Keyguess.

### 6.3 Principal Component Analysis Transformation

The PCA technique is used for dimension reduction and noise elimination. From all the captured EM traces, each trace contains up to 250 to 1000 sample points. However, the relevant information is available in only a few of those sample points. PCA technique reduces the number of sample points from traces while retaining the relevant information. The original EM traces are PCA transformed using only the necessary numbers of Principal Components (PC). Necessary PC are obtained by plotting the eigenvalue vector in descending order along with the plot cumulative variance ratio. This plot of PC and the cumulative variance is called a 'Scree Plot.' Figures 6.10 (a)(b)(c)(d) shows the Scree plot for dataset 2, 3, 4 and 5.

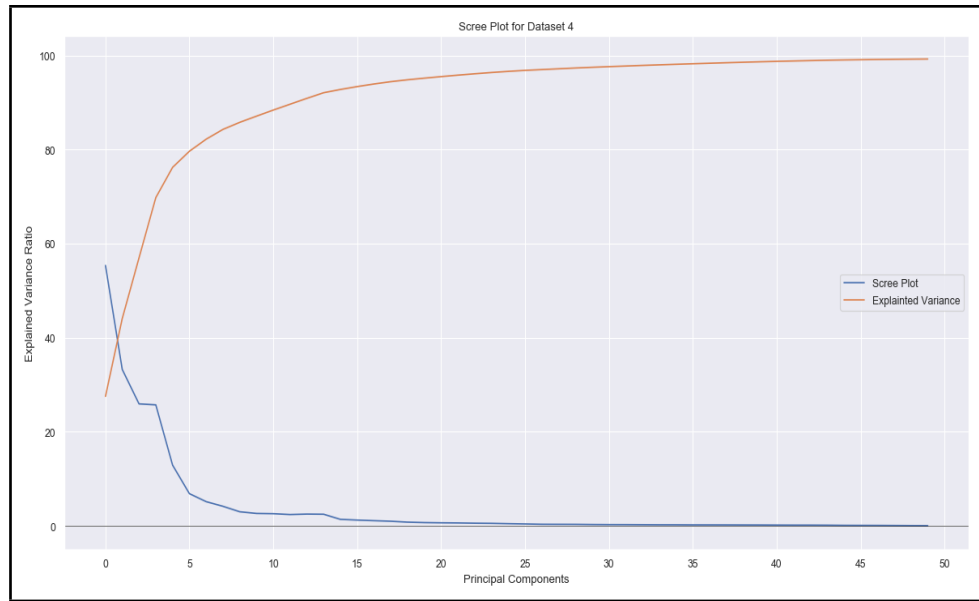


(a) 50 Usable Principal Components for Dataset 2 Contain 99 Percentage of Information

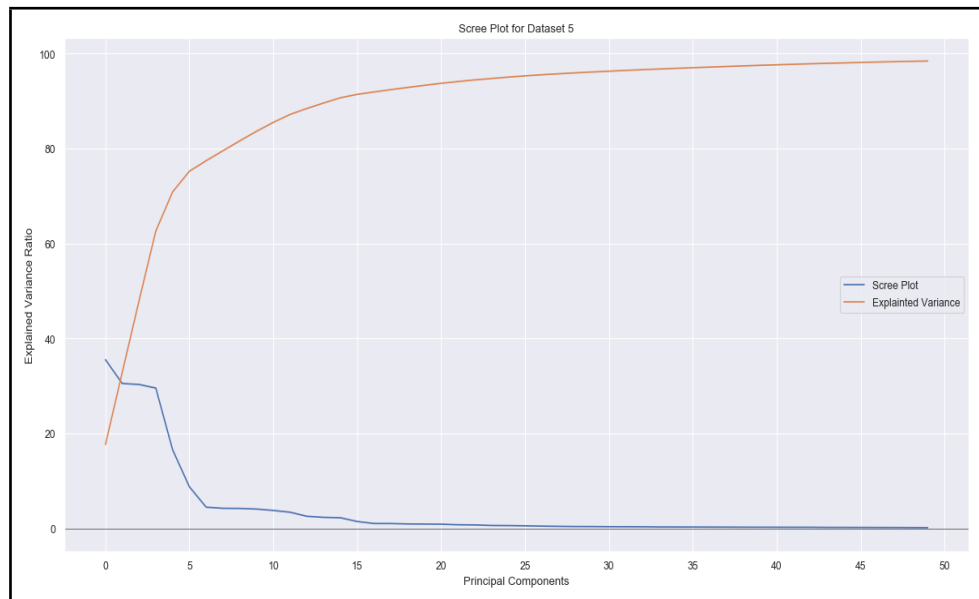


(b) 25 Usable Principal Components for Dataset 3 Contain 99 Percentage of Information

Figure 6.10: Scree Plots (a)(b)(c)(d) Showing Usable Principal Components and Explained Variance Ratio for Datasets 2,3,4 and 5 Respectively.



