RELATING UNDISTURBED BITS TO OTHER PROPERTIES OF SUBSTITUTION BOXES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF APPLIED MATHEMATICS
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

RUSYDI HASAN MAKARIM

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
CRYPTOGRAPHY

JULY 2014

Approval of the thesis:

RELATING UNDISTURBED BITS TO OTHER PROPERTIES OF SUBSTITUTION BOXES

submitted by **RUSYDI HASAN MAKARIM** in partial fulfillment of the requirements for the degree of **Master of Science in Department of Cryptography, Middle East Technical University** by,

Prof. Dr. Bülent Karasözen
Director, Graduate School of **Applied Mathematics** ________________

Prof. Dr. Ferruh Özbudak
Head of Department, **Cryptography** ________________

Assoc. Prof. Dr. Ali Doğanaksoy
Supervisor, **Department of Mathematics, METU** ________________

Examining Committee Members:

Prof. Dr. Ferruh Özbudak
Department of Mathematics, METU ________________

Assoc. Prof. Dr. Ali Doğanaksoy
Department of Mathematics, METU ________________

Dr. Çağdaş Çalık
Department of Cryptography, METU ________________

Dr. Cihangir Tezcan
Department of Cryptography, METU ________________

Asst. Prof. Dr. Fatih Sulak
Department of Mathematics, Atılım University ________________

Date: ________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: RUSYDI HASAN MAKARIM

Signature :

ABSTRACT

RELATING UNDISTURBED BITS TO OTHER PROPERTIES OF
SUBSTITUTION BOXES

Makarim, Rusydi Hasan
M.S., Department of Cryptography
Supervisor : Assoc. Prof. Dr. Ali Doğanaksoy

July 2014, 37 pages

Recently it was observed that for a particular nonzero input difference to an S-Box, some bits in all the corresponding output differences may remain invariant. This specific invariant bit is called *undisturbed bit*. Undisturbed bit can also be seen as a truncated differential with probability 1 for an S-Box. The existence of undisturbed bits was found in the S-Box of PRESENT and its inverse. A 13-round improbable differential attack on PRESENT was provided by Tezcan (2013) and without using the undisturbed bits in the S-Box an attack of this type can only reach 7 rounds. Although the observation and the cryptanalytic application of undisturbed bits are given, its relation with other properties of an S-Box remain unknown. This thesis presents some results on mathematical properties of S-Boxes having undisturbed bits. We show that an S-Box has undisturbed bits if any of its coordinate function has a nonzero linear structure. The relation of undisturbed bits with other cryptanalytic tools such as difference distribution table (DDT) and linear approximation table (LAT) are also given. We show that autocorrelation table is proven to be a more useful tool, compared to DDT, to obtain all nonzero input differences that yield undisturbed bits. Autocorrelation table can then be viewed as a counterpart of DDT for truncated differential cryptanalysis. Given an $n \times m$ balanced S-Box, we state that the S-Box has undisturbed bit whenever the degree of any of its coordinate function is quadratic.

Keywords : block cipher, substitution box, undisturbed bit, truncated differential

ÖZ

RAHATSIZ EDİLMEMİŞ BİTLERİN DEĞİŞİM-KUTULARININ DİĞER ÖZELLİKLERİ İLE İLİŞKİSİ

Makarim, Rusydi Hasan
Yüksek Lisans, Kriptografi Bölümü
Tez Yöneticisi : Doç. Dr. Ali Doğanaksoy

Temmuz 2014, 37 sayfa

Son araştırmalarda, bir değişim-kutusu (s-kutusu)'nun sıfırdan farklı girdi farkı için, karşılık gelen çıktı farkındaki bazı bitlerin değişmeyebileceği gözlemlendi. Bu değişmeyen özel bitler rahatsız edilmemiş bitler olarak adlandırılır. Aslında bu bitler s-kutuları için bir olasılıklı kesik diferansiyel olarak da görülebilir. PRESENT şifreleme algoritmasında kullanılan s-kutularında ve bunların terslerinde rahatsız edilmemiş bitlerin varlığı gösterilmiştir. Bu algoritmaya 13 döngülük olası olmayan diferansiyel atak Tezcan (2013) tarafından uygulanmıştır. Bu atakta s-kutularının rahatsız edilmemiş bitleri kullanılmadan en fazla 7 raunta kadar çıkılabildi. Rahatsız edilmemiş bitlerin kriptografik uygulamaları verilmesine rağmen s-kutularına ait diğer özelliklerle olan ilişkisi bilinmemektedir. Çalışmamızda, rahatsız edilmemiş bitlere sahip s-kutularının matematiksel özellikleriyle ilgili bazı sonuçları sunuyoruz. S-kutularının herhangi bir bileşeni (Boole fonksiyonu) lineer yapıya sahip ise bu s-kutularının rahatsız edilmemiş bitlere sahip olduğunu gösterdik. Ayrıca, s-kutularının bu bitleri ile fark dağılım tablosu (FDT) ve lineer yaklaşım tablosu (LYT) gibi diğer kriptografik araçların ilişkisi incelenmiş ve verilmiştir. Rahatsız edilmemiş bitleri üreten sıfırdan farklı girdi farklarını elde etmek için, FDT ile kıyaslandığında otokorelasyon tablolarının daha kullanışlı olduğunu gösterdik. Otokorelasyon tabloları, kesik diferansiyel kriptoanaliz için FDT' nin karşılığı olarak görülebilir. Verilen herhangi bir $n \times m$ dengeli s-kutusu için, bu s-kutusunun herhangi bir bileşen fonksiyonunun derecesi kuadratik olduğu zaman bu s-kutusunun rahatsız edilmemiş bitlere sahip olduğunu gösterdik.

Anahtar Kelimeler: blok şifreleme, değilim kutusu, rahatsız edilmemiş bitler, kesik diferansiyel

Untuk Hasan Makarim dan Anisah Bawazier
Abi dan Mamah Juara Satu Seluruh Dunia

ACKNOWLEDGMENTS

During the completion of this thesis I had a wonderful opportunity to work with great number of individuals. I would like to thank Assoc. Prof. Dr. Ali Doğanaksoy for his supports and guidance as my supervisor. I remember I had a trouble finding someone who would agree as a supervisor for this work. But he immediately said ”yes” without hesitation even after a short discussion about the research plan. I would also like to thank Dr. Cihangir Tezcan for being such an excellent mentor. His advices and criticisms, especially on my poor writing style, are very important in improving the content of this thesis. Cihangir has also generously shared his research experiences, which are so much valuable for someone who just started working in cryptography like myself. I would like to thank other jury members for their comments and feedbacks: Prof. Ferruh Özbudak, Dr. Çağdaş Çalık, and Asst. Prof. Dr. Fatih Sulak.

I am very grateful to all my colleagues in the Institute of Applied Mathematics for all the joys, jokes, and friendships for the past two years. Many thanks go to all Indonesian students in METU for making this place feel like home. Lastly, I dedicate this part for Nabiel, Zacky, and my parent. This sentence is surely too short to express how grateful I am for all your endless supports.

TABLE OF CONTENTS

CHAPTERS

LIST OF TABLES

LIST OF SYMBOLS

$\mathbb{Z}$	Set of integers
$\mathbb{N}$	Set of natural numbers
$\lvert V\rvert$	Cardinality of the set V
$\mathbb{F}_2$	Finite field with two elements
$\mathbb{F}_2^n$	n-dimensional vector space over $\mathbb{F}_2$
$\overline{x} = (x_{n-1}, \ldots, x_0)$	Element of $\mathbb{F}_2^n$
$\oplus, \bigoplus$	Addition modulo 2 or bitwise exclusive-or (XOR)
$\overline{x} \cdot \overline{y}$	Inner product of $\overline{x}$ and $\overline{y}$
$\overline{0}, \overline{1}$	All zero-vector and all-one vector in $\mathbb{F}_2^n$
$\overline{e}_i$	The i-th standard basis vector
$\neg\overline{x}$	Complement of x
$\text{wt}(\overline{x}),\ \text{wt}(f)$	Hamming weight of the vector $\overline{x}$ and the Boolean function f
$\text{Supp}(\overline{x}),\ \text{Supp}(f)$	Support of the vector $\overline{x}$ and the Boolean function f
$\mathcal{B}^n$	The set of all n-variable Boolean functions
$\mathcal{A}^n$	The set of all n-variable affine Boolean functions
$\mathcal{L}^n$	The set of all n-variable linear Boolean functions
$\neg f$	Complement of the Boolean function f
$\deg(f)$	Algebraic degree of the Boolean function f
$\text{dt}(f, g)$	Hamming distance of the Boolean functions f and g
$\mathcal{W}_f(\overline{\omega})$	Walsh-Hadamard Transform of the Boolean function f at $\overline{\omega}$
$r_f(\overline{\omega})$	Autocorrelation of the Boolean function f at $\overline{\omega}$
$C_{f,g}(\overline{\omega})$	Cross-correlation of the Boolean functions f and g at $\overline{\omega}$
$\mathfrak{C}_{f,g}$	Correlation of the Boolean functions f and g
$D_{\overline{\omega}}f$	Derivative of f at $\overline{\omega}$
$\mathcal{LS}_f$	The set of all linear structures of f
$\mathcal{LS}(n)$	The set of all n-variable Boolean functions with linear structures
$S : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$	$n \times m$ Substitution box (S-Box) S

CHAPTER 1

INTRODUCTION

1.1 Cryptography and Cryptanalysis

Information transmission and storage are typical problems for any individual or organization, especially when its content should be kept confidential. Centuries ago various ways and tools to protect information's content have been developed, from pen-and-paper methods, mechanical devices, and computational machines. The extensive research of information protection leads to the invention of a new field of study on its own, called *cryptography*.

Although there are numerous definitions, one of the elegant description of cryptography was the one stated by Rivest : it is about communication in the presence of adversaries [27]. However, the study of cryptography is not limited in the context of information confidentiality but also data integrity, entity authentication, and non-repudiation. The fundamental aim is to adequately address these areas in theory as well as practice.

Cryptanalysis, on the other side, is a subject that deals with analyzing cryptographic mechanism in order to recover partial or full information about the original information. However, it is important to realize that cryptanalysis is not only targeting the original text. Besides its purpose, cryptanalysis also improves the development of new cryptographic methods that can withstand against known attack procedures. At the same time, new cryptographic techniques may also lead to the invention of new cryptanalytic approaches in the future. Both cryptography and cryptanalysis essentially influence each other.

The basic idea of a cipher system is to conceal confidential information in such a way that makes it look meaningless to any unauthorized party. The original information that we want to encipher is called *plaintext*. The process of transforming a plaintext into a secret message is known as *encryption*. The encrypted plaintext is called *ciphertext* and the process of recovering a ciphertext into its original plaintext is *decryption*, which is the inverse of encryption. In order to make the plaintext accessible only to authorized parties, there is an additional information supplied into the encryption and decryption algorithm known as *key*. We refer to the key used in encryption and decryption algorithm as *encryption key* and *decryption key*, respectively. With the correct decryption key, the receiver of the ciphertext is able to obtain the original plaintext. Hence, the secrecy of the key guarantees the confidentiality of the message from any unauthorized entity.

One class of cryptographic function where the encryption key and decryption key can be easily derived from each other, thus both must be kept secret and securely distributed, is called *secret-key cryptography* or *symmetric-key cryptography*. On the other hand there exist cryptosystems that are designed in such a way that it is difficult for any adversary to obtain the decryption key from the encryption key, which allow the encryption key to be publicly disclosed. Such mechanism is called *public-key cryptography* or *asymmetric-key cryptography*.

Two main types of symmetric-key cryptography are *block ciphers* and *stream ciphers*. As the name suggests, the encryption/decryption algorithm of a block cipher is performed on a block of plaintext/ciphertext, typically with size 64-bit or 128-bit. Stream ciphers operate with a time-varying transformation, usually on a single-bit of plaintext digit. There are also some variant of stream ciphers that operate on a word or a single byte. For more extensive study on stream ciphers, one may refer to the work of Rueppel [28] and Golomb [14].

1.2 Block Ciphers

Let n be the size of plaintext as well as ciphertext block. Let k be the length of the key. The encryption function E and decryption function D of a block cipher is defined as

$$E : \{0,1\}^n \times \{0,1\}^k \mapsto \{0,1\}^n$$
$$D : \{0,1\}^n \times \{0,1\}^k \mapsto \{0,1\}^n$$

where $\{0,1\}^n, \{0,1\}^k$ denote the n-bit string and k-bit string of zeros and ones, respectively . Choosing the value k (length of the key) is the most important consideration when designing a block cipher. The reason is by using one plaintext and its associated ciphertext, any attacker with sufficient computational power can obtain the correct key by exhaustively searching through all elements in the key space $\{0,1\}^k$. This naive attack method is known as *exhaustive key search* or *brute force* attack. The length of the key must be determined in a way that it is computationally infeasible for current processor technology to perform brute force within reasonable time period. However, using the key length as the only criteria for proving the security of a block cipher is certainly insufficient.

The construction of encryption functions in block ciphers come from the idea of *product cipher* proposed by Shannon [30]. He suggested that one may build an encryption function by combining two or more operations such that the resulting cipher (composition of operations) provides better security margin compared to its individual components. He also suggested the notion of *confusion* and *diffusion*, two general design principles for a practical cipher.

Confusion The statistics of ciphertext should depend on the plaintext statistics in such a way that it becomes too complicated to be exploited by the cryptanalyst. This also means that the relationship between the ciphertext and key should be as complex as possible.

Diffusion Each parity bit of a plaintext and a secret key should influence as many bit in the ciphertext as possible. It is intended to spread out bits in the plaintext to obscure any redundancy of plaintext which may appear later in the ciphertext.

The idea of product ciphers proposed by Shannon is applied by employing composition of functions to achieve confusion and diffusion criteria. In practice, a designer of block ciphers implements a nonlinear function to achieve the confusion property and combines it with a linear function to achieve the diffusion criteria.

