CHAPTER 4

DESIGN OF BLOCK CIPHER

4.1 Introduction

The analysis of block cipher in Chapter 3 lends support to the idea of a block cipher with an SP network that uses algorithmic S-Boxes to provide confusion and linear transformation to provide diffusion. In this chapter, the proposed block cipher is presented.

The Data Encryption Standard is the most widely used and successful encryption algorithm in the world. Despite its popularity, DES is nearing the end of its useful life for many applications. Its 56-bit key is too short for commercial security, as shown by advances in recent distributed key search techniques. Although Triple-DES can solve the key length problem, it is too slow for applications that require high performance. The 64-bit block size used by DES and most other popular encryption algorithms makes it vulnerable to attacks when large amount of data are encrypted under the same key.

To meet the requirements of future applications, we propose a new block cipher with improved performance and security. It is a block cipher with a 128-bit block size and a 256-bit key. With the increased key length and block size, the proposed block cipher has greater security over DES. Unlike DES, the proposed block cipher is an iterated block cipher, using a standard substitution-permutation network. It uses S-Boxes that are highly nonlinear to provide confusion to the block cipher. The highly diffusive diffusion layer, which is a combination of linear mixers, ensures that fewer rounds are required to achieve the avalanche effect. Based on the analysis done in Chapter 3, we decided to use an SP network as the structure of the block cipher. This is mainly due to an important drawback of the Feistel structure. Since only half of the bits of the intermediate round output enter the f function in the next round, the round function exhibits a large amount of linearity that can be exploited by differential cryptanalysis and linear cryptanalysis. In addition, diffusion is slower on Feistel cipher because only half of the bits are transformed in each round. Although Feistel structure is not used, the design of the block cipher enables the use of the same round function for both encryption and decryption. This feature is desirable because the reuse of the encryption module for decryption can save memory, space and cost. The block cipher generally uses table lookup and simple Boolean operations, which enhances performance. This leads to a block cipher with good security level and performance.

This chapter is organized as follows. First, the major design principles underlying the proposed block cipher are discussed. This is followed by an overview of the basic building blocks of the block cipher. Next, we discuss the design strategy and explain the selection of the various components of the block cipher, including the structure of the cipher, S-Boxes, diffusion layer and key scheduling.

4.2 Design Principles

The major principles that guide the design of the proposed block cipher are as follows:

a) Resistance against known attacks

Security is the most important criterion in the design of the block cipher. The S-Boxes are carefully selected and analysed to provide resistance against major cryptanalysis techniques. This is achieved by constructing the S-Boxes by the composition of two transformations; a multiplicative inverse in GF(2^8) followed by an affine mapping over GF(2).

b) Simplicity of design

One of the main guiding principles behind the design process is simplicity of the algorithm. Although simplicity is not equitable to security, but a simple design is more amenable to analysis than a more complex design. This enables more thorough analysis on the various cryptographic properties of the algorithm and the resistance against cryptanalysis. The following steps are taken to simplify the design:

i) *Reusability* – The design makes sure that the same primitives can be reused in multiple parts of the cipher. There are many theories in designing secure block cipher. We chose to reuse the same design primitives in the encryption operation and key scheduling. This simplifies the analysis of the algorithm and allows more intensive analysis to be performed.

ii) *Reversibility* - Reversibility of a block cipher enables ciphertext to be decrypted back to plaintext. Although Feistel structure is not used, the design of the block cipher enables the use of the same round function for both encryption and decryption. This is a nice feature since it allows software and hardware module to reuse the same functionality for both encryption and decryption. Analysis of the cipher is simpler since it can be done at once for both operations.

c) Flexibility in implementation

Implementation characteristics are important in the design of the cipher. Flexibility in implementation includes:

i) Ability of the algorithm to handle multiple key and block sizes.

ii) Implementation as hash function, random number generator, stream cipher and other cryptographic services.

iii) Suitability in hardware and software.