(c) 30 Usable Principal Components for Dataset 4 Contain 98 Percentage of Information



(d) 35 Usable Principal Components for Dataset 5 Contain 97 Percentage of Information

Figure 6.10: Scree Plots (a)(b)(c)(d) Showing Usable Principal Components and Explained Variance Ratio for Datasets 2,3,4 and 5 Respectively.

The Scree plot shows information about how many PC can be utilized for the PCA transformation, and the explained variance shows how many percentages of relevant information is contained in the selected PC. A PC having a value less than 0

should be discarded as they do not contain the relevant information. From Scree plot for dataset 3, the number of usable PC is almost up to 20 PC. These 20 PC contain up to 95% to 98% of the total relevant information as seen from the curve for the cumulative variance. After determining 20 PC for dataset 3, the original EM traces are PCA transformed by using 20 PC, and the resulting PCA transformed traces contain only 20 columns of sample points in each trace. The total amount of time taken to extract the first key byte from dataset 3 using 30000 traces is 63 seconds. It takes 50 seconds to extract the same key byte using PCA transformed EM traces. The total difference between the calculation time is 13 seconds. This time difference in the analysis time may not seem significant, however, it is important to note that the EM traces are captured in a controlled environment and using a high-performance computer system. In a practical scenario, an attacker will need to capture a significantly large number of EM traces to yield a more successful result. Nowadays, cryptographic system use simple countermeasure techniques such as masking or hiding. To attack such a system using the CEMA method requires a few hundred thousand to a million number EM traces. Using the PCA analysis method can improve the time of attack significantly by reducing the size and dimensions of the data.

PCA can also improve the quality of the EM traces by reducing the noise component at the same time, keeping the original data dimension. Figure 6.11 below shows the maximum correlation graph for a possible key guess for dataset 3 key byte 14. The correct key guess is 0x1F. However, CEMA yields the wrong key guess as 0xFC. After applying PCA transformation to reduce noise elements from the EM traces, new traces yield the correct key guess. In Figure 6.11, the correlation coefficient value for the wrong key guess is high at 0.026. After the PCA transformation, the correct result shows an increase in the correlation coefficient value of the correct key guess at 0.029.

The process of PCA transforming the EM traces for noise reduction is the

same as the PCA transformation except for the dimension reduction. The inverse PCA transformation retains the original data dimension while transforming the EM traces. The selected PC used for data transformation contain only the most relevant information.

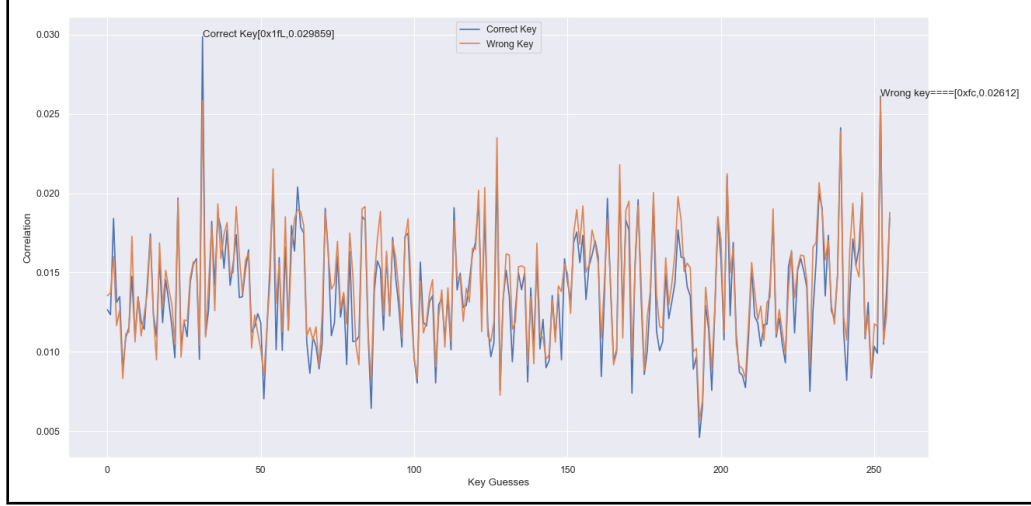


Figure 6.11: Maximum Correlation Coefficient Graph for Correct and Wrong Keyguess.

