Robust Generalized Punctured Cubic Codes

Yaara Neumeier and Osnat Keren

***Abstract*— Security-oriented codes are used in cryptographic devices to maximize the probability of detecting fault injection attacks. This paper introduces a new class of binary security-oriented codes of rate >1/2. The codes are derived from the cubic code by applying a linear transformation on the codewords before puncturing. The codes are systematic and robust in that any nonzero error can be detected with a probability >0. The error masking probability of the codes is upper bounded by 2^{-r+1} where r is the number of redundancy bits. It is shown that in some cases, by choosing the proper transformation and puncturing matrices, it is possible to increase the minimum distance of the code, or to reduce the maximal error masking probability to meet its lower bound.**

***Index Terms*— Robust codes, security, undetected error probability, error masking probability, puncturing, fault injection attacks.**

I. Introduction

CLASSIC coding theory addresses the problem of reliability; i.e., the reliability of information transmitted over a noisy channel or stored in a storage medium. In classic coding theory, the errors are assumed to be *random* and the probability that the channel will introduce an error is relatively small. Consequently, these codes are designed to protect data from errors of small multiplicity. A different class of problems addresses the security of data that goes through a noiseless (or even a noisy) channel. Here we assume that errors are injected by an attacker, and thus there is no restriction on the multiplicity of the error.

Security-oriented codes are used, for example, to detect active fault analysis attacks on the data that flow between the modules of cryptographic devices [3], [14], [16]. Each time a module is activated, it generates a codeword. The attacker, who *knows the codebook* but cannot predict which codeword will appear at the outputs, distorts the outputs of the device by injecting an error vector [2]. Regardless of what the error is, there should be at least one codeword that can detect its presence.

The attacker is assumed to be 'lazy' [2]; i.e., the data change much faster than the error. In this sense, the error can be considered as *fixed* and the information is considered as random. In turn, the definition of the undetected error probability in the context of security-oriented codes is different; the undetected error probability in classic coding theory is defined as the probability that an error vector will map a *given codeword* c to another codeword, i.e.

$$P(c) = \sum_{e \neq 0} \Pr(e) \Pr(c + e \in \mathcal{C}).$$

In security-oriented coding, the undetected error probability is the probability that a *given error* e will map a codeword to another codeword, that is,

$$Q(e) = \sum_{c \in \mathcal{C}} \Pr(c) \Pr(c + e \in \mathcal{C}).$$

In order to distinguish between the two probabilities, we refer to $Q(e)$ as the *error masking probability*.

The detection kernel of a code is denoted by K_d. The kernel consists of all the error vectors that are never detected; that is, $K_d = \{e | Q(e) = 1\}$. Codes that can detect any nonzero error vector e with probability greater than zero ($Q(e) < 1$) are called *robust codes*. Partially robust codes are codes whose detection kernel is of size $1 < |K_d| < |\mathcal{C}|$.

In applications where all the codewords are equally likely to occur, the error masking probability is $Q(e) = |\{c|\ c, c + e \in \mathcal{C}\}|/|\mathcal{C}|$. In binary codes, the number of codewords that mask an error is even. The sum of $Q(e)$ over all the nonzero error vectors equals $|\mathcal{C}| - 1$, and hence, the average $Q(e)$ equals $(|\mathcal{C}|-1)/(2^n-1)$, where n is the length of the code. Similarly, the average error masking probability over all the error vectors (including the all-zero vector) equals $|\mathcal{C}|/2^n$. The maximal error masking probability of detectable errors is denoted by Q_{mc}. For binary robust codes of size $|\mathcal{C}| = 2^k$, Q_{mc} is lower bounded by [18]

$$Q_{mc} \geq \max\left\{\frac{2}{2^k}, \frac{2^k}{2^n}\right\}.$$

Consequently, a robust binary code with minimal error masking probability has a rate smaller or equal to $k/(2k-1)$. If the code rate is greater than one half, the code may become partially robust and/or its maximal error masking probability may become larger than $2/2^k$. Robust and partially robust nonlinear codes of rate greater than one half were presented in [7], [9], and [10].

Security-oriented codes may have error correction capability [6], [17], [18]. As shown in these papers, nonlinear multi-error correcting codes can be constructed by concatenating linear and nonlinear redundant bits. For example, the generalized Phelps code [13] and Vasilev code [15] are partially robust nonsystematic codes of rate larger than one half and a minimum distance of $d = 3$. However, their error masking probability is relatively high [18].

Manuscript received January 6, 2013; revised January 6, 2014; accepted February 24, 2014. Date of publication March 11, 2014; date of current version April 17, 2014. This work was supported by the Israel Science Foundation under Grant 1200/12. This paper was presented in part at the 2012 18th IEEE International On-Line Testing Symposium.

The authors are with the Faculty of Engineering, Bar-Ilan University, Ramat Gan, 52900, Israel (e-mail: nyaara@yahoo.com; osnat.keren@biu.ac.il).

Communicated by N. Kashyap, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2014.2310464

The cubic code is an optimal robust systematic code of rate one half; it consists of the columns of the check matrix of a two-error correcting binary BCH code and the all-zero codeword. A class of systematic codes based on the cubic code, and of rate greater than one half was presented by Karpovsky and Taubin in [10]. The Karpovsky-Taubin (KT) code is robust if the number of redundancy bits is $r = k$ or $k - 1$. Another technique that is used to increase the rate of the cubic code is puncturing [1], [12]. In this paper we generalize the idea of puncturing by applying a linear transformation on the codewords before deleting part of the redundancy bits. We show that unlike KT codes, for any $r > 1$ the generalized punctured cubic code *remains robust*; the error masking probability of the code may take one of three possible values $0, 2^{-r}$ or 2^{-r+1}. In some cases, by choosing the proper puncturing matrix, it is possible to increase the minimum distance of the code to $d \geq 2$, or to reduce the error masking probability so as to meet the lower bound.

This paper is organized as follows: The generalized punctured cubic code is presented in Section II. The properties of the conventionally-punctured cubic code are discussed in Section III, and generalized puncturing is studied in Section IV. The minimum distance of the generalized punctured cubic code is discussed in Section V. Section VI concludes the paper.

II. Punctured Cubic Codes

A codeword in a binary systematic code $\mathcal{C}(n, k)$ of length $n = k + r$ and size 2^k has two parts: an information part and a redundancy part. Each part can be referred to as an element of a finite field or as a binary vector. For example the information part is a vector in k-dimensional space $\mathbb{F}_2^k$, ($\mathbb{F}_2 = GF(2)$) which can also be referred to as an element of the finite field $\mathbb{F}_{2^k} = GF(2^k)$. In this paper we use both representations. For example, the expression $x^3 P$ where P is a binary $k \times r$ matrix, should be read as: refer to x as an element in $\mathbb{F}_{2^k}$ and compute x^3, then refer to the result as a vector in $\mathbb{F}_2^k$ and multiply it by the matrix P modulo 2, the outcome of this operation is an element in $\mathbb{F}_{2^r}$. Note that addition of two elements of $\mathbb{F}_{2^k}$ means coordinate-wise addition modulo 2.

The cubic code (x, x^3), $x \in \mathbb{F}_{2^k}$ is a robust code of rate one-half. It is possible to increase the code rate by puncturing; i.e., by deleting some redundancy bits. In this paper we generalize the concept of puncturing as follows:

Construction 1 (Punctured Cubic Code): Let Λ be a non-singular binary $k \times k$ matrix of rank k. Let P be a binary $k \times r$ matrix of rank $r \leq k$ and let J be a binary $k \times r$ matrix. The (generalized) punctured cubic $\mathcal{C}(k + r, k)$ code is,

$$\mathcal{C} = \{(x, w) | x \in \mathbb{F}_{2^k}, w = (x\Lambda)^3 P + xJ \in \mathbb{F}_{2^r}\}. \quad (1)$$

Denote the all-zero matrix by $\mathbf{0}$ and the identity matrix by I. If $J = \mathbf{0}_{k\times r}$, $\Lambda = I_{k\times k}$, and the Hamming weight of each column of P equals one, the mapping is equivalent to conventional puncturing. That is, the $k - r$ redundancy bits corresponding to the zero rows in P are deleted from the cubic (x, x^3) code.

