On Additive $GF(4)$ Codes

Philippe Gaborit*, W. Cary Huffman†, Jon-Lark Kim‡, Vera Pless‡

Abstract

In this paper we discuss the status of the classification of the extremal additive codes over $GF(4)$ which are self-dual under the trace inner product. These codes are of interest because of their connection to quantum codes. There are 5 inequivalent codes of length 8, 8 of length 9, at least 56 of length 10, and exactly 1 of length 11. We discuss the methods of classification, the concepts of lengthening and shortening, and connections to binary codes.

1 Introduction

An *additive code* $\mathcal{C}$ *over* $GF(4)$ *of length* n is an additive subgroup of $GF(4)^n$. As $\mathcal{C}$ is a free $GF(2)$-module, it has size 2^k for some $0 \leq k \leq 2n$. We call $\mathcal{C}$ an $(n, 2^k)$ code. It has a basis, as a $GF(2)$-module, consisting of k basis vectors; a *generator matrix* of $\mathcal{C}$ will be a $k \times n$ matrix with entries in $GF(4)$ whose rows are a basis of $\mathcal{C}$. Interest in additive codes over $GF(4)$ has arisen because of their correspondence to quantum codes as described in [3]. There is a natural inner product arising from the trace map, on the additive codes induced by this correspondence. If we let $GF(4) = \{0, 1, \omega, \overline{\omega}\}$ where $\overline{\omega} = \omega^2 = 1 + \omega$, the *trace* map $\mathrm{Tr} : GF(4) \to GF(2)$ is given by

$$\mathrm{Tr}(x) = x + x^2.$$

In particular $\mathrm{Tr}(0) = \mathrm{Tr}(1) = 0$ and $\mathrm{Tr}(\omega) = \mathrm{Tr}(\overline{\omega}) = 1$. The *conjugate* of $x \in GF(4)$, denoted $\overline{x}$, is the image of x under the Frobenius automorphism; in other words, $\overline{0} = 0$ $\overline{1} = 1$, and $\overline{\overline{\omega}} = \omega$. We now define the *trace inner product* of two vectors $\mathbf{x} = x_1x_2 \cdots x_n$ and $\mathbf{y} = y_1y_2 \cdots y_n$ in $GF(4)^n$ to be

$$\mathbf{x}\star\mathbf{y} = \sum_{i=1}^{n} \mathrm{Tr}(x_i\overline{y_i}).$$

Example 1.1 Let $\mathcal{G}_6$ be the $[6, 3, 4]$ *hexacode* whose generator matrix as a linear $GF(4)$-code is

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

*LACO, Université de Limoges, France; e-mail: gaborit@unilim.edu

†Department of Mathematical and Computer Sciences, Loyola University, Chicago, IL 60626, USA; e-mail: wch@math.luc.edu

‡Department of Mathematics, Statistics, and Computer Science, 322 SEO, University of Illinois–Chicago, 851 S. Morgan, Chicago, IL 60607-7045, USA; e-mail: jlkim@math.uic.edu, pless@math.uic.edu

This is also an additive code; thinking of $\mathcal{G}_6$ as an additive code, it has generator matrix

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

If $\mathcal{C}$ is an additive code, its *dual*, denoted $\mathcal{C}^\perp$, is the additive code $\{\mathbf{x} \in GF(4)^n \mid \mathbf{x}\star\mathbf{c} = 0 \text{ for all } \mathbf{c} \in \mathcal{C}\}$. If $\mathcal{C}$ is an $(n, 2^k)$ code, then $\mathcal{C}^\perp$ is an $(n, 2^{2n-k})$ code. As usual, $\mathcal{C}$ is *self-orthogonal* if $\mathcal{C} \subseteq \mathcal{C}^\perp$ and *self-dual* if $\mathcal{C} = \mathcal{C}^\perp$. In particular, if $\mathcal{C}$ is self-dual, $\mathcal{C}$ is an $(n, 2^n)$ code. The code $\mathcal{G}_6$ in Example 1.1 is self-dual as an additive code. (Any $GF(4)$-linear code that is self-orthogonal under the Hermitian inner product is a self-orthogonal additive code under the trace inner product.)

As usual, the *weight* $\mathrm{wt}(\mathbf{c})$ of $\mathbf{c} \in \mathcal{C}$ is the number of nonzero components of $\mathbf{c}$. The minimum weight d of $\mathcal{C}$ is the smallest weight of any nonzero codeword in $\mathcal{C}$. If $\mathcal{C}$ is an $(n, 2^k)$ additive code of minimum weight d, $\mathcal{C}$ is called an $(n, 2^k, d)$ code. $\mathcal{C}$ is *Type II* if $\mathcal{C}$ is self-dual and all codewords have even weight; as we see later Type II codes of length n exist only if n is even. If $\mathcal{C}$ is self-dual but some codeword has odd weight (in which case the code cannot be $GF(4)$-linear), the code is *Type I*. There is a bound on the minimum weight of an additive self-dual code [10, Theorem 33]. If d_I, d_{II} is the minimum distance of an additive self-dual Type I, Type II code, respectively, of length $n > 1$, then

$$d_I \le \begin{cases} 2\left\lfloor \frac{n}{6} \right\rfloor + 1 & \text{if } n \equiv 0 \pmod 6 \\ 2\left\lfloor \frac{n}{6} \right\rfloor + 3 & \text{if } n \equiv 5 \pmod 6 \\ 2\left\lfloor \frac{n}{6} \right\rfloor + 2 & \text{otherwise} \end{cases} \tag{1}$$

$$d_{II} \le 2\left\lfloor \frac{n}{6} \right\rfloor + 2. \tag{2}$$

A code that meets the appropriate bound is called *extremal*. Type II codes meeting the bound d_{II} have a unique weight enumerator. This property is not true for Type I codes, to have a unique weight enumerator they would have to meet the [n/2]+1 bound (cf [8]), which is never reached except for the hexacode.

We say that two additive codes $\mathcal{C}_1$ and $\mathcal{C}_2$ are *equivalent* provided there is a map sending the codewords of $\mathcal{C}_1$ onto the codewords of $\mathcal{C}_2$ where the map consists of a permutation of coordinates followed by a scaling of coordinates by elements of $GF(4)$ followed by conjugation of some of the coordinates. In terms of generator matrices, if G_1 is a generator matrix of $\mathcal{C}_1$, then there is a generator matrix of $\mathcal{C}_2$ obtained from G_1 by permuting columns, scaling columns, and then conjugating some of the resulting columns. Notice that permuting coordinates, scaling coordinates, and conjugating some coordinates of a self-orthogonal (or self-dual) code does not change self-orthogonality (or self-duality). The *automorphism group* of $\mathcal{C}$, denoted $\mathrm{Aut}(\mathcal{C})$, consists of all maps which permute coordinates, scale coordinates, and conjugate coordinates that send codewords of $\mathcal{C}$ to codewords of $\mathcal{C}$.