#### 6.4 Proposed Key Update Countermeasure Scheme

A proposed key update scheme is applied to mitigate the risk of the EM and power side channel attack. Based on the results the CEMA attack on the AES-128 encryption, the lowest value for LNT was determined around 5000 traces, and maximum LNT around 7000 traces. The measurement environment to capture the EM traces can randomly change. For example, the noise in the environment can be reduce by turning off other electronic devices, which can yield better quality of the EM traces. Hence, the value of the update frequency counter was set at 3000 to offset any random deviation. Results for the key update scheme are shown in the figure below. The CEMA attack on dataset 4 shows the high correlation for correct key guesses at the mark of 5000 EM traces and above. Figure 6.13 shows the correlation graph for four correct key bytes from dataset 4. Figure 6.12 shows the result for the

same key bytes after applying the proposed key update scheme. After applying the proposed key update scheme, none of the key bytes is revealed even after using 30000 EM traces. The key update scheme has disturbed the leakage model completely by frequently changing the encryption key.

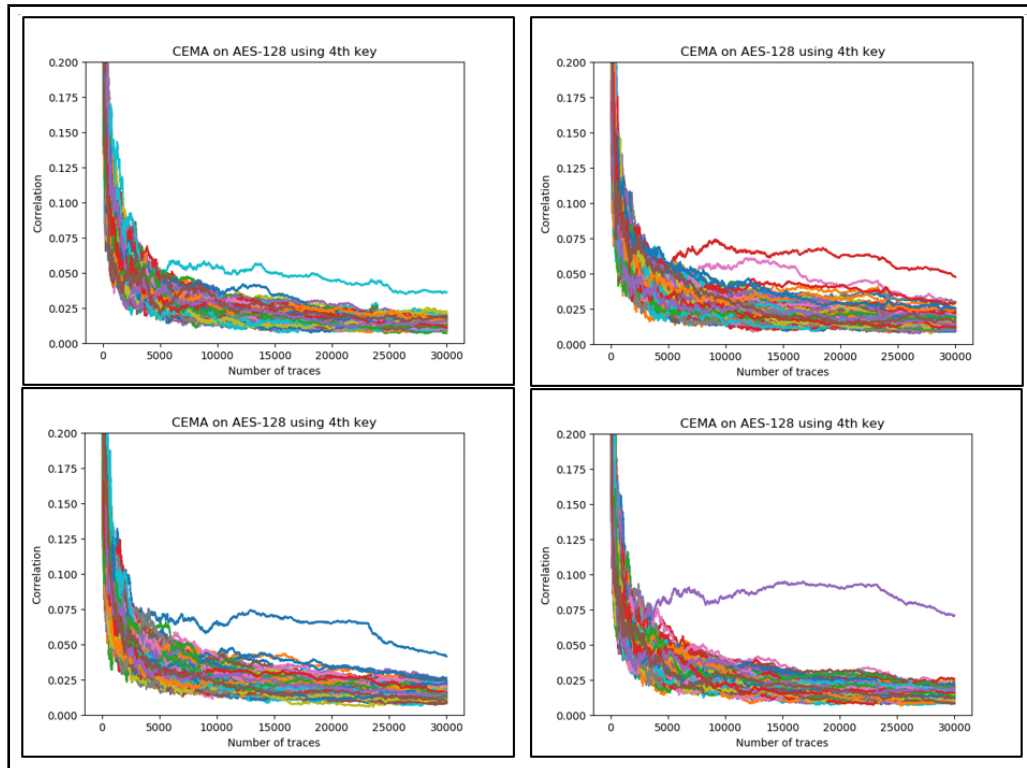


Figure 6.12: Correlation VS Number of Traces for Dataset 4 for Correct Keyguess



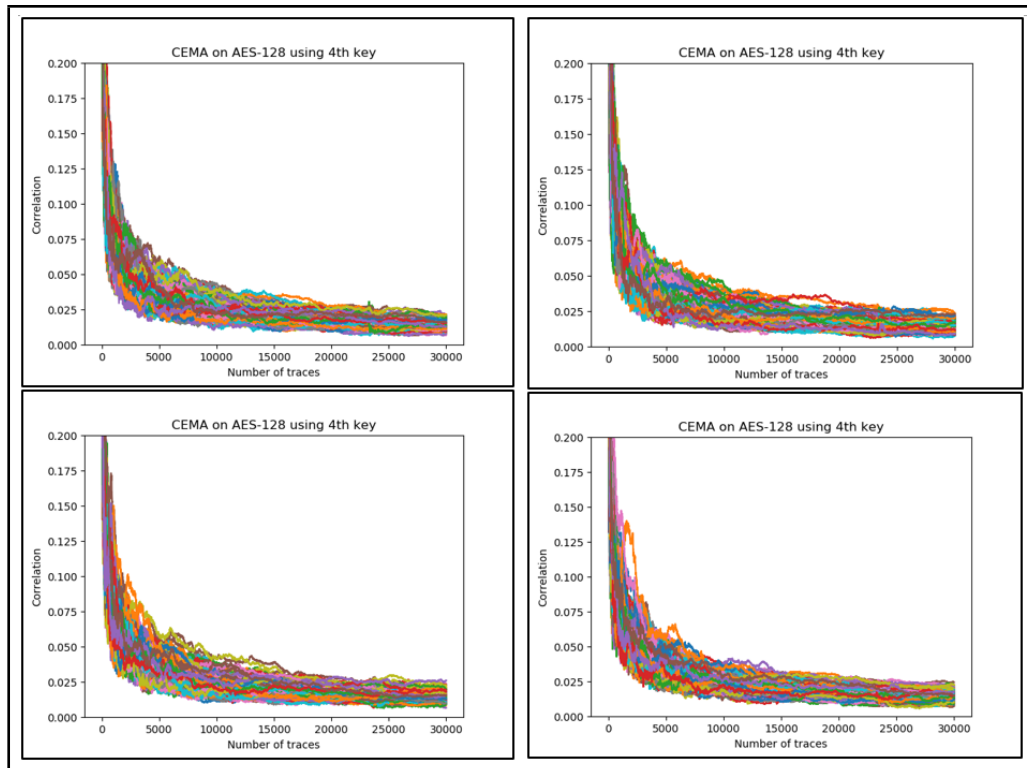


Figure 6.13: Correlation VS Number of Traces for Dataset 4 for Wrong Keyguess.

## CHAPTER 7: Conclusion and Future Research

### 7.1 Conclusion

In this thesis, the FPGA based hardware implementation of the AES-128 encryption was successfully attacked using the CEMA side channel attack. Existence of the EM emanation was proved through SEMA method. The CEMA attack was performed on five different sets of plaintext and encryption key combination. Each CEMA result successfully extracted the encryption key bytes with a maximum of two wrong key guesses. Brute force attack on the AES-128 encryption requires to make  $2^{128} = 340,282,366,920,938,000,000,000,000,000,000,000,000,000$  combinations of key. To extract the correct key from such a huge combination will take many lifetimes. However, the CEMA attack can reduce this number of key combinations significantly. For the 16 byte AES key with two wrong guesses of the key bytes from the CEMA attack can reduce this number to only  $2^{16} = 65536$  different key combinations. This number of the key combination is only  $2.9387358770557187699218413430556 \times 10^{-37}\%$  of the total number of key combinations required to extract all 16 bytes of AES key, which is significantly far less.