Note that this construction is different from the KT code and from the generalized KT code[1] in that the codes differ in the order of the linear and non-linear operations. The order of operations in the KT and the generalized KT codes creates a kernel of dimension $k - r$, and hence, makes them partially robust. In contrast, here, the order of operations (a nonlinear operation followed by a linear operation) makes the code robust.

Let $c = (x, w) \in \mathcal{C}$ be a codeword and let $e = (e_x, e_w)$ be a nonzero error vector, $e_x \in \mathbb{F}_{2^k}, e_w \in \mathbb{F}_{2^r}$. The error vector will be masked if $(x + e_x, w + e_w) \in \mathcal{C}$. In other words, e will be masked by c if

$$((x + e_x)\Lambda)^3 P + (x + e_x)J = (x\Lambda)^3 P + xJ + e_w, \quad (2)$$

One of the following three scenarios may occur:

1) The error will always be detected. That is, there exists no $x \in \mathbb{F}_{2^k}$ that solves Eq. 2, and thus, $Q(e) = 0$. The set of errors of this type is denoted by E_a.
2) The error will never be detected. That is, $e \in K_d$. This occurs when Eq. 2 is fulfilled for all $x \in \mathbb{F}_{2^k}$.
3) The error will be detected with probability $0 < 1 - Q(e) < 1$. That is, there exists at least one $x \in \mathbb{F}_{2^k}$ that solves Eq. 2 and there is at least one $x' \in \mathbb{F}_{2^k}$ that does not solve Eq. 2 (i.e., x' detects the presence of the error). The set of errors of this type is denoted by E_p.

Definition 1 (Optimal Robust Code): A binary robust code with k information bits and r redundancy bits having $Q(e) \leq \max(2^{-k+1}, 2^{-r})$, for all $e \neq 0$, is called an optimal robust code.

Let $f(z)$ be the characteristic function of the code; i.e., $f(z) = 1$ if $z \in C$ and it equals zero otherwise. Denote by $R(\tau)$ the autocorrelation of the characteristic function of the code,

$$R(\tau) = \sum_{z \in \mathbb{F}_2^{k+r}} f(z) f(z + \tau).$$

Denote by $X(e)$ the set of x's that mask the error e. The size of $X(e)$ is equal to the value of the autocorrelation function at $\tau = e$. Hence, the error masking probability (for equally likely codewords) is

$$Q(e) = \frac{|X(e)|}{2^k} = \frac{R(e)}{2^k}.$$

Example 1: Consider the case where $k = 3$, $\Lambda = I$ and $J = \mathbf{0}$. Table I presents the codewords of three codes derived from the cubic code by puncturing the last $k - r$ bits (i.e. $P^T = [I_{r\times r} | \mathbf{0}_{r\times(k-r)}]$). The autocorrelation functions of the three codes are shown in Figs. 1–3. The error vectors are identified by their integer value (e.g. (010011) = 19). It is clear from the figures that code $\mathcal{C}(6, 3)$ is optimal, since its autocorrelation function is flat and thus $Q(e)$ reaches the minimal possible value (in binary codes the autocorrelation values are even). Code $\mathcal{C}(5, 3)$ is robust but not optimal, and code $\mathcal{C}(4, 3)$ is partially robust since in one nonzero position the autocorrelation value is 2^k.

[1]The generalized KT code is $\mathcal{C} = \{(x, w) | x \in \mathbb{F}_{2^k}, w = (xP)^3 + a(xP)^2 + b(xP) + c \in \mathbb{F}_{2^r}\}$ where P is a full rank $k \times r$ matrix and $a, b, c \in \mathbb{F}_{2^r}$ [6].

TABLE I
THE CODES IN EXAMPLE 1

Code	$\mathcal{C}(6,3)$	$\mathcal{C}(5,3)$	$\mathcal{C}(4,3)$
Redundancy bits	$r=k=3$	$r=2$	$r=1$
codewords	(000|000) (001|001) (010|011) (011|100) (100|101) (101|110) (110|111) (111|010)	(000|00) (001|00) (010|01) (011|10) (100|10) (101|11) (110|11) (111|01)	(000|0) (001|0) (010|0) (011|1) (100|1) (101|1) (110|1) (111|0)
Example of $e \in E_a$	(000|001)	(000|01)	(000|1)
$\|E_a\|$	35	9	2
Example of $e \in K_d \setminus 0$	none	none	(100|1)
$\|K_d\|$	1	1	2
Example of $e \in E_p$	(001|001)	(001|00)	(001|0)
$\|E_p\|$	28	22	12

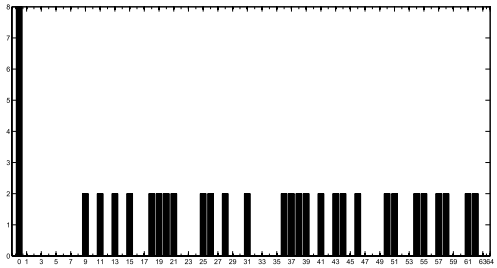

Fig. 1. Autocorrelation values of a punctured cubic code $\mathcal{C}(6,3)$. The 28 nonzero errors that are masked with probability $2/2^3$ are the ones having $R(e)=2$.

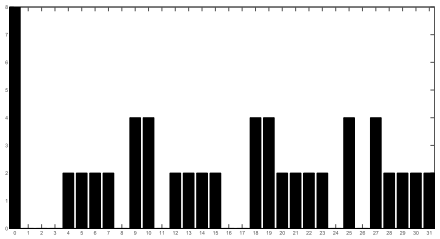

Fig. 2. Autocorrelation values for punctured cubic code $\mathcal{C}(5,3)$. There are two types of nonzero errors in E_p: 6 errors are masked with probability $4/2^3$ and 16 errors are masked with probability $2/2^3$.

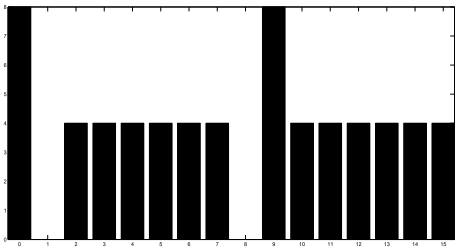

Fig. 3. Autocorrelation values of a punctured cubic code $\mathcal{C}(4,3)$. The code is not robust, a single nonzero error $e=9=(1001)$ is never detected since $R(e)=8$, and 12 errors are masked with probability $4/2^3$.

The autocorrelation function $R_{\mathcal{C}(5,3)}$ can be derived from the autocorrelation $R_{\mathcal{C}(6,3)}$ by adding the autocorrelation values in two adjacent positions (see Figs. 1 and 2). For example, $R_{\mathcal{C}(5,3)}(0)=R_{\mathcal{C}(6,3)}(0)+R_{\mathcal{C}(6,3)}(1)$, $R_{\mathcal{C}(5,3)}(1)=R_{\mathcal{C}(6,3)}(2)+R_{\mathcal{C}(6,3)}(3)$, etc. Similarly, $R_{\mathcal{C}(4,3)}$ (shown in Fig. 3) can be derived from the autocorrelation $R_{\mathcal{C}(6,3)}$ by summing the autocorrelation values of four adjacent positions. In general, one can calculate the autocorrelation values of a punctured cubic code $\mathcal{C}(k+r,k)$ from the autocorrelation values of the cubic $\mathcal{C}(2k,k)$ code, but the computational complexity will be exponential in k. In Section III we show that it is possible to determine these values analytically; i.e., without calculating all the 2^{2k} autocorrelation values of the cubic $\mathcal{C}(2k,k)$ code.

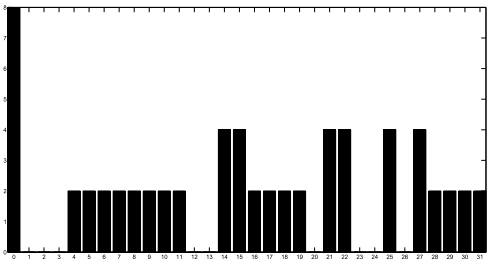

Fig. 4. Autocorrelation values for generalized punctured cubic code $\mathcal{C}(5,3)$ constructed by P_2. There are two types of errors in E_p: six errors are masked with probability $4/2^3$ and 16 errors are masked with probability $2/2^3$.

