AUSTRALASIAN JOURNAL OF COMBINATORICS
Volume **37** (2007), Pages 257–264

# On the [32, 16, 8] BCH and quadratic residue codes

M. Esmaeili

*Department of Mathematical Sciences*
*Isfahan University of Technology*
*Isfahan*
*Iran*
emorteza@cc.iut.ac.ir

T. A. Gulliver

*Department of Electrical and Computer Engineering*
*University of Victoria*
*P.O. Box 3055, STN CSC*
*Victoria, B.C., V8W 3P6*
*Canada*
agullive@ece.uvic.ca

**Abstract**

In this paper we consider the properties of the [32, 16, 8] BCH and quadratic residue codes. It is shown that from the projection onto $F_4$ perspective, the two codes are very similar. In particular, the kernel of the mapping Proj from the BCH code to $F_4^8$ has dimension 8 while the dimension for the quadratic residue code is 7. Due to their simple Tanner graph-trellis structure, the soft-decision maximum-likelihood decoding complexity of these codes is shown to be low.

## 1 Introduction

The complexity of soft-decision maximum-likelihood decoding of a linear block code depends very much on the structure of the code. One approach to decoding a linear block code $C$ that contains a geometrically simple subcode is to project $C$ onto a larger field and perform two level decoding [2, 8, 9, 10, 12]. Using this method, structurally equivalent decoding algorithms were proposed for the [24, 12, 8] Golay code [2, 8, 12] by projecting it onto the quaternary [6, 3, 4] hexacode. In [15], a two level decoding algorithm was given for a Type II [32, 16, 8] self-dual code $C$ considered by the authors to be the [32, 16, 8] extended Quadratic Residue (QR) code $q_{32}$.

It has been shown in [4] that the decoding method used in [9, 12] can be expressed in the language of Tanner graphs [11] and trellis diagrams [7, 14]. This presentation of a code $C$ is called the Tanner graph-trellis representation of $C$. It consists of an acyclic Tanner graph of a subcode $C_0$ together with a trellis diagram of $C/C_0$.

Let $M_0$ and $\begin{bmatrix} M_0 \\ M_c \end{bmatrix}$ be generator matrices for $C_0$ and $C$, respectively, and let $H_0$ be a parity check matrix for $C_0$. The Tanner graph of $C_0$ is a bipartite graph with two types of nodes called parity nodes (corresponding to the rows of $H_0$), and symbol nodes (corresponding to the columns of $H_0$). A parity node $p_i$ is adjacent to a symbol node $x_j$ iff the entry $h_{ij}$ in $H_0$ is nonzero. When $C_0$ has a simple acyclic Tanner graph and the code with generator matrix $M_{PS} := M_c H_0^T$ (called the parity space), has a well-structured trellis diagram, $C$ can be decoded efficiently.

In the next section, we consider the $[32, 16, 8]$ BCH code and show that the Type II $[32, 16, 8]$ self-dual code $C$ considered in [15] is actually this code. Section 3 is devoted to the QR code $q_{32}$. A low complexity Tanner graph-trellis representation is given for this code. The two $[32, 16, 8]$ codes are then compared based on their projections onto $F_4^8$.

## 2 A projection for the $[32, 16, 8]$ BCH code

A two level decoding method was presented in [15] for the $[32, 16, 8]$ QR code. In this section we show that the code considered in [15] is actually the $[32, 16, 8]$ BCH code.

Let $\mathbf{F}_{32} = \mathbf{F}_2[x]/(1 + x^2 + x^5)$, and $\alpha$ be a root of $1 + x^2 + x^5$. Since $1 + x^2 + x^5$ is a primitive polynomial, $\alpha$ is a primitive field element in $\mathbf{F}_{32}$ and $m_i(x)$, the minimal polynomials of $\alpha^i$, $i \in \{1, 3, 5, 7\}$, are

$$\begin{aligned} m_1(x) &= 1 + x^2 + x^5, \\ m_3(x) &= 1 + x^2 + x^3 + x^4 + x^5, \\ m_5(x) &= 1 + x + x^2 + x^4 + x^5, \\ m_7(x) &= 1 + x + x^2 + x^3 + x^5. \end{aligned}$$