The Principal Component Analysis technique is an efficient preprocessing technique to enhance the performance of the CEMA attack. The PCA technique serves two primary aims; 1. Reduction in data size while retaining the relevant information, 2. Noise reduction. These two techniques were applied to captured EM traces, and the CEMA was performed on the PCA transformed traces. The PCA transformation significantly reduces the time to finish the CEMA attack. The CEMA experiment used 30000 EM traces; however, in a real life scenario, an attacker might have to capture a large quantities of EM traces for a successful key extraction. The PCA

transformation can reduce the size of the collected data and increase the efficiency of the attack. The PCA noise reduction technique can help to reduce the noise components from the EM traces to improve the attack efficiency even further. Non-extracted key bytes from the CEMA attack can be extracted using a noise reduction technique, which increases the risk of a brute force attack even further.

There are a few other advantages of using PCA transformation. PCA can work efficiently with misaligned EM traces. There are some countermeasures which add a random amount of delay in some operations, for example, delays in calculating S-box output will change the time instances of the leakage information at peak amplitudes. Since PCA only focus on the variance within all the traces and does not consider the time instances of the sample point, the effect of randomness due to time variation is totally neglected in PCA transformation. With its advantages, PCA also has one major drawback. PCA can not handle calculations performed on traces containing a large number of sample points. Since the PCA algorithm has to calculate the covariance matrix of the dimension  $[n \times n]$  where  $n$  is equal to the number of sample points in the trace file. A dataset with such large number of sample points will reduce the PCA transformation time, which will add up to the total time of the CEMA attack.

