On Decoding of the (89, 45, 17) Quadratic Residue Code

Wen-Ku Su, Pei-Yu Shih, Tsung-Ching Lin, and Trieu-Kien Truong

Department of Information Engineering,
I-Shou University
Kaohshiung County 84001, Taiwan, R.O.C.
E-Mail: d9403003@stmail.isu.edu.tw; d9203005@stmail.isu.edu.tw; joe@isu.edu.tw; truong@isu.edu.tw

***Abstract*—In this paper, a classical decoder for the (89, 45, 17) binary quadratic residue code, the last one not decoded yet of length less than 100, is proposed. It was also verified for all error patterns within the error-correcting capacity of the code without checking all error patterns by computer simulations.**

***Index Terms*—Inverse-free Berlekamp-Massey algorithm, quadratic residues codes, error-locator polynomial, primary unknown syndromes.**

I. Introduction

Prange [1] introduced the class of quadratic residue (QR) codes, a nice family of cyclic codes. These QR codes have code rates greater than or equal to 1/2 and generally have large minimum distances so that most of the known QR codes are the best-known codes. Among them, the notable (23, 12, 7) QR code, in fact, is also called the binary Golay code. It is well known that there are eleven binary QR codes with code length less than 100, namely, 7, 17, 23, 31, 41, 47, 71, 73, 79, 89, and 97. Different algebraic decoding algorithms for these QR codes have been proposed [2]-[8] but the algebraic decoding scheme for the (89, 45, 17) QR code is not available in the literature. Such a (89, 45, 17) QR code, which has the capability to correct up to eight errors, can be constructed in a small field $GF(2^{11})$. Therefore, this code is considered to be one of the best in the family of the binary quadratic residue codes.

In the past decades, the methods used most often to decode binary QR codes are the Sylvester resultant [4], [5] or Gröbner basis methods [9]. These methods can be used to solve the Newton identities that are nonlinear and multivariate equations of high degree. It becomes very difficult when the weight of the occurred error becomes large. As a result, it is not practical for software implementation.

Recently, using the inverse-free Berlekamp-Massey (BM) algorithm [10], an algebraic decoding algorithm for QR codes [8] has been applied to determine the error-locator polynomial. These facts also lead to design the algebraic decoders for many other binary QR codes of lengths up to 113 except for the QR codes of lengths 31, 73, and 89. In this paper, the previous algebraic decoding algorithms can be modified to decode the (89, 45, 17) QR code.

II. The Terminology and Background of the QR Codes

Let n be a prime number of the form $n \equiv \pm 1 \pmod 8$. A binary QR code of length n is an $(n, (n+1)/2)$ cyclic code with the generator polynomial $g(x) = \prod_{i \in Q_n} (x - \beta^i)$, where Q_n is the collection of all nonzero quadratic residues modulo n, and β is a primitive nth root of unity in the finite field $GF(2^m)$ with m, the smallest positive integer such that $n \mid 2^m - 1$.

For an (n, k, d) QR code with minimum distance d, an error pattern is said to be correctable if its weight is less than or equal to the error-correcting capacity $t = \lfloor (d-1)/2 \rfloor$, where $\lfloor x \rfloor$ denotes the greatest integer less than or equal to x. Now, let the codeword $c(x) = c_0 + c_1 x + \cdots + c_{n-1} x^{n-1}$, which is a multiple of $g(x)$, be transmitted through a noisy channel. Also, let $e(x) = e_0 + e_1 x + \cdots + e_{n-1} x^{n-1}$ be the error pattern occurred. Then the received word has the form $r(x) = c(x) + e(x)$, where $r(x) = r_0 + r_1 x + \cdots + r_{n-1} x^{n-1}$.

The set of known syndromes obtained by evaluating $r(x)$ at the roots of $g(x)$ is given by

$$S_i = r(\beta^i) = e(\beta^i), \ i \in Q_n. \tag{1}$$

Suppose that there are v errors occurred in the received word $r(x)$. Then the error pattern has v nonzero terms, namely, $e(x) = x^{l_1} + x^{l_2} + \cdots + x^{l_v}$, where $0 \le l_1 < l_2 < \cdots < l_v \le n-1$. The syndrome S_i can be written as $S_i = X_1^i + X_2^i + \cdots + X_v^i$, where $X_j = \beta^{l_j}$ for $1 \le j \le v$ are said to be the error locators. If i is not found in the set Q_n, the syndrome S_i is, what is called, the unknown syndrome.