The encryption algorithm of a block cipher consists of iterative transformations of its internal function called *round*. It is an r-round block cipher if the encryption function repeatedly applies the round function for r times. Each round function consists of a nonlinear function, a linear function, and a key mixing operation. The nonlinear function in a block cipher typically uses substitution operation and it is implemented using a lookup table called *substitution box* or *S-Box*. The linear layer in a round function is usually done by permuting the bits of the message. The key mixing in general uses bitwise XOR on the internal state of the cipher with the *round key*. Some block ciphers also use modular addition or modular multiplication instead of bitwise XOR in their key mixing operation. Each round key is generated from the user-supplied key by a *key-schedule function*.

This design of a block cipher that employs substitution, permutation, and key mixing operation is known as *substitution-permutation network* or *SPN*. One of the prominent example of block ciphers having SPN structure is *Rijndael* [11], which was chosen as Advanced Encryption Standard (AES). Note that when designing an SPN cipher, each function has to be invertible in order to make decryption operation possible.

Another design principle of block ciphers is *Feistel cipher*. The model was originally proposed by Horst Feistel in the Lucifer algorithm [31]. Initially, n-bit plaintext is divided into two equal size blocks, each with length $n/2$-bit, called L_0 and R_0. Suppose the algorithm runs for r number of rounds, then the ciphertext is obtained by repeatedly apply the following

$$\begin{aligned} L_i &= R_{i-1} && \forall i = 1, \ldots, r-1 \\ R_i &= L_{i-1} \oplus F(K_i, R_{i-1}) && \forall i = 1, \ldots, r-1 \\ \\ L_r &= L_{r-1} \oplus F(K_r, R_{r-1}) \\ R_r &= R_{r-1} \end{aligned}$$

The ciphertext C is obtained by $C = L_r \mid\mid R_r$ where $\mid\mid$ denotes bit concatenation. Notice that in the last round, the swapping of left and right block is omitted since it does not give any cryptographic significance. The core of the encryption function lies in the function F where substitution, permutation, and key mixing operation are implemented. The keys K_i used in each round are generated from the key-schedule function.

An advantage of Feistel ciphers over substitution-permutation networks is the function F need not to be invertible to make decryption possible. This means that the implementation of the decryption function will use the same function as encryption, except for the key used in the decryption will be executed in reverse order. One disadvantage of Feistel ciphers is that in each round the confusion and diffusion operation can only be applied on half of the block size. It then requires more rounds to achieve complete diffusion on the whole plaintext.

1.3 Substitution Boxes

Many nonlinear functions in block ciphers use substitution operations, implemented using S-Boxes. The role of S-Boxes in the security of block ciphers plays a very crucial part. Recall that a block cipher as a product cipher is composed of a nonlinear and a linear mapping. The behavior of a linear mapping can be easily predicted due to its linear properties. Nonlinear mapping, on the other hand, is the main source of security for a block cipher. We will see later that some cryptanalysis techniques for block ciphers turn into probabilistic methods due to nonlinear mappings. A "good" S-Box yields a good cipher that resists against various cryptanalytic attacks. However, designing a good S-Box turns out to be not a trivial task. It involves the study of Boolean functions. S-Boxes used in block ciphers must satisfy different criteria and there are tradeoffs among these criteria.

Size of an S-Box is determined by the number of its input and output bits. For an S-Box with n input bits and m output bits, we call it $n \times m$ S-Box. The choice of an S-Box is influenced by the design goal of that particular block cipher. For instance, a block cipher intended for lightweight devices uses small S-Boxes (e.g. 3×3 and 4×4) in order to reduce the cost of memory and power consumption in the implementation. Some general purpose block ciphers may use larger S-Boxes (e.g. Rijndael [11] uses an 8×8 S-Box).

1.4 Attack Models and Cryptanalysis of Block Ciphers

In a block cipher, the most important component that should be kept secure from the adversaries is the key used for encryption/decryption. Although the details of the cipher can also be kept secret, this paradigm of security does not provide long-term security assurance. It is more plausible to assume that adversaries can obtain details of the encryption system at anytime without inconvenience. Hence, the secrecy of information relies totally on the secrecy of the key. This requirement of encipherment design is known as *Kerckhoffs' principle* [15].

The list of some possible attacks against an encryption system in general can be classified into different models as follows.

- **Ciphertext-Only Attack** : The scenario assumes that an attacker has only information about ciphertexts. In this case, the attacker's only ability is listening to the encrypted communications without having any idea about the corresponding plaintext.

- **Known-Plaintext Attack** : The attacker has access to plaintexts and their corresponding ciphertexts. These information are used to recover the key of the encryption system.
- **Chosen-Plaintext Attack** : In this case, an attacker can choose a specific desired plaintext and has access to the corresponding ciphertext after encryption.
- **Chosen-Ciphertext Attack** : This case is similar with chosen-plaintext attack, but instead of having access to the plaintext, an attacker can choose the ciphertexts and obtain their corresponding plaintexts through decryption function.
- **Adaptively Chosen-Plaintext Attack** : It is a type of chosen-plaintext scenario in which an attacker has an ability to select a plaintext as input to encryption function based on the previous chosen-plaintext query.
- **Adaptively Chosen-Ciphertext Attack** : This scenario is the opposite of adaptively chosen-plaintext attack. It belongs to the class of chosen-ciphertext attack, in which an attacker has an ability to choose a specific ciphertext to the decryption function based on the previous chosen-ciphertext query.

If an adversary could recover some partial or full information about the key, then she can also recover the content of the plaintext. Other than brute force technique, an attacker can exploit the statistical properties of the cipher to distinguish its behaviour from a random permutation. Most of the techniques in the cryptanalysis of block ciphers are based on such approach.

The two most important techniques in cryptanalysis of block ciphers are *differential cryptanalysis* [4] and *linear cryptanalysis* [23]. Differential cryptanalysis uses the relation of two different plaintexts and its propagation during the encryption. The aim of differential cryptanalysis is to find a high probability differences in the plaintext and ciphertext, so that it can be used to distinguish the cipher from a random permutation. Linear cryptanalysis employs different strategy compared to differential cryptanalysis by finding a linear relation between parities of bits in plaintexts, ciphertexts, and keys. In other words, linear cryptanalysis tries to obtain a linear expression between plaintext bits, ciphertext bits, and key bits such that the probability that the equation holds is bounded away from $1/2$.

1.5 Differential Cryptanalysis of Block Ciphers

Differential cryptanalysis is a powerful method of cryptanalysis technique for block ciphers which belongs to the type of chosen-plaintext attack. It was first applied against Data Encryption Standard by Biham and Shamir [4].

For a fixed encryption key K, let P denote a plaintext and C be its corresponding ciphertext, i.e. $C = E_K(P)$. We also let P' and C' be another plaintext and its corresponding ciphertext encrypted using K, $C' = E_K(P')$. The *difference* in the plaintext is defined as $\Delta P = P \oplus P'$ and for the ciphertext as $\Delta C = C \oplus C'$. Differential cryptanalysis studies the propagation of differences throughout the encryption rounds and various operations in the cipher. Usually the differences considered are in the form of XOR operation, since many ciphers perform key mixing with the intermediate data using XOR operation. However this may not always be the case. The most important observation is that the differences considered must allow the propagation of differences to be defined independently from the round keys.

A linear function in the cipher, such as bit permutation, does not affect the differences or they can be predicted with probability equal to one. For a nonlinear layer, the propagation of differences can be studied by exhaustively observing all the input differences and their possible output differences. The main result here is that the output differences of a nonlinear function may not be uniformly distributed.

The possible propagation of differences during an encryption process is defined by *differential characteristic*. Every characteristic has a plaintext difference that is used to predict the difference in the following round. The probability that a characteristic succeeds to predict the differences depends on the probability affecting nonlinear layer in each round. By assuming that the occurence of difference in each round are independent, the total probability is then computed as the product of the probabilites of various operations.

The key recovery process is done by exploiting the expected difference for intermediate data before the last round or some rounds near the last round of encryption. The attacker requests sufficient number of pair of plaintexts selected according to the plaintext difference and their corresponding ciphertext pairs. The attacker then guesses some portion of the last-round key, performs partial decryption with the ciphertext, and checks if the output difference in the characteristic is satisfied. For the correct guess of the key, the difference is expected to appear for a fraction of p or more, where p denotes the probability of the occurence of characteristic. If the probability p is not too low, the correct subkey is the one which yields the highest occurence satisfying the characteristic.

1.5.1 Truncated Differential Cryptanalysis

Truncated differential cryptanalysis [16] is a relaxation of differential cryptanalysis where the difference need not to be fully specified for every bit. It clusters several differentials together and this has been effectively applied to some word-oriented ciphers such as SAFER [19]. The specification of a truncated differential can be done by fixing some bits in the input/output differential and allowing the remaining bits to vary arbitrarily. Truncated differential cryptanalysis plays an important role in some extensions of differential cryptanalysis such as impossible differential cryptanalysis [1] and boomerang attack [35].

1.5.2 Impossible Differential Cryptanalysis

Impossible differential cryptanalysis exploits a differential characteristic in a cipher that never occur, or with probability zero. The term impossible differential was introduced by Biham *et al.* in the cryptanalysis of SKIPJACK [1]. However the concept of impossible differential was used by Knudsen earlier in the proposal of block cipher DEAL [17].

A straightforward method to obtain a differential with probability equal to zero is by encrypting sufficiently many plaintext pairs for a pre-specified input difference and observe the output difference that never occur. However, this method is clearly infeasible due to large search space. One practical way to obtain an impossible differential is by using the *miss-in-the-middle* approach [2]. This technique combines two (truncated)-differential with probability one so that they conflict in the middle after concatenation. The key recovery attack is done by guessing some parts of the last-round key and perform partial decryption. Using sufficiently many plaintext-ciphertext pairs, an attacker can obtain the correct key by taking the key that never yields the output difference in the ciphertext.

1.5.3 Improbable Differential Cryptanalysis

Besides high probability and zero probability differentials, low probability differentials can also be used to distinguish a cipher from random permutation. Referring to the work of Tezcan [33], differential cryptanalysis that uses a low probability differential is called *improbable differential cryptanalysis*. Similar approach was also mentioned independently in the work of Mala, Dakhilalian, and Shakiba [22].

Tezcan introduced a way to construct improbable differential by miss-in-the-middle like technique, called *almost miss-in-the-middle*. After two truncated differential with probability one that contradict in the middle of encryption are found, an attacker can expand the impossible differential using (truncated)-differential in the outer part of the impossible differential. This technique has been applied to attack reduced-round CLEFIA [33] and PRESENT [34].

1.6 Motivations

With more usage of mobile and ubiquotous devices in the recent time, the security and privacy issue have become the primary concerns. Cryptographic community has started developing encryption systems that can be efficiently implemented in terms of memory and power consumption, while at the same time maintaining the security level of the cryptosystem.

PRESENT [5] is one of the block ciphers designed specifically for lightweight devices. It supports 80-bit and 128-bit key length with 64-bit block size. The cipher has 31 rounds and each round consists of an 4×4 S-Box, bitwise permutation, and XOR key addition. PRESENT has been analysed and so far the the best attack is the multidimensional linear cryptanalysis on 26 rounds [9].

In [34] Tezcan observed that for some nonzero input differences to the S-Box of PRESENT, there exist some bits that remain the same in all possible corresponding output differences. These specific invariant bits are called *undisturbed bits*. For instance, with the input difference $\mathbf{9} = (1, 0, 0, 1)$ the least significant bit of every possible corresponding output difference is undisturbed and its value is equal to zero. He also observed that undisturbed bits appear in the inverse of S-Box of PRESENT. The existence of undisturbed bits can also be equally seen as a truncated differential with probability one for a given S-Box. This allows an attacker to have longer truncated differentials for bit-oriented ciphers. In [34], a 13-round improbable differential attack was provided for PRESENT and without using undisturbed bits the best attack of this type can only reach 7 rounds.

Proving the exact security bound of a block cipher against differential cryptanalysis is a challenging task. Typically the designer of a block cipher would perform computer-aided search to find the best differential characteristic on reduced-round version of the cipher. One obvious way to improve the complexity of the searching algorithm is by reducing the search space. In a separated work Sun *et al.* [32] used the undisturbed bits in the S-Box of PRESENT as additional constraints for searching the best differential in related-key settings. The existence of undisturbed bits removes some differential patterns that never occur and, hence, reduces the search space of the differential characteristics. The undisturbed bits are then converted into linear inequalities for Mixed-Integer Linear Programming (MILP) model. The term *conditional differential propagation* is used by the authors to describe this behaviour.

Although previous works have discussed the observations on undisturbed bits and its cryptanalytic applications, the relation of undisturbed bits with other properties of an S-Box remains unknown. The goal of this thesis is to address this open problem and presents the relation of undisturbed bits with other properties of an S-Box.

We breakdown our aim into several subproblems. Firstly, one may ask the implication of undisturbed bits to the component functions of an S-Box. Specifically, we would like to focus on the component functions of an S-Box where the undisturbed bits occur. Secondly, we want to see the notion of undisturbed bits from the point of view of two well-known cryptanalytic tools: *difference distribution table* (DDT) and *linear approximation table* (LAT). Thirdly, we ask whether there exists a dedicated cryptanalytic tool, similar to DDT and LAT, for the case of undisturbed bits. Lastly, we would like to see other properties of an S-Box that can be used to indicate the existence of nonzero input difference which has undisturbed bits in its corresponding output differences.