To mitigate the risk of a CEMA side channel attack, a proposed key update scheme is applied. After the assessment of the CEMA attack, the minimum number of EM traces, required to successfully extract a key byte, was determined at 5000 traces. The frequency of updating the key was set at 3000 runs of the AES. The proposed scheme successfully defeated the EM power model of the CEMA. The proposed scheme is flexible and can accommodate individual devices based on their assessment of the CEMA resistance. TPM integration provides secure key storage as well as key generation environment. In conclusion, the proposed countermeasure is best suited for cryptographic operations where encryption keys are updated continuously. This

countermeasure will fail under the possibility when the same list of the encryption key is reused, and an attacker has physical access of the device to perform CEMA using chosen plaintext. Encryption of two similar plaintexts with one key generates the same ciphertext result. By using the same plaintext to perform the encryption operation multiple times, the attacker can find the value of an update frequency counter.

## 7.2 Future Research

The side channel attack technique and the countermeasure described in this thesis are only a small part of a bigger picture. This CEMA experiment was performed in a controlled environment with the best equipment available to us. The success of the side channel attack is highly dependent on the availability of the necessary equipment. An attacker with abundant access to high-performance measurement equipment will pose a considerable risk to the cryptographic systems. In this thesis, the target of the CEMA attack was the FPGA based hardware implementation of the AES-128 bit encryption algorithm. In the future, we would explore the side channel attacks and countermeasures on asymmetric key cryptographic algorithms such as Elliptic-curve cryptography and the RSA cryptographic algorithm. The most time consuming process of performing the CEMA attack is the amount of time spent while capturing the EM traces. It will be easier and less time consuming for the research study to have a way to perform the CEMA attack as the traces are collected. We want to devise a technique which will make this attack scenario possible.

## REFERENCES

- [1] J. Daemen and V. Rijmen, *The Design of Rijndael*. Berlin, Heidelberg: Springer-Verlag, 2002.
- [2] M. Lipp, M. Schwarz, D. Gruss, T. Prescher, W. Haas, A. Fogh, J. Horn, S. Mangard, P. Kocher, D. Genkin, Y. Yarom, and M. Hamburg, “Meltdown: Reading kernel memory from user space,” in *Proceedings of the 27th USENIX Conference on Security Symposium*, SEC’18, (Berkeley, CA, USA), pp. 973–990, USENIX Association, 2018.
- [3] G. Maisuradze and C. Rossow, “Ret2spec: Speculative execution using return stack buffers,” in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, CCS ’18, (New York, NY, USA), pp. 2109–2122, ACM, 2018.
- [4] E. F. Foundation, *Cracking DES: Secrets of Encryption Research, Wiretap Politics and Chip Design*. Sebastopol, CA, USA: O’Reilly & Associates, Inc., 1998.
- [5] B. Jones, “European emc directive 2004/108/ec harmonised standards,” in *2008 IEEE International Symposium on Electromagnetic Compatibility*, pp. 1–4, Aug 2008.
- [6] W. Trappe and L. C. Washington, *Introduction to Cryptography with Coding Theory (2Nd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2005.
- [7] NIST, “Fips 197,” November 2001.
- [8] NIST, “Nist special publication 800-38a. online.”
- [9] C. Cid and R.-P. Weinmann, *Block Ciphers: Algebraic Cryptanalysis and Gröbner Bases*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009.
- [10] M. Matsui, “Linear cryptanalysis method for des cipher,” in *Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology*, EUROCRYPT ’93, (Berlin, Heidelberg), pp. 386–397, Springer-Verlag, 1994.
- [11] E. Biham and A. Shamir, “Differential cryptanalysis of des-like cryptosystems,” in *Proceedings of the 10th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO ’90, (Berlin, Heidelberg), pp. 2–21, Springer-Verlag, 1991.
- [12] M. Wang, Y. Sun, N. Mouha, and B. Preneel, “Algebraic techniques in differential cryptanalysis revisited,” in *ACISP*, 2011.
- [13] C. Cid, “Some algebraic aspects of the advanced encryption standard,” in *Advanced Encryption Standard – AES* (V. Dobbertin, Hansand Rijmen and A. Sowa, eds.), (Berlin, Heidelberg), pp. 58–66, Springer Berlin Heidelberg, 2005.

