## ACKNOWLEDGMENT

The author wishes to thank the anonymous reviewers for their helpful comments.

## REFERENCES

[1] R. A. Kennedy, G. Pulford, B. D. O. Anderson, and R. R. Bitmead, "When has a decision-directed equalizer converged?," *IEEE Trans. Commun.*, vol. 37, pp. 879–884, Aug. 1989.
[2] K. Doğançay and R. A. Kennedy, "Testing for the convergence of a linear decision directed equaliser," *IEE Proc. Vision, Image Signal Processing*, vol. 141, no. 2, pp. 129–136, Apr. 1994.
[3] R. A. Kennedy, B. D. O. Anderson, and R. R. Bitmead, "Blind adaptation of decision feedback equalizers: Gross convergence properties," *Int. J. Adaptive Contr. Signal Processing*, vol. 7, pp. 497–523, 1993.
[4] M. B. Priestley, *Spectral Analysis and Time Series*, vol. I. London, U.K.: Academic, 1981.
[5] G. M. Jenkins and D. G. Watts, *Spectral Analysis and Its Applications*. San Francisco, CA: Holden-Day, 1968.
[6] I. A. Ibragimov, "A note on the central limit theorem for dependent random variables," *Theory Prob. Appl.*, vol. XX, no. 1, pp. 135–141, 1975.
[7] T. W. Anderson and A. M. Walker, "On the asymptotic distribution of the autocorrelations of a sample from a linear stochastic process," *Ann. Math. Statist.*, vol. 35, pp. 1296–1303, 1964.
[8] C. R. Rao and S. K. Mitra, *Generalized Inverse of Matrices and Its Applications*. New York: Wiley, 1971.
[9] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Trans. Commun.*, vol. COM-28, pp. 1867–1875, Nov. 1980.
[10] J. R. Treichler and B. G. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, no. 2, pp. 459–472, Apr. 1983.

# Perfect Byte-Correcting Codes

Tuvi Etzion, *Member, IEEE*

***Abstract*—We present a few new constructions for perfect linear single byte-correcting codes. These constructions generate some perfect single byte-correcting codes with new parameters, and some perfect single byte-correcting codes with known parameters and simpler presentation and implementation over the known codes. It is also shown that nonequivalent perfect linear single byte-correcting codes exist when all the bytes have the same size.**

***Index Terms*— Byte-correcting code, equivalent codes, perfect code, perfect mixed code, syndrome.**

## I. INTRODUCTION

In most memory and storage systems, the information is stored in bytes. In many of these systems, when an error event occurs it does in a few places of the same byte. Hence, when we consider error detection and correction in such systems, we want to be able to detect and correct all errors that occur in the same byte. Therefore, we consider $t$-byte-correcting codes for these systems. For more information on these systems and codes see [1], [2], and [13].

Let $F^n$ be the set of all words of length $n$ over a finite field $F$. A binary code $C$ of length $n$, in a given metric, is perfect if for some integer $t \geq 0$, every word $x \in F^n$, is within distance $t$ from exactly one codeword of $C$ in the given metric. The study of perfect codes has always been one of the most fascinating subjects in coding theory. They were investigated for various metrics such as the Hamming, Johnson, and Lee [3].

In this correspondence we consider perfect codes which are byte-correcting codes. In this metric, the distance between two words $x$ and $y$ is $d$, if they differ in exactly $d$ bytes. Such perfect codes were considered by Hong and Patel [10]. Our goal is to find when perfect byte-correcting codes can exist, when they do exist, and when they cannot exist. We will consider these codes both for practical use as the "best" byte-correcting codes, and for theoretical point of view as perfect codes.

We distinguish between five types of byte-correcting codes, according to the different sizes of the bytes in the code. Some of these types have practical use, and some do not have such practical use, at least at this point in time.

**Type 1:** All bytes have the same size.
**Type 2:** One byte is of size $b_1$ and the other bytes are of size $b_2$.
**Type 3:** Each byte is of either size $b_1$ or size $b_2$.
**Type 4:** The size of each byte is a power of $2$.
**Type 5:** All the other cases.

Note that a specific byte-correcting code can belong to a few different types at the same time. For all these types there is a simple necessary condition for the existence of the corresponding perfect byte-correcting code. Given a code $C$ of length $n$, for a given codeword $c \in C$ and an integer $t$, let $S$ be the set of words

Manuscript received February 2, 1997; revised December 24, 1997. This work was supported in part under Grant 95-522 from the United-States–Israel Binational Science Foundation (BSF), Jerusalem, Israel, and by the EPSRC of the United Kingdom under Grant GR/L07260. Part of this work was performed while the author visited the Computer Scicence Department at Royal Holloway College, Egham, Surrey TW20 0EX, U.K.

The author is with the Computer Science Department, Technion–Israel Institute of Technology, Haifa 32000, Israel (e-mail: etzion@cs.technion.ac.il).

Publisher Item Identifier S 0018-9448(98)07365-9.

which differ in no more than $t$ bytes from $c$. We say that $S$ is the sphere of radius $t$ from $c$. $C$ is a perfect $t$-byte-correcting code if its minimum distance is $2t+1$ and $|C||S| = 2^n$. If $C$ is a linear code of dimension $k$, then $C$ is a perfect $t$-byte-correcting code if and only if each syndrome of length $n-k$ is produced by exactly one linear combination of columns from no more than $t$ bytes in the parity-check matrix of $C$. Therefore, a necessary condition for the existence of perfect linear single-byte-correcting code with $m$ sizes of bytes, $s_i$ bytes of size $b_i$, $1 \le i \le m$, is that

$$2^{n-k} - 1 = \sum_{i=1}^{m} s_i(2^{b_i} - 1). \tag{1}$$

Byte-correcting codes are closely related to mixed codes. A code of length $n$ is called *mixed* if its codewords are subset of $A_1 \times A_2 \times \cdots \times A_m$, where $A_i$, $1 \le i \le m$, is some alphabet whose size is at least 2. Some necessary conditions and constructions for perfect single-error-correcting mixed codes can be found in [6]–[9] and [11]. If we consider the rows of a byte of size $b$, in a perfect $t$-byte-correcting code $C$, as elements in GF$(2^b)$ then the byte becomes a single coordinate over GF$(2^b)$. If $C$ is a perfect $t$-byte-correcting with $s_i$ bytes of size $b_i$, $1 \le i \le m$, then the corresponding code is $t$-error-correcting mixed code with codewords taken from $A_1 \times A_2 \times \cdots \times A_m$, where $A_i = \text{GF}(2^{b_i})^{s_i}$. Etzion and Greenberg [4] have constructed nonlinear perfect two-error-correcting mixed codes over $Z_2^{2^n} \times Z_{2^n-1}^1$, $n$ even and greater than 2. This code implies the existence of a perfect two-byte-error-correcting code with $2^n$ bytes of size 1 and one byte of size $2^{n-1}$. No other nontrivial perfect $t$-byte-correcting codes with $t > 1$ are known.