For binary QR codes, there is an obvious relation between syndromes, namely, $S_{2i} = S_i^2$, with indices modulo n. It is well known that the generator polynomial of the (89, 45, 17) QR code is reducible over $GF(2)$, and all the known syndromes

TABLE I

VALUES OF INDICES t_i FOR THE SYNDROME IDENTIFIES $S_{t_i} = S_j^{2^i}$

i	1	2	3	4	5	6	7	8	9	10
$j=1$	2	4	8	16	32	64	39	78	67	45
$j=5$	10	20	40	80	71	53	17	34	68	47
$j=9$	18	36	72	55	21	42	84	79	69	49
$j=11$	22	44	88	87	85	81	73	57	25	50
$j=3$	6	12	24	48	7	14	28	56	23	46
$j=13$	26	52	15	30	60	31	62	35	70	51
$j=19$	38	76	63	37	74	59	29	58	27	54
$j=33$	66	43	86	83	77	65	41	82	75	61

(resp., unknown syndromes) can be expressed as some powers of S_1, S_5, S_9, and S_{11} (resp., S_3, S_{13}, S_{19}, and S_{33}). Table I shows the values of indices t_i if $S_{t_i} = S_j^{2^i}$ for the (89, 45, 17) QR code.

Assume that v errors occur in the received word. The error-locator polynomial is defined to be a polynomial of degree v; that is, $\sigma(x) = \prod_{j=1}^{v}(1+X_j x) = 1+\sum_{j=1}^{v}\sigma_j x^j$ where $\sigma_1 = X_1+\cdots+X_v$, $\sigma_2 = X_1X_2+\cdots+X_{v-1}X_v$, …, and $\sigma_v = X_1\cdots X_v$. The inverse-free BM algorithm is known to be the most efficient method for determining the error-locator polynomial. In order to use the inverse-free BM algorithm to decode the QR code up to eight errors, i.e., $t=8$, one needs to find, in sequence, the first $2t$ consecutive syndromes $S_1, S_2, \ldots, S_{2t}$. However, the only syndromes $S_1, S_2, S_4, S_5, S_8, S_9, S_{10}, S_{11}, S_{16}$ can be calculated directly from $r(x)$; the others, namely, S_3, S_6, S_7, S_{12}, S_{13}, S_{14}, S_{15} not computed directly from $r(x)$ are unknown syndromes. Evidently, these known syndromes (resp., unknown syndromes) are expressed as some powers of S_1, S_5, S_9, S_{11} (resp., S_3 and S_{13}). The values of known syndromes (resp., unknown syndromes) can thus be obtained if the values of S_1, S_5, S_9, and S_{11} (resp., S_3 and S_{13}) are determined. For this reason, S_1, S_5, S_9, and S_{11} (resp., S_3 and S_{13}) are so-called the primary known syndromes (resp., unknown syndromes) of the QR code.

Using a technique similar to that given in [8], the strategy in this paper is developed to obtain each of the needed primary unknown syndromes. All of the unknown syndromes can be calculated once the values of the primary unknown syndromes are determined. The following is a brief review of the technique mentioned in [8] for the (89, 45, 17) QR code.

Assume that v errors occur in the received word. Let $\boldsymbol{I} = \{i_1, i_2, \ldots, i_{v+1}\}$ and $\boldsymbol{J} = \{j_1, j_2, \ldots, j_{v+1}\}$ denote two subsets of {0, 1, 2,…, 88}, respectively. These index subsets can be found by an explicit use of the fast algorithm developed in [2]. Next, consider the matrix $\boldsymbol{S(I, J)}$ of size $(v+1)\times(v+1)$ given by

$$S(I,J) = \begin{bmatrix} S_{i_1+j_1} & S_{i_1+j_2} & \cdots & S_{i_1+j_{v+1}} \\ S_{i_2+j_1} & S_{i_2+j_2} & \cdots & S_{i_2+j_{v+1}} \\ \vdots & \vdots & \ddots & \vdots \\ S_{i_{v+1}+j_1} & S_{i_{v+1}+j_2} & \cdots & S_{i_{v+1}+j_{v+1}} \end{bmatrix}, \quad (2)$$