1.7 Contribution of the Thesis

We begin this thesis by providing the main background and properties of Boolean functions in Chapter 2. We investigate further the notion of undisturbed bits and provide our main results in Chapter 3. We will show that the occurence of undisturbed bits is related with the existence of nonzero linear structures in the coordinate functions of an S-Box. We also propose autocorrelation table as a dedicated tool to obtain all nonzero input differences of an S-Box which may yield undisturbed bits in the output differences. Autocorrelation table can then be seen as a counterpart of DDT for truncated differential cryptanalysis. In the same chapter, we also prove that by using algebraic degree of coordinate functions and balancedness property of an S-Box, the existence of input difference that yield undisturbed bits in the corresponding output difference can be shown for an $n \times m$ S-Box. The conclusions of our work and some possible open problems are presented in Chapter 4.

CHAPTER 2

BOOLEAN FUNCTIONS AND SUBSTITUTION BOXES

The aim of this chapter is to provide all necessary tools in order to understand the structures and properties of Boolean functions in the context of cryptography. Boolean functions are essentially the basic building blocks of various cryptographic primitives such as block ciphers, stream ciphers, hash functions, message authentication codes, etc.

We begin the first part by giving the definition of Boolean functions, its representations in terms of truth table and algebraic normal form, together with some important tools such as Walsh-Hadamard transform and autocorrelation. The concept of autocorrelation and linear structures of a Boolean function constitute the main tools that we will use to study S-Boxes with undisturbed bits. Several cryptographic criteria related with the concept of autocorrelation will also be defined in this section.

Generalization of Boolean functions in terms of vectorial Boolean functions, which is called Substitution Boxes in cryptography literature, is described in the second section. We start with the definition of an S-Box, followed by description of balanced S-Boxes. The two well-known cryptanalytic tools for an S-Box, which are difference distribution table (DDT) and linear approximation table (LAT), will also be introduced.

Note that this chapter only covers elementary topics about Boolean functions. For more extensive discussion on Boolean functions, S-Boxes, and other related topics, the reader may consult [10].

Notations

We define the set of integers $\mathbb{Z} = \{0, \pm 1, \pm 2, \pm 3, \ldots\}$ and the set of natural numbers $\mathbb{N} = \{1, 2, 3, \ldots\}$. Let $\mathbb{F}_2 = \{0, 1\}$ be a finite field with two elements and $\mathbb{F}_2^n = \{(x_{n-1}, \ldots, x_1, x_0) \mid x_i \in \mathbb{F}_2,\ 0 \leq i \leq n-1\}$ be an n-dimensional vector space over $\mathbb{F}_2$. The elements of $\mathbb{F}_2^n$ will be denoted $\overline{x} = (x_{n-1}, \ldots, x_1, x_0)$ where x_i are the *components* or *coordinates* of $\bar{x}$. Note that in this thesis every vector is considered as a column vector, but we will continue writing it in row-wise manner. The subscript indexing is reserved to refer to the components of a vector except for the case of standard basis in $\mathbb{F}_2^n$. The symbol $\oplus$ is used to denote the addition in $\mathbb{F}_2$ and $\mathbb{F}_2^n$.

One way to represent an elements of $\mathbb{F}_2^n$ is by using integer/decimal representation via the mapping $\varphi : \mathbb{F}_2^n \mapsto \mathbb{Z}$ such that for any $\overline{x} \in \mathbb{F}_2^n$ we have

$$\boldsymbol{x} = \varphi(\overline{x}) = \varphi(x_{n-1}, \ldots, x_1, x_0) = \sum_{i=0}^{n-1} x_i 2^i$$

The associated integer representation of vector $\overline{x}$ is written using boldface type font $\boldsymbol{x}$. The *lexicographical* ordering of the elements of $\mathbb{F}_2^n$ is defined as follows : $\overline{x} \leq \overline{y}$ if and only if $\varphi(\overline{x}) \leq \varphi(\overline{y})$. The standard basis of $\mathbb{F}_2^n$ is denoted by

$$\overline{e}_{n-1} = (1, 0, \ldots, 0), \quad \overline{e}_{n-2} = (0, 1, 0, \ldots, 0), \quad \cdots, \quad \overline{e}_0 = (0, \ldots, 0, 1)$$

We call the vector $\overline{e}_i \in \mathbb{F}_2^n$ as the i-th standard basis of $\mathbb{F}_2^n$. The integer representation of each i-th standard basis of $\mathbb{F}_2^n$ is given by $\mathbf{2}^i$. For arbitrary $\overline{x} \in \mathbb{F}_2^n$, we may write $\overline{x}$ as $\overline{x} = x_{n-1}\overline{e}_{n-1} \oplus \cdots \oplus x_0\overline{e}_0$. The elements $(0, \ldots, 0)$ and $(1, \ldots, 1)$ of $\mathbb{F}_2^n$, i.e. the zero-vector and one-vector, are denoted $\overline{0}$ and $\overline{1}$ respectively. The complement of $\overline{x} \in \mathbb{F}_2^n$ is denoted $\neg\overline{x}$ where $\neg\overline{x} = (x_{n-1} \oplus 1, \ldots, x_0 \oplus 1) = \overline{x} \oplus \overline{1}$.

Example 2.1. Let $\overline{x} = (1, 0, 0, 0, 1) \in \mathbb{F}_2^5$. We can write $\overline{x}$ as $\boldsymbol{x} = 17 \in \mathbb{Z}$ in the integer/decimal representation.

Let $\overline{x}, \overline{y} \in \mathbb{F}_2^n$, the *inner product* of $\overline{x}$ and $\overline{y}$ is defined as

$$\overline{x} \cdot \overline{y} = \bigoplus_{i=0}^{n-1} x_i y_i$$

The *Hamming weight* of a vector $\overline{x}$, $\mathrm{wt}(\overline{x})$, is defined as the number of nonzero components of $\overline{x}$. The set $\mathrm{Supp}(\overline{x})$ contains the index of nonzero components of the vector $\overline{x}$, i.e. $\mathrm{Supp}(x) = \{i \mid x_i \neq 0\}$. It can be easily seen that $|\mathrm{Supp}(\overline{x})| = \mathrm{wt}(\overline{x})$.

2.1 Boolean Functions

Definition 2.1. Let $n \in \mathbb{N}$. An n-variable Boolean function f is defined as $f : \mathbb{F}_2^n \mapsto \mathbb{F}_2$ that is a mapping from n-dimensional vector space over $\mathbb{F}_2$ into $\mathbb{F}_2$.

We denote $\mathcal{B}^n$ as the set of all n-variable Boolean functions. One way to represent a Boolean function is by exhaustively listing down the possible values of $f(\overline{x})$ for every $\overline{x} \in \mathbb{F}_2^n$ and order it lexicographically. The vector $(f(\mathbf{0}), \ldots, f(\mathbf{2}^n - \mathbf{1}))$ is called the *truth table* of f. The complement of a Boolean function $\neg f$ is the complement of its truth table. For $f \in \mathcal{B}^n$, its truth table has 2^n components in which for every $\overline{x} \in \mathbb{F}_2^n$, $f(\overline{x})$ has two possible values in $\mathbb{F}_2$. This leads to the following proposition of the number of n-variable Boolean functions.

Proposition 2.1. *Let $n \in \mathbb{N}$. The number of n-variable Boolean functions is 2^{2^n}, i.e.* $|\mathcal{B}^n| = 2^{2^n}$

The associated *sign function* $\widehat{f}(\overline{x})$ for every Boolean function f is defined by $\widehat{f}(\overline{x}) = (-1)^{f(\overline{x})} = 1 - 2f(\overline{x})$, whose values belong to the set $\{-1, 1\}$. The corresponding vector for the sign function $\widehat{f}$ represented by $(\widehat{f}(\mathbf{0}), \dots, \widehat{f}(\mathbf{2}^n - \mathbf{1}))$ is called *polarity truth table*. The wt(f), *weight* of a Boolean function f, is the weight of its truth table. The *support* of f is defined to be Supp$(f) = \{\overline{x} \in \mathbb{F}_2^n \mid f(\overline{x}) \neq 0\}$. A Boolean function $f \in \mathcal{B}^n$ where wt$(f) = 2^{n-1}$ is called a *balanced* function, i.e. there are equal number of zeros and ones in its truth table. For every $\overline{x} \in \mathbb{F}_2^n$, the Boolean function f where $f(\overline{x}) = c$ for a fixed $c \in \mathbb{F}_2$ is called a *constant function*. The *distance* of two Boolean functions f, g is defined as the number of entries in which they differ, i.e. dt$(f, g) = |\{\overline{x} \in \mathbb{F}_2^n \mid f(\overline{x}) \neq g(\overline{x})\}|$. It is trivial to check that dt$(f, g) =$ wt$(f \oplus g)$.

The second representation of Boolean functions is using algebraic expression introduced by Zhegalkin in 1927 [12].

$$f(\overline{x}) = f(x_{n-1}, \dots, x_1, x_0) = \bigoplus_{\overline{u} \in \mathbb{F}_2^n} a_{\overline{u}} x_{n-1}^{u_{n-1}} \cdots x_0^{u_0} = \bigoplus_{\overline{u} \in \mathbb{F}_2^n} a_{\overline{u}} \overline{x}^{\overline{u}} \tag{2.1}$$

The coefficient $a_{\overline{u}}$ is obtained by $a_{\overline{u}} = \bigoplus_{\overline{x} \preceq \overline{u}} f(\overline{x})$ where $\overline{x} \preceq \overline{u}$ means that $x_i \leq u_i$ for all $0 \leq i \leq n-1$ (we say that $\overline{u}$ covers $\overline{x}$). We refer to the expression given in Equation 2.1 as the *algebraic normal form* (ANF) of f. We call the product $x_{n-1}^{u_{n-1}} \cdots x_0^{u_0}$ a *monomial* and we refer to $a_{\overline{u}} x_{n-1}^{u_{n-1}} \cdots x_0^{u_0}$ as a *term*. For $\overline{u} = \overline{0}$ we indicate its associated term as the *constant term*. The degree of a Boolean function, $\deg(f)$, is defined as the maximal monomial degree in its ANF representation. The following proposition gives an upper bound for the degree of balanced functions.

Proposition 2.2 ([29]). *For a balanced n-variable Boolean function with $n \geq 2$, $\deg(f) \leq n - 1$*

An *affine function* is a Boolean function such that its ANF is of the form $\overline{\omega} \cdot \overline{x} \oplus \epsilon = \omega_{n-1}x_{n-1} \oplus \cdots \oplus \omega_0 x_0 \oplus \epsilon$ for $\overline{\omega} = (\omega_{n-1}, \dots, \omega_0) \in \mathbb{F}_2^n$ and $\epsilon \in \mathbb{F}_2$. The vector $\overline{\omega}$ is the *coefficient vector* of the affine function. We denote $\mathcal{A}^n$ as the set of all n-variable affine functions.

An element of the subset $\mathcal{L}^n \subseteq \mathcal{A}^n$ where $\epsilon = 0$, i.e. $\overline{\omega} \cdot \overline{x}$, is called a *linear function*. We also denote the linear function $\overline{\omega} \cdot \overline{x}$ using notation $l_{\overline{\omega}}(\overline{x})$.

Let $\overline{a}, \overline{b} \in \mathbb{F}_2^n$ and $l_{\overline{\omega}} : \mathbb{F}_2^n \mapsto \mathbb{F}_2$ be a linear function with coefficient vector $\overline{\omega} = (\omega_{n-1}, \dots, \omega_0)$. The linear Boolean function has the following properties

$$\begin{aligned} l_{\overline{\omega}}(\overline{a}) \oplus l_{\overline{\omega}}(\overline{b}) &= (\overline{\omega} \cdot \overline{a}) \oplus (\overline{\omega} \cdot \overline{b}) \\ &= (\omega_{n-1}a_{n-1} \oplus \cdots \oplus \omega_0 a_0) \oplus (\omega_{n-1}b_{n-1} \oplus \cdots \oplus \omega_0 b_0) \\ &= \omega_{n-1}a_{n-1} \oplus \omega_{n-1}b_{n-1} \oplus \cdots \oplus \omega_0 a_0 \oplus \omega_0 b_0 \\ &= \omega_{n-1}(a_{n-1} \oplus b_{n-1}) \oplus \cdots \oplus \omega_0(a_0 \oplus b_0) \\ &= \overline{\omega} \cdot (\overline{a} \oplus \overline{b}) = l_{\overline{\omega}}(\overline{a} \oplus \overline{b}) \end{aligned}$$

Thus we may see a linear Boolean function as a group homomorphism from $\mathbb{F}_2^n$ into $\mathbb{F}_2$. We define a *nonzero linear function* as a linear function with nonzero coefficient vector. Using the homomorphicity of linear Boolean functions, we can now prove the following result.

Theorem 2.3. *Every nonzero linear Boolean function is balanced.*

Proof. Suppose $l_{\overline{\omega}} : \mathbb{F}_2^n \mapsto \mathbb{F}_2$ is a nonzero linear Boolean function. For every $\overline{a}, \overline{b}$ we have $l_{\overline{\omega}}(\overline{a} \oplus \overline{b}) = l_{\overline{\omega}}(\overline{a}) \oplus l_{\overline{\omega}}(\overline{b})$ which makes us able to see $l_{\overline{\omega}}$ as a group homomorphism from $\mathbb{F}_2^n$ into $\mathbb{F}_2$ with respect to $\oplus$ operation in $\mathbb{F}_2^n$ and $\mathbb{F}_2$. Let $\text{Ker}_{l_{\overline{\omega}}} = \{\overline{x} \in \mathbb{F}_2^n \mid l_{\overline{\omega}}(\overline{x}) = 0\}$ be the kernel of $l_{\overline{\omega}}$ and $\text{Im}_{l_{\overline{\omega}}} = \{l_{\overline{\omega}}(\overline{x}),\ \forall \overline{x} \in \mathbb{F}_2^n\}$ be the image of $l_{\overline{\omega}}$. From the first group isomorphism theorem, the quotient group $\mathbb{F}_2^n / \text{Ker}_{l_{\overline{\omega}}}$ is isomorphic to the set $\text{Im}_{l_{\overline{\omega}}}$.