In the rest of this correspondence we consider only linear codes. In Section II we show that the necessary condition (1) for the existence of perfect linear single-byte-correcting codes is not always sufficient. In Section III we consider constructions for perfect single-byte-correcting codes of all the five types mentioned above. In Section IV we consider the question of equivalence between perfect single-byte-correcting codes. In the Hamming scheme, which corresponds to the case when all the bytes are of size 1, the linear Hamming code is the unique perfect linear single-error-correcting code. We show that one can find nonequivalent perfect linear single-byte-correcting codes. We consider in this context the case in which all bytes have the same size which is the most interesting type of byte-correcting codes.

## II. Nonexistence Theorem

Two linear subspaces are called *disjoint* if their intersection is the zero element. If the sum of the sizes of the two largest bytes is greater than the number of parity-check symbols of the code, then clearly the necessary condition (1) is not sufficient, since in a space of dimension $\rho$ there cannot be two disjoint subspaces of dimensions $b_1$ and $b_2$, where $b_1 + b_2 > \rho$. Therefore, in a perfect linear single-byte-correcting code with redundancy $\rho$ and bytes of size $b_1$ and $b_2$ we must have $b_1 + b_2 \le \rho$. Now, we will show that this condition and condition (1) are not sufficient for the existence of perfect single byte-correcting codes of Type 2.

*Lemma 1:* If $C$ is a perfect single-byte-correcting code with redundancy $\rho$, one byte of size $b_1$ and $s$ bytes of size $b_2$, then

$$s = 2^{b_1}(2^{\rho - b_1} - 1)/(2^{b_2} - 1)$$

and $b_2$ divides $\rho - b_1$.

*Proof:* By condition (1) we have that

$$2^{\rho} - 1 = 2^{b_1} - 1 + s(2^{b_2} - 1)$$

and hence

$$s = 2^{b_1}(2^{\rho - b_1} - 1)/(2^{b_2} - 1).$$

Therefore, $b_2$ divides $\rho - b_1$. □

*Theorem 1:* A perfect single-byte-correcting code with one byte of size $b_1$ and the other bytes of size $b_2$, with $b_1 < b_2$, cannot exist.

*Proof:* Assume that $H$ is an $\rho \times n$ parity-check matrix of a perfect single-byte-correcting code of length $n$ and $\rho$ parity-check symbols, with one byte of size $b_1$, and $s$ bytes of size $b_2$, $b_1 < b_2$. The $b_1$ columns in the byte of size $b_1$ are linearly independent and hence, without loss of generality (w.l.o.g.) we can assume that column $i$, $1 \le i \le b_1$, is a vector of weight one with a ONE in the $i$th entry. Let $H_1$ be the $b_1 \times n$ matrix whose rows are the first $b_1$ rows of $H$. Since all the $2^{\rho} - 1$ linear combinations of nonempty subsets of columns from the bytes of $H$ consists of all nonzero binary $r$-tuples, it follows that in $H_1$ these linear combinations result in $1 + s(2^{b_2} - 1)/2^{b_1}$ (equal $2^{\rho - b_1}$ by Lemma 1) occurrences of each nonzero $b_1$-tuple, and $s(2^{b_2} - 1)/2^{b_1} = 2^{\rho - b_1} - 1$ occurrences of the all-zero $b_1$-tuple. Given a $b_1 \times b_2$ matrix $A$ with rank $m \le b_1$, there are $2^{b_2 - m} - 1$ linear combinations of nonempty subsets of columns from $A$ which result in the all-zero $b_1$-tuple. For each (of the $2^m - 1$) nonzero $b_1$-tuple $v$ in the subspace spanned by the columns of $A$, there are exactly $2^{b_2 - m}$ linear combinations of nonempty subsets of columns from $A$ which result in $v$. The first $b_1$ rows of $H$ in each byte of size $b_2$ can be viewed as such matrix $A$. Since each nonzero $b_1$-tuple is a result of exactly one linear combination of the byte of size $b_1$ in $H_1$, it follows that it should be a result of $2^{\rho - b_1} - 1$ linear combinations of columns in the other bytes. But this is not possible since for any $m \le b_1$, $2^{b_2 - m}$ is even and greater than 1. Thus there is no perfect single-byte-correcting code with one byte of size $b_1$ and the other bytes of size $b_2$, with $b_1 < b_2$. □

Theorem 1 excludes the possibility of perfect single-byte-correcting codes of length $2^{b_1}b_2 + b_1$ with $2^{b_1}$ bytes of size $b_2$ and one byte of size $b_1$. In this case, $\rho = b_1 + b_2$ and condition (1) is satisfied since

$$2^{b_1}(2^{b_2} - 1) + 2^{b_1} - 1 = 2^{b_1 + b_2} - 1 = 2^{\rho} - 1.$$

## III. Constructions for Byte-Correcting Codes

This section is devoted to constructions of perfect single-byte-correcting codes. We will consider all the first four types of byte-correcting codes mentioned in Section I. If all bytes are of the same size $b$, then it is well known, e.g., [10] that perfect single-byte-correcting code with redundancy $\rho$ exists if and only if $2^b - 1$ divides $2^{\rho} - 1$, i.e., $b$ divides $\rho$. Hence for codes of Type 1 condition (1) is also sufficient. The code can be derived from a perfect single-error-correcting code of length $\frac{2^{\rho}-1}{2^b-1}$ over GF$(2^b)$, similarly as was explained in Section I for perfect mixed codes. For completeness we describe now one way to construct a parity-check matrix of such code. Throughout this section and the next one when $\gamma \in$ GF$(2^m)$ is written in a parity-check matrix, we consider $\gamma$ as the column of its binary $m$-tuple representation in the field. If $\alpha$ be a primitive element in GF$(2^{rb})$ and $s = (2^{rb} - 1)/(2^b - 1)$, then $\alpha^s$ is a primitive element of the subfield GF$(2^b)$ and $\alpha^0, \alpha^s, \alpha^{2s}, \cdots, \alpha^{(b-1)s}$ is a basis for the subfield. Thus for each $i$, $0 \le i \le s-1$, the $2^b$ elements $\{0, \alpha^i, \alpha^{i+s}, \alpha^{i+2s}, \cdots, \alpha^{i+(2^b-2)s}\}$ are closed under addition in GF$(2^{rb})$ and $\alpha^i, \alpha^{i+s}, \alpha^{i+2s}, \cdots, \alpha^{i+(b-1)s}$ are linearly independent. Therefore, the matrix