- [14] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology CRYPTO 99* (M. Wiener, ed.), (Berlin, Heidelberg), pp. 388–397, Springer Berlin Heidelberg, 1999.
- [15] D. Agrawal, B. Archambeault, J. R. Rao, and P. Rohatgi, "The em side channel(s) : Attacks and assessment methodologies," 2003.
- [16] P. C. Kocher, "Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems," in *Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology, CRYPTO '96*, (London, UK, UK), pp. 104–113, Springer-Verlag, 1996.
- [17] S. Mangard, E. Oswald, and T. Popp, *Power Analysis Attacks: Revealing the Secrets of Smart Cards (Advances in Information Security)*. Berlin, Heidelberg: Springer-Verlag, 2007.
- [18] D. Agrawal, B. Archambeault, J. R. Rao, and P. Rohatgi, "The em side-channel(s)," vol. 2523, pp. 29–45, 08 2002.
- [19] H. Ott, *Electromagnetic Compatibility Engineering*. Wiley Publishing, 1st ed., 2009.
- [20] K. Gandolfi, C. Mourtel, and F. Olivier, "Electromagnetic analysis: Concrete results," in *Proceedings of the Third International Workshop on Cryptographic Hardware and Embedded Systems, CHES '01*, (London, UK, UK), pp. 251–261, Springer-Verlag, 2001.
- [21] J.-S. Coron, P. C. Kocher, and D. Naccache, "Statistics and secret leakage," in *Financial Cryptography*, 2000.
- [22] T. S. Messerges, E. A. Dabbish, and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks," *IEEE Trans. Comput.*, vol. 51, pp. 541–552, May 2002.
- [23] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *CHES*, 2004.
- [24] R. Mayer-Sommer, "Smartly analyzing the simplicity and the power of simple power analysis on smartcards," in *CHES*, 2000.
- [25] D. J. Bernstein, "Cache-timing attacks on aes," tech. rep., 2005.
- [26] J.-J. Quisquater and D. Samyde, "Electromagnetic analysis (ema): Measures and counter-measures for smart cards," in *Proceedings of the International Conference on Research in Smart Cards: Smart Card Programming and Security, E-SMART '01*, (London, UK, UK), pp. 200–210, Springer-Verlag, 2001.

- [27] C. H. Gebotys, S. Ho, and C. C. Tiu, “Em analysis of rijndael and ecc on a wireless java-based pda,” in *Proceedings of the 7th International Conference on Cryptographic Hardware and Embedded Systems*, CHES’05, (Berlin, Heidelberg), pp. 250–264, Springer-Verlag, 2005.
- [28] Y. Hori, T. Katashita, A. Sasaki, and A. Satoh, “Electromagnetic side-channel attack against 28-nm fpga device,” 2012.
- [29] L. Sauvage, S. Guilley, and Y. Mathieu, “Electromagnetic radiations of fpgas: High spatial resolution cartography and attack on a cryptographic module,” *TRETS*, vol. 2, pp. 4:1–4:24, 2009.
- [30] L. Sauvage, S. Guilley, F. Flament, J. Danger, and Y. Mathieu, “Cross-correlation cartography,” in *2010 International Conference on Reconfigurable Computing and FPGAs*, pp. 268–273, Dec 2010.
- [31] D. Real, F. Valette, and M. Drissi, “Enhancing correlation electromagnetic attack using planar near-field cartography,” in *2009 Design, Automation Test in Europe Conference Exhibition*, pp. 628–633, April 2009.
- [32] Y. Souissi, M. Nassar, S. Guilley, J.-L. Danger, and F. Flament, “First principal components analysis: A new side channel distinguisher,” in *ICISC*, 2010.
- [33] J. L. Rodgers and W. A. Nicewander, “Thirteen ways to look at the correlation coefficient,” *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.
- [34] K. Pearson, *On Lines and Planes of Closest Fit to Systems of Points in Space*. University College, 1901.
- [35] H. Abdi and L. J. Williams, “Principal component analysis,” *WIREs Comput. Stat.*, vol. 2, pp. 433–459, July 2010.
- [36] I. Jolliffe, *Principal Component Analysis*. Springer Verlag, 1986.
- [37] L. I. Smith, “A tutorial on principal components analysis,” tech. rep., Cornell University, USA, February 26 2002.
- [38] D. Agrawal, J. R. Rao, and P. Rohatgi, “Multi-channel attacks,” vol. 2779, pp. 2–16, 09 2003.
- [39] S. E. Sarma, S. A. Weis, D. W. Engels, M. Gagné, D. Agrawal, B. Archambeault, S. Chari, J. R. Rao, and P. Rohatgi, “Advances in side-channel cryptanalysis , electromagnetic analysis and template attacks,” 2003.
- [40] S. Mangard, “Exploiting radiated emissions - em attacks on cryptographic ics,” in *Proceedings of Austrochip 2003*, pp. 13–16, 2003.

