# A MATHEMATICAL FRAMEWORK FOR COMBINING ERROR CORRECTION AND ENCRYPTION

by

Chetan N. Mathur

A DISSERTATION

Submitted to the Faculty of the Stevens Institute of Technology
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chetan N. Mathur, Candidate

ADVISORY COMMITTEE

Dr. K.P. Subbalakshmi, Chair Date

Dr. R. Chandramouli Date

Dr. Yu-Dong Yao Date

Dr. Barry Bunin Date

Dr. Zhenqi Zhu Date

STEVENS INSTITUTE OF TECHNOLOGY
Castle Point on Hudson,
Hoboken, NJ 07030
2007

A Mathematical Framework for Combining Error Correction and Encryption

## Abstract

Error resilience and energy efficiency are two main challenges facing block ciphers in noisy and resource constrained wireless environments. Traditionally, error correcting codes are used to recover from channel induced errors. However, this two step operation: encryption followed by error correction adds extra burden on an already resource constrained environment. Combining the two operations into one primitive has the potential to achieve efficient error correcting ciphers. However, if such a joint system is not designed carefully both error correcting capacity and security could be compromised. For this reason, the design of error correcting ciphers has remained an open problem for the past 25 years.

In this work, we propose two error correcting block ciphers: the High Diffusion (HD) cipher and the Pyramid cipher. Both ciphers use our recently proposed HD codes in the diffusion layer. The HD cipher is a ten round cipher which uses many small HD codes, whereas the Pyramid cipher is a five round cipher which uses a single large HD code. We show that the Pyramid cipher is as secure as the Advanced Encryption Standard (AES) against linear, differential and square attacks. We derive bounds on the error correcting capacity of the proposed ciphers and through simulations show that they are as error resilient as the Reed Solomon codes, outperform the convolution codes by 60% and are 10% more energy efficient compared to the traditional systems. We show that in stream modes our ciphers have higher encryption throughput compared to the AES. Energy analysis verifies that the HD cipher in stream mode is 40% more energy efficient compared to the AES.

Author: Chetan N. Mathur
Advisor: Dr. K.P. Subbalakshmi
Degree: Doctor of Philosophy
Department of Electrical and Computer Engineering
April 10, 2006

# Acknowledgements

I would like to thank Professor K.P. Subbalakshmi, my thesis supervisor, for her mentorship and constant support during this research. I am also thankful to Professor R. Chandramouli for the support provided and his many suggestions and ideas. I thank Prof. Yu-Dong Yao, Prof. Barry Bunin and Prof. Zhenqi Zhu, my dissertation committee members for providing valuable suggestions and criticisms on the dissertation proposal and during the completion of the dissertation. I also thank my colleagues Dr. M. Haleem and Dr. Yiping Xing for the extensive collaboration in research activities. I thank my cousin Karthik Narayan, with whom I had the opportunity to work with during his study at Stevens. Some parts of this dissertation includes the joint work carried out with him. I thank Dr. Goce Jakimoski for his valuable insights into the area of block cipher cryptanalysis.

The research projects carried out at Stevens Institute of Technology during my PhD program were supported by grants from National Science Foundation, US Army Picatinny Arsenal and WiNSec. The completion of this dissertation may not have been a reality without these supports. I also thank the Department of Electrical and Computer Engineering for the lab equipment and facilities made available to me during the course of my study here. I thank my colleagues at the MSYNC lab with whom I share all the happy memories over the last five years.

Beyond all I am grateful to my loving parents, my caring wife and my adorable brothers for their encouragement and *love* during my efforts to successfully complete the program. My wife helped me prepare for my presentations and proofread some of the thesis chapters. Without her support and patience, this work would never have come into existence.

Chetan N. Mathur

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

We are increasingly relying on wireless mobile devices for our day to day transactions and commercial applications. Wireless devices like personal data assistants (PDAs) are used to execute online transactions and store valuable data such as credit card numbers. Hence, wireless communication security has gained importance in recent years. The mobile nature of the wireless devices make them dependent on battery power which is a constantly depleting resource. Unlike the wired counterparts, the wireless medium is physically unprotected and can be extremely noisy and bursty. To protect wireless transmissions, security protocols which were traditionally applied at the upper layers like application and transport layer are now applied at the lower layers as well. For example, security protocols like the Wired Equivalent Privacy (WEP), Temporal Key Integrity Protocol [65] and the Wifi Protected Access (WPA) [1] are now employed in the link layer. However, the application of encryption at link layer creates other issues. For example, encryption and decryption increase transmission delay and hence causes frequent timeouts in the upper layers. The error sensitivity of decryption operation triggers retransmissions and decreases the transmission throughput. The sensitivity of ciphers towards channel errors is due to the phenomenon called the avalanche effect [16]. This causes one or more bit errors before

decryption to spread randomly to the entire cipherstate with in few round. To illustrate the error expansion due to avalanche effect we plot the post decryption bit error rate for various block lengths of a generic block cipher over a range of channel bit error rates in Fig. 1.1. We can observe from the figure that for most of the practical channel conditions the avalanche effect causes a significant error expansion and this effect increases with the block length.

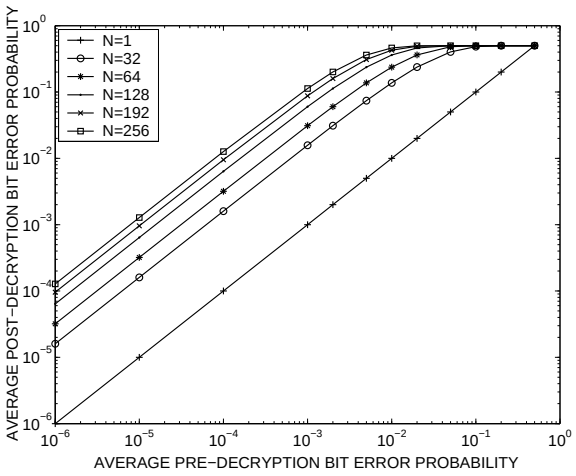


Figure 1.1: Error expansion due to avalanche effect

The most widely used technique to combat error propagation in block ciphers is to use them in stream modes. This is because, when a block cipher is operated in any of the stream modes, the plaintext is XORed with key stream generated by the block cipher. As XOR-ing is an error preserving [63] operation, there is no error propagation in stream modes. When block ciphers need to be used in block modes, they are concatenated with forward error correction (FEC) codes to minimize the number of retransmissions due to error propagation. The use of error correction codes however requires allocation of additional computational resources and transmission power which strains an already resource constrained environment. In order to con-

serve battery power, newer security protocols [53][43][44][54] are being developed that are more light weight and energy efficient. Some of the common techniques that have been used to reduce the energy consumed by cryptographic primitives are: reduction in number of rounds, use of simple operations (e.g. XORs and shifts), merging multiple operations, use of lookup tables, reduced block length, etc. However the light weight security protocols if not designed carefully could cause compromise in security [9] [34]. We identify energy efficiency, error resilience and speed of the underlying cryptographic primitive are the key factors that need to be jointly addressed by any wireless security protocol or primitive. The LEX cipher proposed in [8] achieves the energy efficiency and speed by converting the Rijndael block cipher [15] into stream cipher using an unconventional approach of leaking intermediate cipherstate bytes. Although this approach appears to be secure, leaking intermediate state information is known to cause weakness in the cipher [23]. A security-throughput optimization approach is proposed in [48] that utilizes the flexible block length in certain block ciphers to maximize the error resilience. Although this approach achieves error resilience, the energy consumption due to context switching between different block lengths could be potentially high. An alternate approach for jointly achieving energy efficiency and error resilience in a cipher is to combine error correction and encryption. [46] was the first to propose a public key cipher based on algebraic coding theory. The security of this system is based on the hardness of the decoding problem [3]. In order to achieve meaningful security against present day adversaries, the parameters of this system have to be very large, making it infeasible in practice. This work was followed by a series of improvements and attacks [26][52] [35][5][22][62]. However, none of these systems were based on modern cryptographic design principles and they compromised security for error resilience. Cryptocoding [21] is one

of the more recently proposed techniques for joint error correction and encryption. This technique is based on quasigroup (Latin square) string transformation. Here, the large space of quasigroups translates to a large key space. In order to achieve error resilience, the data is padded with zeros before encryption and the decryption algorithm literately corrects errors until the padded zeros are correctly recovered. This makes the decoding procedure extremely complicated and hence cannot be used in practice. The difficulty in designing error correcting ciphers arise from the fact that error correction and encryption work at cross purposes to each other. Codes add redundancy while ciphers remove redundancy and randomize the source. Codes are usually linear whereas ciphers have to be non-linear. We approach the problem of using codes in ciphers by observing the similarities between codes and ciphers. Specifically we concentrate on the property of diffusion, which is exhibited by block codes and required by block ciphers to spread the non-linearity. Most of the modern block ciphers including the AES finalists like Rijndael [15] and Two-Fish [56] derive their diffusion transformations from Maximum Distance Separable (MDS) codes [10]. However, the generation of diffusion matrices in these approaches are ad-hoc, rely on brute force search and are not intended to make the block cipher error resilience. Also, using arbitrary FECs in the diffusion layer of block ciphers may render the cipher insecure and achieve sub-optimal error resilience. In this work, we provide a *mathematical frame work to combine error correction with encryption that maximizes the security and the error resilience of the cipher.* We use a specific class of channel codes called High Diffusion codes [41][49] that we recently proposed. These codes possess the branch number property [15] required by the diffusion layer of a block cipher and their burst error correction capability is well suited for wireless environments. We use the HD codes to build two error correcting block ciphers that we call the HD cipher

and the Pyramid cipher. The HD cipher is a ten round block cipher with a coding rate of $\frac{128}{288}$ whereas the Pyramid cipher is a five round block cipher with a coding rate of $\frac{128}{192}$. We show that the both the HD and the Pyramid ciphers are as secure as the popular Advanced Encryption Standard (AES) cipher [17] against the well known attacks. Based on the minimum distance decoding we show that the error correcting capacity of the HD cipher and the Pyramid cipher used in block modes [59] is seven and four bytes per block respectively. To evaluate the energy requirement of our proposed ciphers we setup a testbed consisting of 32 bit processors that have comparable architecture to most of the wireless devises. We compare the energy consumption of the Pyramid cipher with a traditional concatenated system comprising of AES cipher followed by a Reed Solomon (RS) codes that match the error correcting capacity of the HD and the Pyramid ciphers. Experimental results reveal that the HD cipher and Pyramid cipher encryption operations are 30% and 10% more energy efficient compared to the encryption-encoding operations of the concatenated system. Whereas, HD and Pyramid decryptions are 12% and 6% more energy efficient compared to the decoding-decryption operation of the concatenated system. We also evaluate the error resilience of the proposed ciphers under various channel models. Simulation results reveal that under wireless like channel conditions, the error resilience of the HD and Pyramid ciphers is equivalent to that of the concatenated system. We then implement the proposed ciphers in the counter mode. Due to the expansion of the cipherstate in HD and Pyramid, they have higher encryption throughput compared to the AES cipher. Moreover, the encryption throughput of these ciphers can be further increased by decreasing the coding rate of the HD codes used in these ciphers. Also, in the counter mode, like other secure block ciphers, the HD and the Pyramid ciphers act as pseudorandom number generators. To test the quality of random numbers generated

by the proposed ciphers, we subjected them to the National Institute of Standards and Technology (NIST) recommended DIEHARD statistical test suite. Pseudorandom tests reveal that both HD and the Pyramid are cryptographically sound random number generators. We propose to replace the AES cipher in the counter mode with cipher block chaining protocol (CCMP) (see Section 4.4.2) that is used in the current IEEE 802.11i standard with the HD cipher. The higher encryption throughput of the HD cipher translates to 40% improvement in energy efficiency of our HD-CCMP protocol.

# Chapter 2

# Background and Related Work

In this chapter, we give a brief background on block ciphers, their properties and the design strategies. We also briefly introduce the Advance Encryption Standard (AES) which is the current standard on block ciphers. We then describe the classification of error correcting codes and give a brief description on linear block codes. Then we discuss some notable previous work in the area of joint error correction and encryption.

## 2.1 Classification of Ciphers

Cipher is the term used to describe algorithms/techniques that have two distinct functions: encryption and decryption. Encryption is a process of scrambling (enciphering) information, the plaintext, using some secret knowledge, the secret key, into unintelligible form, the ciphertext. Decryption is the inverse process of encryption, where the ciphertext is unscrambled (deciphered) back to plaintext using the same or a different secret key. Ciphers for which the encryption secret key is different from the decryption secret key are called public ciphers. The RSA, Elgamal, Elliptic curves [60] are some of the commonly used public key ciphers. On the other hand, ciphers which have the same encryption and decryption keys are called private key or symmetric key ciphers. The Data Encryption Standard (DES), RC4 and Advanced Encryption

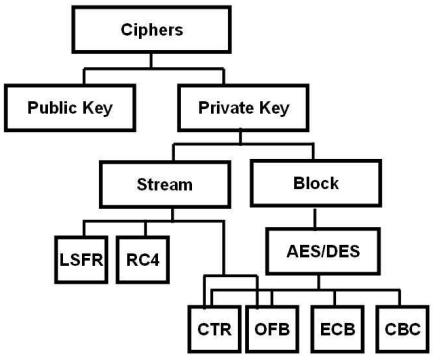


Figure 2.1: Classification of Ciphers.

Standard (AES) are some of the commonly used symmetric ciphers. Symmetric ciphers can encrypt the plaintext one bit or a block (more than a bit) at a time. The sub-class of symmetric ciphers that encrypt the plaintext one bit at a time are called stream ciphers. For example, the RC4 cipher is a stream cipher. Symmetric ciphers that encrypt the plaintext a block at a time are called block ciphers. For example, the AES block cipher encrypts 128 bits of the plaintext in one encryption. Block ciphers have many modes of operation. The most commonly used block modes are the electronic code book mode (ECB) and the cipher block chaining mode (CBC). However, block ciphers can also be used to encrypt the plaintext one bit at a time by employing them in the stream modes. The counter mode (CTR) and the output feed back mode (OFB) are some of the standard stream modes for any block cipher. Fig. 2.1 summarizes the classification of ciphers.

## 2.2 Block Ciphers

Block ciphers encrypt the plaintext one block (more than 1 bit) at a time. In most of the modern block ciphers a single function $F$, which may not be secure by itself, is repeated a number of times, until the desired level of security is achieved. Such

ciphers are called iterated block ciphers.

The security provided by any cipher can be measured in terms of the plaintext, $\mathcal{P}$, the ciphertext, $\mathcal{C}$ and the key, $\mathcal{K}$, used. The, conditional entropy $H(\mathcal{K}|\mathcal{C})$, called *key equivocation*, is the measure of how much information about the key is revealed by the ciphertext in case of ciphertext only attack [60]. This is given by,

$$H(\mathcal{K}|\mathcal{C}) = H(\mathcal{K}) + H(\mathcal{P}) - H(\mathcal{C})$$

where $H$ is entropy function. However, most of the modern day block ciphers are secure against this type of attack. There are other kinds of attacks possible such as known plaintext, chosen plaintext and chosen ciphertext attacks. Most of the practical cryptanalysis techniques are a combination of the above attacks. For example, the differential cryptanalysis [7] is a chosen plaintext attack and the linear cryptanalysis [45] is a known plaintext attack. Any new block cipher is believed to be secure, if it is computationally infeasible to derive the secret key under all known types of cryptanalysis. Further, researchers over the years have identified important properties that can be used to gauge the security of cryptographic algorithms. We discuss these properties in the next Section.

### 2.2.1 Properties of Block Ciphers

Claude Shannon in 1949 described the general setting for treating cryptosystems in the seminal work 'Communications Theory of Secrecy Systems' [58]. Here, he suggests two properties diffusion and confusion, as essentials for the design of ciphers. These properties are relevant even today and almost all of the modern day block ciphers exhibit these properties. A brief description of diffusion and confusion follows.