$$H = [H_0 H_1 \cdots H_{s-1}]$$

where $s = (2^{rb} - 1)/(2^b - 1)$ and

$$H_i = [\alpha^i, \alpha^{i+s}, \alpha^{i+2s}, \cdots, \alpha^{i+(b-1)s}], \qquad 0 \le i \le s-1$$

is a parity-check matrix for a perfect single-byte-correcting code of length $sb$, $rb$ parity-check symbols, and bytes of size $b$. Clearly, the set $\{0, \alpha^i, \alpha^{i+s}, \alpha^{i+2s}, \cdots, \alpha^{i+(2^b-2)s}\}$ can be further partitioned into disjoint linear subspaces of size $2^{b'}$ if and only if $2^{b'} - 1$ divides

$2^b - 1$, i.e., $b'$ divides $b$. This result is easily generalized to any subspace of rank $b$.

*Lemma 2:* Any subspace of rank $b$ can be partitioned into $s = (2^b - 1)/(2^{b'} - 1)$ disjoint subspaces of rank $b'$, for each $b'$ which divides $b$.

*Proof:* Let $\alpha$ be a primitive element in GF$(2^b)$ and $L$ be a subspace of rank $b$. Let $\beta_0, \beta_1, \cdots, \beta_{b-1}$ be any $b$ linearly independent elements in $L$. We define the following mapping $h$ from GF$(2^b)$ into $L$, $h(0) = 0$, $h(\alpha^i) = \beta_i$, $0 \le i \le b-1$, and $h(\alpha^i) = \beta_i$, $b \le i \le 2^b - 2$, where

$$\alpha^i = \sum_{j=0}^{b-1} c_j \alpha^j, \qquad c_j \in \{0,1\}$$

and

$$\beta_i = \sum_{j=0}^{b-1} c_j \beta_j.$$

We claim that for each $i$, $0 \le i \le s-1$, the subset

$$\{0, \beta_i, \beta_{i+s}, \beta_{i+2s}, \cdots, \beta_{i+(2^{b'}-2)s}\}$$

is a subspace of rank $b'$. Clearly, $\alpha^{i+k_1 s} = \sum_{j=0}^{b-1} d_j \alpha^j$ and $\alpha^{i+k_2 s} = \sum_{j=0}^{b-1} e_j \alpha^j$, $d_j, e_j \in \{0,1\}$, $0 \le j \le b-1$ and hence

$$\alpha^{i+k_1 s} + \alpha^{i+k_2 s} = \alpha^{i+k_3 s} = \sum_{j=0}^{b-1} (d_j + e_j)\alpha^j.$$

Therefore, by the definition of the mapping $h$ we have that

$$\begin{aligned}\beta_{i+k_1 s} + \beta_{i+k_2 s} &= \sum_{j=0}^{b-1} d_j \beta_j + \sum_{j=0}^{b-1} e_j \beta_j \\ &= \sum_{j=0}^{b-1} (d_j + e_j)\beta_j = \beta_{i+k_3 s}.\end{aligned}$$

Thus the set

$$\{0, \beta_i, \beta_{i+s}, \beta_{i+2s}, \cdots, \beta_{i+(2^{b'}-2)s}\}$$

is a subspace of rank $b'$. □

Lemma 2, and condition (1) implies the following result.

*Theorem 2:* A perfect single-byte-correcting code with redundancy $\rho$ and $s_i$ bytes of size $b_i$, $1 \le i \le m$, such that $b_i$, $1 \le i \le m-1$, divides $b_{i+1}$ and $b_m$ divides $\rho$, exists if and only if

$$\sum_{i=1}^{m} s_i (2^{b_i} - 1) = 2^\rho - 1.$$

Theorem 2 provides a proof to the fact that the necessary condition (1) for the existence of perfect single-byte-correcting codes of Type 4 is also sufficient if the number of parity-check symbols is also a power of 2. Next, we consider the case when one byte is of size $b_1$ and the other bytes are of size $b_2$. We have shown already in Lemma 1 and Theorem 1 that a necessary condition for the existence of such code with $\rho$ parity-check symbols is that $b_2$ divides $\rho - b_1$ and $b_1 > b_2$. Hong and Patel [10] proved that this necessary condition is also sufficient if condition (1) holds. We will present a construction for codes with the same parameters, which have a simpler presentation and implementation over the codes of Hong and Patel [10].

*Construction A*

Let $b_1 > b_2$, $\alpha$ be a primitive element in GF$(2^{rb_2})$, $\beta$ a primitive element in GF$(2^{b_1})$, and $s = (2^{rb_2} - 1)/(2^{b_2} - 1)$. Let $H$ be the matrix defined by

$$H = [A \quad B \quad C].$$

1) $A$ is the $(rb_2 + b_1) \times b_1$ matrix defined by

$$A = \begin{bmatrix} I \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

where $I$ is the identity matrix of order $b_1$ and 0 is a $b_2 \times b_1$ all-zero matrix.

2) $B$ is the $(rb_2 + b_1) \times (sb_2)$ matrix of the form

$$B = [B_0 \quad B_1 \quad \cdots \quad B_{s-1}]$$

where $B_i$, $0 \le i \le s-1$, is an $(rb_2 + b_1) \times b_2$ matrix defined by

$$B_i = \begin{bmatrix} \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ \alpha^i & \alpha^{i+s} & \cdots & \alpha^{i+(b_2-1)s} \end{bmatrix}.$$

3) $C$ is the $(rb_2 + b_1) \times s(2^{b_1} - 1)b_2$ matrix of the form

$$C = \begin{bmatrix} C_0 & C_1 & \cdots & C_{s(2^{b_1}-1)-1} \end{bmatrix}$$

where $C_k$, $k = js + i$, $0 \le j \le 2^{b_1} - 2$, $0 \le i \le s-1$, is an $(rb_2 + b_1) \times b_2$ matrix defined by

$$C_k = \begin{bmatrix} \beta^j & \beta^{j+1} & \cdots & \beta^{j+b_2-1} \\ \alpha^i & \alpha^{i+s} & \cdots & \alpha^{i+(b_2-1)s} \end{bmatrix}.$$

*Theorem 3:* Construction A produces a parity-check matrix for a perfect single-byte-correcting code with redundancy $rb_2 + b_1$, one byte of size $b_1$ and the other bytes of size $b_2$.