Therefore, the generator polynomial of the $[31, 16, 7]$ BCH code is [13]

$$g(x) = m_1(x)m_3(x)m_5(x) = 1 + x + x^2 + x^3 + x^5 + x^7 + x^8 + x^9 + x^{10} + x^{11} + x^{15}.$$

This was given in [15] as the generator polynomial of the $[31, 16, 7]$ binary quadratic residue code. Adding an overall parity check to this BCH code, we obtain the

following generator matrix for the extended $[32, 16, 8]$ binary BCH code

$$
\begin{array}{l}
11110101111100010000000000000001 \\
01111010111110001000000000000001 \\
00111101011111000100000000000001 \\
00011110101111100010000000000001 \\
00001111010111110001000000000001 \\
00000111101011111000100000000001 \\
00000011110101111100010000000001 \\
00000001111010111110001000000001 \\
00000000111101011111000100000001 \\
00000000011110101111100010000001 \\
00000000001111010111110001000001 \\
00000000000111101011111000100001 \\
00000000000011110101111100010001 \\
00000000000001111010111110001001 \\
00000000000000111101011111000101 \\
00000000000000011110101111100011
\end{array}
\tag{1}
$$

Applying the permutation

$$
\pi = \left(\begin{array}{cccccccccccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\
3 & 30 & 8 & 10 & 18 & 20 & 1 & 2 & 21 & 5 & 6 & 15 & 19 & 29 & 9 & 4 \\
\hline
17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 & 26 & 27 & 28 & 29 & 30 & 31 & 32 \\
17 & 23 & 11 & 24 & 13 & 22 & 28 & 7 & 27 & 16 & 31 & 25 & 12 & 32 & 14 & 26
\end{array}\right)
$$

to (1) gives the matrix $M$ below. Putting this into trellis oriented form results in matrix $M'$, as reported in [4].

$$
M = \begin{array}{l}
01111101010000100001100001000100 \\
10001101010000101110100001000100 \\
01001101010000100111001001001000 \\
10000100111000100111100001001000 \\
11011000100000100111000101001000 \\
11010100100010001011100001001000 \\
11011000100000101000111001001000 \\
01011100101000001010101001010000 \\
00011110001000101000101101001000 \\
00001100101010101010001101100000 \\
00010100001010110010011101001000 \\
00000000101010101010010101011010 \\
00010010100010000010011111011000 \\
00010010101110001000010001111000 \\
00010010100000011000011101110001 \\
00010010001011011000001001110010
\end{array}
\qquad
M' = \begin{array}{l}
11111111100000000000000000000000 \\
00001111111100000000000000000000 \\
00000000111111110000000000000000 \\
00000000000011111111100000000000 \\
00000000000000001111111100000000 \\
00000000000000000000001111111110 \\
00000000000000000000000011111111 \\
10001000100010001000100010001000 \\
01010000011010101100000000000000 \\
00110000010100110110000000000000 \\
00000101111110011010110000000000 \\
00000011000010101100011000000000 \\
00000000011000001100011011000000 \\
00000000001100001010001110100000 \\
00000000000001011100000000111010 \\
00000000000000011011000001011100
\end{array}
$$

This is a doubly even $[32, 16, 8]$ code and so does not contain a weight 14 codeword as claimed in [6], page 599.

Consider the following odd and even interpretations of the elements of $F_4 = \{0, 1, \omega, \bar{\omega} = \omega^2\}$:

| | | | | |
|---|---|---|---|---|
| 0 | 01 | 10 | 10 | 10 |
| 1 | 01 | 10 | 01 | 01 |
| $\omega$ | 01 | 01 | 10 | 01 |
| $\bar{\omega}$ | 01 | 01 | 01 | 10 |
| | 0 | 1 | $\omega$ | $\bar{\omega}$ |