When a linear transformation is applied on a Boolean function, its autocorrelation values are permuted [8]. An interesting question is whether it is possible to improve the code's properties by applying a linear transformation on the codewords before puncturing. The following example illustrates this idea:

Example 2: Let $k=3, r=2, \Lambda=I$ and $J=\mathbf{0}$. Consider two codes C_1 and C_2 constructed with the matrices P_1 and P_2,

$$P_1^T=\begin{pmatrix}1&0&0\\0&1&0\end{pmatrix} \quad P_2^T=\begin{pmatrix}1&1&1\\0&1&0\end{pmatrix}.$$

The corresponding autocorrelation functions are shown in Figs. 2 and 4. The two codes have the same error masking probabilities and the same number of errors are associated with each probability. Yet, a specific error may or may not be associated with the same error masking probability. For example, $R_1(7)=R_2(7)=2$, but $R_1(8)=0\neq R_2(8)=2$.

Clearly, there are many matrices P of dimension $k\times r$ and rank r. For example, for $k=3, r=2$ there are $\binom{7}{2}$ such different matrices. Can one decrease the maximal error masking probability of the code by choosing a proper P? In Section IV we show that for odd values of k, it is impossible to do so. However, for even values of k, it is possible to derive better codes.

Finally, consider the $\mathcal{C}(5,3)$ code from Ex. 1. The autocorrelation function R equals zero at nine positions $1,2,3,8,11,16,17,24$ and 26. Therefore, nine errors are always detected: $00001, 00010, 00011, \ldots, 11010$. The error 00100 is not included in this set and hence the minimum distance of the code is $d=1$. The question we address in Section V is whether it is possible to find Λ, J and P matrices that will move the zeros of the autocorrelation to positions of Hamming weight greater than one. We show that for large enough r there exist Λ, J and P matrices for which the code has $Q(e)=0$ for all e's of Hamming weight one, i.e. the code has $d\geq 2$.

III. Conventional Puncturing

In this section we discuss the properties of the code obtained by applying conventional puncturing on the cubic code. That is, $J = \mathbf{0}$, $\Lambda = I$ and $P = (I_{r\times r}|\mathbf{0}_{r\times(k-r)})^T$, and the code becomes (x, x^3P). The general case is discussed in the following sections.

The error masking probability depends on the number of distinct solutions in $\mathbb{F}_{2^k}$ to the error masking equation (Eq. 2) that becomes $(x+e_x)^3P = x^3P + e_w$. Equivalently, $Q(e)$ is determined by the number of solutions in $\mathbb{F}_{2^k}$ to the equation

$$(x^2e_x + xe_x^2 + e_x^3)P = e_w. \tag{3}$$

To simplify the analysis we transform Eq. 3 into the following equation over $\mathbb{F}_{2^k}$

$$x^2e_x + xe_x^2 + e_x^3 = \tilde{b}, \tag{4}$$

where $\tilde{b} \in \mathbb{F}_{2^k}$ is obtained from $e_w \in \mathbb{F}_2^r$ by appending an arbitrary tail $b \in \mathbb{F}_2^{k-r}$. That is,

$$\tilde{b} = (e_w, b) = (e_{w,r-1}, \ldots e_{w,1}, e_{w,0},\ b_{k-r-1}, \ldots b_1, b_0).$$

The following lemma defines the relationship between the two equations.

Lemma 1: Denote by $X(e)$ the set of x's in $\mathbb{F}_{2^k}$ that satisfy Eq. 3 for a given error vector e, and denote by $X_b(e)$ the set of x's that satisfy Eq. 4 for a given e and b. Then,

$$Q(e) = \frac{|X(e)|}{2^k} = \frac{\sum_{b\in\mathbb{F}_2^{k-r}} |X_b(e)|}{2^k}.$$

Proof: Let $c = (x, w)$ be the transmitted codeword and denote by $e = (e_x, e_w)$ an error vector. Multiplication by P maps an element from $\mathbb{F}_{2^k}$ to an element of $\mathbb{F}_{2^r}$ by deleting the last $k-r$ coordinates of the element. Consequently, the last $k-r$ coordinates of $x^2e_x + xe_x^2 + e_x^3$ have no effect on Eq. 3. Therefore, if for a given error e and a given b, there exists an $x \in X_b(e)$ that solves Eq. 4, then the same x solves Eq. 3. Hence, $X_b(e) \subseteq X(e)$.

On the other hand, if $x \in X(e)$, then there exists a *unique* $b \in \mathbb{F}_2^{k-r}$ such that x solves Eq. 4 with $\tilde{b} = (e_w, b)$. Namely, if $b_1 \neq b_2$ then $X_{b_1}(e) \cap X_{b_2}(e) = \phi$. Therefore, for a given error vector e, the size of $X(e)$ (the set of x's that satisfy Eq. 3) is equal to

$$|X(e)| = |\bigcup_{b\in\mathbb{F}_2^{k-r}} X_b(e)| = \sum_{b\in\mathbb{F}_2^{k-r}} |X_b(e)|.$$

■

Note that if $e_x \neq 0$, Eq. 4 can be further simplified; Define $z = xe_x^{-1}$, $a = 1 + \tilde{b}e_x^{-3}$ and $e_x^{-3} = \alpha^s$. Using these notations, Eq. 4 becomes,

$$z^2 + z + a = 0. \tag{5}$$

Definition 2 (Trace): Let $\beta \in \mathbb{F}_{2^m}$. The sum $Tr(\beta) = \sum_{j=0}^{m-1} \beta^{2^j} \in \mathbb{F}_2$ is called the trace of β.

Since $\tilde{b} = (e_w, b)$ is an element of $\mathbb{F}_{2^k}$, it can be represented as a sum of the first k powers of a primitive element $\alpha \in \mathbb{F}_{2^k}$,

$$\tilde{b} = \sum_{i=0}^{r-1} e_{w,i}\alpha^{i+k-r} + \sum_{i=0}^{k-r-1} b_i\alpha^i,$$

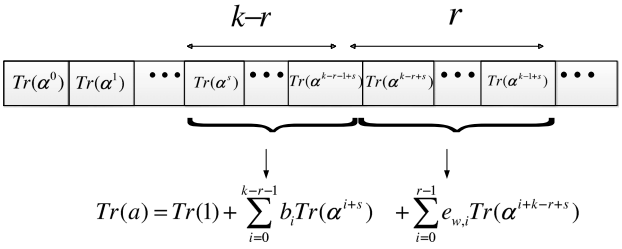

Fig. 5. Illustration of Eq. 6.

and,

$$\begin{aligned} Tr(\tilde{b}e_x^{-3}) &= \sum_{i=k-r}^{k-1} e_{w,i-k+r)} Tr(\alpha^{i+s}) + \sum_{i=0}^{k-r-1} b_i Tr(\alpha^{i+s}) \\ &= \sum_{i=0}^{r-1} e_{w,i} Tr(\alpha^{i+k-r+s}) + \sum_{i=0}^{k-r-1} b_i Tr(\alpha^{i+s}). \end{aligned} \tag{6}$$

The value of $Tr(\tilde{b}e_x^{-3})$ is a weighted sum of the traces of k successive powers of α, starting from α^s up to α^{s+k-1}, (as illustrated in Fig. 5). Define,

$$T_s = (Tr(\alpha^{s+k-1}), \ldots, Tr(\alpha^{s+1}), Tr(\alpha^s)) = (t_w, t_b),$$

where $t_w \in \mathbb{F}_2^r$ and $t_b \in \mathbb{F}_2^{k-r}$. Using this notation

$$Tr(\tilde{b}e_x^{-3}) = T_s \cdot \tilde{b}^T = (t_w, t_b)\cdot(e_w, b)^T.$$

By Prop. 2,

Lemma 2: For a given error vector $e = (e_x \neq 0, e_w)$ and a given b, Eq. 5 has two solutions in $\mathbb{F}_{2^k}$ iff

$$Tr(1) + (t_w, t_b)\cdot(e_w, b)^T = 0. \tag{7}$$

Theorem 1: Let C be a punctured cubic code defined by $J = \mathbf{0}$, $\Lambda = I$ and $P = (I_{r\times r}|\mathbf{0}_{r\times(k-r)})^T$. Then, for $1 < r < k$ the code is robust - the kernel contains only the all-zero word. The $2^n - 1$ nonzero error vectors are detected with probability 1 or $1 - 2^{-r}$ or $1 - 2^{-r+1}$.