*Proof:* We have to show that each nonzero syndrome of length $rb_2 + b_1$ is produced by exactly one linear combination of columns from one byte of $H$. The syndromes which are produced from $A$ are exactly all those vectors whose last $rb_2$ entries are zeros. The syndromes which are produced from the linear combinations of the columns inside the bytes of $B$ are exactly all those with zeros in the first $b_1$ entries. It remains to be shown that each syndrome which is nonzero in the first $b_1$ entries and nonzero in the last $rb_2$ entries is produced by linear combination of exactly one of the $C_k$'s. Given such syndrome $v$, where $v^T = ((\beta^{l_1})^T | (\alpha^{l_2})^T)$, by Lemma 2 there exists a unique $i$, $0 \le i \le s-1$, and a unique linear combination such that

$$\alpha^{l_2} = \sum_{m=0}^{b_2-1} c_m \alpha^{i+ms}, \qquad c_m \in \{0,1\}.$$

Since $b_1 > b_2$, it follows that

$$\gamma = \sum_{m=0}^{b_2-1} c_m \beta^m \neq 0$$

and there exists a unique $j$, $0 \le j \le 2^{b_1} - 2$, such that $\beta^{l_1} = \beta^j \gamma$, and hence $v$ is obtained by a unique linear combination from $C_{js+i}$. □

As we said, the parameters of the codes obtained by Construction A are the same as the ones obtained in [10]. But, our presentation and the implementation of Construction A is simpler. We only need primitive elements in GF$(2^{rb_2})$ and GF$(2^{b_1})$, while in [10] the presentation and the implementation require primitive elements in each one of the fields GF$(2^{ib_2+b_1})$, $0 \le i \le r-1$. As a consequence of Lemma 1, and Theorems 1 and 3, we have the following.

*Corollary 1:* A perfect single-byte-correcting code with redundancy $\rho$, one byte of size $b_1$ and the other bytes of size $b_2$, exists if and only if $b_2$ divides $\rho - b_1$ and $b_1 > b_2$.

A more generalized construction is the following one.

*Construction B*

Let $H_1$ be the parity-check matrix of a perfect single-byte-correcting code $C_1$ with redundancy $\rho$, and $n_i$ bytes of size $b_i$, $1 \le i \le m$. Let further $\alpha$ be a primitive element in GF $(2^\rho)$ and $b_{m+1}$ be a positive integer less than or equal to $\rho$ ($b_{m+1}$ is not necessarily distinct from the other $b_i$'s). We define a matrix $H_2$ as follows:

$$H_2 = \begin{bmatrix} H_1 & 0 & A_0 & A_1 & \cdots & A_{2^\rho-2} \\ 0 & I & I & I & \cdots & I \end{bmatrix}$$

where $I$ is the identity matrix of order $b_{m+1}$ and $A_i$, $0 \le i \le 2^\rho - 2$, is a $\rho \times b_{m+1}$ matrix defined by

$$A_i = [\alpha^i \quad \alpha^{i+1} \quad \cdots \quad \alpha^{i+b_{m+1}-1}].$$

The proof of the following theorem is very similar to the one of Theorem 3.

*Theorem 4:* Construction B produces a parity-check matrix for a perfect single-byte-correcting code with redundancy $\rho + b_{m+1}$, where $b_{m+1} \le \rho$, with $n_i$ bytes of size $b_i$, $1 \le i \le m$, and $2^\rho$ bytes of size $b_{m+1}$.

Construction B can be further applied to obtain perfect single-byte-correcting codes with various parameters. The parameters of the perfect single-byte-correcting codes obtained in Construction A can be also obtained via Construction B, but the presentation in Construction A is simpler and can be easily implemented. To obtain more perfect single-byte-correcting codes we should use Lemma 2 to replace a byte of size $b$ by $(2^b - 1)/(2^{b'} - 1)$ bytes of size $b'$ for any $b'$ which divides $b$. Several other methods in which several bytes of size $b_1$ are replaced by several bytes of size $b_2$ to obtain perfect single-byte-correcting codes with other parameters are presented in the Appendix.

## IV. Nonequivalent Byte-Correcting Codes

Two linear codes $C_1$ and $C_2$ are called *equivalent* if there exists a permutation $\pi$ such that $C_1 = \{\pi(c) : c \in C_2\}$. It is well known that the unique linear perfect single-error-correcting code is the $[2^\rho - 1, 2^\rho - \rho - 1]$ Hamming code. When we consider other perfect single-byte-correcting codes we have to make some slight changes in the definition of equivalent codes, such that a permutation $\pi$ can permute elements only inside bytes and permute bytes. For simplicity, we only give the formal definition for byte-correcting codes with bytes of size $b$. Two linear byte-correcting codes $C_1$ and $C_2$ of length $mb$ with $m$ bytes of size $b$ are called *equivalent* if there exist a permutation $\pi$ such that $C_1 = \{\pi(c) : c \in C_2\}$ and $\pi = (\pi_0, \pi_1, \cdots, \pi_{mb-1})$, where for each $i$, $0 \le i \le m-1$

$$\{\pi_{ib}, \pi_{ib+1}, \cdots, \pi_{ib+b-1}\} = \{jb, jb+1, \cdots, jb+b-1\}$$

for some $j$, $0 \le j \le m-1$. A similar definition can be given to other types of byte-correcting codes. We will now show that for each redundancy $rb$, $r \ge 3$, $b \ge 2$, there exist two nonequivalent perfect single-byte-correcting codes $C_1$ and $C_2$ with bytes of size $b$, and redundancy $rb$.

*Construction C*

For a primitive element $\alpha$ in GF $(2^{(r-1)b})$ construct the following four sets of $rb \times b$ matrices.

1) The first set consists of one $rb \times b$ matrix

$$A = \begin{bmatrix} I \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

where $I$ is the identity matrix of order $b$.

