Robustness of Security-Oriented Binary Codes Under Non-Uniform Distribution of Codewords

Osnat Keren and Igor Shumsky
Faculty of Engineering
Bar-Ilan University
Ramat-Gan, Israel 52900
Email: osnat.keren@biu.ac.il, ig.shum@gmail.com

Mark Karpovsky
Department of Electrical and Computer Engineering
Boston University
Boston, Massachusetts 02215
Email: markkar@bu.edu

***Abstract*—Robust and partially robust codes are used in cryptographic devices for detecting active side channel attacks on the hardware. The codes are usually designed for uniformly distributed codewords. In practice, however, there are codewords that are much more likely to appear than others. This paper addresses the question of how good are existing robust codes in this context. The worst case scenario is analyzed and a method that allows the designer to avoid this scenario with a relatively low cost is presented.**

***Index Terms*—Robust codes; security; undetected error probability; puncturing; fault analysis attacks; non-uniform distribution;**

I. Introduction

The security of cryptographic devices is threatened by fault injection attacks on the hardware. By injecting faults an adversary can obtain secret or private information that is stored in the device. Modern fault injection techniques allow an adversary to introduce faults at any physical point of the circuitry. A fault can flip bits, stuck a gate at a certain value, or change data on wires [2], [8], [10]. In turn, an attack can be mathematically modeled as an additive (i.e., symmetric) error that distorts the correct output of that circuit. Unlike random errors, i.e., errors caused by nature, an error induced by an adversary can be of any multiplicity.

Fault injection attacks can be detected with relatively high probability by security-oriented codes. It is convenient to classify fault injection attacks by their strength; In weak attacks the adversary *cannot* control which codeword will appear at the output of the circuitry, while in *strong attacks*, he can determine the outputs by choosing the inputs.A schematic architecture, which provides robustness against weak attacks is shown in Fig. 1; Its equivalent mathematical model is shown in Fig. 2.

Codes for detecting weak attacks, e.g., [1], [3]–[6], [11], are usually designed under the assumption that the codewords are equally likely to occur. However, when the source of the information is a computation channel, i.e., a combinatorial logic or a sequential machine, this assumption is almost always violated. Indeed, the distribution of vectors applied at run-time to the inputs of the combinational portion of a sequential machine is highly skewed due to the fact that some state transitions are more common than others and that some

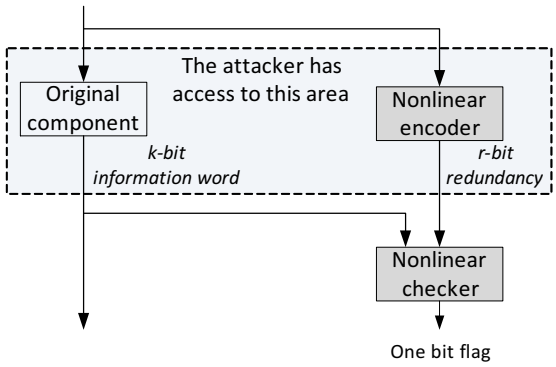

Fig. 1. A schematic architecture of a circuit component protected by a systematic security-oriented code. The shaded area is accessible to the attacker.

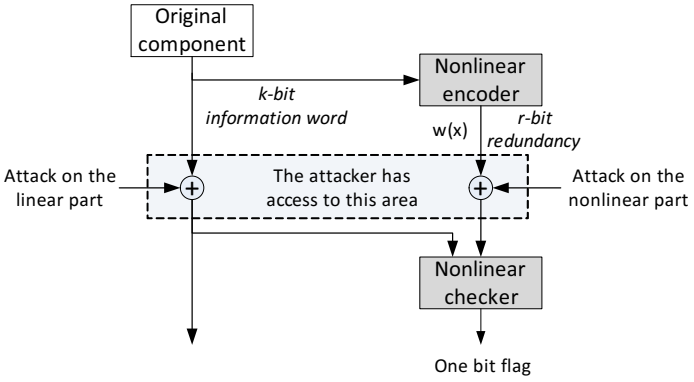

Fig. 2. Mathematical model of a circuit component protected by a systematic security-oriented code.

input combinations are invalid and hence cannot occur. The distribution of the outputs of arithmetic modules is also highly non-uniform. For example, it is more likely to have a '0' at the output of a multiplier than other values. In arithmetic modules and in sequential state machines, the probability of having a certain output can be easily computed. A judicious attacker can use this information to choose an error that is hardly (if ever) detected.