where the summation of the indices of S_i's is modulo n and the rank of $\boldsymbol{S(I, J)}$ is at most v which, in turn, implies

$$\det(\boldsymbol{S(I, J)}) = 0. \quad (3)$$

If all of the unknown syndromes among the entries of $\boldsymbol{S(I, J)}$ given in (2) can be expressed as some powers of one of the primary unknown syndromes, say S_r, and if $\det(\boldsymbol{S(I, J)})$ is a non-zero polynomial in S_r for each weight-v pattern errors, then the actual value of S_r is one of the roots of (3). The determination of the primary unknown syndromes S_3 and S_{13} of the (89, 45, 17) QR code will be shown in Section IV.

III. Mapping Relationship Between Error Patterns and Primary Known Syndromes

Let $\alpha \in GF(2^{11})$ be a root of the primitive polynomial $x^{11}+x^2+1$. Then α generates the multiplicative group of nonzero elements in the finite field $GF(2^{11})$. Evidently, $\beta = \alpha^{23}$ is a primitive 89th root of unity because of $(2^{11}-1)/89 = 23$. Let Q_{89} be the set of quadratic residues. The generator polynomial factors into four binary irreducible polynomials as follows:

$$g(x) = \prod_{i\in Q_{89}}(x-\beta^i) = g_1(x)\, g_5(x)\, g_9(x)\, g_{11}(x) \quad (4)$$

where $g_1(x)$, $g_5(x)$, $g_9(x)$, and $g_{11}(x)$ are the minimal polynomials of β, β^5, β^9, and β^{11} with coefficients in $GF(2)$, respectively.

Based on the generator polynomial of the (89, 45, 17) QR code, the mapping relationship between each error pattern with a weight less than or equal to eight and its primary known syndromes S_1, S_5, S_9 and S_{11} is given in the following:

*L*et $(S_1', S_5', S_9', S_{11}')$ and $(S_1'', S_5'', S_9'', S_{11}'')$ be the ordered 4-tuples of the primary known syndromes of error patterns $e_1(x)$ and $e_2(x)$ with weights less than or equal to 8, respectively.

Proposition 1: Let $e_1(x)=0$, i. e., the ordered 4-tuple $(S_1', S_5', S_9', S_{11}')=(0, 0, 0, 0)$ and $e_2(x)\neq e_1(x)$ with their weights less than or equal to eight. Then there exists at least one nonzero component of $(S_1'', S_5'', S_9'', S_{11}'')$.

Proposition 2: Let the ordered 4-tuple (S_1, S_5, S_9, S_{11}) of the received word $r(x)=c(x)+e(x)$ be calculated directly from the received word and the ordered 4-tuple $(S_1', S_5', S_9', S_{11}')$ of the error pattern $e_1(x)$, obtained by the decoding algorithm, be computed via (1). $e_1(x)=e(x)$, i.e., $e_1(x)$ is correct if $(S_1, S_5, S_9, S_{11})=(S_1', S_5', S_9', S_{11}')$.

Detailed analysis on *Propositions 1* and *2* can be found in [11].

IV. Determination of Unknown Syndromes S_3 and S_{13} of the (89, 45, 17) QR Code

In what follows, the primary unknown syndromes S_3 and S_{13} usually needed to realize the inverse-free BM algorithm are determined. An index with parentheses is attached to the unknown syndrome "S_r" to obtain the notation "$S_r^{(v)}$", indicating that the formulae obtained are valid for the v-error case only. Furthermore, "Case va)" and "Case vb)" denote two

sub-cases for determining the primary unknown syndromes $S_3^{(v)}$ and $S_{13}^{(v)}$, respectively.