2) The second set consists of $s = (2^{(r-1)b} - 1)/(2^b - 1)$ matrices of the form

$$B_i = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \alpha^i & \alpha^{i+s} & \cdots & \alpha^{i+(b-1)s} \end{bmatrix}, \qquad 0 \le i \le s-1.$$

3) The third set consists of $2^{(r-1)b} - 1$ matrices of the form

$$C_k = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 \\ \alpha^k & \alpha^{k+1} & \cdots & \alpha^{k+b-1} \end{bmatrix}, \qquad 0 \le k \le 2^{(r-1)b} - 2.$$

4) The fourth set consists of $2^{(r-1)b} - 1$ matrices of the form

$$D_k = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 \\ \alpha^{i+js} & \alpha^{i+(j+1)s} & \cdots & \alpha^{i+(j+b-1)s} \end{bmatrix},$$
$$0 \le i \le s-1, \quad 0 \le j \le 2^b - 2, \quad k = js + i.$$

Define $F_1$ as the code whose parity-check matrix is

$$H_1 = [A \ B_0 \cdots B_{s-1} \ C_0 \ \cdots \ C_{2^{(r-1)b}-1}]$$

and $F_2$ as the code whose parity-check matrix is

$$H_2 = [A \ B_0 \ \cdots \ B_{s-1} \ D_0 \ \cdots \ D_{2^{(r-1)b}-1}].$$

Each $rb \times b$ matrix corresponds to a distinct byte of size $b$.

*Theorem 5:* The codes $F_1$ and $F_2$ defined in Construction C are perfect single-byte-correcting codes of length $b(2^{rb} - 1)/(2^b - 1)$ and bytes of size $b$.

*Proof:* We have to show that each nonzero syndrome of length $rb$ is produced by exactly one linear combination of columns from one byte of the parity-check matrix $H_i$, $i = 1, 2$. The syndromes which are produced from $A$ are exactly all those vectors whose last $(r-1)b$ entries are zeros. By Lemma 2, the syndromes which are produced from the linear combinations of the columns inside the bytes of $B$ are exactly all those with zeros in the first $b$ entries. It remains to be shown that each syndrome which is nonzero in the first $b$ entries and nonzero in the last $(r-1)b$ entries is produced by linear combination of exactly one of the $C_k$'s and exactly one linear combination of the $D_k$'s. Given such syndrome $v$, where $v^T = ((u)^T | (\alpha^l)^T)$, $u = (u_0 u_1 \cdots u_{b-1})$, let

$$\beta = \sum_{i=0}^{b-1} u_i \alpha^i$$

and

$$\gamma = \sum_{i=0}^{b-1} u_i \alpha^{is}.$$

There exists a unique $k$, $0 \le k \le 2^{(r-1)b} - 2$, such that $\alpha^l = \alpha^k \beta$, and hence $v$ is obtained by a unique linear combination from $C_k$. There is also a unique $k$, $0 \le k \le 2^{(r-1)b} - 2$, such that $\alpha^l = \alpha^k \gamma$, and hence $v$ is obtained by a unique linear combination from $D_k$. Thus the lemma is proved. □

*Theorem 6:* The code $F_1$ and $F_2$ defined in Construction C are not equivalent.

We will assume that $F_1$ and $F_2$ are equivalent. Therefore, also their dual codes $G_1$ and $G_2$ are equivalent. $H_1$ and $H_2$ are the generator matrices of $G_1$ and $G_2$, respectively. Hence, by elementary row operations on $H_1$, with possible permutations on rows, columns within bytes, and bytes we can obtain the matrix $H_2$. We will first prove that no elementary row operations are needed. Let $D_i'(C_i')$ be the $(r-1)b \times b$ matrix consisting of the last $(r-1)b$ rows of $D_i$ $(C_i)$, $0 \le i \le 2^{(r-1)b}-2$. First note that $C_i'$, $0 \le i \le 2^{(r-1)b}-2$, consists of columns $i$ through $i+b-1$, where computation is done modulo $2^{(r-1)b}-1$, of the $(r-1)b \times (2^{(r-1)b}-1)$ matrix

$$M = \begin{bmatrix} \alpha^0 & \alpha^1 & \cdots & \alpha^{2^{(r-1)b}-2} \end{bmatrix}.$$

Each row of $M$ is a shift of an $m$-sequence $\tilde{S}$ of order $(r-1)b$ [5], [12], when the $m$-sequence $\tilde{S}$ and the field GF$(2^{(r-1)b})$ are generated from the same primitive polynomial. This means that each nonzero $[(r-1)b]$-tuples appears exactly once as a window of length $(r-1)b$ in each (cyclic) row. Next, we want to remind that addition of two different cyclic shifts of $\tilde{S}$ is a different cyclic shift of $\tilde{S}$ [5], and since the rank of $M$ is $(r-1)b$, it follows that any linear combination of nonempty set of rows from $M$ is another cyclic shift of $\tilde{S}$. Each nonzero $b$-tuple appears

$$\frac{2^{(r-1)b}}{2^b} = 2^{(r-2)b}$$

times as a window of width $b$ in $\tilde{S}$, and the allzero $b$-tuple appears $2^{(r-2)b}-1$ times. Hence, in a given row of the $C_i'$'s, each nonzero $b$-tuple appears in $2^{(r-2)b}$ bytes, and the allzero $b$-tuple in $2^{(r-2)b}-1$ bytes. Note, that in each of the bytes of the first $b$ rows of both $H_1$ and $H_2$, in each row we have a $b$-tuple of weight one which appears $2^{(r-1)b}$ times and the all-zero $b$-tuple which appears $s$ times. Hence, any linear combination of at least one of the last $(r-1)b$ rows and at least one of the first $b$ rows will not result in a row which includes in the bytes a $b$-tuple of weight one $2^{(r-1)b}$ times. Therefore, the first $b$ rows of $H_1$ should remain unchanged with possible permutations of rows and columns.

As for the last $(r-1)b$ rows of $H_1$, note that in $A$ these rows consist of zeros only. In order to have in $H_1$ such a matrix after the row operations and the permutations applied, we have to consider two cases:

*Case 1:* We take only linear combinations of the last $(r-1)b$ rows which practically leave these rows unchanged (in the sense that we obtain $(r-1)s$ linearly independent shifts of $\tilde{S}$).