The proof of the theorem is given in Appendix. The proof is based on the following properties:

Property 1 ([4]): The trace equals one for one half of the elements in $\mathbb{F}_{2^m}$, and equals zero for the other half.

Property 2 ([4]): Let $\beta \in \mathbb{F}_{2^m}$. The quadratic equation $x^2 + x + \beta = 0$ has two roots in $\mathbb{F}_{2^m}$ iff $Tr(\beta) = 0$.

Property 3 ([6]): Let α be a primitive element in $\mathbb{F}_{2^m}$ and let $B = (b_{m-1}, \ldots, b_0)$ be a binary vector of length m. Then, there is a unique element $\gamma \in \mathbb{F}_{2^m}$ such that $b_i = \mathrm{Tr}(\gamma\,\alpha^i)$, for $0 \leq i < m$.

Property 4: Let $b \in \mathbb{F}_{2^m}\backslash\{0\}$. If m is *odd* then b has a single cubic root. If m is *even*, then b may have three different cubic roots, or no root at all.

By Theorem 1, the error masking probability of a nonzero error can take only three possible values $0, 2^{-r}$, and 2^{-r+1}. Namely, $Q(e)$ is upper bounded by 2^{-r+1}. Thus, all the punctured codes having $r > 1$ are robust.

Next we address the question of how many error vectors are associated with each value. Theorem 2 shows that most of the error vectors are detected with probability $1 - 2^{-r}$.

TABLE II
PUNCTURING MATRICES FOR OPTIMAL CODES

k	r	r_{UB}	P^T	k	r	r_{UB}	P^T
4	2	3	A, 5	6	3	4	11, A, 5
8	4	6	82, 26, 13, 9	10	4	8	21C, 101, 82, 3B
12	4	10	412, 276, 11E, E7	14	4	12	224D, 122E, 161, 85
16	4	14	8DBD, 4DA7, 2121, 1BB3				

Theorem 2 (Odd Values of k): Let C be a punctured cubic code with $J = \mathbf{0}$, $\Lambda = I$ and $P = (I_{r\times r}|\mathbf{0}_{r\times(k-r)})$, where k is odd and $r > 1$. Then, $2^r(2^k - 2^r)$ error vectors are detected with probability $1 - 2^{-r}$, $(2^r - 1)(2^{r-1})$ error vectors are detected with probability $1 - 2^{-r+1}$, the remaining $(2^r - 1)(2^{r-1} + 1)$ nonzero errors are always detected.

The proof of the theorem is given in Appendix.

Remark 1: If $k = r$ there are no error vectors for which $Q(e) = 2^{-r}$ since in this case $2^{k+r} - 2^{2r} = 0$. Namely, for $r = k$ the error masking probability meets the lower bound of $2/2^k$.

Remark 2: For $k = 2sr$ (s is an integer) the Quadratic-Sum code is optimal [9]. However, for $k = 2sr - \Delta$ where $r < \Delta < 2r$ the Quadratic-Sum code is not optimal, its error masking probability is $Q_{mc} \geq 2^{-r+1}$. In this sense, for $\Delta > r$, punctured cubic codes are at least as good as Quadratic-Sum codes. For example, the quadratic code for $k = 10$, $r = 4$ is not optimal; however, as shown in Table II, there is a P matrix that gives an optimum generalized punctured cubic code for these parameters.

IV. GENERALIZED PUNCTURING WITH AN ARBITRARY MATRIX P

A generalized punctured cubic code is defined with a matrix J, a nonsingular matrix Λ and an $k\times r$ binary matrix P of rank r. In what follows we show that for even values of k, it is possible to improve (reduce) the error masking probability by choosing a proper P. However, for odd values of k, it is impossible to do so.

Let $J = \mathbf{0}$ and $\Lambda = I$. (The general case where J is a $k\times r$ matrix and Λ is a $k\times k$ matrix of rank k will be discussed in section V). The construction (x, x^3P), with an arbitrary $k\times r$ matrix P can be interpreted as a linear transformation applied on the redundancy bits (x^3) before its $k - r$ least significant bits are deleted. The linear transformation is performed by multiplying x^3 by a *nonsingular* $k\times k$ matrix π,

$$\pi = \left(P, \bar{P}\right), \tag{8}$$

where $\bar{P}$ is a $k\times(k-r)$ matrix of rank $k - r$. $\bar{P}$ is a complementary matrix of P in terms of rank, i.e. $\bar{P}$ is chosen so that the rank of π is k (and thus $\bar{P}$ is not unique). Denote the inverse of π as

$$\pi^{-1} = \begin{pmatrix} \bar{H} \\ H \end{pmatrix}, \tag{9}$$

where $\bar{H}$ is an $r\times k$ matrix and H is a $(k-r)\times k$ matrix.

Lemma 3: Let $\tilde{b} = (e, b)$ be the vector constructed by appending a tail $b \in \mathbb{F}_2^{k-r}$ to the error vector e. Denote by R_P the autocorrelation function of the $(k + r, k)$ code constructed with matrix P, and by R_π the autocorrelation function of the $(2k, k)$ code constructed with π. Then, $R_P(e) = \sum_{b\in\mathbb{F}_2^{k-r}} R_\pi(\tilde{b})$.

The proof is similar to the proof of Lemma 1.

Theorem 3 (Odd k): Let C be the code constructed with a binary $k\times r$ matrix P of rank $r > 1$ and $J = \mathbf{0}, \Lambda = I$. Then the kernel of the code is of dimension 0. If k is odd, then $2^r(2^k - 2^r)$ error vectors are detected with probability $1 - 2^{-r}$, $(2^r - 1)(2^{r-1})$ error vectors are detected with probability $1 - 2^{-r+1}$, The remaining $(2^r - 1)(2^{r-1} + 1)$ nonzero errors are always detected.

Proof: Any error vector of the form $(e_x = 0, e_w \neq 0)$ is always detected.

As for $e_x \neq 0$, the error masking equation over $\mathbb{F}_{2^r}$ is $(x^2e_x + xe_x^2 + e_x^3)P = e_w$, and the equivalent error masking equation over $\mathbb{F}_{2^k}$ is

$$(x^2e_x + xe_x^2 + e_x^3)\pi = \tilde{b} \tag{10}$$

or,

$$x^2e_x + xe_x^2 + e_x^3 = \tilde{b}\pi^{-1}. \tag{11}$$

Define $\alpha^s = {e_x}^{-3}$. From Prop. 2, the error masking equation has two solutions iff $Tr(1 + (\tilde{b}\pi^{-1})\alpha^s) = 0$. Using the notations defined in Section III, the error masking equation has solutions iff

$$Tr(1) + e_w\bar{H}T_s^T + bHT_s^T = 0. \tag{12}$$

Matrix H is of rank $k - r$. It can be referred to as a check matrix of a linear code $\mathcal{C}_l$ of length k and dimension r. There are two cases:

- $T_s \in \mathcal{C}_l$, and hence $HT_s^T = \mathbf{0}$.
  Recall that from Prop. 4, for odd values of k, there is a one-to-one mapping between e_x and Ts. Thus, there are exactly $2^r - 1$ nonzero e_x's, and hence, $(2^r - 1)2^r$ distinct error vectors whose corresponding T_s is a codeword in $\mathcal{C}_l$. For these error vectors, the value of $Tr(a)$ does not depend on b. Therefore, Eq. 12 is either satisfied for all the 2^{k-r} b's, or for none of them. Since, $\pi^{-1}T_s \neq 0$, then $\bar{H}T_s^T \neq 0$. Thus, the value of $Tr(a)$ depends only on e_w. In other words, one half of these error vectors (i.e. $(2^r - 1)2^{r-1}$ vectors) are masked with probability $(2\cdot 2^{k-r})/2^k$, and the other half are always detected.
- $T_s \notin \mathcal{C}_l$.
  There are $2^k - 2^r$ e_x's, for which $HT_s^T \neq 0$. Therefore, $Tr(a)$ equals zero for one half of the b's, and equals one for the other half. Namely $(2^k - 2^r)2^r$ error vectors are masked with probability 2^{-r}. ■

From Prop. 4 we conclude that for *even* values of k, $\alpha^s = e_x^{-3}$ may take only one third of the nonzero values in $\mathbb{F}_{2^k}$. Therefore, the number of error vectors associated with each

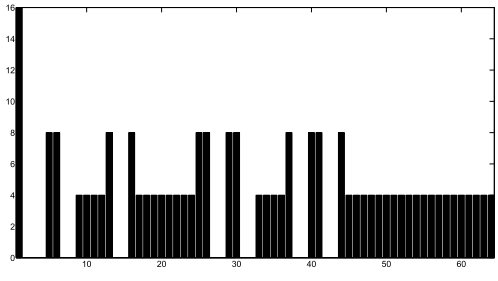

Fig. 6. Autocorrelation values of the code $(x, x^3 P_1)$. Fifteen error vectors are always detected, 36 error vectors are masked with probability $Q(e) = 2^{-2}$ and 12 are masked with $Q(e) = 2^{-1}$.

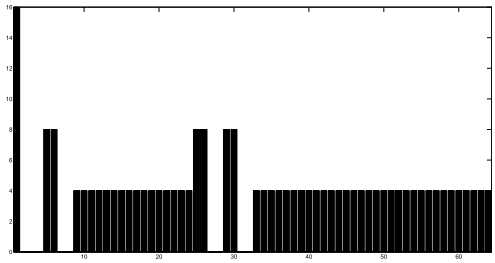

Fig. 7. Autocorrelation values of the code $(x, x^3 P_2)$. Nine error vectors are always detected, 48 error vectors are masked with probability $Q(e) = 2^{-2}$ and 6 are masked with $Q(e) = 2^{-1}$.

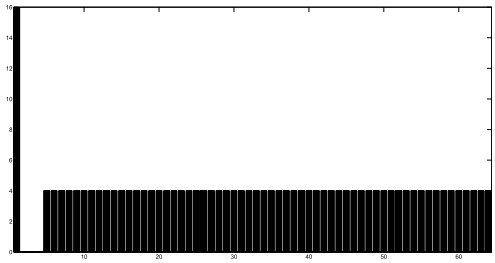

Fig. 8. Autocorrelation values of the code $(x, x^3 P_3)$. Three error vectors are always detected, sixty error vectors are masked with probability $Q(e) = 2^{-2}$ and no error vectors are associated with $Q(e) = 2^{-1}$.

$Q(e)$ depends on the chosen matrix P. To see this consider the following example.

Example 3: Let $k = 4, r = 2$. Consider three codes $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_3$ defined by the puncturing matrices

$$P_1^T = \begin{pmatrix} 1\ 0\ 0\ 0 \\ 0\ 0\ 0\ 1 \end{pmatrix},\ P_2^T = \begin{pmatrix} 1\ 0\ 0\ 0 \\ 0\ 1\ 0\ 0 \end{pmatrix},$$

$$P_3^T = \begin{pmatrix} 1\ 1\ 0\ 0 \\ 0\ 0\ 1\ 0 \end{pmatrix}.$$

The autocorrelation functions of these codes are shown in Figs. 6–8. It is clear from the figures that one can decrease the number of error vectors associated with $Q(e) = 2^{-r+1}$ and even completely eliminate them by choosing an appropriate puncturing matrix P.

Denote by $\mathcal{T}$ the set of T_s's associated with the e_x^{-3}'s. Clearly, $\mathbf{0} \notin \mathcal{T}$. Any $T_s \in \mathcal{T}$ that satisfies $HT_s^T = 0$ is associated with 2^{r-1} error vectors which are masked with probability 2^{-r+1} (refer to the proof of Th. 3). Therefore, if there exists a check matrix H that defines a linear code $\mathcal{C}_l$ (of length k and dimension r) for which $\mathcal{C}_l \cap \mathcal{T} = \phi$, the corresponding punctured cubic code is optimal. Clearly as r increases the chances of finding a good H decrease.

Matrix H defines the generalized punctured cubic code. In other words, H can be extended to a $k \times k$ matrix by adding r linearly independent rows (denote these rows as $\bar{H}$). The resulting matrix $(\bar{H}, H)^T$ can be referred to as the inverse of matrix π. The leftmost r columns of π form matrix P.

Lemma 4: If there exists a $P_{r\times k}$ matrix that defines an optimal punctured cubic code $\mathcal{C}(k, r)$, then for any $\hat{r} \leq r$ there exists a matrix $\hat{P}_{\hat{r}\times k}$ that defines an optimal $\mathcal{C}(k, \hat{r})$ code.

The correctness of the lemma follows from the fact that by adding a single independent row, say h, to H one can create a new linear code $\hat{\mathcal{C}}_l \subset \mathcal{C}_l$ of dimension $r - 1$. If the linear $\mathcal{C}_l$ code defines an optimal code $\mathcal{C}$, so does the new code $\hat{\mathcal{C}}_l$.

Define $M = \mathbb{F}_2^k \setminus \mathcal{T}$, $|M| = \frac{2}{3}(2^k - 1) + 1$. As was shown in [5], the maximal cardinality of a subset $M \subseteq \mathbb{F}_2^k$, that contains the zero vector and does not contain any r-dimensional subspace of $\mathbb{F}_2^k$ $(r > 0)$ is

$$|M| \leq 2^k - 2^{k-r+1} + 1.$$

Therefore,

Property 5: For $r \leq 2$, there exists a linear code $\mathcal{C}_l \subseteq M$, and hence, a puncturing matrix P for which the punctured cubic code is optimal.

An immediate result stemming from the above is that for even k there always exists a P matrix that gives an optimal robust code for $r = 1$. Denote by R_M the autocorrelation function of the characteristic function of the set M. If M contains a subspace V of dimension w, then for all $v \in V$, $R_M(v) \geq 2^w$. Therefore, the following *naive* upper bound can be derived: If $\mathcal{C}(k, r)$ is an optimal code, r is upper bounded by

$$r \leq r_{UB} = \left\lfloor log_2(max_{t\neq 0}(R_M(t))) \right\rfloor. \tag{13}$$

Table II gives examples of puncturing matrices P^T of optimal codes for even values of k. The rows of the matrices are written as hexadecimal words. The puncturing matrices were constructed by a greedy algorithm (a pseudo code is given in Appendix). It is clear from the table that the gap between the maximal r obtained by the algorithm and the upper bound r_{UB} increases with k. This is not surprising, since the upper bound on r is based only on the *maximal value* of the autocorrelation function $R_M(t)$ and disregards its multiplicity. An exhaustive analysis of all matrices with size up to $k \times r_{UB}$ for $k = 4, 6$ and 8, shows that the maximal value of r for which there exists an optimal punctured cubic code, is equal to the value found by the greedy algorithm. Namely, the maximal r's for $k = 4, 6$ and 8, are equal to $2, 3$ and 4, respectively.

V. The Minimum Distance of the Generalized Punctured Cubic Code

In this section we show that in some cases it is possible to use the fact that the autocorrelation functions of the generalized punctured cubic codes have many zeros to construct codes with a minimum distance greater than one.