For every $\overline{a} \in \mathbb{F}_2^n$ we have $l_{\overline{\omega}}(\overline{0}) = l_{\overline{\omega}}(\overline{a} \oplus \overline{a}) = l_{\overline{\omega}}(\overline{a}) \oplus l_{\overline{\omega}}(\overline{a}) = 0$ and since $l_{\overline{\omega}}$ is a nonzero linear function, then there exists a $\overline{b} \in \mathbb{F}_2^n$ such that $l_{\overline{\omega}}(\overline{b}) \neq 0$. It follows that $l_{\overline{\omega}}$ is onto/surjective function. We may then deduce that $\text{Im}_{l_{\overline{\omega}}} = \mathbb{F}_2$. Since $|\mathbb{F}_2^n| = 2^n$ and $|\text{Im}_{l_{\overline{\omega}}}| = |\mathbb{F}_2| = 2$, it implies that $|\text{Ker}_{l_{\overline{\omega}}}| = 2^{n-1}$. Clearly, $l_{\overline{\omega}}$ is balanced. □

Corollary 2.4. *Every affine function with nonzero coefficient vector is balanced. If the coefficient vector is zero vector, the affine function is a constant function.*

Proof. Let $\overline{\omega} \cdot \overline{x} \oplus \epsilon$ be an affine function. The case when $\overline{\omega} \neq \overline{0}$ and $\epsilon = 0$ follows from Theorem 2.3. The case for $\overline{\omega} \neq \overline{0}$ and $\epsilon = 1$ follows from the fact that $\overline{\omega} \cdot \overline{x} \oplus 1$ is complement of $\overline{\omega} \cdot \overline{x}$, and hence, it is also balanced. The case when $\overline{\omega} = \overline{0}$ is obvious. □

Corollary 2.5. *If $\overline{\omega} \in \mathbb{F}_2^n$ then we have*

$$\sum_{\overline{x} \in \mathbb{F}_2^n} (-1)^{\overline{\omega} \cdot \overline{x}} = \begin{cases} 2^n & \textit{if } \overline{\omega} = \overline{0} \\ 0 & \textit{otherwise} \end{cases}$$

Proof. The proof for $\overline{\omega} = \overline{0}$ is trivial. The case for $\overline{\omega} \neq \overline{0}$ follows immediately from Theorem 2.3 □

The relation between two Boolean functions can be seen from the point of view of *cross-correlation*, which is a real-valued function. We define cross-correlation between two Boolean functions f and g below.

Definition 2.2 (Cross-correlation)**.** Let $f, g \in \mathcal{B}^n$ be n-variable Boolean functions. The cross-correlation of f and g at $\overline{\omega} \in \mathbb{F}_2^n$ is defined as

$$C_{f,g}(\overline{\omega}) = \sum_{\overline{x} \in \mathbb{F}_2^n} (-1)^{f(\overline{x})} (-1)^{g(\overline{x} \oplus \overline{\omega})} = \sum_{\overline{x} \in \mathbb{F}_2^n} \widehat{f}(\overline{x}) \widehat{g}(\overline{x} \oplus \overline{\omega})$$

Intuitively, cross-correlation tries to relate the function f with permutation of function g where $\overline{\omega}$ acts as the permutation parameter for g. We may see that for $\overline{\omega} = \overline{0}$, it gives an identity permutation and cross-correlation obviously shows the relation of f and the original function g. From this perspective, the notion of *correlation* is introduced.

Definition 2.3 (Correlation)**.** Let f, g be n-variable Boolean functions. The correlation of f and g is defined as

$$\begin{aligned}\mathfrak{C}_{f,g} = C_{f,g}(\overline{0}) &= \sum_{\overline{x}\in\mathbb{F}_2^n} (-1)^{f(\overline{x})}(-1)^{g(\overline{x})} \\ &= \sum_{\overline{x}\in\mathbb{F}_2^n} \widehat{f}(\overline{x})\widehat{g}(\overline{x})\end{aligned}$$

Theorem 2.6. *Let $f, g \in \mathcal{B}^n$ be n-variable Boolean functions. The relation of $\mathfrak{C}_{f,g}$ and the distance of f and g is given as*

$$\mathfrak{C}_{f,g} = 2^n - 2 \cdot dt(f, g)$$

Proof. For some $\overline{u} \in \mathbb{F}_2^n$, the case when $f(\overline{u}) = g(\overline{u})$ implies that $\widehat{f}(\overline{x})\widehat{g}(\overline{x}) = 1$. For some $\overline{v} \in \mathbb{F}_2^n$, the case when $f(\overline{v}) \neq g(\overline{v})$ implies that $\widehat{f}(\overline{x})\widehat{g}(\overline{x}) = -1$. From the Definition 2.3, the number of -1 in the summation is equal to $|\{\overline{x} \in \mathbb{F}_2^n \mid f(\overline{x}) \neq g(\overline{x})\}| = \mathrm{dt}(f, g)$. Similarly, the number of $+1$ in the summation can be expressed as $2^n - \mathrm{dt}(f, g)$. Clearly we have

$$\mathfrak{C}_{f,g} = (2^n - \mathrm{dt}(f, g)) - \mathrm{dt}(f, g) = 2^n - 2 \cdot \mathrm{dt}(f, g)$$

□

2.1.1 Walsh-Hadamard Transform

In the analysis of a Boolean function, *Walsh-Hadamard transform* is an important tool that can determine various properties of the function. We give the following definition of Walsh-Hadamard transform as well as its inverse transform.

Definition 2.4 (Walsh-Hadamard Transform)**.** The Walsh-Hadamard Transform of f at $\overline{\omega} \in \mathbb{F}_2^n$ is defined by

$$\mathcal{W}_f(\overline{\omega}) = \sum_{\overline{x}\in\mathbb{F}_2^n} (-1)^{f(\overline{x})}(-1)^{\overline{\omega}\cdot\overline{x}} = \sum_{\overline{x}\in\mathbb{F}_2^n} \widehat{f}(\overline{x})(-1)^{\overline{\omega}\cdot\overline{x}}$$

The inverse transform is defined by

$$\widehat{f}(\overline{x}) = 2^{-n} \sum_{\overline{\omega}\in\mathbb{F}_2^n} \mathcal{W}_f(\overline{\omega})(-1)^{\overline{x}\cdot\overline{\omega}}$$

The vector $(\mathcal{W}_f(\mathbf{0}), \ldots, \mathcal{W}_f(\mathbf{2^n{-}1}))$ is called the *Walsh spectrum* of f.

One of the properties of a Boolean function that can be determined from the Walsh value is balancedness. Note that $\mathcal{W}_f(\overline{0}) = \sum_{\overline{x}\in\mathbb{F}_2^n}(-1)^{f(\overline{x})}$ and if f is a balanced function, clearly $\mathcal{W}_f(\overline{0}) = 0$. The converse is also true.

Proposition 2.7. *A Boolean function f is balanced if and only if $\mathcal{W}_f(\overline{0}) = 0$.*

2.1.2 Autocorrelation and Derivative

Another important tool in the analysis of a Boolean function, which will also be used to study S-Boxes with undisturbed bits throughout this thesis, is the idea of *autocorrelation.*

Definition 2.5 (Autocorrelation)**.** The autocorrelation of an n-variable Boolean function f at $\overline{\alpha} \in \mathbb{F}_2^n$ is defined by

$$r_f(\overline{\alpha}) = \sum_{\overline{x} \in \mathbb{F}_2^n} (-1)^{f(\overline{x})}(-1)^{f(\overline{x} \oplus \overline{\alpha})} = \sum_{\overline{x} \in \mathbb{F}_2^n} (-1)^{f(\overline{x}) \oplus f(\overline{x} \oplus \overline{\alpha})}$$

.

We refer to the vector $(r_f(\mathbf{0}), \ldots, r_f(\mathbf{2^n{-}1}))$ as the *autocorrelation spectrum* of f. The relation of autocorrelation and Walsh-transform is given by the following theorem.

Theorem 2.8 (Wiener-Khinthcine [25])**.** *The expression of autocorrelation in terms of Walsh value is equal to*

$$r_f(\overline{\alpha}) = 2^{-n} \sum_{\overline{\omega} \in \mathbb{F}_2^n} \mathcal{W}_f^2(\overline{\omega})(-1)^{\overline{\alpha} \cdot \overline{\omega}}$$

and the inverse relation is given by

$$\mathcal{W}_f^2(\overline{\omega}) = \sum_{\overline{\alpha} \in \mathbb{F}_2^n} r_f(\overline{\alpha})(-1)^{\overline{\omega} \cdot \overline{\alpha}}$$

The *derivative* of f at $\overline{\alpha} \in \mathbb{F}_2^n$ is defined as $D_{\overline{\alpha}} f(\overline{x}) = f(\overline{x}) \oplus f(\overline{x} \oplus \overline{\alpha})$. The derivative of f at any point in $\mathbb{F}_2^n$ can also be treated as an n-variable Boolean function. The autocorrelation of a Boolean function can then be expressed in terms of its derivative as $r_f(\overline{\alpha}) = \sum_{\overline{x} \in \mathbb{F}_2^n} (-1)^{D_{\overline{\alpha}} f(\overline{x})}$. The following proposition gives an upper bound to the degree of the derivative of a function.

Proposition 2.9 ([21])**.** *If f is an n-variable Boolean function and $\overline{\alpha} \in \mathbb{F}_2^n$, then* $\deg(D_{\overline{\alpha}} f) \leq \deg(f) - 1$.

If $D_{\overline{\alpha}} f(\overline{x})$ is a constant function, then $\overline{\alpha}$ is a *linear structure* of f [20] [13]. The zero vector $\overline{0} \in \mathbb{F}_2^n$ is a trivial linear structure since $D_{\overline{0}} f(\overline{x}) = 0$ for all $\overline{x} \in \mathbb{F}_2^n$. We say that the function f has a linear structure if there exists a nonzero vector $\overline{\alpha} \in \mathbb{F}_2^n$ such that $D_{\overline{\alpha}} f(\overline{x})$ is a constant function. The notation $\mathcal{LS}_f$ is used to define the set of all linear structures of f. The set of all n-variable Boolean functions that has linear structure is denoted by $\mathcal{LS}(n)$. From the point of view of autocorrelation, a vector in $\mathbb{F}_2^n$ is a linear structure if it satisfies the following proposition.

Proposition 2.10. *The vector $\overline{\alpha} \in \mathbb{F}_2^n$ is a linear structure of f if and only if $r_f(\overline{\alpha}) = \pm 2^n$.*

Proposition 2.11. *Any vector in $\mathbb{F}_2^n$ is a linear structure of every affine function.*

Proof. Let $\overline{\alpha} \in \mathbb{F}_2^n$. Recall that we can represent an affine function as $\overline{\omega} \cdot \overline{x} \oplus \epsilon$ with $\overline{\omega} \in \mathbb{F}_2^n$ and $\epsilon \in \mathbb{F}_2$. The derivative of the affine function $\overline{\omega} \cdot \overline{x} \oplus \epsilon$ at $\overline{\alpha}$ is equal to

$$\begin{aligned}(\overline{\omega} \cdot \overline{x} \oplus \epsilon) \oplus (\overline{\omega} \cdot (\overline{x} \oplus \overline{\alpha}) \oplus \epsilon) &= (\overline{\omega} \cdot \overline{x} \oplus \epsilon) \oplus ((\overline{\omega} \cdot \overline{x} \oplus \overline{\omega} \cdot \overline{\alpha}) \oplus \epsilon) \\ &= \overline{\omega} \cdot \overline{\alpha}\end{aligned}$$

This implies that the derivative of the affine function $\overline{\omega} \cdot \overline{x} \oplus \epsilon$ at $\overline{\alpha}$ is equal to $\overline{\omega} \cdot \overline{\alpha}$ for all $\overline{x} \in \mathbb{F}_2^n$ and, hence, is a constant function. Clearly, $\overline{\alpha}$ is a linear structure of $\overline{\omega} \cdot \overline{x} \oplus \epsilon$. □

2.1.2.1 Related Cryptographic Criteria

A cryptographic criteria which is closely related to autocorrelation is *Strict Avalanche Criterion* (SAC). An n-variable Boolean function f satisfies SAC if changing any one of the n bits in the input results in the output of the function being changed with probability $1/2$. It is clear that the following proposition follows from the definition of SAC and could be treated as an equivalent definition.

Proposition 2.12. *An n-variable Boolean function f satisfies SAC if and only if the function $f(\overline{x}) \oplus f(\overline{x} \oplus \overline{\alpha})$ is balanced for every $\overline{\alpha} \in \mathbb{F}_2^n$ with wt$(\overline{\alpha}) = 1$. Equivalently, the function f satisfies SAC if and only if $r_f(\overline{\alpha}) = 0$, with wt$(\overline{\alpha}) = 1$.*

An n-variable Boolean function is said to satisfy *propagation criterion* of degree k, PC(k), if changing any i ($1 \leq i \leq k$) of the n bits in the input results in the output of the function being changed for half of the times. This definition generalizes the notion of SAC, which is clearly equal to PC(1) function. The following proposition is analogous to the one given in Proposition 2.12.

Proposition 2.13. *An n-variable Boolean function f satisfies PC(k) if and only if*

$$r_f(\overline{\alpha}) = \sum_{\overline{x} \in \mathbb{F}_2^n} (-1)^{f(\overline{x})} (-1)^{f(\overline{x} \oplus \overline{\alpha})} = 0, \qquad 1 \leq wt(\overline{\alpha}) \leq k$$

We can also restate Proposition 2.13 in terms of the derivative of f as follows.

