T.C.
İSTANBUL UNIVERSITY
INSTITUTE OF GRADUATE STUDIES IN
SCIENCE AND ENGINEERING

**Ph.D. THESIS**

# CONSTRUCTING SELF-DUAL CODES OVER THE RINGS $\mathbb{F}_2+v\mathbb{F}_2$ AND $\mathbb{F}_2\times\mathbb{F}_2+v\mathbb{F}_2$

**Refia AKSOY**

**Department of Mathematics**

**Mathematics Programme**

**SUPERVISOR**
**Assist. Prof. Dr. Fatma ÇALIŞKAN**

Çalışkan

**June, 2019**

**İSTANBUL**

This study was accepted on 19.06.2019 as a Ph.D. thesis in Department of Mathematics, Mathematics Programme by the following Committee.

**Examining Committee Members**

Çalışkan

Assist. Prof. Dr. Fatma ÇALIŞKAN (Supervisor)
İstanbul University
Faculty of Science

K. Saygılı

Prof. Dr. Kamuran SAYGILI
İstanbul University
Faculty of Science

Assist. Prof. Dr. Erol SERBEST
Yeditepe University
Faculty of Arts and Sciences

Assoc. Prof. Dr. Fatih DEMİRKALE
Yıldız Technical University
Faculty of Arts and Sciences

Assoc. Prof. Dr. Ayten PEKİN
İstanbul University
Faculty of Science

As required by the 9/2 and 22/2 articles of the Graduate Education Regulation which was published in the Official Gazette on 20.04.2016, this graduate thesis is reported as in accordance with criteria determined by the Institute of Graduate Studies in Science and Engineering by using the plagiarism software to which İstanbul University is a subscriber.

This study is supported by the project numbered 29539 of İstanbul University Scientific Research Projects Executive.

## FOREWORD

First of all, I would like to express my candid appreciativeness to my thesis supervisor Assist. Prof. Dr. Fatma ÇALIŞKAN who does not refrain to support and to guide me with ingenious suggestions through the research period. I also would like to express my deep gratitude to Assoc. Prof. Dr. Bahattin YILDIZ who kindly shared his immense knowledge with me. I am grateful to Assist. Prof. Dr. Suat KARADENİZ for his emboldening and motivation.

I want to thank the committee members Prof. Dr. Kamuran SAYGILI, Assist. Prof. Dr. Erol SERBEST, Assoc. Prof. Dr. Fatih DEMİRKALE and Assoc. Prof. Dr. Ayten PEKİN for their helpful advices. I also want to thank Dr. Hikmet ÇAKMAK, Assist. Prof. Dr. Özkan DEĞER and Assist. Prof. Dr. Ali Uğur SAZAKLIOĞLU for their help with LaTeX.

I would like to send my special thanks to TÜBİTAK BİDEB for their financial support during my Ph.D. research.

Finally, I would like to thank my family, especially to my nephew Alim Mirza, for their motivation, encouragement and support.

June, 2019 Refia AKSOY

## TABLE OF CONTENTS

**Page**

# LIST OF FIGURES

**Page**

# LIST OF TABLES

**Page**

# LIST OF SYMBOLS AND ABBREVIATIONS

| Symbol | Explanation |
|---|---|
| $\mathbb{F}_2$ | : the binary field |
| $\mathcal{R}_1$ | : the ring $\mathbb{F}_2 + v\mathbb{F}_2$ |
| $\mathcal{R}_2$ | : the ring $\mathbb{F}_2 \times \mathcal{R}_1$ |

| Abbreviation | Explanation |
|---|---|
| BDCC | : the bordered double-circulant construction |
| CRT | : Chinese remainder theorem |
| CWE | : complete weight enumerator |
| DCC | : the double-circulant construction |
| EIP | : Euclidean inner product |
| HIP | : Hermitian inner product |
| LWE | : Lee weight enumerator |
| SC | : the symmetric construction |
| SWE | : symmetrized weight enumerator |

## SUMMARY

# Ph.D. THESIS

### CONSTRUCTING SELF-DUAL CODES OVER THE RINGS $\mathbb{F}_2 + v\mathbb{F}_2$ AND $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$

**Refia AKSOY**

**İstanbul University**

**Institute of Graduate Studies in Science and Engineering**

**Department of Mathematics**

**Supervisor: Assist. Prof. Dr. Fatma ÇALIŞKAN**

In this dissertation, self-dual codes over the rings $\mathbb{F}_2 + v\mathbb{F}_2$ and $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$ with $v^2 = v$ are considered. The necessary and sufficient conditions in order to obtain Euclidean and Hermitian self-dual codes over both rings are given. By using the distance preserving Gray maps, codes over these rings are transferred to the binary field $\mathbb{F}_2$. By using some circulant and some symmetric matrices, free Euclidean and free Hermitian self-dual codes over the ring $\mathbb{F}_2 + v\mathbb{F}_2$ of even length are obtained. A new shortening method which enables us to obtain Hermitian self-dual codes of odd length by using Hermitian self-dual codes of even length is presented. The Hermitian self-dual code of length 31 with minimum Hamming weight 8 whose existence was not known previously is found by using the shortening method. The structure of the ring $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$, which is a commutative non-chain ring of characteristic 2, is investigated. Linear codes over this ring are defined. By using some circulant and some symmetric matrices, free Euclidean and free Hermitian self-dual codes over the ring $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$ of even length are found. The complete, symmetrized, Hamming and Lee weight enumerators for the linear codes over the ring $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$ are defined and the MacWilliams identities are proved.

June, 2019, 93 pages.

**Keywords:** Codes over rings, self-dual codes, Gray map, weight enumerators, MacWilliams identities.

# ÖZET

## DOKTORA TEZİ

### $\mathbb{F}_2 + v\mathbb{F}_2$ VE $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$ HALKALARI ÜZERİNE SELF-DUAL KODLARIN İNŞAASI

**Refia AKSOY**

**İstanbul Üniversitesi**

**Fen Bilimleri Enstitüsü**

**Matematik Anabilim Dalı**

**Danışman: Dr. Öğr. Üyesi Fatma ÇALIŞKAN**

Bu tezde, $\mathbb{F}_2 + v\mathbb{F}_2$ ve $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$ ($v^2 = v$) halkaları üzerine self-dual kodlar göz önüne alınmıştır. Her iki halka üzerindeki Öklid ve Hermit self-dual kodları elde etmek için gerek ve yeter koşullar verilmiştir. Uzaklık koruyan Gray dönüşümleri kullanılarak bu halkalar üzerindeki kodlar $\mathbb{F}_2$ cismine taşınmıştır. Bazı dairesel (circulant) ve simetrik matrisler kullanılarak $\mathbb{F}_2 + v\mathbb{F}_2$ halkası üzerindeki çift uzunluklu serbest Öklid ve serbest Hermit self-dual kodlar elde edilmiştir. Çift uzunluklu Hermit self-dual kodlar kullanılarak tek uzunluklu Hermit self-dual kodları elde etmeye imkan tanıyan yeni bir kısaltma metodu sunulmuştur. Varlığı daha önce gösterilemeyen, minimum Hamming ağırlığı 8 olan 31 uzunluğundaki Hermit self-dual kod kısaltma metodu kullanılarak bulunmuştur. Karakteristiği 2 olan $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$ değişmeli halkasının yapısı araştırılmıştır. Bu halka zincir olmayan bir halkadır. Bu halka üzerine lineer kodlar tanımlanmıştır. Bazı dairesel ve simetrik matrisler kullanılarak $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$ halkası üzerindeki çift uzunluklu serbest Öklid ve serbest Hermit self-dual kodlar bulunmuştur. $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$ halkası üzerindeki lineer kodlar için tam, simetrikleştirilmiş, Hamming ve Lee ağırlık dağılımları tanımlanmış ve MacWilliams eşitlikleri ispatlanmıştır.

Haziran, 2019, 93 sayfa.

**Anahtar kelimeler:** Halkalar üzerindeki kodlar, self-dual kodlar, Gray dönüşüm, ağırlık dağılımları, MacWilliams bağıntıları.

# 1. INTRODUCTION

Coding theory emerged in response to the problem of how to transmit information in a secure manner through the channel despite a possible external noise. The inception of coding theory dates back to 1948. In the breakthrough paper "A Mathematical Theory of Communication" [1], Claude Shannon showed that, if convenient encoding and decoding procedures are used, then a trustworthy communication can be accomplished over a noisy communication channel (see Figure 1.1). Although it may seem that coding theory is oriented towards engineering and computer science, deep mathematical techniques and results are commonly used in the advancement of coding theory.

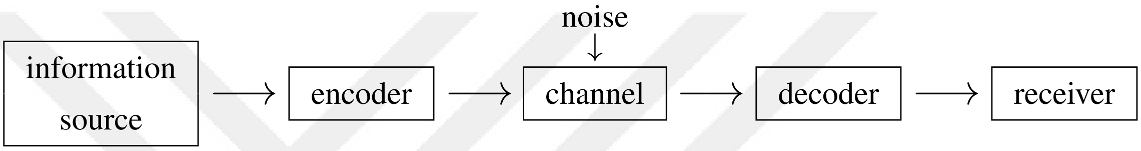

**Figure 1.1:** The communication process.

One of the tasks in coding theory is to correct the errors that occur in the transmission. The error correcting capability of a code is related to its minimum distance. If the minimum distance of a code increases, the error correcting capability of the code also increases. Researchers have found different bounds on minimum distance of linear codes. In this study, we are looking for the codes whose error correcting capabilities are maximum.

The first studies about error correcting codes were done over finite fields, particularly over the binary field $\mathbb{F}_2$. After the study about codes over the ring $\mathbb{Z}_4$ was published in 1994 [2], the excessive attention has been directed to codes over finite rings. Hammons et al. [2] established a relationship between codes over $\mathbb{Z}_4$ and binary codes by way of the Gray map defined. In [3], Wood used the notion "Frobenius rings" which are the largest family of rings in order to study codes. In that study, some connections between a linear code and its dual, called the MacWilliams identities, were built. Various Frobenius rings were investigated by dealing with different points.

Codes over commutative rings of order 4 have been studied intensively. The generalizations of these rings to other rings by lifting have also been considered. For instance the ring $\mathbb{Z}_4$ has been checked out from many aspects in [4–6]. Then in [7], by lifting the codes over $\mathbb{Z}_4$, they obtained the codes over $\mathbb{Z}_4 + u\mathbb{Z}_4$ with $u^2 = 0$.

Codes over $\mathbb{F}_2+v\mathbb{F}_2$ (which we denote as $\mathcal{R}_1$) with $v^2=v$, which is a commutative ring of order 4, have generated a considerable amount of interest in the coding theory literature for the last two decades. In [8], Bachoc introduced self-dual codes over $\mathbb{F}_2\times\mathbb{F}_2$, which is isomorphic to $\mathcal{R}_1$. Several researchers have studied codes over $\mathcal{R}_1$ from different perspectives [9–13]. Hermitian self-dual codes over $\mathcal{R}_1$ of lengths up to 32 were classified in [14]. Optimal $\theta$-cyclic Hermitian self-dual codes over $\mathcal{R}_1$ were investigated in [15] together with Hermitian Type IV self-dual codes of even length less than 28. Quadratic residue codes over $\mathbb{F}_p+v\mathbb{F}_p$, with $p$ a prime number, were studied in [16], and cyclic isodual and formally self-dual codes over $\mathbb{F}_q+v\mathbb{F}_q$, $q$ a prime power, were constructed in [17]. $\mathcal{R}_1$ was generalized to the infinite family of rings $A_k=\mathbb{F}_2[v_1,v_2,\ldots,v_k]/\langle v_i^2=v_i, v_iv_j=v_jv_i\rangle$, $1\leq i,j\leq k$ in [18]. In that paper, the structure of the ring $A_k$ in detail was studied and some results about Hermitian self-dual codes over this ring were given.

Recently, various types of codes over the direct product of different finite rings have been arousing interest. For instance, in [19], Borges et al. showed $\mathbb{Z}_2\mathbb{Z}_4$-additive codes as $\mathbb{Z}_4$-submodules of $\mathbb{Z}_2^{\alpha}\times\mathbb{Z}_4^{\beta}$ where $\alpha$ and $\beta$ are positive integers. Later, in [20], Aydogdu et al. presented $\mathbb{Z}_2\mathscr{R}$ codes as $\mathscr{R}$-submodules of $\mathbb{Z}_2^{\alpha}\times\mathscr{R}^{\beta}$ where $\mathscr{R}=\{0,1,u,1+u\}$ with $u^2=0$, $\alpha$ and $\beta$ are positive integers. For more information about the direct product of different rings, we refer the reader to [21–25].

In Section 2, we first give some basic concepts and definitions about finite fields and finite rings. We mention self-dual codes briefly. Linear codes, Euclidean and Hermitian self-dual codes and a Gray map over $\mathcal{R}_1$ are given and MacWilliams identities are checked over for linear codes over this ring. By considering the generator matrices of free self-dual codes over this ring, the necessary and sufficient conditions in order to obtain Euclidean and Hermitian self-dual codes are given. Then we introduce the direct product ring $\mathbb{F}_2\times\mathbb{F}_2+v\mathbb{F}_2$ (which we denote as $\mathcal{R}_2$) with $v^2=v$ and linear codes over this ring are defined. We present two duality preserving Gray maps to determine the distances of linear codes over this ring. We define two inner products on this ring, namely Euclidean inner product (EIP) and Hermitian inner product (HIP), and we deal with Euclidean and Hermitian self-dual codes over this ring. By introducing the generator matrices of free self-dual codes over this ring, the necessary and sufficient conditions in order to obtain Euclidean and Hermitian self-dual codes are given. Several weight enumerators are defined and the MacWilliams identities for linear codes are proven. Later, we recall some construction methods which are used in order to find free Euclidean and free Hermitian self-dual codes of even length over both rings and we present a new shortening method to obtain Hermitian self-dual codes of odd length from Hermitian self-dual codes of even length over $\mathcal{R}_1$.

In Section 3, we first construct Euclidean and Hermitian self-dual codes of even length over $\mathcal{R}_1$ by using the double-circulant construction (DCC), the bordered double-circulant construction (BDCC) and the symmetric construction (SC) methods and then we apply the new shortening method to obtain Hermitian self-dual codes of odd length from Hermitian self-dual codes of even length. We apply the same construction methods to obtain Euclidean and Hermitian self-dual codes over $\mathcal{R}_2$. While applying the construction methods, we use the Magma computer program [26] to expedite the calculations.

In Section 4, we interpret the results that we obtained. The last section concludes the thesis.

# 2. MATERIALS AND METHODS

## 2.1. BASIC CONCEPTS

In this subsection we mention some basic notions of coding theory over finite fields and finite rings. We refer to [27–30] for more.

Let $F_q$ be a finite set with $q$ elements, which is called an *alphabet*. Let $F_q^n$ be the set of all $n$-tuples such as $\mathbf{a} = \mathbf{a}_1\mathbf{a}_2\ldots\mathbf{a}_n$ where each $\mathbf{a}_i$ belongs to $F_q$. An element of $F_q^n$ is called a *word*. A code $C$ of length $n$ over the alphabet $F_q$ is a subset of $F_q^n$. An element of $C$ is called a *codeword*. A code $C$ is called a *binary code* if $q = 2$ and a *ternary code* if $q = 3$.

If $\mathbb{F}_q$ is a finite field with $q$ elements, where $q$ is a prime power, then an $[n,k]$-*linear code* over $\mathbb{F}_q$ is a $k$-dimensional subspace of $\mathbb{F}_q^n$. If $R$ is a finite commutative ring, then a *linear code* over $R$ of length $n$ is an $R$-submodule of $R^n$.

For any two words $\mathbf{x} = \mathbf{x}_1\mathbf{x}_2\ldots\mathbf{x}_n$ and $\mathbf{y} = \mathbf{y}_1\mathbf{y}_2\ldots\mathbf{y}_n$ of $F_q^n$, the *Hamming distance* from $\mathbf{x}$ to $\mathbf{y}$, denoted as $d_H(\mathbf{x},\mathbf{y})$, is defined as

$$d_H(\mathbf{x},\mathbf{y}) := |\{i : \mathbf{x}_i \neq \mathbf{y}_i\}|$$

which is a metric on $F_q^n$. The *minimum Hamming distance* of a code $C$, denoted as $d(C)$, is defined by

$$d_H(C) := min\{d_H(\mathbf{x},\mathbf{y}) |\ \mathbf{x} \neq \mathbf{y},\ \mathbf{x},\mathbf{y} \in C\}.$$

The *Hamming weight* of a word $\mathbf{x}$, denoted as $w_H(\mathbf{x})$, is defined to be the number of non-zero coordinates. The *minimum Hamming weight* of a code $C$, denoted as $w_H(C)$, is defined as the smallest of the weights of the non-zero codewords of $C$. When a code is linear, the notions $w_H(C)$ and $d_H(C)$ coincide.

A linear code is represented by three parameters; length $n$, dimension $k$ and minimum distance $d$. A good $[n,k,d]$-code has small $n$, large $k$ and large $d$ [30]. However, these parameters are conflicting. This discrepancy is often ascribed as "the main problem of coding theory" and this problem seeks one of the parameters $n$, $k$ and $d$ for the given values of the other two. A code whose minimum distance is $d$ can detect up to $d-1$ errors and correct $\lfloor \frac{d-1}{2} \rfloor$ errors.

An $[n,k]$-code $C$ can be obtained from a $k \times n$ generator matrix $G$ whose rows form a basis for $C$. The dual code of $C$, which is defined as the annihilator of $C$ with respect to the inner product defined on the alphabet, is obtained from an $(n-k) \times n$ parity-check matrix $H$ whose rows span the space orthogonal to $C$. For an $[n,k]$-code over a field, the standard generator matrix is of the form $[I_k, A]$, where $I_k$ is the $k \times k$ identity matrix and $A$ is a $k \times (n-k)$ matrix. If the generator matrix of an $[n,k]$-code is of the form $[I_k, A]$, then the dual code has the generator matrix of the form $[-A^T, I_{n-k}]$, where $A^T$ is the transpose of $A$.

The *weight enumerator* of a linear code $C$ of length $n$ is defined as

$$W_C(x) = \sum_{i=0}^{n} A_i x^i,$$

where $A_i$ is the number of codewords of weight $i$. This weight enumerator can also be denoted as a homogeneous polynomial

$$W_C(x,y) = \sum_{i=0}^{n} A_i x^{n-i} y^i$$

depending on two indeterminates $x$ and $y$.

Two codes are *permutation-equivalent* if one can be obtained from the other by permuting the coordinates. A code $C$ is called:

- *formally self-dual* if $C$ and the dual code of $C$, which is denoted as $C^{\perp}$, have the same weight enumerators.
- *isodual* if it is equivalent to $C^{\perp}$.
- *self-orthogonal* if it is a subset of $C^{\perp}$.
- *self-dual* if it is equal to $C^{\perp}$.

Note that the dual of a linear code $C$ is also a linear code.

**Example 2.1.1.** Let $C$ be a binary linear code given as

$$C = \{0000, 1100, 0011, 1111\}.$$

The Hamming distance from 1100 to 0011 is $d_H(1100, 0011) = 4$. The Hamming weight of 1100 is $w_H(1100) = 2$. The minimum Hamming distance $(d_H(C))$ is 2. The minimum Hamming weight $(w_H(C))$ is also 2 since $C$ is linear. This code is a [4,2,2]-code. The generator matrix of the code is equal to its parity check matrix which is

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

The weight enumerator of the code is $W_C(x) = 1 + 2x^2 + x^4$. The dual code $C^{\perp}$ is equal to $C$, hence $C$ is self-dual.

The characterization of formally self-dual codes by divisibility over a finite field $\mathbb{F}_q$ of order $q$ is given as in the following which is proven shortly in [31]:

**Theorem 2.1.1.** Suppose that $C$ is a formally self-dual code of length $n$ over $\mathbb{F}_q$, which has the Hamming weights of all codewords divisible by a positive integer $\delta$. Then one of the following holds:

$$\begin{aligned}
&\text{Type I}: && q = 2 \text{ and } \delta = 2, \\
&\text{Type II}: && q = 2 \text{ and } \delta = 4, \\
&\text{Type III}: && q = 3 \text{ and } \delta = 3, \\
&\text{Type IV}: && q = 4 \text{ and } \delta = 2 \text{ or} \\
&\text{Type V}: && q \text{ is arbitrary}, \delta = 2 \text{ and } \ W_C(x,y) = (x^2 + (q-1)y^2)^{n/2}.
\end{aligned}$$

Based on Theorem 2.1.1, a self-dual code over an alphabet of order 4 is *Type IV* if the Hamming weights of all codewords are even. We modify the notion "*Type I*" for a self-dual code over an alphabet of order 4 if it has at least one codeword of odd weight.

A linear code corresponds to different kind of sets depending on the algebraic structure defined. If a linear code is defined over a finite field then it corresponds to a subspace of the finite field. If a linear code is defined over a finite ring then it corresponds to a submodule of the finite ring. Linear codes over finite fields have a dimension while linear codes over finite rings have a type.

**Definition 2.1.1.** [32] Let $R$ be any finite commutative ring. The intersection of all maximal ideals in the ring is the *Jacobson radical* $J(R)$. The sum of all the minimal ideals is called the *socle* of the ring, $soc(R)$. If $R/J(R)$ is isomorphic to $soc(R)$, then it is said that $R$ is *Frobenius*.

One of the most important results of codes over Frobenius rings is the following:

**Theorem 2.1.2.** [3] Let $C^{\perp}$ be the dual of the code $C$ over a Frobenius ring $R$ of length $n$. Then

$$|C||C^{\perp}| = |R|^{n}.$$

By means of the following notions, one can determine whether a ring is Frobenius.

**Definition 2.1.2.** [33] Let $R$ be a finite commutative ring with unity. The mapping $\chi : R \to \mathbb{C}^{*}$ is a *character* for $R$ if $\chi(a+b) = \chi(a)\chi(b)$ for all $a,b \in R$.

**Lemma 2.1.1.** [3] Let $\chi$ be a character of a finite ring $R$. $\chi$ is a generating character for $R$ if and only if $ker\chi$ contains no non-zero ideals of $R$.

**Theorem 2.1.3.** [3] A finite ring $R$ is Frobenius if and only if it has a generating character.

The following notions and properties are useful tools for our research.

**Definition 2.1.3.** Let $I_1$ and $I_2$ be two ideals of any ring $R$. $I_1$ and $I_2$ are said to be *relatively prime* if $I_1 + I_2 = R$.

In [34], Dougherty et al. stated the Chinese remainder theorem (CRT) as follows:

**Theorem 2.1.4.** Let $\mathrm{I}_1, \ldots, \mathrm{I}_k$ be relatively prime ideals of any commutative ring $R$ with unity such that $S_i = R/\mathrm{I}_i$ be finite for all $1 \leq i \leq k$. Set $\mathrm{I} = \cap \mathrm{I}_i$ and $S = R/\mathrm{I}$. Then the map

$$\eta : S \to (R/\mathrm{I}_1) \times \cdots \times (R/\mathrm{I}_k),$$

by

$$\eta(\alpha) = (\alpha(\text{mod } \mathrm{I}_1), \ldots, \alpha(\text{mod } \mathrm{I}_k))$$

is a ring isomorphism. Moreover, $\eta^{-1}$ is also an isomorphism and denoted as $CRT$.

They also gave the following definition to denominate the code over $R$:

**Definition 2.1.4.** [34] Let $C_1,\ldots,C_k$ be codes where $C_i$ is a code over $S_i$ and define the code

$$CRT(C_1,\ldots,C_k)=\{\eta^{-1}(m_1,\ldots,m_k)|m_i\in C_i\},$$

where $C_i$ is a code over $S_i$. The code $CRT(C_1,\ldots,C_k)$ is the *Chinese product* of codes $C_1,\ldots,C_k$. It is obvious that $|CRT(C_1,\ldots,C_k)|=\prod\limits_{i=1}^{k}|C_i|$.

The CRT is also given in [35] in terms of maximal ideals but we first need to clarify what the index of stability is.

