On Determination of the Weight Distribution of Binary (168, 84, 24) Extended Quadratic Residue Code

Wen-Ku Su
Department of Information Engineering
I-Shou University
Kaohsiung County 84001, Taiwan, R.O.C.
Email: d9403003@stmail.isu.edu.tw

Chong-Dao Lee
Department of Communication Engineering
I-Shou University
Kaohsiung County 84001, Taiwan, R.O.C.
Email: chongdao@isu.edu.tw

Tsung-Ching Lin, Trieu-Kien Truong
Department of Information Engineering
I-Shou University
Kaohsiung County 84001, Taiwan, R.O.C.
Email: joe@isu.edu.tw; truong@isu.edu.tw

Yaotsu Chang
Department of Applied Mathematics
I-Shou University
Kaohsiung County 84001, Taiwan, R.O.C.
Email: ytchang@isu.edu.tw

Abstract—This paper proposes a novel scheme which consists of a weight-counting algorithm, the combinatorial designs of the Assmus-Mattson theorem, and the weight polynomial of Gleason’s theorem to determine the weight distributions of binary extended quadratic residue codes. As a consequence, the weight distribution of binary (168, 84, 24) extended quadratic residue code is given.

I. INTRODUCTION

Let \( c = c_0c_1 \cdots c_{n-1} \) be a codeword of a code \( C \) with length \( n \). The weight of \( c \) represents the number of nonzero terms in \( c \). Let \( A_i \) denote the number of codewords of weight \( i \) in \( C \). The sequence \( A_0, A_1, \ldots, A_n \) is called the weight distribution of the code \( C \). The weight enumerator of \( C \) is the polynomial

\[
W(x, y) = \sum_{i=0}^{n} A_i x^{n-i} y^i.
\]

(1)

Quadratic residue (QR) codes are good cyclic codes with high error-correcting capacity and code rate greater than or equal to 1/2. However, it is difficult to design a practical decoder and to determine the weight distributions of binary QR codes. This research is focused on the second problem, the determination of weight distributions of binary QR codes. The reason is that the weight distribution is a key factor to evaluate the error detection and error correction performances in a reliable communication. In the past, many excellent studies of finding weight distributions for QR codes or its extended codes were presented in [1-7].

In this paper, a new scheme to determine the weight distributions of binary EQR codes is proposed. The procedure consists of three steps. First, a weight-counting algorithm presented in Section 3 is used to calculate the numbers of codewords whose weights are even and the values of three fixed positions are all one. Next, these numbers can determine part of the weight distribution of an EQR code based on the Assmus-Mattson theorem described in Theorem 2. Finally, we use the weight polynomial of Gleason’s theorem to obtain the weight enumerator of the EQR code. Such a scheme has a lower computation over the method mentioned in [6]. As a result, the weight distribution of the binary (168, 84, 24) EQR code is determined by using this scheme.

The paper is organized as follows: Section 2 provides the definitions of various codes and gives the preliminary concepts of Gleason’s theorem and the Assmus-Mattson theorem. Next, a new scheme to determine the weight distributions of binary EQR codes is completely described in Section 3. Applying the proposed scheme, the weight distribution of the binary EQR code of length 168 is determined in Section 4. Conclusions are given in the last section of the paper.

II. PRELIMINARIES

Let \( n \) be a prime number of the form \( n \equiv \pm 1 \pmod{8} \). A binary quadratic residue code \( C \) of length \( n \) is an \( (n, (n+1)/2) \) cyclic code with a generator polynomial \( g(x) = \prod_{i \in Q} (x - \beta^i) \) or \( h(x) = \prod_{i \in N} (x - \beta^i) \), where \( Q = \{i | i \equiv j^2 \pmod{n} \text{ for } 1 \leq j \leq n - 1 \} \) (resp., \( N \)) is the collection of all nonzero quadratic residues (resp., non-quadratic residues) modulo \( n \) and \( \beta \) is a primitive \( n \)th root of unity in \( GF(2^m) \), where \( m \) is the smallest positive integer such that \( n \mid 2^m - 1 \). An \( (n, (n+1)/2, d) \) QR code, where \( d \) is odd, can be extended to an \( (n+1, (n+1)/2, d+1) \) extended quadratic residue code whose codewords are obtained by adjoining a parity-check bit to a fixed position of every codeword of \( C \).