The purpose of this paper is to examine the classification of the extremal Type I and II additive codes.

n	d_I	num$_I$	d_{II}	num$_{II}$	n	d_I	num$_I$	d_{II}	num$_{II}$
2	1^O	1	2^E	1	10	4^E	≥ 51	4^E	≥ 5
3	2^E	1	–	–	11	5^E	1	–	–
4	2^E	1	2^E	1	12	5^E	≥ 7	6^E	1
5	3^E	1	–	–	13	5^O	≥ 5	–	–
6	3^E	1	4^E	1	14	6^E	?	6^E	≥ 490
7	3^O	3	–	–	15	6^E	≥ 2	–	–
8	4^E	2	4^E	3	16	6^E	≥ 3	6^E	≥ 4
9	4^E	8	–	–					

Table 1: Type I and II Codes of Length $2 \leq n \leq 16$

2 Results

We summarize what is currently known about extremal (or optimal) Type I and II codes in Table 1. In the table, codes have length 'n'; 'd_I' is the largest minimum weight for which a Type I code is known to exist while 'd_{II}' is the largest minimum weight for which a Type II code is known to exist. The superscript 'E' indicates that the code is extremal; the superscript 'O' indicates the code is not extremal but no code of the given type can exist with a larger minimum weight (and so the code is *optimal*). The number of inequivalent Type I and II codes of the given minimum weight is listed under 'num$_I$' and 'num$_{II}$', respectively; when the number in the column is exact (without $\geq$), the classification of those codes is complete.

The numbers for $2 \leq n \leq 7$ come from the work of Höhn [8]. Höhn also found that there are exactly 3 inequivalent extremal Type II codes of length 8; we found that there are exactly 2 inequivalent extremal Type I codes of length 8. We also found the entries for $n = 9$, 10, and 11. The unique extremal Type II code of length 12 is called the *dodecacode* [3]; as we will see, using the fact that the extremal self-dual code of length 11 is known to be unique, it is easy to show that the extremal Type II code of length 12 is also unique. It is known [10] that there is no extremal length 13 codes. We found that there are at least 5 optimal self-dual codes of length 13 and at least 3 extremal Type I codes of length 16; there are exactly 4 extremal Hermitian self-dual $GF(4)$-linear codes of length 16 [5] showing that there are at least 4 extremal Type II codes of length 16. The Type II entry for length $n = 14$ comes from [1].

3 Techniques

In this section, we describe the techniques used to produce some of the results in the previous section.

3.1 The Balance Principle

We notice that a result found in [9] applies to additive self-dual codes with minor modifications. This is sometimes called the Balance Principle.

Theorem 3.1 *Let $\mathcal{C}$ be a self-dual additive $(n, 2^n)$ code. Let $n = n_1 + n_2$. We can choose* *a generator matrix G for $\mathcal{C}$ of the following form:*

$$G = \begin{bmatrix} A & O \\ O & B \\ D & E \end{bmatrix}. \tag{3}$$

Here A is a $k_1 \times n_1$ matrix where $[A \quad O]$ generates the largest subcode of $\mathcal{C}$ which is zero on the right-most n_2 coordinates. Similarly B is a $k_2 \times n_2$ matrix where $[O \quad B]$ generates the largest subcode of $\mathcal{C}$ which is zero on the left-most n_1 coordinates. (O is the appropriate size zero matrix.) Let $\mathcal{A}$ be the additive code of length n_1 generated by A, $\mathcal{A}_D$ the additive code of length n_1 generated by the rows of A and D, $\mathcal{B}$ the additive code of length n_2 generated by B, and $\mathcal{B}_E$ the additive code of length n_2 generated by the rows of B and E. The following hold.

(i) $n - k_2 = 2n_1 - k_1$ *and* $n - k_1 = 2n_2 - k_2$.

(ii) $\mathcal{A}^\perp = \mathcal{A}_D$ *and* $\mathcal{B}^\perp = \mathcal{B}_E$.

We apply the Balance Principle in the following manner. Suppose we wish to construct a self-dual additive $(n, 2^n, d)$ code $\mathcal{C}$. By rearranging coordinates, we may assume that there is a minimum weight codeword whose nonzero components are the left-most $n_1 = d$. Then by scaling the coordinates appropriately, we may assume that the first row of A is all 1's. There is at most one more row of A and that row can have only ω's and $\overline{\omega}$'s. By conjugation of columns, we may assume that row has all ω's; as the two rows must be orthogonal under the trace inner product, d must be even. Thus we can always choose A to be either

$$(a) \quad A = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \end{bmatrix} \qquad \text{or} \qquad (b) \quad A = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ \omega & \omega & \omega & \cdots & \omega \end{bmatrix} \tag{4}$$

with (b) occurring only when d is even. Thus we have $k_1 = 1$ or $k_1 = 2$. This determines the value for k_2.

Example 3.2 If $\mathcal{C}$ is a self-dual additive $(8, 2^8, 4)$ code, then the Balance Principle shows that a generator matrix for G can be assumed to be of the form

$$\left[\begin{array}{cccc|cccc} 1 & 1 & 1 & 1 & & & & \\ \hline & & & & 1 & 1 & 1 & 1 \\ \hline 1 & 0 & 0 & 0 & & & & \\ 0 & 1 & 0 & 0 & & & & \\ 0 & 0 & 1 & 0 & \multicolumn{4}{c}{E} \\ \omega & \omega & 0 & 0 & & & & \\ \omega & 0 & \omega & 0 & & & & \\ \omega & 0 & 0 & \omega & & & & \end{array}\right] \quad \text{or} \quad \left[\begin{array}{cccc|cccc} 1 & 1 & 1 & 1 & & & & \\ \omega & \omega & \omega & \omega & & & & \\ \hline & & & & 1 & 1 & 1 & 1 \\ & & & & \omega & \omega & \omega & \omega \\ \hline 1 & 1 & 0 & 0 & & & & \\ \omega & \omega & 0 & 0 & \multicolumn{4}{c}{E} \\ 1 & 0 & 1 & 0 & & & & \\ \omega & 0 & \omega & 0 & & & & \end{array}\right].$$

Using these two forms, self-duality, the minimum weight, and equivalence, MAGMA ([2]) was used to fill in the possible entries for E and thereby produce exactly 5 inequivalent self-dual additive $(8, 2^8, 4)$ codes, 3 being of Type II.

Example 3.3 Suppose we want to produce a self-dual additive $(9, 2^9, 4)$ code $\mathcal{C}$. We let a generator matrix for $\mathcal{C}$ have form (3). If A has form (4)(a), up to equivalence there are 3 choices for the 2×5 matrix B; if A has form (4)(b), up to equivalence there is only 1 choice for the 3×5 matrix B. Given A, D can be explicitly written down and E can be determined by MAGMA. These lead to exactly 8 inequivalent codes.

Example 3.4 Suppose we want to produce a self-dual additive $(11, 2^{11}, 5)$ code $\mathcal{C}$. A generator matrix for $\mathcal{C}$ has form (3) where this time A has form (4)(a) only and D can explicitly be written down. Up to equivalence there are 2 choices for the 2×6 matrix B. Using MAGMA E can be determined. This leads to exactly 1 code up to equivalence.

3.2 Equivalence of Codes

We determine the equivalence or the nonequivalence between two additive codes over $GF(4)$ by operations on binary codes. Let $\mathcal{C}$ be an additive code over $GF(4)$. The transformation from $\mathcal{C}$ into a binary code is done by applying the map β: $0 \to 000, 1 \to 011, \omega \to 101, \overline{\omega} \to 110$ to the coordinates of each generator of $\mathcal{C}$. The map β sends an $(n, 2^k)$ additive code $\mathcal{C}$ to $\beta(\mathcal{C})$, a $[3n, k]$ binary code.