This paper addresses two questions: a) how good are the known robust codes, and in particular the Quadratic-Sum

codes and codes derived from the cubic code, against an adversary that knows the distribution of the codewords, and b) is it possible to reduce the error masking probability of the code without adding more redundancy?

The remaining of the paper is organized as follows. Section II briefly describes security oriented codes and presents the Punctured-Cubic and the Quadratic-Sum codes. Section III analyzes the worst case scenario. Section IV introduces methods to avoid this scenario by mapping the set of most probable words to a predefined set. An upper bound on the error masking probability when using this mapping is also presented. Section V concludes the paper.

II. Preliminaries - Security oriented codes

A binary code $\mathcal{C}(n,k)$ is a subset of size 2^k of an n-dimensional binary vector space $\mathbb{F}_2^n$, ($\mathbb{F}_2 = GF(2)$). In conventional coding theory, codes are designed to provide reliability against *random errors*, i.e., errors of low multiplicity. The codes are therefore characterized by their rate (i.e., k/n), the minimal distance between the codewords, and the undetected (random) error probability. All these parameters are determined by the chosen code; They are indifferent to the encoding scheme.

In cases where the reliability of the system is the main concern, a *systematic code*, that is, a code in which the information word is embedded in the codeword in its original form, has an advantage over non-systematic codes since it simplifies the decoding procedure and usually has a lower implementation cost. However, in security oriented coding, the most important property of a code is its robustness, i.e its ability to provide immunity against weak attacks. As we show next, when some codewords are more probable to appear than others, the encoding (i.e., the mapping between an information word $m \in \mathbb{F}_2^k$ to a codeword $c \in \mathbb{F}_2^n$) plays a crucial role in determining the robustness of a code.

A. Definition of robustness

Let $\mathcal{C}$ be a code and denote by $p(c)$ is the probability that the codeword $c \in \mathcal{C}$ will be used. The robustness of $\mathcal{C}$ is measured in terms of its undetected error probability, which is also referred to as the *error masking probability*. The error masking probability is the probability, $Q(e)$, that a given error $e \in \mathbb{F}_2^n$ will map a codeword onto another codeword, i.e.,

$$Q(e) \equiv \sum_{c \in \mathcal{C}} p(c)\delta(c \oplus e)$$

where $\delta(z)$ is the characteristic function of the code, $\delta(z) = 1$ if $z \in \mathcal{C}$ and it equals 0 otherwise.

When the adversary induces an error e one of the following three scenarios may happen:

1) The error will always be detected ($Q(e) = 0$). The set of errors of this type is denoted by E_a.
2) The error will never be detected ($Q(e) = 1$). Errors that are never detected form a group. The group, denoted by K_d, is called the *Kernel* of the code.

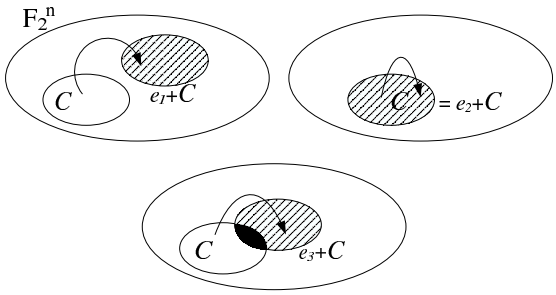

Fig. 3. The error $e_1 \in E_a$ is always detected since $\mathcal{C} \cap \{e_1 \oplus \mathcal{C}\} = \varnothing$. The error $e_2 \in K_d$ is never detected since $\mathcal{C} = \{e_2 \oplus \mathcal{C}\}$, and e_3 is detected with probability $Q(e_3) = |\mathcal{C} \cap \{e_3 \oplus \mathcal{C}\}|/|\mathcal{C}|$.