Even Interpretations

| | | | | |
|---|---|---|---|---|
| 0 | 10 | 01 | 01 | 01 |
| 1 | 01 | 10 | 01 | 01 |
| $\omega$ | 01 | 01 | 10 | 01 |
| $\bar{\omega}$ | 01 | 01 | 01 | 10 |
| | 0 | 1 | $\omega$ | $\bar{\omega}$ |

Odd Interpretations

Using these interpretations we define a mapping Proj from $F_2^4$ to $F_4$ by

$$\text{Proj}(x_1x_2x_3x_4) := < x_1, x_2, x_3, x_4 >< 0, 1, \omega, \bar{\omega} >= x_1 0 + x_2 1 + x_3\omega + x_4\bar{\omega}.$$

This mapping can be used to project $F_2^{4m}$ onto $F_4^m$. Using the mapping Proj, the extended code given by $M'$ is projected onto the $[8, 4, 4]$ quaternary code given by the generator matrix $G_b$

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

The first eight rows of $M'$ form the kernel of this projection.

In summary, the code given by [[15], Definition 1] is neither the QR code $q_{32}$, as claimed in [15], nor one of the three type I $[32, 16, 8]$ codes, as claimed in [6]. It is in fact the extended $[32, 16, 8]$ BCH code. Thus all references to the $[32, 16, 8]$ QR code in [3, 4] should be replaced by the extended $[32, 16, 8]$ BCH code. Finally, it is worth mentioning that since the $[32, 16, 8]$ BCH code is equivalent to the $[32, 16, 8]$ Reed-Muller code, it has efficient soft [4, 9] and hard [5] maximum-likelihood decoding algorithms.

## 3 The $[32, 16, 8]$ quadratic residue code

In this section, we provide a Tanner graph-trellis representation [4] for the $[32, 16, 8]$ QR code $q_{32}$. Both the Tanner graph of the base code and the trellis structure of the associated parity space are well structured and hence the given presentation is well suited for the purpose of maximum-likelihood soft-decoding. We refer the interested reader to [4] for more details on efficient soft-decoding using a Tanner graph-trellis representation.

The QR code $q_{32}$ has the following generator matrix [1].

```
11111111000000000000000000000000
11100010111100000000000000000000
10111000110011000000000000000000
10100110010010110000000000000000
00000000000000000000000011111111
00000000000000000000111101010101
00000000000000000011001101110010
00000000000000001101011100000110
00011110111000100101000001000001
01100110010000000000010001000010
00100010011000100001000001000100
10001000100010000001010001001000
00111100010000100000000001010000
11100010011010100100010001100000
01100000011000000101010100000000
00110000100010000101011000000000
```

Applying the permutation $\pi'$, this matrix is changed to the matrix $M_{q_{32}}$ given below.

$$\pi' = \left(\begin{array}{cccccccccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\ 2 & 3 & 4 & 8 & 6 & 7 & 1 & 5 & 9 & 11 & 10 & 12 & 13 & 15 & 14 & 16 \\ \hline 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 & 26 & 27 & 28 & 29 & 30 & 31 & 32 \\ 18 & 20 & 17 & 19 & 21 & 23 & 22 & 24 & 30 & 26 & 31 & 28 & 32 & 25 & 29 & 27 \end{array}\right)$$

$$M_{q_{32}} = \left[\begin{array}{c} M_5 \\ M_c \end{array}\right] = \left[\begin{array}{cccccccc} 1111 & 1111 & 0000 & 0000 & 0000 & 0000 & 0000 & 0000 \\ 0000 & 1111 & 1111 & 0000 & 0000 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 1111 & 1111 & 1111 & 1111 & 0000 & 0000 \\ 0000 & 0000 & 0000 & 0000 & 0000 & 1111 & 1111 & 0000 \\ 0000 & 0000 & 0000 & 0000 & 0000 & 0000 & 1111 & 1111 \\ \hline 0111 & 0111 & 0111 & 0111 & 0000 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 0000 & 0000 & 0111 & 0111 & 0111 & 0111 \\ \hline 1010 & 1010 & 1010 & 1010 & 0000 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 0000 & 0000 & 1010 & 1010 & 1010 & 1010 \\ 0110 & 1001 & 0000 & 1111 & 0011 & 0110 & 0000 & 0000 \\ 0000 & 0110 & 0101 & 0110 & 0101 & 1100 & 1100 & 0000 \\ 0000 & 0011 & 1010 & 1100 & 0101 & 0000 & 0000 & 0000 \\ 0000 & 0000 & 1001 & 1001 & 0101 & 0101 & 0000 & 0000 \\ 0000 & 0000 & 0011 & 1101 & 0100 & 1010 & 0000 & 0000 \\ 0000 & 0000 & 0011 & 0011 & 1001 & 1100 & 0110 & 0011 \\ 0000 & 0000 & 0000 & 0101 & 1001 & 1010 & 0110 & 0000 \end{array}\right]$$