The six different operations, on a column of an additive code which do not change the weight (multiplication by a constant with or without the conjugation) are then expressed on F_2^3 as the six possible permutations of 3 elements.

Let Ω be the $(n, 2^{2n})$ additive code containing all the elements of $GF(4)^n$. The automorphism group of $\beta(\Omega)$ is isomorphic to the permutation group $S_3^n.S_n$. The straightforward lemma of [3] follows:

Lemma 3.5 *Let $\mathcal{C}$ be a $(n, 2^k)$ additive code over $GF(4)$, then $Aut(\mathcal{C})$ is isomorphic to $Aut(\beta(\mathcal{C})) \cap Aut(\beta(\Omega))$.*

The following lemma will be mainly used in the proof of Lemma 3.7.

Lemma 3.6 *Let $\mathcal{B}_1$, $\mathcal{B}_2$ be binary codes with a permutation P such that $\mathcal{B}_1 P = \mathcal{B}_2$. Then there exists a permutation Q such that $\mathcal{B}_1 Q = \mathcal{B}_2$ if and only if $Q \in Aut(\mathcal{B}_1)P$, a right coset of $Aut(\mathcal{B}_1)$ in the symmetric group on the length of $\mathcal{B}_1$.*

Proof: Suppose that there exists a permutation Q such that $\mathcal{B}_1 Q = \mathcal{B}_2$. Then we have $\mathcal{B}_1 P = \mathcal{B}_2 = \mathcal{B}_1 Q$, implying $\mathcal{B}_1 Q P^{-1} = \mathcal{B}_1$. Hence $QP^{-1} \in Aut(\mathcal{B}_1)$, i.e. $Q \in Aut(\mathcal{B}_1)P$. Conversely suppose $Q \in Aut(\mathcal{B}_1)P$. Then $Q = AP$ for some $A \in Aut(\mathcal{B}_1)$. Hence $\mathcal{B}_1 Q = (\mathcal{B}_1 A)P = \mathcal{B}_1 P = \mathcal{B}_2$, as required. $\square$

Suppose $\mathcal{C}_1$ and $\mathcal{C}_2$ are additive codes over $GF(4)$ of length n associated to binary codes $\beta(\mathcal{C}_1) = \mathcal{B}_1$ and $\beta(\mathcal{C}_2) = \mathcal{B}_2$ of length $3n$. If M is a map with $\mathcal{C}_1 M = \mathcal{C}_2$, where M permutes columns, scales columns, and conjugates columns, then let $\widetilde{M}$ be the associated permutation on the $3n$ columns of $\mathcal{B}_1$. Then immediately we see that $\mathcal{C}_1$ is equivalent to $\mathcal{C}_2$ as additive codes over $GF(4)$ if and only if there is a permutation $\widetilde{M}$ on $3n$ elements and $\widetilde{M} \in Aut(\beta(\Omega))$ with $\mathcal{B}_1 \widetilde{M} = \mathcal{B}_2$.

If two additive codes are equivalent, clearly their binary images by the map β are also equivalent. In particular if the binary images of two additive codes are not equivalent, neither are the additive codes. The following lemma relates the equivalence of additive codes over $GF(4)$ to the equivalence of binary codes by the map β.

Lemma 3.7 *Suppose P is a permutation on $3n$ elements such that $\mathcal{B}_1 P = \mathcal{B}_2$. Then $\mathcal{C}_1$ is equivalent to $\mathcal{C}_2$ as additive codes over $GF(4)$ if and only if $Aut(\mathcal{B}_1)P \cap Aut(\beta(\Omega)) \neq \emptyset$.*

Proof: Suppose $\mathcal{C}_1$ is equivalent to $\mathcal{C}_2$. Then there exists a map M such that $\mathcal{C}_1 M = \mathcal{C}_2$. So we have

$$\mathcal{B}_1 \widetilde{M} = \mathcal{B}_2 \Longleftrightarrow \widetilde{M} \in Aut(\mathcal{B}_1)P \quad \text{by Lemma 3.6.}$$

Since $\widetilde{M} \in Aut(\beta(\Omega))$ it follows that

$$\widetilde{M} \in Aut(\mathcal{B}_1)P \cap Aut(\beta(\Omega)) \neq \emptyset.$$

Conversely suppose $Aut(\mathcal{B}_1)P \cap Aut(\beta(\Omega)) \neq \emptyset$. Let $\widetilde{M} \in Aut(\mathcal{B}_1)P \cap Aut(\beta(\Omega))$. Then $\widetilde{M} = AP$ for some $A \in Aut(\mathcal{B}_1)$. It follows from Lemma 3.6 that $\mathcal{B}_1\widetilde{M} = \mathcal{B}_2$. Since $\widetilde{M} \in Aut(\beta(\Omega))$, we have that $\mathcal{C}_1 M = \mathcal{C}_2$. Hence $\mathcal{C}_1$ and $\mathcal{C}_2$ are equivalent as additive codes. This completes the proof. □

This method was written with the MAGMA [2] system. Although it is a way of checking the equivalence of two additive codes, the search for $Aut(\mathcal{B}_1)P \cap Aut(\beta(\Omega)) \neq \emptyset$ of the lemma slows considerably the process and makes it too long for codes over length 16.

In fact we also developed a concept of *restricted* equivalence which consists in checking intermediate fillings of the matrix E (as described in Example 3.2). The method used is almost the same as the method described in [7]. We considered intermediate fillings of the matrix E. We kept only one possible intermediate filling when different intermediate codes corresponding to different fillings were equivalent, as additive codes, through equivalnece maps which preserved the first four or five columns.

Eventually this process enabled us to considerably decrease the number of equivalence to check.

3.3 Lengthening and Shortening

If $\mathcal{C}$ is a self-dual additive $(n, 2^n, d)$ code, we can obtain a code of length $n-1$ by a process we call *shortening*. We use this term even though it differs from the usual meaning as our notion can be regarded as a generalization. Let G be a generator matrix of $\mathcal{C}$. As $\mathcal{C}$ is self-dual, no column of G is all 0's. Choose any column of G, say the i^{th} one. The entries in column i can be any of 0, 1, ω, or $\overline{\omega}$. By row reducing G (to obtain another generator matrix we call G again), we can make all the entries in column i equal to 0 except for one or two entries; if two they would be two of the three values 1, ω, or $\overline{\omega}$. The *shortened code of* $\mathcal{C}$ *on coordinate* i, denoted $\mathcal{C}'$ is the code with generator matrix G' obtained from G by eliminating one row of G with a nonzero entry in column i and then eliminating column i. If there is only one nonzero entry in column i of G, then $\mathcal{C}'$ is $\mathcal{C}$ shortened in the usual sense. Clearly $\mathcal{C}'$ is a self-dual additive $(n-1, 2^{n-1}, d')$ code with $d' \geq d - 1$.

Example 3.8 Let

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

We obtain two inequivalent codes by shortening on the first column. They have the following generator matrices:

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 1 \\ \omega & \omega \end{bmatrix}.$$

We would get the same codes shortening on the second column. To shorten on the third column, we would replace G by

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

Shortening yields the same codes as before.

Example 3.9 If

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

then no matter which column we shorten on, we obtain a shortened code with generator matrix

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

which is the usual shortened code. It also is the same as the first of the shortened codes in Example 3.8.