**Definition 2.1.5.** Let $I$ be an ideal of a finite commutative ring. The chain $I\supset I^2\supset I^3\supset\ldots$ stabilizes after some power of $I$. The smallest positive integer $e$ such that $I^e=I^{e+i}$ for $i\geq 0$ is called the *index of stability* of $I$.

**Proposition 2.1.1.** Let $R$ be a finite commutative ring with maximal ideals $m_1,\ldots,m_s$ where the index of stability of $m_i$ is $e_i$. Then the map

$$\mu:R\to\prod_{i=1}^{s}R/m_i^{e_i},$$

defined by $\mu(x)=(x+m_1^{e_1},\ldots,x+m_s^{e_s})$, is a ring isomorphism for any $x\in R$.

**Proposition 2.1.2.** [35] A direct product of local rings is isomorphic to the finite commutative ring $R$ via the CRT.

**Definition 2.1.6.** A ring $R$ is said to be a *Boolean ring* if all the elements of the ring are idempotent, that is, $x^2=x$ for all $x\in R$.

**Proposition 2.1.3.** [36] A finite Boolean ring has $2^k$ elements for some positive integer $k$. It has a unit element and is isomorphic to the direct product of $k$ fields $\mathbb{F}_2$.

In our research, we deal with constructing self-dual codes. Self-dual codes over finite fields and finite rings are one of the most momentous and on a vast scale studied families of codes since they have connections to some structures such as groups, designs and lattices. There are lots of studies in the literature about constructing self-dual codes over different alphabets. Self-dual codes are linear and self-orthogonal. Binary self-dual codes exist for even lengths and the weights of all codewords of binary self-dual codes are even. The dimension of a binary self-dual code of length $n$ is $n/2$.

Binary self-dual codes have the following bound on their minimum distance:

**Theorem 2.1.5.** [37] Suppose that $d_I(n)$ and $d_{II}(n)$ are the minimum distance of a Type I and Type II binary code of length $n$, respectively. Then

$$d_I(n) \leq \begin{cases} 4\lfloor \frac{n}{24} \rfloor + 4 & \text{if } n \not\equiv 22 \pmod{24} \\ 4\lfloor \frac{n}{24} \rfloor + 6 & \text{if } n \equiv 22 \pmod{24} \end{cases}$$

and

$$d_{II}(n) \leq 4\lfloor \frac{n}{24} \rfloor + 4.$$

A code whose minimum distance attains the highest possible value is called *extremal*. A code which has the largest minimum distance among all codes of a given length and type (self-dual, cyclic, etc.) is named *optimal*.

The studies which are performed over binary field to obtain self-dual codes can be carried out to the rings whose characteristic is 2 by using some tools such as lifting. Hence, we can obtain self-dual codes over different alphabets.

Depending on the automorphism group of an alphabet, we are able to define different inner products such as EIP and HIP. We will represent these two inner products on two different rings which are considered in Subsection 2.2 and Subsection 2.3 in detail, but we would like to mention the following results here since the theorems are also valid for some other rings than those two.

**Definition 2.1.7.** A linear code $C$ over any ring $R$ of length $n$ is said to be *reversible* if the words obtained by reversing the order of the components of all codewords of $C$ are included in $C$, i.e, for all elements $(c_1, c_2, \ldots, c_n)$ of $C$, the element $(c_n, c_{n-1}, \ldots, c_1)$ is also in $C$.

**Theorem 2.1.6.** [38] If $C$ is a Euclidean self-dual code of even length over a commutative ring $R$ with characteristic 2, then $C$ is reversible.

**Proof:** Assume that $C$ is a Euclidean self-dual code of length $2n$ over a commutative ring $R$ with characteristic 2. If $(c_1, c_2, \dots, c_{2n})$ is in $C$, then

$$\begin{aligned} \langle (c_1, c_2, \dots, c_{2n}), (c_{2n}, c_{2n-1}, \dots, c_1) \rangle &= c_1 c_{2n} + c_2 c_{2n-1} + \dots + c_{2n} c_1 \\ &= 0, \end{aligned}$$

which means $(c_{2n}, c_{2n-1}, \dots, c_1)$ is in $C$. ■

**Remark 2.1.1.** Theorem 2.1.6 is not true for Hermitian self-dual codes.

**Example 2.1.2.** Let $C_1$ be given as

$$C_1 = \{00, v1+v, 01+v, v0\}$$

and $C_2$ be given as

$$C_2 = \{000, 11v, vvv, 1+v1+v0, vv0, 1+v1+vv, 00v, 110\}.$$

Both $C_1$ and $C_2$ are Hermitian self-dual codes over the ring $\mathbb{F}_2 + v\mathbb{F}_2$ but they are not reversible.

**Theorem 2.1.7.** [38] Let $C$ be a Euclidean self-dual code over a ring $R$. If $R$ is Boolean, then $C$ contains the element $(1, 1, \dots, 1)$.

**Proof:** Let $R$ be a Boolean ring and $(c_1, c_2, \dots, c_n)$ be a codeword of $C$. Since $C$ is a Euclidean self-dual code,

$$\begin{aligned} \langle (c_1, c_2, \dots, c_n), (c_1, c_2, \dots, c_n) \rangle &= c_1 c_1 + c_2 c_2 + \dots + c_n c_n \\ &= c_1^2 + c_2^2 + \dots + c_n^2 \\ &= c_1 + c_2 + \dots + c_n \\ &= 0. \end{aligned}$$

Hence

$$\langle (c_1, c_2, \dots, c_n), (1, 1, \dots, 1) \rangle = c_1 + c_2 + \dots + c_n = 0$$

which implies that $(1,1,\ldots,1)$ is a codeword of $C$. ■

**Remark 2.1.2.** Theorem 2.1.7 does not hold for Hermitian self-dual codes.

**Example 2.1.3.** Let $C_2$ be the code given in Example 2.1.2. Eventhough the ring $\mathbb{F}_2+v\mathbb{F}_2$ is Boolean, $C_2$ does not contain the word 111.

## 2.2. SELF-DUAL CODES OVER THE RING $\mathbb{F}_2+v\mathbb{F}_2$

### 2.2.1. The Ring $\mathcal{R}_1$

The ring $\mathcal{R}_1=\mathbb{F}_2+v\mathbb{F}_2$ is a ring of characteristic 2 with the condition $v^2=v$ and it can be defined as

$$\mathcal{R}_1=\{a+vb \,|\, v^2=v,\, a,b\in\mathbb{F}_2\}.$$

Also, we have the isomorphism $\mathcal{R}_1\cong\mathbb{F}_2[X]/\langle X^2-X\rangle$. $\mathcal{R}_1$ is a finite commutative non-chain ring of order 4. Each element of $\mathcal{R}_1$ is idempotent. Hence, $\mathcal{R}_1$ is a Boolean ring. 1 is the only unit of $\mathcal{R}_1$. Therefore, this ring has four ideals: $\langle 0\rangle$, $\langle 1\rangle$, $\langle v\rangle$ and $\langle 1+v\rangle$. From the ideal lattice given in Figure 2.1, it is clear that this ring is a semilocal ring with two maximal ideals given as $\langle v\rangle$ and $\langle 1+v\rangle$.

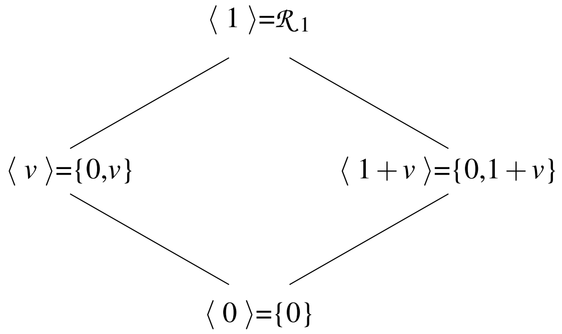
$\langle\ 1\ \rangle$=$\mathcal{R}_1$
$\langle\ v\ \rangle$={0,$v$}
$\langle\ 1+v\ \rangle$={0,$1+v$}
$\langle\ 0\ \rangle$={0}

**Figure 2.1:** The lattice of the ideals of $\mathcal{R}_1$.

Let $a+vb$ and $d+ve$ be two elements in $\mathcal{R}_1$. Then addition is given as

$$(a+vb)+(d+ve)=(a+d)+v(b+e)$$

and multiplication is given as

$$(a+vb)(d+ve)=(ad)+v(ae+bd+be)$$

where $a,b,d,e \in \mathbb{F}_2$.

Note that, $\mathcal{R}_1$ has two maximal ideals $\mathrm{I}_1 = \langle v \rangle$ and $\mathrm{I}_2 = \langle 1+v \rangle$ as mentioned above. Therefore,

$$\boldsymbol{\phi} : \mathcal{R}_1 \to \mathcal{R}_1/\mathrm{I}_1 \times \mathcal{R}_1/\mathrm{I}_2$$

is a ring isomorphism via the CRT, where

$$\boldsymbol{\phi}(a+vb) = (a, a+b).$$

Also, $\boldsymbol{\phi}_i : \mathcal{R}_1 \to \mathcal{R}_1/\mathrm{I}_i$ is a canonical homomorphism and $\mathcal{R}_1/\mathrm{I}_i \cong \mathbb{F}_2$ for $i = 1,2$. Therefore, the isomorphism $\mathcal{R}_1 \cong \mathbb{F}_2 \times \mathbb{F}_2$ is obvious to see. This result can also be obtained from Proposition 2.1.3.

This ring isomorphism corresponds to the Gray map defined from $\mathcal{R}_1$ to the binary field. This Gray map $\boldsymbol{\phi}$ is $\mathbb{F}_2$-linear. It can be extended to $\mathcal{R}_1^n$ naturally. For any $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{R}_1^n$, where $x_i = p_i + vq_i$, $1 \leq i \leq n$, one may uniquely define as

$$\phi(x) = (p, p+q),$$

for $p = (p_1, p_2, \ldots, p_n)$ and $q = (q_1, q_2, \ldots, q_n)$.

From two different perspectives, this ring is Frobenius. For the finite commutative ring $\mathcal{R}_1$ the Jacobson radical is

$$J(\mathcal{R}_1) = \{0\}$$

and the socle is

$$soc(\mathcal{R}_1) = \{0, 1, v, 1+v\}.$$

Since $\mathcal{R}_1/J(\mathcal{R}_1) = \mathcal{R}_1/\langle 0 \rangle \cong \mathcal{R}_1$ and $soc(\mathcal{R}_1) = \mathcal{R}_1$, we can say that

$$\mathcal{R}_1/J(\mathcal{R}_1) \cong soc(\mathcal{R}_1).$$

Therefore, from Definition 2.1.1, $\mathcal{R}_1$ is Frobenius.

The generating character for $\mathcal{R}_1$ can be defined as

$$\chi(a+vb)=(-1)^b. \tag{2.1}$$

By Theorem 2.1.3, $\mathcal{R}_1$ is Frobenius.

Three different weights are considered over $\mathcal{R}_1$, which are the Hamming, Lee and Bachoc weights. The term *Hamming weight* is used for the number of non-zero components of the element. The *Lee weights* of the elements of $\mathcal{R}_1$ are determined in [12] as

$$w_L(x)=\begin{cases} 0 & \text{if } x=0,\\ 1 & \text{if } x=v \text{ or } 1+v,\\ 2 & \text{if } x=1. \end{cases}$$

Hence, the Gray map $\phi$ is a weight-preserving map from ($\mathcal{R}_1^n$, Lee weight) to ($\mathbb{F}_2^{2n}$, Hamming weight). The connection between the weights is given by $w_L(x)=w_H(\phi(x))$. The *Bachoc weights* of the elements of $\mathcal{R}_1$ are determined in [8] as

$$w_B(x)=\begin{cases} 0 & \text{if } x=0,\\ 1 & \text{if } x=1,\\ 2 & \text{if } x=v \text{ or } 1+v. \end{cases}$$

The Lee and Bachoc weights of any word in $\mathcal{R}_1$ are the rational sums of the Lee and Bachoc weights of its components, respectively.

Now let us give two inner products defined on $\mathcal{R}_1$. Let $\mathbf{x}=(x_1,x_2,...,x_n)$ and $\mathbf{y}=(y_1,y_2,...,y_n)$ be two words of $\mathcal{R}_1^n$. The *EIP* is defined as

$$\langle \mathbf{x},\mathbf{y}\rangle=\sum_{i=1}^{n} x_i y_i,$$

and the *HIP* is defined as

$$[\mathbf{x},\mathbf{y}]=\sum_{i=1}^{n} x_i \overline{y_i},$$

where $\overline{0}=0$, $\overline{1}=1$, $\overline{v}=1+v$, $\overline{1+v}=v$. Note that this corresponds, in general, to $\overline{a+vb}=(a+b)+vb$ for all $a,b\in\mathbb{F}_2$, which is a ring automorphism different from the identity.

### 2.2.2. Linear Codes over $\mathcal{R}_1$

A *linear code* $C$ over $\mathcal{R}_1$ of length $n$ is defined to be an $\mathcal{R}_1$-submodule of $\mathcal{R}_1^n$. The *dual code* $C^{\perp}$ with respect to the EIP of $C$ is defined as

$$C^{\perp}=\{\mathbf{x}\in\mathcal{R}_1^n \mid \langle\mathbf{x},\mathbf{c}\rangle=0 \text{ for all } \mathbf{c}\in C\}$$

and the *dual code* $C^*$ with respect to the HIP of $C$ is defined as

$$C^{*}=\{\mathbf{x}\in\mathcal{R}_1^n \mid [\mathbf{x},\mathbf{c}]=0 \text{ for all } \mathbf{c}\in C\}.$$

A code $C$ is *Euclidean self-orthogonal* if $C\subseteq C^{\perp}$ and *Hermitian self-orthogonal* if $C\subseteq C^*$. Also, $C$ is *Euclidean self-dual* if $C=C^{\perp}$ and *Hermitian self-dual* if $C=C^*$.

**Example 2.2.1.** Let us consider the subset $C$ of $\mathcal{R}_1^3$ given as

$$\begin{aligned}C=\{&\,000,\ 101,\ 0v1,\ 1v0,\ v0v,\ 0vv,\ vv0,\ 1+v01+v,\ 001+v,\ 1+v00,\ 1v1+v,\ 10v,\\ &vv1+v,\ v01,\ 1+vvv,\ 1+vv1\}.\end{aligned}$$

Then $C$ is a linear code over $\mathcal{R}_1$ and $|C|=16$. The Euclidean dual of $C$ is

$$C^{\perp}=\{\,000,\ 01+v0,\ v1v,\ vvv\}$$

and the Hermitian dual of $C$ is

$$C^{*}=\{\,000,\ 0v0,\ 1+v11+v,\ 1+v1+v1+v\}.$$

A non-zero linear code $C$ over $\mathcal{R}_1$ is permutation equivalent to a code with generator matrix of the form

$$G=\begin{pmatrix} I_{k_1} & A_1 & B_1 & D_1+vD_2\\ 0 & vI_{k_2} & 0 & vE_1\\ 0 & 0 & (1+v)I_{k_3} & (1+v)F_1\end{pmatrix},$$

where $A_1$, $B_1$, $D_1$, $D_2$, $E_1$ and $F_1$ are binary matrices and $|C| = 4^{k_1}2^{k_2}2^{k_3}$ [13].

In our research we deal with the free codes that we define as follows:

**Definition 2.2.1.** A code that is generated as a free $\mathcal{R}_1$-module is called a *free code* over $\mathcal{R}_1$. By using the row operations and the properties of $\mathcal{R}_1$, any free code over $\mathcal{R}_1$ can be brought to an equivalent form which is generated by the rows of the matrix $[I_k, A+vB]$, where $A$ and $B$ are $k \times k$ binary matrices and $I_k$ is the $k \times k$ identity matrix. The size of a free code is $4^k$.

Let $C = \phi^{-1}(C_1, C_2)$ be a code over $\mathcal{R}_1$. In this case, $C$ can be denoted as $CRT(C_1, C_2)$, that is, the Chinese product of $C_1$ and $C_2$, where $C_1$ and $C_2$ are binary codes and they are uniquely determined for each $C$.

**Proposition 2.2.1.** Let $C$ be the free code given by $CRT(C_1, C_2)$. If $G = [I_k, A+vB]$ generates $C$, then the corresponding binary codes $C_1$ and $C_2$ have generator matrices $G_1 = [I_k, A]$ and $G_2 = [I_k, A+B]$, respectively.

**Proof:** Let the matrix $G = [I_k, A+vB]$ generate $C$. From Corollary 3.2 in [13], one can easily observe that the Gray image of $C$ is generated by

$$G' = \begin{bmatrix} \phi((1+v)G) \\ \phi(vG) \end{bmatrix} = \begin{bmatrix} I_k & A & 0 & 0 \\ 0 & 0 & I_k & A+B \end{bmatrix}.$$

This proves the result. ■

The minimum Hamming and Lee distances of a linear code $C$ is the smallest Hamming and Lee weights of non-zero codewords of $C$, respectively. Since we study linear codes, the notions Hamming distance and Lee distance coincide with the notions Hamming weight and Lee weight, respectively.

We consider the following result which is about the minimum Lee weight of a code $C$ over $\mathcal{R}_1$.

**Proposition 2.2.2.** [14] Let $d_H$ and $d_L$ be the minimum Hamming distance and the minimum Lee distance of a code $C = CRT(C_1, C_2)$, respectively. Then

$$d_H(C) = d_L(C) = min\{d(C_1), d(C_2)\},$$

where $d(C_1)$ and $d(C_2)$ denote the minimum distances of the binary codes $C_1$ and $C_2$, respectively.

### 2.2.3. Euclidean Self-Dual Codes over $\mathcal{R}_1$

We begin with the following theoretical results.

**Proposition 2.2.3.** [34] A code $C = CRT(C_1, C_2)$ is Euclidean self-dual if and only if $C_1$ and $C_2$ are both binary self-dual codes.

**Remark 2.2.1.** Binary self-dual codes of length $n$ exist if and only if $n$ is even. Moreover, binary self-dual codes have even weights.

**Corollary 2.2.1.** [12] A Euclidean self-dual code of length $n$ exists if and only if $n$ is even.

**Proposition 2.2.4.** [12] A code $CRT(C_1, C_2)$ is Euclidean Type IV self-dual if and only if $C_1 = C_2$.

The following theorem gives the necessary and sufficient conditions on the generator matrix of a free Euclidean self-dual code over $\mathcal{R}_1$.

**Theorem 2.2.1.** The matrix $G = [I_k, A + vB]$ generates a Euclidean self-dual code over $\mathcal{R}_1$ of length $2k$ if and only if $A$ is an orthogonal matrix and $AB^T + BA^T + BB^T = 0$, where $A$ and $B$ are $k \times k$ square matrices.

**Proof:** Let $G$ be the generator matrix of a Euclidean self-dual code. Then

$$G \cdot G^T = 0.$$

This means

$$[I_k, A + vB] \cdot [I_k, A + vB]^T = [I_k, A + vB] \cdot \begin{bmatrix} I_k \\ A^T + vB^T \end{bmatrix} = 0.$$

After the necessary algebraic operations, we find

$$(I_k + AA^T) + v(AB^T + BA^T + BB^T) = 0.$$

Hence we have $AA^T = I_k$ and $AB^T + BA^T + BB^T = 0$.

Conversely, assume that $AA^T = I_k$ and $AB^T + BA^T + BB^T = 0$. Then $G \cdot G^T = 0$ which implies $G$ generates a Euclidean self-orthogonal code. Since $G$ has $k$ rows and $2k$ columns, $G$ generates a Euclidean self-dual code. ■

We obtain the following results depending on what the matrix $B$ is.

**Remark 2.2.2.** Let $C$ be a free Euclidean self-dual code with the generator matrix $G = [I_k, A + vB]$, where $A$ and $B$ are $k \times k$ square matrices. According to Theorem 2.2.1, $B$ cannot be equal to $A$. Otherwise we would obtain $I_k = 0$ which is a contradiction.

**Corollary 2.2.2.** A Euclidean self-dual code which is generated by the matrix $G = [I_k, A + vB]$ is of Type IV if and only if the matrix $B$ is a zero matrix.

**Proof:** Let $CRT(C_1, C_2)$ be a Euclidean self-dual code of Type IV generated by the matrix $G = [I_k, A + vB]$. By Proposition 2.2.1, $C_1$ and $C_2$ have a generator matrices of the form $[I_k, A]$ and $[I_k, A + vB]$, respectively. Then by Proposition 2.2.4, $C_1$ is equal to $C_2$. Hence the generator matrices $[I_k, A]$ and $[I_k, A + vB]$ must generate the same code, which is only possible when $B$ is zero.

Conversely, let $B$ be a zero matrix. Then $C_1$ and $C_2$ have the same generator matrices, which implies that $CRT(C_1, C_2)$ is a Euclidean self-dual code of Type IV. ■

### 2.2.4. Hermitian Self-Dual Codes over $\mathcal{R}_1$

By Corollary 2.2.1, Euclidean self-dual codes exist for even lengths, but Hermitian self-dual codes over $\mathcal{R}_1$ exist for all lengths. For example, $1 + v$ generates a Hermitian self-dual code of length 1.

The following results are useful tools for Hermitian self-dual codes.

**Proposition 2.2.5.** [8] A code $C = CRT(C_1, C_2)$ is Hermitian self-dual if and only if $C_2 = C_1^{\perp}$.

**Proposition 2.2.6.** [12] A code $CRT(C_1, C_2)$ is Hermitian Type IV self-dual if and only if $C_1$ and $C_2$ are even.

**Corollary 2.2.3.** [12] A Euclidean Type IV self-dual code $CRT(C_1,C_2)$ is also Hermitian Type IV self-dual.

The following proposition gives an upper bound on minimum weights of Hermitian self-dual codes over $\mathcal{R}_1$.

**Proposition 2.2.7.** [14] Let $d_{max}(n,k)$ be the highest minimum weight among all binary linear $[n,k]$ codes. The highest minimum weight $d_{SD}(n)$ among all Hermitian self-dual codes of length n over $\mathcal{R}_1$ is bounded by

$$d_{SD}(n) \leq d_{max}(n, \lfloor (n+1)/2 \rfloor).$$

The following theorem gives the necessary and sufficient conditions on the generator matrix of a free Hermitian self-dual code over $\mathcal{R}_1$.

**Theorem 2.2.2.** The matrix $G = [I_k, A+vB]$ generates a Hermitian self-dual code over $\mathcal{R}_1$ of length $2k$ if and only if $A$ is an invertible matrix and $B = A + (A^T)^{-1}$, where $A$ and $B$ are $k \times k$ square matrices.

**Proof:** First, recall that $\overline{a+vb} = (a+b)+vb$. Since $G$ generates a Hermitian self-dual code, $G \cdot \overline{G^T} = 0$. This implies

$$[I, A+vB] \cdot \overline{[I, A+vB]^T} = [I, A+vB] \cdot \begin{bmatrix} I \\ (A^T+B^T)+vB^T \end{bmatrix} = 0.$$

After the necessary matrix operations and simplifications, we obtain

$$(I + AA^T + AB^T) + v(AB^T + BA^T) = 0.$$

Hence $A$ must be an invertible matrix with the inverse $(A+B)^T$. Then $B = A + (A^T)^{-1}$.

Conversely, assume that $A$ is an invertible matrix and $B = A + (A^T)^{-1}$, then $G \cdot \overline{G^T} = 0$. Therefore, $G$ generates a Hermitian self-orthogonal code. Since $G$ has $k$ rows and $2k$ columns, $G$ generates a Hermitian self-dual code. ■

The next observation is worth pointing out. In view of Proposition 2.2.1 and Proposition 2.2.5, we have that a free Hermitian self-dual code $C$ is of the form $C = CRT(C_1, C_1^{\perp})$, where $C_1$ and $C_1^{\perp}$ have generator matrices of the form $[I_k, A]$ and $[I_k, A']$, respectively. Since $[A^T, I_k]$ is the generator matrix of $C_1^{\perp}$, $[A^T, I_k]$ is not row equivalent to a matrix of the form $[I_k, A']$ unless $A$ is invertible as seen in the following example:

**Example 2.2.2.** The matrix

$$\left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{array}\right]$$

generates a binary formally self-dual code $C$ and its dual code $C^{\perp}$ is generated by

$$\left[\begin{array}{ccc|ccc} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

which cannot be converted into the form $[I, X]$ using elementary row operations.