By the mapping Proj, this presentation of $q_{32}$ is projected onto an additive $(8, 2^9, 3)$ quaternary code, and this mapping has a 7-dimensional kernel with basis consisting of the first seven rows of matrix $M_{q_{32}}$. This is only one dimension less than the kernel of the BCH code.

The subcode $C_5$ introduced by $M_5$, the first 5 rows of $M_{q_{32}}$, has parity check matrix $H_5$

$$H_5 := I_8 \otimes [4, 3, 2] + I_2 \otimes ([1111] \otimes [1000]) + [00011000] \otimes [1000],$$

where $\otimes$ denotes the Kronecker product. The associated parity space has generator matrix $M_{PS} := M_c H_5^T$. A Tanner graph of $H_5$ together with a 3-section minimal trellis diagram of $M_{PS}$ (a Tanner graph-trellis representation), is shown in Figure 1. The nodes $p_1$, $p_2$ and $p_3$ are called the root parities. The set of four parity nodes located on the $i$th (left to right) branch of the Tanner graph are denoted by $\mathbf{p}_{bi}$.

$$M_{PS} = \left[\begin{array}{c|cccccccc} p_1p_2p_3 & \mathbf{p}_{b1} & \mathbf{p}_{b2} & \mathbf{p}_{b3} & \mathbf{p}_{b4} & \mathbf{p}_{b5} & \mathbf{p}_{b6} & \mathbf{p}_{b7} & \mathbf{p}_{b8} \\ \hline 000 & 100 & 100 & 100 & 100 & 000 & 000 & 000 & 000 \\ 010 & 111 & 111 & 111 & 111 & 000 & 000 & 000 & 000 \\ 000 & 000 & 000 & 000 & 000 & 100 & 100 & 100 & 100 \\ 010 & 000 & 000 & 000 & 000 & 111 & 111 & 111 & 111 \\ 010 & 000 & 000 & 101 & 101 & 111 & 111 & 000 & 000 \\ 111 & 000 & 000 & 010 & 011 & 110 & 111 & 000 & 000 \\ 000 & 000 & 101 & 111 & 101 & 111 & 010 & 010 & 000 \\ 000 & 001 & 001 & 001 & 001 & 101 & 010 & 000 & 000 \\ 000 & 000 & 010 & 010 & 111 & 000 & 111 & 000 & 000 \\ 000 & 000 & 000 & 111 & 111 & 110 & 001 & 001 & 110 \\ 000 & 000 & 000 & 010 & 101 & 100 & 001 & 100 & 110 \end{array}\right]$$

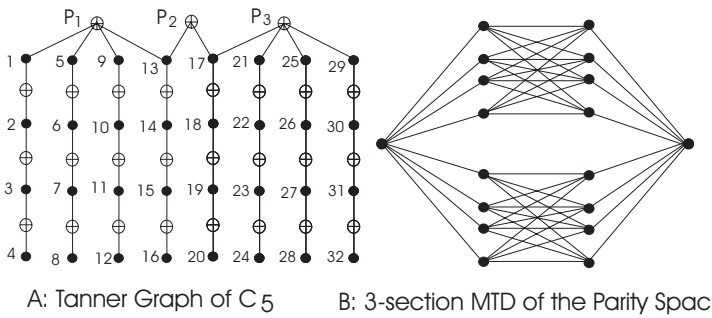


Figure 1: A Tanner graph-trellis representation of the quadratic residue code $q_{32}$. A: The cycle-free Tanner graph of $C_5$; B: A minimal 3-section trellis representing the parity space $M_{PS}$.