From Examples 3.8 and 3.9, we see that inequivalent codes can have equivalent shortened codes.

We can reverse shortening by *lengthening* a self-dual additive $(n-1, 2^{n-1}, d')$ code $\mathcal{C}'$ to obtain a self-dual additive $(n, 2^n, d)$ code $\mathcal{C}$. To do this, take a generator matrix G' of $\mathcal{C}'$ and adjoin an arbitrary n^{th} row $\mathbf{x}$ at the bottom. Then add an n^{th} column on the right according to the following scheme: First place a 1 in the n^{th} row. Second place a 0 or ω in each of the first $n-1$ rows so that each of the new length n rows is orthogonal to the length n bottom row. (Note that if we begin with two orthogonal rows of length $n-1$ and want to adjoin a nonzero element to one of the rows, we must adjoin the same nonzero element to the other row if we wish to create two orthogonal rows of length n.) The resulting matrix G, which we denote by

$$G = \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & \omega \\ \hline \mathbf{x} & 1 \end{array}\right], \tag{5}$$

clearly generates a self-dual additive $(n, 2^n, d)$ code with $d \leq d' + 1$. If the new row of length $n-1$ added to the bottom of G' is already in $\mathcal{C}'$, the lengthened code is a direct sum of $\mathcal{C}'$ and the $(1, 2, 1)$ code generated by $[1]$.

Example 3.10 Let

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

If we adjoin the row $[\omega\ \omega]$, then

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

Note that by multiplying column 3 by $\overline{\omega}$ and then conjugating column 3, we obtain a generator matrix for the $(3, 2^3)$ code of Example 3.8.

Example 3.11 Let

$$G' = \begin{bmatrix} 1 & 1 \\ \omega & \omega \end{bmatrix}.$$

If we adjoin the row [1 0], then

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

which is the code $(3, 2^3)$ of Example 3.8.

Examples 3.8 and 3.9 show that equivalent codes can be obtained by lengthening inequivalent codes.

We want to expand the concept of lengthening. Instead of (5) we could use the following generator matrices to lengthen:

$$\left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & \overline{\omega} \\ \hline \mathbf{x} & 1 \end{array}\right], \quad \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & 1 \\ \hline \mathbf{x} & \omega \end{array}\right], \quad \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & \overline{\omega} \\ \hline \mathbf{x} & \omega \end{array}\right], \quad \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & 1 \\ \hline \mathbf{x} & \overline{\omega} \end{array}\right], \text{ or } \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & \omega \\ \hline \mathbf{x} & \overline{\omega} \end{array}\right]. \tag{6}$$

However by scaling and/or conjugating the last column of the matrices in (6), we obtain the matrix (5) giving the following result.

Lemma 3.12 *The codes generated by any of the matrices in* (5) *or* (6) *are equivalent.*

As a result of this lemma we expand our definition of lengthening to allow us to lengthen using any of the six matrices in (5) and (6). Up to equivalence lengthening and shortening are inverse operations. Thus we have the following theorem.

Theorem 3.13 *Every self-dual additive $(n-1, 2^{n-1})$ code is shortened from a self-dual additive $(n, 2^n)$ code, and every self-dual additive $(n, 2^n)$ code is lengthened from a self-dual additive $(n-1, 2^{n-1})$ code.*

Example 3.14 Every self-dual additive $(13, 2^{13}, 5)$ code comes from lengthening a self-dual additive $(12, 2^{12}, d)$ code with $d \geq 4$. Lengthening the dodecacode we obtained 4 inequivalent $(13, 2^{13}, 5)$ codes; all possibilities were attempted. One can also shorten a length 14 code to obtain a length 13 code; shortening the length 14 extended quadratic residue code q_{14} leads to a $(13, 2^{13}, 5)$ code as indicated in [10]. The latter is not equivalent to the four codes found from lengthening.

Example 3.15 By lengthening q_{14}, we found one self-dual additive $(15, 2^{15}, 6)$ code inequivalent to the known $(15, 2^{15}, 6)$ cyclic code c_{15} found in [10].

Example 3.16 Every self-dual additive $(16, 2^{16}, 6)$ code comes from lengthening a self-dual additive $(15, 2^{15}, d)$ code with $d \geq 5$. Lengthening c_{15} yielded 3 inequivalent $(16, 2^{16}, 6)$ Type I codes; not all possibilities were attempted.

In [8], Höhn shows that a Type II code of length n exists only if n is even. If n is even, he defines (analogously to the situation of doubly-even binary codes of length $8r$ and their length $8r-2$ children [4]) a child of a Type II length n code. Beginning with a self-dual additive code of length $n-1$, he also shows how to construct a parent, which will be of Type II. The length $n-1$ children are, in our terminology, shortenings of an even length n code. The parent of a length $n-1$ code $\mathcal{C}'$ is obtained by taking the lengthening vector $\mathbf{x}$ to be a vector not in $\mathcal{C}'$ but orthogonal to the subcode $\mathcal{C}_0$ of $\mathcal{C}'$ consisting of all its even weight vectors. The vector $\mathbf{x}$ must have odd weight; otherwise the code $\mathcal{C}_0$ and vector $\mathbf{x}$ generate an additive Type II code of odd length, a contradiction.

Example 3.17 Let

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

Then G generates a Type II code $\mathcal{C}$ of length 4. By shortening we obtain inequivalent children $\mathcal{C}_1$ and $\mathcal{C}_2$ with generator matrices

$$G_1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \quad \text{and} \quad G_2 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ \omega & \omega & \omega \end{bmatrix}.$$

Rearrange the rows of G_1 to obtain

$$G_1' = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

so that the first two rows of G_1' generate the even weight subcode $\mathcal{C}_0$ of $\mathcal{C}_1$. Lengthening using the second form in (6) with $\mathbf{x} = [\omega\ \omega\ \omega]$ gives $\mathcal{C}$ with generator matrix

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

Lengthening $\mathcal{C}_2$ using form (5) with $\mathbf{x} = [1\ 0\ 0]$ gives $\mathcal{C}$ with generator matrix

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

Theorem 3.18 *Let $\mathcal{C}'$ be a Type I code of odd length $n-1$. Then there is a unique Type II code $\mathcal{C}$ of length n obtained by lengthening $\mathcal{C}'$.*

Proof: There is a unique $(n-1, 2^{n-2})$ subcode $\mathcal{C}_0$ of $\mathcal{C}'$ consisting of even weight vectors. Let G' be a generator matrix of $\mathcal{C}'$ with the first $n-2$ rows a basis of $\mathcal{C}_0$; the bottom row has odd weight. If we lengthen using (5) and we adjoin 0 to each of the first $n-2$ rows and ω to the last row, then we must choose the lengthening vector $\mathbf{x}$ to be in $\mathcal{C}_0^\perp$;

to avoid a code of minimum weight 1, $\mathbf{x} \notin \mathcal{C}'$. Furthermore two choices of $\mathbf{x}$ in the same coset of $\mathcal{C}'$ give the same code $\mathcal{C}$. As there is only one coset of $\mathcal{C}'$ in $\mathcal{C}_0^{\perp}$ not equal to $\mathcal{C}'$, we have a unique lengthening to a Type II code. $\square$

The code $\mathcal{C}$ in Theorem 3.18 is the parent of $\mathcal{C}'$. Thus the theorem states that an odd length Type I code has a unique Type II extension and that extension is the parent.