Most of the following terms and notations can also be found in the book of Huffman and Pless [8]. The dual code \( C^\perp \) of a binary linear code \( C \) having length \( n \) is defined to
be \( C⊥ = \{x ∈ GF(2)^n | x · c = 0 \text{ for all } c ∈ C\} \), where 
\( x · c = x_0c_0 + \cdots + x_{n−1}c_{n−1} \mod 2 \) if \( x = (x_0, \ldots, x_{n−1}) \) 
and \( c = (c_0, \ldots, c_{n−1}) \). An \((n, n/2)\) code \( C\) is said to be \textit{self-dual} if \( C = C⊥ \). And a \textit{doubly-even} self-dual code is a self-dual code in which all weights are divisible by 4. It is well known that in the binary case all EQR codes with lengths multiples of 8 are doubly-even self-dual. In 1971, Gleason [9] showed that the weight enumerator of a binary doubly-even self-dual code is a sum of products of the two polynomials \( x^8 + 14x^4 + y^8 \) and \( x^4y^4(x^4 − y^4)^4 \). To simplify the expression of weight enumerator, we set \( x = 1 \) for the binary code. The weight enumerator of doubly-even self-dual codes is listed in the following.

\[
W(y) = \sum_{j=0}^{\lfloor \frac{n+1}{2} \rfloor} a_j (1 + 14 y^4 + y^8)^{\lfloor \frac{n+1}{2} \rfloor - 3j} (y^4 (1 − y^4)^4)^j.
\]

(2)

Since every self-dual code contains the zero vector among its codewords, the coefficient \( a_0 \) is equal to 1. To determine the weight distribution of a doubly-even self-dual code, it is sufficient to determine the coefficients \( a_j \) for \( 1 ≤ j ≤ \lfloor (n+1)/24 \rfloor \). The formula (2) will be used in the third step of the proposed scheme.

Next, a brief review of [10] about \( t \)-designs is given. Let \( X \) be a set of \( v \) elements, called points. For \( k < v \), let \( B \) be a collection of distinct \( k \)-subsets of \( X \), called blocks. The pair \((X, B)\) is called a \( t \)-(\( v, k, \lambda \)) design if every \( t \)-subset of \( X \) is contained in exactly \( \lambda \) blocks in \( B \). For two disjoint subsets \( I \) and \( J \) of \( X \) with \(|I| = i \) and \(|J| = j \), let \( \lambda^t_{ij} \) be the number of blocks in \( B \) which contain \( I \) but disjoint from \( J \).

\textit{Theorem 1:} [8, p. 295, Theorem 8.2.1] Let \((X, B)\) be a \( t \)-(\( v, k, \lambda \)) design. Then the number \( \lambda^t_{ij} \) is independent of the choice of \( I \) and \( J \) if \( i + j ≤ t \). More precisely, one has

\[
\lambda^t_{ij} = \lambda \left( \frac{v-i-j}{k-i} \right) \left( \frac{v-t}{k-t} \right) \cdot \lambda^t_{ij}.
\]

(3)

\textit{Theorem 2:} [11, p. 137, Theorem 4.1] For all \( n + 1 ≡ 0 \) (mod 8), the codewords of every even weight of an EQR code form a 3-design \((X, B)\).

Theorems 1 and 2 will be used at the second step of the proposed scheme.

III. A NOVEL SCHEME

To specify the proposed scheme, a weight-counting algorithm and its notations are introduced here. Recall that \( E \) is a binary \((2k, k)\) EQR code with the generator matrix \( G \). Let \( G_1 \) and \( G_2 \) shown below be the matrices obtained from \( G \) by elementary row operations and shift properties,

\[
G_1 = \begin{pmatrix}
0 & I_3 & S & Y_1 \\
I_{k-3} & 0 & U & Y_2 \\
\end{pmatrix}
\quad \text{and} \quad
G_2 = \begin{pmatrix}
T & I_3 & 0 & Z_1 \\
0 & V & I_{k-3} & Z_2 \\
\end{pmatrix},
\]

where \( I_m \) denotes the identity matrix of size \( m \), both \( U \) and \( V \) denote the square matrices of size \( k-3 \), and the submatrices \( Y_1 \) and \( Z_1 \) (resp., \( Y_2 \) and \( Z_2 \)) have the same size \( 3 × 3 \) (resp., \( k-3 × 3 \)). Then, \( G_1 \) and \( G_2 \) are generator matrices of \( E \).