*Case 2:* We obtain the matrix $A$ from $C_k$ by adding $\tilde{C}_k$ (given below), for some $k$ ,$0 \le k \le 2^{(r-1)b}-2$, (it is euivalent for some row operations) to $A$ and all the $C_i$'s

$$\tilde{C}_k = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \\ \alpha^k & \alpha^{k+1} & \cdots & \alpha^{k+b-1} \end{bmatrix}.$$

Now, for $i \neq k$

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 \\ \alpha^i & \alpha^{i+1} & \cdots & \alpha^{i+b-1} \end{bmatrix} + \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \\ \alpha^k & \alpha^{k+1} & \cdots & \alpha^{k+b-1} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 \\ \alpha^i + \alpha^k & \alpha^{i+1} + \alpha^{k+1} & \cdots & \alpha^{i+b-1} + \alpha^{k+b-1} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 \\ \alpha^i + \alpha^k & (\alpha^i + \alpha^k)\alpha & \cdots & (\alpha^i + \alpha^k)\alpha^{b-1} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 \\ \alpha^j & \alpha^{j+1} & \cdots & \alpha^{j+b-1} \end{bmatrix}.$$

Since $\alpha^{i_1} + \alpha^k \neq \alpha^{i_2} + \alpha^k$ for $i_1 \neq i_2$, it follows that the set of matrices containing $A$ and the $C_i$'s are left invariant under the addition of $\tilde{C}_k$. Therefore, except for possible permutations of rows and columns, the first $b$ rows of $H_1$ remain unchanged and the last $(r-1)b$ rows remain as $(r-1)b$ different linearly independent cyclic shifts of $\tilde{S}$.

Now, note that $C_i'$ and $C_{i+1}'$, $0 \le i \le 2^{(r-1)b}-2$, share $b-1$ columns, and also $D_i'$ and $D_{i+s}'$, $0 \le i \le 2^{(r-1)b}-2$, share $b-1$ columns. Let $G(C) = (V_1, E_1)$ be a graph whose set of vertices is $V_1 = \{v_i : 0 \le i \le 2^{(r-1)b}-2\}$ and set of edges $E_1$ consists of all the pairs $\{v_i, v_j\}$ such that $C_i'$ and $C_j'$ share $b-1$ columns in common. Clearly, $G$ is a cycle of length $2^{(r-1)b}$. Now, note that by permuting columns within bytes, the newly defined graph $G$ (by the same definition) remains a cycle of length $2^{(r-1)b}$. Similarly, we define a graph $G(D) = (V_2, E_2)$ whose set of vertices is $V_2 = V_1$ and set of edges $E_2$ consists of all the pairs $\{v_i, v_j\}$ such that $D_i'$ and $D_j'$ share $b-1$ columns in common. The graph $G(D)$ contains $s$ cycles of length $2^b-1$. Thus there is no permutation on bytes and columns within bytes which maps the $C_i'$'s into the $D_{ij}'$'s, i.e., $F_1$ and $F_2$ are not equivalent.

Thus we have proved

*Corollary 2:* For each redundancy $rb$, $b \ge 2$, $r \ge 3$, there exist two nonequivalent perfect single-byte-correcting codes with bytes of size $b$, and redundancy $rb$.

## Appendix

In this appendix we present a few more constructions for perfect byte-correcting codes. For the first method we need the following two lemmas.

*Lemma 3:* Let $b_1$ and $b_2$ be two distinct integers smaller than $b$, which divide $b$. If $s_i = (2^b-1)/(2^{b_i}-1)$, $i = 1, 2$, then g.c.d. $(s_1, s_2) > 1$, where g.c.d. is the *greatest common divisor* of $s_1$ and $s_2$.

*Proof:* Clearly, $b_1 \le \lceil \frac{b}{2} \rceil$ and $b_2 \le \lceil \frac{b}{2} \rceil$. Without loss of generality, we can assume that $b_2 < b_1$ and hence $s_1 \ge 2^{\lceil \frac{b}{2} \rceil}$ and $s_2 \ge 2^{\lceil \frac{b}{2} \rceil + 1}$, and therefore, $s_1 s_2 > 2^b$. Since both $s_1$ and $s_2$ divides $2^b-1$ and $s_1 s_2 > 2^b - 1$, it follows that g.c.d. $(s_1, s_2) > 1$. □

*Lemma 4:* Let $b_1$ and $b_2$ be two distinct integers smaller than $b$, which divide $b$. If $s_i = (2^b - 1)/(2^{b_i} - 1)$, $i = 1, 2$, then there is no solution for the equation $is_1 + js_2 \equiv 0 \pmod{2^b - 1}$, for $0 \le i \le \frac{s_2}{s_3} - 1$, $0 \le j \le 2^{b_2} - 2$, where $s_3 = \text{g.c.d.}(s_1, s_2)$.

*Proof:* By definition, $s_1 = s_3 x_1$ and $s_2 = s_3 x_2$ for two integers $x_1$ and $x_2$ such that g.c.d. $(x_1, x_2) = 1$. Hence, $2^b - 1 = s_3 x_1 x_2 y$ and, therefore, $2^{b_1} - 1 = x_2 y$ and $2^{b_2} - 1 = x_1 y$. Now, assume there is a solution for the equation $is_1 + js_2 \equiv 0 \pmod{2^b - 1}$, for $0 \le i \le \frac{s_2}{s_3} - 1$, $0 \le j \le 2^{b_2} - 2$, i.e.,

$$i\frac{2^b - 1}{2^{b_1} - 1} + j\frac{2^b - 1}{2^{b_2} - 1} = k(2^b - 1), \qquad k \ge 1$$

and hence

$$i + j\frac{2^{b_1} - 1}{2^{b_2} - 1} = k(2^{b_1} - 1)$$

or

$$i + j\frac{x_2}{x_1} = k(2^{b_1} - 1).$$

It follows that $j$ must be a multiple of $x_1$ and hence its maximum value can be $x_1(y - 1)$ and our equation can take the form

$$i + x_2 y - x_2 \ge k(2^{b_1} - 1) = k x_2 y$$

and hence $i \ge x_2$. But $0 \le i \le \frac{s_2}{s_3} - 1 = x_2 - 1$, a contradiction. Thus the lemma follows. □