Example 3.19 By Theorem 3.13, (1), and (2), every self-dual additive $(10, 2^{10}, 4)$ code can be obtained from a self-dual additive $(9, 2^9, d)$ code with $d = 3$ or 4. We lengthened some of the 8 $(9, 2^9, 4)$ codes obtained in Example 3.3 and quickly found 51 Type I codes. Five of the 8 parents of the $(9, 2^9, 4)$ codes were inequivalent $(10, 2^{10}, 4)$ Type II codes; the other three parents had minimum weight 2. It seems to be of little interest to try to enumerate all self-dual additive $(10, 2^{10}, 4)$ codes as the number appears to be very high.

While the Type II extension of a Type I code of odd length is unique, this example shows that there may be many more Type I extensions. If we wish to extend Type II codes, we may get many extensions, as Example 3.14 illustrates. The following result was stated in [3] without proof.

Corollary 3.20 *The dodecacode is the unique* $(12, 2^{12}, 6)$ *Type II code.*

Proof: A $(12, 2^{12}, 6)$ Type II code must be a parent of a self-dual additive $(11, 2^{11}, d)$ code $\mathcal{C}'$ with $d \geq 5$. This code, being of odd length, must be Type I; by (1), $d = 5$. By Example 3.4, $\mathcal{C}'$ is unique, and so there is a unique $(12, 2^{12}, 6)$ Type II code by Theorem 3.18. $\square$

4 Enumeration of Extremal Self-Dual Codes

4.1 Extremal Self-Dual Codes of lengths 8, 9, 11 and 12

In this section we enumerate the extremal self-dual additive codes for the lengths 8,9,and 11. We also list the dodecacode, the unique extremal Type II code of length 12. Table 2 gives the different codes and their weight distributions, where 'n' and 'd' stand respectively for the length and the minimum distance. The codes are then enumerated in the particular form described in the previous section.

• n=8:

$$QC\_8a = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & \omega & \omega & \omega & \omega \\ 0 & 1 & 0 & 0 & 1 & \omega & \overline{\omega} & 0 \\ 0 & 0 & 1 & 0 & \omega & \overline{\omega} & \overline{\omega} & \omega \\ \omega & \omega & 0 & 0 & 1 & 0 & \omega & \omega \\ \omega & 0 & \omega & 0 & 1 & \omega & \overline{\omega} & 1 \\ \omega & 0 & 0 & \omega & \overline{\omega} & \overline{\omega} & 0 & 1 \end{bmatrix}, \quad QC\_8b = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & \omega & \omega & \omega & \omega \\ 0 & 1 & 0 & 0 & \omega & \omega & \overline{\omega} & \overline{\omega} \\ 0 & 0 & 1 & 0 & \omega & \overline{\omega} & \overline{\omega} & \omega \\ \omega & \omega & 0 & 0 & \overline{\omega} & \overline{\omega} & 0 & 1 \\ \omega & 0 & \omega & 0 & 1 & \omega & \overline{\omega} & 1 \\ \omega & 0 & 0 & \omega & \omega & 1 & \overline{\omega} & 1 \end{bmatrix},$$

$$QC\_8c = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & \omega & \omega & 0 \\ 0 & 1 & 0 & 0 & 1 & \overline{\omega} & \overline{\omega} & 0 \\ 0 & 0 & 1 & 0 & \overline{\omega} & 1 & 0 & \omega \\ \omega & \omega & 0 & 0 & \omega & \omega & \omega & \omega \\ \omega & 0 & \omega & 0 & 1 & 1 & \omega & \omega \\ \omega & 0 & 0 & \omega & \overline{\omega} & \overline{\omega} & 1 & 1 \end{bmatrix}, \quad QC\_8d = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & \omega & \omega & 0 \\ 0 & 1 & 0 & 0 & 1 & \omega & 0 & \omega \\ 0 & 0 & 1 & 0 & \overline{\omega} & 1 & 0 & \omega \\ \omega & \omega & 0 & 0 & \omega & 1 & \overline{\omega} & 1 \\ \omega & 0 & \omega & 0 & 1 & 0 & 1 & 0 \\ \omega & 0 & 0 & \omega & \overline{\omega} & \omega & \omega & \overline{\omega} \end{bmatrix}, QC\_8e = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ \omega & \omega & \omega & \omega & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ \omega & \omega & 0 & 0 & \omega & \omega & 0 & 0 \\ 1 & 0 & 1 & 0 & \omega & \overline{\omega} & \omega & \overline{\omega} \\ \omega & 0 & \omega & 0 & \overline{\omega} & 1 & \overline{\omega} & 1 \end{bmatrix}.$$

Code	n	d	$\vert Aut(C)\vert$	A_0	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}
QC_8a	8	4	128	1	26	64	72	64	29				
QC_8b	8	4	384	1	26	64	72	64	29				
QC_8c	8	4	1152	1	42	0	168	0	45				
QC_8d	8	4	336	1	42	0	168	0	45				
QC_8e	8	4	6.1344	1	42	0	168	0	45				
QC_9a	9	4	12	1	18	72	120	144	117	40			
QC_9b	9	4	16	1	18	72	120	144	117	40			
QC_9c	9	4	12	1	26	48	136	160	93	48			
QC_9d	9	4	144	1	18	72	120	144	117	40			
QC_9e	9	4	16	1	26	48	136	160	93	48			
QC_9f	9	4	64	1	18	72	120	144	117	40			
QC_9g	9	4	192	1	26	48	136	160	93	48			
QC_9h	9	4	18	1	18	72	120	144	117	40			
QC_11	11	5	18	1	0	66	198	330	495	550	330	78	
QC_12	12	6	648	1	0	0	396	0	1485	0	1980	0	234

Table 2: Extremal Self-dual codes for lengths $8, 9, 11$ and 12

- n=9:

$$QC\_9a = \begin{bmatrix} 1&1&1&1&0&0&0&0&0\\ 0&0&0&0&1&1&1&1&1\\ 0&0&0&0&\omega&\omega&\omega&\omega&0\\ 1&0&0&0&\overline{\omega}&0&\overline{\omega}&\omega&\overline{\omega}\\ 0&1&0&0&1&1&0&\omega&\omega\\ 0&0&1&0&0&\omega&\overline{\omega}&\overline{\omega}&\omega\\ \omega&\omega&0&0&1&0&0&\overline{\omega}&\omega\\ \omega&0&\omega&0&1&\omega&1&\omega&1\\ \omega&0&0&\omega&0&\omega&0&\omega&1 \end{bmatrix}, \quad QC\_9b = \begin{bmatrix} 1&1&1&1&0&0&0&0&0\\ 0&0&0&0&1&1&1&1&1\\ 0&0&0&0&\omega&\omega&\omega&\omega&0\\ 1&0&0&0&\overline{\omega}&0&\overline{\omega}&\omega&\overline{\omega}\\ 0&1&0&0&1&1&0&\omega&\omega\\ 0&0&1&0&0&\omega&\overline{\omega}&\overline{\omega}&\omega\\ \omega&\omega&0&0&1&0&0&\overline{\omega}&\omega\\ \omega&0&\omega&0&\omega&0&0&0&\overline{\omega}\\ \omega&0&0&\omega&0&1&1&0&0 \end{bmatrix},$$