3) The error will be detected with probability $0 < 1 - Q(e) < 1$. That is, there exists at least one codeword that detects the error, and there exists at least one codewords that masks it.

The three scenarios are illustrated in Fig. 3.

Definition 1 (Robust and partially robust codes): Robust codes are codes for which the dimension of K_d equals zero, that is, no attack is masked. *Partially robust* codes are codes for which the dimension of K_d is greater than zero but less than k.

B. The error masking equation

Let $\mathcal{C}(n,k)$ be a binary systematic code of length $n = k + r$ and size 2^k. A codeword $c \in \mathcal{C}(n,k)$ has two parts: an information part denoted by x and a redundancy part w, which is a function of x. Each part can be referred to as an element of a finite field or as a vector over a finite field. For example, the information part x can be considered as a binary vector in k-dimensional space $\mathbb{F}_2^k$; It can be also referred to as an element of the finite field $\mathbb{F}_{2^k} = GF(2^k)$. For example, the expression Px^3 where P is a $r \times k$ matrix, has to 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, the outcome of this operation is an element in $\mathbb{F}_{2^r}$.

Let $c = (x, w) \in \mathcal{C}$ be a codeword, where $w = w(x)$. Let $e = (e_x, e_w)$ be a nonzero error vector, $e_x \in \mathbb{F}_{2^k}, e_w \in \mathbb{F}_{2^r}$. An error is undetected (masked) by the codeword c if $c \oplus e \in \mathcal{C}$. Equivalently, e is masked by c if

$$w(x \oplus e_x) = w(x) \oplus e_w. \tag{1}$$

Equation (1) is called the *error masking equation* for systematic codes. The number of solutions (x's) to (1) and the probability of each determine $Q(e)$. Namely, let $X(e)$ be the set of x's that satisfy this equation,

$$X(e) = \{x | c(x) \oplus e \in \mathcal{C}\}.$$

Then,

$$Q(e) = \sum_{x \in X(e)} p(x),$$

where $p(x)$ is the probability of the codeword $c = (x, w)$, i.e., $p(x) = p(c)$.

The error masking probabilities of $\mathcal{C}$ and error masking probabilities of a coset of $\mathcal{C}$ are identical. Therefore, without loss of generality, we assume that $\mathbf{0} = (0,0) \in \mathcal{C}$. Consequently,

Property 1: If $\mathbf{0} \in X(e)$, then $e \in \mathcal{C}$.

The error masking probability for uniformly distributed codewords is lower bounded by [6],

$$Q(e) \geq \max(2/2^k, 2^k/2^n).$$

Codes that achieve this bound are called *optimum codes*.

C. The Punctured-Cubic code and the Quadratic-Sum code

In this paper, we analyze two robust codes, the Punctured-Cubic (PC) code derived from the cubic (x, x^3) code by deleting some redundancy bits, and the Quadratic-Sum (QS) code. Both codes are robust *systematic* codes of rate higher than one-half [1], [4], [7]. Moreover, both codes are optimum or close to optimum.

Construction 1 (Punctured-Cubic code [1]):
Let P be a binary $r \times k$ matrix of rank $r \leq k$. The code

$$\mathcal{C} = \{(x, w) : x \in \mathbb{F}_{2^k}, w = Px^3 \in \mathbb{F}_{2^r}\}$$

is called a Punctured Cubic $\mathcal{C}(k+r, k)$ code.

The error masking equation of the PC code is

$$P(x \oplus e_x)^3 = Px^3 \oplus e_w.$$

Construction 2 (Quadratic-Sum code [4]):
Let $k = 2sr$ and $x = (x_1, x_2, \cdots x_{2s})$, where $x_i \in \mathbb{F}_{2^r}$ for $1 \leq i \leq 2s$. The code

$$\mathcal{C} = \{(x, w) : x \in \mathbb{F}_{2^k}, w = x_1x_2 \oplus \cdots \oplus x_{2s-1}x_{2s} \in \mathbb{F}_{2^r}\}$$

is called a Quadratic-Sum $\mathcal{C}(k+r, k)$ code.