For \( 1 ≤ i ≤ k - 3 \), let \( U_i \) (resp., \( V_i \)) denote the \( i \)-th row of the submatrix \( U \) of \( G_1 \) (resp., \( V \) of \( G_2 \)). Also, let \( G' = [I_{k-3}]0[U]Y_2 \) and \( G'' = [V]0[I_{k-3}]Z_2 \) be the submatrices of \( G_1 \) and \( G_2 \), respectively, and let \( L_1 \) (resp., \( L_2 \)) be the codeword which is the sum of the first two rows of \( G_1 \) (resp., \( G_2 \), \( S \), \( T \)). Finally, let \( wt(Ω) \) denote the weight of a vector \( Ω \), i.e., the number of nonzero elements in \( Ω \). To determine the number of codewords \( c = c_0 \cdots c_{n-1}c_n \) whose \( c_{k-3} = c_k-c_2 = c_k-1 = 1 \) and \( wt(c) = 2j \), we count the following two numbers:

(i) for \( 1 ≤ w ≤ j - 2 \), the number of codewords \( c \) of weight \( 2j \) which are the linear combinations of \( w \) rows of \( G' \) with \( wt(U_{i_1} + U_{i_2} + \cdots + U_{i_w} + S_1) ≥ w \):

\[
c = G'_{i_1} + G'_{i_2} + \cdots + G'_{i_w} + L_1,
\]

(4)

(ii) for \( 1 ≤ w ≤ j - 2 \), the number of codewords \( c \) of weight \( 2j \) which are the linear combinations of \( w \) rows of \( G'' \) with \( wt(V_{i_1} + V_{i_2} + \cdots + V_{i_w} + T_1) > w \):

\[
c = G''_{i_1} + G''_{i_2} + \cdots + G''_{i_w} + L_2,
\]

(5)

where \( 1 ≤ i_1 < i_2 < \cdots < i_w ≤ k - 3 \).

All combinations of \( w \) rows with indices \( i_1, i_2, \ldots, i_w \) from those \((k - 3)\) rows of the matrices \( G' \) and \( G'' \), respectively, are determined by the revolving door algorithm which is introduced in the book of Nijenhuis and Wilf [12, p. 28]. In this algorithm, only one row is exchanged for each neighbor combination of \( w \) rows with indices \( i_1, i_2, \ldots, i_w \).

Now, we are ready to give a description of the main result of this paper. The algorithm to determine the weight enumerator of a binary EQR code with length \( 2k \equiv 0 \) (mod 8) consists of three steps. These three steps are described as follows:

\textbf{Step 1:} Count the number \( \lambda_{4j} \) of codewords of weight \( 4j \) from the algorithm mentioned above, where \((d + 1)/4 \leq j \leq \lfloor k/12 \rfloor\), of a doubly-even self-dual and EQR code.

\textbf{Step 2:} Determine part of weight distribution and the corresponding coefficients of (2). That is, compute

\[
A_{4j} = \lambda_{4j} \begin{pmatrix}
\frac{2k}{4j} \\
\frac{2k-3}{4j} \\
\frac{4j-3}{3}
\end{pmatrix}.
\]

(6)

by \( 3-(2k, 4j, \lambda_{4j}) \) design and determine \( a_j \) of (2) by the known \( A_{4j} \), where \((d + 1)/4 \leq j \leq \lfloor k/12 \rfloor\).

\textbf{Step 3:} Use the weight polynomial of Gleason’s theorem, i.e., (2), to determine the weight enumerator of a doubly-even self-dual and EQR code.

For the determination of the weight enumerator of a \((2k, k)\) EQR code, the proposed scheme is more efficient than the method mentioned in [6]. For example, in [6] to determine the number of codewords of weight \( 2j \), \( \binom{k}{j} + 2 \sum_{i=1}^{j-1} \binom{k-3}{i-1} \) combinations have to be calculated. In this paper only \( 2 \sum_{i=1}^{j-2} \binom{k-3}{i-1} \) combinations are necessary if \( 2k \equiv 0 \).
By using this new scheme, the weight distribution of the (168, 84, 24) EQR code is determined and will be illustrated in the next section.