### Diffusion

The property of diffusion implies that the statistical structure of the message space, which leads to its redundancy, is dissipated into long range statistics of the ciphertext. It is a *quantitative* notion in the sense that dependency on each plaintext and key bit is to be spread to several ciphertext bits. This makes the relation between the plaintext and ciphertext as complex as possible. Diffusion is usually achieved by repeated permutations. The part of the round function $F$ that achieves diffusion is called the *diffusion layer*. Efficiency of the cipher is also affected by its diffusion properties, that is, if diffusion can be achieved in fewer operations we would require fewer rounds to achieve security. Moreover there are cryptanalysis techniques that exploit slow diffusion of ciphers, for example are differential cryptanalysis of the DES [7] exploits the slow diffusion in the Feistel structure [16].

### Confusion

This is more of a *qualitative* concept, in the sense that a non-linear relationship is to be expected between all ciphertext bits and plaintext/key bits. The property of confusion makes the relation between the ciphertext key space complex. Confusion is provided in block ciphers by substitution boxes (S-boxes). These components are the main non-linear operators in most block ciphers. An S-box fulfils the criteria for confusion if every bit in its output depends non-linearly on each input bit and vice versa (for invertible S-boxes). If this is not the case then the bias in the S-boxes may be exploited to break the cipher. For example, the linear cryptanalysis gathers information about the key by first finding approximate linear expressions for S-boxes and then extending them to the whole cipher. Therefore design of S-boxes is crucial to maintain the security of block ciphers.

**Avalanche Effect**

The term 'avalanche' comes from Feistel [16], and refers to the observed property of difference propagation with respect to a tiny change in the input. The change of a single input bit generally produces multiple bit-changes after one round, many more bit-changes after the second round, and so on, until about half of the block will randomly change. In any block cipher we would want a single bit to affect every output bit: if a single bit is flipped we would want half the bits in the output to be flipped (diffusion) independent of the position of the bits (confusion). Therefore avalanche in a cipher has been used widely as criterion to study cryptographic functions. In the next section, we give an overview of two basic strategies used to design block ciphers and how they cause and spread the avalanche effect.

### 2.2.2 Prominent Block Cipher Design Strategies

The two most prominent block cipher design strategies are the Feistel structure and the wide trail strategy. In this section, we briefly introduce both these strategies.

**Feistel structure**

This is one of the most widely used block cipher design strategies. Block ciphers like the DES, Blowfish, RC5 and FEAL [47] are all based on this structure. In the Feistel structure, the plaintext in each round $i$, is treated as two separate halves ($X_i = X_i^L || X_i^R$), left and right. In each round, only one half of the input cipherstate is operated upon. Any non-linear F-function which non-linearly transforms half of the plaintext using key bits can be used in the Feistel structure. That is, $F : \{0,1\}^{n/2} \times \{0,1\}^m \rightarrow \{0,1\}^{n/2}$, where $m$ is the number key bits. The round function in the Feistel structure can be described as follows, $X_{i+1} = (F_{k_i}(X_i^L) \oplus X_i^R)||X_i^L$. Where $k_i$ is the round key. This design makes the encryption identical to the decryption

except for the reversal of the key schedule. Feistel structures that operate on unequal divisions in the plaintext (i.e. $|X^L| \neq |X^R|$) are called unbalanced Feistel structures. The main weakness of Fiestel structures is that each round transformation always keeps some bits of the input block constant. This fact is used in many attacks. For example the differential [7] and linear [45] cryptanalysis attacks on the DES cipher extends differential and linear characteristics of one round to multiple rounds using this weakness.

**Wide trail strategy**

The wide trail strategy [13] is based partially on the substitution permutation network [59]. Here the entire input block is transformed in every round. Although this approach makes each round heavier compared to the Feistel structure, it helps in decreasing the number of rounds required for encryption. Block ciphers like Rijndael [17], Square [12] and Shark [55] are based on this strategy.

Most cryptanalytic attacks make use of the imbalances in the mappings between the differences/correlation in the ciphertext to a particular difference/correlation in the plaintext or the round key. The wide trail strategy aims to spread the difference/correlation characteristics to the entire cipherstate in a few rounds. This approach would prevent the cryptanalytic attacks that rely on the propagation of difference/correlation characteristics within sub-blocks of the input block. The spreading strength of the diffusion layer of a cipher is the key to achieve the wide trail strategy. However, diffusion is just a concept. In order to measure diffusion, a metric called branch number is often used (see 3.1). Branch number is the sum of the input and output active bytes (nonzero difference in input/output blocks). The wide trail strategy provides a simplified technique to maximize the sum of the active bytes (trail of active bytes) over a few rounds. The lower bound on the sum of active bytes also

provides a lower bound on the resistance offered by the cipher to many cryptanalytic attacks. In fact the Rijndael block cipher which is based on the wide trail strategy has been selected as the Advanced Encryption Standard (AES). In the next section we give an overview of the AES cipher.

## 2.3 Cryptanalysis of Block Ciphers

### 2.3.1 Differential Cryptanalysis

Differential cryptanalysis, as the name suggests analyzes the propagation of differences (plaintext/ciphertext) through the cipher to derive the key bits. Consider two plaintexts $P$ and $P'$. The difference between these plaintexts is $\delta P = P \oplus P'$. Since $\oplus$ is the key mixing operation, $\delta P$ is key independent. Difference between the cipherstates of the corresponding plaintexts after round $s$ is denoted by $\delta C_s$. A $s$ round differential characteristic is a $s+1$ tuple that lists the difference in the cipherstate starting from the first round, $(\alpha_0, \alpha_1, ..., \alpha_s)$. The probability, $p_D$, of this characteristic is given by,

$$p_D = Pr(\delta C_s = \alpha_s, \delta C_{s-1} = \alpha_{s-1}, ..., \delta C_1 = \alpha_1 / \delta P = \alpha_0). \tag{2.3.1}$$

However, this probability is very difficult to calculate. In [33] Lai and Massey proposed a Markov cipher model for iterated block ciphers and showed that for independent and uniformly random round keys, the probability of $s$ round characteristic can be approximated by,

$$p_D = \prod_{i=1}^{s} Pr(\delta C_1 = \alpha_i | \delta P = \alpha_{i-1}). \tag{2.3.2}$$

About $p_D 2^N$ plaintext differences (right pairs) follow the characteristic, where $N$ is the block length of the cipher. The steps required to attack a cipher using differential cryptanalysis as given in [31] and is summarized here. The attacker finds an $r-1$

round characteristic $(\delta P, \delta C_1, ..., \delta C_{r-1})$. Then the attacker uniformly selects $P$ and $P'$ with difference $\delta P$ and obtains the corresponding ciphertexts $C_r$ and $C'_r$. Then the attacker guesses the round keys such that the output difference $\delta C_{r-1}$ is observed. For every correct output difference observed a counter for the corresponding key is incremented. Eventually, after observing several plaintext pairs the correct key would emerge. The effectiveness of this technique can be quantified by the signal to noise ratio,

$$SNR = \frac{k \times p_D}{\lambda \times \gamma}, \tag{2.3.3}$$

where $k$ is the number of possible values of the key, $\gamma$ is the number of keys suggested by each right pair and $\lambda$ is the ratio of right pairs to all pairs. We can observe that higher the difference propagation probability $p_D$ higher is the success of differential cryptanalysis. The non-zero bytes in $\delta C_i$ are called active S-boxes/bytes. The difference propagation probability, $P_s$, of an active S-box is the relative number of all input pairs, that for a given input difference, gives rise to a specific output difference [14]. The probability of one round characteristic in terms of $P_s$ is,

$$Pr(\delta C_1 | \delta P) \leq (P_s)^{N_s}, \tag{2.3.4}$$

where, $N_s$ is the number of active S-boxes in the one round trail. Hence, a lower bound on the number of active bytes/S-boxes in any differential trail will give a lower bound on the resistance of the cipher to differential cryptanalysis.

### 2.3.2 Linear Cryptanalysis

Linear cryptanalysis [45] is a method of deriving key bits by forming linear expressions of the form,

$$(P \cdot \alpha) \oplus (C \cdot \beta) = (K \cdot \gamma) \tag{2.3.5}$$

where $P$ and $C$ are plain and ciphertexts, $K$ is the secret key, $\alpha,\ \beta,\ \gamma$ are selection vectors and $\cdot$ is the dot product. $|p_L - 1/2|^{-2}$ measures the success of linear cryptanalysis, where $p_L$ is probability of the linear expression (2.3.5). A linear characteristic is a collection of selection vectors that maximize the success of linear cryptanalysis. One of the first steps to construct linear expressions is to form the first round linear characteristic. The $r$ round characteristic can be obtained by concatenating the first round characteristic $r$ times. The success probability of an $r$ round linear expression is,

$$1/2 + 2^{N-1} \prod_{i=1}^{r} (p_{L_i} - 1/2) \tag{2.3.6}$$

where $N$ is the block length. Since substitution is the only non-linear step in most of the block ciphers including the proposed Pyramid cipher. We can express the success probability of the linear expression in terms of the number of active S-boxes in a multiple round linear trail. The active S-boxes in a round are determined by the non zero symbols in the selection vectors at the input of the round. The linearity of an active S-box can be approximated to the maximum input-output correlation exhibited by it. The correlation (measure of linearity) of a linear trail (multiple rounds) can be approximated to the product of input-output correlations of its active S-boxes [14]. Hence, a lower bound on the number of active bytes in any linear trail will give a lower bound on the resistance of the cipher to linear cryptanalysis.

### 2.3.3 Advanced Encryption Standard (AES)

The iterated block cipher Rijndael [15] was selected as the Advanced Encryption Standard (AES)[17], to replace the weaker Data Encryption Standard (DES)[59]. AES has a block length of 128 bits, and there are three allowable key lengths, namely 128, 192 and 256 bits. The number of rounds depends on the key length. For key lengths 128, 192 and 256 bits respectively 10, 12 and 14 rounds are used. A block

diagram of one round function of the AES cipher is given in Fig. 2.2. The round function of AES is composed of substitution (by S-boxes), shift row (SR) and mix column (MC). Every round function is preceded by a key mixing operation with the round key. The 11 round keys are generated from the secret key using a key expansion algorithm. The initial 128 bit plaintext is arranged as a $4 \times 4$ matrix of 16 bytes referred to as the cipherstate. During the substitution operation, each byte is independently substituted using the S-box. The SR operation shifts rows in the cipherstate corresponding to the row number. That is, the first row is not shifted, the second row is shifted by one byte and so on. The MC operation (see Fig. 2.3) multiplies each column of the cipherstate independently with a $4 \times 4$, 16 byte MC matrix. The key mixing is just a simple XOR of the cipherstate with the round key. Note that, the MC operation is not performed in the final (10th) round of the cipher. This is done to facilitate the use of the same algorithm for both encryption and decryption.

In the design of the Rijndael cipher a MC matrix with a branch number of 5 is used. The lower bound on the number of active bytes over any trail of four rounds is shown to be at least 25 [17]. The substitution boxes in the Rijndael have difference ratios and correlations of $2^{-6}$ and $2^{-3}$ respectively. For a four round difference and linear trail, the maximum difference propagation ratio is $2^{-150}$ and the maximum correlation ratio $2^{-75}$. This explains the resistance to difference and linear cryptanalysis.

## 2.4 Classification of Codes

The term codes refers to a broad class of techniques and algorithms that have two basic operations: encoding and decoding. Encoding is a procedure that adds redundancy into a given message and transforms it into a larger codeword. These codewords are

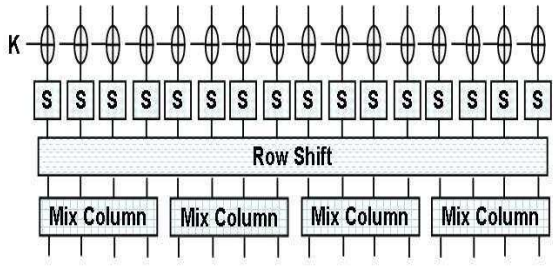


Figure 2.2: The Advanced Encryption Standard (AES) Encryption.

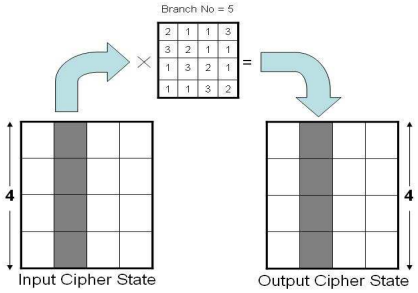


Figure 2.3: The Mix Column operation in the AES cipher.

decoded back into the corresponding messages by the decoding procedure. Depending on the amount of redundancy and the type of code, the decoding may recover the correct message even in the presence of one or more errors in the received codeword. There are two basic types of error correction codes: convolutional codes and linear block codes. Encoding procedure in convolutional codes is memory based and the message to codeword mapping depends on both the current state of the encoder and the previous message. Linear block codes on the other hand have a predetermined message to codeword mapping usually defined by a generator matrix. There are many sub categories in linear block codes. Some of the interesting ones are cyclic codes, perfect codes and low density parity check (LDPC) codes. Cyclic codes are linear block codes where any valid codeword is a cyclic shift of on another valid codeword

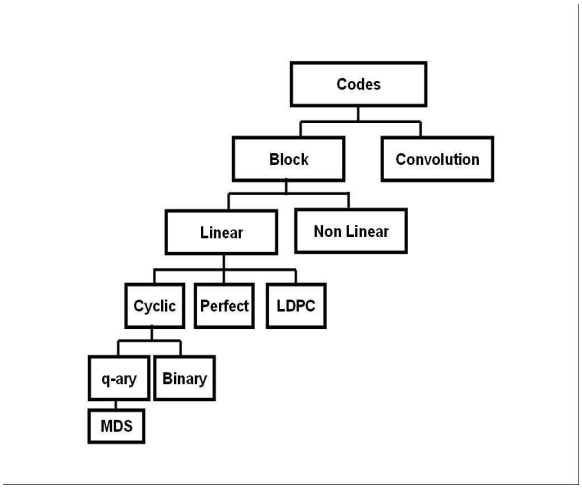


Figure 2.4: Classification of Codes.

in the same codespace. Cyclic codes can be binary (all codewords belong to the field GF(2)) or $q$-ary (codewords belong to GF($q > 2$)). Examples of cyclic codes are BCH codes and Reed Solomon (RS) codes. The summery of classification is given in Fig. 2.4

## 2.5 Linear Block Codes

A linear block code, usually denoted by $[n, k, q]$, encodes $\{GF(q)\}^k$ messages to $\{GF(q)\}^n$ codewords by multiplying them with a $k \times n$ generator matrix or $(n-k)$ degree generator polynomial. For example, if $m(x)$ represents the input message and $g(x)$ represents the generator polynomial, then the codeword is, $c(x) = m(x)g(x)$. The linearity comes from the fact that, for any two messages $m_1(x)$ and $m_2(x)$, we have $c_1(x) \oplus c_2(x) = (m_1(x) \oplus m_2(x))g(x)$. If the received codewords are not perfectly devisable by the generator, an error is detected. However, in order to correct errors, sophisticated algorithms are used. The performance or the error correcting capacity of any linear block code is heavily dependent on the minimum distance, $d_{min}$, between

the codewords. This is the minimum hamming distance between any two valid codewords in the code space. A larger $d_{min}$ implies better error resilience. The Singleton bound, $d_{min} \leq n - k + 1$, upper bounds this minimum distance. From this bound we can observe that a larger codeword length and a smaller message length increases the error resilience. However, this also increases the redundancy and hence transmission costs. The coding rate of any code is the fraction $k/n$. The fundamental problem of coding theory is to find codes that jointly maximize coding rate and $d_{min}$.

An important class of linear block codes that is of interest to our work is the Maximum Distance Separable (MDS) codes. The MDS codes are the sub class of linear, cyclic, q-ary block codes that satisfy the Singleton bound to equality. That is to say that, a $[n, k, q]$ MDS code has a minimum distance of $d_{min} = n - k + 1$. This is the largest possible minimum distance achievable by a linear block code.