**Corollary 2.2.4.** If $C = CRT(C_1, C_1^{\perp})$ is a Hermitian self-dual code generated by $G = [I_k, A + vB]$, then $C_1$ and $C_1^{\perp}$ have generator matrices $G_1 = [I, A]$ and $G_2 = [I, (A^T)^{-1}]$, respectively.

**Proof:** Since $G_1 = [I_k, A]$, $G_2 = [I_k, A + B]$ and $B = A + (A^T)^{-1}$, the proof directly follows from Theorem 2.2.2, Propositions 2.2.1 and 2.2.5. ■

### 2.2.5. The Weight Enumerators and the MacWilliams Identities over $\mathcal{R}_1$

In this subsection, we mention several weight enumerators of a linear code over $\mathcal{R}_1$. In [33], Klemm constituted the MacWilliams identities for codes over $\mathbb{Z}_4$. In [3], Wood gave the MacWilliams relations for the complete weight enumerator (CWE), Hamming weight enumerator (HWE) and symmetrized weight enumerator (SWE) over any finite Frobenius ring. In light of these two studies ([3],[33]), Dougherty et al. implied the result about the MacWilliams relation for the SWE between a code and its dual over any commutative ring of order 4 in [12]. Now we summarise briefly the MacWilliams identities for the CWE, SWE, HWE and Lee Weight Enumerator (LWE) over $\mathcal{R}_1$, which yield some equalities between different weight enumerators.

Let $\mathcal{R}_1 = \{f_1, f_2, f_3, f_4\}$ be in fixed order as

$$\mathcal{R}_1 = \{0, 1, v, 1+v\}.$$

The CWE of a linear code $C$ over $\mathcal{R}_1$ of length $n$ is defined as

$$cwe_C(x_1,x_2,x_3,x_4) = \sum_{\mathbf{c}\in C} \left( x_1^{w_{f_1}(\mathbf{c})} x_2^{w_{f_2}(\mathbf{c})} x_3^{w_{f_3}(\mathbf{c})} x_4^{w_{f_4}(\mathbf{c})} \right)$$

where $w_{f_i}(\mathbf{c}) = |\{j|c_j = f_i\}|$ for $\mathbf{c} = (c_1, c_2, \ldots, c_n)$, $1 \le i \le 4$ and $1 \le j \le n$. $cwe_C$ is a homogeneous polynomial with four indeterminates $x_1$, $x_2$, $x_3$, and $x_4$, and the total degree of each term is $n$.

Since $C$ is a linear code, the term $x_1^n$ appears in $cwe_C(x_1,x_2,x_3,x_4)$. We see that $cwe_C(1,1,1,1) = |C|$. Permutation equivalent codes have the same CWE.

From (2.1), we construct the matrix $K = (k_{i,j})_{4\times 4}$ defined as $k_{i,j} = \chi(f_i f_j)$, where $f_i, f_j \in \mathcal{R}_1$. Therefore the matrix $K$ is obtained as

$$K = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}.$$

**Theorem 2.2.3.** Let $C^\perp$ be the Euclidean dual code of a linear code $C$ over $\mathcal{R}_1$ of length $n$. Then

$$cwe_{C^\perp}(x_1,x_2,x_3,x_4) = \frac{1}{|C|} cwe_C(K(x_1,x_2,x_3,x_4)^T),$$

where $(\ )^T$ denotes the transpose.

**Proof:** The matrix $K$ is constructed depending on the character given in (2.1). From Theorem 3.2 in [35], the proof is completed. ■

**Example 2.2.3.** Let $C$ be the linear code over $\mathcal{R}_1$ in Example 2.2.1. Then

$$cwe_C(x_1,x_2,x_3,x_4) = x_1^3 + x_1x_2^2 + 3x_1x_3^2 + 2x_1^2x_4 + 2x_3^2x_4 + x_1x_4^2 + 4x_1x_2x_3 + 2x_2x_3x_4$$

and

$$cwe_{C^{\perp}}(x_1,x_2,x_3,x_4) = x_1^3 + x_1^2x_4 + x_2x_3^2 + x_3^3.$$

One can obtain that

$$cwe_{C^{\perp}}(x_1,x_2,x_3,x_4) = \frac{1}{16}cwe_C\left(K(x_1,x_2,x_3,x_4)^T\right).$$

For an equivalence class of codes, the suitable weight enumerator is the SWE. We can obtain the SWE from the CWE by grouping the elements of the same weight. In $\mathcal{R}_1$, by identifying $x_1$ as $a$, $x_3,x_4$ as $b$ and $x_2$ as $d$ in the CWE of a linear code $C$, we obtain the SWE as in the following:

$$\begin{aligned} swe_C(a,b,d) &= cwe_C(a,d,b,b) \\ &= \sum_{\mathbf{c}\in C}\left(a^{n_0(\mathbf{c})}b^{n_1(\mathbf{c})}d^{n_2(\mathbf{c})}\right), \end{aligned}$$

where $n_0(\mathbf{c}) = w_{f_1}(\mathbf{c})$, $n_1(\mathbf{c}) = w_{f_3}(\mathbf{c}) + w_{f_4}(\mathbf{c})$, and $n_2(\mathbf{c}) = w_{f_2}(\mathbf{c})$. $swe_C$ is a homogeneous polynomial of degree $n$ with three indeterminates $a,b$ and $d$, that is, $n = n_0(\mathbf{c}) + n_1(\mathbf{c}) + n_2(\mathbf{c})$.

**Theorem 2.2.4.** Let $C^{\perp}$ be the Euclidean dual code of a linear code $C$ over $\mathcal{R}_1$ of length $n$. Then

$$swe_{C^{\perp}}(a,b,d) = \frac{1}{|C|}swe_C(S(a,b,d)^T),$$

where

$$S = \begin{pmatrix} 1 & 2 & 1 \\ 1 & 0 & -1 \\ 1 & -2 & 1 \end{pmatrix}.$$

**Proof:** From the definition of the SWE and Theorem 2.2.3, we obtain that

$$\begin{aligned} swe_{C^{\perp}}(a,b,d) = &\ cwe_{C^{\perp}}(a,d,b,b) \\ = &\ \tfrac{1}{|C|}\, cwe_C\left(K(a,d,b,b)^T\right) \\ = &\ \tfrac{1}{|C|}\, swe_C(a+2b+d, a-d, a-2b+d) \\ = &\ \tfrac{1}{|C|}\, swe_C(S(a,b,d)^T). \end{aligned}$$

■

**Example 2.2.4.** Let $C$ be the linear code over $\mathcal{R}_1$ in Example 2.2.1. Then

$$swe_C(a,b,d) = a^3 + 2d^3 + ab^2 + 4ad^2 + 2a^2d + 4abd + 2bd^2$$

and

$$swe_{C^{\perp}}(a,b,d) = a^3 + a^2d + bd^2 + d^3.$$

One can obtain that

$$swe_{C^{\perp}}(a,b,d) = \frac{1}{16}\, swe_C(S(a,b,d)^T).$$

**Remark 2.2.3.** The CWE of the Hermitian dual of a linear code $C$ can be obtained in a similar way. Actually, if we interchange the coordinates $x_3$ and $x_4$ in the CWE of the Euclidean dual code, we find the following equality

$$cwe_{C^*}(x_1,x_2,x_3,x_4) = cwe_{C^{\perp}}(x_1,x_2,x_4,x_3).$$

For example, the CWE of $C^*$, given in Example 2.2.1, is

$$\begin{aligned} cwe_{C^*}(x_1,x_2,x_3,x_4) &= x_1^3 + x_1^2x_3 + x_2x_4^2 + x_4^3 \\ &= cwe_{C^{\perp}}(x_1,x_2,x_4,x_3). \end{aligned}$$

The SWE of the Hermitian dual of a linear code $C$ is

$$\begin{aligned} swe_{C^*}(a,b,d) &= cwe_{C^*}(a,d,b,b) \\ &= cwe_{C^\perp}(a,d,b,b) \\ &= swe_{C^\perp}(a,b,d) \end{aligned}$$

since the Lee weights of $v$ and $1+v$ are the same.

The HWE of a linear code $C$ over $\mathcal{R}_1$ is given by

$$W_C(x,y) = \sum_{\mathbf{c}\in C} x^{n-w_H(\mathbf{c})} y^{w_H(\mathbf{c})} = cwe_C(x,y,y,y)$$

which is a homogeneous polynomial of degree $n$ with two indeterminates $x$ and $y$.

**Theorem 2.2.5.** Let $C^\perp$ be the Euclidean dual code of a linear code $C$ over $\mathcal{R}_1$. Then

$$W_{C^\perp}(x,y) = \frac{1}{|C|} W_C(x+3y, x-y).$$

**Proof:** From Theorem 2.2.3, we have

$$\begin{aligned} W_{C^\perp}(x,y) &= cwe_{C^\perp}(x,y,y,y) \\ &= \tfrac{1}{|C|} cwe_C(K(x,y,y,y)^T) \\ &= \tfrac{1}{|C|} cwe_C(x+3y, x-y, x-y, x-y) \\ &= \tfrac{1}{|C|} W_C(x+3y, x-y). \end{aligned}$$

■

**Example 2.2.5.** Let $C$ be the linear code over $\mathcal{R}_1$ in Example 2.2.1. Then

$$W_C(x,y) = x^3 + 2x^2y + 9xy^2 + 4y^3$$

and

$$\begin{aligned} W_{C^\perp}(x,y) = {} & \tfrac{1}{16}((x+3y)^3 + 2(x+3y)^2(x-y) + 9(x+3y)(x-y)^2 + 4(x-y)^3) \\ = {} & x^3 + 2y^3 + x^2y. \end{aligned}$$

The Lee weight of $\mathbf{c} \in \mathcal{R}_1^n$ can be denoted as $w_L(\mathbf{c}) = n_1(\mathbf{c}) + 2n_2(\mathbf{c})$. The LWE of a linear code $C$ over $\mathcal{R}_1$ is defined to be

$$Lee_C(x,y) = W_{\phi(C)}(x,y) = \sum_{\mathbf{c}\in C} x^{2n-w_L(\mathbf{c})} y^{w_L(\mathbf{c})}.$$

**Example 2.2.6.** Let $C$ be the linear code over $\mathcal{R}_1$ in Example 2.2.1. Then

$$Lee_C(x,y) = x^6 + 2x^5y + 4x^4y^2 + 6x^3y^3 + 3x^2y^4.$$

**Theorem 2.2.6.** Let $C$ be a linear code over $\mathcal{R}_1$. Then

$$swe_C(x^2, xy, y^2) = Lee_C(x,y).$$

**Proof:** From the definition of the SWE, we have

$$\begin{aligned} swe_C(x^2,xy,y^2) = & \sum_{\mathbf{c}\in C} (x^2)^{n_0(\mathbf{c})} (xy)^{n_1(\mathbf{c})} (y^2)^{n_2(\mathbf{c})} \\ = & \sum_{\mathbf{c}\in C} x^{2n_0(\mathbf{c})+n_1(\mathbf{c})} y^{n_1(\mathbf{c})+2n_2(\mathbf{c})} \\ = & \sum_{\mathbf{c}\in C} x^{2n-w_L(\mathbf{c})} y^{w_L(\mathbf{c})} \\ = & \; Lee_C(x,y). \end{aligned}$$

■

**Theorem 2.2.7.** Let $C^\perp$ be the Euclidean dual code of a linear code $C$ over $\mathcal{R}_1$. Then

$$Lee_{C^\perp}(x,y) = \frac{1}{|C|} Lee_C(x+y, x-y).$$

**Proof:** By using Theorem 2.2.4 and Theorem 2.2.6, we obtain

$$\begin{aligned} Lee_{C^\perp}(x,y) &= swe_{C^\perp}(x^2,xy,y^2) \\ &= \tfrac{1}{|C|} swe_C(S(x^2,xy,y^2)^T) \end{aligned}$$

$$= \frac{1}{|C|} swe_C((x+y)^2, (x+y)(x-y), (x-y)^2)$$

$$= \frac{1}{|C|} Lee_C(x+y, x-y).$$

■

## 2.3. SELF-DUAL CODES OVER THE RING $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$

### 2.3.1. The Ring $\mathcal{R}_2$

The direct product ring $\mathcal{R}_2 = \mathbb{F}_2 \times \mathcal{R}_1 = \mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$ can be defined as

$$\begin{aligned} \mathcal{R}_2 &= \{(f,r) | f \in \mathbb{F}_2 \text{ and } r \in \mathbb{F}_2 + v\mathbb{F}_2\} \\ &= \{(0,0), (0,1), (0,v), (0,1+v), (1,0), (1,1), (1,v), (1,1+v)\} \end{aligned}$$

where $v^2 = v$. This ring is a finite commutative non-chain ring of order 8. The characteristic of the ring is 2. Each element of the ring is idempotent. Hence, this ring is Boolean. The zero element of the ring is $(0,0)$ and the identity is $(1,1)$.

Let $(a, b+vd)$ and $(e, f+vg)$ be two elements of $\mathcal{R}_2$. Then the addition is defined by

$$(a, b+vd) + (e, f+vg) = (a+e, b+f+v(d+g))$$

and the multiplication is defined by

$$(a, b+vd) \cdot (e, f+vg) = (ae, bf+v(bg+df+dg)).$$

The addition and multiplication of elements of $\mathcal{R}_2$ are given in Table 2.1 and Table 2.2. Throughout the thesis, we will ignore the notation "$\cdot$" for convenience.

Let $(a, b+vd)$ and $r = (r_1, r_2+vr_3)$ be two elements of $\mathcal{R}_2$. We define the scalar multiplication on $\mathcal{R}_2$ as

$$r(a, b+vd) = (r_1a, r_2b + v(r_3b + (r_2+r_3)d)).$$

This multiplication is well-defined and can be extended to $\mathcal{R}_2^n$ as in the following:

$$r\mathbf{x} = ((r_1a_1, r_2b_1 + v(r_3b_1 + (r_2+r_3)d_1), \ldots, (r_1a_n, r_2b_n + v(r_3b_n + (r_2+r_3)d_n))),$$

where $\mathbf{x} = (x_1, \ldots, x_n) \in \mathcal{R}_2^n$ such that $x_i = (a_i, b_i + vd_i) \in \mathcal{R}_2$ for $1 \leq i \leq n$. Therefore, $\mathcal{R}_2^n$ is an $\mathcal{R}_2$-module under the scalar multiplication.

**Table 2.1:** The addition operation performed on $\mathcal{R}_2$.

| $+$ | $(0,0)$ | $(0,1)$ | $(0,v)$ | $(0,1+v)$ | $(1,0)$ | $(1,1)$ | $(1,v)$ | $(1,1+v)$ |
|---|---|---|---|---|---|---|---|---|
| $(0,0)$ | $(0,0)$ | $(0,1)$ | $(0,v)$ | $(0,1+v)$ | $(1,0)$ | $(1,1)$ | $(1,v)$ | $(1,1+v)$ |
| $(0,1)$ | $(0,1)$ | $(0,0)$ | $(0,1+v)$ | $(0,v)$ | $(1,1)$ | $(1,0)$ | $(1,1+v)$ | $(1,v)$ |
| $(0,v)$ | $(0,v)$ | $(0,1+v)$ | $(0,0)$ | $(0,1)$ | $(1,v)$ | $(1,1+v)$ | $(1,0)$ | $(1,1)$ |
| $(0,1+v)$ | $(0,1+v)$ | $(0,v)$ | $(0,1)$ | $(0,0)$ | $(1,1+v)$ | $(1,v)$ | $(1,1)$ | $(1,0)$ |
| $(1,0)$ | $(1,0)$ | $(1,1)$ | $(1,v)$ | $(1,1+v)$ | $(0,0)$ | $(0,1)$ | $(0,v)$ | $(0,1+v)$ |
| $(1,1)$ | $(1,1)$ | $(1,0)$ | $(1,1+v)$ | $(1,v)$ | $(0,1)$ | $(0,0)$ | $(0,1+v)$ | $(0,v)$ |
| $(1,v)$ | $(1,v)$ | $(1,1+v)$ | $(1,0)$ | $(1,1)$ | $(0,v)$ | $(0,1+v)$ | $(0,0)$ | $(0,1)$ |
| $(1,1+v)$ | $(1,1+v)$ | $(1,v)$ | $(1,1)$ | $(1,0)$ | $(0,1+v)$ | $(0,v)$ | $(0,1)$ | $(0,0)$ |

**Table 2.2:** The multiplication operation performed on $\mathcal{R}_2$.

| $\cdot$ | $(0,0)$ | $(0,1)$ | $(0,v)$ | $(0,1+v)$ | $(1,0)$ | $(1,1)$ | $(1,v)$ | $(1,1+v)$ |
|---|---|---|---|---|---|---|---|---|
| $(0,0)$ | $(0,0)$ | $(0,0)$ | $(0,0)$ | $(0,0)$ | $(0,0)$ | $(0,0)$ | $(0,0)$ | $(0,0)$ |
| $(0,1)$ | $(0,0)$ | $(0,1)$ | $(0,v)$ | $(0,1+v)$ | $(0,0)$ | $(0,1)$ | $(0,v)$ | $(0,1+v)$ |
| $(0,v)$ | $(0,0)$ | $(0,v)$ | $(0,v)$ | $(0,0)$ | $(0,0)$ | $(0,v)$ | $(0,v)$ | $(0,0)$ |
| $(0,1+v)$ | $(0,0)$ | $(0,1+v)$ | $(0,0)$ | $(0,1+v)$ | $(0,0)$ | $(0,1+v)$ | $(0,0)$ | $(0,1+v)$ |
| $(1,0)$ | $(0,0)$ | $(0,0)$ | $(0,0)$ | $(0,0)$ | $(1,0)$ | $(1,0)$ | $(1,0)$ | $(1,0)$ |
| $(1,1)$ | $(0,0)$ | $(0,1)$ | $(0,v)$ | $(0,1+v)$ | $(1,0)$ | $(1,1)$ | $(1,v)$ | $(1,1+v)$ |
| $(1,v)$ | $(0,0)$ | $(0,v)$ | $(0,v)$ | $(0,0)$ | $(1,0)$ | $(1,v)$ | $(1,v)$ | $(1,0)$ |
| $(1,1+v)$ | $(0,0)$ | $(0,1+v)$ | $(0,0)$ | $(0,1+v)$ | $(1,0)$ | $(1,1+v)$ | $(1,0)$ | $(1,1+v)$ |

The ideals of $\mathcal{R}_2$ are given by

- $\langle(0,0)\rangle = \{(0,0)\}$,
- $\langle(0,v)\rangle = \{(0,0),(0,v)\}$,
- $\langle(0,1+v)\rangle = \{(0,0),(0,1+v)\}$,
- $\langle(1,0)\rangle = \{(0,0),(1,0)\}$,
- $\mathrm{I}_1 = \langle(0,1)\rangle = \{(0,0),(0,1),(0,v),(0,1+v)\}$,

- $\mathrm{I}_2 = \langle (1,v) \rangle = \{(0,0),(0,v),(1,0),(1,v)\}$,
- $\mathrm{I}_3 = \langle (1,1+v) \rangle = \{(0,0),(0,1+v),(1,0),(1,1+v)\}$,
- $\mathcal{R}_2 = \langle (1,1) \rangle = \{(0,0),(0,1),(0,v),(0,1+v),(1,0),(1,1),(1,v),(1,1+v)\}$.

Note that $\mathrm{I}_1$, $\mathrm{I}_2$ and $\mathrm{I}_3$ are the maximal ideals. Since the number of the maximal ideals is finite, this ring is a semilocal ring. The ideal lattice is given in Figure 2.2.

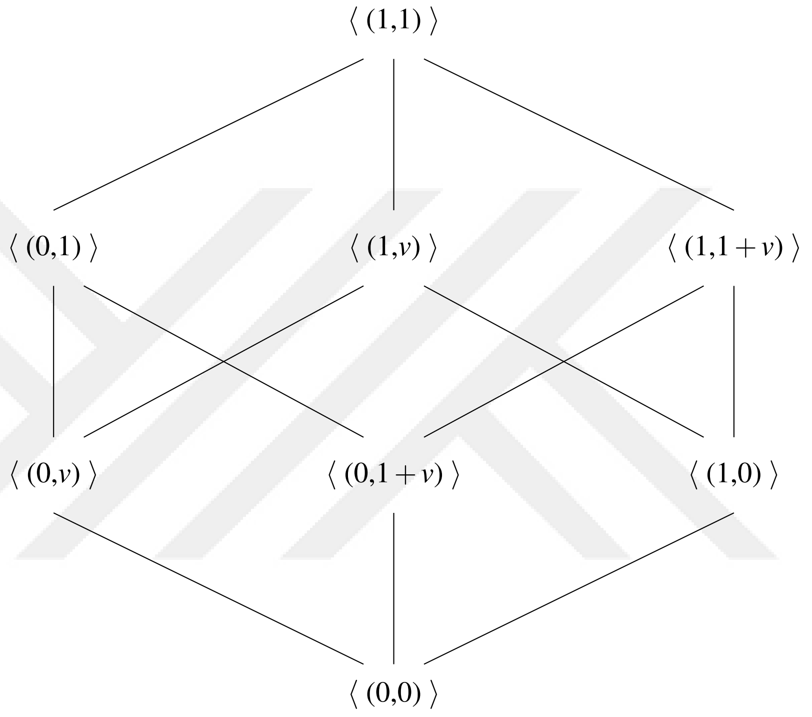

**Figure 2.2:** The lattice of the ideals of $\mathcal{R}_2$.

We can indicate in two different ways that $\mathcal{R}_2$ is Frobenius. For the finite commutative ring $\mathcal{R}_2$, the Jacobson radical is

$$J(\mathcal{R}_2) = \{(0,0)\}$$

and the socle is

$$soc(\mathcal{R}_2) = \{(0,0),(0,1),(0,v),(0,1+v),(1,0),(1,1),(1,v),(1,1+v)\}.$$

Since $\mathcal{R}_2/J(\mathcal{R}_2) = \mathcal{R}_2/\langle(0,0)\rangle \cong \mathcal{R}_2$ and $soc(\mathcal{R}_2) = \mathcal{R}_2$, we can say that

$$\mathcal{R}_2/J(\mathcal{R}_2) \cong soc(\mathcal{R}_2).$$

Therefore, from Definition 2.1.1, $\mathcal{R}_2$ is Frobenius.

The generating character for $\mathcal{R}_2$ can be defined as

$$\chi((a,b+vd)) = (-1)^{a+d}. \tag{2.2}$$

Hence, by Theorem 2.1.3, this ring is Frobenius.

From Proposition 2.1.1 and Proposition 2.1.2, $\mathcal{R}_2$ is isomorphic to the ring $\mathbb{F}_2 \times \mathbb{F}_2 \times \mathbb{F}_2$:

Since $\mathrm{I}_1$, $\mathrm{I}_2$ and $\mathrm{I}_3$ are the maximal ideals of $\mathcal{R}_2$ as listed above, by the CRT we obtain

$$\mathcal{R}_2 \cong \prod_{i=1}^{3} \mathcal{R}_2/\mathrm{I}_i \cong \mathbb{F}_2 \times \mathbb{F}_2 \times \mathbb{F}_2.$$

This result can also be obtained directly from Proposition 2.1.3.

### 2.3.2. Linear Codes over $\mathcal{R}_2$

A linear code $C$ over $\mathcal{R}_2$ of length $n$ is an $\mathcal{R}_2$-submodule of $\mathcal{R}_2^n$. Let us assume that a code $C$ over $\mathcal{R}_2$ of length $n$ is permutation equivalent to the direct product of codes $C_1$ and $C_2$, where $C_1$ is a binary code of length $n$ and $C_2$ is a code over $\mathcal{R}_1$ of length $n$. From now on, we can denote $C = (C_1, C_2)$.