The minimum distance d of the code can be defined via its autocorrelation function:

Definition 3 (Minimum Distance): The minimum distance of a code $\mathcal{C}$ is the maximal integer d for which

$$\sum_{e,0<wt(e)<d} R_{\mathcal{C}}(e) = 0.$$

Let $\mathcal{C}$ be a code of length n. Let σ be a nonsingular $n \times n$ matrix. Define the linearly transformed code $\hat{\mathcal{C}}$ as $\hat{\mathcal{C}} = \{c\sigma \,|c \in \mathcal{C}\}$. Let f and $\hat{f}$ be the characteristic functions of the codes, $\hat{f}(z) = f(z\sigma^{-1})$. The autocorrelation function of $\hat{\mathcal{C}}$ is [8]:

$$\begin{aligned}\hat{R}(e) &= \sum_{z\in\mathbb{F}_2^{k+r}} \hat{f}(z)\hat{f}(z+e) \\ &= \sum_{z\in\mathbb{F}_2^{k+r}} f(z)f(z+e\sigma^{-1}) = R(e\sigma^{-1}).\end{aligned}$$

In some cases it is possible to increase the minimum distance of a code by applying a linear transformation σ on the codewords. In fact, this is the role of the $k \times n$ generator matrix G in linear codes. In binary linear codes, a linear code of distance one, $\mathcal{C} = span\{\delta_i\}_{i=r}^{n-1}$ (where δ_i is a binary vector with a single "one" at the i'th position), is transformed by a nonsingular matrix $\sigma = \begin{pmatrix} G \\ \bar{G} \end{pmatrix}$ to the linear subspace $\hat{\mathcal{C}}$ spanned by the rows of $G_{k\times n}$. In terms of the autocorrelation function, the 2^k nonzero values of the autocorrelation function of the original code $\mathcal{C}$ occur at positions of Hamming weight less than or equal to k. The linear transformation matrix σ permutes these autocorrelation values in a way that $\hat{R}(e) = 0$ for all e, $wt(e) < d$.

The same principle can be applied to non-linear codes. As we show next, matrices J and Λ (refer to Def. 1 in Section II) play the role of the generator matrix.

Theorem 4: Let $\mathcal{C}$ be the code (x, x^3P). Denote by E_0 a set of k linearly independent error vectors $\{e_i = (e_{x,i} \neq 0, e_{w,i})\}_{i=1}^{k}$ for which $e_{x,i} \neq e_{x,j}$ for $i \neq j$. If there exists a set E_0 of k errors such that $Q(e_i) = 0$, there exists a non singular matrix Λ and a matrix J that consists of at most k ones, that together with the matrix P define a robust code $\hat{\mathcal{C}} = (x, (x\Lambda)^3P + xJ)$ that has $d \geq 2$.

Proof: All the error vectors of the form $e = (e_x = 0, e_w)$ (and in particular the errors $\{\delta_i\}_{i=0}^{r-1}$) are always detected. There are $2^r - 1$ such vectors, forming (together with the zero vector) a linear subspace of dimension r. For odd values of k, it follows from the proof of Th. 1 that if there exists an error vector $e_i = (e_{x,i} \neq 0, e_{w,i} \neq 0)$ that is always detected, the vector $(e_{x,i}, 0)$ is also always detected. For even values of k, if there exists an error vector $(e_{x,i} \neq 0, e_{w,i} \neq 0)$ that is always detected, there is a $\delta_{j(i)}$ $0 \leq j(i) < r$ such that the vector $(e_{x,i}, \delta_{j(i)})$ is also always detected. Define

$$\sigma = \begin{pmatrix} L & LJ \\ 0 & I \end{pmatrix}, \quad \sigma^{-1} = \begin{pmatrix} \Lambda & J \\ 0 & I \end{pmatrix}$$

where $\Lambda = L^{-1}$, and the i'th row of Λ contains $e_{x,i}$, and the i'th row of J contains zeros for odd k and $\delta_{j(i)}$ for even k (i.e. matrix J is either the zero matrix (for odd k) or a matrix whose rows are of Hamming weight one (for even k)). Then, the linearly transformed code $\hat{\mathcal{C}}$ defined by σ has the property that for all $0 \leq i \leq k + r - 1$,

$$\sum_{i=0}^{k+r-1} \hat{R}(\delta_i) = \sum_{i=0}^{k+r-1} R(\delta_i\sigma^{-1}) = 0.$$

Namely, code $\hat{\mathcal{C}}$ has $d \geq 2$ since any single error is detected by all the codewords. ■

Remark 3: Matrix J was chosen to minimize the implementation cost of the code in hardware (the number of XOR gates is proportional to the number of 'ones' in the matrix).

Note that for odd values of k there are $2^r - 1$ distinct error vectors of the form $(e_x \neq 0, 0)$ that are always detected. Thus, at least r of them are linearly independent.

Corollary 1 (Necessary Conditions for $d \geq 2$): A punctured cubic code $\mathcal{C}(k + r, k)$ with $d \geq 2$ satisfies

$$r \geq \begin{cases} log_2(k+1) & k \text{ is odd,} \\ log_2(k/3+1) & k \text{ is even.} \end{cases}$$

Clearly, the minimum distance of the codes depends on the choice of H (which is equivalent to choosing P). That is, P defines the group E_a of error vectors that are always detected. If P is chosen such that n distinct errors from E_a are linearly independent, then there exist Λ and J that yield a code of $d > 1$. Otherwise, no Λ and J will yield such a code. For example, the following matrix P_1 can define a code with $d = 2$ whereas the matrix P_2 can only define a code with $d = 1$,

$$P_1^T = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}, \; P_2^T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

Lemma 5: If there exists a $P_{r\times k}$ matrix that yields a robust code with $d \geq 2$, then for any $\hat{r} \geq r$ there exists a matrix $\hat{P}_{\hat{r}\times k}$ for which the corresponding punctured cubic code can be transformed into a robust code with $d \geq 2$.

Proof: Let E_0 be the set of k error vectors $(e_x \neq 0, e_w)$ that constitute the k upper rows of σ^{-1}. The elements of E_0 are always detected, namely, $Tr(1) + e_w\bar{H}T_s^T + bHT_s^T = 1$ for all b, or equivalently,

$$\begin{aligned} HT_s^T &= 0 \\ \bar{H}T_s^T &\neq 0, \end{aligned}$$

where T_s is the trace vector associated with e_x^{-3} and H and $\bar{H}$ are the matrices derived from P as described in the previous section. Recall that matrix π is

$$\pi^{-1} = \begin{pmatrix} \bar{H} \\ H \end{pmatrix} = \begin{pmatrix} \bar{\hat{H}} \\ \hat{H} \end{pmatrix},$$

where $\bar{\hat{H}}$ and $\hat{H}$ are $\hat{r} \times k$ and $(k-\hat{r}) \times k$ matrices, respectively. Since $\hat{r} \geq r$ we have

$$\begin{aligned} \hat{H}T_s^T &= 0, \\ \bar{\hat{H}}T_s^T &\neq 0. \end{aligned}$$

Therefore, any e_x can be associated with $\hat{e}_w = (e_w, \mathbf{0}) \in \mathbb{F}_2^{\hat{r}}$ such that $Tr(1) + \hat{e}_w\bar{\hat{H}}T_s^T + bHT_s^T = 1$ for all b. In other

TABLE III
GENERALIZED PUNCTURED ROBUST CODES WITH $d \geq 2$

k	r	P^T	L	J^T
4	2 *	D, 6	4, 1, F, 3	9, 6
5	3 *	5, 13, D	D, 16, 1C, 10, 18	
6	3	3A, 2D, 2B	3E, E, 1, 1E, 3, 6	11, 24, A
7	3 *	76, 2F, 69	75, 54, 40, 3C, 18, 5F, 3	
8	3	F6, 21, BA	FE, B0, F1, 97, 4F, 93, 91, 1A	93, 48, 24
9	4 *	B2, 109, 143, 2B	102, 76, 17A, 19, 1CC, 1B5, AC, 11F, 13	
10	4	A9, 31F, 2AF, 2D4	25D, 347, 2CC, A0, 1E0, 2BF, E0, 180, 30F, 230	88, 244, 10, 123
11	4 *	4B4, 659, 1BB, 12E	717, 166, 7BC, 523, 4C6, 20A, 290, 3BF, 38F, 3C, 3D9	
12	3 *	4E9, 2F2, 134	2A6, 761, BD0, 19B, 7D2, A95, A4D, CDF, 1D1, D76, 6BB, 713	8B6, 309, 440
13	4 *	101F, C35, 211, F7	139F, 32, 31C, 119A, FEB, 9CE, 558, A0F, 37C, 970, 1252, B2, 1247	
14	4	25B6, 15E7, 87B, 3EA	2797, 6B6, 1401, 9F5, 1962, 1B36, 145, FC8, 15B, 1ED8, 1C79, 183B, 1E30, AD7	332A, 8C5, 0, 410
15	4 *	512C, 3141, F40, 91	38C9, 2711, 48C9, 316F, 4CBF, 677B, 3F06, 194E, 2F9F, 62BA, 3FCA, 6E66, 3241, 5199, 181A	
16	4	87A3, 4726, 27A3, 79	A47D, DA92, FE32, 475E, 6F60, 3C60, 5EF8, 3C9A, CCE3, CC8B, 7B00, 623C, 620C, 2B33, DB10, 1	2043, C18C, 1A10, 420

words, matrix $\hat{P}$ constructed from $\hat{H}$ defines a robust code that can be transformed into a code with $d = 2$. ■