Proposition 2.14. *An n-variable Boolean function f satisfies PC(k) if and only if $D_{\overline{\alpha}} f(\overline{x})$ is a balanced function for each $1 \leq wt(\overline{\alpha}) \leq k$.*

2.2 Substitution Boxes

An $n \times m$ *substitution box* (or S-Box in short) is a mapping $S : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$. The internal structure of an S-Box can be decomposed into Boolean functions. Let $\overline{y} = (y_{m-1}, \ldots, y_0) \in \mathbb{F}_2^m$ and $\overline{y} = S(\overline{x})$. The component of $\overline{y}$ can be computed by $y_i = h_i(\overline{x})$ for each $i \in \{0, \ldots, m-1\}$. The functions $h_i : \mathbb{F}_2^n \mapsto \mathbb{F}_2$ are called the *coordinate functions* of the S-Box S. The *component functions* of the S-Box S are the mapping $\overline{b} \cdot S(\overline{x})$ for all nonzero $\overline{b} \in \mathbb{F}_2^m$. The component functions are essentially generalization of coordinate functions of an S-Box by considering its linear combinations, i.e.

for nonzero $\overline{b} = (b_{m-1}, \ldots, b_0) \in \mathbb{F}_2^m$ we have $\overline{b} \cdot S(\overline{x}) = b_{m-1}h_{m-1}(\overline{x}) \oplus \ldots \oplus b_0h_0(\overline{x})$. It follows that the coordinate function $h_i(\overline{x}) = \overline{e}_i \cdot S(\overline{x})$ where $\overline{e}_i$ is the i-th standard basis of $\mathbb{F}_2^m$.

An $n \times m$ S-Box S is *balanced* (or regular) if it takes every value of $\mathbb{F}_2^m$ the same number 2^{n-m} of times. The following proposition characterizes a balanced S-Box from the balancedness of its component functions.

Proposition 2.15 ([6]). *An $n \times m$ S-Box is balanced if and only if its component functions are balanced, that is if and only if for every nonzero $\overline{b} \in \mathbb{F}_2^m$, the Boolean function $\overline{b} \cdot S(\overline{x})$ is balanced.*

The notion of linear structures in Boolean functions can be extended for the case of S-Boxes. The definition of an S-Box that has linear structure was originally proposed by Chaum [8] and Evertse [13]. They defined that an S-Box has linear structure by considering the existence of nonzero linear structures in any of the component functions of the S-Box.

Definition 2.6 (S-Box with linear structures [8][13][24]). An $n \times m$ S-Box S is said to have a linear structure if there exists a nonzero vector $\overline{\alpha} \in \mathbb{F}_2^n$ together with a nonzero vector $\overline{b} \in \mathbb{F}_2^m$ such that $\overline{b} \cdot S(\overline{x}) \oplus \overline{b} \cdot S(\overline{x} \oplus \overline{\alpha})$ takes the same value $c \in \mathbb{F}_2$ for all $\overline{x} \in \mathbb{F}_2^n$.

Proposition 2.16. *An $n \times m$ S-Box S is said to have a linear structure if there exists a nonzero vector $\overline{\alpha} \in \mathbb{F}_2^n$ together with a nonzero vector $\overline{b} \in \mathbb{F}_2^m$ such that $r_{\overline{b} \cdot S}(\overline{\alpha}) = \pm 2^n$.*

In the cryptanalysis of block ciphers, the two most important cryptanalytic tools to analyse properties of an S-Box are *difference distribution table* (DDT) [4] and *linear approximation table* (LAT) [23].

Let $\overline{x}, \overline{x}' \in \mathbb{F}_2^n$ be two inputs to S-Box S and $\overline{y} = S(\overline{x}),\ \overline{y}' = S(\overline{x}')$ be their corresponding outputs. We refer to the difference in the input $\overline{x} \oplus \overline{x}' = \overline{\alpha}$ as the *input difference* to S. Similarly $\overline{y} \oplus \overline{y}' = \overline{\beta}$ is the *output difference* of S correponding to input difference $\overline{\alpha}$. DDT examines how many times a certain output difference of an S-Box occurs for a given input difference. The definition of DDT is given as follows.

Definition 2.7. For an $n \times m$ S-Box S, the entry in the row $\overline{s} \in \mathbb{F}_2^n$ and column $\overline{t} \in \mathbb{F}_2^m$ (considering their integer representation) of difference distribution table of S is defined by $\mathsf{DDT}(\boldsymbol{s}, \boldsymbol{t}) = |\{\overline{x} \in \mathbb{F}_2^n \mid S(\overline{x}) \oplus S(\overline{x} \oplus \overline{s}) = \overline{t}\}|$.

The probability of an input difference $\overline{\alpha}$ yielding the output difference $\overline{\beta}$ is then defined by

$$\begin{aligned}\mathbf{Pr}_S[\overline{\alpha} \to \overline{\beta}] &= 2^{-n}|\{\overline{x} \in \mathbb{F}_2^n \mid S(\overline{x}) \oplus S(\overline{x} \oplus \overline{\alpha}) = \overline{\beta}\}| \\ &= 2^{-n} \cdot \mathsf{DDT}(\boldsymbol{\alpha}, \boldsymbol{\beta})\end{aligned}$$

On the other hand, LAT is used to find the best linear approximation for an S-Box involving the parity bits of its input and output. The definition of linear approximation table is given below.

Definition 2.8. For an $n \times m$ S-Box S, the linear approximation table of S at row $\overline{s} \in \mathbb{F}_2^n$ and column $\overline{t} \in \mathbb{F}_2^m$ (considering their integer representation) is defined as

$$\mathsf{LAT}(\boldsymbol{s}, \boldsymbol{t}) = |\{\overline{x} \in \mathbb{F}_2^n \mid \overline{s} \cdot \overline{x} = \overline{t} \cdot S(\overline{x})\}| - 2^{n-1}$$

CHAPTER 3

UNDISTURBED BITS

One of the earliest techniques of block cipher cryptanalysis was due to Biham and Shamir, called differential cryptanalysis [3]. Let E_K be an encryption function with a fixed key K. Let P_1, P_2 be two different plaintexts that lead to the ciphertexts $C_1 = E_K(P_1)$ and $C_2 = E_K(P_2)$ after applying E_K. The goal of differential cryptanalysis is to find a relation $\Delta P = P_1 \oplus P_2$ that leads to $\Delta C = C_1 \oplus C_2$ with probability higher than a random permutation. The pair $(\Delta P, \Delta C)$ is called a differential. The strategy to obtain ΔP that leads to ΔC with high probability is by combining differential characteristics, a sequence of input and output differences to the round function in a block cipher.

In [16] Knudsen improved the previous differential cryptanalysis using truncated differentials. In classical differential cryptanalysis, the differential is fully specified for every bit, whereas truncated differentials provide a way to significantly improve the differential cryptanalysis by specifiying only some part of the differentials.

During the last decades, extensive usage of ubiquitous devices as well as low-end devices such as RFID, leads to dramatic needs for security and privacy of data stored in such devices. The main challenges to design encryption mechanisms suitable for low-end devices are the limited memory and power available. Some of the lightweight block ciphers such as PRESENT [5] and RECTANGLE [36] are designed in bit-oriented fashion. This is due to the efficiency of bit-level operations for hardware implementation.

In order to mount truncated differential cryptanalysis on a bit-oriented block cipher, the only part which should be examined closely is the nonlinear operation, usually done by substitution boxes. In [34] Tezcan provided observations on the S-Box of PRESENT that help achieving longer truncated differentials. For a particular nonzero input difference to the S-Box of PRESENT, there exist some bits that remain equal in all the possible corresponding output differences. These specific bits are called *undisturbed bits*.

$\overline{x}$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$S(\overline{x})$	12	5	6	11	9	0	10	13	3	14	15	8	4	7	1	2

Table 3.1: The 4×4 S-Box of PRESENT.

As a brief example, let $(1,0,0,1) = \mathbf{9}$ be an input difference to the S-Box of PRESENT. By looking at its DDT, all the possible correponding output differences are the vectors $\mathbf{2}, \mathbf{4}, \mathbf{6}, \mathbf{8}, \mathbf{12}, \mathbf{14}$. Notice that in binary form, the rightmost bit value remains invariant for all the possible output differences, which is equal to zero (See Table 3.2).

Input Difference	Possible Output Differences	Probability
$\mathbf{9} = (1,0,0,1)$	$\mathbf{2} = (0,0,1,0)$	2^{-3}
	$\mathbf{4} = (0,1,0,0)$	2^{-2}
	$\mathbf{6} = (0,1,1,0)$	2^{-3}
	$\mathbf{8} = (1,0,0,0)$	2^{-3}
	$\mathbf{12} = (1,1,0,0)$	2^{-3}
	$\mathbf{14} = (1,1,1,0)$	2^{-2}
	$(*,*,*,0)$	1

Table 3.2: Example of undisturbed bit in the PRESENT's S-Box. The symbol $*$ denotes arbitrary value of zero and one.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	4	0	0	0	4	0	4	0	0	0	4	0	0
2	0	0	0	2	0	4	2	0	0	0	2	0	2	2	2	0
3	0	2	0	2	2	0	4	2	0	0	2	2	0	0	0	0
4	0	0	0	0	0	4	2	2	0	2	2	0	2	0	2	0
5	0	2	0	0	2	0	0	0	0	2	2	2	4	2	0	0
6	0	0	2	0	0	0	2	0	2	0	0	4	2	0	0	4
7	0	4	2	0	0	0	2	0	2	0	0	0	2	0	0	4
8	0	0	0	2	0	0	0	2	0	2	0	4	0	2	0	4
9	0	0	2	0	4	0	2	0	2	0	0	0	2	0	4	0
10	0	0	2	2	0	4	0	0	2	0	2	0	0	2	2	0
11	0	2	0	0	2	0	0	0	4	2	2	2	0	2	0	0
12	0	0	2	0	0	4	0	2	2	2	2	0	0	0	2	0
13	0	2	4	2	2	0	0	2	0	0	2	2	0	0	0	0
14	0	0	2	2	0	0	2	2	2	2	0	0	2	2	0	0
15	0	4	0	0	4	0	0	0	0	0	0	0	0	0	4	4

Table 3.3: DDT of the S-Box of PRESENT.

Moreover, in [34] similar occurrences in the rightmost bit of the output difference vector is also observed for input differences $\mathbf{1}$ and $\mathbf{8}$, but with undisturbed bit value equal to 1. In the same paper, the author also shows the existence of undisturbed bits in the inverse of PRESENT's S-Box. However, it is sufficient to study the notion of undisturbed bits in terms of mapping from $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$. For a bijective S-Box, the results from this chapter can be applied to its inverse.

In this chapter, we further study the undisturbed bits and give more mathematical treatment on the subject. The definition of undisturbed bits and its connection with the concept of linear structures in Boolean functions are given in the first section. The relation of difference distribution table and linear approximation table with undisturbed bits are presented in Section 3.2. Autocorrelation table is introduced in Section 3.3, as a tool to analyse an S-Box in order to obtain nonzero input differences that may yield undisturbed bits in the S-Box. In Section 3.4, we recall some results from [7] about the existence of nonzero linear structures in balanced quadratic Boolean functions and use it to prove the existence of undisturbed bits for balanced $n \times m$ S-Boxes with quadratic coordinate functions. The result is also used as an alternative proof for the proposition given by Tezcan in [34] about the existence of undisturbed bits in bijective 3×3 S-Boxes. We summarize and conclude the results of this chapter in Section 3.5.

3.1 Undisturbed Bits and Linear Structures

In this section we recall the definition of undisturbed bits and provide its relations with autocorrelation, derivative, and linear structure of coordinate functions in an S-Box. The notation $S = (h_{m-1}, \ldots, h_0)$ will be used consistently for the rest of this chapter to denote an $n \times m$ S-Box $S : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$ with coordinate functions $h_{m-1}, \ldots, h_0$, where $h_i : \mathbb{F}_2^n \mapsto \mathbb{F}_2$.

Definition 3.1 (Undisturbed Bits)**.** Let $\overline{\alpha} \in \mathbb{F}_2^n$ be a nonzero input difference to the S-Box S and $\Omega_{\overline{\alpha}} = \{\overline{\beta} = (\beta_{m-1}, \ldots, \beta_0) \in \mathbb{F}_2^m \mid \mathbf{Pr}_S[\overline{\alpha} \to \overline{\beta}] > 0\}$ be the set of all possible output differences of S corresponding to $\overline{\alpha}$. If $\beta_i = c$ for a fixed $c \in \mathbb{F}_2$ and for all $\overline{\beta} \in \Omega_{\overline{\alpha}}$ with $i \in \{0, \ldots, m-1\}$, then the S-Box S has undisturbed bits. In particular, we say that for input difference $\overline{\alpha}$, the i-th bit of the output difference of S is undisturbed (and its value is c).

Example 3.1. For PRESENT's S-Box with input difference **1**, the least significant bit (rightmost) of all possible output differences is equal to 1. Following Definition 3.1, we say that for input difference **1**, the 0-th bit of the output difference of PRESENT's S-Box is undisturbed and its value is 1. Similarly, for input difference **9**, the 0-th bit of the output difference of PRESENT's S-Box is undisturbed and its value is 0.

Recall that any output of an S-Box as the element of $\mathbb{F}_2^m$ can be computed componentwisely using coordinate functions of the S-Box. If $\mathbf{Pr}_S[\overline{\alpha} \to \overline{\beta}] > 0$, then there exists a $\overline{v} \in \mathbb{F}_2^n$ such that $S(\overline{v}) \oplus S(\overline{v} \oplus \overline{\alpha}) = \overline{\beta}$. It follows that the component of the output difference vector $\overline{\beta} = (\beta_{m-1}, \ldots, \beta_0)$ can be computed by $\beta_i = h_i(\overline{v}) \oplus h_i(\overline{v} \oplus \overline{\alpha})$. The following result is an implication from this observation.