The error masking equation for the QS code is

$$\sum_{i=1}^{s}(x_{2i-1} \oplus e_{x,2i-1})(x_{2i} \oplus e_{x,2i}) = \sum_{i=1}^{s} x_{2i-1}x_{2i} \oplus e_w.$$

D. The robustness of the PC and QS codes under uniform distribution

If the codewords are uniformly distributed, then each codeword may appear on the output with probability of $1/|\mathcal{C}|$. The worst case error masking probability under uniform distribution of the codewords is denoted by Q_{mc}. The subscript mc stands for maximal correlation, since in this case

$$Q(e) = \frac{R(e)}{R(0)},$$

and,

$$Q_{mc} = \frac{\max_{e \neq 0} R(e)}{R(0)},$$

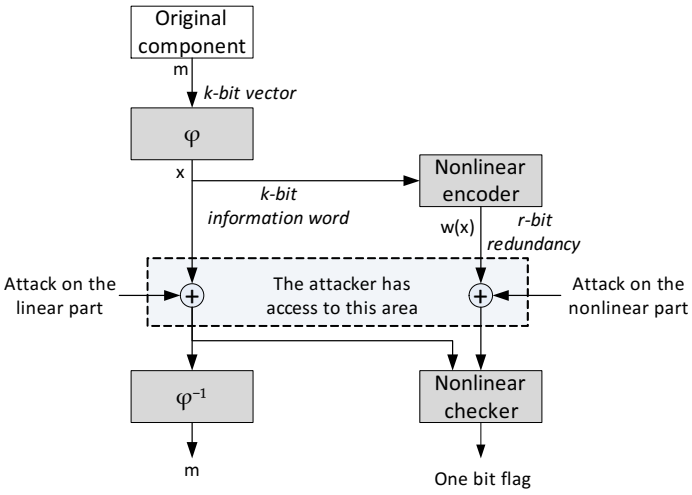

Fig. 4. A mathematical model of a computation channel protected by a one-to-one mapping φ followed by a systematic error detecting code $\mathcal{C}$.

where R is the autocorrelation function of δ, that is,

$$R(e) = \sum_{z \in \mathbb{F}_2^n} \delta(z)\delta(z \oplus e).$$

The error masking probabilities of the PC and QS codes are the following:

Theorem 1 ([7]): Let C be a PC code defined by a binary $r \times k$ matrix P of rank $r > 1$, Then the kernel of the code is of dimension 0. For odd values of k, $Q_{mc} = 2^{-r+1}$. For even values of k, there exist P matrices for which $Q_{mc} = 2^{-r}$.

Theorem 2 ([4]): Let C be a QS code. Then the kernel of the code is of dimension 0. For $k = 2sr$, the error masking probability is $Q_{mc} = 2^{-r}$.

III. The worst case scenario

Consider a computation channel that produces each cycle an output vector $m \in \mathbb{F}_2^k$. Let φ be a one to one mapping between m and an information word x, i.e., $x = \varphi(m)$. To provide immunity, each cycle a codeword $c = (x, w(x))$ is generated from the information word x (as shown in Fig. 4). The probability that a codeword $c(x) = c(\varphi(m))$ is used equals to the probability that the output m is produced, that is,

$$p(c) = p(x) = p(m).$$

Since for a given code, $X(e)$ is fixed, and

$$Q(e) = \sum_{x \in X(e)} p(x) = \sum_{m,\ \varphi(m) \in X(e)} p(m),$$

the error masking probability under non-uniform distribution of the outputs depends solely on φ.

The following lemma provides a lower bound on the error masking probability when the worst φ is used. In the next section we show that if one uses a φ that maps the most

probable vectors m to a *predefined* set S, s/he can reduce the error masking probabilities.