- [41] E. De Mulder, P. Buysschaert, S. B. Ors, P. Delmotte, B. Preneel, G. Vandenbosch, and I. Verbauwhede, "Electromagnetic analysis attack on an fpga implementation of an elliptic curve cryptosystem," in *EUROCON 2005 - The International Conference on Computer as a Tool*, vol. 2, pp. 1879–1882, Nov 2005.
- [42] V. Carlier, H. Chabanne, E. Dottax, and H. Pelletier, "Generalizing square attack using side-channels of an aes implementation on an fpga," in *International Conference on Field Programmable Logic and Applications, 2005.*, pp. 433–437, Aug 2005.
- [43] Huiyun Li, A. T. Markettos, and S. Moore, "Security evaluation against electromagnetic analysis at design time," in *Tenth IEEE International High-Level Design Validation and Test Workshop, 2005.*, pp. 211–218, Nov 2005.
- [44] M. G. Kuhn, "Security limits for compromising emanations," in *Proceedings of the 7th International Conference on Cryptographic Hardware and Embedded Systems, CHES'05*, (Berlin, Heidelberg), pp. 265–279, Springer-Verlag, 2005.
- [45] N. Homma, S. Nagashima, T. Sugawara, T. Aoki, and A. Satoh, "A high-resolution phase-based waveform matching and its application to side-channel attacks," *IE-ICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E91-A, pp. 193–202, Jan. 2008.
- [46] Y. Oren and A. Shamir, "Remote password extraction from rfid tags," *IEEE Trans. Comput.*, vol. 56, pp. 1292–1296, Sept. 2007.
- [47] E. Peeters, F.-X. Standaert, and J.-J. Quisquater, "Power and electromagnetic analysis: Improved model, consequences and comparisons," *Integr. VLSI J.*, vol. 40, pp. 52–60, Jan. 2007.
- [48] C. H. Gebotys and B. A. White, "A phase substitution technique for dema of embedded cryptographic systems," *Fourth International Conference on Information Technology (ITNG'07)*, pp. 868–869, 2007.
- [49] L. Sauvage, S. Guilley, J.-L. Danger, Y. Mathieu, and M. Nassar, "Successful attack on an fpga-based wddl des cryptoprocessor without place and route constraints," in *Proceedings of the Conference on Design, Automation and Test in Europe, DATE '09*, (3001 Leuven, Belgium, Belgium), pp. 640–645, European Design and Automation Association, 2009.
- [50] V. Lomné, P. Maurine, L. Torres, M. Robert, R. Soares, and N. L. V. Calazans, "Evaluation on fpga of triple rail logic robustness against dpa and dema," *2009 Design, Automation Test in Europe Conference Exhibition*, pp. 634–639, 2009.
- [51] M. Vuagnoux and S. Pasini, "Compromising electromagnetic emanations of wired and wireless keyboards," in *Proceedings of the 18th Conference on USENIX Security Symposium, SSYM'09*, (Berkeley, CA, USA), pp. 1–16, USENIX Association, 2009.