In order to show that det($\boldsymbol{S(I, J)}$) in either $S_3^{(v)}$ or $S_{13}^{(v)}$ is a nonzero polynomial for the cases $v \geq 2$, some special subsets $\boldsymbol{I}$ and $\boldsymbol{J}$ are given. It is obvious that det($\boldsymbol{S(I, J)}$) is a nonzero polynomial in $S_3^{(v)}$ (resp., $S_{13}^{(v)}$) for case 2, because the coefficient of a term of det($\boldsymbol{S(I, J)}$) in $S_3^{(v)}$ (resp., $S_{13}^{(v)}$) is equal to one. For the cases $v \geq 3$, there exist four monomials of det($\boldsymbol{S(I, J)}$) in either $S_3^{(v)}$ or $S_{13}^{(v)}$ whose coefficients can be expressed as some powers of the primary known syndromes S_1, S_5, S_9, and S_{11}. By *Proposition 1*, det($\boldsymbol{S(I, J)}$) is a nonzero polynomial in either $S_3^{(v)}$ or $S_{13}^{(v)}$ for $v \geq 3$. Then the value of either $S_3^{(v)}$ or $S_{13}^{(v)}$ is one of the roots of the equation det($\boldsymbol{S(I, J)}$) = 0. In other words, $S_3^{(v)}$ (resp., $S_{13}^{(v)}$) can be determined by using the Chien search to solve the roots of det($\boldsymbol{S(I, J)}$)=0 in $S_3^{(v)}$ (resp., $S_{13}^{(v)}$). In each odd-error case, the syndrome S_0 is always equal to one, but zero in every even-error case.

Case 0: For the zero-error case, the primary known syndromes $S_3^{(0)} = S_{13}^{(0)} = 0$.

Case 1: For the one-error case, $S_3^{(1)} = S_1^{3}$ and $S_{13}^{(1)} = S_1^{13}$.

Case 2: One endeavors to find the unknown syndromes $S_3^{(2)}$ and $S_{13}^{(2)}$ by solving the roots of the equations shown in sub-cases a) and b), respectively. Two sub-cases are to be considered as follows:

a) Let $\boldsymbol{I} = \{0, 1, 47\}$ and $\boldsymbol{J} = \{1, 6, 45\}$. From (3), the equation, det($\boldsymbol{S(I, J)}$) = 0 is given in the following:

$$S_3^{1042} + S_1^{1024} S_3^{48} + S_1 S_3^{33} + S_1^{2} S_3^{3} = 0. \quad (6)$$

Notice that the leading coefficient of (6) in S_3 equal to one; that is, the left side of (6) is a nonzero polynomial in S_3 for all error patterns. Thus, the unknown syndrome $S_3^{(2)}$ is a root of (6).

b) Let $I_2 = \{0, 15, 17\}$ and $J_2 = \{15, 34, 45\}$. From (3), the equation det($\boldsymbol{S(I, J)}$) = 0 which is expressed in terms of some powers of the unknown syndrome S_{13} is

$$S_{13}^{1064} + S_1^{1024} S_{13}^{1040} + S_5^{256} S_{13}^{144} + S_9^{1024} S_{13}^{136} + S_1^{32} S_5^{256} S_{13}^{32} + S_1^{1056} S_9^{1024} = 0 \quad (7)$$

Similarly, the coefficient of S_{13}^{1064} is equal to one. Thus, the unknown syndrome $S_{13}^{(2)}$ for the two-error case is a root of (7).

Case 3: If the number of errors is three, similarly, two sub-cases are considered as follows:

a) Let $\boldsymbol{I} = \{0, 1, 5, 47\}$ and $\boldsymbol{J} = \{1, 6, 17, 45\}$. By *Proposition 1*, there exists at least one nonzero component of the ordered 4-tuple (S_1, S_5, S_9, S_{11}) except the zero-error case. Additionally, four monomials $S_{11}^{2} S_3^{1042}$, $S_1^{64} S_3^{1028}$, $S_5^{192} S_3^{1026}$, and $S_9^{2} S_3^{5}$ are contained in det($\boldsymbol{S(I, J)}$). Consequently, $S_3^{(3)}$ can be determined by one of the roots of (3).

b) Let $\boldsymbol{I} = \{0, 15, 17, 45\}$ and $\boldsymbol{J} = \{15, 34, 45, 55\}$. Similarly, there are four monomials $S_1^{1024} S_{13}^{1568}$, $S_9^{16} S_3^{1088}$, $S_{11} S_{13}^{1064}$, and $S_5^{256} S_{13}^{672}$ which are contained in the left side of (3). By solving one of the roots of (3), one yields the value of $S_{13}^{(3)}$.