## 2.6 Previous Work on Joint Error Correction and Encryption

Traditionally error correction and encryption in communication networks have been addressed independently. Although many mathematical relationships between coding and cryptography have been analyzed and explored [64], there have been only a few successful attempts in the past to combine coding and cryptography operations into one function. Several researchers have studied the trade-off between encryption and error correction by trying to combine these functionalities in one unit. Some of the notable works are briefly discussed.

### 2.6.1 McEliece Public Key Cryptosystem

McEliece proposed the first public-key cryptosystem based on algebraic coding theory in 1978 [46]. The idea behind this scheme was based on the fact that the decoding of

an arbitrary linear code is an NP-hard problem [3]. This scheme constructs a $(k, n)$, $t$-error correcting code where $k$ is the message length and $n$ is the codeword length using a generator matrix $G$. The message, $m$, is encoded to produce the ciphertext $c$ using the equation $c = mSGP+e$, where $S$ and $P$ refer to the substitution and permutation matrices respectively and $e$ is a random vector with weight $t' \leq t$. The key, $SGP$ (the product of the matrices $S, G$ and $P$, is published and the authorized users are provided with the private keys $S, G$ and the key to generate $e$. This is not really a joint error correction and encryption code, since the code is unable to correct any channel induced errors. However, these schemes have the advantage of lower power consumption by using the same hardware components available for error correction to achieve security. Conceptually, however, it is easy to extend this code by setting $t' \leq t'_{\max} < t$, where $t'_{\max}$ is the maximum weight of $e$ so that the code is able to correct $t - t'_{\max}$ link induced errors. Since the first scheme by McEliece, other researchers have worked on improving the information rate of the system [29, 52, 35, 57] by making the error vector carry some extra information. However, most of these systems suffer from information leakage which introduces weaknesses in these systems. Berson proposed two attacks [5] on the McEliece system that makes it easier for an attacker to crack the code. In [62], the authors propose two variations to the McEliece scheme: one using a hash function on the error vector $e$ and the other a trap-door function $f(m, e)$ to enhance the security of the basic McEliece scheme.

### 2.6.2 Sun's Private Key Cryptosystem

This is a private-key cryptosystem based on burst error correcting codes [61]. Here, the generator matrix of a burst error correcting code is and a permutation matrix are kept secret between the sender and the receiver. The ciphertext is obtained by encoding the message XORed with a predetermined burst sequence and permuting

the result using the permutation matrix. The decryption is performed by inverse permuting the ciphertext and then decoding the received codeword. However, due to the addition of burst errors prior to transmission, the true error correcting capacity of this scheme is significantly reduced.

### 2.6.3 Kak's D-sequences

In [30] Subhash Kak proposed a unique approach to joint encryption and error-correction. This solution is based on decimal expansions of fractions known as D-sequences. It is shown that the encoding operation is equivalent to that of exponentiation in finite field, which is similar to encryption in public key ciphers. However, this work has not been scalable and did not attract further research.

### 2.6.4 Godoy and Pereira Scheme

The functions of error correction and security were truly integrated in works like the Godoy and Pereira Scheme [22] which was intended for incorporation into existing systems without making any fundamental changes. The idea behind this scheme is to derive new generator matrices from existing generator matrices by row permutations. The security of the system relies on the change and secrecy of the generator matrices. Since the number of generator matrices for the given code are finite and countable, this scheme is susceptible to brute force attack on the generator matrix.

### 2.6.5 Hwang and Rao Scheme

Hwang and Rao proposed two secret error-correcting code (SECC) schemes [26]. This is a private key cipher that uses Perparata codes [39], which are a class of non-linear channel codes. Like many of the other schemes, this scheme suffers from reduction in error correcting capacity. In fact, in order to achieve meaningful error correction

capacity, the parameters of the system have to be very large leading to higher computational complexity. Also, this system was found vulnerable to the known plaintext attack conducted in [67].

### 2.6.6 Cryptcoding

Cryptocoding [21] is one of the more recently proposed techniques for joint error correction and encryption. This technique is based on quasigroup (Latin square) string transformation. A quasigroup of order 16 is chosen over $2^{480}$ possibilities when encoding and decoding functions are generated. The space of quasigroup gives the security for such a technique. Every message is padded with a bunch of zeros before the encoding/encryption operation. The presence of zeros is then verified by the decoding procedure and the decoding is repeated by flipping the received codeword symbols until the zero pad is recovered. Although this technique achieves both security and error correction. The decoding procedure is extremely complicated and cannot be used in a resource constrained environment.

# Chapter 3

# High Diffusion Codes

[1]

We look for channel codes that are not just good in error correction capabilities but also possess properties that make them useful as building blocks of ciphers. Specifically, we derive two distinct criteria that these codes must satisfy:

- *Security Criterion:* Since we intend to use the new code in the diffusion layer of a block cipher, we require the code to spread the statistical properties of the input block to a large section of the output block.

- *Error Resilience Criterion:* We do not want to compromise the error correction capabilities of the code in order to meet the security criterion. Hence we want the codes to have best possible error correction capability. The number of errors that can be corrected by block codes is governed by the pairwise minimum distance between the codewords [66]. A large minimum distance would ensure good error resilience property.

In this chapter we look at the existence of such codes and analyze their properties. A metric to measure diffusion called "Branch Number" is described in the

[1] part of this work was done in collaboration with Karthik Narayan ("On the Design of Secure Error Resilient Diffusion Layers for Block Ciphers")

next section. Then, we show the existence of codes that possess maximum possible branch number that do not compromise on the error correction capability. We call these error correcting codes as High Diffusion (HD) codes. Further we discuss the properties of HD codes and show that they indeed possess the best possible diffusion. Hence, making them an ideal candidate for block ciphers. We then describe several ways of constructing HD codes.

## 3.1 Branch Number

To define High Diffusion codes, we need a metric to measure diffusion. We use branch number of a function as a primary measure of its diffusion. For any function there are two ways of measuring the branch number, one is the differential measure and the other is the linear measure.

**Definition 3.1.1.** The differential branch number of a transformation $\phi$ mapping a $k$-tuple onto an $n$-tuple is defined as

$$\mathcal{B}_d^{diff}(\phi) = \min_{d_H(x_1,x_2)\neq 0} \{d_H(x_1, x_2) + d_H(\phi(x_1), \phi(x_2))\} \tag{3.1.0}$$

where $x_1$ and $x_2$ are two input $k$-tuples ($x_1 \neq x_2$) and $d_H$ is the byte Hamming distance [24].

**Definition 3.1.2.** The linear branch number of a transformation $\phi$ on mapping a $k$-tuple $x$ onto an $n$-tuple is defined as

$$\mathcal{B}_d^{lin}(\phi) = \min_{x\neq 0} \{w(x) + w(\phi(x))\} \tag{3.1.0}$$

where $w(.)$ is the Hamming weight in number of non-zero symbols.

If the function $\phi$ is linear, both linear and differential branch numbers for that function are the same.

## 3.2 Definition of HD Codes

Let us consider an $[n, k, q]$ block code, defined on the Galois field (GF) of order $q$; where $n$ refers to the number of output symbols and $k$ refers to the number of input

symbols. The HD codes are defined as follows:

**Definition 3.2.1.** A $[n, k, q]$ code $\mathcal{C}$, is said to be a High-Diffusion (HD) code with the encoding operation, $\theta$, if $\mathcal{B}_d^{lin,diff}(\theta) = n + 1$.

That is, the branch number of HD codes should be exactly equal to $n + 1$. We denote the function that measures branch number as $\mathcal{B}()$.

## 3.3 Properties of HD Codes

In this section, we show that the HD codes possess the maximum possible diffusion and error correction capacity as specified in the design criteria.

### 3.3.1 Optimality in diffusion

By definition, HD codes have a branch number of $n + 1$. By Lemma 3.3.1, this is the upper bound. Hence the diffusion is optimal.

**Lemma 3.3.1.** *The upper bound of branch number is* $n + 1$.

*Proof.* For a one byte difference in the messages, the corresponding codewords have to differ by all $n$ bytes to maintain the branch number of $n + 1$. Since there are only $n$ bytes in every codeword, it is not possible to get a branch number greater than $n + 1$. □

### 3.3.2 Optimality in error correction

We prove that HD codes are maximum distance separable codes (MDS) [39] and hence show that they are optimal in terms of the minimum distance.

**Theorem 3.3.2.** *An [n, k, q] HD code with encoding operation* $\theta$*, is an MDS code with minimum distance* $d_{min} = n - k + 1$.

*Proof.* Consider two messages $m_i$ and $m_j$ and the corresponding codewords $c_i$ and $c_j$. By the definition of HD codes (Definition 3.2.1) we have,

$$\begin{aligned} d_H(c_i, c_j) + d_H(m_i, m_j) &\geq \mathcal{B}(\theta) \\ d_H(c_i, c_j) + d_H(m_i, m_j) &\geq n + 1 \\ d_H(c_i, c_j) &\geq n - d_H(m_i, m_j) + 1 \end{aligned}$$

Since the messages are from a $k$-dimensional space maximum value of $d_H(m_i, m_j)$ is $k$,

$$\therefore d_H(c_i, c_j) = \quad d_{min} \quad \geq \quad n - k + 1 \tag{3.3.-3}$$

From Equation 3.3.-3 we see that HD codes satisfy the Singleton bound [39] with equality, which implies that HD codes are in fact MDS codes. □

The bound on error correction capacity, $t$, of HD codes is derived from the minimum distance between codewords as follows:

$$\begin{aligned} t &= \lfloor \frac{d_{min}}{2} \rfloor \\ \therefore t &= \lfloor \frac{n-k+1}{2} \rfloor \end{aligned} \tag{3.3.-3}$$

### 3.3.3 Totally positive generator matrix

**Definition 3.3.1.** A rectangular matrix $\mathcal{G} = (a_{ij}), i = 1, \cdots, k; j = 1, \cdots, n$ is called *totally positive* if all its minors (determinants of sub-matrices) of any order are positive [19].

Although the original definition in [19] is for matrices of real values, it can be easily extended to the case with elements in Galois field $GF(2^m)$.

**Theorem 3.3.3.** *Over a field $\mathcal{F}$, the linear transformation of $k$-tuples in $k$ dimensional space $V^k$ into $n$-tuples in $n(\geqslant k)$ dimensional space $V^n$ by an operation $y = x\mathcal{G}$ achieves the branch number of $n+1$ if (sufficient) and only if (necessary) $\mathcal{G}$ is a totally positive matrix.*

*Proof.* First we prove that the necessary condition to satisfy the branch number properties is the total positivity. From Definitions 3.1.1, 3.1.2, and Lemma 3.3.1, for transformation $\mathcal{G}$ to be diffusive, we require that

$$\begin{aligned} & d(x_1, x_2) + d(x_1\mathcal{G}, x_2\mathcal{G}) \geqslant n + 1 \\ \Rightarrow\ & w(x_1 \oplus x_2) + w(x_1\mathcal{G} \oplus x_2\mathcal{G}) \geqslant n + 1 \end{aligned} \tag{3.3.-3}$$

Since $\mathcal{G}$ is a linear transformation, (3.3.3) implies

$$w(x_1 \oplus x_2) + w((x_1 \oplus x_2)\mathcal{G}) \geqslant n + 1 \tag{3.3.-3}$$

Let $x_1 \oplus x_1 = e$. Then (3.3.3) reduces to

$$w(e) + w(e\mathcal{G}) \geqslant n + 1 \tag{3.3.-3}$$

| $w(e)$ | $\min\{w(e\mathcal{G})\}$ |
|---|---|
| 0 | 0 |
| 1 | $n$ |
| 2 | $n-1$ |
| $\vdots$ | $\vdots$ |
| $r$ | $n-(r-1)$ |
| $\vdots$ | $\vdots$ |
| $k$ | $n-k+1$ |

Table 3.1: Minimum change in the output to maintain branch number.

The minimum values of $w(e\mathcal{G})$ corresponding to the values of $w(e)$ to satisfy (3.3.3) are as given in Table I.

It can be seen that for $w(e) = r$, $\min\{w(e\mathcal{G})\} = n-(r-1)$. Let the columns of $\mathcal{G}$ be denoted by $h_j, j = 1, \cdots, n$. Then with a given $r$ for $r = 1, \cdots, k$ we require $\mathcal{G}$ to have at most $r-1$ columns such that $e \cdot h_j = 0$. This implies that in the $r \times n$ sub-matrix formed by selecting the rows of $\mathcal{G}$ corresponding to the non-zero elements of $e$, every $r \times r$ sub-matrix (contiguous as well as non-contiguous) should be of full rank. Since the $r$ non-zero elements in $e$ can occur at any $r$ out of $k$ positions, this implies that every $r \times r$ sub-matrix of $\mathcal{G}$ should be of full rank *i.e.*, positive for $r = 1, \cdots, k$. Thus by Definition 3.3.1, $\mathcal{G}$ should be a totally positive matrix.

Next we prove that the total positivity of the transformation matrix is sufficient to achieve the maximum branch number. If $\mathcal{G}$ is a totally positive matrix, every $r \times r$ sub-matrix is positive *i.e.*, has full rank for $r = 1, \cdots, k$. Let the rows of $\mathcal{G}$ be $a_i, i = 1, \cdots, k$. Then the linear combination of any $r$ rows, $\sum_{i=1}^{r} \alpha_i a_i$ with $\alpha_i > 0$ results in an $n$-tuple with at-most $r-1$ zero elements leading to $w(e)+w(e\mathcal{G}) = n+1$ and hence achieves the branch number. While this proof explicitly addresses the case of differential branch number property, the case of linear branch number property is implicit. □

## 3.4 Construction of HD Codes

*Unlike usual error correcting codes, the definition of HD codes involves pairs of messages and their associated codewords.* This makes deriving a closed form expression for the construction of the codes tricky. A brute force search produces the complete mapping but has the highest expected runtime. We have, therefore, developed three different shortcut techniques to generate HD codes.

### 3.4.1 Transformation from Reed Solomon (RS) codes

We have shown that all HD codes are MDS codes (See Theorem 3.3.2). Reed Solomon (RS) codes are a subclass of MDS codes. So another way of constructing a subclass of HD codes is as follows: a) start with the generator matrix of any $[q-1, k, q]$ RS code in systematic form ($\mathcal{G}_{rs} = [IP]$) b) $P$ sub-matrix of $\mathcal{G}_{rs}$ satisfies the branch number properties (Theorem 3.4.1). Therefore set the generator matrix of a HD code to P, (i.e. $\mathcal{G}_{hd} = P$). For example, to generate a $[6, 4, 256]$ HD code, we can take a $[10, 4, 256]$ RS code. The generator matrix of this RS code has a $4 \times 4$ identity matrix and a $4 \times 6$ parity check matrix. The parity check sub-matrix of this RS code is actually one of the generators of a $[6, 4, 256]$ HD code.

**Theorem 3.4.1.** *Parity check submatrix of a systematic RS generator matrix generates a HD codes.*

*Proof.* The parity check submatrix of a RS generator matrix in systematic form is totally positive (Theorem 15.6 in [25]). From Theorem 3.3.3 it follows that the branch number of the parity check matrix is $n + 1$. □

### 3.4.2 Searching totally positive generator matrices

Theorem 3.3.3 serves as a guideline for designing transforms to achieve the desired branch number properties. However, the testing of all possible square sub matrices of a matrix for positivity has an exponential order complexity. This can be reduced to polynomial order by testing only for initial minors (see Theorem 9 of [18]). This approach reduces the number of minors required to be tested for an $n \times n$ matrix from $\begin{pmatrix} 2n \\ n \end{pmatrix} - 1$ to $n^2$.

### 3.4.3 Puncturing existing codes

This gives us an easy way to generate new HD codes from existing HD codes.