- [52] O. Meynard, S. Guilley, J.-L. Danger, and L. Sauvage, "Far correlation-based ema with a precharacterized leakage model," *2010 Design, Automation Test in Europe Conference Exhibition (DATE 2010)*, pp. 977–980, 2010.
- [53] Y. Hayashi, N. Homma, T. Mizuki, T. Aoki, H. S. Sone, L. Sauvage, and J. Danger, "Analysis of electromagnetic information leakage from cryptographic devices with different physical structures," *IEEE Transactions on Electromagnetic Compatibility*, vol. 55, pp. 571–580, 2013.
- [54] D. Merli, J. Heyszl, B. Heinz, D. Schuster, F. Stumpf, and G. Sigl, "Localized electromagnetic analysis of ro pufs," *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pp. 19–24, 2013.
- [55] A. G. Zajić and M. Prvulovic, "Experimental demonstration of electromagnetic information leakage from modern processor-memory systems," *IEEE Transactions on Electromagnetic Compatibility*, vol. 56, pp. 885–893, 2014.
- [56] R. L. Callan, N. B. Popovic, A. A. Daruna, E. Pollmann, A. G. Zajić, and M. Prvulovic, "Comparison of electromagnetic side-channel energy available to the attacker from different computer systems," *2015 IEEE International Symposium on Electromagnetic Compatibility (EMC)*, pp. 219–223, 2015.
- [57] M. Yoshikawa and Y. Nozaki, "Electromagnetic analysis method for ultra low power cipher midori," in *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pp. 70–75, Oct 2017.
- [58] Y. Kim, T. Sugawara, N. Homma, T. Aoki, A. Satoh, A. Aoba, and A. ku Sendai-shi, "Biasing power traces to improve correlation in power analysis attacks," 2010.
- [59] S. Mangard, "Why the masking of cmos gates does not prevent dpa attacks?," in *CT-RSA 05*.
- [60] F. Standaert, E. Peeters, and J. Quisquater, "On the masking countermeasure and higher-order power analysis attacks," in *International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume II*, vol. 1, pp. 562–567 Vol. 1, April 2005.
- [61] J. Cooper and E. F. J. Demulder, "Test vector leakage assessment ( tvla ) methodology in practice ( extended abstract )," 2013.
- [62] T. Schneider and A. Moradi, "Leakage assessment methodology - a clear roadmap for side-channel evaluations." Cryptology ePrint Archive, Report 2015/207, 2015. <https://eprint.iacr.org/2015/207>.
- [63] G. Goodwill, B. Jun, J. Jaffe, and P. Rohatgi, "P.: A testing methodology for side-channel resistance validation, niat," 2011.

- [64] J. Danger, S. Guilley, S. Bhasin, and M. Nassar, "Overview of dual rail with precharge logic styles to thwart implementation-level attacks on hardware cryptoprocessors," in *2009 3rd International Conference on Signals, Circuits and Systems (SCS)*, pp. 1–8, Nov 2009.
- [65] A. Moradi, M. T. M. Shalmani, and M. Salmasizadeh, "Dual-rail transition logic: A logic style for counteracting power analysis attacks," *Comput. Electr. Eng.*, vol. 35, pp. 359–369, Mar. 2009.
- [66] S. Guilley, F. Flament, Y. Mathieu, and R. Pacalet, "Security evaluation of a balanced quasi-delay insensitive library (seclib)," 2008.
- [67] K. Tiri, M. Akmal, and I. Verbauwhede, "A dynamic and differential cmos logic with signal independent power consumption to withstand differential power analysis on smart cards," in *Proceedings of the 28th European Solid-State Circuits Conference*, pp. 403–406, Sep. 2002.
- [68] K. Tiri and I. Verbauwhede, "A logic level design methodology for a secure dpa resistant asic or fpga implementation," in *Proceedings Design, Automation and Test in Europe Conference and Exhibition*, vol. 1, pp. 246–251 Vol.1, Feb 2004.
- [69] I. Verbauwhede, K. Tiri, D. Hwang, and P. Schaumont, "Circuits and design techniques for secure ics resistant to side-channel attacks," in *2006 IEEE International Conference on IC Design and Technology*, pp. 1–4, May 2006.
- [70] K. Tiri, D. Hwang, A. Hodjat, B. cheng Lai, S. Yang, P. Schaumont, and I. Verbauwhede, "Prototype ic with wddl and differential routing - dpa resistance assessment," in *Cryptographic Hardware and Embedded Systems à CHES 2005, 7th International Workshop*, pp. 354–365, Springer.
- [71] L. Sauvage, S. Guilley, J. Danger, Y. Mathieu, and M. Nassar, "Successful attack on an fpga-based wddl des cryptoprocessor without place and route constraints," in *2009 Design, Automation Test in Europe Conference Exhibition*, pp. 640–645, April 2009.
- [72] X. Xi, A. Aysu, and M. Orshansky, "Fresh re-keying with strong pufs: A new approach to side-channel security," in *2018 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pp. 118–125, April 2018.
- [73] M. Medwed, F.-X. Standaert, J. Großschdl, and F. Regazzoni, "Fresh re-keying: Security against side-channel and fault attacks for low-cost devices," in *AFRICACRYPT*, 2010.
- [74] Infineon, "Optiga tpm slb 9670 tpm2.0 data sheet. available: [https://www.infineon.com/dgdl/infineon-data-sheet-slb9670\\_2.0\\_rev1.3\\_-\\_ds\\_-\\_v01\\_03\\_-\\_en.pdf?fileid=5546d462689a790c016929ed3b5e4ffb](https://www.infineon.com/dgdl/infineon-data-sheet-slb9670_2.0_rev1.3_-_ds_-_v01_03_-_en.pdf?fileid=5546d462689a790c016929ed3b5e4ffb)."