Case 4: For the four-error case, one considers the following two sub-cases:

a) Let $\boldsymbol{I} = \{0, 1, 4, 5, 47\}$ and $\boldsymbol{J} = \{1, 2, 6, 17, 45\}$. By *Proposition 1*, the left side of (3) is nonzero and $S_3^{(4)}$ is determined by solving one of the roots of (3).

b) Let $\boldsymbol{I} = \{0, 15, 17, 34, 45\}$ and $\boldsymbol{J} = \{15, 17, 34, 45, 55\}$. Again, by *Proposition 1*, det($\boldsymbol{S(I, J)}$) is a nonzero polynomial in $S_{13}^{(4)}$ for all error patterns. The unknown syndrome $S_{13}^{(4)}$ is then determined by one of the roots of (3).

Case 5: For $v = 5$, two sub-cases need to be considered as follows:

a) Let $\boldsymbol{I} = \{0, 1, 3, 4, 5, 47\}$, and $\boldsymbol{J} = \{1, 2, 3, 6, 17, 45\}$. By *Proposition 1*, det($\boldsymbol{S(I, J)}$) is a nonzero polynomial in $S_3^{(5)}$ for all error patterns. The value of $S_3^{(5)}$ is obtained by solving one of the roots of (3).

b) Let $\boldsymbol{I} = \{0, 15, 17, 34, 35, 45\}$ and $\boldsymbol{J} = \{15, 17, 34, 36, 45, 55\}$. By *Proposition 1*, det($\boldsymbol{S(I, J)}$) is a nonzero polynomial in $S_{13}^{(5)}$ for all error patterns. The unknown syndrome $S_{13}^{(5)}$ is thus determined by solving one of the roots of (3).

Case 6: For the six-error case, the following two sub-cases are considered:

a) Let $\boldsymbol{I} = \{0, 1, 3, 4, 5, 8, 47\}$, and $\boldsymbol{J} = \{1, 2, 3, 6, 17, 45, 88\}$. According to *Proposition 1*, det($\boldsymbol{S(I, J)}$) is a non-zero polynomial in $S_3^{(6)}$. Similarly, $S_3^{(6)}$ must be one of the roots of the nonzero polynomial.

b) As usual, let $\boldsymbol{I} = \{0, 15, 17, 34, 35, 45, 54\}$ and $\boldsymbol{J} = \{15, 17, 34, 36, 45, 55, 70\}$. By *Proposition 1*, det($\boldsymbol{S(I, J)}$) is a non-zero polynomial in $S_{13}^{(6)}$. Similarly, $S_{13}^{(6)}$ can be determined by one of the roots of (3).

Case 7: The seven-error case similar to the six-error case is given as follows:

a) Let $\boldsymbol{I} = \{0, 1, 3, 4, 5, 8, 19, 47\}$, and $\boldsymbol{J} = \{1, 2, 3, 6, 17, 45, 84, 88\}$. By *Proposition 1*, det($\boldsymbol{S(I, J)}$) is a nonzero polynomial in $S_3^{(7)}$. The value of $S_3^{(7)}$ is thus determined by one of the roots of (3).

b) In this sub-case, one chooses $\boldsymbol{I} = \{0, 15, 17, 34, 35, 36, 45, 54\}$, and $\boldsymbol{J} = \{15, 17, 34, 35, 36, 45, 55, 70\}$. The matrix $\boldsymbol{S(I, J)}$ obtained by (2) is used to yield (3). By *Proposition 1*, det($\boldsymbol{S(I, J)}$) is a non-zero polynomial in $S_{13}^{(7)}$. Similarly, $S_{13}^{(7)}$ is determined by one of the roots of (3).