Table III presents the matrices P, Λ and J for $4 \leq k \leq 16$. The rows of the matrix are written as hexadecimal words. For example, matrix P_1^T in the example above is written as $3A, 01$. A code whose r meets the lower bound in Cor. 1 is marked with an asterisk. A pseudo code of the greedy algorithm used to construct these matrices is given in Appendix. The complexity of the algorithm is $\mathcal{O}(k2^r)$.

VI. CONCLUSION

The paper presents a class of robust codes of rate greater than one-half. The codes are based on generalized puncturing of the cubic code. The generalized punctured cubic codes are robust for any $1 < r \leq k$. The error masking probabilities of the punctured cubic codes may take one of three possible values: 0, 2^{-r} or 2^{-r+1}. It is shown that for odd values of k, most of the errors are detected with a probability of $1 - 2^{-r}$. For even values of k and small r's it is possible to reduce the worst case error masking probability to 2^{-r} by choosing a proper puncturing matrix. If r is large enough, it is possible to use the fact that some errors are masked with a probability of 2^{-r+1} to increase the minimum distance of the code.

APPENDIX A

Proof of Theorem 1

Let $c = (x, w)$ be the transmitted codeword and denote the error vector by $e = (e_x, e_w)$. The error will be masked if $(x + e_x, w + e_w) \in \mathcal{C}$. One of the following may occur:

- If $e_x = 0$ and $e_w = 0$, no error has occurred.
- If $e_x = 0$ and $e_w \neq 0$, the error will always be detected, $Q(e) = 0$.
- If $e_x \neq 0$, the error will be detected with probability $0 < 1 - Q(e) \leq 1$.

For a given error vector $e = (e_x \neq 0, e_w)$ and a given b, the error masking probability $Q(e)$ is determined by the number of x's that solve Eq. 4. The number of x's depends on the value of $Tr(a)$ as formulated in Eq. 7. Consider the following two cases:

- *Case I:* $e_x \neq 0$ *and* $e_w = 0$. This case can be divided into two sub-cases:
  1) $T_s = (t_w, 0)$. In this case $Tr(a)$ does not depend on the value of b. Therefore, if k is even, then $Tr(1) = 0$ and Eq. 4 has two roots for each possible value of b. Namely, $Q(e) = (2 \cdot 2^{k-r})/2^k = 2^{-r+1}$. If k is odd, there are no roots to Eq. 4 and the error is always detected ($Q(e) = 0$).
  2) $T_s = (t_w, t_b)$ where $t_b \neq 0$. In this case the value of $Tr(a)$ depends only on the value of b. Since b may get any value in $\mathbb{F}_2^{k-r}$, for half of the values of b the second term in Eq. 7 equals 1 and for the other half it equals 0. Thus, Eq. 7 is fulfilled for $2^{k-r}/2$ values of b regardless of the value of $Tr(1)$. Each b that solves Eq. 7 is associated with two distinct x's. Therefore,

$$\sum_{b \in \mathbb{F}_2^{k-r}} |X_b(e)| = \frac{2^{k-r}}{2} \cdot 2,$$

  and the value of $Q(e)$ in this case equals to 2^{-r}.
- *Case II:* $e_x \neq 0$ *and* $e_w \neq 0$. This case can be divided into four sub-cases:
  1) $T_s = (0, 0)$. In this case the value of $Tr(a)$ does not depend on e_w or on the b's. From Prop. 3, there is a unique element $e_x^{-3} \in \mathbb{F}_{2^k}$ that satisfies $Tr(\alpha^i e_x^{-3}) = 0$ for all $0 \leq i < k$; this element is $e_x^{-3} = 0$. Since $e_x \neq 0$ this case is irrelevant.
  2) $T_s = (0, t_b)$ where $t_b \neq 0$. In this case the value of $Tr(a)$ does not depend on e_w but it depends on the b's. In this sense, this case is equivalent to the case where $e_w = 0$. Hence the value of $Q(e)$ is 2^{-r}.
  3) $T_s = (t_w, 0)$ where $t_w \neq 0$. In this case the value of $Tr(a)$ does not depend on the b's. Namely, Eq. 7 which depends only on (e_x, e_w) may be fulfilled for all the 2^{k-r} b's, or for neither of them. When Eq. 7 is fulfilled, since each b is associated with two distinct x's, the value of $Q(e)$ is

$$Q(e) = \frac{2(2^{k-r})}{2^k} = 2^{-r+1},$$

  and when Eq. 7 is not fulfilled then $Q(e) = 0$.
  4) $T_s = (t_w, t_b)$ where $t_w, t_b \neq 0$. In this case the value of $Tr(a)$ depends on both e_w, and b Since b may get any value over $\mathbb{F}_2^{k-r}$, for half of the values of b

TABLE IV
CONSTRUCTION OF P FOR AN OPTIMAL CODE

Set $G = \{\}; V = M;$

While $V \neq \Phi$
1) Pick a random vector $u \in V$
2) If $span(G \cup \{u\}) \setminus M = \Phi$ then $G = G \cup \{u\}$
Else break;
3) $V = V \setminus span(G)$
end

Construct the matrix for the $\mathcal{C}(k, r)$ code (r is the size of G):
1) Find a parity check matrix H for the linear code $\mathcal{C}_l$ spanned by the vectors in G.
2) Calculate π from $\pi^{-1} = \begin{pmatrix} \bar{H} \\ H \end{pmatrix}$ where $\bar{H}$ is any complement matrix (in terms of rank) of H .
2) Extract P from the r right most columns of π, $\pi = (P, \bar{P})$.

TABLE V
CONSTRUCTION OF A PC CODE WITH DISTANCE $d \geq 2$

Set $\Lambda = I_{k\times k}; G = \{\}; E_0 = \{\};$

$j = 0; r = 0;$

While $(j < k)$
1) Pick a random vector $u \in \mathbb{F}_2^k$ and set its j'th bit to one.
2) Calculate $e_x = u\Lambda$. (The multiplication by Λ makes the new e_x linearly independent in the e_xs aggregated in E_0 so far).
3) Calculate the corresponding T_s.
4) Add T_s as a row to the generator matrix of $\mathcal{C}_l$, $G = \begin{pmatrix} G \\ T_s \end{pmatrix}$.
5) Find the check matrix of $\mathcal{C}_l$ and update the set E $E = \{e_x | HT_s^T = 0, T_s \in \mathcal{C}_l\}$.
6) Find the maximal set of linearly independent vectors in E.
7) Set $j = dim(E)$.
8) Construct Λ by placing the j linearly independent vectors as the bottom rows of Λ.
end

Construct the matrices for the $\mathcal{C}(k, r)$ code (r is the rank of G):
1) Calculate π from $\pi^{-1} = \begin{pmatrix} \bar{H} \\ H \end{pmatrix}$ where $\bar{H}$ is any complement matrix (in terms of rank) of H .
2) Extract P from the r right most columns of π, $\pi = (P, \bar{P})$.
3) If k is odd set $J = 0$, otherwise, choose the position w of the single 'one' in i'th row of J from the support of $\bar{H}T_{s_i}^T$ where T_{s_i} is the trace vector associated with the e_x in the i'th row of Λ.