The next construction is in some sense based on a double application of Lemma 2

*Construction D*

Let $b$ be an integer divisible by $b_1$ and $b_2$, $s_i = (2^b - 1)/(2^{b_i} - 1)$, $i = 1, 2$, and $s_3 = \text{g.c.d.}(s_1, s_2)$. For a given $m$, $0 \le m \le s_3 - 1$, let

$$H = \begin{bmatrix} H_0 H_1 & \cdots & H_{\frac{s_1}{s_3} - 1} A \end{bmatrix}$$

where

$$H_i = [\alpha^{m+is_3}, \alpha^{m+is_3+s_1}, \alpha^{m+is_3+2s_1}, \cdots, \alpha^{m+is_3+(b_1-1)s_1}], \qquad 0 \le i \le \frac{s_1}{s_3} - 1$$

corresponds to a byte of size $b_1$, and $\alpha$ is a primitive element in GF$(2^b)$, be the parity-check matrix of a perfect single-byte-correcting code $C$. Let

$$G_i = [\alpha^{m+is_1}, \alpha^{m+is_1+s_2}, \alpha^{m+is_1+2s_2}, \cdots, \alpha^{m+is_1+(b_2-1)s_2}], \qquad 0 \le i \le \frac{s_2}{s_3} - 1$$

and define

$$H' = \begin{bmatrix} G_0 G_1 & \cdots & G_{\frac{s_2}{s_3} - 1} A \end{bmatrix}.$$

*Theorem 7:* The matrix $H'$ defined in Construction D is a parity-check matrix of a perfect single-byte-correcting code.

*Proof:* We only have to prove that the syndromes generated by the matrices $H_i$, $0 \le i \le \frac{s_1}{s_3} - 1$, are the same as the ones generated by the matrices $G_i$, $0 \le i \le \frac{s_2}{s_3} - 1$. Without loss of generality, we assume that $m = 0$. $H_i$, $0 \le i \le \frac{s_1}{s_3} - 1$, produces the syndromes $\{\alpha^{is_3 + js_1} : 0 \le j \le 2^{b_1} - 2\}$. These are exactly all the $\frac{2^b - 1}{s_3}$ multiples of $s_3$. $G_i$, $0 \le i \le \frac{s_2}{s_3} - 1$, produces the syndromes $\{\alpha^{is_1 + js_2} : 0 \le j \le 2^{b_2} - 2\}$. Again, these are clearly $\frac{2^b - 1}{s_3}$ multiples of $s_3$. By Lemma 4, no two of these multiples are equal and hence these are all the multiples of $s_3$ and thus the $H_i$'s and the $G_i$'s generate the same syndromes. □

The second method will replace $2^{b_1} - 1$ bytes of size $b_2$ by $2^{b_2} - 1$ bytes of size $b_1$, where $b_1 < b_2$, if the $2^{b_1} - 1$ bytes of size $b_2$ arranged in a certain form which can be obtained for most parameters of byte-correcting codes mentioned before.

*Construction E*

Let $b_1$ and $b_2$ be two integers greater than 1 such that $b_1 < b_2$, and $\alpha$ be a primitive element in GF$(2^{b_2})$. Let $\{0, \alpha^{i_1}, \alpha^{i_2}, \cdots, \alpha^{i_{(2^{b_1}-1)}}\}$ be the subspace spanned by $\alpha^0$, $\alpha^1, \cdots, \alpha^{b_1 - 1}$. Let

$$H = [R_1 R_2 \quad \cdots \quad R_{2^{b_1}-1} A]$$

be the parity-check matrix of a perfect single-byte-correcting code $C$, where

$$R_k = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \alpha^0 & \alpha^1 & \cdots & \alpha^{b_2 - 1} \\ \alpha^{i_k} & \alpha^{1+i_k} & \cdots & \alpha^{b_2 - 1 + i_k} \end{bmatrix}, \qquad 1 \le k \le 2^{b_1} - 1$$

where 0 is all-zero $r$-tuple. Let $T_j$, $0 \le j \le 2^{b_2} - 2$, be the $(r + 2b_2) \times b_1$ matrix

$$T_j = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \alpha^j & \alpha^{j+1} & \cdots & \alpha^{j+b_1-1} \\ \alpha^j & \alpha^{j+2} & \cdots & \alpha^{j+2(b_1-1)} \end{bmatrix}$$

and define

$$H' = [T_0 T_1 \quad \cdots \quad T_{2^{b_2}-2} A].$$

*Theorem 8:* The matrix $H'$ defined in Construction E is a parity-check matrix of a perfect single-byte-correcting code.

*Proof:* $H$ is a parity-check matrix of a perfect single-byte-correcting code $C$ and hence the structure of $A$ is irrelevant to the proof. We only have to prove that the syndromes generated by the matrices $R_k$, $1 \le k \le 2^{b_1} - 1$, are the same as the ones generated by the matrices $T_j$, $0 \le j \le 2^{b_2} - 2$. Since $\alpha^0, \alpha^1, \cdots, \alpha^{b_2 - 1}$ is a basis of GF$(2^{b_2})$, it follows that $R_k$ produces exactly the $2^{b_2} - 1$ syndromes of the form

$$\begin{bmatrix} 0 \\ \alpha^l \\ \alpha^{l+i_k} \end{bmatrix}, \qquad 0 \le l \le 2^{b_2} - 2.$$

Recall now that GF$(2^{b_2})$ is a field with characteristic 2 and hence $\alpha^{i_m} + \alpha^{i_l} = \alpha^{i_n}$ if and only if $\alpha^{2i_m} + \alpha^{2i_k} = \alpha^{2i_n}$. Also, recall that $\alpha^0, \alpha^1, \cdots, \alpha^{b_1 - 1}$ span the subspace $\{0, \alpha^{i_1}, \alpha^{i_2}, \cdots, \alpha^{i_{(2^{b_1}-1)}}\}$. It follows that $T_j$, $0 \le j \le 2^{b_2} - 2$, produces exactly the $2^{b_1} - 1$ syndromes of the form

$$\begin{bmatrix} 0 \\ \alpha^{j+i_k} \\ \alpha^{j+2i_k} \end{bmatrix}, \qquad 1 \le k \le 2^{b_1} - 1.$$

Now, it can be readily verified that the $R_k$'s and the $T_j$'s produce the same $(2^{b_1} - 1)(2^{b_2} - 1)$ syndromes. □

Now, we want to show that there exist perfect single-byte-correcting codes that include the $(r + 2b_2) \times b_2$ matrices $R_k$, $0 \le k \le 2^{b_1} - 2$, as submatrices (which corresponds to bytes of size $b_2$) of their $(r + 2b_2) \times n$ parity-check matrices. The construction given can be easily applied to obtain codes with the parameters of the codes obtained by Construction A.