Any edge in the second section of the trellis represents four parallel paths. These four parallel paths correspond to the 2-dimensional space introduced by the 5th and 6th rows of the parity matrix $M_{PS}$. In fact, the four parallel paths are distinguished by the four distinct root parities 000, 010, 101 and 111. This together with the symmetry and the low complexity structure of the 4-dimensional subspace of the parity space introduced by the first four rows of $M_{PS}$ results in low decoding complexity for $q_{32}$. The structure of this trellis allows one to apply the efficient computational techniques used for decoding the Golay code $\mathcal{G}_{24}$ [12] and Reed-Muller codes [9] to $q_{32}$.

## 4 Conclusions

The [32, 16, 8] BCH and QR codes were examined. It was shown that these two codes have a relatively similar structure, and that from the projection onto $F_4^8$ and maximum-likelihood decoding perspectives, the two codes behave similarly. As future work, it would be useful to apply the given Tanner graph-trellis representation to the design of a maximum-likelihood decoder for $q_{32}$.

## References

[1] H. Chen and J.T. Coffey, Trellis structure and higher weights of extremal self-dual codes, *Designs, Codes Crypt.* **24** (2001), 15–36.

[2] J.H. Conway and N.J.A. Sloane, Decoding techniques for codes and lattices, including the Golay code and the Leech lattice, *IEEE Trans. Inform. Theory* **32** (1986), 41–50.

[3] M. Esmaeili, T.A. Gulliver and A.K. Khandani, On the Pless construction and ML Decoding of the (48, 24, 12) quadratic residue code, *IEEE Trans. Inform. Theory* **49** no. 6 (2003), 1527–1535.

[4] M. Esmaeili and A.K. Khandani, Acyclic Tanner graph and maximum-likelihood decoding of linear block codes, *IEE Proc. Commun.* **147** no. 6 (2000), 322–332.

[5] P. Gaborit, J.-L. Kim and V. Pless, Decoding binary R(2,5) by hand, *Discrete Math.* **264** (2003), 55–73.

[6] J.-L. Kim, K.E. Mellinger and V. Pless, Projections of binary linear block codes onto larger fields, *SIAM J. Discrete Math.* **16** no. 4 (2003), 591–603.

[7] D. Muder, Minimal trellises for block codes, *IEEE Trans. Inform. Theory* **34** no. 5 (1988), 1049–1053.

[8] V. Pless, Decoding the Golay code, *IEEE Trans. Inform. Theory* **32** (1986), 561–567.

[9] M. Ran and J. Snyders, Constrained designs for maximum likelihood soft decoding of $RM(2, m)$ and the extended Golay codes, *IEEE Trans. Commun.* **43** no. 2/3/4 (1995), 812–820.

[10] J. Snyders and Y. Be'ery, Maximum likelihood soft decoding of binary block codes and decoders for the Golay codes, *IEEE Trans. Inform. Theory* **35** no. 5 (1989), 963–975.

[11] R.M. Tanner, A recursive approach to low complexity codes, *IEEE Trans. Inform. Theory* **27** no. 5 (1981), 533–547.

[12] A. Vardy and Y. Be'ery, More efficient soft decoding of the Golay codes, *IEEE Trans. Inform. Theory* **37** no. 3 (1991), 667–672.

[13] S.B. Wicker, *Error Control Systems for Digital Communication and Storage*, Englewood Cliffs, NJ: Prentice Hall, 1995.

[14] J.K. Wolf, Maximum likelihood decoding of linear block codes using a trellis, *IEEE Trans. Inform. Theory* **24** no. 1 (1978), 76–80.

[15] J. Yuan, C.S. Chen and S. Ma, Two-level decoding of $(32, 16, 8)$ quadratic residue code, *IEE Proc. I* **140** (1993), 409–414.

(Received 16 Feb 2006; revised 28 Oct 2006)