**Proposition 2.3.1.** $C$ is linear if and only if $C_1$ and $C_2$ are linear.

**Example 2.3.1.** Let us consider the subset $C$ of $\mathcal{R}_2^2$ given as

$$\begin{aligned} C = \{ & (0,0)(0,0),\ (0,v)(0,v),\ (0,0)(1,0),\ (0,v)(1,v),\ (0,1+v)(0,0),\ (1,0)(1,0), \\ & (1,1+v)(1,0),\ (0,1)(0,v),\ (1,v)(1,v),\ (1,1)(1,v),\ (0,1+v)(1,0), \\ & (1,0)(0,0), (1,1+v)(0,0),\ (0,1)(1,v),\ (1,v)(0,v),\ (1,1)(0,v) \}. \end{aligned}$$

Then $C$ is a linear code over $\mathcal{R}_2$ and $|C| = 16$. If we take the linear code $C_1 = \{00, 01, 11, 10\}$ over $\mathbb{F}_2$ and the linear code $C_2 = \{00, vv, 1+v0, 1v\}$ over $\mathcal{R}_1$, then $C$ is permutation equivalent to $C_1 \times C_2$.

Any free code over $\mathcal{R}_2$ is permutation equivalent to a code whose generator matrix is of the form

$$[(I_k,I_k)|(A,B+vD)],$$

where

$$(I_k,I_k)=\begin{bmatrix}(1,1) & (0,0) & \cdots & (0,0)\\ (0,0) & (1,1) & \cdots & (0,0)\\ \vdots & \vdots & \ddots & \vdots\\ (0,0) & (0,0) & \cdots & (1,1)\end{bmatrix}$$

and

$$(A,B+vD)=\begin{bmatrix}(a_{11},b_{11}+vd_{11}) & (a_{12},b_{12}+vd_{12}) & \cdots & (a_{1k},b_{1k}+vd_{1k})\\ (a_{21},b_{21}+vd_{21}) & (a_{22},b_{22}+vd_{22}) & \cdots & (a_{2k},b_{2k}+vd_{2k})\\ \vdots & \vdots & \ddots & \vdots\\ (a_{k1},b_{k1}+vd_{k1}) & (a_{k2},b_{k2}+vd_{k2}) & \cdots & (a_{kk},b_{kk}+vd_{kk})\end{bmatrix}$$

with $a_{ij},b_{ij},d_{ij}\in\mathbb{F}_2$ for $1\leq i,j\leq k$. In short, we can write

$$(I_k,I_k)=[\beta_{ij}]_{k\times k}$$

and

$$(A,B+vD)=[\gamma_{ij}]_{k\times k}$$

such that $\beta_{ij}=(\delta_{ij},\delta_{ij})$ with $\delta_{ij}$ denoting the Kroenecker delta function and $\gamma_{ij}=(a_{ij},b_{ij}+vd_{ij})\in\mathbb{F}_2\times\mathcal{R}_1$ for $1\leq i,j\leq k$. In this dissertation, we study free self-dual codes whose generator matrix is of the form $[(I_k,I_k)|(A,B+vD)]$.

For linear codes over $\mathcal{R}_2$, we consider two different Gray maps and two different inner products.

### 2.3.3. Euclidean Self-Dual Codes over $\mathcal{R}_2$

**Definition 2.3.1.** Let $(a, b+vd)$ be an element of $\mathcal{R}_2$. We define the Lee weight of $(a, b+vd)$ as

$$w_L((a, b+vd)) = w_H(a) + w_L(b+vd),$$

where $w_H(a)$ denotes the Hamming weight of $a \in \mathbb{F}_2$ and $w_L(b+vd)$ denotes the Lee weight of $b+vd \in \mathcal{R}_1$. Recall that $w_L(0) = 0$, $w_L(1) = 2$, $w_L(v) = 1$ and $w_L(1+v) = 1$.

As a result of Definition 2.3.1, we can see that

$$w_L(x) = \begin{cases} 0 & \text{if } x = (0,0), \\ 1 & \text{if } x = (0,v), (0,1+v), (1,0), \\ 2 & \text{if } x = (0,1), (1,v), (1,1+v), \\ 3 & \text{if } x = (1,1). \end{cases}$$

The Lee weight of a word $\mathbf{x} = (x_1, \ldots, x_n) \in \mathcal{R}_2^n$ is the rational sum of the Lee weight of its components, i.e., $w_L(\mathbf{x}) = \sum_{i=1}^{n} w_L(x_i)$. The Lee distance between two elements is given as $d_L(\mathbf{x}, \mathbf{y}) = w_L(\mathbf{x} - \mathbf{y})$ for any $\mathbf{x}, \mathbf{y} \in \mathcal{R}_2^n$. The Lee weight of a code $C$, denoted as $w_L(C)$, is the smallest non-zero Lee weight among all codewords of $C$. The Lee distance of $C$, denoted as $d_L(C)$, is the smallest non-zero Lee distance between all pairs of distinct codewords of $C$.

**Example 2.3.2.** Let $C$ be the linear code over $\mathcal{R}_2$ given in Example 2.3.1. Then the Lee weight of the code word $\mathbf{x} = (0, 1+v)(1,0)$ of $C$ is $w_L(\mathbf{x}) = 2$. The Lee distance between $\mathbf{y} = (1, 1+v)(1,0)$ and $\mathbf{z} = (0,1)(1,v)$ is $d_L(\mathbf{y}, \mathbf{z}) = 3$. The Lee weight and the Lee distance of $C$ coincide, which is $w_L(C) = d_L(C) = 1$, since $C$ is linear.

We define the Gray map $\boldsymbol{\phi_1}$ as

$$\boldsymbol{\phi_1} : \mathcal{R}_2 \to \mathbb{F}_2^3, \ \ (a, b+vd) \mapsto (a, b, b+d),$$

which is a ring isomorphism by the CRT. This map is extended to $\mathcal{R}_2^n$ as follows:

$$\phi_1((\mathbf{a},\mathbf{b}+v\mathbf{d})) = (\mathbf{a},\mathbf{b},\mathbf{b}+\mathbf{d}),$$

where $\mathbf{a},\mathbf{b},\mathbf{d} \in \mathbb{F}_2^n$.

**Proposition 2.3.2.** The Gray map $\phi_1$ is $\mathbb{F}_2$-linear.

**Proof:** Let $\mathbf{x} = (\mathbf{a},\mathbf{b}+v\mathbf{d})$ and $\mathbf{y} = (\mathbf{e},\mathbf{f}+v\mathbf{g})$ be two elements of $\mathcal{R}_2^n$ and $r,s \in \mathbb{F}_2$. Then $r\mathbf{x}+s\mathbf{y} = (r\mathbf{a}+s\mathbf{e}, r\mathbf{b}+s\mathbf{f}+v(r\mathbf{d}+s\mathbf{g}))$.

$$\begin{aligned}
\phi_1(r\mathbf{x}+s\mathbf{y}) &= \phi_1((r\mathbf{a}+s\mathbf{e}, r\mathbf{b}+s\mathbf{f}+v(r\mathbf{d}+s\mathbf{g}))) \\
&= (r\mathbf{a}+s\mathbf{e}, r\mathbf{b}+s\mathbf{f}, r\mathbf{b}+s\mathbf{f}+r\mathbf{d}+s\mathbf{g}) \\
&= (r\mathbf{a}, r\mathbf{b}, r\mathbf{b}+r\mathbf{d}) + (s\mathbf{e}, s\mathbf{f}, s\mathbf{f}+s\mathbf{g}) \\
&= r(\mathbf{a},\mathbf{b},\mathbf{b}+\mathbf{d}) + s(\mathbf{e},\mathbf{f},\mathbf{f}+\mathbf{g}) \\
&= r\phi_1(\mathbf{x}) + s\phi_1(\mathbf{y}).
\end{aligned}$$

■

**Proposition 2.3.3.** If $C$ is a linear code over $\mathcal{R}_2$ of length $n$, then $\phi_1(C)$ is a binary linear code of length $3n$.

**Example 2.3.3.** Let $C$ be the linear code over $\mathcal{R}_2$ given in Example 2.3.1. Then

$$\begin{aligned}
\phi_1(C) = \{\, &000000, 000011, 010000, 010011, 001000, 110000, 111000, 001011, \\
&110011, 111011, 011000, 100000, 101000, 011011, 100011, 101011 \,\}
\end{aligned}$$

is a binary linear code of length 6.

**Proposition 2.3.4.** The Gray map $\phi_1$ is a distance preserving map from $\mathcal{R}_2^n$ (Lee distance-$d_L$) to $\mathbb{F}_2^{3n}$ (Hamming distance-$d_H$).

**Proof:** From Proposition 2.3.2,

$$\phi_1(\mathbf{x}-\mathbf{y}) = \phi_1(\mathbf{x}) - \phi_1(\mathbf{y})$$

for $\mathbf{x},\mathbf{y} \in \mathcal{R}_2^n$. By Definition 2.3.1,

$$d_L(\mathbf{x},\mathbf{y}) = w_L(\mathbf{x}-\mathbf{y}) = w_H(\phi_1(\mathbf{x}-\mathbf{y})) = w_H(\phi_1(\mathbf{x}) - \phi_1(\mathbf{y})) = d_H(\phi_1(\mathbf{x}),\phi_1(\mathbf{y})).$$

■

Now, in order to present the dual of a linear code over $\mathcal{R}_2$, we first need to define an inner product. We define the EIP on $\mathcal{R}_2^n$ as

$$\langle \mathbf{x},\mathbf{y}\rangle = \sum_{i=1}^{n} x_i y_i,$$

where $\mathbf{x} = (x_1,x_2,\ldots,x_n), \mathbf{y} = (y_1,y_2,\ldots,y_n) \in \mathcal{R}_2^n$. This inner product is exactly like the natural EIP. Also we can define the EIP with regard to the ordinary dot product in $\mathbb{F}_2$ and the EIP in $\mathcal{R}_1$ as follows:

$$\langle\, (\mathbf{a},\mathbf{b}+v\mathbf{d}),(\mathbf{e},\mathbf{f}+v\mathbf{g})\,\rangle = (\mathbf{a}\cdot\mathbf{e},\langle\, \mathbf{b}+v\mathbf{d},\ \mathbf{f}+v\mathbf{g}\,\rangle)$$

where $\mathbf{a},\mathbf{b},\mathbf{d},\mathbf{e},\mathbf{f},\mathbf{g} \in \mathbb{F}_2^n$.

**Proposition 2.3.5.** The EIP on $\mathcal{R}_2^n$ is $\mathcal{R}_2$-linear.

**Proof:** Let $\mathbf{x} = (\mathbf{a},\mathbf{b}+v\mathbf{d})$, $\mathbf{y} = (\mathbf{e},\mathbf{f}+v\mathbf{g})$ and $\mathbf{z} = (\mathbf{h},\mathbf{j}+v\mathbf{k})$ be three elements of $\mathcal{R}_2^n$ and let $r = (r_1,r_2+vr_3)$ and $s = (s_1,s_2+vs_3)$ be two elements of $\mathcal{R}_2$. Then

$$\begin{aligned}\langle r\mathbf{x}+s\mathbf{y},\mathbf{z}\rangle = {} & \langle(r_1,r_2+vr_3)(\mathbf{a},\mathbf{b}+v\mathbf{d})+(s_1,s_2+vs_3)(\mathbf{e},\mathbf{f}+v\mathbf{g}),(\mathbf{h},\mathbf{j}+v\mathbf{k})\rangle \\ = {} & \langle(r_1\mathbf{a}+s_1\mathbf{e},r_2\mathbf{b}+s_2\mathbf{f}+v(r_2\mathbf{d}+r_3\mathbf{b}+r_3\mathbf{d}+s_2\mathbf{g}+s_3\mathbf{f}+s_3\mathbf{g})),(\mathbf{h},\mathbf{j}+v\mathbf{k})\rangle \\ = {} & (r_1\mathbf{ah}+s_1\mathbf{eh},r_2\mathbf{bj}+s_2\mathbf{fj}+v(r_2\mathbf{bk}+s_2\mathbf{fk}+r_2\mathbf{dj}+r_3\mathbf{bj}+r_3\mathbf{dj}+s_2\mathbf{gj} \\ & +s_3\mathbf{fj}+s_3\mathbf{gj}+r_2\mathbf{dk}+r_3\mathbf{bk}+r_3\mathbf{dk}+s_2\mathbf{gk}+s_3\mathbf{fk}+s_3\mathbf{gk}))\end{aligned}$$

$$\begin{aligned} = {} & \sum_{i=1}^{n}(r_1a_ih_i+s_1e_ih_i, r_2b_ij_i+s_2f_ij_i+v(r_2b_ik_i+s_2f_ik_i+r_2d_ij_i+r_3b_ij_i \\ & +r_3d_ij_i+s_2g_ij_i+s_3f_ij_i+s_3g_ij_i+r_2d_ik_i+r_3b_ik_i+r_3d_ik_i+s_2g_ik_i \\ & +s_3f_ik_i+s_3g_ik_i)) \end{aligned}$$

and

$$\begin{aligned} r\langle \mathbf{x},\mathbf{z}\rangle+s\langle \mathbf{y},\mathbf{z}\rangle = {} & (r_1,r_2+vr_3)\langle(\mathbf{a},\mathbf{b}+v\mathbf{d}),(\mathbf{h},\mathbf{j}+v\mathbf{k})\rangle \\ & +(s_1,s_2+vs_3)\langle(\mathbf{e},\mathbf{f}+v\mathbf{g}),(\mathbf{h},\mathbf{j}+v\mathbf{k})\rangle \\ = {} & (r_1,r_2+vr_3)\sum_{i=1}^{n}(a_ih_i,b_ij_i+v(b_ik_i+d_ij_i+d_ik_i)) \\ & +(s_1,s_2+vs_3)\sum_{i=1}^{n}(e_ih_i,f_ij_i+v(f_ik_i+g_ij_i+g_ik_i)) \\ = {} & \sum_{i=1}^{n}(r_1a_ih_i,r_2b_ij_i \\ & +v(r_2b_ik_i+r_2d_ij_i+r_2d_ik_i+r_3b_ij_i+r_3b_ik_i+r_3d_ij_i+r_3d_ik_i)) \\ & +\sum_{i=1}^{n}(s_1e_ih_i,s_2f_ij_i \\ & +v(s_2f_ik_i+s_2g_ij_i+s_2g_ik_i+s_3f_ij_i+s_3f_ik_i+s_3g_ij_i+s_3g_ik_i)) \\ = {} & \sum_{i=1}^{n}(r_1a_ih_i+s_1e_ih_i,r_2b_ij_i+s_2f_ij_i \\ & +v(r_2b_ik_i+r_2d_ij_i+r_2d_ik_i+r_3b_ij_i+r_3b_ik_i+r_3d_ij_i+r_3d_ik_i \\ & +s_2f_ik_i+s_2g_ij_i+s_2g_ik_i+s_3f_ij_i+s_3f_ik_i+s_3g_ij_i+s_3g_ik_i)). \end{aligned}$$

Therefore

$$\langle r\mathbf{x}+s\mathbf{y},\mathbf{z}\rangle = r\langle \mathbf{x},\mathbf{z}\rangle+s\langle \mathbf{y},\mathbf{z}\rangle.$$

■

**Definition 2.3.2.** Let $C$ be a linear code over $\mathcal{R}_2$ of length $n$. Then we define the dual code $C^{\perp}$ of $C$ with respect to the EIP as

$$C^{\perp} := \{\mathbf{y}\in\mathcal{R}_2^n \mid \langle \mathbf{x},\mathbf{y}\rangle=(0,0) \text{ for all } \mathbf{x}\in C\}.$$

**Example 2.3.4.** Let $C$ be the linear code over $\mathcal{R}_2$ given in Example 2.3.1. Then

$$C^{\perp} = \{\, (0,0)(0,0),\, (0,v)(0,v),\, (0,v)(0,1),\, (0,0)(0,1+v)\,\}.$$

**Definition 2.3.3.** A code $C$ is *Euclidean self-orthogonal* if $C \subseteq C^{\perp}$ and *Euclidean self-dual* if $C = C^{\perp}$.

**Corollary 2.3.1.** Let us assume that $C = (C_1, C_2)$ is a linear code over $\mathcal{R}_2$ of length $n$. $C$ is Euclidean self-dual over $\mathcal{R}_2$ if and only if $C_1$ a is binary self-dual code of length $n$ and $C_2$ is a Euclidean self-dual code over $\mathcal{R}_1$ of length $n$.

**Proposition 2.3.6.** Euclidean self-dual codes over $\mathcal{R}_2^n$ exist for even lengths.

**Proof:** By Corollary 2.3.1, $n$ must be even, since binary self-dual codes exist for even lengths. ■

We now prove some of the properties of the Gray map $\phi_1$.

**Theorem 2.3.1.** The Gray map $\phi_1$ is an orthogonality preserving map based on the EIP.

**Proof:** Let $\mathbf{x} = (\mathbf{a}, \mathbf{b} + v\mathbf{d})$ and $\mathbf{y} = (\mathbf{e}, \mathbf{f} + v\mathbf{g})$ be two elements of $\mathcal{R}_2^n$. Using the EIP on $\mathcal{R}_2^n$, we see that

$$\begin{aligned}
\langle \mathbf{x}, \mathbf{y} \rangle &= (\mathbf{ae}, \mathbf{bf} + v(\mathbf{bg} + \mathbf{df} + \mathbf{dg})) \\
&= \left( \sum_{i=1}^{n} a_i e_i, \sum_{i=1}^{n} b_i f_i + v \left( \sum_{i=1}^{n} b_i g_i + \sum_{i=1}^{n} d_i f_i + \sum_{i=1}^{n} d_i g_i \right) \right) \\
&= (0,0)
\end{aligned}$$

if and only if

$$\sum_{i=1}^{n} a_i e_i = 0, \ \sum_{i=1}^{n} b_i f_i = 0 \ \text{ and } \ \sum_{i=1}^{n} (b_i g_i + d_i f_i + d_i g_i) = 0.$$

Therefore

$$\begin{aligned}\langle \phi_1(\mathbf{x}), \phi_1(\mathbf{y})\rangle &= (\mathbf{a},\mathbf{b},\mathbf{b}+\mathbf{d})\cdot(\mathbf{e},\mathbf{f},\mathbf{f}+\mathbf{g})\\ &= \sum_{i=1}^{n} a_ie_i + \sum_{i=1}^{n} b_if_i + \sum_{i=1}^{n}(b_if_i+b_ig_i+d_if_i+d_ig_i)\\ &= \sum_{i=1}^{n}(a_ie_i+b_ig_i+d_if_i+d_ig_i)\\ &= 0.\end{aligned}$$

■

**Corollary 2.3.2.** Let $C^{\perp}$ be the dual code of $C$. Then $\phi_1(C^{\perp}) = (\phi_1(C))^{\perp}$, where $(\phi_1(C))^{\perp}$ is the dual of $\phi_1(C)$ as a binary code. Moreover, if $C$ is a Euclidean self-dual code, so is the Gray image $\phi_1(C)$.

**Proof:** For any element $\mathbf{c} = (\mathbf{a}, \mathbf{b}+v\mathbf{d})$ of $C$ and any element $\mathbf{c}' = (\mathbf{a}', \mathbf{b}'+v\mathbf{d}')$ of $C^{\perp}$, we obtain

$$\langle \mathbf{c}, \mathbf{c}'\rangle = (0,0),$$

where $\mathbf{a}$, $\mathbf{b}$, $\mathbf{d}$, $\mathbf{a}'$, $\mathbf{b}'$, $\mathbf{d}' \in \mathbb{F}_2^n$. Then we have

$$\mathbf{aa}' = 0,\ \mathbf{bb}' = 0 \text{ and } \mathbf{bd}'+\mathbf{db}'+\mathbf{dd}' = 0.$$

$$\begin{aligned}\langle \phi_1(\mathbf{c}), \phi_1(\mathbf{c}')\rangle &= (\mathbf{a},\mathbf{b},\mathbf{b}+\mathbf{d})(\mathbf{a}',\mathbf{b}',\mathbf{b}'+\mathbf{d}')\\ &= \mathbf{aa}'+\mathbf{bb}'+\mathbf{bb}'+\mathbf{bd}'+\mathbf{db}'+\mathbf{dd}'\\ &= 0\end{aligned}$$

which means $\phi_1(C^{\perp}) \subseteq (\phi_1(C))^{\perp}$.

If $C$ is a code of length $n$ with $s$ elements, then by Theorem 2.1.2,

$$|C^{\perp}| = \frac{|\mathcal{R}_2|}{|C|} = \frac{|\mathcal{R}_2|^n}{s} = \frac{8^n}{s}.$$

Since $\phi_1$ is one-to-one, we have

$$|C| = |\phi_1(C)| = s \text{ and } |C^{\perp}| = |\phi_1(C^{\perp})| = \frac{8^n}{s}.$$

Once again by Theorem 2.1.2,

$$|(\phi_1(C))^{\perp}| = \frac{\mathbb{F}_2^{3n}}{|\phi_1(C)|} = \frac{2^{3n}}{|\phi_1(C)|} = \frac{8^n}{s} = |\phi_1(C^{\perp})|.$$

Therefore,

$$\phi_1(C^{\perp}) = (\phi_1(C))^{\perp}.$$

■

Now let us give the necessary and sufficient conditions to obtain a Euclidean self-dual code with the generator matrix $G = [(I_k, I_k)|(A, B + vD)]$.

**Theorem 2.3.2.** The matrix $G = [(I_k, I_k)|(A, B + vD)]$ generates a Euclidean self-dual code over $\mathcal{R}_2$ of length $2k$ if and only if $A$ and $B$ are orthogonal matrices and $BD^T + DB^T + DD^T = 0$, where $A$, $B$ and $D$ are $k \times k$ binary matrices.

**Proof:** Let $G$ be the generator matrix of a Euclidean self-dual code. Then

$$G \cdot G^T = 0.$$

This means

$$[(I_k, I_k)|(A, B+vD)] \cdot [(I_k, I_k)|(A, B+vD)]^T = [(I_k, I_k)|(A, B+vD)] \cdot \begin{bmatrix} (I_k, I_k) \\ (A^T, B^T + vD^T) \end{bmatrix}$$

$$= (I_k, I_k) + (AA^T, BB^T + v(BD^T + DB^T + DD^T))$$

$$= (I_k + AA^T, I_k + BB^T + v(BD^T + DB^T + DD^T))$$

$$= (0, 0).$$

Hence $AA^T = I_k$, $BB^T = I_k$ and $BD^T + DB^T + DD^T = 0$.

Conversely, assume that

$$AA^T = I_k, \quad BB^T = I_k \quad \text{and} \quad BD^T + DB^T + DD^T = 0.$$

Then

$$G\cdot G^{T}=0$$

which implies that $G$ generates a Euclidean self-orthogonal code. Since the number of the rows of $G$ is the half of the number of the columns of $G$, $G$ generates a Euclidean self-dual code. ■

**Proposition 2.3.7.** Let $G=[(I_k,I_k)|(A,B+vD)]$ generate a Euclidean self-dual code $C$ over $\mathcal{R}_2$ of length $2k$. Then the Gray image of $C$ is generated by

$$G'=\begin{bmatrix} I_k & 0 & 0 & A & 0 & 0\\ 0 & I_k & 0 & 0 & B & 0\\ 0 & 0 & I_k & 0 & 0 & B+D\end{bmatrix}.$$

**Proof:** Recall that the minimal ideals of $\mathcal{R}_2$ are generated by $(1,0)$, $(0,1+v)$ and $(0,v)$. In order to obtain the Gray image matrix $G'$, we first multiply the generator matrix $G$ with the elements $(1,0)$, $(0,1+v)$ and $(0,v)$, then we take the Gray image of these matrices and gather up as a matrix as shown in the following.

$$\begin{aligned}\phi_1((1,0)G)=\phi_1((1,0)[(I_k,I_k)|(A,B+vD)]) &= \phi_1([(I_k,0)|(A,0)])\\ &= [I_k,0,0,A,0,0],\end{aligned}$$

$$\begin{aligned}\phi_1((0,1+v)G)=\phi_1((0,1+v)[(I_k,I_k)|(A,B+vD)]) &= \phi_1([(0,I_k+vI_k)|(0,B+vB)])\\ &= [0,I_k,0,0,B,0],\end{aligned}$$

and

$$\begin{aligned}\phi_1((0,v)G)=\phi_1((0,v)[(I_k,I_k)|(A,B+vD)]) &= \phi_1([(0,vI_k)|(0,vB+vD)])\\ &= [0,0,I_k,0,0,B+D].\end{aligned}$$

By writing these matrices as rows of $G'$, we obtain

$$G'=\begin{bmatrix}\phi_1((1,0)G)\\ \phi_1((0,1+v)G)\\ \phi_1((0,v)G)\end{bmatrix}=\begin{bmatrix} I_k & 0 & 0 & A & 0 & 0\\ 0 & I_k & 0 & 0 & B & 0\\ 0 & 0 & I_k & 0 & 0 & B+D\end{bmatrix}$$

which generates the Gray image of $C$. ■

**Proposition 2.3.8.** Assume that $C = (C_1, C_2)$ is a linear code over $\mathcal{R}_2$. Then the minimum Lee weight of the code $C = (C_1, C_2)$ is determined by

$$d_L(C) = \min \{d_H(C_1), d_L(C_2)\},$$

where $d_H(C_1)$ and $d_L(C_2)$ denote the minimum Hamming weight of the binary code $C_1$ and the minimum Lee weight of the code $C_2$ over $\mathcal{R}_1$, respectively.