$$QC\_9c = \begin{bmatrix} 1&1&1&1&0&0&0&0&0\\ 0&0&0&0&1&1&1&1&1\\ 0&0&0&0&\omega&\omega&\omega&\omega&0\\ 1&0&0&0&\overline{\omega}&0&\overline{\omega}&\omega&\overline{\omega}\\ 0&1&0&0&1&1&0&\omega&\omega\\ 0&0&1&0&\omega&0&0&\omega&1\\ \omega&\omega&0&0&\omega&0&1&1&\overline{\omega}\\ \omega&0&\omega&0&\omega&\omega&\omega&0&\omega\\ \omega&0&0&\omega&1&1&\overline{\omega}&\overline{\omega}&1 \end{bmatrix}, \quad QC\_9d = \begin{bmatrix} 1&1&1&1&0&0&0&0&0\\ 0&0&0&0&1&1&1&1&1\\ 0&0&0&0&\omega&\omega&\omega&\omega&0\\ 1&0&0&0&\overline{\omega}&0&\overline{\omega}&\omega&\overline{\omega}\\ 0&1&0&0&0&\overline{\omega}&1&0&\omega\\ 0&0&1&0&\omega&1&0&1&\overline{\omega}\\ \omega&\omega&0&0&\omega&0&0&0&\overline{\omega}\\ \omega&0&\omega&0&1&0&1&\omega&\omega\\ \omega&0&0&\omega&0&\overline{\omega}&1&\omega&0 \end{bmatrix},$$

$$QC\_9e = \begin{bmatrix} 1&1&1&1&0&0&0&0&0\\ 0&0&0&0&1&1&1&1&1\\ 0&0&0&0&\omega&\omega&\omega&\omega&0\\ 1&0&0&0&\overline{\omega}&0&\overline{\omega}&\omega&\overline{\omega}\\ 0&1&0&0&0&\overline{\omega}&1&0&\omega\\ 0&0&1&0&\omega&1&0&1&\overline{\omega}\\ \omega&\omega&0&0&1&0&1&0&1\\ \omega&0&\omega&0&0&\omega&0&\omega&1\\ \omega&0&0&\omega&1&\omega&1&\omega&1 \end{bmatrix}, \quad QC\_9f = \begin{bmatrix} 1&1&1&1&0&0&0&0&0\\ 0&0&0&0&1&1&1&1&1\\ 0&0&0&0&\omega&\omega&\omega&\omega&0\\ 1&0&0&0&\overline{\omega}&0&\overline{\omega}&\omega&\overline{\omega}\\ 0&1&0&0&\omega&1&0&1&\overline{\omega}\\ 0&0&1&0&1&0&1&\omega&\overline{\omega}\\ \omega&\omega&0&0&1&0&0&\overline{\omega}&\omega\\ \omega&0&\omega&0&1&\omega&1&\omega&1\\ \omega&0&0&\omega&\omega&\omega&\overline{\omega}&1&\omega \end{bmatrix},$$

$$QC\_9g = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega & 0 \\ 1 & 0 & 0 & 0 & \overline{\omega} & 0 & \overline{\omega} & \omega & \overline{\omega} \\ 0 & 1 & 0 & 0 & \omega & 1 & 0 & 1 & \overline{\omega} \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & \omega & \overline{\omega} \\ \omega & \omega & 0 & 0 & 1 & 1 & \omega & \omega & 1 \\ \omega & 0 & \omega & 0 & 1 & \omega & 1 & \omega & 1 \\ \omega & 0 & 0 & \omega & \overline{\omega} & 0 & 0 & \overline{\omega} & 0 \end{bmatrix}, \text{ and } QC\_9h = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega & 0 \\ 1 & 0 & 0 & 0 & \overline{\omega} & 0 & \overline{\omega} & \omega & \overline{\omega} \\ 0 & 1 & 0 & 0 & \omega & 1 & 0 & 1 & \overline{\omega} \\ 0 & 0 & 1 & 0 & \omega & 0 & 1 & \overline{\omega} & 1 \\ \omega & \omega & 0 & 0 & 1 & 0 & 0 & \overline{\omega} & \omega \\ \omega & 0 & \omega & 0 & \omega & 0 & 0 & 0 & \overline{\omega} \\ \omega & 0 & 0 & \omega & 1 & \omega & 1 & \omega & 1 \end{bmatrix}.$$

- n=11,12:

$$QC\_11 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ \overline{\omega} & 1 & \omega & 0 & 0 & 0 & \omega & \omega & \omega & \overline{\omega} & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ \omega & \omega & \omega & \omega & \omega & \omega & 0 & 0 & 0 & 0 & 0 \\ \overline{\omega} & 1 & \omega & 1 & \omega & \overline{\omega} & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \omega & \overline{\omega} & 1 & 0 & 0 & 0 & \omega & \overline{\omega} \\ 1 & \overline{\omega} & \omega & 0 & 0 & 0 & 1 & \overline{\omega} & \omega & 0 & 0 \\ \omega & 1 & \overline{\omega} & 0 & 0 & 0 & \omega & 1 & \overline{\omega} & 0 & 0 \\ 0 & 0 & 0 & 1 & \overline{\omega} & \omega & \omega & \overline{\omega} & 1 & 0 & 0 \\ 0 & 0 & 0 & \omega & 1 & \overline{\omega} & 1 & \omega & \overline{\omega} & 0 & 0 \\ 1 & \omega & \overline{\omega} & 0 & 0 & 0 & 0 & 0 & 0 & \overline{\omega} & \omega \end{bmatrix}, QC\_12 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega & \omega & \omega \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \omega & \omega & \omega & \omega & \omega & \omega & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \omega & \overline{\omega} & 0 & 0 & 0 & 1 & \omega & \overline{\omega} \\ 0 & 0 & 0 & \omega & \overline{\omega} & 1 & 0 & 0 & 0 & \omega & \overline{\omega} & 1 \\ 1 & \overline{\omega} & \omega & 0 & 0 & 0 & 1 & \overline{\omega} & \omega & 0 & 0 & 0 \\ \omega & 1 & \overline{\omega} & 0 & 0 & 0 & \omega & 1 & \overline{\omega} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \overline{\omega} & \omega & \omega & \overline{\omega} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \omega & 1 & \overline{\omega} & 1 & \omega & \overline{\omega} & 0 & 0 & 0 \\ 1 & \omega & \overline{\omega} & 0 & 0 & 0 & 0 & 0 & 0 & \overline{\omega} & \omega & 1 \\ \overline{\omega} & 1 & \omega & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \overline{\omega} & \omega \end{bmatrix}$$

4.2 Examples of Extremal Self-Dual Codes for lengths 10, 12 and 13

In this section we present in Table 4 codes constructed by the lengthening technique described in section 3.

For length 10, we constructed some lengthened codes of the 8 extremal self-dual codes of length 9. We found 51 extremal Type I codes and 5 extremal Type II codes (Example 3.19). For length 12, there is only one code of minimum distance 6, and Type I codes are extremal for minimum distance 5. We constructed all possible lengthened codes of the unique extremal self-dual code of length 11 and besides the dodecacode we found 7 extremal Type I codes. The same technique applied to the dodecacode lead to 4 extremal Type I codes of length 13 (Example 3.14).