Without loss of generality assume that

$$1 \geq p(m_1) \geq p(m_2) \geq \cdots \geq p(m_{2^k}) \geq 0$$

and

$$\sum_{i=1}^{2^k} p(m_i) = 1.$$

Consider the mapping $x_i = m_i$. For this mapping we have,

$$1 \geq p(x_1) \geq p(x_2) \geq \cdots \geq p(x_{2^k}) \geq 0.$$

Denote by $P(S)$ the accumulated probability $\sum_{x_i \in S} p(x_i)$ and assume that there is a set $S \subseteq \mathbb{F}_2^k$ for which $P(C \setminus S)$ is *negligible*. In the worst case scenario there exists an error e such that either $S \subseteq X(e)$ or $X(e) \subset S$. Namely,

Lemma 1: The worst case error masking probability, Q_{wc}, is lower bounded by

$$Q_{wc} \geq \begin{cases} P(S) & |S| \leq Q_{mc}2^k \\ \frac{Q_{mc}2^k}{|S|}P(S) & otherwise \end{cases}$$

Example 1: Let $k = 3$ and $r = 1$. The eight codewords of the corresponding PC code (represented by their integer values) are

$$(0,0),(1,0),(2,0),(3,1),(4,1),(5,1),(6,1),(7,0).$$

Table I shows the $X(e)$ of each error vector.

TABLE I
THE ERROR VECTORS AND THEIR MASKING CODEWORDS

e	$\lvert X(e)\rvert$	$X(e)$
(0,0)	8	all x's
(0,1)	0	-
(1,0)	4	0,1,4,5
(1,1)	4	2,3,6,7
(2,0)	4	0,2,4,6
(2,1)	4	1,3,5,7
(3,0)	4	1,2,5,6
(3,1)	4	0,3,4,7
(4,0)	0	-
(4,1)	8	all x's
(5,0)	4	2,3,6,7
(5,1)	4	0,1,4,5
(6,0)	4	1,3,5,7
(6,1)	4	0,2,4,6
(7,0)	4	0,3,4,7
(7,1)	4	1,2,5,6

The rows of the table are written in pairs. In each pair, one error vector is a codeword and the second is a non-codeword. By Prop. 1, an error vector whose $X(e)$ contains the all-zero word, is a codeword. It is clear from the table that the code is *partially robust* since the non-zero error $(4,1)$ is masked by all codewords. However, all the remaining error vectors are either always detected or they are masked by half of the codewords. Therefore, for uniformly distributed codewords, $Q_{mc}(e) = 0.5$. Although this paper deals with robust codes, to simplify the presentation, we assume that the adversary cannot induce the error $(4,1)$. This assumption allows us to use the $\mathcal{C}(4,3)$ partially robust PC code.

Assume now that the m's are not uniformly distributed,

$$p(m) = \begin{cases} (1-\epsilon)/5 & m \in \{2,3,4,6,7\} \\ \epsilon/3 & otherwise \end{cases}$$

If no mapping is used (i.e., $x_i = m_i$), then a judicious attacker would apply the error $(5,0)$ whose corresponding error masking probability is the maximal, $Q((5,0)) = \frac{4}{5}(1-\epsilon)$. However, if φ is a Gray mapping,[1] the highly probable m's are mapped to the set $S = \{2,3,4,5,6\}$, and the worst case error masking probability becomes $\frac{3}{5}(1-\epsilon)$. As we show next, no better mapping can be found.

IV. Constructive upper bounds on the error masking probability

For uniformly distributed codewords, the error masking probability of the PC and the QS codes is upper bounded by Q_{mc}. Therefore, any error vector is masked by at most $2^k Q_{mc}$ codewords. Consequently, if the size of S, is greater than $2^k Q_{mc}$, then any error will be detected with probability of at least

$$1 - \frac{2^k Q_{mc}}{|S|}P(S) > 0.$$

Obviously, if the size of S is smaller than that, the probability that the error will be masked increases. In what follows we discuss the case where

$$|S| \leq \min_{e \neq 0} |X(e)|,$$

and present mappings for which any nonzero error will never be masked.