**Theorem 3.4.2.** *Punctured HD codes are HD codes.*

*Proof.* Let $\mathcal{C}$ be an $[n, k, q]$ HD code and $\mathcal{C}'$ be the punctured $[n-1, k, q]$ code obtained from $\mathcal{C}$. Let $m_i$, $m_j$ be any two messages with their corresponding codewords $c_i$, $c_j$ in $\mathcal{C}$ and $c_i'$,$c_j'$ in $\mathcal{C}'$. We know that $\mathcal{C}$ is an HD code, therefore $d_H(m_i, m_j) + d_H(c_i, c_j) \geq n + 1$. We know that, $c_i'$ and $c_j'$ are obtained by puncturing $c_i$ and $c_j$ in one symbol position. This implies that $d_H(m_i, m_j) + d_H(c_i', c_j') \geq n$. Hence, $\mathcal{C}'$ is an HD code. □

## 3.5 Conclusions

High Diffusion codes possess the best possible diffusion and yet satisfy the Singleton bound for the minimum distance between codewords thus making them ideal candidates for error resilient cryptographic primitives. Although there is no systematic technique to generate HD codes, the flexibility to generate HD generator matrices from RS generator matrices makes it easy to derive large HD codes without having to go through brute force search. The close relationship of HD codes with the popular Reed Solomon codes makes them easy to study, analyze and port into existing systems.

# Chapter 4

# The High Diffusion Cipher

The diffusion property of HD codes can be used in the construction of the diffusion layer of a block cipher. We look at the Advanced Encryption Standard (AES) cipher design structure and propose to replace its diffusion layer with HD codes. We call this the High Diffusion cipher. Security of HD cipher is analyzed with respect to the best known cryptanalytic techniques like linear, differential and square attacks. We show that HD cipher is as secure as the AES under these attacks. Finally, we analyze the error resilience of the HD cipher to bursty channel errors.

## 4.1 Structure and Design

The HD cipher [42] is a key-alternating [11] block cipher, composed of 10 iterations of round function and key mixing operations. The round function consists of three layers: a) the non linear substitution layer, b) symbol transposition layer and c) the High Diffusion encoding layer. A block diagram of the HD cipher encryption is given in Fig. 4.1. Note that, the HD encoding is not performed in the final round. The input data, as it goes through each round of the cipher, is referred to as the *cipher state*. Note that, the output cipher state of the key mixing layer of round $r-1$ forms the input cipher state to the next round $r$. However, when $r = 10$, the output cipher

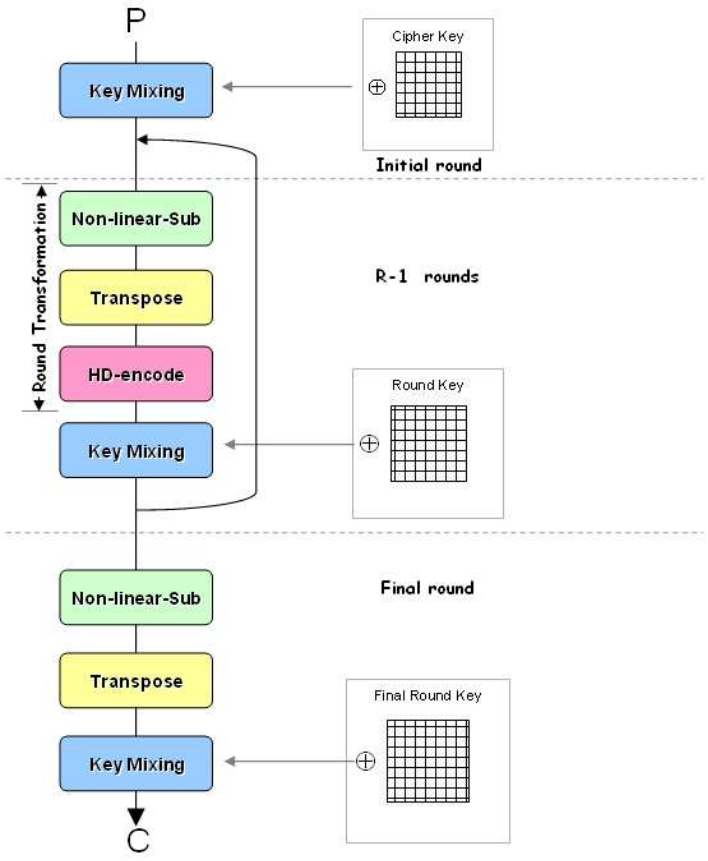


Figure 4.1: Block Diagram of High Diffusion Cipher.

state is the ciphertext (keystream in CTR mode). The 10 round HD-cipher operates on plaintext size of 128 bits to produce an output ciphertext (or keystream in CTR mode) of 288 bits. The secret key size required by the HD cipher is 288 bits. All the operations in HD cipher are performed in the finite field of order $2^8$, denoted by GF(256). A detailed description of all the layers of HD cipher follows.

### 4.1.1 Key mixing layer

The key mixing layer (see Fig. 4.1) follows every round function and is also performed once before the first round. Key mixing is a bitwise XOR operation of the cipher

state with the round key. The eleven round keys required for the eleven key mixing operations are generated using a key expansion algorithm. In HD cipher we use a key expansion algorithm which is similar to that of the AES key expansion algorithm [17]. However, we redesign the key expansion to expand a 288 bit secret key instead of the regular $128 - 256$ bit secret key. Since, the AES key expansion algorithm is easily expandable to any byte size, we do not concentrate on the design details.

### 4.1.2 Substitution layer

The substitution layer uses an invertible local non-linear transformation called the S-box. The non-linearity in S-box is designed to cause intra symbol avalanche [16] (that is every bit in the output symbol of the S-box flips with a probability of half for a single bit flip in the input symbol), which is essential for the security of the cipher. Nyberg proved that substitution functions generated by inverting elements in $GF(2^8)$ are differentially 4 uniform and are highly nonlinear [50]. The S-boxes thus constructed are used in the substitution layer of the HD cipher. Note that, these S-boxes are also used in the substitution layer of the AES cipher.

### 4.1.3 Symbol transposition layer

The symbol transposition layer is the first of the two diffusion operations used in the HD cipher. The aim of this layer is to permute the cipher state using a diffusion optimal transformation. We use the matrix transpose operation which was shown to be diffusion optimal [42].

### 4.1.4 HD encoding layer

The HD encoding transformation is the second diffusion operation used in the HD cipher. The aim of this layer is to diffuse the intra symbol avalanche caused by the

substitution layer to a large number of symbols in the resulting cipher state. We propose to use our novel HD codes in the encoding layer. By picking appropriate parameters for the HD code, it is possible to achieve the desirable level of error correction capability and expansion in the diffusion layer. We expect this to lead to resilience to channel errors when the cipher is used in block modes and appropriate amount of savings in energy consumption when used in stream modes.

In this work we use a [4,4,256] HD code for rounds 1 through 7 and a [6,4,256] HD code for rounds 8 and 9. The generator matrices for these HD codes are,

$$\mathcal{G}(r)_{r=[1...7]} = \begin{pmatrix} 1 & 1 & 3 & 2 \\ 2 & 1 & 1 & 3 \\ 3 & 2 & 1 & 1 \\ 1 & 3 & 2 & 1 \end{pmatrix}$$

$$\mathcal{G}(r)_{r=[8,9]} = \begin{pmatrix} 1 & 1 & 3 & 2 & 189 & 71 \\ 2 & 1 & 1 & 3 & 169 & 27 \\ 3 & 2 & 1 & 1 & 192 & 209 \\ 1 & 3 & 2 & 1 & 91 & 179 \end{pmatrix}$$

To perform HD encoding, each column of the input cipher state is multiplied with $\mathcal{G}(r)$ to obtain the output cipher state. The branch number (see Section 3.1) $\mathcal{B}(\mathcal{G}(r))$ of $\mathcal{G}(r)_{r=[1...7]}$ is 5 and $\mathcal{G}(r)_{r=[8,9]}$ is 7.

## 4.2 Security Analysis

In this section, we briefly analyze the security of HD ciphers by looking at the resistance it offers against some well known cryptanalytic attacks.

### 4.2.1 Resistance to differential and linear cryptanalysis

Differential cryptanalysis [6, 7] is a chosen plaintext-ciphertext attack that makes use of the difference propagation property of a cipher to deduce the key bits. The difference propagation property of an S-box is the relative number of all input pairs, that for a given input difference, gives rise to a specific output difference. It is expressed as propagation ratio [11]. Let $x_1^r$ be any intermediate cipher state at round $r$ resulting from the input plaintext $P_1$. Similarly, let $x_2^r$ be the corresponding intermediate cipher state resulting from plaintext $P_2$. The non zero symbols in $x_1^r \oplus x_2^r$ are called active symbols or S-boxes. The difference propagation of consecutive rounds can be concatenated across several rounds to form a differential trail. The propagation ratio over all the rounds of a differential trail can be approximated by the product of the propagation ratios of its active S-boxes. Differential cryptanalysis can break the HD cipher with complexity less than $\mathcal{O}(2^{128})$ if the maximum possible propagation ratio over all rounds is significantly larger than $2^{-127}$.

Linear cryptanalysis [45] is a known plaintext-ciphertext attack that makes use of linearity in the cipher to obtain the key bits. Substitution is the only non-linear step in most of the block ciphers including the proposed HD cipher. The linearity of an active S-box can be approximated to the maximum input-output correlation exhibited by it. The active S-boxes in a round are determined by the non zero symbols in the selection vectors at the input of the round. The linearity of one round can be extended to multiple rounds to form a linear trail. The correlation (measure of linearity) of a linear trail (multiple rounds) can be approximated to the product of input-output correlations of its active S-boxes. Linear cryptanalysis can break the HD cipher with complexity less than $\mathcal{O}(2^{128})$ if the maximum possible correlation of any linear trail over all rounds is significantly larger than $2^{-64}$.

Hence, a lower bound on the number of active symbols in any linear or differential trail will give a lower bound on the resistance of the cipher to linear and differential cryptanalysis. In Theorem 4.2.3 we show that this lower bound for any four rounds of HD cipher, starting with round $r$ is $\mathcal{B}(\mathcal{G}(r)) \times \mathcal{B}(\mathcal{G}(r+1))$. The lower bound on the number of active S-boxes in any linear or differential trail in the last four rounds of the HD cipher proposed here is $\mathcal{B}(\mathcal{G}(7))\mathcal{B}(\mathcal{G}(8))$ or 35. The S-boxes used in the substitution layer of HD cipher have a maximum propagation ratio of $2^{-6}$ and a maximum input and output correlation of $2^{-3}$. This shows that there are no four round differential trails with predicted propagation ratio above $2^{-215}$ and no four round linear trails with predictable input output correlation above $2^{-105}$. The initial six rounds are added as a security margin towards future attacks, just as in AES.

**Lemma 4.2.1.** *The total number of active columns of one round function is lower bounded by the branch number of $\mathcal{G}$, $\mathcal{B}(\mathcal{G})$.*

This is true for any diffusion optimal transformation. Proof given in [13].

**Theorem 4.2.2.** *The number of active S-boxes or symbols for a two round trail of HD cipher is lower bounded by the branch number of the first round of HD code, $\mathcal{B}(\mathcal{G}(1))$.*

*Proof.* Consider the first two rounds of HD cipher. Since substitution and key mixing operate on the symbols locally, they do not affect the propagation pattern. Hence the number of active S-boxes or symbols for a two round trail is bounded by the propagation property of $\mathcal{G}(1)$. From the definition of HD codes the sum of active S-boxes before and after HD encoding of the first round is lower bounded by $\mathcal{B}(\mathcal{G}(1))$. □

**Theorem 4.2.3.** *The number of active S-boxes or symbols for a four round trail (starting with round $r$) of HD cipher is lower bounded by $\mathcal{B}(\mathcal{G}(r)) \times \mathcal{B}(\mathcal{G}(r+1))$.*

*Proof.* Proof given in [42]. □

### 4.2.2 Resistance to square attack

The Square attack [12] (also known as Integral attack [32] or the Saturation attack [38]) makes use of the byte oriented nature of the Square block cipher which was

the predecessor of AES. As AES is also a byte oriented cipher, this attack has been extended to reduced versions of AES [37, 20]. The proposed HD cipher also comprises of byte oriented operations which are loosely based on AES, hence HD ciphers with fewer than seven rounds would be as weak as reduced versions of the AES.

Although the HD cipher is as secure as AES against most of the well known attacks, the HD cipher uses a larger key length to achieve the same security level as that of AES. Since, the key expansion is performed only once every session, its computational overhead is negligible.

## 4.3 Error Correction Capacity

In this section, we prove bounds on the error correction capacity of the HD cipher. After encryption, the ciphertext of length 36 bytes (equivalently 288 bits) is transmitted across a noisy channel. Specifically, we consider a bursty channel [10] and use the term "full weight burst error" to denote an error burst where all the symbols in the burst are in error. We do this to calculate the lower bound on the error correction capability. In order to formalize our analysis we introduce the following assumptions, definitions and notations. A symbol of the cipher state that is in error (due to channel or propagation due to decryption) is referred to as an *error symbol*. If a row/column in the $4 \times 4$ representation of the cipher state has more than one error symbols, it is said to be an error row/column. Error correction capacity of a four round HD cipher decryption is analyzed in Theorem 4.3.3.

**Lemma 4.3.1.** *If there is at most* 1 *error row or column in the cipher state before the first HD decoding, then the error correction is complete after the second HD decoding.*

*Proof.* Consider the first three rounds of HD cipher decryption. Since the inverse non-linear transform and round key addition and the transpose operations do not convert an error symbol to an error free symbol and vice versa, it can be excluded from the analysis. If there is only one error row in the cipher state before the first HD decoding, then the error correction will be complete after the first decoding. This

is because, the decoding takes place columnwise and each HD code has 1 byte error correction capacity. If there is one error column, it will remain an error column even after the first HD decoding, however, the transpose operation will convert it to an error row before the second HD decoding is performed. The second HD decoding then completes the error correction. □

**Lemma 4.3.2.** *If there are at least* 2 *error columns in the cipher state before the first HD decoding, the error correction may remain incomplete after the second HD decoding.*

*Proof.* The two error columns before the first HD decoding will remain in error even after decoding. The transpose will make the two error columns into two error rows. Now every column in the cipher state may have more than one error byte. Thus, the second HD decoding may not be able to correct all errors. □

We now analyze the maximum full weight burst error length that is guaranteed to be corrected by a four round HD cipher. We assume columnwise transmission of the ciphertext. Our analysis is independent of the starting and ending locations of the burst with respect to the cipher state.

**Theorem 4.3.3.** *The full weight burst error correcting capacity of a four round HD cipher is* 7*.*

*Proof.* The largest full weight burst error that can occur without causing a single error column in the cipher state before the first HD decoding is 6. An extra byte in error either next to the starting/ending location of the burst will create one error column. From lemma 4.3.1 we know that this is correctable. Hence, a full weight burst of 7 bytes is correctable. However, a full weight burst of 8 bytes will create two error columns and from lemma 4.3.2 we know that the decoding may fail. □

From Theorem 4.3.3 we get the lower bound on the burst length for burst error correction per block of HD cipher. A $[36, 16, 256]$ RS code has a burst correction capability of 10 bytes. However, since we use many small HD codes instead of one large RS code, we expect HD cipher to be more energy efficient than a traditional cipher concatenated with the large RS code.

## 4.4 Modes of Operation

Encrypting each plaintext block independently in Electronic Codebook (ECB) mode does not provide semantic security. This is because, the adversary can distinguish between two different plaintexts just by observing the ciphertext. Moreover, if the block length is not very large, the adversary can construct a table of known plaintext-ciphertext pairs and use it as a lookup table to decode unknown plaintexts encrypted with the same key. To improve the semantic security of block ciphers several other modes have been suggested. The most popular of these are the Cipher Block Chaining (CBC) mode and the Counter (CTR) mode. The CBC mode encrypts plaintexts one block at a time, hence it is referred to as block mode encryption. However, CTR employs the underlying block cipher to produce a pseudo random key stream, which is then bitwise XORed with the plaintext in the stream cipher style. Hence, CTR mode is usually referred to as a stream mode. It has been shown in [28] that both CBC and CTR modes have equivalent security for a given block cipher. In this section, we construct and analyze the performance of HD cipher in both block and stream modes.