IV. NUMERICAL RESULTS

Let $E$ be the (168, 84, 24) EQR code. Since the code length 168 is a multiple of 8, $E$ is doubly-even self-dual and the weight enumerator of this code thus satisfies (2). To determine the weight enumerator of $E$, it suffices to know the coefficients $a_3$, where $0 \leq j \leq 7$, of (2). Since the minimum distance of $E$ is 24, it is easy to see that part of the weight distribution of $E$ are $A_0 = 1$ and $A_4 = A_8 = A_{12} = A_{16} = A_{20} = 0$. These data determine the first six coefficients of (2), say $a_0 = 1$, $a_1 = -294$, $a_2 = 31731$, $a_3 = -537886$, $a_4 = 32840241$, and $a_5 = -25947110$. The only two unknowns in (2) are the coefficients $a_6$ and $a_7$.

Next, the scheme given in Section 3 is used for the determination of the weight enumerator of this code. The weight-counting algorithm described in the previous section is utilized to calculate the number $\lambda_{5j}$, where $0 \leq j \leq 7$. Programs written in C++ language were run on Pentium IV PCs to obtain $\lambda_{24} = 2024$ and $\lambda_{28} = 76518$. According to (6), the numbers of codewords of weights 24 and 28 in $E$ are $A_{24} = 776216$ and $A_{28} = 18130188$, respectively. Some MAPLE programs were written to compute (2) and $a_0 = 483796992$ and $a_7 = -52069620$ are thus obtained.

Finally, the weight enumerator of the (168, 84, 24) EQR code is determined by substituting the known coefficients $a_0$ through $a_7$ mentioned above into (2). Because the weight distribution is symmetric; that is, $A_i = A_{n-i}$ for $0 \leq i \leq n$, only the first half of weight distribution of this code is listed in Table 1. Additionally, the weight enumerator of the (167, 84, 23) QR code is also listed.

V. CONCLUSIONS

In this paper, the proposed scheme is more efficient than the method mentioned in [6]. As a consequence, this scheme determines the weight distribution of the (168, 84, 24) EQR code.

ACKNOWLEDGMENT

This work was supported by National Science Council, R.O.C., under Grants NSC95-2221-E-214-042 and NSC97-2218-E-214-002.

REFERENCES


TABLE I

<table>
<thead>
<tr>
<th>Weight</th>
<th>(167, 84, 23) QR Code</th>
<th>(168, 84, 24) EQR Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>23</td>
<td>110888</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>665328</td>
<td>776216</td>
</tr>
<tr>
<td>27</td>
<td>302168</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>15108490</td>
<td>18130188</td>
</tr>
<tr>
<td>31</td>
<td>1057206192</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>4493126316</td>
<td>550332508</td>
</tr>
<tr>
<td>35</td>
<td>268132007827</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>983150694636</td>
<td>1251282702264</td>
</tr>
<tr>
<td>39</td>
<td>39540857275985</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>126530743283152</td>
<td>166071600559137</td>
</tr>
<tr>
<td>43</td>
<td>3417107288264670</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>9630029630564070</td>
<td>13047136918827840</td>
</tr>
<tr>
<td>47</td>
<td>179728153597349776</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>449320388493374440</td>
<td>629048534890724162</td>
</tr>
<tr>
<td>51</td>
<td>5907921405841099432</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>1317920928954805656</td>
<td>19087130695706615088</td>
</tr>
<tr>
<td>55</td>
<td>124033230083117023704</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>248066460166234047408</td>
<td>3720996024931071112</td>
</tr>
<tr>
<td>59</td>
<td>16926041140555359010</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>304668740538099658216</td>
<td>473929151949555024522</td>
</tr>
<tr>
<td>63</td>
<td>152280606333676763990128</td>
<td>225696677721889507184</td>
</tr>
<tr>
<td>64</td>
<td>247546073042226164839589</td>
<td>3997367337359030474086</td>
</tr>
<tr>
<td>67</td>
<td>9135341717529060648884</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>1343432605518903010100</td>
<td>225696677721889507184</td>
</tr>
<tr>
<td>71</td>
<td>36867476906651174659040</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>491566437955348995398672</td>
<td>860241108921860741947667</td>
</tr>
<tr>
<td>75</td>
<td>1007629118755817710057564</td>
<td></td>
</tr>
<tr>
<td>76</td>
<td>121976514089674070669782</td>
<td>222739068356459178012742</td>
</tr>
<tr>
<td>79</td>
<td>1873856945935044844028880</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>2061246240528549328431768</td>
<td>3935099586463594172460648</td>
</tr>
<tr>
<td>83</td>
<td>2377873706297857672048688</td>
<td></td>
</tr>
<tr>
<td>84</td>
<td>2377873706297857672048688</td>
<td>475574741259571354419376</td>
</tr>
</tbody>
</table>