Theorem 3.1. *For a nonzero input difference $\overline{\alpha} \in \mathbb{F}_2^n$ and $i \in \{0, \ldots, m-1\}$, the i-th bit of the output difference of S is undisturbed if and only if $D_{\overline{\alpha}}h_i(\overline{x}) = h_i(\overline{x}) \oplus h_i(\overline{x} \oplus \overline{\alpha})$ is a constant function.*

Proof. Suppose for an input difference $\overline{\alpha}$, the i-th bit of the output difference of S is undisturbed. Let $\Omega_{\overline{\alpha}} = \{\overline{\beta} = (\beta_{m-1}, \ldots, \beta_0) \in \mathbb{F}_2^m \mid \mathbf{Pr}_S[\overline{\alpha} \to \overline{\beta}] > 0\}$ be the set of all possible output differences of S corresponding to $\overline{\alpha}$. Definition 3.1 tells us that for all $\overline{\beta} = (\beta_{m-1}, \ldots, \beta_0) \in \Omega_{\overline{\alpha}}$ the component $\beta_i = c$ for a fixed $c \in \mathbb{F}_2$. Since $\beta_i = h_i(\overline{v}) \oplus h_i(\overline{v} \oplus \overline{\alpha})$ for some $\overline{v} \in \mathbb{F}_2^n$ and because the computation of output differences in $\Omega_{\overline{\alpha}}$ runs through all the elements of $\mathbb{F}_2^n$, clearly $D_{\overline{\alpha}}h_i(\overline{x}) = h_i(\overline{x}) \oplus h_i(\overline{x} \oplus \overline{\alpha}) = c$ for all $\overline{x} \in \mathbb{F}_2^n$. Hence $D_{\overline{\alpha}}h_i(\overline{x})$ is a constant function. The converse part of the proof can be done by reversing the previous step. □

The value of undisturbed bits can then be deduced whether the constant function $D_{\overline{\alpha}}h_i(\overline{x})$ is equal to zero or one, for each $\overline{x} \in \mathbb{F}_2^n$. Because $D_{\overline{\alpha}}h_i(\overline{x})$ is a constant function, then the nonzero vector $\overline{\alpha}$ is a linear structure of the coordinate function h_i. Equivalently, since $\overline{\alpha}$ is a nonzero vector, then h_i is a function with linear structure. This result shows that a particular S-Box has undisturbed bits if any of its coordinate function has nontrivial linear structures. In order to see if an S-Box has undisturbed bits, it is then sufficient to check the derivative of each coordinate function at every nonzero element of $\mathbb{F}_2^n$.

Theorem 3.1 also relates an S-Box which has undisturbed bits with Definition 2.6 about an S-Box with linear structures. It shows that an S-Box that has undisturbed bits belongs to a special class of S-Boxes with linear structures by only considering the existence of linear structures in its coordinate functions. This can be described by the following proposition, and it can be treated as an equivalent definition for an S-Box that has undisturbed bits.

Proposition 3.2. *An $n \times m$ S-Box S is said to have an undisturbed bit if there exists a nonzero vector $\overline{\alpha} \in \mathbb{F}_2^n$ together with a nonzero vector $\overline{b} \in \mathbb{F}_2^m$ with wt$(b) = 1$ such that $\overline{b} \cdot S(\overline{x}) \oplus \overline{b} \cdot S(\overline{x} \oplus \overline{\alpha})$ takes the same value $c \in \mathbb{F}_2$ for all $\overline{x} \in \mathbb{F}_2^n$.*

In other words, if an S-Box S has undisturbed bits, then S has a linear structure. However, the converse is not true in general. Thus, Definition 2.6 can be seen as a generalization of undisturbed bits.

The existence of undisturbed bits in an S-Box may also be used to describe the unsatisfiability of the corresponding coordinate functions against SAC. We state it in the following remark.

Remark 3.1. Let $\mathcal{I}_i = \{\overline{\alpha} \in \mathbb{F}_2^n,\ \overline{\alpha} \neq \overline{0} \mid h_i(\overline{x}) \oplus h_i(\overline{x} \oplus \overline{\alpha}) \text{ is a constant function}\}$ be the set such that for any $\overline{\alpha} \in \mathcal{I}_i$ the i-th bit of the output difference of S is undisturbed. Equivalently $\mathcal{I}_i$ is the set of all nonzero linear structures of the coordinate function h_i, i.e. $\mathcal{I}_i = \mathcal{LS}_{h_i} \setminus \{\overline{0}\}$. We set

$$d = \min_{\overline{\alpha} \in \mathcal{I}_i} \mathrm{wt}(\overline{\alpha})$$

If $d = 1$, then from Proposition 2.12 it follows that the coordinate function h_i does not satisfy Strict Avalanche Criterion (SAC).

For input difference **1**, **8**, and **9**, the 0-th bit of the output difference of PRESENT's S-Box is undisturbed. Here we have $d = 1$ and it follows that the coordinate function h_0 of PRESENT's S-Box does not satisfy Strict Avalanche Criterion (SAC).

Note that when a coordinate function of an S-Box does not satisfy SAC, this does not mean that the S-Box has undisturbed bits. This remark also can not be generalized for $d > 1$. The reason is because if there exists a d' with $1 \leq d' < d$ such that the coordinate function does not satisfy PC(d') then d is not a proper bound for the unsatisfiability condition.

A trivial lemma can be derived from Theorem 3.1 to indicate whether an S-Box has undisturbed bits from the autocorrelation of its coordinate functions. We will use the following lemma to show the relation of other cryptanalytic tools with undisturbed bits.

Lemma 3.3. *For a nonzero input difference $\overline{\alpha} \in \mathbb{F}_2^n$, the i-th bit of the output difference of S is undisturbed if and only if*

$$r_{h_i}(\overline{\alpha}) = \pm 2^n$$

for $i \in \{0, \ldots, m-1\}$.

Proof. Suppose for a nonzero input difference $\overline{\alpha} \in \mathbb{F}_2^n$, the i-th bit of the output difference of S is undisturbed. From Theorem 3.1 the vector $\overline{\alpha}$ is a linear structure of the coordinate function h_i. It follows that from Proposition 2.10 we have $r_{h_i}(\overline{\alpha}) = \pm 2^n$. The converse can be proven by reversing the previous steps. □

Lemma 3.3 tells us that one can observe the existence of undisturbed bits in an S-Box by computing the autocorrelation spectrum of each coordinate function of the S-Box. This approach gives a more straightforward way to find nonzero input differences that yield some bits in its corresponding output difference undisturbed.

3.2 Undisturbed Bits, LAT, and DDT

Recall that DDT of an $n \times m$ S-Box S at row a and column b is defined by $\mathsf{DDT}(\boldsymbol{a}, \boldsymbol{b}) = |\{\overline{x} \in \mathbb{F}_2^n \mid S(\overline{x}) \oplus S(\overline{x} \oplus \overline{a}) = \overline{b}\}|$. The following theorem given in [37] provides a relation between DDT and autocorrelation of the component functions of an S-Box. Using Lemma 3.3 the relation of undisturbed bits and DDT can be easily shown in Corollary 3.5.

Theorem 3.4 ([37])**.** *The relation between difference distribution table and the autocorrelation of the component functions of S is given by*

$$r_{\overline{j} \cdot S}(\overline{\alpha}) = \sum_{\overline{v} \in \mathbb{F}_2^m} \mathsf{DDT}(\boldsymbol{\alpha}, \boldsymbol{v})(-1)^{\overline{j} \cdot \overline{v}}$$

for $\overline{\alpha} \in \mathbb{F}_2^n$ and $\overline{j} \in \mathbb{F}_2^m$.

Proof. See [37]. □

Corollary 3.5 (DDT and Undisturbed Bits)**.** *For a nonzero input difference $\overline{\alpha} \in \mathbb{F}_2^n$, the i-th bit of the output difference of S is undisturbed if and only if*

$$\sum_{\overline{v} \in \mathbb{F}_2^m} \mathsf{DDT}(\boldsymbol{\alpha}, \boldsymbol{v})(-1)^{\overline{e}_i \cdot \overline{v}} = \pm 2^n$$

for $i \in \{0, \ldots, m-1\}$ and $\overline{e}_i$ is the i-th standard basis of $\mathbb{F}_2^m$.

Proof. Suppose for a nonzero input difference $\overline{\alpha} \in \mathbb{F}_2^n$, the i-th bit of the output difference of S is undisturbed. From Lemma 3.3 we have $r_{h_i}(\overline{\alpha}) = \pm 2^n$. Since $r_{h_i}(\overline{\alpha}) = r_{\overline{e}_i \cdot S}(\overline{\alpha})$ it follows from Theorem 3.4 that $\sum_{\overline{v} \in \mathbb{F}_2^m} \mathsf{DDT}(\boldsymbol{\alpha}, \boldsymbol{v})(-1)^{\overline{e_i} \cdot \overline{v}} = \pm 2^n$. The converse can be trivially proved by reversing the previous steps. □

LAT, on the other hand, is used as a counterpart of DDT in the domain of linear cryptanalysis. For an $n \times m$ S-Box S, the entry of LAT of S at row $\overline{a} \in \mathbb{F}_2^n$ and column $\overline{b} \in \mathbb{F}_2^m$ is defined as $\mathsf{LAT}(\boldsymbol{a}, \boldsymbol{b}) = |\{\overline{x} \in \mathbb{F}_2^n \mid \overline{a} \cdot \overline{x} = \overline{b} \cdot S(\overline{x})\}| - 2^{n-1}$. Although undisturbed bits are useful in constructing truncated differentials for a bit-oriented cipher, one may also indicate the existence of undisturbed bits from LAT. We will use a well-known relation of LAT and the Walsh value of component functions of an S-Box in Lemma 3.6. Together with Theorem 2.8 (Wiener-Khintchine) and Lemma 3.3, the relation of LAT and undisturbed bits can be established. The main result is given in Theorem 3.7.

Lemma 3.6. *The relation between linear approximation table of S and the Walsh value of the component functions of S is given by*

$$\mathsf{LAT}(\boldsymbol{a}, \boldsymbol{b}) = \frac{1}{2} \mathcal{W}_{\overline{b} \cdot S}(\overline{a})$$

for $\overline{a} \in \mathbb{F}_2^n$ and $\overline{b} \in \mathbb{F}_2^m$.

Proof.

$$\begin{aligned} \mathsf{LAT}(a, b) &= |\{\overline{x} \in \mathbb{F}_2^n \mid \overline{a} \cdot \overline{x} = \overline{b} \cdot S(\overline{x})\}| - 2^{n-1} \\ &= |\{\overline{x} \in \mathbb{F}_2^n \mid l_{\overline{a}}(\overline{x}) = \overline{b} \cdot S(\overline{x})\}| - 2^{n-1} = (2^n - \mathrm{dt}(l_{\overline{a}}, \overline{b} \cdot S)) - 2^{n-1} \\ &= 2^{n-1} - \left(2^{n-1} - \frac{1}{2}\mathfrak{C}_{l_{\overline{a}}, \overline{b} \cdot S}\right) = \frac{1}{2}\mathfrak{C}_{l_{\overline{a}}, \overline{b} \cdot S} = \frac{1}{2}\mathcal{W}_{\overline{b} \cdot S}(\overline{a}) \end{aligned}$$

□

Theorem 3.7 (LAT and Undisturbed Bits). *For a nonzero input difference $\overline{\alpha} \in \mathbb{F}_2^n$, the i-th bit of the output difference of S is undisturbed if and only if*

$$2^{2-n} \sum_{\overline{a} \in \mathbb{F}_2^n} \mathsf{LAT}(\boldsymbol{a}, \boldsymbol{2^i})^2 (-1)^{\overline{\alpha} \cdot \overline{a}} = \pm 2^n$$

for $i \in \{0, \ldots, m-1\}$.

Proof. Firstly, we claim that $2^{2-n} \sum_{\overline{a} \in \mathbb{F}_2^n} \mathsf{LAT}(\boldsymbol{a}, \boldsymbol{b})^2 (-1)^{\overline{\alpha} \cdot \overline{a}} = r_{\overline{b} \cdot S}(\overline{\alpha})$. The proof of the claim is as follows:

$$\begin{aligned} & 2^{2-n}\sum_{\overline{a}\in\mathbb{F}_2^n}\mathsf{LAT}(\boldsymbol{a},\boldsymbol{b})^2(-1)^{\overline{\alpha}\cdot\overline{a}} & \\ = \; & 2^{-n}\sum_{\overline{a}\in\mathbb{F}_2^n}2^2\cdot\mathsf{LAT}(\boldsymbol{a},\boldsymbol{b})^2(-1)^{\overline{\alpha}\cdot\overline{a}} & \\ = \; & 2^{-n}\sum_{\overline{a}\in\mathbb{F}_2^n}(2\cdot\mathsf{LAT}(\boldsymbol{a},\boldsymbol{b}))^2(-1)^{\overline{\alpha}\cdot\overline{a}} & \\ = \; & 2^{-n}\sum_{\overline{a}\in\mathbb{F}_2^n}\mathcal{W}_{\overline{b}\cdot S}(\overline{a})^2(-1)^{\overline{\alpha}\cdot\overline{a}} & \text{from Lemma 3.6} \\ = \; & r_{\overline{b}\cdot S}(\overline{\alpha}) & \text{from Theorem 2.8} \end{aligned}$$

Clearly we have

$$2^{2-n}\sum_{\overline{a}\in\mathbb{F}_2^n}\mathsf{LAT}(\boldsymbol{a},\boldsymbol{2^i})^2(-1)^{\overline{\alpha}\cdot\overline{a}} = r_{\overline{e}_i\cdot S}(\overline{\alpha}) = r_{h_i}(\overline{\alpha}) = \pm 2^n$$

where $\overline{e}_i$ is the i-th standard basis of $\mathbb{F}_2^m$. Immediately from Lemma 3.3, for nonzero input difference $\overline{\alpha}$ the i-th bit of the output difference of S is undisturbed.