### 4.4.1 Cipher block chaining (CBC) mode

In the CBC mode, every plaintext block is XORed with the previous ciphertext block before encryption. The first plaintext block is XORed with an Initialization Vector (IV). The chaining of ciphertext block makes CBC mode more semantically secure compared to ECB mode. In our work, we implement HD cipher in CBC mode and compare it with traditional concatenated systems in terms of error correction capabilities.

To evaluate the energy efficiency, we measured the actual energy consumption

of the HD cipher on a testbed and compared it with that of traditional systems. The testbed (Fig 4.2) consists of an Intrinsyc CerfCube [27] with a 233 MHz ARM processor, 16MB Flash and 32 MB SDRAM, running Debian Linux operating system. The power consumed by the CPU in running the encryption algorithms is measured as a function of input power supply to the CerfCube. A separate DC power supply is given to the CerfCube to permit measurements. The current is measured using Labview from the GPIB interface of the power supply. To eliminate effects of any programs running in the background, the current consumption is first tested when no other tasks are running. The difference in currents when the algorithm is running and the idle current (in Amperes) is taken as the actual current consumption. In the experiments, since voltage variation is seen to be extremely small (measured to be less than 0.025%) we use a constant value. We use OProfile [51] to measure the exact time taken by the algorithms to run. The energy consumed by the algorithms is the product of power drawn from the DC source and the time required to complete execution.

The energy measurements are given in Tables 4.1 and 4.2. We can observe from the tables that the HD cipher saves 30% and 12% energy per byte during encryption and decryption compared to the traditional systems.

To evaluate the performance (error correction) of the HD cipher, we compare it with concatenated systems A and B (described below) with respect to error correction capacity.

- *Concatenated system A:* uses AES (128-bit) cipher concatenated with [36,16,256] Reed Solomon code.

- *Concatenated system B:* uses AES (128-bit) cipher concatenated with convolutional codes having rates varying from 1/2 to 1/6.

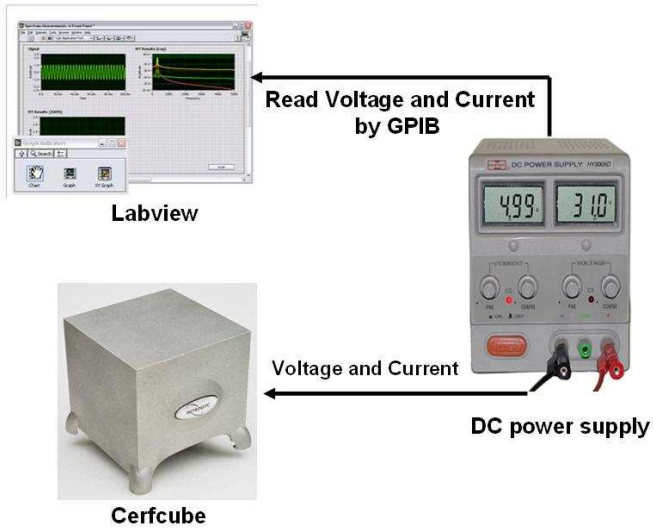


Figure 4.2: Hardware Setup

| Block Mode | Voltage (volts) | Current (amps) | Time (secs) | Energy (Joules) |
|---|---|---|---|---|
| HD Encryption | 5.003 | 0.237 | 8.9 | 10.55 |
| AES Encryption | 5.003 | 0.237 | 6.6 | 7.82 |
| RS Encoding | 5.003 | 0.238 | 6.3 | 7.05 |
| HD Decryption | 5.003 | 0.239 | 24.1 | 28.51 |
| AES Decryption | 5.003 | 0.238 | 8.2 | 9.76 |
| RS Decoding | 5.003 | 0.239 | 19.3 | 23.07 |

Table 4.1: Voltage, current, time and energy measurements for the one million HD, AES and RS encryption/encoding and decryption/decoding operations.

| Per Byte Energy ($\mu J$) | HD cipher | AES-RS cipher-code |
|---|---|---|
| Encode/Encryption | 0.65 | 0.95 |
| Decode/Decryption | 1.80 | 2.05 |

Table 4.2: Per byte energy consumption for encoding/encryption, decoding/decryption operations of the error correcting HD cipher and the AES-RS concatenated system.

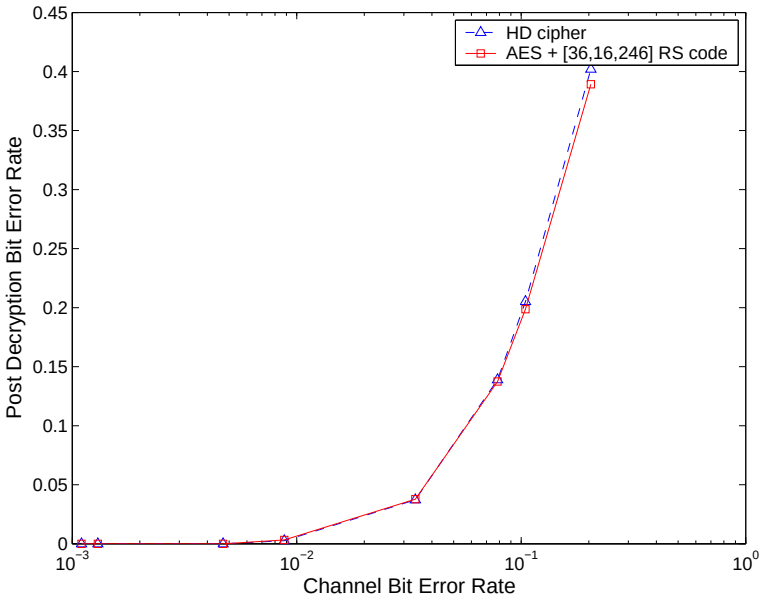


Figure 4.3: Comparison of error resilience of HD cipher and AES concatenated with [36,16,256] Reed Solomon codes.

Wireless communication medium is characterized by bursty errors and fading phenomenon. Which implies that the bit errors occurring in wireless channels have memory. Alajaji, et al.[2] proposed an additive Markov channel (AMC) model for slow fading wireless channels. According to this model, the channel can be described by bit error rate and correlation parameters. The burstyness of the channel can be controlled by the correlation parameter. In our experiments we set the correlation to 0.9 and varied the bit error rate from 0.0005 to 0.2.

Fig. 4.3 plots the post decryption bit error rate of the proposed 128 bit HD cipher and the concatenated system A against channel bit error rate. It can be observed that HD cipher and the concatenated system are comparable in terms of error correction capacity over all the channel bit error rates. This is because, both HD cipher and the Reed Solomon code used in the concatenated system are burst error correcting codes with similar coding rates. However, as the error correction is

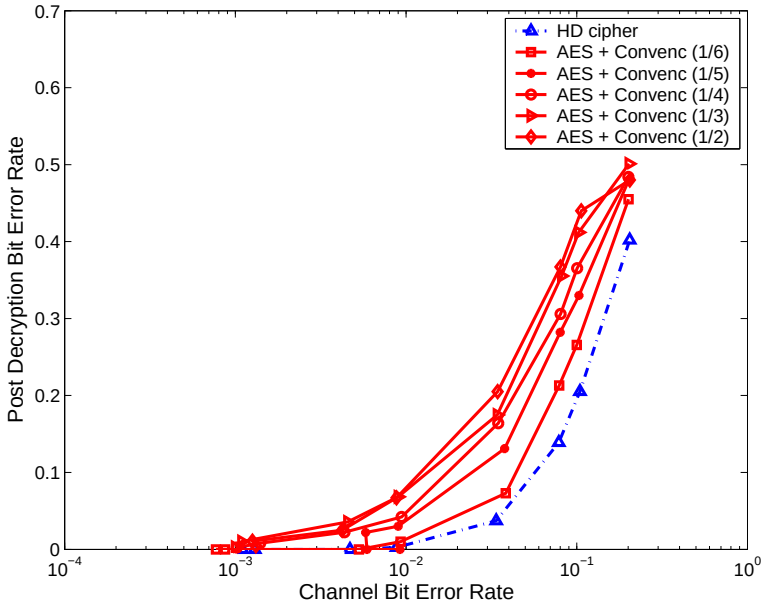


Figure 4.4: Comparison of error resilience of HD cipher and AES concatenated with Convolutional codes. Notice that the coding rate of HD cipher is between 1/5 and 1/6, yet it outperforms the 1/6 rate concatenated system.

performed during decryption within the HD cipher, there is roughly a savings of two rounds per encryption/decryption compared to the concatenated system.

For the second set of experiments, we compare the proposed 128 bit HD cipher with the concatenated system B. Different convolutional codes with rates 1/2, 1/3, 1/4, 1/5 and 1/6 are considered. Since, the channel is assumed to be bursty, a block interleaver is added after convolutional encoder to optimize the performance of the concatenated system. Hard decision Viterbi decoder [10] is used at the receiver. Fig. 4.4 plots the post decryption bit error rate of the proposed HD cipher and the concatenated system B. The HD cipher clearly outperforms the concatenated system for all rates 1/2 through 1/6. Note that, the coding rate of the HD cipher is between that of the concatenated systems with rate 1/5 and 1/6 yet it outperforms the rate 1/6 concatenated system. Although convolutional codes are more light weight compared to Reed Solomon codes, the total number of operations when it is combined with 10

round AES cipher is approximately equal to the number of operations in a 10 round HD cipher.

### 4.4.2 Counter (CTR) mode

In the CTR mode, the block cipher is used to encrypt a counter value which is incremented for successive encryptions. The encrypted counter values makeup a keystream which is XORed with the plaintext bits to produce the ciphertext bits. Block ciphers act as pseudo random number generators (PRNGs), hence this mode is semantically more secure compared to the ECB mode. Since, the encryption of plaintext takes place one bit at a time, this mode of encryption is usually referred to as the stream mode. In this work, we implement the HD cipher in CTR mode as a component of the current 802.11 wireless LAN security protocol called the Counter Mode Encryption with Cipher Block Chaining Message Authentication Protocol (often referred to as CCMP). We then compare the performance of our proposed HD-CCMP with the traditional AES-CCMP.

The CCMP currently uses the AES block cipher (see Fig. 4.5) to provide both authentication and confidentiality.

The drawback of AES-CCMP is that, it consumes more energy compared to its predecessor, the Wired Equivalent Privacy (WEP). This is because, the RC4 cipher used in WEP is a stream cipher; whereas, the AES used in CCMP is inherently a block cipher used in stream (counter or CTR) mode. *Therefore, a full 10 round AES needs to be performed to encrypt every* 128 *bits of Message Protocol Data Unit (MPDU)* (see Fig 4.5).

By using the HD cipher instead of the AES (see Fig. 4.5) we propose to make CCMP more energy efficient. Let $C$ denote the 128 bit counter, $X$ denote the MPDU stream (data payload + MIC) of length $N$-bits to be encrypted and $X_i$ denote the $i$-th

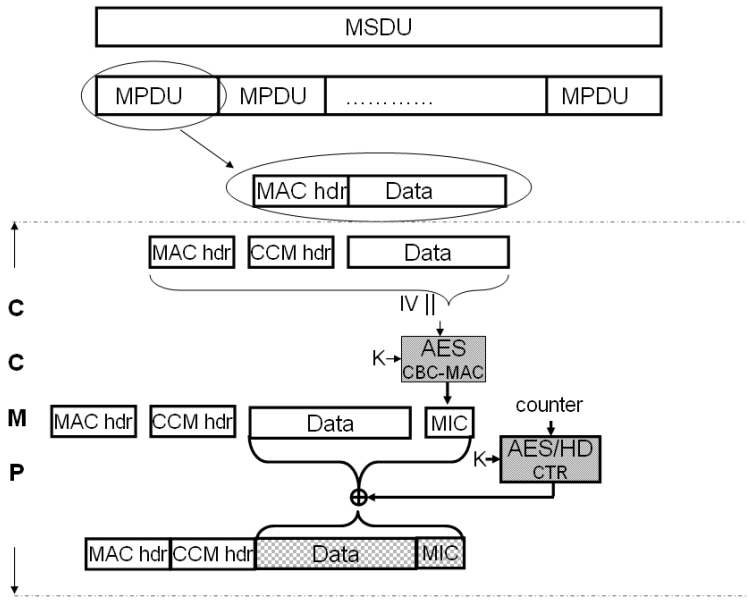


Figure 4.5: Block Diagram of CCMP.

288 bit block of $X$ (where, $i \in \{1...\lceil N/288 \rceil\}$). The HD encryption under key $K$, denoted by $E_K$, in the CTR mode is $Y_i = E_K(\text{counter}+(i-1)) \oplus X_i$, $\forall i \in \{1...\lceil N/288 \rceil\}$. Here, $Y$ represents the encrypted MPDU stream. In the CTR mode, the HD cipher securely expands the 128 bit counter to 288 bit keystream, thus encrypting more than twice the number of bits per encryption compared to the AES. The number of encryptions required by the HD cipher per $N$ bit frame is $\lceil N/288 \rceil$, whereas for the AES it is $\lceil N/128 \rceil$. We therefore expect to do only 50% of work to achieve the same level of confidentiality compared to the AES-CCMP. Moreover, as $n$ increases we expect to observe larger reduction in energy consumption. Although, from a security standpoint we can use the HD cipher in CBC-MAC mode for authentication as well, this does not cause additional savings in terms of energy, hence we use the HD-cipher in the CTR mode for confidentiality only. From an implementation standpoint, most of the operations of the HD cipher are similar to the AES, hence significant portions of

code can be reused. Therefore, using two different ciphers in CCMP does not impose a significantly larger code space requirement.

The HD and the AES ciphers consume $0.29\mu$ $J$ and $0.49\mu$ $J$ (from Table 4.1) of energy per byte respectively. Hence, HD cipher in counter mode results in about 40% reduction in energy consumption compared to the AES cipher.

## 4.5 Conclusions

The construction of HD cipher from HD codes proves the latter's potential as building blocks for ciphers. The branch number of HD codes provides adequate diffusion to the HD cipher. Hence, making them secure against well known cryptanalytic attacks. The error resilience analysis of HD cipher revealed that the error correction capacity of HD codes are not compromised because of their use inside the cipher. The joint security and error resilience properties favors HD cipher as a potential solution to current cipher design challenges. The HD cipher is employed both in block and stream modes. In block mode, the HD cipher corrects bursty channel errors and hence is more error resilient compared to traditional ciphers. Furthermore, HD cipher when used in stream mode has more encryption throughput compared to the popular AES cipher. Energy consumption analysis revealed that HD cipher is 40% more energy efficient. The error resilience, higher encryption throughput and energy efficiency properties makes the proposed HD cipher an ideal replacement for the AES cipher in resource constrained and noisy wireless environments.

# Chapter 5

# The Pyramid Cipher

In Chapter 1 we pointed out error resilience, energy efficiency, speed (number of rounds) and encryption throughput as four important present day design challenges in block ciphers. In Chapter 4 we showed that the High Diffusion ciphers address three of the four challenges: error resilience, energy efficiency and encryption throughput. However, the number of rounds in the HD cipher, which is 10, is equal to that of the AES (Rijndael) cipher. This can be largely attributed to the structural similarity between the AES cipher and the HD cipher.