d) Performance

Performance plays an important role in the design of the block cipher. Performance can be evaluated based on:

i) Number of rounds

ii) Speed of encryption and decryption

iii) Performance on different platform

e) Strong key scheduling

Key scheduling is an integral part of the cipher design. It is carefully designed to avoid weak keys, complement keys, equivalent keys, etc. Linear key scheduling is not used to prevent exhaustive key search. Nonlinear and highly diffusive key schedule is used to maximize avalanche in the subkeys. This will immunize the key schedule against related-key attacks. The speed of the key scheduling is also taken into consideration.

f) Highly nonlinear substitution layer

The substitution layer uses algorithmic S-Boxes that possess the desired cryptographic properties to provide confusion and nonlinearity in the cipher.

g) Fast diffusive characteristics

The diffusion layer uses four-input-port-four-output-port mixers that mix data from every part of the plaintext block to guarantee the overall diffusion in block cipher. It has fast diffusive characteristic that guarantees changes in one input block will produce changes in all output port.

4.3 Overview of Block Cipher Structure

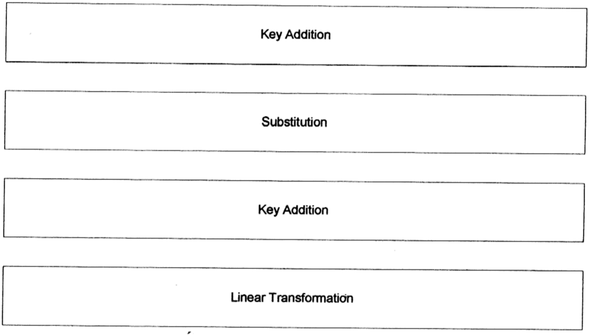

Figure 4.1 Basic Building Block

Figure 4.1 shows the basic building block of the proposed block cipher. It is an iterated cipher that uses a 12-round Substitution-Permutation network, not the Feistel structure. In Feistel structure, commonly used by most ciphers, part of the bits of the intermediate state are transposed without any changes to the other position.

Using SP network, a uniform transformation can be applied to the whole block. Unlike typical SP network where the decryption process is performed by inverting the round transformation, the cipher is designed such that the encryption and decryption can be performed using the same operation. This feature greatly simplifies implementation and analysis of the cipher.

4.3.1 The Round Transformation

The round transformation of the proposed block cipher consists of four uniform transformation layers:

i) Round Key addition layer

ii) Substitution layer

iii) Round Key addition layer

iv) Linear transformation layer

The round transformation is shown in Figure 4.2.

The final round is slightly different. It is composed of:

i) Round Key addition layer

ii) Substitution layer

iii) Round Key addition layer

The linear transformation layer is omitted to ensure that the encryption and decryption routines have the same structure. This does not improve nor reduce the security of the block cipher. This is similar to the omission of swap function in the last round of DES. The function of every layer in the round transformation is described in the following sections.

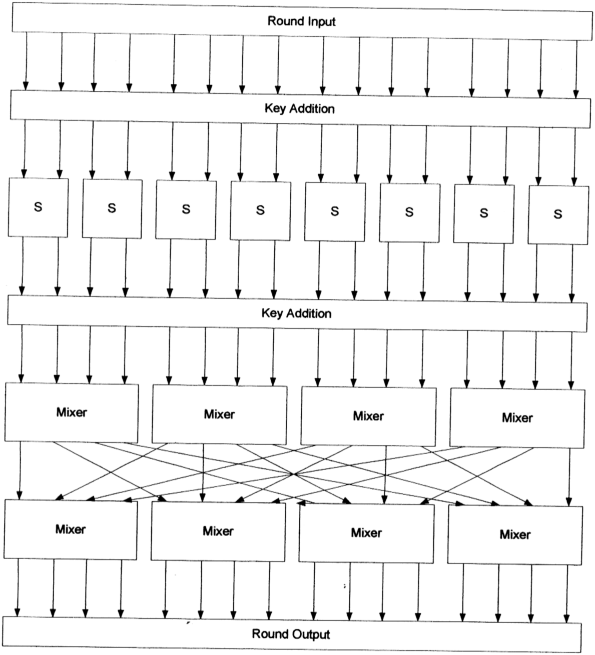

Figure 4.2 The Round Transformation

4.3.2 Substitution Layer, γ

The substitution layer is a nonlinear substitution that provides confusion and nonlinearity properties of the cipher. It consists of an algorithmic 8 x 8 bit S-Box, S0 and its inverse, S1. The S-Box is constructed by the composition of two transformations:

i) The multiplicative inverse over GF(2^8)