There presumably exist other extremal codes for these lengths since extremal codes may also be constructed from codes with smaller distance. For instance extremal Type I codes of length 10 could also be constructed from Type I codes of length 9 and minimum distance 3.

In the following we give the generator matrices of the 5 extremal length 10 Type II codes that we found. We also list in Table 3 the lengthened codes obtained for lengths 12 and 13. We define these codes only by giving the vectors 'x', which when added to the dodecacode or QC_11 by the lengthening technique of Example 3.10, give the corresponding lengthened codes.

Code	x											
QC_12a	0	0	0	0	0	0	0	1	ω	$\overline{\omega}$	1	
QC_12b	0	0	0	0	0	0	0	ω	$\overline{\omega}$	ω	$\overline{\omega}$	
QC_12c	0	0	0	0	0	0	0	ω	$\overline{\omega}$	$\overline{\omega}$	ω	
QC_12d	0	0	0	0	0	0	1	0	$\overline{\omega}$	1	$\overline{\omega}$	
QC_12e	0	0	0	0	0	0	ω	1	0	ω	1	
QC_12f	0	0	0	0	ω	0	0	0	1	$\overline{\omega}$	ω	
QC_12g	0	0	0	0	ω	0	0	0	$\overline{\omega}$	$\overline{\omega}$	1	
QC_13a	0	0	0	0	0	0	0	0	1	1	$\overline{\omega}$	$\overline{\omega}$
QC_13b	0	0	0	0	0	0	0	1	ω	0	ω	1
QC_13c	0	0	0	0	0	0	0	1	ω	0	ω	$\overline{\omega}$
QC_13d	0	0	0	0	0	1	0	0	1	0	1	1

Table 3: Lengthened codes of lengths 12 and 13

$$QC\_10a = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & \omega & 1 & \omega & \overline{\omega} & \omega & 0 \\ 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega & 0 & 0 \\ 1 & 1 & 0 & 0 & \omega & 1 & \overline{\omega} & 0 & 1 & 0 \\ \omega & 1 & 0 & \omega & 1 & \overline{\omega} & 0 & 0 & \overline{\omega} & 0 \\ 0 & 1 & 1 & 0 & 1 & \overline{\omega} & \overline{\omega} & 1 & 0 & 0 \\ \omega & \omega & 1 & 0 & 1 & \omega & \overline{\omega} & 0 & 0 & 0 \\ 0 & \omega & \omega & 0 & 0 & \omega & 1 & 1 & \overline{\omega} & 0 \\ \omega & 0 & 0 & \omega & 0 & \omega & 0 & \omega & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & \omega & 0 & \omega & \omega & \omega \end{bmatrix}, QC\_10b = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & \omega & 1 & \omega & \overline{\omega} & \omega & 0 \\ 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega & 0 & 0 \\ 1 & 1 & 0 & 0 & \omega & 1 & \overline{\omega} & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & \overline{\omega} & \overline{\omega} & 1 & 0 & 0 \\ \omega & \omega & 1 & 0 & 1 & \omega & \overline{\omega} & 0 & 0 & 0 \\ \omega & 0 & \omega & 0 & \omega & 0 & 0 & 0 & \overline{\omega} & 0 \\ \omega & 0 & 0 & \omega & 0 & 1 & 1 & 0 & 0 & 0 \\ \omega & \omega & 0 & 0 & 1 & 0 & 0 & \overline{\omega} & \omega & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & \overline{\omega} & \omega \end{bmatrix}$$

$$QC\_10c = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & \omega & 1 & \omega & \overline{\omega} & \omega & 0 \\ 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & \overline{\omega} & \overline{\omega} & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \overline{\omega} & 1 & 0 & \omega & 0 \\ \omega & 0 & 1 & \omega & \omega & \omega & 1 & \overline{\omega} & \overline{\omega} & 0 \\ \omega & \omega & 0 & 0 & \omega & 0 & 0 & 0 & \overline{\omega} & 0 \\ \omega & 0 & \omega & 0 & 1 & 0 & 1 & \omega & \omega & 0 \\ \omega & 0 & 0 & \omega & 0 & \overline{\omega} & 1 & \omega & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \omega & \omega & \overline{\omega} & \omega \end{bmatrix}, QC\_10d = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & \omega & 1 & \omega & \overline{\omega} & \omega & 0 \\ 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & \overline{\omega} & \overline{\omega} & 0 & 0 \\ 0 & 1 & 1 & 0 & \overline{\omega} & 1 & 1 & \overline{\omega} & 0 & 0 \\ \omega & \omega & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & \omega & \omega & 0 & 0 & \omega & 1 & 1 & \overline{\omega} & 0 \\ 0 & 0 & \omega & \omega & \overline{\omega} & 0 & \omega & \overline{\omega} & \overline{\omega} & 0 \\ \omega & 0 & 0 & \omega & \omega & \omega & \overline{\omega} & 1 & \omega & 1 \\ 0 & 0 & 0 & 0 & 0 & \omega & 0 & \omega & \omega & \omega \end{bmatrix}$$

$$QC\_10e = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & \omega & 1 & \omega & \overline{\omega} & \omega & 0 \\ 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & \overline{\omega} & \overline{\omega} & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & \omega & \omega & 0 \\ \omega & \omega & 1 & 0 & \overline{\omega} & 0 & 1 & 0 & \overline{\omega} & 0 \\ 0 & \omega & 0 & \omega & 0 & \omega & 1 & 1 & \overline{\omega} & 0 \\ \omega & 0 & \omega & 0 & \omega & 0 & 0 & 0 & \overline{\omega} & 0 \\ \omega & 0 & 0 & \omega & 1 & \omega & 1 & \omega & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & \omega & 0 & \omega \end{bmatrix}$$

5 Connections with Binary Codes

Höhn [8] presents two constructions that begin with a self-dual additive $(n, 2^n)$ code and produce a self-dual binary code of length $4n$. We describe these two (Constructions A

Code	n	d	$\lvert Aut(C)\rvert$	A_0	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}
QC_10a	10	4	48	1	30	0	300	0	585	0	108			
QC_10b	10	4	32	1	30	0	300	0	585	0	108			
QC_10c	10	4	1440	1	30	0	300	0	585	0	108			
QC_10d	10	4	256	1	30	0	300	0	585	0	108			
QC_10e	10	4	18	1	30	0	300	0	585	0	108			
QC_12a	12	5	2	1	0	48	188	432	765	1040	972	528	122	
QC_12b	12	5	4	1	0	40	212	424	725	1080	980	504	130	
QC_12c	12	5	2	1	0	48	188	432	765	1040	972	528	122	
QC_12d	12	5	4	1	0	40	212	424	725	1080	980	504	130	
QC_12e	12	5	6	1	0	40	212	424	725	1080	980	504	130	
QC_12f	12	5	4	1	0	40	212	424	725	1080	980	504	130	
QC_12g	12	5	1	1	0	48	188	432	765	1040	972	528	122	
QC_13a	13	5	8	1	0	15	236	356	1197	1530	2012	1956	650	239
QC_13b	13	5	24	1	0	15	236	356	1197	1530	2012	1956	650	239
QC_13c	13	5	6	1	0	15	236	356	1197	1530	2012	1956	650	239
QC_13d	13	5	2	1	0	15	236	356	1197	1530	2012	1956	650	239

Table 4: Examples of Extremal Self-dual Codes for Lengths 10, 12 and 13

and B) and add a third (C). If $\mathcal{C}$ is a self-dual additive $(n, 2^n)$ code, let $\widehat{\mathcal{C}}$ be the binary linear $[4n, n]$ code obtained from $\mathcal{C}$ by replacing each $GF(4)$ component by a 4-tuple in $GF(2)^4$ as follows: $0 \to 0000$, $1 \to 0110$, $\omega \to 1010$, $\overline{\omega} \to 1100$. Let d_4 be the $[4,1]$ binary linear code $\{0000, 1111\}$. Finally let $(d_4^n)_0$ be the $[4n, n-1]$ binary linear code consisting of all codewords of weights divisible by 8 from the $[4n, n]$ code d_4^n.