**Proof:** The proof directly follows from Definition 2.3.1 and Proposition 2.3.7. ■

### 2.3.4. Hermitian Self-Dual Codes over $\mathcal{R}_2$

**Definition 2.3.4.** We define the Gray weight $w_G$ on $\mathcal{R}_2$ as

$$w_G(x) = \begin{cases} 0 & \text{if } x = (0,0), \\ 1 & \text{if } x = (1,1),(1,v),(1,1+v), \\ 2 & \text{if } x = (0,1),(0,v),(0,1+v), \\ 3 & \text{if } x = (1,0). \end{cases}$$

The Gray weight of a word $\mathbf{x} = (x_1, \ldots, x_n) \in \mathcal{R}_2^n$ is the rational sum of the Gray weight of its components, i.e., $w_G(\mathbf{x}) = \sum\limits_{i=1}^{n} w_G(x_i)$. For any $\mathbf{x}, \mathbf{y} \in \mathcal{R}_2^n$, the Gray distance between two elements is given as $d_G(\mathbf{x}, \mathbf{y}) = w_G(\mathbf{x} - \mathbf{y})$. The Gray weight of a code $C$ is the smallest non-zero Gray weight among all codewords of $C$. The Gray distance of $C$ is the smallest non-zero Gray distance between all pairs of distinct codewords of $C$.

We define the second Gray map $\boldsymbol{\phi_2}$ as

$$\boldsymbol{\phi_2} : \mathcal{R}_2 \to \mathbb{F}_2^3, \ \ (a, b+vd) \mapsto (a+b, a+d, a+b+d).$$

This Gray map is extended to $\mathcal{R}_2^n$ as

$$\phi_2((\mathbf{a}, \mathbf{b} + v\mathbf{d})) = (\mathbf{a}+\mathbf{b}, \mathbf{a}+\mathbf{d}, \mathbf{a}+\mathbf{b}+\mathbf{d}),$$

where $\mathbf{a}, \mathbf{b}, \mathbf{d} \in \mathbb{F}_2^n$.

**Theorem 2.3.3.** The Gray map $\phi_2$ is $\mathbb{F}_2$-linear.

**Proof:** Let $\mathbf{x} = (\mathbf{a}, \mathbf{b} + v\mathbf{d})$ and $\mathbf{y} = (\mathbf{e}, \mathbf{f} + v\mathbf{g})$ be two elements of $\mathcal{R}_2^n$ and $r, s \in \mathbb{F}_2$. Then $r\mathbf{x} + s\mathbf{y} = (r\mathbf{a} + s\mathbf{e}, r\mathbf{b} + s\mathbf{f} + v(r\mathbf{d} + s\mathbf{g}))$. Therefore,

$$\begin{aligned}
\phi_2(r\mathbf{x} + s\mathbf{y}) &= \phi_2((r\mathbf{a} + s\mathbf{e}, r\mathbf{b} + s\mathbf{f} + v(r\mathbf{d} + s\mathbf{g}))) \\
&= (r\mathbf{a} + s\mathbf{e} + r\mathbf{b} + s\mathbf{f}, r\mathbf{a} + s\mathbf{e} + r\mathbf{d} + s\mathbf{g}, r\mathbf{a} + s\mathbf{e} + r\mathbf{b} + s\mathbf{f} + r\mathbf{d} + s\mathbf{g}) \\
&= (r\mathbf{a} + r\mathbf{b}, r\mathbf{a} + r\mathbf{d}, r\mathbf{a} + r\mathbf{b} + r\mathbf{d}) + (s\mathbf{e} + s\mathbf{f}, s\mathbf{e} + s\mathbf{g}, s\mathbf{e} + s\mathbf{f} + s\mathbf{g}) \\
&= r(\mathbf{a} + \mathbf{b}, \mathbf{a} + \mathbf{d}, \mathbf{a} + \mathbf{b} + \mathbf{d}) + s(\mathbf{e} + \mathbf{f}, \mathbf{e} + \mathbf{g}, \mathbf{e} + \mathbf{f} + \mathbf{g}) \\
&= r\phi_1(\mathbf{x}) + s\phi_1(\mathbf{y}).
\end{aligned}$$

■

**Proposition 2.3.9.** If $C$ is a linear code over $\mathcal{R}_2$ of length $n$, then $\phi_2(C)$ is a binary linear code of length $3n$.

**Example 2.3.5.** Let $C$ be the linear code over $\mathcal{R}_2$ given in Example 2.3.1. Then

$$\begin{aligned}
\phi_2(C) = \{\, &000000, 001111, 010101, 011010, 101000, 111111, 010111, 100111, \\
&110000, 011000, 111101, 101010, 000010, 110010, 100101, 001101 \,\}
\end{aligned}$$

is a binary linear code of length 6.

**Proposition 2.3.10.** The Gray map $\phi_2$ is a distance preserving map from $\mathcal{R}_2^n$ (Gray distance-$d_G$) to $\mathbb{F}_2^{3n}$ (Hamming distance-$d_H$).

Now, we define the HIP on $\mathcal{R}_2^n$ similar to the natural HIP as

$$[\mathbf{x}, \mathbf{y}] = \sum_{i=1}^{n} x_i \overline{y_i},$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_n), \mathbf{y} = (y_1, y_2, \ldots, y_n) \in \mathcal{R}_2^n$. In general,

$$\overline{y_i} = \overline{(e_i, f_i + vg_i)} = (e_i, f_i + g_i + vg_i)$$

for all $1 \leq i \leq n$. Also we can define the HIP with regard to the ordinary dot product in $\mathbb{F}_2$

and the HIP in $\mathcal{R}_1$ as follows:

$$\begin{aligned}[(\mathbf{a},\mathbf{b}+v\mathbf{d}),(\mathbf{e},\mathbf{f}+v\mathbf{g})] &= (\mathbf{a}\cdot\mathbf{e},[\,\mathbf{b}+v\mathbf{d},\ \mathbf{f}+v\mathbf{g}\,])\\ &= (\mathbf{a}\cdot\mathbf{e},\langle\,\mathbf{b}+v\mathbf{d},\ \overline{\mathbf{f}+v\mathbf{g}}\,\rangle),\end{aligned}$$

where $\overline{\mathbf{f}+v\mathbf{g}} = (\mathbf{f}+\mathbf{g}+v\mathbf{g})$ and $\mathbf{a},\mathbf{b},\mathbf{d},\mathbf{e},\mathbf{f},\mathbf{g}\in\mathbb{F}_2^n$.

**Proposition 2.3.11.** The HIP on $\mathcal{R}_2^n$ is $\mathcal{R}_2$-linear.

**Proof:** Let $\mathbf{x}=(\mathbf{a},\mathbf{b}+v\mathbf{d})$, $\mathbf{y}=(\mathbf{e},\mathbf{f}+v\mathbf{g})$ and $\mathbf{z}=(\mathbf{h},\mathbf{j}+v\mathbf{k})$ be three elements of $\mathcal{R}_2^n$ and let $r=(r_1,r_2+vr_3)$ and $s=(s_1,s_2+vs_3)$ be two elements of $\mathcal{R}_2$.

$$\begin{aligned}[r\mathbf{x}+s\mathbf{y},\mathbf{z}] = {}& [(r_1,r_2+vr_3)(\mathbf{a},\mathbf{b}+v\mathbf{d})+(s_1,s_2+vs_3)(\mathbf{e},\mathbf{f}+v\mathbf{g}),(\mathbf{h},\mathbf{j}+v\mathbf{k})]\\ = {}& [(r_1\mathbf{a}+s_1\mathbf{e},r_2\mathbf{b}+s_2\mathbf{f}+v(r_2\mathbf{d}+r_3\mathbf{b}+r_3\mathbf{d}+s_2\mathbf{g}+s_3\mathbf{f}+s_3\mathbf{g})),(\mathbf{h},\mathbf{j}+v\mathbf{k})]\\ = {}& (r_1\mathbf{ah}+s_1\mathbf{eh},r_2\mathbf{bj}+r_2\mathbf{bk}+s_2\mathbf{fj}+s_2\mathbf{fk}+v(r_2\mathbf{bk}+s_2\mathbf{fk}+r_2\mathbf{dj}+r_3\mathbf{bj}\\ & +r_3\mathbf{dj}+s_2\mathbf{gj}+s_3\mathbf{fj}+s_3\mathbf{gj}))\\ = {}& \sum_{i=1}^{n}(r_1a_ih_i+s_1e_ih_i,r_2b_ij_i+r_2b_ik_i+s_2f_ij_i+s_2f_ik_i+v(r_2b_ik_i+s_2f_ik_i\\ & +r_2d_ij_i+r_3b_ij_i+r_3d_ij_i+s_2g_ij_i+s_3f_ij_i+s_3g_ij_i+s_3f_ik_i+s_3g_ik_i))\end{aligned}$$

and

$$\begin{aligned}r[\mathbf{x},\mathbf{z}]+s[\mathbf{y},\mathbf{z}] = {}& (r_1,r_2+vr_3)[(\mathbf{a},\mathbf{b}+v\mathbf{d}),(\mathbf{h},\mathbf{j}+v\mathbf{k})]\\ & +(s_1,s_2+vs_3)[(\mathbf{e},\mathbf{f}+v\mathbf{g}),(\mathbf{h},\mathbf{j}+v\mathbf{k})]\\ = {}& (r_1,r_2+vr_3)\left(\sum_{i=1}^{n}(a_ih_i,b_ij_i+b_ik_i+v(b_ik_i+d_ij_i))\right)\\ & +(s_1,s_2+vs_3)\left(\sum_{i=1}^{n}(e_ih_i,f_ij_i+f_ik_i+v(f_ik_i+g_ij_i))\right)\\ = {}& \sum_{i=1}^{n}(r_1a_ih_i,r_2b_ij_i+r_2b_ik_i+v(r_2b_ik_i+r_2d_ij_i+r_3b_ij_i+r_3d_ij_i)\\ & +\sum_{i=1}^{n}(s_1e_ih_i,s_2f_ij_i+s_2f_ik_i+v(s_2f_ik_i+s_2g_ij_i+s_3f_ij_i+s_3g_ij_i))\end{aligned}$$

$$\begin{aligned} &= \sum_{i=1}^{n} (r_1 a_i h_i + s_1 e_i h_i, r_2 b_i j_i + r_2 b_i k_i + s_2 f_i j_i + s_2 f_i k_i \\ &\quad + v(r_2 b_i k_i + r_2 d_i j_i + r_3 b_i j_i + r_3 d_i j_i + s_2 f_i k_i + s_2 g_i j_i + s_3 f_i j_i + s_3 g_i j_i)). \end{aligned}$$

Therefore

$$[r\mathbf{x} + s\mathbf{y}, \mathbf{z}] = r[\mathbf{x}, \mathbf{z}] + s[\mathbf{y}, \mathbf{z}].$$

■

**Definition 2.3.5.** Let $C$ be a linear code over $\mathcal{R}_2$ of length $n$. Then the dual code $C^*$ of $C$ with respect to the HIP is defined as

$$C^* := \{\mathbf{y} \in \mathcal{R}_2^n | \ [\mathbf{x}, \mathbf{y}] = (0,0) \text{ for all } \mathbf{x} \in C\}.$$

A code $C$ is *Hermitian self-orthogonal* if $C \subseteq C^*$ and *Hermitian self-dual* if $C = C^*$.

**Example 2.3.6.** Let $C$ be the linear code over $\mathcal{R}_2$ given in Example 2.3.1. Then

$$C^* = \{\ (0,0)(0,0),\ (0,1+v)(0,1+v),\ (0,0)(0,v),\ (0,1+v)(0,1)\}.$$

**Theorem 2.3.4.** The Gray map $\phi_2$ is an orthogonality preserving map based on the HIP.

**Proof:** Let $\mathbf{x} = (\mathbf{a}, \mathbf{b} + v\mathbf{d})$ and $\mathbf{y} = (\mathbf{e}, \mathbf{f} + v\mathbf{g})$ be two elements of $\mathcal{R}_2^n$. We see that $[\mathbf{x}, \mathbf{y}] = (0,0)$ if and only if

$$\sum_{i=1}^{n} a_i d_i = 0, \ \sum_{i=1}^{n} (b_i f_i + b_i g_i) = 0 \quad \text{and} \quad \sum_{i=1}^{n} (b_i g_i + d_i f_i) = 0.$$

Therefore

$$\begin{aligned} \langle \phi_2(\mathbf{x}), \phi_2(\mathbf{y}) \rangle &= \langle (\mathbf{a}+\mathbf{b}, \mathbf{a}+\mathbf{d}, \mathbf{a}+\mathbf{b}+\mathbf{d}), (\mathbf{e}+\mathbf{f}, \mathbf{e}+\mathbf{g}, \mathbf{e}+\mathbf{f}+\mathbf{g}) \rangle \\ &= \sum_{i=1}^{n} (a_i e_i + a_i f_i + b_i e_i + b_i f_i) + \sum_{i=1}^{n} (a_i e_i + a_i g_i + d_i e_i + d_i g_i) \\ &\quad + \sum_{i=1}^{n} (a_i e_i + a_i f_i + a_i g_i + b_i e_i + b_i f_i + b_i g_i + d_i e_i + d_i f_i + d_i g_i) \\ &= \sum_{i=1}^{n} (a_i e_i + b_i g_i + d_i f_i) = 0. \end{aligned}$$

■

**Corollary 2.3.3.** Assume that $C = (C_1, C_2)$ is a linear code over $\mathcal{R}_2$. $C$ is Hermitian self-dual if and only if $C_1$ is a binary self-dual code and $C_2$ is a Hermitian self-dual code over $\mathcal{R}_1$.

**Corollary 2.3.4.** Let $C^*$ be the dual code of $C$ with respect to the HIP. Then $\phi_2(C^*) = (\phi_2(C))^\perp$, where $(\phi_2(C))^\perp$ is the dual of $\phi_2(C)$ as a binary code. Moreover, if $C$ is a self-dual code with respect to the HIP, so is the Gray image $\phi_2(C)$.

**Proof:** For any $\mathbf{c} = (\mathbf{a}, \mathbf{b} + v\mathbf{d})$ of $C$ and $\mathbf{c}' = (\mathbf{a}', \mathbf{b}' + v\mathbf{d}')$ of $C^*$, we have

$$[\mathbf{c}, \mathbf{c}'] = (0,0),$$

where $\mathbf{a}$, $\mathbf{b}$, $\mathbf{d}$, $\mathbf{a}'$, $\mathbf{b}'$, $\mathbf{d}' \in \mathbb{F}_2^n$. Since we find

$$\mathbf{aa}' = 0,\ \ \mathbf{bb}' + \mathbf{bd}' = 0 \ \text{ and } \ \mathbf{bd}' + \mathbf{db}' = 0,$$

$$\begin{aligned} \langle \phi_2(\mathbf{c}), \phi_2(\mathbf{c}') \rangle &= (\mathbf{a}+\mathbf{b}, \mathbf{a}+\mathbf{d}, \mathbf{a}+\mathbf{b}+\mathbf{d})(\mathbf{a}'+\mathbf{b}', \mathbf{a}'+\mathbf{d}', \mathbf{a}'+\mathbf{b}'+\mathbf{d}') \\ &= \mathbf{aa}' + \mathbf{bd}' + \mathbf{db}' \\ &= 0 \end{aligned}$$

which means $\phi_2(C^*) \subseteq (\phi_2(C))^\perp$.

If $C$ is a code of length $n$ with $s$ elements, then by Theorem 2.1.2, $C^*$ has

$$\frac{|\mathcal{R}_2|^n}{s} = \frac{8^n}{s}$$

elements. Since $\phi_2$ is one-to-one, then we have

$$|C| = |\phi_2(C)| = s \ \text{ and } \ |C^*| = |\phi_2(C^*)| = \frac{8^n}{s}.$$

Once again by Theorem 2.1.2,

$$|(\phi_2(C))^\perp| = \frac{\mathbb{F}_2^{3n}}{|\phi_2(C)|} = \frac{2^{3n}}{|\phi_2(C)|} = \frac{8^n}{s} = |\phi_2(C^*)|.$$

Therefore,

$$\phi_2(C^*) = (\phi_2(C))^{\perp}.$$

■

**Corollary 2.3.5.** A self-dual code over $\mathcal{R}_2$ of length $n$ with respect to the HIP exists if and only if $n$ is even.

**Proof:** Binary self-dual codes of length $n$ exist if and only if $n$ is even. ■

The following theorem gives the necessary and sufficient condition on the matrices that generate Hermitian self-dual codes:

**Theorem 2.3.5.** The matrix $G = [(I_k, I_k)|(A, B+vD)]$ generates a Hermitian self-dual code over $\mathcal{R}_2$ of length $2k$ if and only if $A$ is an orthogonal matrix, $BB^T + BD^T = I_k$ and $BD^T + DB^T = 0$, where $A$, $B$ and $D$ are $k \times k$ binary matrices.

**Proof:** Let $G$ be the generator matrix of a Hermitian self-dual code. Then

$$\begin{aligned}
G \cdot \overline{G}^T &= [(I_k,I_k)|(A,B+vD)] \cdot [(I_k,I_k)|(A,B+D+vD)]^T \\
&= [(I_k,I_k)|(A,B+vD)] \cdot \begin{bmatrix} (I_k,I_k) \\ (A^T,B^T+D^T+vD^T) \end{bmatrix} \\
&= (I_k,I_k) + (AA^T, BB^T+BD^T+v(BD^T+DB^T)) \\
&= (I_k+AA^T, I_k+BB^T+BD^T+v(BD^T+DB^T)) \\
&= (0,0).
\end{aligned}$$

Hence $AA^T = I_k$, $BB^T + BD^T = I_k$ and $BD^T + DB^T = 0$.

Conversely, assume that $AA^T = I_k$, $BB^T + BD^T = I_k$ and $BD^T + DB^T = 0$. Then $G \cdot \overline{G}^T = 0$ which implies $G$ generates a Hermitian self-orthogonal code. Since the number of the rows of $G$ is the half of the number of the columns of $G$, $G$ generates a Hermitian self-dual code. ■

**Example 2.3.7.** Let $A = [1]$, $B = [1]$ and $D = [0]$. Then the conditions $AA^T = I_k$, $BB^T + BD^T = I_k$ and $BD^T + DB^T = 0$ are satisfied. The matrix $G = [(1,1)|(1,1)]$ generates the code

$$C=\{(0,0)(0,0),(0,1)(0,1),(0,v)(0,v),(0,1+v)(0,1+v),(1,0)(1,0),(1,1)(1,1),\\ (1,v)(1,v),(1,1+v)(1,1+v)\}$$

which is a Hermitian self-dual code over $\mathcal{R}_2$ of length 2.

**Proposition 2.3.12.** Let $G=[(I_k,I_k)|(A,B+vD)]$ generate a Hermitian self-dual code $C$ over $\mathcal{R}_2$ of length $2k$. The Gray image of $C$ is generated by

$$G'=\begin{bmatrix} I_k & I_k & I_k & A & A & A \\ I_k & I_k & 0 & B & B & 0 \\ 0 & I_k & I_k & 0 & B+D & B+D \end{bmatrix}.$$

**Proof:** Recall that the minimal ideals of $\mathcal{R}_2$ are generated by $(1,0)$, $(0,1+v)$ and $(0,v)$. In order to obtain the Gray image matrix $G'$, we first multiply the generator matrix $G$ with the elements $(1,0)$, $(0,1+v)$ and $(0,v)$, then we take the Gray image of these matrices and gather up as a matrix as shown in the following.

$$\begin{aligned}\phi_2((1,0)G)=\phi_2((1,0)[(I_k,I_k)|(A,B+vD)]) &= \phi_2([(I_k,0)|(A,0)]) \\ &= [I_k,I_k,I_k,A,A,A],\end{aligned}$$

$$\begin{aligned}\phi_2((0,1+v)G)=\phi_2((0,1+v)[(I_k,I_k)|(A,B+vD)]) &= \phi_2([(0,I_k+vI_k)|(0,B+vB)]) \\ &= [I_k,I_k,0,B,B,0],\end{aligned}$$

and

$$\begin{aligned}\phi_2((0,v)G)=\phi_2((0,v)[(I_k,I_k)|(A,B+vD)]) &= \phi_2([(0,vI_k)|(0,vB+vD)]) \\ &= [0,I_k,I_k,0,B+D,B+D].\end{aligned}$$

By writing these matrices as rows of $G'$, we obtain

$$G'=\begin{bmatrix} \phi_2((1,0)G) \\ \phi_2((0,1+v)G) \\ \phi_2((0,v)G) \end{bmatrix}=\begin{bmatrix} I_k & I_k & I_k & A & A & A \\ I_k & I_k & 0 & B & B & 0 \\ 0 & I_k & I_k & 0 & B+D & B+D \end{bmatrix}$$

which generates the Gray image of $C$. ■

**Proposition 2.3.13.** Let $C=(C_1,C_2)$ be a linear code over $\mathcal{R}_2$. Then the minimum Gray weight of the code $C=(C_1,C_2)$ is bounded by

$$d_G(C)\leq \min\{3d_H(C_1),2d_L(C_2)\}$$

where $d_H(C_1)$ and $d_L(C_2)$ denote the minimum Hamming weight of the codes $C_1$ and the minimum Lee weight of the code $C_2$ over $\mathcal{R}_1$, respectively.

**Proof:** Let $(x,y+vz)$ be a non-zero codeword of the code $C=(C_1,C_2)$, where $x\in C_1$ and $y+vz\in C_2$ are non-zero codewords. Since $C$ is linear, it contains the elements

$$(x,0),\ (0,v(y+z)) \text{ and } (0,y+vy).$$

The Gray images (based on the Gray map $\phi_2$) of these elements are

$$(x,x,x),\ (0,y+z,y+z) \text{ and } (y,y,0),$$

respectively. Therefore, the minimum Gray weight of the code $C=(C_1,C_2)$ is less than or equal to one of the Hamming weight of these images. By Proposition 2.2.2, we write

$$d_L(C_2)\leq \min\{d_H(y),d_H(y+z)\}.$$

Therefore, the minimum Gray weight of $C=(C_1,C_2)$ is bounded by

$$d_G(C)\leq \min\{3d_H(C_1),2d_L(C_2)\}.$$

■

### 2.3.5. The Weight Enumerators and the MacWilliams Identities over $\mathcal{R}_2$

In this section we discuss the weight enumerators of a linear code over $\mathcal{R}_2$. Let $\mathcal{R}_2=\{f_1,f_2,f_3,f_4,f_5,f_6,f_7,f_8\}$ be in fixed order as

$$\mathcal{R}_2=\{(0,0),(0,1),(0,v),(0,1+v),(1,0),(1,1),(1,v),(1,1+v)\}.$$

The CWE of a linear code $C$ over $\mathcal{R}_2$ of length $n$ is defined as

$$cwe_C(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8) = \sum_{\mathbf{c}\in C} \left( x_1^{w_{f_1}(\mathbf{c})} x_2^{w_{f_2}(\mathbf{c})} x_3^{w_{f_3}(\mathbf{c})} x_4^{w_{f_4}(\mathbf{c})} x_5^{w_{f_5}(\mathbf{c})} x_6^{w_{f_6}(\mathbf{c})} x_7^{w_{f_7}(\mathbf{c})} \right. \\ \left. x_8^{w_{f_8}(\mathbf{c})} \right)$$

where $w_{f_i}(\mathbf{c}) = |\{j|c_j = f_i\}|$ for $\mathbf{c} = (c_1,c_2,\ldots,c_n)$, $1 \leq i \leq 8$ and $1 \leq j \leq n$. $cwe_C$ is a homogeneous polynomial with eight indeterminates $x_1$, $x_2$, $x_3$, $x_4$, $x_5$, $x_6$, $x_7$ and $x_8$, and the total degree of each term is $n$.