$$H_2 = \begin{bmatrix} H_1 & 0 & 0 & 0 & 0 & \cdots & 0 & B_0 & \cdots & B_{2^\rho-2} & B_0 & \cdots & B_{2^\rho-2} \\ 0 & 0 & I & A_0 & A_0 & \cdots & A_0 & I & \cdots & I & 0 & \cdots & 0 \\ 0 & I & 0 & A_0 & A_1 & \cdots & A_{2^{b_2}-2} & 0 & \cdots & 0 & I & \cdots & I \end{bmatrix}$$

$$\begin{bmatrix} B_0 & \cdots & B_{2^\rho-2} & B_0 & \cdots & B_{2^\rho-2} & \cdots & B_0 & \cdots & B_{2^\rho-2} \\ A_0 & \cdots & A_0 & A_0 & \cdots & A_0 & \cdots & A_0 & \cdots & A_0 \\ A_0 & \cdots & A_0 & A_1 & \cdots & A_1 & \cdots & A_{2^{b_2}-2} & \cdots & A_{2^{b_2}-2} \end{bmatrix}$$

*Construction F*

Let $C_1$ be a perfect single-byte-correcting code with redundancy $\rho$, $s_i$ bytes of size $b_i'$, $1 \le i \le m$, and parity-check matrix $H_1$. Let further, $\alpha$ be a primitive element in GF$(2^{b_2})$ and $\beta$ be a primitive element in GF$(2^\rho)$. Construct the parity-check matrix shown at the top of this page, where $I$ is the identity matrix of order $b_2$, $A_i$ is a $b_2 \times b_2$ matrix of the form

$$A_i = [\alpha^i \quad \alpha^{i+1} \quad \cdots \quad \alpha^{i+b_2-1}], \qquad 0 \le i \le 2^{b_2} - 2$$

and $B_i$ is an $\rho \times b_2$ matrix of the form

$$B_i = [\beta^i \quad \beta^{i+1} \quad \cdots \quad \beta^{i+b_2-1}], \qquad 0 \le i \le 2^{\rho} - 2.$$

By similar techniques to those used in the Proof of Theorem 3 we can prove the following theorem.

*Theorem 9:* The parity-check matrix $H_2$ of Construction F is a parity-check matrix of a perfect single-byte-correcting code $C_2$ with $s_i$ bytes of size $b_i'$, $1 \le i \le m$, and $2^\rho(2^{b_2}+1)$ bytes of size $b_2$. Furthermore, $R_k$, $1 \le k \le 2^{b_1} - 1$, of Construction E is the matrix

$$\begin{bmatrix} 0 \\ A_0 \\ A_{i_k} \end{bmatrix}.$$

## ACKNOWLEDGMENT

The author would like to thank the referees for their constructive comments and suggestions.

## REFERENCES

[1] C. L. Chen, "Error-correcting codes with byte error detection capability," *IEEE Trans. Comput.*, vol. C-32, pp. 615–621, 1983.
[2] ——, "Byte oriented error-correcting code for semiconductor memory systems," *IEEE Trans. Comput.*, vol. C-35, pp. 646–648, 1986.
[3] G. Cohen, I. Honkala, S. Litsyn, and A. Lobstein, *Covering Codes*. Amsterdam, The Netherlands: North-Holland, 1997.
[4] T. Etzion and G. Greenberg, "Constructions of perfect mixed codes and other covering codes," *IEEE Trans. Inform. Theory*, vol. 39, pp. 209–214, 1993.
[5] S. W. Golomb, *Shift Register Sequences*. Laguna Hills, CA: Aegean Park, 1982.
[6] O. Heden, "A generalized Lloyd theorem and mixed perfect codes," *Math. Scand.*, vol. 37, pp. 13–26, 1975.
[7] ——, "A new construction of group and nongroup perfect codes," *Inform. Contr.*, vol. 34, pp. 314–323, 1977.
[8] M. Herzog and J. Schönheim, "Linear and nonlinear single-error correcting perfect mixed codes," *Inform. Contr.*, vol. 18, pp. 364–368, 1971.
[9] ——, "Group partition, factorization and the vector covering problem," *Canad. Math. Bull.*, vol. 15, pp. 207–214, 1972.
[10] S. J. Hong and A. M. Patel, "A general class of maximal codes for computer applications," *IEEE Trans. Comput.*, vol. C-21, pp. 1322–1331, 1972.
[11] B. Lindstrom, "On group and non-group perfect codes in $q$ symbols," *Math. Scand.*, vol. 25, pp. 149–158, 1969.
[12] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: North-Holland, 1977.
[13] T. R. N. Rao and E. Fujiwara, *Error-Control Coding for Computer Systems*. London, U.K.: Prentice-Hall, 1989.

# On Integer-Valued Rational Polynomials and Depth Distributions of Binary Codes

Chris J. Mitchell, *Member, IEEE*

***Abstract*—The notion of the depth of a binary sequence was introduced by Etzion. In this correspondence we show that the set of infinite sequences of finite depth corresponds to a set of equivalence classes of rational polynomials. We go on to characterize infinite sequences of finite depth in terms of their periodicity. We conclude by giving the depth distributions for all linear cyclic codes.**

***Index Terms*—Cyclic code, depth, depth distribution, derivative, linear complexity.**

## I. INTRODUCTION

In this correspondence we are concerned with considering the *depths* of binary sequences, where depth is as defined by Etzion, [1]. Etzion showed that a linear code of dimension $k$ contains codewords of $k$ distinct depths, and also gave the distribution of codeword depths for certain classes of codes.

We first show that the set of infinite sequences of finite depth corresponds to a set of equivalence classes of rational polynomials. We secondly establish an equivalence between infinite sequences of finite depth and sequences of specified periodicity. Finally, we give the depth distributions for all linear cyclic codes, generalizing the results in [1].

## II. DEFINITIONS AND PRELIMINARY REMARKS

### A. Binary Sequences

Suppose $\boldsymbol{s} = (s_i)$ $(i \ge 0)$ is a binary sequence (either finite or infinite). Then we say $\boldsymbol{s}$ is periodic with period $t$ $(t > 0)$ if $s_i = s_{i+t}$ for every $i$ $(i \ge 0)$. If $t$ is the smallest positive integer for which $\boldsymbol{s}$ is periodic with period $t$, then $\boldsymbol{s}$ is said to have least period $t$ (in which case $\boldsymbol{s}$ has period $t'$ if and only if $t|t'$).

Manuscript received March 21, 1997; revised January 14, 1998.

The author is with the Information Security Group, Royal Holloway, University of London, Egham, Surrey TW20 0EX, U.K.

Publisher Item Identifier S 0018-9448(98)06750-9.