Construction A: $\rho_A(\mathcal{C}) = \widehat{\mathcal{C}} + d_4^n$

Construction B: Assume that n is even. $\rho_B(\mathcal{C}) = \widehat{\mathcal{C}} + (d_4^n)_0 + e_B$ where e_B is the $[4n, 1]$ code generated by

$$\begin{cases} 0001\ 0001\ \cdots\ 0001 & \text{if } n \equiv 0 \pmod 4 \\ 0001\ 0001\ \cdots\ 1110 & \text{if } n \equiv 2 \pmod 4 \end{cases}.$$

Construction C: Assume that n is even. $\rho_C(\mathcal{C}) = \widehat{\mathcal{C}} + (d_4^n)_0 + e_C$ where e_C is the $[4n, 1]$ code generated by

$$\begin{cases} 0001\ 0001\ \cdots\ 1110 & \text{if } n \equiv 0 \pmod 4 \\ 0001\ 0001\ \cdots\ 0001 & \text{if } n \equiv 2 \pmod 4 \end{cases}.$$

The next result is a straightforward extension of Lemma 2 of [8].

Theorem 5.1 *Let $\mathcal{C}$ be a self-dual additive code. The following hold.*

(i) *$\widehat{\mathcal{C}}$ is a self-orthogonal code.*

(ii) *$\rho_A(\mathcal{C})$, $\rho_B(\mathcal{C})$, and $\rho_C(\mathcal{C})$ are self-dual binary codes.*

(iii) *If $\mathcal{C}$ is Type II, $\rho_A(\mathcal{C})$ and $\rho_B(\mathcal{C})$ are doubly-even whereas if $\mathcal{C}$ is Type I, $\rho_A(\mathcal{C})$ and $\rho_B(\mathcal{C})$ are singly-even.*

(iv) *$\rho_C(\mathcal{C})$ is singly-even.*

$\mathcal{C}$	$\lvert\mathrm{Aut}(\mathcal{C})\rvert$	$\lvert\mathrm{Aut}(\rho_B(\mathcal{C}))\rvert$	$\lvert\mathrm{Aut}(\rho_C(\mathcal{C}))\rvert$
$\mathcal{G}_6$	$2^4 \cdot 3^3 \cdot 5$	$2^{10} \cdot 3^3 \cdot 5 \cdot 7 \cdot 11 \cdot 23$	$2^{10} \cdot 3^3 \cdot 5$
$\mathcal{C}_1$	$2^7 \cdot 3^2$	$2^{15} \cdot 3^2 \cdot 5 \cdot 7$	$2^{15} \cdot 3^2$
$\mathcal{C}_2$	$2^4 \cdot 3 \cdot 7$	$2^{12} \cdot 3 \cdot 7$	$2^{12} \cdot 3 \cdot 7$
$\mathcal{C}_3$	$2^7 \cdot 3^2 \cdot 7$	$2^{15} \cdot 3^2 \cdot 5 \cdot 7 \cdot 31$	$2^{15} \cdot 3^2 \cdot 7$

Table 5: Some Automorphism Group Orders

We give a proof of the following, which we have alluded to throughout the paper and was first proved in [8] in a more complicated manner.

Theorem 5.2 *A self-dual additive Type II code of odd length n cannot exist.*

Proof: If $\rho_A(\mathcal{C})$ is a binary self-dual doubly-even code of length $4n$, then n is even as $4n$ must be a multiple of 8. The result follows by Theorem 5.1(iii). □

Example 5.3 Let $\mathcal{G}_6$ be the $(6, 2^6, 4)$ hexacode given in Example 1.1. We verified that $\rho_A(\mathcal{G}_6)$ is the $[24, 12, 8]$ binary Golay code and $\rho_C(\mathcal{G}_6)$ is the $[24, 12, 6]$ *odd Golay code*. It is interesting to note that beginning with the unique extremal self-dual additive code of length 6, we obtain the unique extremal self-dual doubly-even and singly-even binary codes of length 24 using constructions A and C. Note also by Table 5 that $\rho_C(\mathcal{G}_6)$ has automorphism group of size 2^6 times that of $\mathcal{G}_6$.

Example 5.4 By Example 3.2 there are exactly three $(8, 2^8, 4)$ Type II codes $\mathcal{C}_i (i = 1, 2, 3)$ with generator matrices G_i respectively QC_8c, QC_8d, and QC_8e.

Applying Construction B to each of these codes produces three of the five inequivalent self-dual doubly-even binary $[32, 16, 8]$ codes [4]. The codes $\rho_B(\mathcal{C}_i)$ are respectively C83 (or $2g_{16}$), C84 (or $8f_4$), and C82 (or r_{32}) in the notation of [4]. Notice that the extended quadratic residue code is not in this list but the second order Reed–Muller code is. Applying Construction C to each produces all three of the inequivalent self-dual singly-even binary $[32, 16, 8]$ codes [6]. Analogous to Example 5.3, $\rho_C(\mathcal{C}_i)$ has automorphism group of size 2^8 times that of $\mathcal{C}_i$ as shown in Table 5.

References

[1] C. Bachoc, "Harmonic weight enumerators of nonbinary codes and MacWilliams identities," preprint, 1999.

[2] W. Bosma and J. Cannon, *Handbook of Magma functions*, Sydney, 1995.

[3] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, "Quantum error correction via codes over $GF(4)$," *IEEE Trans. Inform. Theory* **IT-44** (1998), 1369–1387.

[4] J. H. Conway and V. Pless, "On the enumeration of self-dual codes," *J. Comb. Theory* **28A** (1980), 26–53.

[5] J. H. Conway, V. Pless, and N. J. A. Sloane, "Self-dual codes over $GF(3)$ and $GF(4)$ of length not exceeding 16," *IEEE Trans. Inform. Theory* **IT-25** (1979), 312–322.

[6] J. H. Conway and N. J. A. Sloane, "A new upper bound on the minimal distance of self-dual codes," *IEEE Trans. Inform. Theory* **IT-36** (1990), 1319–1333.

[7] J. Fields, P. Gaborit, C. Huffman and V. Pless, "On the classification of extremal even formally self-dual codes of length 20 and 22", preprint, 1998.

[8] G. Höhn, "Self-dual codes over the Kleinian four group," preprint, 1996.

[9] V. Pless, *Introduction to the Theory of Error-Correcting Codes*, 3rd ed. New York: J. Wiley & Sons, 1998.

[10] E. M. Rains and N. J. A. Sloane, "Self-dual codes," in *Handbook of Coding Theory*, ed. V. S. Pless and W. C. Huffman. Amsterdam: Elsevier, 1998, pp. 177–294.