In this chapter, we investigate techniques to reduce the number of rounds in the cipher. Specifically, we observe that by making the diffusion layer as large as possible, avalanche effect can be caused within fewer rounds. Based on this philosophy we design a five round error correcting cipher called the Pyramid. We show that the reduction in the number of rounds, does not adversely impact the security of the cipher. Further, we show that the Pyramid cipher is as secure as the AES cipher against linear, differential and square attacks. We derive bounds on the error correcting capacity of the Pyramid cipher and through simulations show that they are as error resilient as the Reed Solomon (RS) codes and outperform convolutional codes by 60%. Energy analysis experiments on a 32 bit test bed reveals that, Pyramid

cipher is $6 - 10\%$ faster and energy efficient than a concatenated system. Finally, we implement the Pyramid cipher in stream (counter) mode and show that they are secure random number generators using the DIEHARD statistical test suite with an higher encryption throughput $\geq 192$ bits (compared to the AES cipher).

## 5.1 Structure and Design

The Pyramid is a five round cipher that encrypts 128 bit plaintexts using a 192 bit secret key to produce 192 bit ciphertexts as shown in Fig. 5.1. Since the cipherstate expands as it goes through the cipher, we call it the Pyramid cipher. The round function in the Pyramid cipher consists of three distinct layers: the key mixing layer, the non linear substitution layer and the linear diffusion layer. The Pyramid decryption is the exact inverse of the encryption operation.

### 5.1.1 Key mixing layer

In the key mixing layer the round key is XORed with the cipherstate. As Pyramid cipher is a key iterated block cipher, there are six key mixing operations with the round key (denoted by $\oplus$ in Fig. 5.1). The six round keys are generated from the 192 bit secret key using a key expansion algorithm, which is similar to that of the AES key expansion algorithm.

### 5.1.2 Substitution layer

The substitution layer consists of simple table lookup operations. The substitution tables are usually referred to as the S-boxes. Each byte in the input cipherstate is substituted for a byte in the output cipherstate. Let $C_{i,j}$ represent the $j$-th byte of

the $i$-th round cipherstate. Then the cipherstate after substitution $C^s_{i+1}$ is,

$$\forall_j C^s_{i+1,j} = S[C_{i,j}] \tag{5.1.0}$$

The substitution operations are denoted by $S$ in Fig. 5.1. During decryption, the substitution boxes are replaced by inverse substitution boxes. The S-boxes and the inverse S-boxes are identical to those used in the HD cipher.

### 5.1.3 Diffusion layer

In the diffusion layer, the cipherstate is multiplied with a diffusion matrix $\mathcal{G}$.

$$C_{i+1} = C^s_{i+1} \times \mathcal{G} \tag{5.1.0}$$

In the first three rounds the entire 16 byte cipherstate is multiplied by a single $16 \times 16$ diffusion matrix called the Mix Column (MC) matrix (see Fig. 5.1), $\mathcal{G}_{\text{MC}}$ with operations in GF(256). In the fourth round however, the 16 byte input cipherstate is multiplied with a single $16 \times 24$ diffusion matrix called the HD encoding matrix (see Fig. 5.1), $\mathcal{G}_{\text{HD}}$, in GF(256) to produce a 24 byte output cipherstate. The MC and the HD matrices have branch numbers 17 and 24 respectively. The high branch numbers are important to achieve resistance against cryptanalytic attacks (see Section 5.2). Details on the construction of the diffusion matrices is provided in Section 5.1.3 and 5.1.3. The fifth (final) round does not have any diffusion operation. However, the fifth round cipherstate consists of 24 bytes and hence requires 24 substitution operations instead of 16. Notice that, in the diffusion layer of the Pyramid cipher is larger and does not have any ShiftRow [15] operation (when compared to the HD and AES ciphers). This is because we operate on the entire cipherstate in each round. As the cipher expands the input state as it goes through the encryption process, we call it the Pyramid cipher. During decryption, the inverse MC matrix replace the MC matrix

and HD decoding operation is replaced by the HD encoding operation. We use the Euclidean errors and erasures decoding [36] with slight modification to decode the 24 byte cipherstate. The decoding procedure is described in Section5.3.

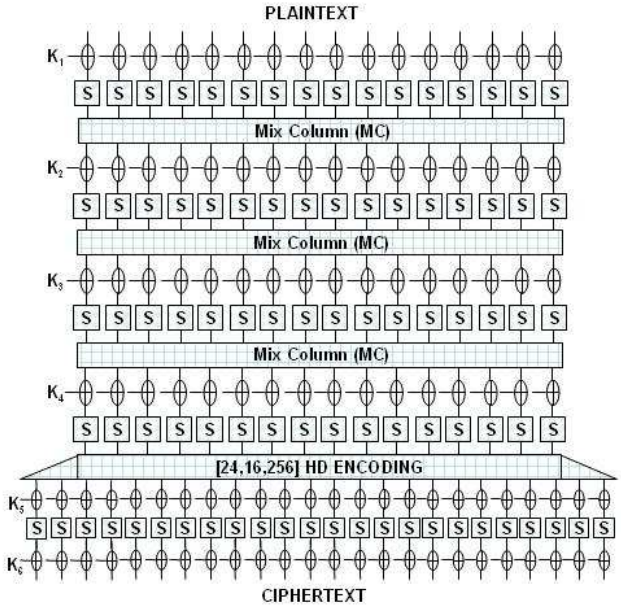


Figure 5.1: Full five round Pyramid block cipher

### Construction of the HD encoding matrix

The HD encoding refers to multiplying the cipherstate by the generator matrix of a $[24, 16, 256]$ High Diffusion (HD) code (see Chapter 3. We construct the $[24, 16, 256]$ HD code from $[32, 16, 256]$ shortened Reed Solomon (RS) code [10] with generator polynomial,

$$g_{\mathrm{RS}}(X) = X^8 + X^4 + X^3 + X^2 + 1 \tag{5.1.0}$$

From $g_{\mathrm{RS}}(X)$ we get the generator matrix, $\mathcal{G}_{\mathrm{RS}}^{\mathrm{SYS}}$, of the RS code in systematic form [36]. The first 8 columns of $\mathcal{G}_{\mathrm{RS}}^{\mathrm{SYS}}$ are punctured to derive the diffusion matrix, $\mathcal{G}_{\mathrm{HD}}$.

From Theorem 3.4.1 it follows that $\mathcal{G}_{\text{HD}}$ obtained by puncturing $\mathcal{G}_{\text{RS}}^{\text{SYS}}$ in this manner generates a $[24, 16, 256]$ HD code. The branch number of $[24, 16, 256]$ HD code is,

$$\mathcal{B}(\mathcal{G}_{\text{HD}}) = 24 + 1 = 25. \tag{5.1.0}$$

**Construction of the MC matrix**

We generate the MC matrix, $\mathcal{G}_{\text{MC}}$, by puncturing the first 8 columns of $\mathcal{G}_{\text{HD}}$. As punctured HD codes are HD codes (Theorem 3.4.2), $\mathcal{G}_{\text{MC}}$ is actually a generator for $[16, 16, 256]$ HD codes with branch number,

$$\mathcal{B}(\mathcal{G}_{\text{MC}}) = 16 + 1 = 17 \tag{5.1.0}$$

The inverse MC matrix $\mathcal{G}_{\text{MC}}^{-1}$ is obtained by inverting $\mathcal{G}_{\text{MC}}$ in $GF(256)$.

### 5.1.4 Rationale for larger diffusion operations

In this section we discuss the rationale for employing larger diffusion operations (compared to AES) in the Pyramid cipher. Lets take the wide trail structure of AES and replace multiple smaller diffusion operations in each round with one large diffusion operation. Fig. 5.2 and Fig. 5.3 represent the block diagrams of the traditional wide trail structure and our modified wide trail structure respectively. Note that in the wide trail structure, a single active byte (one byte difference) in the input plaintext, will travel to all the bytes in the cipherstate by the end of the second round. However in the modified wide trail structure, a single active byte in the input travels to all the 16 bytes of the cipherstate in just one round. This suggests that by using larger diffusion operations, we can achieve trails with higher number of active bytes in fewer rounds. However, reduction in the number of rounds does not necessarily imply reduction in energy consumption. This is because, as the diffusion operations

get larger each round gets heavier. The actual savings in energy depends on the number of rounds reduced, efficient implementation of each of the rounds and the weight of the larger diffusion operations. The number of rounds is determined by looking at the best possible attack on the cipher. In the next section we analyze the resistance of the Pyramid cipher against some well known attacks.

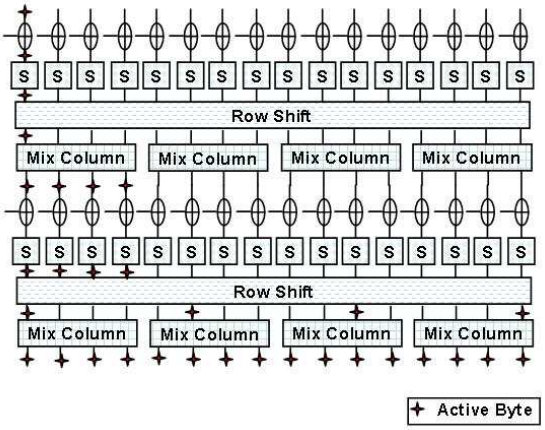


Figure 5.2: Active byte propagation in the wide trail strategy

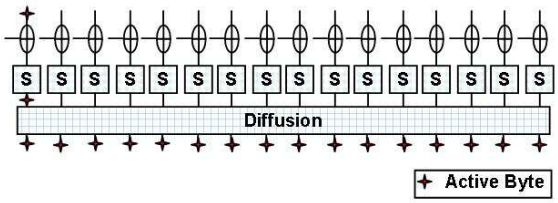


Figure 5.3: Active byte propagation due to large diffusion operation

## 5.2 Security Analysis

In this section, we briefly analyze the security of the Pyramid cipher by looking at the resistance it offers against some well known cryptanalytic attacks.

### 5.2.1 Resistance to linear and differential cryptanalysis

From discussion on linear and differential cryptanalysis from Section 4.2.1 it follows that a lower bound on the number of active bytes in any linear or differential trail will give a lower bound on the resistance of the cipher to linear and differential cryptanalysis. For any three round trail of the Pyramid cipher, the minimum number of active bytes is shown to be greater than or equal to 34 in Theorem 5.2.2. This shows that there are no three round linear trails with predictable input output correlation above $2^{-3\times 34} = 2^{-102}$ and no three round differential trails with predictable propagation ratio above $2^{-6\times 34} = 2^{-204}$.

**Lemma 5.2.1.** *The minimum number of active bytes in any one round trail of the Pyramid cipher is* 17*.*

*Proof.* The key XOR and substitution do not turn an active byte into an inactive byte and vice versa. The sum of active bytes in the input and the output cipherstate of a one round trail entirely depends on the branch number of the HD encoding matrix $\mathcal{G}_{\text{HD}}$. We know from (5.1.3)that $\mathcal{B}(\mathcal{G}_{\text{HD}}) = 17$. □

**Theorem 5.2.2.** *The minimum number of active bytes in any three round trail of the Pyramid cipher is* 34*.*

*Proof.* Fig. 5.4 represents a three round trail. The minimum number of active bytes in any three round trail of the Pyramid cipher is $\min \Sigma_0^3(\delta C_i)$. This is equal to, $\min(\Sigma_0^1(\delta C_i) + \Sigma_2^3(\delta C_i))$. From Lemma 5.2.1 we have $\min(\Sigma_0^1(\delta C_i)) = \min(\Sigma_2^3(\delta C_i)) =$

17. Therefore, the minimum number of active bytes in any three round trail of the Pyramid cipher is 34. □

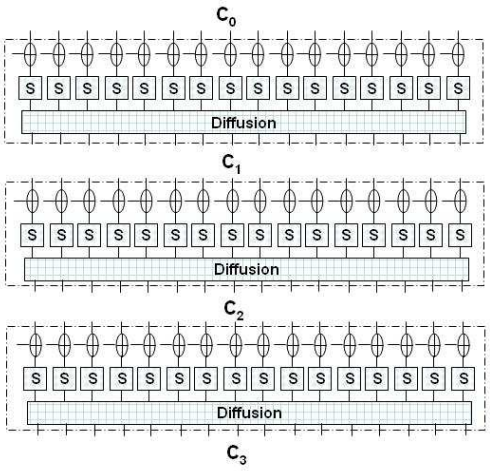


Figure 5.4: Three round trail of the Pyramid cipher

### 5.2.2 Resistance to square attack

The proposed ciphers also comprises of byte oriented operations which are loosely based on the HD cipher. Here we show the application of Square attack on four round Pyramid cipher. However, extending this attack to five rounds has not been possible.

#### Square attack on four round Pyramid cipher

The Square attack [12] (also known as Integral attack [32] or the Saturation attack [38]) utilizes the byte oriented nature of the Square block cipher. As AES is also a byte oriented cipher, this attack has been extended to reduced versions of AES [37, 20]. The proposed cipher also comprises of byte oriented operations which are

loosely based on the AES and Square [12] ciphers. Here we show the application of Square attack on four round Pyramid cipher. However, extending this attack to five rounds has not been possible.

**Square attack on four round Pyramid cipher**

We now describe some notations, that are similar to those used in the original Square attack. Let $\Lambda$-set be a set of 256 states that are all different in some of the state bytes (the active) and all equal in the other state bytes (the passive). Let $\lambda$ be the set of indices of the active bytes. We have,

$$\forall x, y \in \Lambda : \begin{cases} x_{i,j} \neq y_{i,j} & \text{for } (i,j) \in \lambda \\ x_{i,j} = y_{i,j} & \text{for } (i,j) \notin \lambda \end{cases} \tag{5.2.0}$$

Consider a $\Lambda$-set in which all 16 bytes are active. After the first round, the minimum number of active bytes in the $\Lambda$-set is 1. This is because, the branch number of MC matrix in round one is 17. However, at the end of second round the $\Lambda$-set will have all 16 bytes active. This is still the case at the input to the third round. Let $a_i$, $b_i$, $i \in \{1...16\}$ denote the $\Lambda$-set at the input and the output of the third round respectively. Then,

$$\begin{aligned} \bigoplus_{a \in \Lambda} b_i &= \bigoplus (a\mathcal{G}_{\mathrm{MC}}) \\ &= g_{1,i} \bigoplus_{a \in \Lambda} a_1 \oplus g_{2,i} \bigoplus_{a \in \Lambda} a_2 \oplus \cdots \oplus g_{16,i} \bigoplus_{a \in \Lambda} a_{16} \\ &= 0 \oplus 0 \oplus \cdots \oplus 0 = 0 \end{aligned}$$

Since the bytes at the input to the third round range over all possible values, they are balanced over the $\lambda$ set. Due to this, the balance property is preserved at the end

of round three [12]. However, the substitution operations in round four destroys the balance property.

In order to perform the four round attack, 256 plaintexts which differ in all the 16 byte positions are selected. The ciphertexts for these plaintexts are obtained. The first few bytes of the fifth and the sixth round key are guessed. The intermediate cipherstate at the end of third round is calculated for all the known ciphertexts. The balancedness of the derived intermediate cipherstate is tested. If the cipherstate is balanced, then the guessed sub-key is correct with a high probability. Although this attack works well for four round Pyramid cipher, it has not been possible to meaningfully extend it to five rounds.

## 5.3 Error Correcting Capacity

Based on minimum distance decoding and Theorem 5.3.1 the error correcting capacity of the Pyramid cipher is four bytes.