A. Sufficient conditions for $Q < 1$

In cases where $|S| = 2$, no mapping can help; An adversary who knows the two most probable outputs, say m_1 and m_2, and the mapping φ may choose an error

$$e = c(\varphi(m_1)) \oplus c(\varphi(m_2)),$$

for which $Q(e) \geq P(S) = 1 - \epsilon$.

The following theorem suggests a lower bound on the size of S for which there exists a mapping that can reduce $Q(e)$.

Theorem 3: Let $\mathcal{C}$ be a PC or a QS code. Then, there exists at least one set S of size s,

$$\frac{k+1}{-log_2(Q_{mc})} + 1 \leq s \leq \min(2^k Q_{mc}, 2^{k-2}),$$

such that $S \setminus X(e) \neq \phi$ for all non-zero e.

[1] A Gray code maps $m = (m_{k-1}, \ldots m_0)$ to $x = (x_{k-1}, \ldots x_0)$ as follows: $x_i = m_{i+1} \oplus m_i$ for $i = 0, \ldots k-1$ where $m_k = 0$. For example $m = (010)$ is encoded to (011).

Example 2: Let $k = 16$ and $r = 4$. Assume that twenty vectors (out of the 2^{16}) may appear with probability $1-\epsilon$ at the output of the device to be protected. Since there exists an error for which $\min(|X(e)|) = 2^{12}$, and $20 << 2^{12}$, in the worst case scenario the error will not be noticed. For a PC code we have,

$$\frac{16+1}{4-1} + 1 \leq |S| = 20 \leq min(2^{13}, 2^{16-2}).$$

therefore, by Theorem 3, there exist a subset S of twenty vectors such that any error is detected with probability of at least $\frac{1-\epsilon}{20}$.

Although Th. 3 states that it is possible to find a set that can detect any error, it does not provide an efficient way to do so. In the following sections we introduce two mappings, i.e., two sets, for which any non-zero error can be detected.

B. Generalized Hamming ball mapping

We define a generalized Hamming ball as follows:

Definition 2: Let $V = \{v_i\}_{i=1}^{u} \subset \mathbb{F}_2^k$ be an arbitrary set of u, $u \leq k$, linearly independent vectors. A generalized Hamming ball $B_{(u,w)} \subseteq \mathbb{F}_2^k$ is a set (or a coset of a set) that consists of the vectors

$$\left\{ \sum_{i=1}^{u} a_i v_i \mid a = (a_u, \dots a_1) \in \mathbb{F}_2^u,\ wt_H(a) \leq w \right\}$$

where $wt_H(a)$ stands for the Hamming weight of a.

Theorem 4: Let $\mathcal{C}$ be a PC or a QS code. Let $S \subseteq B_{(u,w)}$ where $u \geq k + log_2(Q_{mc}) + 1$ and w is the smallest integer such that $\sum_{j=0}^{w} \binom{u}{j} \geq |S|$. Then, the code $\mathcal{C}$ can detect all the nonzero errors with probability greater or equal to

$$\frac{|S| - \sum_{j=0}^{w} \binom{k+log_2(Q_{mc})}{j}}{|S|}.$$

The proof of Theorem 4 follows directly from the fact that the PC code and the QS code have the following property:

Theorem 5: Let $\mathcal{C}$ be a PC or a QS code. Then, $X(e)$ is a subspace iff e belongs to $\mathcal{C}$ and a coset otherwise.

Corollary 1: The minimal size of a set that can detect any non-zero error e with $Q(e) > 0$ is greater than two and less or equal to $k + log_2(Q_{mc}) + 2$.