Eq. 7 is fulfilled. Hence, the value of $Q(e)$ is

$$Q(e) = 2(\frac{2^{k-r}}{2}\frac{1}{2^k}) = 2^{-r}.$$

To conclude this proof, the error masking probability of nonzero error takes one of three possible values: $0, 2^{-r}$ and 2^{-r+1}. ■

APPENDIX B

Proof of Theorem 2

For odd values of k, $Tr(1) = 1$ and, from Prop. 4, e_x^{-3} takes all the possible values in $\mathbb{F}_{2^k}$. Denote by $D = (d_{k-1}, \ldots, d_0)$ a binary vector of length k. From Prop. 3 there is a one-to-one mapping between e_x^{-3} (and hence between e_x, $e_x \neq 0$) and $T_s \in \mathbb{F}_2^k \setminus 0$.

Let $c = (x, w) \in C$ be the transmitted codeword and let $e = (e_x, e_w)$ be the error vector. It follows from Theorem 1 that for odd values of k, the autocorrelation equals 2^{k-r+1} only when $e_x \neq 0$, $e_w \neq 0$, and $T_s = (t_w \neq 0, 0)$. From Prop. 3, there are $2^r - 1$ distinct e_x's in $\mathbb{F}_{2^k}$ for which $t_b = 0$.

Since $t_w \neq 0$ there exists at least one i, $0 \leq i < r$ for which $Tr(\alpha^i e_x^{-3}) = 1$. Hence, one half of the 2^r possible e_w's result in $Tr(a) = 1$ and the other half result in $Tr(a) = 0$. Consequently, there are

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

error vectors of the form (e_x, e_w), $e_x, e_w \neq 0$ which are always detected and $(2^r - 1)2^{r-1}$ vectors are masked with probability $Q(e) = 2^{-r+1}$.

The sum over the autocorrelation values is constant, $\sum_{e\in\mathbb{F}_2^{k+r}} R(e) = 2^{2k}$, and $R(0) = 2^k$. Since exactly $(2^r - 1)$ 2^{r-1} error vectors are associated with $Q(e) = 2^{-r+1}$, any other nonzero error vectors having an autocorrelation value that is not zero must have $Q(e) = 2^{-r}$. That is,

$$(2^{2k} - 2^k - (2^r - 1)2^{r-1}2^{k-r+1})/(2^{k-r}) = 2^r(2^k + 2^r)$$

error vectors have $Q(e) = 2^{-r}$.

To summarize, for odd values of k, $2^r(2^k - 2^r)$ error vectors are detected with probability $1 - 2^{-r}$, $(2^r - 1)(2^{r-1})$ error vectors are detected with probability $1 - 2^{-r+1}$. The remaining $(2^r - 1)(2^{r-1} + 1)$ nonzero errors are always detected. ■

APPENDIX C

A pseudo code for construction of a generalized optimal punctured cubic robust code for even values of k is given in Table IV. The algorithm follows directly from the proof of Th. 3. The algorithm generates a set of r linearly independent vectors $G \subseteq M$ (where $M = \mathbb{F}_2^k \setminus \mathcal{T}$), that spans a linear code $\mathcal{C}_l$ whose parity check matrix is H.

APPENDIX D

A pseudo code for construction of a generalized punctured cubic robust code with $d \geq 2$ is given in Table V. The algorithm follows directly from the proof of Th. 4.

REFERENCES

[1] N. Admaty, S. Litsyn, and O. Keren, "Punctuating, expurgating and expanding the q-ary BCH based robust codes," in *Proc. 27th IEEE Conv. Electr. Electron. Eng. Israel*, Nov. 2012, pp. 1–5.

[2] H. Bar-El, H. Choukri, D. Naccache, M. Tunstall, and C. Whelan, "The sorcerers apprentice guide to fault attacks," *Proc. IEEE*, vol. 94, no. 2, pp. 370–382, Feb. 2006.

[3] A. Barenghi, L. Breveglieri, I. Koren, and D. Naccache "Fault injection attacks on cryptographic devices: Theory, practice, and countermeasures," *Proc. IEEE*, vol. 100, no. 11, pp. 3056–3076, Nov. 2012.

[4] R. Berlekamp, H. Rumsey, and G. Solomon, "On the solution of algebraic equations over finite fields," *Inf. Control*, vol. 10, no. 6, pp. 553–564, Jun. 1967.

[5] M. Deza and F. Hoffman, "Some results related to generalized Varshamov-Gilbert bounds (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 23, no. 4, pp. 517–518, Jul. 1977.

[6] S. Engelberg and O. Keren, "A comment on the Karpovsky–Taubin code," *IEEE Trans. Inf. Theory*, vol. 57, no. 12, pp. 8007–8010, Dec. 2011.

[7] G. Gaubatz, B. Sunar, and M. G. Karpovsky, "Non-linear residue codes for robust public-key arithmetic," in *Proc. Workshop FDTC*, 2006, pp. 173–184.
[8] M. G. Karpovsky, *Finite Orthogonal Series in the Design of Digital Devices*. New York, NY, USA: Wiley, 1976.
[9] M. G. Karpovsky, K. Kulikowski, and Z. Wang, "Robust error detection in communication and computation channels," presented in the Int. Workshop Spectral Tech., 2007.
[10] M. G. Karpovsky and A. Taubin, "A new class of nonlinear systematic error detecting codes," *IEEE Trans. Inf. Theory*, vol. 50, no. 8, pp. 1818–1820, Aug. 2004.
[11] M. G. Karpovsky and P. Nagvajara, "Optimal codes for the minimax criterion on error detection," *IEEE Trans. Inf. Theory*, vol. 35, no. 6, pp. 1299–1305, Nov. 1989.
[12] Y. Neumeier and O. Keren, "Punctured Karpovsky-Taubin binary robust error detecting codes for cryptographic devices," in *Proc. 18th IEEE IOLTS*, Jun. 2012, pp. 156–161.
[13] K. T. Phelps, "A combinatorial construction of perfect codes," *SIAM J. Algebraic Discrete Methods*, vol. 4, no. 3, pp. 398–403, 1983.
[14] S. P. Skorobogatov, "Semi-invasive attacks–a new approach to hardware security analysis," Comput. Lab., Univ. Cambridge, Cambridge, U.K., Tech. Rep. 630, 2005.
[15] Ju. L. Vasilev "On nongroup close-packed codes," *Trans. Problemy Kybernit*, vol. 8, pp. 375–378, 1965.
[16] I. M.R. Verbauwhede, *Secure Integrated Circuits and Systems*. New York, NY, USA: Springer-Verlag, 2010.
[17] Z. Wang, M. G. Karpovsky, and A. Joshi, "Reliable MLC NAND flash memories based on non-linear t-error correcting codes," in *Proc. Int. Conf. Depend. Syst. Netw.*, Jul. 2010, pp. 41–50.
[18] Z. Wang, M. G. Karpovsky, and K. Kulikowski, "Design of memories with concurrent error detection and correction by non-linear SEC-DED codes," *J. Electron. Test.*, vol. 26, no. 5, pp. 559–580, 2010.

Yaara Neumeier received the B.Sc and M.Sc. degrees in Computer Engineering from the Bar-Ilan University, Israel in 2010 and 2013 respectively. Between 2009 and 2011 she held a programmer position at Orckit-Corrigent. Since 2013, she is a Ph.D. student at the Faculty of Engineering at Bar-Ilan University, Israel. Her research topic is codes for reliable and trustworthy hardware systems.

Osnat Keren received the M.Sc. degree in Electrical Engineering from the Technion-Israeli Institute of Technology and the Ph.D. degree from the Tel-Aviv University, Israel in 1988 and 1999 respectively. Between 1988 and 1994 she held a chip design and senior DSP engineer position at National Semiconductor, and between 1999 and 2003 she was the Senior Scientist of Millimetrix Broadband Networks. Since 2004, Dr. Keren has been with the Faculty of Engineering at Bar-Ilan University, Israel.