**Theorem 5.3.1.** *The error correcting capacity of the Pyramid cipher consisting of [24,16,256] HD code is four bytes.*

*Proof.* As the substitution and the key XOR operations are performed one byte at a time, a byte of ciphertext in error before decryption will translate to an errored byte at the same exact location after key XORs and inverse substitution but before the HD decoding operation. HD decoding is the only error correcting operation performed in the Pyramid cipher. Therefore the byte error correcting capacity of the Pyramid cipher is directly related to that of the HD code used in the fourth round. All HD codes are MDS codes (Theorem 3.3.2), therefore the $[24, 16, 256]$ HD code should satisfy the Singleton bound with equality. The minimum distance, $d_{\min}$, between any

two codewords in the $[24, 16, 256]$ HD code is therefore,

$$d_{\min} = 24 - 16 + 1 = 9 \tag{5.3.0}$$

The error correcting capacity, $t$, of any linear block code is $\llcorner \frac{d_{\min}}{2} \lrcorner$ symbols. Therefore, the error correcting capacity of the $[24, 16, 256]$ HD code and hence the Pyramid cipher is $\llcorner \frac{9}{2} \lrcorner = 4$ bytes. □

## 5.4 Decoding Procedure

We use the Euclidean error and erasure correcting decoding procedure as described in [36] with slight modification. Let $v(X)$ and $r(X)$ represent the transmitted and the received codewords respectively. The error pattern is $e(X) = r(X) - v(X)$. As $v(X) = m(X)g(X)$, where $m(X)$ is the message and $g(X)$ is the generator polynomial of $[32, 16, 256]$ shortened RS code. Solutions $\alpha^i$ to $g(X)$ are also solutions to $v(X)$. $g(X)$ has 16 solutions. Therefore, for $1 \leq i \leq 16,\ v(\alpha^i) = 0$. $r(\alpha^i) = v(\alpha^i) + e(\alpha^i) = e(\alpha^i)$ are called as the syndromes, denoted by $S_i$. An all zero syndrome indicates that there are no errors less than or equal to 16 bytes. If there are say $v \leq 8$ errors, we get equations of the form,

$$S_i = e_{j_1}\alpha^{ij_1} + e_{j_2}\alpha^{ij_2} + ... + e_{j_v}\alpha^{ij_v} \tag{5.4.0}$$

For $1 \leq i \leq v$, $\beta_i = \alpha^{j_i}$ are the error locations and $\delta_i = e_{j_i}$ are the error values. The first step in error correction is to determine the error locations. We form the error location polynomial $\sigma(X)$ such that $\beta_i$'s are the solutions of this polynomial. That is,

$$\sigma(X) = (1 - \beta_1(X))(1 - \beta_2(X)) \cdots (1 - \beta_v(X)) \tag{5.4.0}$$

The syndrome polynomial is given by,

$$S(X) = S_1 + S_2X + \cdots + S_{17}X^{16} + \cdots \tag{5.4.0}$$

However, only the first 16 coefficients in (5.4) are known. The first 16 terms in the expansion $\sigma(X)S(X)$ are denoted by $Z_0(X)$. That is, $Z_0(X) = [\sigma(X)S(X)]_{16}$. Therefore, unique pair of solutions to

$$\sigma(X)S(X) \equiv Z_0(X) \mathrm{mod} X^{16} \tag{5.4.0}$$

can be used to correct $v \leq 8$ errors in the received codeword. Either the Berlekamp's algorithm [4] or the Euclidean algorithm [36] can be used to solving the key equation (5.4).

However, the HD codes we use in the Pyramid cipher are obtained by puncturing 24 bytes of the shortened $[32, 16, 256]$ RS code. This puncturing will result in poor error correction if the above decoding procedure is used directly. The punctured codeword bytes are called erasures. Since, we know the location of erasures, we can construct the erasure location polynomial $\beta(X) = \prod_{l=1}^{24+e}(1 - \alpha^{j_l}(X))$, where $e$ is the number of additional erasures that may have occurred beyond the puncturing used to obtain HD codes from shortened RS codes. In the received codeword, the location of erasures is substituted by zeros. This introduces $24 + e$ errors additional to $v$ errors that are channel induced. However we only need to solve for $v$ error locations. The modified key equation is now,

$$\sigma(X)\beta(X)S(X) \equiv Z_0(X) \mathrm{mod} X^{16} \tag{5.4.0}$$

Since, $S(X)$ and $\beta(X)$ are known, we can calculate $T(X) = [\beta(X)S(X)]_{16}$. Equation 5.4 can be reduced to,

$$\sigma(X)T(X) \equiv Z_0(X) \mathrm{mod} X^{16} \tag{5.4.0}$$

Berlekamp's or Euclidean algorithm can be used to solve this equation to get the location and value of $v \leq 4, e = 0$ errors or $e \leq 8, v = 0$ erasures or $e/2 + v \leq 4$ errors and erasures.

## 5.5 Modes of Operation

In this section we construct and analyze the energy efficiency and error resilience of the Pyramid cipher in Cipher Block Chaining (CBC) block mode and counter (CTR) stream mode.

### 5.5.1 Cipher block chaining (CBC) mode

In the CBC mode, every plaintext block is XORed with the previous ciphertext block before encryption. The first plaintext block is XORed with an Initialization Vector (IV). We implement HD cipher in CBC mode and compare it with traditional concatenated systems in terms of energy efficiency and error resilience.

We use the testbed consisting of an Intrinsyc CerfCube [27] as described in Section 4.4.1. We measured the energy consumed for one million encryptions/decryptions of Pyramid and the AES cipher and one million RS encoding and decoding operations. The measurements are given in Table 5.1. The measured energy consumption is divided by 16 million to obtain the per byte energy consumption. The per byte energy consumed by Pyramid and AES operating in block modes like ECB and CBC is about 0.8 $\mu J$ and 0.49 $\mu J$ respectively. The energy consumption per byte by the concatenated system, AES cipher followed by RS code (AES-RS), is 0.9 $\mu J$. This shows that our proposed joint approach to encryption and encoding is 10% more energy efficient compared to the traditional disjoint approach. Similarly, the Pyramid decryption is 6% more energy efficient compared to the concatenated AES-RS decoding and decryption.

To evaluate the error resilience of the Pyramid cipher, we perform simulations with both burst error and uniform error channel models and compare it with the

| Block Mode | Voltage (volts) | Current (amps) | Time (secs) | Energy (Joules) |
|---|---|---|---|---|
| Pyramid Encryption | 5.003 | 0.238 | 11.2 | 13.14 |
| AES Encryption | 5.003 | 0.237 | 6.6 | 7.825 |
| RS Encoding | 5.003 | 0.237 | 5.8 | 6.87 |
| Pyramid Decryption | 5.003 | 0.239 | 23.7 | 28.33 |
| AES Decryption | 5.003 | 0.238 | 8.2 | 9.76 |
| RS Decoding | 5.003 | 0.238 | 17.2 | 20.48 |

Table 5.1: Voltage, current, time and energy measurements for the one million Pyramid, AES and RS encryption/encoding and decryption/decoding operations.

error resilience of AES concatenated with a) Reed Solomon codes (AES-RS), b) Convolutional codes (AES-Conv) and c) Low Density Parity Check (AES-LDPC) codes.

Wireless communication medium is characterized by bursty errors and fading phenomenon. Which implies that the bit errors occurring in wireless channels have memory. [2] proposed an additive Markov channel (AMC) model for slow fading wireless channels. According to this model, the channel can be described by bit error rate and correlation parameters. The burstyness of the channel can be controlled by the correlation parameter. In our experiments we set the correlation to 0.8 and varied the bit error rate from $10^{-3}$ to $5 \times 10^{-1}$.

First, we compare the Pyramid cipher with AER-RS under the AMC channel conditions. We use $[24, 16, 256]$ RS codes in AES-RS to maintain the same coding rate as the Pyramid cipher. Fig. 5.5 plots the post decryption bit error rate of the Pyramid cipher and AES-RS. We can observe that both the systems perform comparably under all the channel conditions. This shows that there is no loss or gain of error resilience due to joint error correction and encryption.

Next, we compare the Pyramid cipher with AES-CONV under the AMC channel conditions. The convolutional codes with rates 0.5, 0.33, 0.25, 0.2, 0.167 are used.

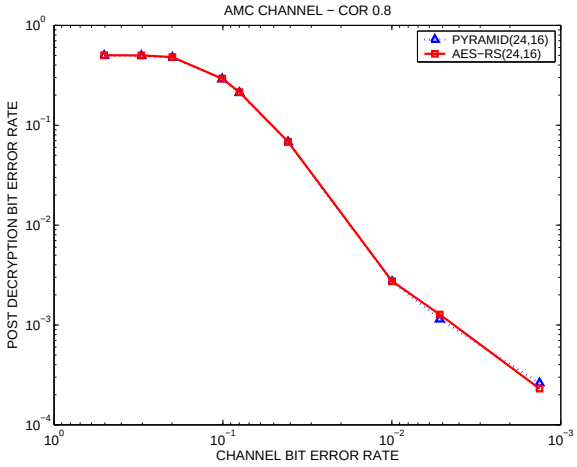


Figure 5.5: Post decryption BER of PYRAMID and AES concatenated with [24,16,256] RS codes under AMC channel model with correlation 0.8.

Since, the convolutional codes are not burst error correcting codes, we use an interleaver with a depth of 16 to improve its error correcting capacity. Fig. 5.6 plots the post decryption bit error rate of Pyramid cipher and AES-CONV for different coding rates. We can observe that Pyramid cipher with a coding rate of 0.67 clearly outperforms AES-CONV with coding rates 0.5, 0.33 and 0.25. This shows that although convolutional codes are lighter than RS and HD codes, they do not perform as well under bursty channel conditions (like in wireless medium). We can observe from the Fig. 5.6 that the convolutional codes require about 60% more redundancy to equal the performance of the Pyramid cipher.

Finally, we compare the Pyramid cipher and AES-LDPC with similar coding rate. The LDPC codes are known to perform extremely well in non-bursty (uniformly distributed errors) channels. Therefore, for this simulation, we use binary symmetric channel [66] model to generate uniformly random errors. The LDPC decoding is an iterative process and the error resilience of LDPC codes improve with the number of iterations. However, the energy spent in decoding is also proportional to the number

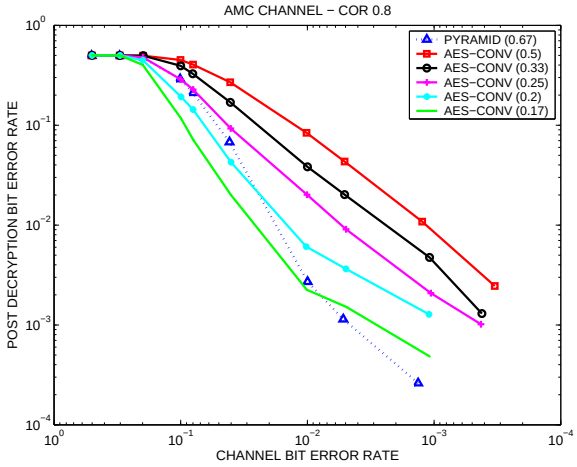


Figure 5.6: Post decryption BER of PYRAMID and AES concatenated with Convolutional codes under AMC channel model with correlation 0.8.

of iterations. Hence in energy constrained systems it may not be feasible to perform too many iterations. In the concatenated system AES-LDPC we first use 2 step LDPC decoding (which is as heavy as HD decoding) and then repeat the experiment with 250 step LDPC decoding. Fig. 5.7 plots the post decryption bit error rates of Pyramid cipher and AES-LDPC. We can observe that the performance of the Pyramid cipher is in between the 2 step and 250 step LDPC decoding. This shows that even in binary symmetric channel conditions, the performance of the Pyramid cipher is comparable to LDPC codes.

### 5.5.2 Counter (CTR) mode

In the CTR mode, the block cipher is used to encrypt a counter value which is incremented for successive encryptions. The encrypted counter values makeup a pseudo-random keystream which is XORed with the plaintext bits to produce the ciphertext

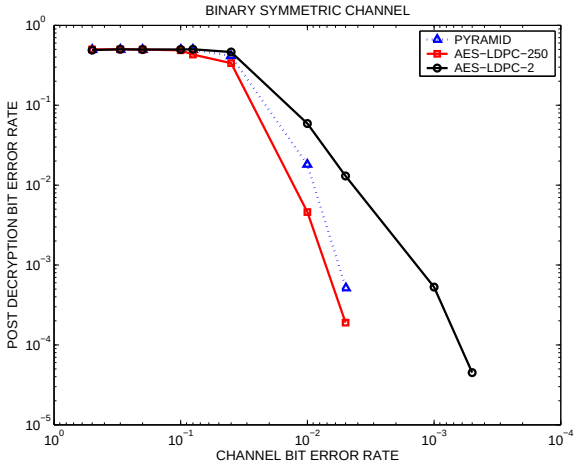


Figure 5.7: Post decryption BER of PYRAMID and AES concatenated with LDPC codes under BSC channel model.

bits. Block ciphers act as pseudorandom number generators (PRNGs). Since, the encryption of plaintext takes place one bit at a time, this mode of encryption is usually referred to as the stream mode. Pyramid cipher can be used in CTR mode as well. However, the error correction property of the Pyramid cipher cannot be used when it is operated in this mode. An advantage of using the Pyramid cipher in CTR mode is that the expansion of cipherstate during encryption results in higher encryption throughput compared to the AES cipher.

we slightly improvise the encryption throughput of the Pyramid cipher by using $[36, 16, 256]$ HD codes in round four instead of $[24, 16, 256]$. This will increase the encryption throughput to 256 bits. Since only encryption is performed in stream modes, we do not consider the decoding and decryption procedures for the enhanced Pyramid cipher. We empirically test the quality (randomness) of the pseudorandom keystreams generated by the enhanced Pyramid cipher using the National Institute of Standards and Technology (NIST) recommended DIEHARD battery of statistical tests. The DIEHARD test suite [40] consists of a variety of statistical tests like, the

| CTR Mode | Energy/byte ($\mu$ Joules) |
|---|---|
| Pyramid (36) | 0.50 |
| AES | 0.49 |

Table 5.2: Energy consumption per byte for the enhanced Pyramid cipher and the AES cipher operating in CTR mode.

birthday spacings, overlapping permutations, ranks of matrices, count the 1s, minimum distance test, random spheres test and runs test. We implement the Pyramid cipher in the CTR mode and initialize the seed to an all zero value. The secret key is initialized to a random 192 bit value. Using this setup, we generate about 10 MB of pseudorandom sequence. The DIEHARD statistical test suite is executed on the pseudorandom sequence. These tests are then repeated for different seed values. Experimental results reveal that, for all the chosen seed values, the pseudorandom sequence generated by the Pyramid cipher passed each of the statistical tests. This shows that, the Pyramid cipher is a cryptographically secure pseudorandom number generator.

We calculate the energy consumption of the enhanced Pyramid cipher on the testbed (for details on the testbed see Section 4.4.1). The per byte energy consumption for the Pyramid and the AES used in stream modes are given in Table 5.2. We can observe that although the Pyramid and the AES ciphers consume almost the same amount of energy per byte, the Pyramid cipher has twice the encryption throughput compared to the AES cipher.

## 5.6 Conclusions

The Pyramid cipher is a major improvement over the HD cipher in terms of round reduction. In comparison with the popular ten round AES cipher, the Pyramid cipher uses larger diffusion operations, consists of only five rounds, corrects four bytes of errors per ciphertext block during decryption (no error correction in AES) and has a variable encryption throughput $\geq 192$ bits. The bounds on the error and erasure correcting capacity of the Pyramid cipher showed that they are as efficient as the RS codes. Energy consumption analysis and experiments on the 32 bit testbed revealed that the Pyramid cipher used in block modes consume $6-10\%$ lesser energy compared to the concatenated system: AES followed by RS codes. Employing the Pyramid cipher in the stream mode we showed that they are good pseudorandom number generators and have a higher encryption throughput compared to the AES cipher.

# Chapter 6

# Summary

We identified error resilience, energy efficiency, encryption throughput and speed of encryption as four main challenges facing present day secure wireless communications in resource constrained environments. These challenges serve as motivational factors to combine error correction with encryption, which forms the basis of our approach. Although codes and ciphers have contrasting properties, we concentrated on the similarities in order to combine them. In particular, we identified the property of diffusion that is exhibited by most block codes and required by most block ciphers. We proposed the High Diffusion codes that possess the best possible diffusion and yet satisfy the Singleton bound for the minimum distance between codewords thus making them ideal candidates for error resilient cryptographic primitives. Although there is no systematic technique to generate HD codes, the flexibility to generate HD generator matrices from RS generator matrices makes it easy to derive large HD codes without having to go through brute force search. The close relationship of HD codes with the popular Reed Solomon codes makes them easy to study, analyze and port into existing systems.