Since $C$ is a linear code, the term $x_1^n$ appears in $cwe_C(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8)$. We see that $cwe_C(1,1,1,1,1,1,1,1) = |C|$. Permutation equivalent codes have the same CWE.

From (2.2), we construct the matrix $M = (m_{i,j})_{8\times 8}$ defined as $m_{i,j} = \chi(f_i f_j)$, where $f_i, f_j \in \mathcal{R}_2$. Therefore the matrix $M$ is obtained as

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \end{pmatrix}.$$

**Theorem 2.3.6.** Let $C^{\perp}$ be the Euclidean dual code of a linear code $C$ over $\mathcal{R}_2$ of length $n$. Then

$$cwe_{C^{\perp}}(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8) = \frac{1}{|C|} cwe_C \left( M(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8)^T \right),$$

where $(\ )^T$ denotes the transpose.

**Proof:** The matrix $M$ is constructed depending on the character given in (2.2), from Theorem 3.2 in [35], the proof is completed. ■

**Example 2.3.8.** Let $C$ be the linear code over $\mathcal{R}_2$ in Example 2.3.1 and let $C^{\perp}$ be the

Euclidean dual of $C$ given in Example 2.3.4. Then

$$\begin{aligned} cwe_C(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8) = {} & x_1^2+x_3^2+x_5^2+x_7^2+2x_1x_5+2x_3x_7+x_1x_4+x_1x_8 \\ & +x_2x_3+x_2x_7+x_3x_6+x_4x_5+x_5x_8+x_6x_7 \end{aligned}$$

and

$$cwe_{C^\perp}(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8) = x_1^2+x_3^2+x_1x_4+x_2x_3.$$

It can be obtained that

$$cwe_{C^\perp}(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8) = \frac{1}{16} cwe_C\left(M(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8)^T\right).$$

In $\mathcal{R}_2$, by identifying $x_1$ as $a$, $x_3,x_4,x_5$ as $b$, $x_2,x_7,x_8$ as $d$ and $x_6$ as $e$ in the CWE of a linear code $C$ over $\mathcal{R}_2$, we obtain the SWE as in the following:

$$\begin{aligned} swe_C(a,b,d,e) & = cwe_C(a,d,b,b,b,e,d,d) \\ & = \sum_{\mathbf{c}\in C} \left(a^{n_0(\mathbf{c})} b^{n_1(\mathbf{c})} d^{n_2(\mathbf{c})} e^{n_3(\mathbf{c})}\right), \end{aligned}$$

where $n_0(\mathbf{c}) = w_{f_1}(\mathbf{c})$, $n_1(\mathbf{c}) = w_{f_3}(\mathbf{c}) + w_{f_4}(\mathbf{c}) + w_{f_5}(\mathbf{c})$, $n_2(\mathbf{c}) = w_{f_2}(\mathbf{c}) + w_{f_7}(\mathbf{c}) + w_{f_8}(\mathbf{c})$ and $n_3(\mathbf{c}) = w_{f_6}(\mathbf{c})$. $swe_C$ is a homogeneous polynomial of degree $n$ with four indeterminates $a,b,d$ and $e$, that is, $n = n_0(\mathbf{c}) + n_1(\mathbf{c}) + n_2(\mathbf{c}) + n_3(\mathbf{c})$.

**Theorem 2.3.7.** Let $C^\perp$ be the Euclidean dual code of a linear code $C$ over $\mathcal{R}_2$ of length $n$. Then

$$swe_{C^\perp}(a,b,d,e) = \frac{1}{|C|} swe_C(N(a,b,d,e)^T),$$

where

$$N = \begin{pmatrix} 1 & 3 & 3 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{pmatrix}.$$

**Proof:** From the definition of the SWE and Theorem 2.3.6, we obtain that

$$\begin{aligned} swe_{C^{\perp}}(a,b,d,e) = & \ cwe_{C^{\perp}}(a,d,b,b,b,e,d,d) \\ = & \ \tfrac{1}{|C|}\, cwe_C\left(M(a,d,b,b,b,e,d,d)^T\right) \\ = & \ \tfrac{1}{|C|} swe_C(a+3b+3d+e, a+b-d-e, a-b-d+e, \\ & \qquad\qquad\qquad\qquad\qquad\qquad a-3b+3d-e) \\ = & \ \tfrac{1}{|C|}\, swe_C(N(a,b,d,e)^T). \end{aligned}$$

■

**Example 2.3.9.** Let $C$ be the linear code over $\mathcal{R}_2$ in Example 2.3.1 and let $C^{\perp}$ be the Euclidean dual of $C$ given in Example 2.3.4. Then

$$swe_C(a,b,d,e) = a^2+3b^2+2d^2+3ab+4bd+ad+de+be$$

and

$$swe_{C^{\perp}}(a,b,d,e) = a^2+b^2+bd+ab.$$

It can be obtained that

$$swe_{C^{\perp}}(a,b,d,e) = \frac{1}{16}\, swe_C(N(a,b,d,e)^T).$$

The HWE of a linear code $C$ over $\mathcal{R}_2$ is given by

$$W_C(x,y) = \sum_{\mathbf{c}\in C} x^{n-w_H(\mathbf{c})} y^{w_H(\mathbf{c})} = cwe_C(x,y,y,y,y,y,y,y)$$

which is a homogeneous polynomial of degree $n$ with two indeterminates $x$ and $y$.

**Theorem 2.3.8.** Let $C^{\perp}$ be the Euclidean dual code of a linear code $C$ over $\mathcal{R}_2$. Then

$$W_{C^{\perp}}(x,y) = \frac{1}{|C|} W_C(x+7y, x-y).$$

**Proof:** From Theorem 2.3.6, we have

$$\begin{aligned} W_{C^{\perp}}(x,y) &= cwe_{C^{\perp}}(x,y,y,y,y,y,y,y) \\ &= \frac{1}{|C|} cwe_C(M(x,y,y,y,y,y,y,y)^T) \\ &= \frac{1}{|C|} cwe_C(x+7y,x-y,x-y,x-y,x-y,x-y,x-y,x-y) \\ &= \frac{1}{|C|} W_C(x+7y,x-y). \end{aligned}$$

■

**Example 2.3.10.** Let $C$ be the linear code over $\mathcal{R}_2$ in Example 2.3.1. Then

$$W_C(x,y) = x^2 + 11y^2 + 4xy$$

and

$$\begin{aligned} W_{C^{\perp}}(x,y) = &\ \frac{1}{16}((x+7y)^2 + 11(x-y)^2 + 4(x+7y)(x-y)) \\ = &\ x^2 + 2y^2 + xy. \end{aligned}$$

The Lee weight of $\mathbf{c} \in \mathcal{R}_2^n$ can be denoted as

$$w_L(\mathbf{c}) = n_1(\mathbf{c}) + 2n_2(\mathbf{c}) + 3n_3(\mathbf{c}).$$

The LWE of a linear code $C$ over $\mathcal{R}_2$ is defined to be

$$Lee_C(x,y) = W_{\phi_1(C)}(x,y) = \sum_{\mathbf{c} \in C} x^{3n-w_L(\mathbf{c})} y^{w_L(\mathbf{c})}.$$

**Example 2.3.11.** Let $C$ be the linear code over $\mathcal{R}_2$ in Example 2.3.1. Then

$$Lee_C(x,y) = x^6 + 3x^5y + 4x^4y^2 + 4x^3y^3 + 3x^2y^4 + xy^5.$$

**Theorem 2.3.9.** Let $C$ be a linear code over $\mathcal{R}_2$. Then

$$swe_C(x^3, x^2y, xy^2, y^3) = Lee_C(x,y).$$

**Proof:** From the definition of the SWE, we have

$$\begin{aligned} swe_C(x^3,x^2y,xy^2,y^3) = &\sum_{\mathbf{c}\in C}(x^3)^{n_0(\mathbf{c})}(x^2y)^{n_1(\mathbf{c})}(xy^2)^{n_2(\mathbf{c})}(y^3)^{n_3(\mathbf{c})} \\ = &\sum_{\mathbf{c}\in C}x^{3n_0(\mathbf{c})+2n_1(\mathbf{c})+n_2(\mathbf{c})}y^{n_1(\mathbf{c})+2n_2(\mathbf{c})+3n_3(\mathbf{c})} \\ = &\sum_{\mathbf{c}\in C}x^{3n-w_L(\mathbf{c})}y^{w_L(\mathbf{c})} \\ = &\; Lee_C(x,y). \end{aligned}$$

■

**Example 2.3.12.** Let $Lee_C(x,y)$, found in Example 2.3.11, be the LWE of the code $C$ given in Example 2.3.1. Then the SWE of the code $C$, found in Example 2.2.2, can be found depending on four indeterminates $x^3$, $x^2y$, $xy^2$ and $y^3$ as

$$\begin{aligned} swe_C(x^3,x^2y,xy^2,y^3) &= (x^3)^2+3(x^2y)^2+2(xy^2)^2+3x^3x^2y+4x^2yxy^2+x^3xy^2+xy^2y^3 \\ &\quad +x^2yy^3 \\ &= x^6+3x^4y^2+2x^2y^4+3x^5y+4x^3y^3+x^4y^2+xy^5+x^2y^4 \\ &= x^6+3x^5y+4x^4y^2+4x^3y^3+3x^2y^4+xy^5 \end{aligned}$$

which is equal to $Lee_C(x,y)$.

**Theorem 2.3.10.** Let $C^{\perp}$ be the Euclidean dual code of a linear code $C$ over $\mathcal{R}_2$. Then

$$Lee_{C^{\perp}}(x,y)=\frac{1}{|C|}Lee_C(x+y,x-y).$$

**Proof:** By using Theorem 2.3.7 and Theorem 2.3.9, we obtain

$$\begin{aligned} Lee_{C^{\perp}}(x,y) &= swe_{C^{\perp}}(x^3,x^2y,xy^2,y^3) \\ &= \tfrac{1}{|C|}swe_C(N(x^3,x^2y,xy^2,y^3)^T) \\ &= \tfrac{1}{|C|}swe_C((x+y)^3,(x+y)^2(x-y),(x+y)(x-y)^2,(x-y)^3) \\ &= \tfrac{1}{|C|}Lee_C(x+y,x-y). \end{aligned}$$

■

**Remark 2.3.1.** Similar calculations can be made for the Gray weight. If we determine

$n_0(\mathbf{c}) = w_{f_1}(\mathbf{c})$, $n_1(\mathbf{c}) = w_{f_6}(\mathbf{c}) + w_{f_7}(\mathbf{c}) + w_{f_8}(\mathbf{c})$, $n_2(\mathbf{c}) = w_{f_2}(\mathbf{c}) + w_{f_3}(\mathbf{c}) + w_{f_4}(\mathbf{c})$ and $n_3(\mathbf{c}) = w_{f_5}(\mathbf{c})$, then we obtain exactly the same identities that we obtained for the Lee weight.

**Remark 2.3.2.** The CWE of the Hermitian dual of a linear code $C$ can be obtained in a similar way. Actually, if we interchange the coordinates $x_3$-$x_4$ and $x_7$-$x_8$ in the CWE of the Euclidean dual code, we find the following equality

$$cwe_{C^{\perp}}(x_1,x_2,x_4,x_3,x_5,x_6,x_8,x_7) = cwe_{C^{*}}(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8).$$

## 2.4. THE CONSTRUCTION METHODS

### 2.4.1. The Construction Methods for Free Euclidean and Free Hermitian Self-Dual Codes of Even Length over $\mathcal{R}_1$ and $\mathcal{R}_2$

There are many ways to construct self-dual codes. In this subsection, we recall some construction methods which are much used to obtain special types of linear codes in coding theory. These are DCC, BDCC and SC methods.

In [29], Huffman and Pless left an exercise requesting to prove the following, which is proved in [39] by Karadeniz et al. for the codes over the ring $R_k$:

**Theorem 2.4.1.** Let A be a $k \times k$ matrix and let $I_k$ be the $k \times k$ identity matrix, then the following assertions hold:
$(a)$ A code with DCC, i.e., a code with generating matrix $[I_k|A]$, where $A$ is a circulant matrix, is isodual.
$(b)$ A code with BDCC, i.e., a code with generating matrix

$$G = \left[\begin{array}{c|c|c} & \alpha & \beta \quad . \quad . \quad . \quad \beta \\ \cline{2-3} & \gamma & \\ I_{k+1} & . & A \\ & . & \\ & . & \\ & \gamma & \end{array}\right],$$

where $A$ is circulant, is isodual provided that $\beta = \gamma = 0$ or both $\beta$ and $\gamma$ are non-zero.
$(c)$ A code with SC, i.e., a code with generating matrix $[I_k|A]$, where $A$ is a symmetric matrix, is isodual.

In this thesis we use DCC, BDCC and SC methods which provide to obtain codes of length $2k$.

### 2.4.2. A New Construction Method for Hermitian Self-Dual Codes of Odd Length over $\mathcal{R}_1$

Euclidean self-dual codes over $\mathcal{R}_1$ exist only for even lengths while Hermitian self-dual codes over $\mathcal{R}_1$ exist for all lengths. In this subsection, we present a new shortening method which makes possible to obtain Hermitian self-dual codes of odd length derived from Hermitian self-dual codes of even length.

**Theorem 2.4.2.** Suppose $G = [I_k|A+vB]$ generates a Hermitian self-dual code over $\mathcal{R}_1$ of length $2k$. The matrix that is obtained by multiplying the $i^{th}$ row of $G$ by a non-zero non-unit and puncturing the $i^{th}$ column of $G$ generates a Hermitian self-dual code of length $2k-1$.

**Proof:** Let $G'$ be the matrix obtained by multiplying the $i^{th}$ row of $G$ by $v$ (or $1+v$) and puncturing the $i^{th}$ column. We need to show that the code $C'$ generated by $G'$ has the size $2.4^{k-1}$. The rows of $G'$ are orthogonal to each other. It is clear that the rows of $G'$ are linearly independent. Since the $i^{th}$ row is non-free, the size of $C'$ is $2.4^{k-1}$.

Denote the $s^{th}$ row of $G$ and $G'$ by $R_s$ and $R'_s$, respectively. Let $R_j$ and $R'_j$ represent any rows of $G$ and $G'$ except the $i^{th}$ rows, respectively. Let $R'_i$ correspond to the $i^{th}$ row of $G'$ and $R''_i$ correspond to the $i^{th}$ row of the matrix obtained by puncturing the $i^{th}$ column of $G$. Since all rows of $G$ are orthogonal and the $i^{th}$ coordinate of each row is 0 except the $i^{th}$ row,

$$\langle R_j, R_s\rangle = 0 \text{ implies } \langle R'_j, R'_s\rangle = 0$$

for $1 \leq j, s \leq k$, $j \neq i$, $s \neq i$. Also $R'_i = vR''_i$ (or $R'_i = (1+v)R''_i$), where $R''_i \in \mathbb{F}_2^{2k-1}$. So

$$\langle R'_i, R'_i \rangle = v(1+v)\langle R''_i, R''_i\rangle = 0,$$

which completes the proof. ■

From now on, we will call the code $C$ as parent, and the code obtained by shortening $C$ as child.

**Proposition 2.4.1.** Let $C' = CRT(C_1', C_1'^{\perp})$ be the child code of $C$ generated by shortening $G$ by $v$ in the $i^{th}$ position. Then the generator matrix $G_1'$ of $C_1'$ is obtained by deleting the $i^{th}$ row and the $i^{th}$ column of $G_1 = [I, A]$ and $G_1'^{\perp}$ is the punctured matrix of $G_2 = [I, A+B]$ in the $i^{th}$ column.

**Proof:** The proof follows from the fact that the generator matrix of $\phi(C') = C_1' \times C_1'^{\perp}$ is

$$\begin{bmatrix} \phi((1+v)G') \\ \phi(vG') \end{bmatrix}.$$

■

**Corollary 2.4.1.** *If the minimum distance $d(C)$ of $C$ is $d$, then $d(C') = d$ or $d-1$.*

**Proof:** Let $C = CRT(C_1, C_1^{\perp})$ be a code whose minimum distance is $d$. By Proposition 2.2.2, $d(C_1) = d$ and $d(C_1^{\perp}) \geq d$ or vice versa. Since only one coordinate of the codes is punctured, $d(C_1') = d(C_1)$ or $d(C_1) - 1$ and $d(C_1'^{\perp}) = d(C_1^{\perp})$ or $d(C_1^{\perp}) - 1$. Therefore $d(C') = d$ or $d-1$. ■

Even though one coordinate of the parent code is punctured, the minimum distance of the child code can still be $d$. We assume that shortening by $v$ is applied to $G$. Let $T$ be the matrix whose rows are the codewords of $C_1^{\perp}$ with minimum weight $d$. Let $Y$ be the matrix whose rows are the codewords with minimum weight $d$ of the code generated by the matrix that is obtained by deleting the $i^{th}$ row of $G_1 = [I_k, A]$. If the $i^{th}$ columns of $T$ and $Y$ are zero, then puncturing these columns does not change the minimum distance of the child code. Therefore, the minimum distance of the child code $d(C')$ is $d$. If at least one of $T$ and $Y$ has a non-zero column in the $i^{th}$ position, then the minimum distance of the child code $d(C')$ is $d-1$. A similar observation can be made for shortening by $1+v$.

The following algorithm determines the case when the minimum weights of the parent and the child code are the same. In this algorithm we take $v$ as a non-zero non-unit.

**The algorithm to obtain the minimum weight of a child code**

```
**Input**: Hermitian self-dual code $C = CRT(C_1, C_1^{\perp})$ with minimum distance $d$
    Generate the $k \times 2k$ matrix $T$ whose rows are the codewords of $C_1^{\perp}$ with minimum weight $d$
        If no 0 column occurs in the first $k$ positions, then
            $d(C') = d - 1$ and terminate
        Else if the $i^{th}$ column is zero, then
            Delete the $i^{th}$ row of the generator matrix of $C_1$ and denote this matrix as $E$
            Generate the $j \times 2k$ matrix $Y$ whose rows are the codewords with minimum weight $d$ generated by $E$
                If the $i^{th}$ column is not zero, then
                    $d(C') = d - 1$ and terminate
**Output**: Shortened Hermitian self-dual code of length $2k - 1$ with minimum distance $d$
```

The following is an illustration of the algorithm on a specific example.

**Example 2.4.1.** For BDCC of a code length 12, we take

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & v & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1+v & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1+v & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1+v \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1+v & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1+v & 0 & 1 & 1 \end{bmatrix},$$

which generates a free Hermitian self-dual code of length 12 whose minimum distance is 4. This generator matrix is of the form $[I_6, A + vB]$. By multiplying the first row of $G$ by $v$ and puncturing the first column of $G$ we obtain

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & v & v & v & v & v & v \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1+v & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1+v & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1+v \\ 0 & 0 & 0 & 1 & 0 & 1 & 1+v & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1+v & 0 & 1 & 1 \end{bmatrix},$$

which generates a Hermitian self-dual code of length 11 whose minimum distance is also 4. According to the algorithm, if we list the codewords whose minimum weight is 4 (these codewords are generated by $[I_6, A+B]$), we get

$$T = \begin{bmatrix} 010000110001 \\ 011000001001 \\ 000010100110 \\ 000001100011 \\ 001100010100 \\ 000100101100 \\ 000011000101 \\ 001000111000 \\ 000110001010 \\ 010001010010 \end{bmatrix}.$$

Since the first column of the listed codewords is zero, we delete the first row of $[I_6, A]$ and list the codewords whose minimum weight is 4 (these codewords are generated by row shortened $[I_6, A]$), we get

$$Y = \begin{bmatrix} 001010001100 \\ 010100011000 \\ 010010000011 \\ 001001010001 \\ 000101000110 \end{bmatrix}.$$

Since the first column of the listed codewords is zero, puncturing the first column does not change the minimum weight of the codewords. Therefore, the minimum distance of the child code remains unchanged.

# 3. RESULTS

We use DCC, BDCC and SC methods to obtain free Euclidean and free Hermitian self-dual codes of even length satisfying the conditions given in Theorem 2.2.1, Theorem 2.2.2, Theorem 2.3.2 and Theorem 2.3.5. If we find good binary isodual codes, we can obtain good free Euclidean and free Hermitian self-dual codes over $\mathcal{R}_1$ and $\mathcal{R}_2$ via the Gray maps. By a good free code over the rings, we mean a free code which has the highest minimum Lee (or Gray) weight. Then we apply the new shortening method for Hermitian self-dual codes over $\mathcal{R}_1$, which we introduce in Subsection 2.4.

We use the Magma software package ([26]) to hunt up the codes obtained by the above methods. The Magma algorithm used to obtain the Euclidean self-dual codes of length 4 over $\mathcal{R}_1$ by using DCC method is given as a representative manner in Appendix 1. The symmetric matrices obtained by using SC method, which make way for obtaining the Hermitian self-dual codes over $\mathcal{R}_1$, are given in Appendix 2.

## 3.1. CONSTRUCTION OF EUCLIDEAN SELF-DUAL CODES OVER THE RING $\mathbb{F}_2 + v\mathbb{F}_2$

### 3.1.1. The Euclidean Self-Dual Codes Obtained by DCC over $\mathcal{R}_1$

Let $A$ be a $k \times k$ orthogonal circulant matrix. The code $C_1$ whose generator matrix is of the form $[I_k, A]$ is a binary self-dual $[2k, k, d_1]$-code. We find the matrix $B$ which satisfies the condition given in Theorem 2.2.1. The code $C_2$ whose generator matrix is of the form $[I_k, A + B]$ is a binary self-dual $[2k, k, d_2]$-code. Thus, we obtain the Euclidean self-dual code $C$ over $\mathcal{R}_1$, where $C$ is a $[2k, k, min\{d_1, d_2\}]$-code. The minimum weights of the Euclidean self-dual codes over $\mathcal{R}_1$ that are obtained by using this construction method are given in Table 3.1.

In Table 3.1, $d_{max}(n)$ is the highest minimum weight of a binary linear code of length $n$ and of dimension $n/2$, $d_{DC}(n)$ is the minimum weight of the Euclidean self-dual code over $\mathcal{R}_1$ of length $n$ obtained by applying DCC method.

**Table 3.1:** The Euclidean self-dual codes over $\mathcal{R}_1$ obtained by using DCC.

| $Length\ n$ | $d_{max}(n)$ | $d_{DC}(n)$ | $Length\ n$ | $d_{max}(n)$ | $d_{DC}(n)$ |
|---|---|---|---|---|---|
| 2 | 2 | 2 | 52 | 10-12 | 10 |
| 4 | 2 | 2 | 54 | 11-13 | 10 |
| 6 | 3 | 2 | 56 | 12-14 | 8 |
| 8 | 4 | 4 | 58 | 12-14 | 10 |
| 10 | 4 | 2 | 60 | 12-14 | 12 |
| 12 | 4 | 4 | 62 | 12-15 | 10 |
| 14 | 4 | 2 | 64 | 12-16 | 12 |
| 16 | 5 | 4 | 66 | 12-16 | 12 |
| 18 | 6 | 4 | 68 | 13-16 | 12 |
| 20 | 6 | 4 | 70 | 14-16 | 10 |
| 22 | 7 | 6 | 72 | 15-17 | 12 |
| 24 | 8 | 8 | 74 | 14-18 | 12 |
| 26 | 7 | 6 | 76 | 14-18 | 12 |
| 28 | 8 | 4 | 78 | 15-18 | 12 |
| 30 | 8 | 6 | 80 | 16-19 | 16 |
| 32 | 8 | 8 | 82 | 14-20 | 14 |
| 34 | 8 | 6 | 84 | 15-20 | 12 |
| 36 | 8 | 6 | 86 | 16-20 | 14 |
| 38 | 8-9 | 8 | 88 | 17-20 | 16 |
| 40 | 9-10 | 8 | 90 | 18-21 | 14 |
| 42 | 10 | 6 | 92 | 16-22 | 14 |
| 44 | 10 | 8 | 94 | 16-22 | 14 |
| 46 | 11 | 10 | 96 | 16-22 | 16 |
| 48 | 12 | 12 | 98 | 17-22 | 14 |
| 50 | 10-12 | 10 | 100 | 18-23 | 14 |

### 3.1.2. The Euclidean Self-Dual Codes Obtained by BDCC over $\mathcal{R}_1$

While applying this construction, when $k$ is even, we take $\beta = \gamma = 1$ and $\alpha = 0$ in Theorem 2.4.1. In this case, we may obtain codes whose minimum distance is greater than 2. When $k$ is odd, we take $\beta = \gamma = 0$ and $\alpha = 1$. Since the first row of the matrix $G$ has the minimum weight, this row determines the minimum distance of the code, which is 2. If we take $\beta = \gamma = 1$ and $\alpha = 0$, we cannot obtain a self-dual code. The minimum weights of the Euclidean self-dual codes over $\mathcal{R}_1$ that are obtained by using this construction method can be seen in Table 3.2. Since the minimum weights of the Euclidean self-dual codes of length $4k-2$ are 2 for all positive integers $k$, we only tabulate the minimum weights of the Euclidean self-dual codes of length $4k$ for $k \leq 25$.