Example 3: Let $k = 16$ and $r = 4$. Assume that 650 output vectors (out of the 2^{16} possible combinations) occur with probability of $1-\epsilon$. Since for a PC code,

$$|X(e)| \geq 2^{k-r} = 2^{12} > 650,$$

in the worst case scenario, there may be an error that will be masked with probability greater than $1-\epsilon$. However, for $w = 3$ and $u = 16$ we have

$$|B_{(16,3)}| = \sum_{j=0}^{3} \binom{16}{j} = 697.$$

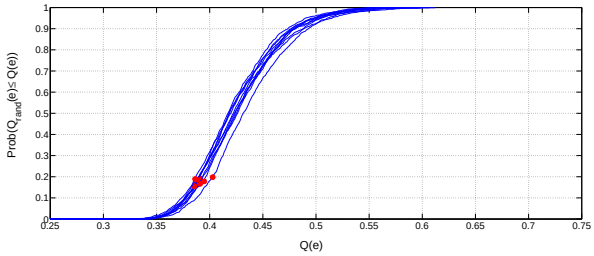

Fig. 5. The probability that a random mapping for a QS code with $k = 6$ and $r = 3$ will provide a maximal error masking probability smaller than $Q(e)$ for ten probability distributions having $|S| = 7$. The red dots denote $Q(e)$ achieved by Const. 3

Therefore, by mapping these 650 m's to a set $S \subset B_{(16,3)}$ that consists of binary vectors of Hamming weight less or equal to three, one can reduce the error masking probability to

$$Q(e) \leq \frac{|S \cap X(e)|}{|S|} \leq \frac{\sum_{j=0}^{3} \binom{k-r+1}{j}}{|S|} = 0.58.$$

Note that if $|S| > k$, then the size of $S \cap X(e)$ decreases as the number of linearly independent vectors u increases. Moreover, as $|S|$ increases, the required w increases. The following construction, presented in [9], is not optimal, however, since it uses binary vectors of weight one, it is simple to implement.

Construction 3 ([9]): Let $p(m_1) \geq p(m_2) \geq \cdots \geq p(m_{2^k})$ Assign to each m_i a binary vector x_i such that the Hamming weight of x_i is smaller or equal to the Hamming weight of x_j for all $i < j$.

Note that if, for example, $k = 6, r = 3$ and $|S| = 7$ with the probability distribution

$$p(m) = \begin{cases} \frac{1-\epsilon}{|S|} & 0 \leq m \leq 6 \\ \frac{\epsilon}{2^k - |S|} & m > 6 \end{cases},$$

then there exist other mappings, which achieve smaller $Q(e)$'s than [9]:

$$\begin{aligned} S = \{0,1,2,4,8,16,32\}\ [9] \quad &\rightarrow \quad Q(e) \leq 0.5714, \\ S = \{0,10,21,27,50,55,62\} \quad &\rightarrow \quad Q(e) \leq 0.4286. \end{aligned}$$

Although the mapping in [9] is not optimal, is it much better than a random mapping. Fig. 5 shows, for ten different probability distributions having $|S| = 7$, the probability that a random mapping will provide a maximal error masking probability smaller than $Q(e)$. The red dots in the figure denote the error masking probability $Q(e)$ achieved by the suggested mapping. Refer only to the x-coordinate of the dots. The y-coordinate has no meaning, the star is placed on the graph just for convenience. On average this mapping has $Q(e) = 0.39$. The probability that a random mapping will provide error masking probability smaller than that is 0.18.

C. Robust-code based mapping

The following theorem states that if the elements of S are the codewords of a robust code, then a nonzero error is never masked.

Theorem 6: Let $\mathcal{C}$ be a PC or a QS code of dimension k, r redundancy bits, and error masking probability Q_{mc}. Let S be a robust code of length $\hat{n} = k$, dimension $\hat{k} = u$ and error masking probability $\hat{Q}_{mc}$. Then, the error masking probability of $\mathcal{C}$ is

$$Q(e) \leq \sqrt{2\hat{Q}_{mc}Q_{mc}2^{k-u}}.$$

Corollary 2: Let $\mathcal{C}$ be a PC or a QS code of dimension k, r redundancy bits, and error masking probability Q_{mc}. Let S be a subset of a robust code of length k, dimension $u = \lceil log_2(|S|) \rceil$ and error masking probability $\hat{Q}_{mc}$. Then,

$$Q(e) \leq \frac{\sqrt{Q_{mc}2^k(\hat{Q}_{mc}2^u + 1)}}{|S|}.$$

Corollary 3: Let $\mathcal{C}$ be a PC or a QS code of dimension k, r redundancy bits. Let S be a QS code of dimension u and $k-u$ redundancy bits. Then we have,

$$Q(e) \leq \sqrt{2 \cdot 2^{-(k-u)} \cdot 2^{-r+1} \cdot 2^{k-u}} \leq 2^{\frac{-r+2}{2}}.$$