The construction of the HD cipher from HD codes proves the latter's potential as building blocks for ciphers. The high branch number of HD codes used the HD

cipher was a key factor in achieving resistance against linear and differential attacks. The error resilience analysis of HD cipher revealed that the error correction capacity of HD codes are not compromised because of their use inside the cipher. The joint security and error resilience properties favors HD cipher as a potential solution to current wireless security challenges. The HD cipher is employed both in block and stream modes. In block mode, the HD cipher is $12 - 30\%$ more energy efficient and performs equivalently in terms of error correction compared to traditional systems. Furthermore, HD cipher when used in stream mode has more encryption throughput compared to the popular AES cipher. Energy consumption analysis revealed that HD cipher is 40% more energy efficient. The error resilience, higher encryption throughput and energy efficiency properties makes the proposed HD cipher an ideal replacement for the AES cipher in the CCMP protocol.

The Pyramid cipher is a major improvement over the HD cipher in which we could achieve reduction in the number of rounds along with error resilience, higher encryption throughput and energy efficiency. The bounds on the error and erasure correcting capacity of the Pyramid cipher showed that they are as efficient as the RS codes. Through simulations we showed that the Pyramid cipher performs comparably with RS and LDPC codes and outperforms convolutional codes by 60%. We also show that the five round Pyramid cipher is as secure as the ten round AES and reduction in the rounds does not compromise the security. By employing the Pyramid cipher in the stream mode, we showed that they are good pseudorandom number generators and have a higher encryption throughput compared to the AES cipher. Energy consumption analysis and experiments on the revealed that the Pyramid cipher used in block modes consume $6 - 10\%$ lesser energy compared to the concatenated system: AES followed by RS codes. In stream modes, both Pyramid and the AES cipher have comparable performance in terms of energy efficiency, however the Pyramid cipher

has a higher encryption throughput.

The construction of the HD and the Pyramid error correcting ciphers from FECs provides a mathematical framework for truly integrating block ciphers with block codes. The extensive set of experiments and simulations show that combining error correction and encryption leads to energy efficient, error resilient and secure ciphers.

# Publications from the Work

1. Chetan Nanjunda Mathur, Karthik Narayan, and K. P. Subbalakshmi, "On the Design of Error-Correcting Ciphers," EURASIP Journal on Wireless Communications and Networking, vol. 2006, Article ID 42871, pp. 1-12, 2006.

2. Chetan N. Mathur, K.P. Subbalakshmi, "Energy Efficient Wireless Encryption," IEEE Globecom Symposium on Network and Information Security Systems, November 2006.

3. M. A. Haleem, Chetan N. Mathur, K. P. Subbalakshmi, "Joint Distributed Compression and Encryption of Correlated Data in Sensor Networks," IEEE Military Communications Conference (MILCOM) 2006.

4. Chetan Nanjunda Mathur, Karthik Narayan, K.P. Subbalakshmi "High Diffusion Cipher: Encryption and Error Correction in a Single Cryptographic Primitive," 4th International Conference on Applied Cryptography and Network Security, June 2006.

5. Chetan Nanjunda Mathur, Karthik Narayan and K.P. Subbalakshmi, "High Diffusion Codes: A Class of Maximum Distance Separable Codes for Error Resilient Block Ciphers," IEEE GLOBECOM Workshop: 2nd IEEE International Workshop on Adaptive Wireless Networks (AWiN), 2005.

# Other Publications

1. Chetan Mathur and K.P. Subbalakshmi, "Digital Signatures for Centralized DSA Networks," Consumer Communications and Networking Conference (CCNC), 2007.

2. Yiping Xing, Chetan Mathur, M.A. Haleem, R. Chandramouli and K.P. Subbalakshmi, "Dynamic Spectrum Access with QoS and Interference Temperature Constraints," IEEE Transactions on Mobile Computing, 2006.

3. M. Haleem, Chetan Mathur, R. Chandramouli and K.P. Subbalakshmi, "On Optimizing the Security-Throughput Trade-off in Wireless Networks with Adversaries," 4th International Conference on Applied Cryptography and Network Security, June 2006.

4. Chetan Mathur and K.P. Subbalakshmi, "Light Weight Enhancement to RC4 Based Security for Resource Constrained Wireless Devices," International Journal of Network Security (IJNS).

5. Yiping Xing, Chetan Mathur, M. Haleem, R Chandramouli and K.P. Subbalakhsmi, "Priority Based Dynamic SPectrum Access with QoS and Interference Temperature Constraints," IEEE International Conference on Communications 2006.

6. Yiping Xing, Chetan Mathur, M. Haleem, R Chandramouli and K.P. Subbalakhsmi, "Real-Time Secondary Spectrum Sharing with QoS provisioning," IEEE Consumer Communications and Networking Conference 2005.

7. Chetan Nanjunda, M Haleem and R. Chandramouli, "Robust Encryption for Secure Image Transmission over Wireless Channels," IEEE International Conference on Communications 2005.

# Bibliography


[1] 802.11i. Amendment 6: Medium Access Control (MAC) Security Enhancements. *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, July 2004.

[2] F. Alajaji and T. Fuja. A communication channel modeled on contagion. *IEEE Transactions on Information Theory*, 40:2035–2041, 1994.

[3] E. Berlekamp, R. McEliece, and H. van Tilborg. On the inherent intractability of certain coding problems. *IEEE Transactions on Information Theory*, May 1978.

[4] E. R. Berlekamp. *Algorithmic Coding Theory*, chapter Ch. 7. New York: McGraw-Hill, 1968.

[5] T. A. Berson. Failure of the McEliece public-key cryptosystem under message-resend and related-message attack. In *Advances in Cryptology-CRYPTO '97, Lecture notes in computer science*, 1997.

[6] E. Biham and A. Shamir. Differential cryptanalysis of Snefru, Khafre, REDOC-II, LOKI and Lucifer (extended abstract). *Lecture Notes in Computer Science*, 576:156, 1991.

[7] E. Biham and A. Shamir. Differential cryptanalysis of the full 16-round des. In *CRYPTO '92: Proceedings of the 12th Annual International Cryptology Conference on Advances in Cryptology*, pages 487–496, London, UK, 1993. Springer-Verlag.

[8] A. Biryukov. The design of a stream cipher LEX. Lecture Notes in Computer Science, 2006.

[9] A. Canteaut. *Attacks on low-weight cryptosystems and construction of t-resilient functions*. PhD thesis, Universit Paris VI, 1996.

[10] X. Chen. *Error-Control Coding for Data Networks*. Kluwer Academic Publishers, Norwell, MA, USA, 1999.

[11] J. Daemen. *Cipher and hash function design strategies based on linear and differential cryptanalysis*. PhD thesis, K.U.Leuven, March 1995.

[12] J. Daemen, L. R. Knudsen, and V. Rijmen. The block cipher square. In *FSE '97: Proceedings of the 4th International Workshop on Fast Software Encryption*, pages 149–165, London, UK, 1997. Springer-Verlag.

[13] J. Daemen and V. Rijmen. The wide trail design strategy. In *Proceedings of the 8th IMA International Conference on Cryptography and Coding*, pages 222–238, London, UK, 2001. Springer-Verlag.

[14] J. Daemen and V. Rijmen. *The Design of Rijndael*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2002.

[15] J. Daemen and V. Rijmen. *The design of Rijndael: AES — the Advanced Encryption Standard*. Springer-Verlag, 2002.

[16] H. Feistel. Cryptography and computer privacy. *Scientific American*, 228(5):15–23, May 1973.

[17] FIPS. Specification for the Advanced Encryption Standard (AES). Federal Information Processing Standards Publication 197, 2001.

[18] S. Fomin and A. Zelevinsky. Total positivity:tests and parameterizations. *Math. Intelligencer*, 22:22–33, 2000.

[19] F. Gantmacher. *The Theory of Matrices*, volume 2. Chelsa Publishing Company, New York, 1964.

[20] H. Gilbert and M. Minier. A Collision Attack on 7 rounds of Rijndael. In *AES Candidate Conference*, pages 230–241, 2000.

[21] D. Gligoroski, S. Knapskog, and S. Andova. Cryptcoding - encryption and error-correction coding in a single step. *International Conference on Security and Management*, June 2006.

[22] W. Godoy and D. Periera. A proposal of a cryptography algorithm with techniques of error correction. *Computer Communications*, 20(15):1374–1380, 1997.

[23] O. Goldreich and R. Ostrovsky. Software protection and simulation on oblivious RAMs. *Journal of the ACM*, 43(3):431–473, 1996.

[24] R. W. Hamming. Error-detecting and error-correcting codes. *Bell System Technical Journal*, 29(2):147–160, 1950.

[25] R. Hill. *A First Course in Coding Theory*. Oxford University Press, 1986.

[26] T. Hwang and T. Rao. Secret Error-Correcting Codes (SECC). In *Advances in Cryptography - Crypto 1988*, 1988.

[27] Intrinsyc. http://www.intrinsyc.com/products/mob_ref_sys/cerfcube_255/.

[28] J. Jonsson. On the Security of CTR + CBC-MAC. In *Selected Areas in Cryptography*, pages 76–93, 2002.

[29] J. P. Jordan. A variant of a public key cryptosystem based on Goppa Codes. *SIGACT News*, 15(1):61–66, 1983.

[30] S. C. Kak. Joint encryption and error-correction coding. In *SP '83: Proceedings of the 1983 IEEE Symposium on Security and Privacy*, page 55, Washington, DC, USA, 1983. IEEE Computer Society.

[31] L. R. Knudsen. *Block ciphers - Analysis, Design and Applications*. Ph. d. thesis, Aarhus University, Denmark, 1994.

[32] L. Kundsen and D. Wagner. Integral cryptanalysis. *Lecture Notes in Computer Science*, 2365:112, Jan 2002.

[33] X. Lai. *On the Design and Security of Block Ciphers*. Phd thesis, ETH, Zurich, Switzerland, 1992.

[34] T. Li and G. Wang. Security analysis of two ultra-lightweight RFID authentication protocols. In *IFIP SEC 2007*, Sandton, Gauteng, South Africa, May 2007. IFIP.

[35] M. C. Lin and H. L. Fu. Information rate of McEliece's public-key cryptosystem. *Electronics Letters*, 26(1):16–18, 1990.

[36] S. Lin and D. J. Costello. *Error Control Coding, Second Edition*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2004.

[37] S. Lucks. Attacking Seven Rounds of Rijndael under 192-bit and 256-bit Keys. In *AES Candidate Conference*, pages 215–229, 2000.

[38] S. Lucks. The Saturation Attack - A Bait for Twofish. *Lecture Notes in Computer Science*, 2355:1, Jan 2002.

[39] F. J. MacWilliams and N. J. A. Sloane. *The theory of error-correcting codes. I and II*. North-Holland Publishing Co., Amsterdam, 1977. North-Holland Mathematical Library, Vol. 16.

[40] G. Marsaglia. A current view of random numbers. in computer science and statistics. *Proceedings of the Sixteenth Symposium on The Interface*, pages 3–10, 1985.

[41] C. Mathur, K. Narayan, and K. Subbalakshmi. High diffusion codes: A class of maximum distance separable codes for error resilient block ciphers. *2nd*

*IEEE International Workshop on Adaptive Wireless Networks (AWiN), Globecom*, November 2005.

[42] C. Mathur, K. Narayan, and K. Subbalakshmi. High diffusion cipher: Encryption and error correction in a single cryptographic primitive. Applied Cryptography and Network Security Conference (ACNS), June 2006.

[43] C. Mathur and K. Subbalakshmi. Energy efficient wireless encryption. *IEEE Globecom Symposium on Network and Information Security Systems*, November 2006.

[44] C. Mathur and K. Subbalakshmi. Light Weight Enhancement to RC4 Based Security for Resource Constrained Wireless Devices. *Accepted for publication in International Journal of Network Security (IJNS)*, 2007.

[45] M. Matsui. Linear cryptanalysis method for DES cipher. In *Advances in cryptology -EUROCRYPT93, Lecture Notes in Computer Science*, volume 765, pages 1–11, 1993.

[46] R. McEliece. A publick key cryptosystem based on algebraic codes. *DNS Progress Reports 42-44, NASA Jet Propulsion Labaratory*, 1978.

[47] A. J. Menezes, S. A. Vanstone, and P. C. V. Oorschot. *Handbook of Applied Cryptography*. CRC Press, Inc., Boca Raton, FL, USA, 1996.

[48] C. Nanjunda, M. Haleem, and R. Chandramouli. Robust encryption for secure image transmission over wireless channels. In *ICC' 2005, IEEE International Conference on Communications, May 16-20, 2005 - Seoul, Korea*, 2005.

[49] K. Narayan. On the design of secure error resilient diffusion layers for block ciphers. Master's thesis, Steven Institute Of Technology, Hoboken, New Jersey, May 2005.

[50] K. Nyberg. Differentially uniform mappings for cryptography. In *EUROCRYPT '93: Workshop on the theory and application of cryptographic techniques on Advances in cryptology*, pages 55–64, Secaucus, NJ, USA, 1994. Springer-Verlag New York, Inc.

[51] Oprofile. http://oprofile.sourceforge.net.

[52] C. S. Park. Improving code rate of McEliece's public-key cryptosystem. *Electronics Letters*, 25(21):1466–1467, 1989.

[53] A. Perrig, R. Canetti, J. D. Tygar, and D. Song. The TESLA Broadcast Authentication Protocol. *RSA CryptoBytes*, 5(Summer), 2002.

[54] A. Poschmann, G. Leander, K. Schramm, and C. Paar. A family of light-weight block ciphers based on DES suited for RFID applications. Printed handout of Workshop on RFID Security – RFIDSec 06, July 2006.

[55] V. Rijmen, J. Daemen, B. Preneel, and A. Bosselaers. The Cipher Shark. *Lecture Notes in Computer Science*, 1039:99, 1996.

[56] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, and N. Ferguson. Comments on Twofish as an AES Candidate. *AES Candidate Conference*, pages 355–356, 2000.

[57] N. Sendrier. Efficient generation of binary words of given weight. In *Fifth IMA Conference on Cryptography and Coding*, 1995.

[58] C. E. Shannon. Communication theory of secrecy systems. *Bell Systems Technical Journal*, 28(4):656—715, 1949.

[59] W. Stallings. *Cryptography and network security (2nd ed.): principles and practice*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1999.

[60] D. Stinson. *Cryptography: Theory and Practice,Second Edition*. CRC/C&H, 2002.

[61] H. M. Sun. Private-key cryptosystem based on burst-error-correcting codes. *Electronics Letters*, 33:2035–2036, Nov. 1997.

[62] H. M. Sun. Enhancing the security of the McEliece public-key cryptosystem. *Journal of Information Science and Engineering*, 16:769–812, Nov. 2000.

[63] A. S. Tosun and W. chi Feng. On error preserving encryption algorithms for wireless video transmission. In *MULTIMEDIA '01: Proceedings of the ninth ACM international conference on Multimedia*, pages 302–308, New York, NY, USA, 2001. ACM Press.

[64] H. van Tilborg. Coding theory at work in cryptology and vice versa. *Handbook of Coding Theory*, pages 1195–1227, 1998.

[65] J. Walker. 802.11 Security Series Part II: The Temporal Key Integrity Protocol (TKIP). *Technical report, Platform Networking Group, Intel Corporation*, 2001.

[66] S. B. Wicker. *Error control systems for digital communication and storage*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1995.

[67] K. Zeng, C. H. Yang, and T. R. N. Rao. Cryptanalysis of the hwang-rao secret error correcting code schemes. In *Third International Conference in Information and Communicatoins Security, ICICS 2001*, volume 2229. Lecture Notes in Computer Science, Springer-Verlag, 2001. Obtained online at http://crypto.nknu.edu.tw/publications/icics2001.pdf.