- This is based on the proposal of Nyberg on several classes of nonlinear S-Boxes [35]. It is a mapping of $f(x) = x^{-1}$ over GF(2^m). In this case, m = 8. This type of mapping is used in several block ciphers, including SHARK, SQUARE and Rijndael.

ii) An affine transformation over GF(2)

- Its purpose is to remove the fixed point exists in the above transformation.

The inverse of this S-Box is obtained by the inverse affine mapping followed by the multiplicative inverse over GF(2^8). Each S-Box is used in alternate rounds. In each round, the input block is split into 16 8-bit blocks. The S-Box takes the 8-bit input to produce an 8-bit output. S0 is used in odd rounds while its inverse, S1 is used in even rounds. This ensures that the encryption and decryption process can be performed using the same function.

4.3.3 Round Key Addition, σ

This consists of the bitwise addition of a round key, k_r.

$$\sigma[\, k_r] : x \; = \; x \oplus k_r$$

The inverse of $\sigma[\, k_r]$ is $\sigma[\, k_r]$ itself.

4.3.4 Linear Transformation, π

Linear transformation is the diffusion layer of the cipher. It can be considered as a mixer that takes four 8-bit inputs and performs mixing operation so that an overall diffusion can be achieved in the whole block. The four-input-four-output mixer is shown in Figure 4.3.

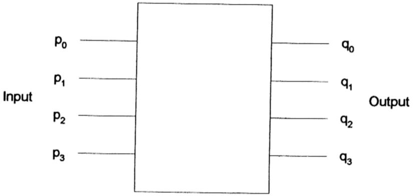

Figure 4.3 Mixer

The mixer operation is performed over a finite field, GF(2^8) modulo an irreducible polynomial, c(x), with c(x) given by

$$c(x) = x^8 + x^4 + x^3 + x^2 + 1$$

The mixer is formed by the following equations:

$$q_0 = 3p_0 \oplus 2p_1 \oplus 2p_2 \oplus 2p_3$$

$$q_1 = 2p_0 \oplus 3p_1 \oplus 2p_2 \oplus 2p_3$$

$$q_2 = 2p_0 \oplus 2p_1 \oplus 3p_2 \oplus 2p_3$$

$$q_3 = 2p_0 \oplus 2p_1 \oplus 2p_2 \oplus 3p_3$$

This can be represented as a matrix multiplication, Q = **M** ⊗ P.

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} 03 & 02 & 02 & 02 \\ 02 & 03 & 02 & 02 \\ 02 & 02 & 03 & 02 \\ 02 & 02 & 02 & 03 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix}$$

where the arithmetic is modulo 285.

The matrix M is self-inverse, where $M^{-1} = M$. This property allows the same function to be used in both encryption and decryption. The selection of this matrix is explained in the design strategy in Section 4.4.

4.3.5 Key Schedule, φ

The purpose of the key schedule is to derive the round keys for the iterated round transformation. The round key size is equal to the block length multiply by twice the number of rounds since two round keys are needed for each round. For a 12-round cipher with a block size of 128 bits, the total key length required is equal to 3072 bits or 96 words. The key schedule will expand the cipher key into an expanded key of 96 words.