Conversely, if for a nonzero input difference $\overline{\alpha}$ the i-th bit of the output difference of S is undisturbed, Lemma 3.3 implies that $r_{h_i}(\overline{\alpha}) = \pm 2^n$. From our claim we can have $\pm 2^n = r_{\overline{e}_i\cdot S}(\overline{\alpha}) = 2^{2-n}\sum_{\overline{a}\in\mathbb{F}_2^n}\mathsf{LAT}(\boldsymbol{a},\boldsymbol{2^i})^2(-1)^{\overline{\alpha}\cdot\overline{a}}$. □

3.3 Autocorrelation Table

One way to check the existence of undisturbed bits in an S-Box is by taking a nonzero input difference and seeing whether there are some bits in all the corresponding output differences that remain invariant. This can be done by observing the DDT of an S-Box. However, this indirect approach can be improved if one is able to find a dedicated cryptanalytic tool for the case of undisturbed bits.

In this section, we extend the result from Lemma 3.3 and provide a tool called *autocorrelation table*, which was also appeared previously in [37]. Though it was introduced earlier, the application of autocorrelation table for cryptanalysis of block ciphers was not mentioned. We will show that an autocorrelation table is proven to be a more useful tool, compared to DDT, to check if an S-Box has undisturbed bits. Moreover, we will be able to obtain all nonzero input differences that have undisturbed bits in its corresponding output differences. Because undisturbed bits are also truncated differentials of probability one in an S-Box, autocorrelation table can be viewed as a counterpart of DDT in the domain of truncated differential cryptanalysis.

Definition 3.2 (Autocorrelation Table [37]). For $\overline{a} \in \mathbb{F}_2^n$ and $\overline{b} \in \mathbb{F}_2^m$, we define autocorrelation table of an S-Box S, denoted as ACT, where the entry in the row $\boldsymbol{a}$ and column $\boldsymbol{b}$ is equal to

$$\mathsf{ACT}(\boldsymbol{a},\boldsymbol{b}) = r_{\overline{b}\cdot S}(\overline{a})$$

Proposition 2.16 provides an equivalent description of an S-Box that has linear structures from the the autocorrelation of its component functions. Autocorrelation table can then be used to determine if an S-Box has linear structures.

Theorem 3.8. *An S-Box S has a linear structure if and only if there exists a nonzero $\overline{\alpha} \in \mathbb{F}_2^n$ and a nonzero $\overline{b} \in \mathbb{F}_2^m$ such that* $\mathsf{ACT}(\boldsymbol{\alpha}, \boldsymbol{b}) = \pm 2^n$.

Proof. This is an immediate consequence of Definition 2.6 and Proposition 2.16. □

Remark 3.2. Let $\overline{\alpha}$ be an input difference to S and let

$$\Omega_{\overline{\alpha}} = \{\overline{\beta} \in \mathbb{F}_2^m \mid \mathbf{Pr}_S[\overline{\alpha} \to \overline{\beta}] > 0\}$$

be the set of all possible output differences of S corresponding to the input difference $\overline{\alpha}$. If $\mathrm{ACT}(\boldsymbol{\alpha}, \boldsymbol{b}) = +2^n$ (resp. -2^n), for $\overline{b} \in \mathbb{F}_2^m$, then $\overline{b} \cdot \overline{\beta} = 0$ (resp. 1) for all $\overline{\beta} \in \Omega_{\overline{\alpha}}$.

To determine if an S-Box has undisturbed bits, it is sufficient to observe all nonzero row entries in each column of the autocorrelation table that correspond to the autocorrelation spectrum of coordinate functions of the S-Box, i.e. the column $\mathbf{2}^i$, $i \in \{0, \ldots, m-1\}$. The result is given as the following corollary.

Corollary 3.9. *For a nonzero input difference $\overline{\alpha}$, the i-th bit of the output difference of S is undisturbed if and only if* $\mathsf{ACT}(\boldsymbol{\alpha}, \mathbf{2}^i) = \pm 2^n$, *for* $i \in \{0, \ldots, m-1\}$.

Proof. From Theorem 3.1, the vector $\overline{\alpha}$ is a linear structure of the coordinate function h_i. Clearly this is a direct consequence of Theorem 3.8. □

The autocorrelation table of the S-Box of PRESENT is provided in Table 3.4. Some input differences that have undisturbed bits in its corresponding output differences can be observed in the column $\mathbf{1}$, which is the autocorrelation spectrum of the rightmost coordinate function. One may see in the row entries $\mathbf{1}$, $\mathbf{8}$, and $\mathbf{9}$ at column $\mathbf{1}$ have value $\pm 2^4 = \pm 16$. Note that the row index represents the input difference and the column index represents the component functions of the S-Box. The magnitude of the entry indicate the value of the undisturbed bits, where the sign "$+$" and "$-$" correspond to the undisturbed bit value equal to zero and one, respectively.

In Table 3.4 one may also find component functions, other than the coordinate functions, that have linear structures. For instance, the component functions in the S-Box of PRESENT represented by $\mathbf{10} \cdot S(\overline{x})$ and $\mathbf{11} \cdot S(\overline{x})$ have nontrivial linear structures (this can be seen in column $\mathbf{10}$ and $\mathbf{11}$ in Table 3.4 where some of the nonzero row entries are equal to $\pm 2^n$). The implication of this result was given in Remark 3.2. However, it remains unknown whether the existence of linear structures in the component functions of an S-Box other than the coordinate functions could improve or lead to a new approach in (truncated)-differential cryptanalysis of bit-oriented block ciphers. We leave it as an open problem.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
0	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
1	16	−16	0	0	0	0	0	0	0	0	−16	16	0	0	0	0
2	16	0	0	−8	−8	0	−8	8	0	−8	0	0	0	0	0	8
3	16	0	−8	0	0	−8	0	0	8	0	0	0	−8	−8	8	0
4	16	0	0	−8	−8	0	0	0	0	−8	0	0	−8	8	0	8
5	16	0	8	0	0	−8	−8	−8	−8	0	0	0	0	0	8	0
6	16	0	−8	8	0	0	0	0	−8	8	0	−16	0	0	0	0
7	16	0	0	0	0	0	8	−8	0	0	0	−16	8	−8	0	0
8	16	−16	−8	8	0	0	0	0	−8	8	0	0	0	0	0	0
9	16	16	0	0	−8	−8	0	0	0	0	0	0	0	0	−8	−8
10	16	0	0	−8	0	8	−8	8	0	−8	0	0	0	0	−8	0
11	16	0	8	0	8	0	0	0	−8	0	0	0	−8	−8	0	−8
12	16	0	0	−8	0	8	0	0	0	−8	0	0	−8	8	−8	0
13	16	0	−8	0	8	0	−8	−8	8	0	0	0	0	0	0	−8
14	16	0	0	0	0	0	0	0	0	0	−16	0	0	0	0	0
15	16	0	0	0	−8	−8	8	−8	0	0	16	0	8	−8	−8	−8

Table 3.4: Autocorrelation table of the S-Box of PRESENT.

Example 3.2. Let $\mathbf{1} = (0, 0, 0, 1)$ be an input difference to the PRESENT's S-Box. All the possible output differences correspond to input difference $\mathbf{1}$ are $\mathbf{3} = (0, 0, 1, 1)$, $\mathbf{7} = (0, 1, 1, 1)$, $\mathbf{9} = (1, 0, 0, 1)$, $\mathbf{13} = (1, 1, 0, 1)$ (See Table 3.3). The entry $\mathsf{ACT}(\mathbf{1}, \mathbf{11})$ in the autocorrelation table of PRESENT's S-Box is equal to 16. One can trivially check that $\mathbf{11} \cdot \overline{\beta} = (1, 0, 1, 1) \cdot \overline{\beta} = 0$ for all possible output differences $\overline{\beta}$ correspond to input difference $\mathbf{1}$.

3.4 Existence of S-Boxes with Undisturbed Bits

Recall from Theorem 3.1 that an S-Box has undisturbed bits if the derivative of any of its coordinate function at a nonzero vector in $\mathbb{F}_2^n$ is a constant function. The existence of an S-Box that has undisturbed bits can then be reduced into a question whether any of the coordinate functions of the S-Box has a nonzero linear structure.

So far the known Boolean functions that have nonzero linear structures are affine functions (from Proposition 2.11). If an S-Box has an affine coordinate function, then definitely the S-Box has undisturbed bits. However, this is unlikely to occur in real case. The reason is because it will lead to a linear approximation that involves input and output bits of the S-Box with probability one, and clearly does not serve the purpose of an S-Box as a nonlinear layer for block ciphers.

In order to find Boolean functions with linear structures, Proposition 2.9 restricts our attention to the Boolean functions of low degree. The following result is due to Carlet [7] and will be extended in Theorem 3.13 to show that an S-Box with at least one quadratic coordinate function has undisturbed bits. The main result is given in Lemma 3.12, and the proof depends on the results from Lemma 3.10 and Lemma 3.11.

Lemma 3.10 ([7])**.** *Let f be an n-variable Boolean function. We have the following relation:*

$$\mathcal{W}_f^2(\overline{0}) = \sum_{\overline{b} \in \mathbb{F}_2^n} \mathcal{W}_{D_{\overline{b}}f}(\overline{0}).$$

Proof.

$$\sum_{\overline{b}\in\mathbb{F}_2^n} \mathcal{W}_{D_{\overline{b}}f}(\overline{0}) = \sum_{\overline{b}\in\mathbb{F}_2^n}\left[\sum_{\overline{x}\in\mathbb{F}_2^n}(-1)^{D_{\overline{b}}f(\overline{x})}(-1)^{\overline{0}\cdot\overline{x}}\right] = \sum_{\overline{b}\in\mathbb{F}_2^n}\left[\sum_{\overline{x}\in\mathbb{F}_2^n}(-1)^{D_{\overline{b}}f(\overline{x})}\right]$$
$$= \sum_{\overline{b}\in\mathbb{F}_2^n} r_f(\overline{b}) = \sum_{\overline{b}\in\mathbb{F}_2^n} r_f(\overline{b})(-1)^{\overline{0}\cdot\overline{b}} = \mathcal{W}_f^2(\overline{0})$$

□

Lemma 3.11 ([7]). *If f is an n-variable Boolean function with* $\deg(f) = 2$ *then*

$$\mathcal{W}_f^2(\overline{0}) = 2^n \sum_{\overline{b}\in\mathcal{LS}_f}(-1)^{D_{\overline{b}}f(\overline{0})}.$$

Proof. Since the degree of f is equal to 2, it follows from Proposition 2.9 that for every $\overline{b}\in\mathbb{F}_2^n$ we have $\deg(D_{\overline{b}}f)\le 1$. Clearly $D_{\overline{b}}f$ is affine, hence from Corollary 2.4 it is either balanced (for nonzero coefficient vector) or constant function (for zero coefficient vector). Consequently, for the case where $D_{\overline{b}}f$ is balanced, we have $\mathcal{W}_{D_{\overline{b}}f}(\overline{0}) = 0$ from Proposition 2.7. Using the result from the Lemma 3.10, then

$$\mathcal{W}_f^2(\overline{0}) = \sum_{\overline{b}\in\mathbb{F}_2^n}\mathcal{W}_{D_{\overline{b}}f}(\overline{0}) = \sum_{\overline{b}\in\mathcal{LS}_f}\mathcal{W}_{D_{\overline{b}}f}(\overline{0}) = \sum_{\overline{b}\in\mathcal{LS}_f}\left[\sum_{\overline{x}\in\mathbb{F}_2^n}(-1)^{D_{\overline{b}}f(\overline{x})}\right]$$
$$= 2^n\sum_{\overline{b}\in\mathcal{LS}_f}(-1)^{D_{\overline{b}}f(\overline{0})}$$

□

Lemma 3.12 ([7]). *If f is a balanced n-variable Boolean function with* $\deg(f) = 2$, *then there exists a nonzero $\overline{\alpha}\in\mathbb{F}_2^n$ such that $D_{\overline{\alpha}}f(\overline{x}) = f(\overline{x})\oplus f(\overline{x}\oplus\overline{\alpha}) = 1$ for all $\overline{x}\in\mathbb{F}_2^n$.*

Proof. Let f be a balanced n-variable Boolean function with $\deg(f) = 2$. Since f is balanced, then $\mathcal{W}_f(\overline{0}) = 0$ and consequently $\mathcal{W}_f^2(\overline{0}) = 0$. The result from Lemma 3.11 implies that the sum $\sum_{\overline{b}\in\mathcal{LS}_f}(-1)^{D_{\overline{b}}f(\overline{0})}$ must be equal to zero. We know that the zero vector $\overline{0}\in\mathbb{F}_2^n$ is a trivial linear structure because $D_{\overline{0}}f(\overline{x}) = 0$ for all $\overline{x}\in\mathbb{F}_2^n$. Clearly $\overline{0}\in\mathcal{LS}_f$. Using the existence of zero vector in the set of linear structures of f, then there must exist a vector $\overline{\alpha}\in\mathbb{F}_2^n$, $\overline{\alpha}\ne\overline{0}$ such that $D_{\overline{\alpha}}f(\overline{x}) = 1$ for all $\overline{x}\in\mathbb{F}_2^n$. □

Remark 3.3. Another result that we can conclude from Lemma 3.12 is that, any linear structure of a balanced quadratic Boolean function comes in pairs. For every linear structure $\overline{\beta}\in\mathbb{F}_2^n$ of f such that $D_{\overline{\beta}}f(\overline{x}) = 0$, there exists a $\overline{\beta'}\in\mathbb{F}_2^n$, $\overline{\beta'}\ne\overline{\beta}$ such that $D_{\overline{\beta'}}f(\overline{x}) = 1$. So the total number of linear structures of a balanced quadratic Boolean functions is always even (multiple of 2).

Theorem 3.13. *Let S be a balanced $n \times m$ S-Box and let $h_{m-1}, \ldots, h_0$ be its coordinate functions. If there exists a coordinate function h_i with $\deg(h_i) = 2$ then the S-Box S has undisturbed bits. More precisely, there exists a nonzero $\overline{\alpha} \in \mathbb{F}_2^n$ such that for input difference $\overline{\alpha}$, the i-th bit of the output difference of S is undisturbed and its value is* 1.