In Table 3.2, $d_{BDC}(n)$ is the minimum weight of the Euclidean self-dual code of length $n$ obtained by applying BDCC method.

**Table 3.2:** The Euclidean self-dual codes over $\mathcal{R}_1$ obtained by using BDCC.

| *Length n* | $d_{max}(n)$ | $d_{BDC}(n)$ | *Length n* | $d_{max}(n)$ | $d_{BDC}(n)$ |
|---|---|---|---|---|---|
| 4 | 2 | 2 | 56 | 12-14 | 12 |
| 8 | 4 | 4 | 60 | 12-14 | 12 |
| 12 | 4 | 4 | 64 | 12-16 | 12 |
| 16 | 5 | 4 | 68 | 13-16 | 12 |
| 20 | 6 | 4 | 72 | 15-17 | 12 |
| 24 | 8 | 8 | 76 | 14-18 | 12 |
| 28 | 8 | 6 | 80 | 16-19 | 12 |
| 32 | 8 | 8 | 84 | 15-20 | 12 |
| 36 | 8 | 8 | 88 | 17-20 | 16 |
| 40 | 9-10 | 8 | 92 | 16-22 | 14 |
| 44 | 10 | 8 | 96 | 16-22 | 16 |
| 48 | 12 | 12 | 100 | 18-23 | 16 |
| 52 | 10-12 | 10 | | | |

### 3.1.3. The Euclidean Self-Dual Codes Obtained by SC over $\mathcal{R}_1$

In this construction, we find all orthogonal symmetric $k \times k$ matrices for $k \leq 8$. When $k \geq 9$, the search field is too big for obtaining all orthogonal symmetric $k \times k$ matrices. We use random orthogonal symmetric $9 \times 9$ matrices to obtain a Euclidean self-dual code of length 18. Since investigating a matrix which is both orthogonal and symmetric is a restrictive criterion, we do our search up to length 18.

The minimum weights of the Euclidean self-dual codes over $\mathcal{R}_1$ that are obtained by using this construction method are given in Table 3.3. In this table, $d_S(n)$ is the minimum weight of the Euclidean self-dual code of length $n$ obtained by applying SC method.

**Table 3.3:** The Euclidean self-dual codes over $\mathcal{R}_1$ obtained by using SC.

| $Length\ n$ | $d_{max}(n)$ | $d_S(n)$ |
|---|---|---|
| 2 | 2 | 2 |
| 4 | 2 | 2 |
| 6 | 3 | 2 |
| 8 | 4 | 4 |
| 10 | 4 | 2 |
| 12 | 4 | 4 |
| 14 | 4 | 4 |
| 16 | 5 | 4 |
| 18 | 6 | 4 |

## 3.2. CONSTRUCTION OF HERMITIAN SELF-DUAL CODES OVER THE RING $\mathbb{F}_2 + v\mathbb{F}_2$

### 3.2.1. The Hermitian Self-Dual Codes Obtained by DCC over $\mathcal{R}_1$

It is well known that each row of any circulant matrix corresponds to a polynomial by taking the coordinates of any row as coefficients. Moreover, a circulant binary $k \times k$ matrix is invertible if and only if the corresponding polynomial of the first row is relatively prime to $x^k + 1$ [40]. Any polynomial that is relatively prime to $x^k + 1$ must have an odd number of terms (non-zero coefficients). Therefore, each row of the circulant matrix should have odd weight.

In using this construction, we take a vector of odd weight from $\mathbb{F}_2^k$ as the first row of a matrix and search for invertible circulant matrices $A$. Then we write the matrix $B$ related to $A$, via $B = A + (A^T)^{-1}$, which is given in Theorem 2.2.2. Since both $A$ and $B$ are circulant, $A + vB$ is also a circulant matrix. Then we build the matrix $[I, A + vB]$ which generates Hermitian self-dual codes over $\mathcal{R}_1$.

**Corollary 3.2.1.** The Hermitian self-dual code that is obtained by using DCC is of Type IV.

**Proof:** Since the matrices $A$ and $(A^T)^{-1}$ are invertible binary circulant matrices, the proof directly follows from Proposition 2.2.6 and Corollary 2.2.4. ■

The minimum weights of the Hermitian self-dual codes over $\mathcal{R}_1$ that are obtained by using this construction method are given in Table 3.4.

In Table 3.4, $d_{max}(n)$ is the highest minimum weight of a binary linear code of length $n$ and of dimension $\lfloor (n+1)/2 \rfloor$, $d_{DC}(n)$ is the minimum weight of the Hermitian self-dual code over $\mathcal{R}_1$ of length $n$ obtained by applying DCC method, $Vector$ is the first row of the circulant matrix $A$.

**Table 3.4:** The Hermitian self-dual codes over $\mathcal{R}_1$ obtained by using DCC.

| *Length n* | *Vector* | $d_{max}(n)$ | $d_{DC}(n)$ |
|---|---|---|---|
| 2 | (1) | 2 | 2 |
| 4 | (10) | 2 | 2 |
| 6 | (001) | 3 | 2 |
| 8 | (1011) | 4 | 4 |
| 10 | (10011) | 4 | 4 |
| 12 | (100101) | 4 | 4 |
| 14 | (1000011) | 4 | 4 |
| 16 | (10000011) | 5 | 4 |
| 18 | (100000101) | 6 | 4 |
| 20 | (1000010111) | 6 | 6 |
| 22 | (10000010111) | 7 | 6 |
| 24 | (100010111101) | 8 | 8 |
| 26 | (1000000010111) | 7 | 6 |
| 28 | (10000011101011) | 8 | 8 |
| 30 | (100001001110110) | 8 | 8 |
| 32 | (1000000010110111) | 8 | 8 |
| 34 | (10000000010110111) | 8 | 8 |
| 36 | (100000000010111101) | 8 | 8 |
| 38 | (1000000000010110111) | 8-9 | 8 |
| 40 | (10000000000010110111) | 9-10 | 8 |
| 42 | (100000000110010111101) | 10 | 10 |
| 44 | (1000000000100111110101) | 10 | 10 |
| 46 | (10000000000100111011011) | 11 | 10 |
| 48 | (100000111011011010111111) | 12 | 12 |
| 50 | (1000000000000100111011011) | 10-12 | 10 |

### 3.2.2. The Hermitian Self-Dual Codes Obtained by BDCC over $\mathcal{R}_1$

While applying this construction, we take $\beta = \gamma = 1$, because otherwise the minimum distance of the codes would be 2. We investigate the bordered $k \times k$ matrices for all $0 \leq k \leq 25$ for the possible values of $\alpha$ of 0 or 1. When $\alpha = 0$ and $k \not\equiv 0 \pmod 4$, we observe that the bordered matrices are always singular. The matrix $A$ can be any circulant matrix without a restriction on the weight of rows.

The minimum weights of the Hermitian self-dual codes over $\mathcal{R}_1$ that are obtained by using this construction method can be seen in Table 3.5. In this table, $d_{BDC}(n)$ is the minimum weight of the Hermitian self-dual code over $\mathcal{R}_1$ of length $n$ obtained by applying BDCC method.

**Table 3.5:** The Hermitian self-dual codes over $\mathcal{R}_1$ obtained by using BDCC.

| *Length n* | $\alpha$ | *Vector* | $d_{max}(n)$ | $d_{BDC}(n)$ | *Type* |
|---|---|---|---|---|---|
| 2 | 1 | none | 2 | 2 | Type IV |
| 4 | 0 | (1) | 2 | 2 | Type IV |
| 6 | 1 | (10) | 3 | 3 | Type I |
| 8 | 0 | (101) | 4 | 4 | Type IV |
| 10 | 1 | (1011) | 4 | 3 | Type I |
| 12 | 0 | (10101) | 4 | 4 | Type I |
| 14 | 1 | (100110) | 4 | 4 | Type I |
| 16 | 1 | (1011010) | 5 | 4 | Type I |
| 18 | 1 | (10000101) | 6 | 5 | Type I |
| 20 | 0 | (100001110) | 6 | 6 | Type IV |
| 22 | 1 | (1000010111) | 7 | 6 | Type I |
| 24 | 0 | (10001011011) | 8 | 8 | Type IV |
| 26 | 1 | (101111100010) | 7 | 6 | Type I |
| 28 | 0 | (1011000011010) | 8 | 8 | Type IV |
| 30 | 1 | (10000010011100) | 8 | 7 | Type I |
| 32 | 0 | (101111001001000) | 8 | 8 | Type I |
| 34 | 1 | (1000000010110111) | 8 | 8 | Type I |
| 36 | 0 | (10000000110010111) | 8 | 8 | Type I |
| 38 | 1 | (101111011100010101) | 8-9 | 8 | Type I |
| 40 | 1 | (1000000001110101101) | 9-10 | 8 | Type I |
| 42 | 1 | (10000000000010110111) | 10 | 8 | Type I |
| 44 | 0 | (100000000011010011011) | 10 | 10 | Type IV |
| 46 | 1 | (1000001001011011111001) | 11 | 10 | Type I |
| 48 | 0 | (10000011011011111010111) | 12 | 12 | Type IV |
| 50 | 1 | (100000000001001101100111) | 10-12 | 10 | Type I |

### 3.2.3. The Hermitian Self-Dual Codes Obtained by SC over $\mathcal{R}_1$

Symmetric matrices can be created by adding/deleting suitable rows and columns to/from any good invertible symmetric matrices. By a good invertible symmetric matrix, we mean a matrix which allows us to obtain a good Hermitian self-dual code for SC method.

In this work, we have considered all invertible symmetric $k \times k$ matrices for $k < 7$. When $k = 7$ and $k = 8$, the search field is too big for finding all invertible symmetric $k \times k$ matrices. Therefore, to make fewer computations, we restrict our search field by using the following matrix form:

$$\left[\begin{array}{c|c|c} & x_0 & x_1 \quad x_2 \quad \cdot \quad \cdot \quad \cdot \quad x_{k-1} \\ \cline{2-3} & x_1 & \\ & x_2 & \\ I_k & \cdot & A \\ & \cdot & \\ & \cdot & \\ & x_{k-1} & \end{array}\right], \tag{3.1}$$

where $x_i$ $(0 \leq i \leq k-1)$ are in $\mathbb{F}_2$ and $A$ is an invertible symmetric $(k-1) \times (k-1)$ matrix. For $k = 7$ and $k = 8$, we use suitable $6 \times 6$ and $7 \times 7$ invertible symmetric $A$ matrices, respectively and add proper row and column to each matrix. For the case $9 \leq k \leq 14$, we take advantage of the generator matrix $[I_{12}|U]$ of the $[24, 12, 8]$ extended binary Golay code with

$$U = \begin{bmatrix} 1&0&0&1&1&1&1&1&0&0&0&1 \\ 0&1&0&0&1&1&1&1&1&0&1&0 \\ 0&0&1&0&0&1&1&1&1&1&0&1 \\ 1&0&0&1&0&0&1&1&1&1&1&0 \\ 1&1&0&0&1&0&0&1&1&1&0&1 \\ 1&1&1&0&0&1&0&0&1&1&1&0 \\ 1&1&1&1&0&0&1&0&0&1&0&1 \\ 1&1&1&1&1&0&0&1&0&0&1&0 \\ 0&1&1&1&1&1&0&0&1&0&0&1 \\ 0&0&1&1&1&1&1&0&0&1&1&0 \\ 0&1&0&1&0&1&0&1&0&1&1&1 \\ 1&0&1&0&1&0&1&0&1&0&1&1 \end{bmatrix}.$$

For $9 \leq k \leq 11$, invertible symmetric matrices can be obtained by deleting suitable rows and columns of $U$. For $k = 12$, we use the matrix $U$. For $k = 13$ and $k = 14$, invertible symmetric matrices can be derived by adding suitable rows and columns to the matrix $U$ as shown in (3.1).

The minimum weights of the Hermitian self-dual codes over $\mathcal{R}_1$ that are obtained by using this construction method are given in Table 3.6. In this table, $d_S(n)$ is the minimum weight of the Hermitian self-dual code over $\mathcal{R}_1$ of length $n$ obtained by applying SC method and $Matrix$ is the invertible symmetric matrix used in SC. The invertible symmetric matrices that we obtained are given in Appendix 2.

**Table 3.6:** The Hermitian self-dual codes over $\mathcal{R}_1$ obtained by using SC.

| *Length n* | *Matrix* | $d_{max}(n)$ | $d_S(n)$ | *Type* | *Length n* | *Matrix* | $d_{max}(n)$ | $d_S(n)$ | *Type* |
|---|---|---|---|---|---|---|---|---|---|
| 2 | sm2 | 2 | 2 | Type IV | 16 | sm16 | 5 | 4 | Type I |
| 4 | sm4 | 2 | 2 | Type I | 18 | sm18 | 6 | 5 | Type I |
| 6 | sm6 | 3 | 3 | Type I | 20 | sm20 | 6 | 6 | Type I |
| 8 | sm8 | 4 | 4 | Type IV | 22 | sm22 | 7 | 7 | Type I |
| 10 | sm10 | 4 | 4 | Type I | 24 | sm24 | 8 | 8 | Type IV |
| 12 | sm12 | 4 | 4 | Type I | 26 | sm26 | 7 | 6 | Type I |
| 14 | sm14 | 4 | 4 | Type I | 28 | sm28 | 8 | 6 | Type I |

### 3.2.4. The Hermitian Self-Dual Codes Obtained by the Shortening Method over $\mathcal{R}_1$

In Tables 3.7-3.9, we list the minimum weights of the Hermitian self-dual codes of odd length obtained by shortening of free Hermitian self-dual codes of even length, which in turn, are obtained by the related construction methods described in Subsection 2.4. In Table 3.8 and Table 3.9, *shorten* is the non-zero non-unit element, $i$ is the row and the column position that the shortening method applied.

The Hermitian self-dual code of length 31 with minimum Hamming weight 8 is obtained by shortening the code of length 32 in the first position by $v$, which is obtained by using BDCC method. The existence of the Hermitian self-dual code of length 31 with minimum Hamming weight 8 was not known in the previous works ([14], [41]).

## 3.3. CONSTRUCTION OF EUCLIDEAN SELF-DUAL CODES OVER THE RING $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$

In order to construct free Euclidean self-dual codes of length $n$ over $\mathcal{R}_2$, we use binary self-dual codes of length $3n$. Since the Gray map $\phi_1$ is a distance preserving map, we are able to carry out our research on $\mathbb{F}_2$ instead of $\mathcal{R}_2$. We first find two orthogonal matrices $A$ and $B$, and seek the matrix $D$ which satisfies the condition $BD^T + DB^T + DD^T = 0$.

**Table 3.7:** The Hermitian self-dual codes over $\mathcal{R}_1$ obtained by applying the shortening method to the codes obtained with DCC.

| *Length n* | *Vector* | $d_{max}(n)$ | $d_{DC}(n)$ |
|---|---|---|---|
| 1 | (1) | 1 | 1 |
| 3 | (10) | 2 | 1 |
| 5 | (001) | 2 | 1 |
| 7 | (1011) | 3 | 3 |
| 9 | (10011) | 3 | 3 |
| 11 | (100101) | 4 | 3 |
| 13 | (1000011) | 4 | 3 |
| 15 | (10000011) | 4 | 3 |
| 17 | (100000101) | 5 | 3 |
| 19 | (1000010111) | 5 | 5 |
| 21 | (10000010111) | 6 | 5 |
| 23 | (100010111101) | 7 | 7 |
| 25 | (1000000010111) | 6 | 5 |
| 27 | (10000011101011) | 7 | 7 |
| 29 | (100001001110110) | 7 | 7 |
| 31 | (1000000010110111) | 8 | 7 |
| 33 | (10000000010110111) | 8 | 7 |
| 35 | (100000000010111101) | 8 | 7 |
| 37 | (1000000000010110111) | 8 | 7 |
| 39 | (10000000000010110111) | 8-9 | 7 |
| 41 | (100000000110010111101) | 9-10 | 9 |
| 43 | (1000000000100111110101) | 9-10 | 9 |
| 45 | (10000000000100111011011) | 10 | 9 |
| 47 | (100000111011011010111111) | 11 | 11 |
| 49 | (1000000000000100111011011) | 10-12 | 9 |

### 3.3.1. The Euclidean Self-Dual Codes Obtained by DCC over $\mathcal{R}_2$

In this method, we take $k \times k$ binary matrices $A$, $B$ and $D$ as circulant matrices and make sure that these matrices satisfy the conditions given in Theorem 2.3.2. Since we make $2^{3k}$ searches for each $k$, obtaining all suitable matrices for $k \geq 8$ is a time-consuming process. For $k = 8, 9$ and 10, we terminate the program after running 72 hours and we note the best results. Therefore we give the minimum Lee weights of the Euclidean self-dual codes up to length 20. The Euclidean self-dual codes obtained by using DCC method is tabulated in Table 3.10. In this table, $d_{max}(n)$ denotes the minimum of the highest minimum weight of a binary self-dual code of length $n$ and the highest minimum weight of a Euclidean

**Table 3.8:** The Hermitian self-dual codes over $\mathcal{R}_1$ obtained by applying the shortening method to the codes obtained with BDCC.

| *Length n* | $\alpha$ | *Vector* | *shorten*, $i$ | $d_{max}(n)$ | $d_{BDC}(n)$ |
|---|---|---|---|---|---|
| 1 | 1 | none | $v$, first | 1 | 1 |
| 3 | 0 | (1) | $1+v$, last | 2 | 2 |
| 5 | 1 | (10) | $v$, last | 2 | 2 |
| 7 | 0 | (101) | $1+v$, last | 3 | 3 |
| 9 | 1 | (1011) | $v$, first | 3 | 3 |
| 11 | 0 | (10101) | $v$, first | 4 | 4 |
| 13 | 1 | (100110) | $1+v$, first | 4 | 4 |
| 15 | 1 | (1011010) | $v$, first | 4 | 4 |
| 17 | 1 | (10000101) | $v$, last | 5 | 4 |
| 19 | 0 | (100001110) | $1+v$, last | 5 | 5 |
| 21 | 1 | (1000010111) | $1+v$, last | 6 | 5 |
| 23 | 0 | (10001011011) | $1+v$, last | 7 | 7 |
| 25 | 1 | (101111100010) | $v$, first | 6 | 6 |
| 27 | 0 | (1011000011010) | $1+v$, last | 7 | 7 |
| 29 | 1 | (10000010011100) | $1+v$, first | 7 | 7 |
| 31 | 0 | (101111001001000) | $v$, first | 8 | 8 |
| 33 | 1 | (1000000010110111) | $1+v$, last | 8 | 7 |
| 35 | 0 | (10000000110010111) | $v$, first | 8 | 8 |
| 37 | 1 | (101111011100010101) | $1+v$, first | 8 | 8 |
| 39 | 1 | (1000000001110101101) | $1+v$, first | 8-9 | 8 |
| 41 | 1 | (10000000000010110111) | $v$, first | 9-10 | 8 |
| 43 | 0 | (100000000011010011011) | $v$, last | 9-10 | 9 |
| 45 | 1 | (1000001001011011111001) | $1+v$, last | 10 | 9 |
| 47 | 0 | (10000011011011111010111) | $v$, last | 11 | 11 |
| 49 | 1 | (100000000001001101100111) | $1+v$, first | 10-12 | 10 |

self-dual code over $\mathcal{R}_1$ of length $n$, $d_{DC}$ is the minimum Lee weight of the Euclidean self-dual code over $\mathcal{R}_2$ of length $n$ obtained by applying DCC method.

### 3.3.2. The Euclidean Self-Dual Codes Obtained by BDCC over $\mathcal{R}_2$

In this method, we take three $k \times k$ binary circulant matrices as $A$, $B$ and $D$ which satisfy the conditions given in Theorem 2.3.2. Then we add suitable vectors as first rows and first columns, which is shown in Theorem 2.4.1. We combine these matrices with identity matrix and calculate the minimum distances of the codes. We make $2^{3k}$ searches to obtain $k \times k$ circulant matrices. We try all possible vector combinations which correspond to make 64 calculations. We look for the Euclidean self-dual codes of lengths up to 20 and we always find the minimum Lee weights of these codes as 2. Therefore, we do not feel

**Table 3.9:** The Hermitian self-dual codes over $\mathcal{R}_1$ obtained by applying the shortening method to the codes obtained with SC.

| *Length n* | *Matrix* | *shorten, i* | $d_{max}(n)$ | $d_S(n)$ | *Length n* | *Matrix* | *shorten, i* | $d_{max}(n)$ | $d_S(n)$ |
|---|---|---|---|---|---|---|---|---|---|
| 1 | sm2 | $1+v$, 1 | 1 | 1 | 15 | sm16 | $1+v$, 8 | 4 | 4 |
| 3 | sm4 | $1+v$, 2 | 2 | 2 | 17 | sm18 | $1+v$, 9 | 5 | 4 |
| 5 | sm6 | $v$, 3 | 2 | 2 | 19 | sm20 | $1+v$, 10 | 5 | 5 |
| 7 | sm8 | $v$, 4 | 3 | 3 | 21 | sm22 | $1+v$, 10 | 6 | 6 |
| 9 | sm10 | $v$, 5 | 3 | 3 | 23 | sm24 | $1+v$, 12 | 7 | 7 |
| 11 | sm12 | $1+v$, 6 | 4 | 4 | 25 | sm26 | $1+v$, 13 | 6 | 5 |
| 13 | sm14 | $1+v$, 7 | 4 | 4 | 27 | sm28 | $1+v$, 14 | 7 | 5 |

**Table 3.10:** The Euclidean self-dual codes over $\mathcal{R}_2$ obtained by using DCC.

| *Length n* | $d_{max}(n)$ | $d_{DC}(n)$ |
|---|---|---|
| 2 | 2 | 2 |
| 4 | 2 | 2 |
| 6 | 3 | 2 |
| 8 | 4 | 4 |
| 10 | 4 | 2 |
| 12 | 4 | 4 |
| 14 | 4 | 2 |
| 16 | 5 | 4 |
| 18 | 6 | 4 |
| 20 | 6 | 4 |

the need to create a table.

### 3.3.3. The Euclidean Self-Dual Codes Obtained by SC over $\mathcal{R}_2$

In order to find all binary $k \times k$ symmetric matrices, we make $2^{\frac{k(k+1)}{2}}$ calculations while we make $2^k$ calculations to find all $k \times k$ circulant matrices. Therefore, after searching all $k \times k$ circulant matrices for $k \leq 4$, we use random matrices for $k = 5$ and $k = 6$ in this construction method. We obtain $[8,4,4]$ Euclidean self-dual code with minimum Lee weight 4. The other Euclidean self-dual codes of length $2k$ obtained by applying SC method have the minimum Lee weight 2, where $1 \leq k \leq 8, k \neq 4$. Therefore, we do not tabulate this result.