Key Expansion

The round keys are defined as an array of 16-byte keys, denoted by k_0, …, k_{24}. The first 2 keys contain the cipher key. The other words are defined recursively through the key expansion operation. The operation consists of three transformations as follows:

i) Substitution, γ

ii) Linear transformation, π

iii) Key Addition, σ

The substitution layer uses the same S-Boxes as in the encryption and decryption process. The linear transformation produces diffusion by reusing the same mixer. The overall transformation is as follows:

$$k_i = \sigma[k_{i-2}] \circ \pi \circ \gamma[k_{i-1}]$$

where i = 3, 4, ..., 24

4.3.6 The Cipher

The block cipher consists of:

- Key expansion
- R-1 round (R=12)
- A final round

4.3.7 The Inverse Cipher

The inverse of the cipher can be done without any changes to the round function. The same round function can be used for decryption of ciphertext. The round transformation of the inverse cipher is given by:

i) Round Key Addition

ii) Substitution

iii) Round Key Addition

iv) Linear Transformation

The final round is given by:

i) Round Key Addition

ii) Substitution

iii) Round Key Addition

This is the same routine as the encryption round with the exception that round subkeys must be provided in reverse order.

4.4 Design Strategy

In the design of the proposed block cipher, the round transformation is divided into three major components:

- Substitution layer
- Linear transformation
- Key scheduling

Each of these components has their own functionalities and properties. The substitution layer is selected to have high nonlinearity to provide resistance against linear and differential cryptanalysis. Diffusion characteristic of the cipher is provided by a uniform and highly diffusive linear transformation layer. Key scheduling provides round keys for each round transformation. By clearly defining the functionality of each component, the analysis of the cipher can be done separately on each component. This greatly increases the efficiency of analysis, and the strength and weaknesses of the cipher can be identified more easily.

In the following sections, the clear criterion is specified for each of the building blocks and the motivation behind the design choice is explained. The design of the block cipher started by selecting the desired cipher structure. Then, each component is then selected based on the design criteria specified.

4.4.1 The Cipher Structure

The proposed block cipher uses SP network. Although Feistel network is the most commonly used and most studied structure for block cipher, SP network was selected because of its uniform round structure, consisting of a substitution layer and a diffusion layer. The uniform structure of the SP network means that diffusion can be achieved faster using a highly diffusive linear transformation compared to Feistel network. For a Feistel cipher, one round only operates on half of the bits in the cipher block. This means that the cipher require more rounds to achieve full diffusion compared to a cipher using SP network. Besides that, the fact that half of the block is kept constant in each round may result in a large amount of linearity in the round function, which leaves the cipher vulnerable to linear and differential cryptanalysis. This is an undesired property and should be avoided.

Although Feistel structure provides the advantage that the cipher is invertible regardless of the function F chosen, the same feature can be achieved in SP network by careful design of the round transformation. This can be done by using involution function which is a self-reverse function. The round transformation of the proposed cipher consists of four layers:

- Key addition layer, σ

- Substitution layer, γ

- Key addition layer, σ

- Linear transformation, π

The key addition operation is self-inverse since it is a simple bitwise XOR operation. The decryption process can be performed using the same functions in the reverse order without the need of separate inverse functions if all the functions are involution. The proposed cipher does not use an involution S-Box because it does not satisfy the required design criteria. The proposed cipher is an involution cipher which takes plaintext to ciphertext, and ciphertext to plaintext, using the exact same operation. This can be done by selecting two S-Boxes which are inverse of each other. They are used in alternate rounds, one is used in the odd rounds while the other in the even rounds. The total number of rounds must be even. The sequences of a twelve-round encryption routine and a twelve-round decryption routine is as follows:

Round 1	σ	σ
	γ	γ
	σ	σ
	π	π
Round 2	σ	σ
	γ^{-1}	γ^{-1}
	σ	σ

.	π	π
.	.	.
.	.	.
	.	.
Round 11	σ	σ
	γ	γ
	σ	σ
	π	π
Round 12	σ	σ
	γ^{-1}	γ^{-1}
	σ	σ