Proof. From Proposition 2.15, for every nonzero $\overline{b} \in \mathbb{F}_2^m$ all the component functions $\overline{b} \cdot S(\overline{x})$ are balanced Boolean functions, including the coordinate functions of S. If there exists a coordinate function h_i with $\deg(h_i) = 2$, Lemma 3.12 says that there is a nonzero $\overline{\alpha} \in \mathbb{F}_2^n$ such that $D_{\overline{\alpha}} h_i(\overline{x}) = 1$ for all $\overline{x} \in \mathbb{F}_2^n$. Theorem 3.1 implies that for the input difference $\overline{\alpha}$, the i-th bit of the output difference of S is undisturbed and its value is 1. □

Corollary 3.14. *If S is a balanced $n \times m$ S-Box with $n = 3$, then S has undisturbed bits. Moreover, for every $i \in \{0, \ldots, m-1\}$ there exists a nonzero $\overline{\alpha} \in \mathbb{F}_2^n$ such that for input difference $\overline{\alpha}$, the i-th bit of the output difference of S is undisturbed and its value is* 1.

Proof. Since S is a balanced S-Box, based on Proposition 2.2 then $\deg(\overline{b} \cdot S) \leq 2$ for all nonzero $\overline{b} \in \mathbb{F}_2^m$. It follows that every coordinate function of S is of degree ≤ 2. The result follows immediately from Theorem 3.13 and Proposition 2.11. □

In [34] the author stated that every bijective 3×3 S-Box has undisturbed bits. The alternative proof we provide below can be seen as an immediate implication from Corollary 3.14.

Corollary 3.15 ([34])**.** *Every 3×3 bijective S-Box has undisturbed bits.*

Proof. Since bijective 3×3 S-Boxes are balanced S-Boxes, the result follows from Corollary 3.14. □

3.5 Conclusions

In this chapter we defined the notion of undisturbed bits and their properties. In the beginning, we showed that an S-Box which has undisturbed bits is related with the existence of a nonzero linear structures in its coordinate functions. We recalled previous work on the S-Boxes with linear structures and showed that S-Boxes with undisturbed bits can be treated as a special class of S-Boxes with linear structures by only considering the nonzero linear structures in its coordinate functions.

We also established relations between undisturbed bits in an S-Box with its DDT and LAT. In addition to that, we proposed autocorrelation table as a tool to obtain all nonzero input differences to an S-Box that may yield some bits in the corresponding output differences undisturbed. Since the existence of undisturbed bits is useful to construct truncated differentials for a block cipher, autocorrelation table can be seen as a counterpart of DDT in the domain of truncated differential cryptanalysis.

The main result in Section 3.4 is that a balanced $n \times m$ S-Box with a quadratic coordinate function will definitely has undisturbed bits. This general result for $n \times m$ S-Boxes is used to provide an alternative proof for the proposition on the existence of undisturbed bits for every bijective 3×3 S-Boxes.

CHAPTER 4

CONCLUSIONS AND OPEN PROBLEMS

Cryptanalysis of block ciphers remain as a challenging task since the design of a secure block cipher is already well-understood. The security of a block cipher relies heavily on the quality of its nonlinear layer, in particular the substitution box. This chapter concludes our work and describes some possible open problems.

4.1 Conclusions

The main contributions of our work are given in Chapter 3. In the beginning, the first step we took was to formally define the notion of undisturbed bits. The structure of an S-Box can be decomposed into coordinate functions. Any output vector from an S-Box can be computed component-wisely using the coordinate functions. Similarly, for any output difference vector corresponding to a specific input difference, the components can also be computed in similar fashion. This observation leads us to put our first result that the existence of a nonzero input difference that yields undisturbed bits in an S-Box is related to the existence of a nonzero linear structure in the coordinate functions of the S-Box. We showed that an S-Box that has undisturbed bits belongs to a special class of S-Boxes with linear structures, by only considering nonzero linear structures in its coordinate functions.

The relation of an S-Box with undisturbed bits can also be characterized from its difference distribution table and linear approximation table. We established this result from the fact that if $\overline{\alpha}$ is a linear structure of an n-variable Boolean function, then autocorrelation of the function at $\overline{\alpha}$ is equal to $\pm 2^n$. Autocorrelation table of an S-Box, that essentially examines autocorrelation spectrums of each component function of the S-Box, can then be used as a tool to observe whether the S-Box has undisturbed bits. Even though the main concern is the autocorrelation spectrums of coordinate functions of an S-Box, autocorrelation spectrums of other component functions of the S-Box are also defined in autocorrelation table for the sake of completeness.

The last result of this thesis is that a balanced S-Box has undisturbed bits if it has a quadratic coordinate function. An alternative proof that every 3×3 bijective S-Box has undisturbed bits is an immediate consequence of this theorem.

4.2 Open Problems

We suggest some possible open problems and directions that may be useful for future research on undisturbed bits.

1. Chaum and Evertse introduced the concept of linear structures in a block cipher [8] [13] earlier before Biham and Shamir proposed differential cryptanalysis technique to attack block ciphers [3]. Knudsen's proposal for truncated differential cryptanalysis was published in 1994 [16]. Generalizing the notion of undisturbed bits for a block cipher and establishing its relation with (truncated)-differential cryptanalysis will give a better understanding on how these three different concepts are related to each other.
2. Definition of an autocorrelation table includes the autocorrelation spectrum of all component functions of an S-Box. While the notion of undisturbed bits is related to the existence of nonzero linear structures in the coordinate functions of an S-Box, one may also find other component functions of the S-Box which may have nonzero linear structures. It remains unknown whether this property in an S-Box could improve or lead to a new approach in cryptanalysis of bit-oriented block ciphers.
3. S-Boxes used in symmetric key encryptions can be randomly generated or constructed using mathematical functions satisfying various cryptographic properties. For example, AES' S-Box is constructed using composition of inversion in finite field $\mathbb{F}_{2^8}$ with an affine transformation. This S-Box has high nonlinearity as well as low differential uniformity. One possible open problem is to study some method for S-Boxes construction and see whether it will eventually yield an S-Box with undisturbed bits.

REFERENCES

[1] E. Biham, A. Biryukov, and A. Shamir, Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials, in J. Stern, editor, *EUROCRYPT*, volume 1592 of *Lecture Notes in Computer Science*, pp. 12–23, Springer, 1999, ISBN 3-540-65889-0.

[2] E. Biham, A. Biryukov, and A. Shamir, Miss in the middle attacks on IDEA and Khufu, in Knudsen [18], pp. 124–138.

[3] E. Biham and A. Shamir, Differential cryptanalysis of DES-like cryptosystems, in A. Menezes and S. A. Vanstone, editors, *CRYPTO*, volume 537 of *Lecture Notes in Computer Science*, pp. 2–21, Springer, 1990, ISBN 3-540-54508-5.

[4] E. Biham and A. Shamir, Differential cryptanalysis of DES-like cryptosystems, J. Cryptology, 4(1), pp. 3–72, 1991.

[5] A. Bogdanov, L. Knudsen, G. Leander, C. Paar, A. Poschmann, M. Robshaw, Y. Seurin, and C. Vikkelsoe, PRESENT: An ultra-lightweight block cipher, in P. Paillier and I. Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007*, volume 4727 of *Lecture Notes in Computer Science*, pp. 450–466, Springer Berlin Heidelberg, 2007.

[6] C. Carlet, *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, chapter Vectorial Boolean Functions for Cryptography, pp. 398–469, Cambridge University Press, 2010.

[7] C. Carlet, *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, chapter Boolean Functions for Cryptography and Error Correcting Codes, pp. 257–397, Cambridge University Press, 2010.

[8] D. Chaum and J.-H. Evertse, Crytanalysis of DES with a reduced number of rounds: Sequences of linear factors in block ciphers, in H. C. Williams, editor, *CRYPTO*, volume 218 of *Lecture Notes in Computer Science*, pp. 192–211, Springer, 1985, ISBN 3-540-16463-4.

[9] J. Y. Cho, Linear cryptanalysis of reduced-round PRESENT, in J. Pieprzyk, editor, *CT-RSA*, volume 5985 of *Lecture Notes in Computer Science*, pp. 302–317, Springer, 2010, ISBN 978-3-642-11924-8.

[10] T. W. Cusick and P. Stănică, *Cryptographic Boolean Functions and Applications*, Elsevier, 2009.

[11] J. Daemen and V. Rijmen, The block cipher Rijndael, in J.-J. Quisquater and B. Schneier, editors, *CARDIS*, volume 1820 of *Lecture Notes in Computer Science*, pp. 277–284, Springer, 1998, ISBN 3-540-67923-5.

[12] P. E. Dunner, *The Complexity of Boolean Networks*, APIC Studies in Data Processing No. 29, Academic Press, 1988.

[13] J.-H. Evertse, Linear structures in blockciphers, in D. Chaum and W. L. Price, editors, *EUROCRYPT*, volume 304 of *Lecture Notes in Computer Science*, pp. 249–266, Springer, 1987, ISBN 3-540-19102-X.

[14] S. W. Golomb, *Shift Register Sequences*, Aegean Park Press, Laguna Hills, CA, USA, 1981, ISBN 0894120484.

[15] A. Kerckhoffs, La cryptographie militaire, Journal des Sciences Militaires, IX, pp. 161–191, February 1883.

[16] L. R. Knudsen, Truncated and higher order differentials, in Preneel [26], pp. 196–211.

[17] L. R. Knudsen, DEAL–a 128-bit block cipher, Technical Report 151, Department of Informatics, University of Bergen, Norway, 1998.

[18] L. R. Knudsen, editor, *Fast Software Encryption, 6th International Workshop, FSE '99, Rome, Italy, March 24-26, 1999 Proceedings*, volume 1636 of *Lecture Notes in Computer Science*, Springer, 1999.

[19] L. R. Knudsen and T. A. Berson, Truncated differentials of SAFER, in D. Gollman, editor, *FSE*, volume 1039 of *Lecture Notes in Computer Science*, pp. 15–26, Springer, 1996.

[20] X. Lai, Additive and linear structures of cryptographic functions, in Preneel [26], pp. 75–85.

[21] X. Lai, Higher order derivatives and differential cryptanalysis, in R. Blahut, J. Costello, DanielJ., U. Maurer, and T. Mittelholzer, editors, *Communications and Cryptography*, volume 276 of *The Springer International Series in Engineering and Computer Science*, pp. 227–233, Springer US, 1994, ISBN 978-1-4613-6159-6.

[22] H. Mala, M. Dakhilalian, and M. Shakiba, Cryptanalysis of block ciphers using almost-impossible differentials, IACR Cryptology ePrint Archive, 2010, p. 485, 2010.

[23] M. Matsui, Linear cryptoanalysis method for DES cipher, in T. Helleseth, editor, *EUROCRYPT*, volume 765 of *Lecture Notes in Computer Science*, pp. 386–397, Springer, 1993, ISBN 3-540-57600-2.

[24] W. Meier and O. Staffelbach, Nonlinearity criteria for cryptographic functions, in J.-J. Quisquater and J. Vandewalle, editors, *EUROCRYPT*, volume 434 of *Lecture Notes in Computer Science*, pp. 549–562, Springer, 1989, ISBN 3-540-53433-4.

[25] B. Preneel, *Analysis and Design of Cryptographic Hash Functions*, Ph.D. thesis, Katholieke Universiteit Leuven, 1993, René Govaerts and Joos Vandewalle (promotors).

[26] B. Preneel, editor, *Fast Software Encryption: Second International Workshop. Leuven, Belgium, 14-16 December 1994, Proceedings*, volume 1008 of *Lecture Notes in Computer Science*, Springer, 1995.

[27] R. L. Rivest, *Handbook of Theoretical Computer Science*, volume A, chapter 13, Elsevier, 1990.

[28] R. A. Rueppel, *Analysis and Design of Stream Ciphers*, Springer-Verlag New York, Inc., New York, NY, USA, 1986, ISBN 0-387-16870-2.

[29] P. Sarkar and S. Maitra, Construction of nonlinear boolean functions with important cryptographic properties, in B. Preneel, editor, *EUROCRYPT*, volume 1807 of *Lecture Notes in Computer Science*, pp. 485–506, Springer, 2000, ISBN 3-540-67517-5.

[30] C. E. Shannon, Communication theory of secrecy systems, Bell System Technical Journal, (28), pp. 656–715, 1949.

[31] A. Sorkin, Lucifer, a cryptographic algorithm, Cryptologia, 8(1), pp. 22–42, 1984.

[32] S. Sun, L. Hu, and P. Wang, Automatic security evaluation for bit-oriented block ciphers in related-key model : Application to PRESENT-80, LBlock, and others, IACR Cryptology ePrint Archive, 2013, p. 676, 2013.

[33] C. Tezcan, The improbable differential attack: Cryptanalysis of reduced round CLEFIA, in G. Gong and K. C. Gupta, editors, *INDOCRYPT*, volume 6498 of *Lecture Notes in Computer Science*, pp. 197–209, Springer, 2010, ISBN 978-3-642-17400-1.

[34] C. Tezcan, Improbable differential attacks on PRESENT using undisturbed bits, Journal of Computational and Applied Mathematics, 259, Part B(0), pp. 503 – 511, 2014, ISSN 0377-0427.

[35] D. Wagner, The boomerang attack, in Knudsen [18], pp. 156–170.

[36] W. Zhang, Z. Bao, D. Lin, V. Rijmen, B. Yang, and I. Verbauwhede, RECTANGLE: A bit-slice ultra-lightweight block cipher suitable for multiple platforms, IACR Cryptology ePrint Archive, 2014, p. 84, 2014.

[37] X.-M. Zhang, Y. Zheng, and H. Imai, Relating differential distribution tables to other properties of substitution boxes, Des. Codes Cryptography, 19(1), pp. 45–63, 2000.