## 3.4. CONSTRUCTION OF HERMITIAN SELF-DUAL CODES OVER THE RING $\mathbb{F}_2 \times \mathbb{F}_2 + v\mathbb{F}_2$

In order to construct free Hermitian self-dual codes of length $n$ over $\mathcal{R}_2$, we use binary self-dual codes of length $3n$. Since the Gray map $\phi_2$ is a distance preserving map, we are able to carry out our research on $\mathbb{F}_2$ instead of $\mathcal{R}_2$. We first find an orthogonal matrix $A$, then we seek the matrices $B$ and $D$ which satisfy the equations $BB^T + BD^T = I_k$ and $BD^T + DB^T = 0$.

### 3.4.1. The Hermitian Self-Dual Codes Obtained by DCC over $\mathcal{R}_2$

In this method, we take $k \times k$ binary circulant matrices $A$, $B$ and $D$ and make sure that these matrices satisfy the conditions given in Theorem 2.3.5. We run the algorithms up to length 30. We give the minimum Gray weights of Hermitian self-dual codes obtained by using DCC in Table 3.11. In this table, $d_{max}(n)$ denotes the upper bound mentioned in Proposition 2.3.13 and $d_{DC}$ is the highest minimum Gray weight of the Hermitian self-dual code over $\mathcal{R}_2$ of length $n$ obtained by applying DCC method.

**Table 3.11:** The Hermitian self-dual codes over $\mathcal{R}_2$ obtained by using DCC.

| *Length n* | $d_{max}(n)$ | $d_{DC}(n)$ |
|---|---|---|
| 2 | 4 | 2 |
| 4 | 4 | 4 |
| 6 | 6 | 4 |
| 8 | 8 | 8 |
| 10 | 8 | 6 |
| 12 | 8 | 6 |
| 14 | 8 | 6 |
| 16 | 10 | 8 |
| 18 | 12 | 8 |
| 20 | 12 | 10 |
| 22 | 14 | 6 |
| 24 | 16 | 6 |
| 26 | 14 | 6 |
| 28 | 16 | 6 |
| 30 | 16 | 6 |

### 3.4.2. The Hermitian Self-Dual Codes Obtained by BDCC over $\mathcal{R}_2$

In using this construction, we run the algorithms up to length 20 and we give the results in Table 3.12. In this table, $d_{BDC}$ is the highest minimum Gray weight of the Hermitian self-dual code over $\mathcal{R}_2$ of length $n$ obtained by applying BDCC method.

**Table 3.12:** The Hermitian self-dual codes over $\mathcal{R}_2$ obtained by using BDCC.

| *Length n* | $d_{max}(n)$ | $d_{BDC}(n)$ |
|---|---|---|
| 2 | 4 | 2 |
| 4 | 4 | 4 |
| 6 | 6 | 4 |
| 8 | 8 | 4 |
| 10 | 8 | 6 |
| 12 | 8 | 6 |
| 14 | 8 | 6 |
| 16 | 10 | 8 |
| 18 | 12 | 6 |
| 20 | 12 | 6 |

### 3.4.3. The Hermitian Self-Dual Codes Obtained by SC over $\mathcal{R}_2$

In this construction, we make the similar investigation that we have done so far. Since the search space is too wide and we have restrictive criterations, we search the codes up to length 12. The results are tabulated in Table 3.13. In this table, $d_S$ denotes the highest minimum Gray weight of the Hermitian self-dual code over $\mathcal{R}_2$ of length $n$ obtained by applying SC method.

**Table 3.13:** The Hermitian self-dual codes over $\mathcal{R}_2$ obtained by using SC.

| *Length n* | $d_{max}(n)$ | $d_S(n)$ |
|---|---|---|
| 2 | 4 | 2 |
| 4 | 4 | 4 |
| 6 | 6 | 4 |
| 8 | 8 | 8 |
| 10 | 8 | 6 |
| 12 | 8 | 4 |

# 4. DISCUSSION

In this dissertation, we apply construction methods given in Section 2.4 in order to obtain Euclidean and Hermitian self-dual codes over $\mathcal{R}_1$ and $\mathcal{R}_2$. We first study Euclidean self-dual codes over $\mathcal{R}_1$. When we compare our results ($d_{DC}(n)$, $d_{BDC}(n)$ and $d_S(n)$) with $d_{max}(n)$, we see that our results are slightly close to $d_{max}(n)$ in DCC and mostly close to $d_{max}(n)$ except when $n \equiv 2 \pmod 4$ in BDCC. The results obtained in SC coincide with the results obtained in DCC except when $n = 14$.

Then, we focus on Hermitian self-dual codes over $\mathcal{R}_1$. When we analogize our results ($d_{DC}(n)$, $d_{BDC}(n)$ and $d_S(n)$) with $d_{max}(n)$, we see that our results are slightly close to $d_{max}(n)$ in SC. For $n = 22$, we obtain extremal codes only by using SC method.

When we make a comparison between the Euclidean self-dual codes and the Hermitian self-dual codes of the same even length over $\mathcal{R}_1$, we see that the minimum distances of the Hermitian self-dual codes are higher than or equal to the minimum distances of the Euclidean self-dual codes in each construction method.

Later, we concentrate on shortening Hermitian self-dual codes over $\mathcal{R}_1$. We shorten Hermitian self-dual codes of even length by multiplying the $i^{th}$ row of the generator matrix with a non-zero non-unit element and puncturing the $i^{th}$ column to obtain Hermitian self-dual codes of odd length, where $1 \leq i \leq n/2$. In DCC, it does not matter which row is chosen to multiply by $v$ or $1+v$. In BDCC, we consider the first row to apply the shortening method. When the results are not satisfying, the last row is chosen to apply the shortening method. In SC, since the search field is too big, we begin with applying the shortening method to the last row. When the results are not satisfying, the previous rows are considered in order. We observe in DCC that the code obtained from shortening by $v$ has the same minimum weight with the code obtained from shortening by $1+v$. On the other hand, in BDCC and SC, the minimum weights of the child codes of the same length may be affected depending on shortening by $v$ or $1+v$. After applying the shortening method to the codes obtained by DCC method, the minimum distances of the child codes decrease from $d$ to $d-1$ for all lengths without any exceptions. However, in BDCC and SC, the minimum distances of the child codes remain unchanged for some lengths.

The Hermitian self-dual code of parameters [31,16,8] is obtained by shortening the code of length 32 in the first position by $v$, which is obtained by using BDCC method. The existence of the Hermitian self-dual code of length 31 with minimum Hamming weight 8 was not known in the previous works ([14], [41]).

We seek the minimum distances of free Euclidean and free Hermitian self-dual codes over $\mathcal{R}_2$ by using DCC, BDCC and SC methods. When we compare the minimum Lee distances of the Euclidean self-dual codes obtained by DCC, BDCC and SC methods, since the minimum Lee distances of the Euclidean self-dual codes obtained by BDCC method are 2 and the minimum Lee distances of the Euclidean self-dual codes obtained by SC method are 2 except for $n = 8$, DCC method seems more efficient to obtain Euclidean self-dual codes over $\mathcal{R}_2$.

For the Hermitian self-dual codes, when we analogize our results $(d_{DC}(n)$, $d_{BDC}(n)$ and $d_S(n))$ with $d_{max}(n)$, we see that DCC method allows to obtain the codes whose minimum Gray weights are higher for the same length.

When we make a comparison between the Euclidean self-dual codes and the Hermitian self-dual codes of the same length over $\mathcal{R}_2$, we see that the minimum Gray distances of the Hermitian self-dual codes are mostly higher than or rarely equal to the minimum Lee distances of the Euclidean self-dual codes in each construction method.

In all methods, we obtain extremal Euclidean and extremal Hermitian self-dual codes for some lengths over $\mathcal{R}_1$ and $\mathcal{R}_2$.

# 5. CONCLUSION AND RECOMMENDATIONS

Huffman left some questions unanswered in [41]. One of them is "Is there a Hermitian self-dual code of length 31 with minimum Hamming weight 8?". Starting from this question, in this dissertation, we first considered Euclidean and Hermitian self-dual codes over $\mathcal{R}_1$. We found the necessary and sufficient conditions to obtain free Euclidean and free Hermitian self-dual codes whose generator matrices are of the form $[I, A + vB]$ over $\mathcal{R}_1$. We used DCC, BDCC and SC methods to construct free Euclidean and free Hermitian self-dual codes of even length over $\mathcal{R}_1$. Moreover, we described a new shortening method. By applying this shortening method to Hermitian self-dual codes of even length, we obtained Hermitian self-dual codes of odd length over $\mathcal{R}_1$. We found the Hermitian self-dual code of parameters $[31, 16, 8]$. We tabulated the minimum weights of the Euclidean self-dual codes of even length up to 100. We extended the table of the minimum weights of the Hermitian self-dual codes of all lengths up to 50. We examined the MacWilliams identities for linear codes over $\mathcal{R}_1$.

Later, we introduced the direct product ring $\mathcal{R}_2$ which is a commutative Frobenius ring of order 8. This direct product ring is new for the coding theory literature. We determined two Gray maps and two inner products on this ring. We identified the necessary and sufficient conditions to obtain free Euclidean and free Hermitian self-dual codes whose generator matrices are of the form $[(I_k, I_k)|(A, B + vD)]$ over $\mathcal{R}_2$. After that, we applied DCC, BDCC and SC methods to obtain free Euclidean and free Hermitian self-dual codes of even length over $\mathcal{R}_2$. Since we considered only self-dual codes and self-dual codes over $\mathcal{R}_2$ exist only for even lengths, we only dealt with the codes of even length. We tabulated our results for the Euclidean and Hermitian self-dual codes of even length over $\mathcal{R}_2$. We proved the MacWilliams identities which establish a connection between the weight enumerator of a linear code and its dual code.

In further research, some other suitable methods can be considered to obtain Euclidean and Hermitian self-dual codes over $\mathcal{R}_1$. Some other codes such as cyclic codes can be studied over $\mathcal{R}_2$. $\mathcal{R}_2$ may be generalized to the ring $\mathbb{F}_p \times \mathbb{F}_p + v\mathbb{F}_p$ with $p$ a prime power, and codes over this ring can be investigated.

# REFERENCES

[1]. Shannon, C.E., 1948, A Mathematical Theory of Communication, *The Bell System Technical Journal*, 27, 379-423.

[2]. Hammons, A.R., Kumar, P.V., Calderbank, A.R., Sloane, N.J.A. and Solé, P., 1994, The $\mathbb{Z}_4$-linearity of Kerdock, Preparata, Goethals and related codes, *IEEE Transactions on Information Theory*, 40, 301-319.

[3]. Wood, J., 1999, Duality for modules over finite rings and applications to coding theory, *American Journal of Mathematics*, 121, 555-575.

[4]. Gulliver T.A. and Harada, M., 2001, Optimal double circulant $\mathbb{Z}_4$-codes, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, 2227, 122-128.

[5]. Huffman, W.C., 1998, Decompositions and extremal Type II codes over $\mathbb{Z}_4$, *IEEE Transactions on Information Theory*, 44, 800-809.

[6]. Wolfmann, J., 1999, Negacyclic and cyclic codes over $\mathbb{Z}_4$, *IEEE Transactions on Information Theory*, 45, 2527-2532.

[7]. Yildiz, B. and Karadeniz, S., 2014, Linear codes over $\mathbb{Z}_4 + u\mathbb{Z}_4$: MacWilliams identities, projections, and formally self-dual codes, *Finite Fields and Their Applications*, 27, 24-40.

[8]. Bachoc, C., 1997, Applications of coding theory to the construction of modular lattices, *Journal of Combinatorial Theory, Series A*, 78, 92-119.

[9]. Al-Ashker M. and Isleem, I., 2008, Simplex linear codes over the ring $\mathbb{F}_2 + v\mathbb{F}_2$, *An-Najah University Journal for Research-A (Natural Sciences)* 22, 25-42.

[10]. Betsumiya, K., Gulliver A. and Harada, M., 2003, Extremal self-dual codes over $\mathbb{F}_2 \times \mathbb{F}_2$, *Designs, Codes and Cryptography*, 28, 171-186.

[11]. Dertli, A. and Cengellenmis, Y., 2011, MacDonald codes over the ring $\mathbb{F}_2 + v\mathbb{F}_2$, *International Journal of Algebra*, 5, 985-991.

[12]. Dougherty, S.T., Gaborit, P., Harada, M., Munemasa, A. and Solé, P., 1999, Type IV self-dual codes over rings, *IEEE Transactions on Information Theory*, 45, 2345-2360.

[13]. Zhu, S., Wang, Y. and Shi, M., 2010, Some results on cyclic codes over $\mathbb{F}_2 + v\mathbb{F}_2$, *IEEE Transactions on Information Theory*, 56, 1680-1684.

[14]. Betsumiya, K. and Harada, M., 2004, Optimal self-dual codes over $\mathbb{F}_2 \times \mathbb{F}_2$ with respect to the Hamming weight, *IEEE Transactions on Information Theory*, 50, 356-358.

[15]. Abualrub, T., Aydin, N. and Seneviratne, P., 2012, On $\theta$-cyclic codes over $\mathbb{F}_2 + v\mathbb{F}_2$, *Australasian Journal of Combinatorics*, 54, 115-126.

[16]. Kaya, A., Yildiz, B. and Siap, I., 2014, Quadratic residue codes over $\mathbb{F}_p + v\mathbb{F}_p$ and their Gray images, *Journal of Pure and Applied Algebra*, 218, 1999-2011.

[17]. Batoul, A., Guenda, K., Kaya, A. and Yildiz, B., 2015, Cyclic isodual and formally self-dual codes over $\mathbb{F}_q + v\mathbb{F}_q$, *European Journal of Pure and Applied Mathematics*, 8, 64-80.

[18]. Cengellenmis, Y., Dertli, A. and Dougherty, S.T., 2014, Codes over an infinite family of rings with a Gray map, *Designs, Codes and Cryptography*, 72, 559-580.

[19]. Borges, J., Fernández-Córdoba, C., Pujol, J., Rifà, J. and Villanueva, M., 2010, $\mathbb{Z}_2\mathbb{Z}_4$-linear codes: generator matrices and duality, *Designs, Codes and Cryptography*, 54, 167-179.

[20]. Aydogdu, I., Abualrub, T. and Siap, I., 2014, On $\mathbb{Z}_2\mathbb{Z}_2[u]$-additive codes, *International Journal of Computer Mathematics*, 92, 1806-1814.

[21]. Aydogdu, I., 2018, The structure of one weight linear and cyclic over $\mathbb{Z}_2^r \times (\mathbb{Z}_2 + u\mathbb{Z}_2)^s$, *An International Journal of Optimization and Control: Theories & Applications*, 8, 92-101.

[22]. Bonnecase, A., Bracco, A.D., Dougherty, S.T., Nochefranca, L.R. and Solé, P., 2003, Cubic self-dual binary codes, *IEEE Transactions on Information Theory*, 49, 2253-2259.

[23]. Borges, J., Dougherty, S. T., Fernández-Córdoba, C. and Ten-Valls, R., 2018, Binary images of $\mathbb{Z}_2\mathbb{Z}_4$-additive cyclic codes, *IEEE Transactions on Information Theory*, 64, 7551-7556.

[24]. Borges, J., Fernández-Córdoba, C. and Ten-Valls, R., 2018, Linear and cyclic codes over direct product of finite chain rings, *Mathematical Methods in the Applied Sciences*, 41, 6519-6529.

[25]. Bilal, M., Borges, J., Dougherty, S.T. and Fernández-Córdoba, C., 2011, Maximum distance separable codes over $\mathbb{Z}_4$ and $\mathbb{Z}_2 \times \mathbb{Z}_4$, *Designs, Codes and Cryptography*, 61, 31-40.

[26]. Bosma, W., Cannon, J. and Playoust, C., 1997, The Magma algebra system I: The user language, *Journal of Symbolic Computation*, 24, 235-265.

[27]. Ling, S. and Xing, C., 2004, *Coding Theory: A First Course*, Cambridge University Press, New York, ISBN-13: 978-0521821919.

[28]. MacWilliams, F.J. and Sloane, N.J.A., 1977, *The Theory of Error Correcting Codes*, Amsterdam, North-Holland.

[29]. Huffman, W.C. and Pless, V., 2003, *Fundamentals of Error Correcting Codes*, University Press, Cambridge.

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

[31]. Assmus, E.F., Mattson, H.F. and Turyn, R.J., 1967, Research to develop the algebraic theory of codes, *Report AFCRL-67-0365*, Air Force Cambridge Research Laboratories, Belford, MA.

[32]. Dougherty, S.T. and Liu, H., 2009, Independence of vectors in codes over rings, *Designs, Codes and Cryptography*, 51(1), 55-68.

[33]. Klemm, M., 1989, Selbstduale Codes über dem ring der ganzen zahlen modulo 4, *Archiv der Mathematik*, 53, 201-207.

[34]. Dougherty, S.T., Harada, M. and Solé, P., 1999, Self-dual codes over rings and the Chinese remainder theorem, *Hokkaido Mathematical Journal*, 28, 253-283.

[35]. Dougherty, S.T., 2017, *Algebraic Coding Theory Over Finite Commutative Rings*, Springer International Publishing, eBook, ISBN: 978-3-319-59806-2.

[36]. McCoy, N.H., 1948, *Rings and Ideals*, The Mathematical Association of America, eBook, ISBN: 978-1-61444-008-6.

[37]. Rains, E.M., 1998, Shadow bounds for self dual codes, *IEEE Transactions on Information Theory*, 44, 134-139.

[38]. Kaya, A., 2017, DNA computing via codes over various alphabets, *BASIC 2017*, Universitas Indonesia, Indonesia.

[39]. Karadeniz, S., Dougherty, S.T. and Yildiz, B., 2014, Constructing formally self-dual codes over $R_k$, *Discrete Applied Mathematics*, 167, 188-196.

[40]. Jungnickel, D., 1993, *Finite Fields: Structure and Arithmetics*, Bibliographisches Institut, Mannheim.

[41]. Huffman, W.C., 2005, On the classification and enumeration of self-dual codes, *Finite Fields and Their Applications*, 11, 451-490.

# APPENDICES

## APPENDIX 1. The Magma Implementation

The Magma algorithm used to obtain the Euclidean self-dual codes of length 4 over $\mathcal{R}_1$ by using DCC method is given as follows.

```
n:=2;
F:=GF(2);
SetLogFile("outec2.txt");
IMat:=KMatrixSpace(F,n,n)!0;
for i:=1 to n do
IMat[i,i]:=1;
end for;
rnd2:=KMatrixSpace(F,n,n);
function CM(rnd1,n)
X:=rnd2!0;
for i:=1 to n do
X[i]:=(rnd1);
rnd1:=Rotate(rnd1,1);
end for;
return(X);
end function;
j1:=2;
```

```
j2:=2;
for i1:=0 to 1 do
for i2:=0 to 1 do
v:= KSpace(F,n);
a:=v![i1,i2];
A:=CM(a,n);
AT:=Transpose(A);
J:=A*AT;
if J eq IMat then
IA:=HorizontalJoin(IMat,A);
C1:=LinearCode(IA);
e1:=Length(C1);
e2:=Dimension(C1);
e3:=MinimumWeight(C1);
if e3 ge j1 then
"c1", e1, e2, e3, "vector", a;
end if;
for k1:=0 to 1 do
for k2:=0 to 1 do
v1:= KSpace(F,n);
b:=v1![k1,k2];
```

```
B:=CM(b,n);
if A*Transpose(B) eq (B*Transpose(A))+(B*Transpose(B)) then
IAB:=HorizontalJoin(IMat,A+B);
C2:=LinearCode(IAB);
e4:=Length(C2);
e5:=Dimension(C2);
e6:=MinimumWeight(C2);
if e6 ge j2 then
"c2", e4, e5, e6, "vector", b;
"the matrix A", A;
"the matrix B", B;
end if;
end if;
end for;
end for;
end if;
end for;
end for;
```

## APPENDIX 2. The Symmetric Matrices

The symmetric matrices obtained by using SC method in Section 3.2.3 are given in the following. Here, sm2$Y$ denotes the $Y \times Y$ invertible symmetric matrix.

$$\text{sm2}=\begin{bmatrix} 1 \end{bmatrix}, \text{sm4}=\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}, \text{sm6}=\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \ \text{sm8}=\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix},$$

$$\text{sm10}=\begin{bmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}, \qquad \text{sm12}=\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix},$$

$$\text{sm14}=\begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \ \text{sm16}=\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix},$$

$$\text{sm18}=\begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}, \ \text{sm20}=\begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix},$$

$$sm22=\begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix},$$

$$sm24=\begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix},$$

$$sm26=\begin{bmatrix}1&0&0&0&0&0&0&1&0&1&1&0&1\\0&1&0&0&1&1&1&1&1&0&0&0&1\\0&0&1&0&0&1&1&1&1&1&0&1&0\\0&0&0&1&0&0&1&1&1&1&1&0&1\\0&1&0&0&1&0&0&1&1&1&1&1&0\\0&1&1&0&0&1&0&0&1&1&1&0&1\\0&1&1&1&0&0&1&0&0&1&1&1&0\\1&1&1&1&1&0&0&1&0&0&1&0&1\\0&1&1&1&1&1&0&0&1&0&0&1&0\\1&0&1&1&1&1&1&0&0&1&0&0&1\\1&0&0&1&1&1&1&1&0&0&1&1&0\\0&0&1&0&1&0&1&0&1&0&1&1&1\\1&1&0&1&0&1&0&1&0&1&0&1&1\end{bmatrix},$$

$$sm28=\begin{bmatrix}1&0&0&0&0&0&0&1&0&1&1&0&1&0\\0&1&0&0&0&0&0&0&1&0&1&1&0&1\\0&0&1&0&0&1&1&1&1&1&0&0&0&1\\0&0&0&1&0&0&1&1&1&1&1&0&1&0\\0&0&0&0&1&0&0&1&1&1&1&1&0&1\\0&0&1&0&0&1&0&0&1&1&1&1&1&0\\0&0&1&1&0&0&1&0&0&1&1&1&0&1\\1&0&1&1&1&0&0&1&0&0&1&1&1&0\\0&1&1&1&1&1&0&0&1&0&0&1&0&1\\1&0&1&1&1&1&1&0&0&1&0&0&1&0\\1&1&0&1&1&1&1&1&0&0&1&0&0&1\\0&1&0&0&1&1&1&1&1&0&0&1&1&0\\1&0&0&1&0&1&0&1&0&1&0&1&1&1\\0&1&1&0&1&0&1&0&1&0&1&0&1&1\end{bmatrix}.$$

# CURRICULUM VITAE

| Personal Information | |
|---|---|
| Name Surname | Refia AKSOY |
| Place of Birth | Keçiören |
| Date of Birth | 01.03.1989 |
| Nationality | ☒ T.C. □ Other: |
| E-mail | refia06@gmail.com |

| Educational Information | |
|---|---|
| **B.Sc.** | |
| University | Ege University |
| Faculty | Faculty of Science and Arts |
| Department | Department of Mathematics |
| Graduation Year | 2010 |

| M.Sc. | |
|---|---|
| University | Fatih University |
| Institute | Graduate School of Sciences and Engineering |
| Department | Department of Mathematics |
| Programme | Mathematics Programme |
| Graduation Year | 2014 |

| Ph.D. | |
|---|---|
| University | İstanbul University |
| Institute | Institute of Graduate Studies in Science and Engineering |
| Department | Department of Mathematics |
| Programme | Mathematics Programme |

## Publications

Karadeniz, S. and Aksoy, R., 2015, Self-dual $R_k$ lifts of binary self-dual codes, *Finite Fields and Their Applications*, 34, 317-326.

Karadeniz, S. and Aksoy, R., Lifts of self-dual codes, ICMS 2014, *The 4th International Congress on Mathematical Software*, Hanyang University, Seoul/South Korea, August 2014.

Aksoy, R. and Karadeniz, S., Constructing Hermitian self-dual codes over $\mathbb{F}_2 + v\mathbb{F}_2$, WAIFI 2016, *International Workshop on the Arithmetic of Finite Fields*, Ghent University, Ghent/Belgium, July 2016.