Figure 4.4 Encryption **Figure 4.5** Decryption

Let's denote a normal round transformation as ρ, where

$$\rho[k] = \pi \circ \sigma[k] \circ \gamma \circ \sigma[k]$$

and

$$\rho^{-1}[k] = \sigma^{-1}[k] \circ \gamma^{-1} \circ \sigma^{-1}[k] \circ \pi^{-1}$$

$$= \sigma[k] \circ \gamma^{-1} \circ \sigma[k] \circ \pi$$

since $\sigma = \sigma^{-1}$ and $\pi = \pi^{-1}$

The final round transformation, β is given by

$\beta[k] = \sigma[k] \circ \gamma \circ \sigma[k]$

Encryption and decryption are denoted as *E* and *D* respectively. Encryption can be represented as

$E[k] = \beta[k_{12}] \circ \rho[k_{11}] \circ \rho[k_{10}] \circ \rho[k_9] \circ \rho[k_8] \circ \rho[k_7] \circ \rho[k_6] \circ \rho[k_5] \circ \rho[k_4] \circ \rho[k_3] \circ \rho[k_2] \circ \rho[k_1]$

$= \sigma[k_{12}] \circ \gamma^{-1} \circ \sigma[k_{12}] \circ \pi \circ \sigma[k_{11}] \circ \gamma \circ \sigma[k_{11}] \circ \ldots \circ \pi \circ \sigma[k_2] \circ \gamma^{-1} \circ \sigma[k_2] \circ \pi \circ \sigma[k_1] \circ \gamma \circ \sigma[k_1]$

A 12-round decryption can be represented as

$D[k] = \rho^{-1}[k_1] \circ \rho^{-1}[k_2] \circ \rho^{-1}[k_3] \circ \rho^{-1}[k_4] \circ \rho^{-1}[k_5] \circ \rho^{-1}[k_6] \circ \rho^{-1}[k_7] \circ \rho^{-1}[k_8] \circ \rho^{-1}[k_9] \circ \rho^{-1}[k_{10}] \circ \rho^{-1}[k_{11}] \circ \beta^{-1}[k_{12}]$

$= \sigma^{-1}[k_1] \circ \gamma \circ \sigma^{-1}[k_1] \circ \pi^{-1} \circ \sigma^{-1}[k_2] \circ \gamma^{-1} \circ \sigma^{-1}[k_2] \circ \pi^{-1} \circ \ldots \circ \sigma^{-1}[k_{11}] \circ \gamma \circ \sigma^{-1}[k_{11}] \circ \pi^{-1} \circ \sigma^{-1}[k_{12}] \circ \gamma^{-1} \circ \sigma^{-1}[k_{12}]$

$= \sigma[k_1] \circ \gamma \circ \sigma[k_1] \circ \pi \circ \sigma[k_2] \circ \gamma^{-1} \circ \sigma[k_2] \circ \pi \circ \ldots \circ \pi \circ \sigma[k_{11}] \circ \gamma \circ \sigma[k_{11}] \circ \pi \circ \sigma[k_{12}] \circ \gamma^{-1} \circ \sigma[k_{12}]$

$= \beta[k_1] \circ \rho[k_2] \circ \rho[k_3] \circ \rho[k_4] \circ \rho[k_5] \circ \rho[k_6] \circ \rho[k_7] \circ \rho[k_8] \circ \rho[k_9] \circ \rho[k_{10}] \circ \rho[k_{11}] \circ \rho[k_{12}]$

$= E[k]$

This shows that the decryption is the same as the encryption, with the key schedule applied in reverse order.

4.4.2 Substitution Layer

The substitution layer is the main component that provides confusion in the cipher, which is crucial to the security of the block cipher. This is the nonlinear layer that provides resistance against two of the most powerful cryptanalysis techniques, i.e. linear cryptanalysis and differential cryptanalysis. The design criteria for the S-Box have significant effect on the overall security of the block cipher, and are listed as follows:

i) High nonlinearity

ii) The maximum XOR table entries should be as low as possible, not greater than 4.

iii) No fixed points and no opposite fixed point.

iv) Resistant to linear cryptanalysis

v) Resistant to differential cryptanalysis

vi) Invertible

There are several types of S-Boxes commonly used by block ciphers, as discussed in Chapter 3. With the consideration of the design criteria listed above, algorithmic S-Box is chosen. Algorithmic S-Box can be generated according to mathematics principles so that they have proven security against linear and differential cryptanalysis. The algorithmic S-Box can be chosen to fulfill all the design criteria required. This is quite difficult to realise using random S-Box or key-dependent S-Box. With the selection of the algorithmic S-Box, another important design criterion is included, i.e.

vii) The S-Box should have complex algebraic expression.