Case 8: For the case of eight errors, $S_3^{(8)}$ and $S_{13}^{(8)}$ are determined in the following two sub-cases, respectively:

a) In this case, the pair ($\boldsymbol{I}$, $\boldsymbol{J}$)'s is used to determine $S_3^{(8)}$; that is, $\boldsymbol{I} = \{0, 1, 3, 4, 5, 8, 19, 22, 47\}$, and $\boldsymbol{J} = \{1, 2, 3, 6, 17, 45, 84, 87, 88\}$. $S_3^{(8)}$ can be determined by one of the roots of (3).

b) Let $\boldsymbol{I} = \{0, 34, 36, 53, 54, 55, 64, 74, 83\}$, and $\boldsymbol{J} = \{15, 16, 17, 26, 45, 51, 70, 85, 87\}$. By an argument similar to the seven-error case, $S_{13}^{(8)}$ can be determined by one of the roots of (3).

After determining the primary unknown syndromes S_3 and S_{13}, one now has enough syndromes to apply the inverse-free BM algorithm proposed in [8]. If an error pattern $e(x)$ is obtained from this algorithm with respect to some S_3, S_{13}, and

v-error case, recompute the values of the known syndromes S_1', S_5', S_9', and S_{11}' from $e(x)$. By *Proposition 2*, the indeed correct error pattern $e(x)$ can be obtained once the ordered 4-tuple (S_1, S_5, S_9, S_{11}) of the received word $r(x)$ equals (S_1', S_5', S_9', S_{11}') of $e(x)$.

V. Conclusion

Binary QR codes are well-known for their good behavior. However, it is very difficult to decode especially the (89, 45, 17) QR code, which is one of the best codes known with the code length n and the dimension k. In this paper, a classical decoding algorithm for the (89, 45, 17) QR codes has been verified by *Propositions 1* and *2*. As a result, all of the binary QR codes of length less than or equal to 113 have been successfully decoded.

References

[1] E. Prange, "Some cyclic error-correcting codes with simple decoding algorithms," *Air Force Cambridge Research Center-TN-58-156*, 1958.

[2] R. He, I. S. Reed, T. K. Truong, and X. Chen, "Decoding of the (47, 24, 11) quadratic residue code," *IEEE Trans. Inform. Theory*, vol. 47, pp. 1181-1186, Mar. 2001.

[3] I. S. Reed, X. Yin, T. K. Truong, and J. K. Holmes, "Decoding the (24, 12, 8) Golay code," *Proc. IEE*, vol. 137, pp. 202-206, May 1990.

[4] M. Elia, "Algebraic decoding of the (23, 12, 7) Golay code," *IEEE Trans. Inform. Theory*, vol. IT-33, pp. 150-151, Jan. 1987.

[5] I. S. Reed, X. Yin, and T. K. Truong, "Algebraic decoding of the (32, 16, 8) quadratic residue code," *IEEE Trans. Inform. Theory*, vol. 36, pp. 876-880, Jul. 1990.

[6] I. S. Reed, T. K. Truong, X. Chen, and X. Yin, "The algebraic decoding of the (41, 21, 9) quadratic residue code," *IEEE Trans. Inform. Theory*, vol. 38, pp. 974-985, May 1992.

[7] X. Chen, I. S. Reed, and T. K. Truong, "Decoding the (73, 37, 13) quadratic residue code," *Proc. IEE*, vol. 141, pp. 253-258, Sep. 1994.

[8] Yaotsu Chang, T. K. Truong, I. S. Reed, H. Y. Cheng, and C. D. Lee, "Algebraic decoding of (71, 36, 11), (79, 36, 11), and (97, 49, 15) quadratic residue codes," *IEEE Trans. Commun.*, vol. 51, pp. 1463-1473, Sep. 2003.

[9] X. Chen, I. S Reed, T. Helleseth, and T. K. Truong, "Use of Gröbner bases to decode binary cyclic codes up to the true minimum distance," *IEEE Trans. Commun.*, vol. 40, pp. 1654-1661, Sep. 1994.

[10] I. S. Reed, M. T. Shih, and T. K. Truong, "VLSI design of inverse-free Berlekamp-Massey algorithm," *Proc. IEE*, vol. 138, pp. 295-298, Sep. 1991.

[11] Trieu-Kien Truong, Pei-Yu Shih, Wen-Ku Su, Chong-Dao Lee, and Yaotsu Chang, "Algebraic decoding of the (89, 45, 17) quadratic residue code," *IEEE Trans. Inform. Theory*, to appear.