This property is very important since algorithmic S-Boxes using simple algebraic manipulation are vulnarable to interpolation attack and higher-order differential cryptanalysis.

Based on Nyberg [35], there are several ways to construct algorithmic S-Boxes that fulfill the criteria listed above. One of the methods uses the mapping of $x \rightarrow x^{-1}$ in $GF(2^8)$. This mapping has another desirable property that it is involution, which means that it can be used in both encryption and decryption. But this mapping has a simple algebraic expression that makes it vulnerable to interpolation attack and higher-order differential cryptanalysis. Daemon et. al. apply an affine transformation over the mapping to overcome this weakness. This method is used in the design of cipher such as SHARK and SQUARE.

In order to make the proposed cipher involution, the cipher uses this S-Box and its inverse alternatively in different rounds as explained in the previous section. The design of the proposed cipher ensures that replacement of S-Boxes is easy to implement. The S-Boxes can be replaced by another invertible S-Box pair which has better cryptographic properties.

4.4.3 Round Key Addition

There are two round key additions in the round transformation. A round key is added to the input block before the substitution process. Another round key addition is performed after the substitution. The process is shown in Figure 4.6. This method was proposed by Biham and Biryukov to make S-Boxes key-dependent [36]. It is clear that this operation is linear and does not have any influence over differential or linear cryptanalysis. However, it makes the improved Davies' attack more complicated.

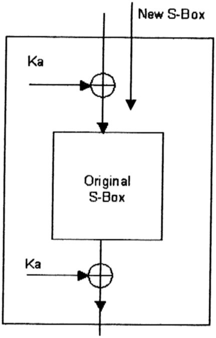

Figure 4.6 New S-Box

4.4.4 Linear Transformation

This is the main layer that provides diffusion in the round function. This purpose of this layer is to provide an avalanche effect. The design criteria for the diffusion layer is as follows:

a) Uniform and good diffusion property

b) Completeness

c) Involution

d) Injective, or one-to-one mapping

e) Good avalanche effect

These criteria can be fulfilled by using the mixing mechanism. The mixer used has four inputs and four outputs which transforms 32-bit input into 32-bit output value. Every mixer takes the outputs from four S-Boxes, which means that every output of each S-Box is connected to an input of the mixer. This is shown in Figure 4.7.

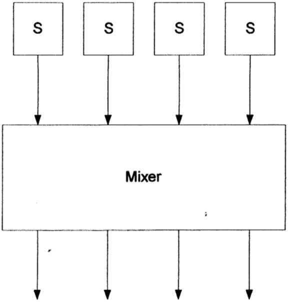

Figure 4.7 Connections between S-Boxes and Mixer

The mixer used has the properties that are desired to produce high diffusion. It is capable of meeting the design criteria of the diffusion layer. This is proved in the following sections.

(i) Completeness

Completeness means that every output bit is a function of all input bits. The mixer is represented algebraically in $GF(2^8)$ as follows:

$$q_0 = 3p_0 \oplus 2p_1 \oplus 2p_2 \oplus 2p_3$$

$$q_1 = 2p_0 \oplus 3p_1 \oplus 2p_2 \oplus 2p_3$$

$$q_2 = 2p_0 \oplus 2p_1 \oplus 3p_2 \oplus 2p_3$$

$$q_3 = 2p_0 \oplus 2p_1 \oplus 2p_2 \oplus 3p_3$$

where p_0, p_1, p_2, p_3 are the inputs of the mixer and q_0, q_1, q_2, q_3 are the outputs.

The diagram of the mixer is shown in Figure 4.3. This shows that every output is a function of all inputs. Hence, the mixer fulfills the completeness requirement.

(ii) Involution

The mixer is self-inverse. It can be used without change for either encryption or decryption. The equations representing the mixer can be written as follows:

$$q_0 = 3p_0 \oplus 2p_1 \oplus 2p_2 \oplus 2p_3 = 2(p_0 \oplus p_1 \oplus p_2 \oplus p_3) \oplus p_0$$

$$q_1 = 2p_0 \oplus 3p_1 \oplus 2p_2 \oplus 2p_3 = 2(p_0 \oplus p_1 \oplus p_2 \oplus p_3) \oplus p_1$$

$$q_2 = 2p_0 \oplus 2p_1 \oplus 3p_2 \oplus 2p_3 = 2(p_0 \oplus p_1 \oplus p_2 \oplus p_3) \oplus p_2$$

$$q_3 = 2p_0 \oplus 2p_1 \oplus 2p_2 \oplus 3p_3 = 2(p_0 \oplus p_1 \oplus p_2 \oplus p_3) \oplus p_3$$

If the mixer is involution, its inverse can be represented as follows:

$$p_0 = 2(q_0 \oplus q_1 \oplus q_2 \oplus q_3) \oplus q_0$$

$$p_1 = 2(q_0 \oplus q_1 \oplus q_2 \oplus q_3) \oplus q_1$$

$$p_2 = 2(q_0 \oplus q_1 \oplus q_2 \oplus q_3) \oplus q_2$$

$$p_3 = 2(q_0 \oplus q_1 \oplus q_2 \oplus q_3) \oplus q_3$$

Proof

By adding all the equations representing the mixer operation, we get

$$q_0 \oplus q_1 \oplus q_2 \oplus q_3 = p_0 \oplus p_1 \oplus p_2 \oplus p_3$$

By adding $2(p_0 \oplus p_1 \oplus p_2 \oplus p_3)$ to q_0,

$$q_0 \oplus 2(p_0 \oplus p_1 \oplus p_2 \oplus p_3) = 2(p_0 \oplus p_1 \oplus p_2 \oplus p_3) \oplus p_0 \oplus 2(p_0 \oplus p_1 \oplus p_2 \oplus p_3)$$

$$= p_0$$

The same applies to p_1, p_2 and p_3.

Hence, an inverse exists for any polynomials q_0, q_1, q_2 and q_3.

(iii) Injective

A function with an inverse on the same range must be one-to-one. Hence, the mixer is injective.

(iv) Avalanche Effect

The mixer has good avalanche criterion if any change in one input affects every output. Suppose some change Δp is added to p0:

$$q_0' = 3(p_0 \oplus \Delta p) \oplus 2p_1 \oplus 2p_2 \oplus 2p_3$$

$$\Delta q_0 = q_0' \oplus q_0 = 3\Delta p$$

Since p_0 is a function of every output, every output is affected by the change. The same result applies to other inputs.

(v) Configuration of Mixers

The proposed block cipher has a block size of 128 bits. A single mixer can mix 32 bits of output form four S-Boxes. Hence, it requires a total of eight mixers arranged in two layers of to mix the whole16 bytes of block size. This is shown in Figure 4.8.

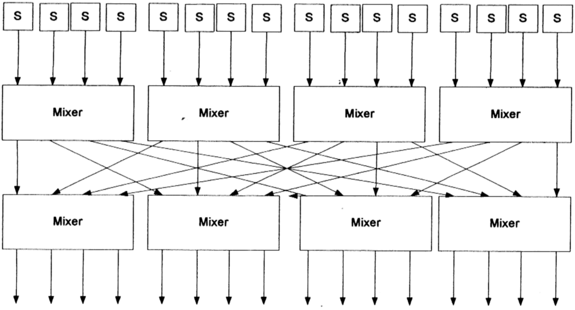

Figure 4.8 Interconnection of Mixers

In the first layer, the mixer mixes the inputs received from the four S-Boxes and each output is a function of every input. These outputs from each mixer in the first layer are then fed into four different mixers in the second layer. Therefore, each output of the mixers in the second layer is influenced by outputs of every S-Box. This creates an avalanche effect where any changes in any sub-block can affect every sub-block across the whole block cipher, which means that a single round has good diffusive properties. The interconnection pattern between the mixers in two layers ensures that all the design criteria for the linear transformation layer, i.e. completeness, involution, injective and good avalanche effect are fulfilled across the whole block cipher.

In terms of performance, the mixer operation can be implemented efficiently in hardware and software. This can be shown by the following equation:

$$q_0 = 2(p_0 \oplus p_1 \oplus p_2 \oplus p_3) \oplus p_0$$

Arithmetic in a Galois Field of $GF(2^m)$ requires only XOR and shift operations. The equation above requires four XOR operations and a left shift. A mixer with four inputs and four outputs requires seven XOR operations and a left shift since $2(p_0 \oplus p_1 \oplus p_2 \oplus p_3)$ can be reused in other equations.

4.4.5 Key Scheduling

A key schedule expands a relatively short master key to a relatively large expanded key for later use in different parts of the cipher that needs it. A poorly designed key schedule may cause undesired cipher properties, e.g. weak keys, equivalent keys, self-inverse key, etc. This may cause the cipher to be vulnerable to various attacks, such as related-key attacks.

The design criteria that guides the key schedule development are listed below:

i) Knowledge of one round subkey does not directly specify bits of other round subkeys.

ii) Avoid using linear key scheduling

iii) Maximize avalanche in the subkey

iv) Reuse the same primitives

The key schedule was designed to ensure that other round subkeys cannot be generated with the knowledge of one round subkey. This can be achieved by using key addition. If an attacker knows one round subkey, he must have the knowledge of the previous subkey in order to generate other round subkeys as shown by the following equation:

$$k_i = \sigma[k_{i-2}] \circ \pi \circ \gamma[k_{i-1}]$$

where $i = 3,\ 4, \ldots, 24$

In order to do so, this equatiốn has to be inverted.

$$k_{i-1} = \sigma^{-1}[k_{i-2}] \circ \pi \circ \gamma^{-1}[k_i]$$

This cannot be done without the knowledge of the value of k_{i-2}, which belongs to another subkey.

Linear key scheduling was avoided during the design of the cipher. This is due to the weaknesses of linear key scheduling itself. Many block ciphers using linear key scheduling has been cryptanalysed, such as LOKI, 3-Way, SAFER and

TEA. Therefore, a substitution layer was introduced to make the key scheduling nonlinear. This is done by reusing the same S-Box as in the round transformation. The high level of nonlinearity prohibits the full determination of round key difference from cipher key differences only.

In order to immunizing the key schedule against related-key attacks, the key schedule is designed to have maximum avalanche effect in round subkeys. Hence, a linear transformation was applied in the key scheduling.

4.5 Conclusion

Due to the vulnerability to attacks of DES, a new block cipher with improved performance and security is proposed. With a 256-bit key length and 128-bit block size, the proposed block cipher has greater security over DES. Unlike DES, the proposed block cipher is an iterated block cipher, using a standard substitution-permutation network. It uses highly nonlinear S-Boxes as the substitution layer to provide confusion to the block cipher and a combination of linear mixers as the diffusion layer to achieve the avalanche effect in fewer rounds of transformation.

Based on analysis done, an SP network is used as the structure of the block cipher. Most of the block ciphers like DES are Feistel Cipher. An important drawback of the Feistel structure is its round function exhibits a large amount of linearity that can be exploited by differential cryptanalysis and linear cryptanalysis. This is because only half of the bits of the intermediate round output enter the f function in the next round. Futhermore, diffusion is slower on Feistel cipher because only half of the bits are transformed in each round.The design of the block cipher enables the use of the same round function for both encryption and decryption without using Fesitel structure. This is desirable since the reuse of the encryption module for decryption can save memory, space and cost. The performance of the block cipher is improved by using table lookup and simple Boolean operations. This leads to a block cipher with good security level and performance.