Example 4: As before, let $k = 16$, $r = 4$ and assume that 400 output vectors may appear with probability $1 - \epsilon$. Here again, in the worst case scenario we have $Q(e) \geq 1-\epsilon$. Define S to be a subset of a $\mathcal{C}(\hat{n} = k = 16, \hat{k} = u = 9, \hat{r} = k-u = 5)$ PC code with $\hat{Q}_{mc} = 2^{-5+1}$. Then,

$$Q(e) \leq \sqrt{2 \cdot 2^{-4} \cdot 2^{-3} \cdot 2^5} = 0.707.$$

Note that in this case, the construction suggested in Th. 4 provides $Q(e) \leq \frac{378}{400} = 0.945$.

The following example shows the relation between the three upper bounds on the error masking probability when a mapping is applied.

Example 5 (Concluding example): Consider a PC code of dimension $k = 16$ and $r = 4$ redundancy bits. Assume that the $|S|$ most probable words are mapped to a set S, and that

$$p(x) = \begin{cases} \frac{1-\epsilon}{|S|} & x \in S \\ \frac{\epsilon}{2^k - |S|} & x \notin S \end{cases}.$$

The efficiency of the mapping, i.e., the error masking probabilities that can be achieved by using the suggested mappings, is shown in Fig. 6.

The X-axis is the size of S and the Y-axis is $\max_{e \neq 0}(Q(e))$. The black line represents a lower bound on worst case scenario (Lemma 1). The other lines represent upper bounds on $Q(e)$. The red line is the bound presented in Theorem 4, the blue line is the bound presented in Theorem 3, and the green line is the bound in Corollary 2.

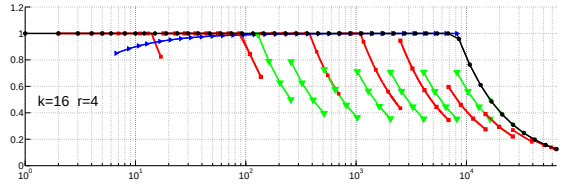

Fig. 6. Error masking probability of punctured cubic code with $k = 16$ and $r = 4$ as a function of $|S|$.

V. Conclusions

The Punctured-Cubic code and the Quadratic-Sum code are systematic robust codes designed for uniformly distributed codewords. The codes can detect any error with non-zero probability regardless its multiplicity. In cases where the codewords are not equally likely to appear, the performance of the codes degrades significantly and the robustness may vanish. The paper addresses this problem. It is shown that by mapping the most probable data patterns to a predefined set before the encoding, it is possible to significantly reduce the error masking probability and maintain the robustness of the codes.

Acknowledgment

The work of the first two authors was supported by the Israel Science Foundation (ISF) grant No. 1200/12. The work of the third author was supported by the NSF Grant CNS 1012910

References

[1] N. Admaty, S. Litsyn, and O. Keren, "Punctuating, Expurgating and Expanding the q-ary BCH Based Robust Codes", The 27-th IEEE Convention of Electrical and Electronics Engineers in Israel, 2012, pp.1–5.

[2] H. Bar-El, H. Choukri, D. Naccache, M. Tunstall, and C. Whelan, "The sorcerers apprentice guide to fault attacks", Vol.94, No.2, 2006, pp.370–382.

[3] S. Engelberg and O. Keren, "A Comment on the Karpovsky-Taubin Code," IEEE Trans. Info. Theory, Vol. 57, No. 12, 2011, pp. 8007–8010.

[4] M.G.Karpovsky, K. Kulikowski, and Z. Wang, "Robust Error Detection in Communication and Computation Channels" Keynote paper in the Int. Workshop on Spectral Techniques, 2007.

[5] M. G. Karpovsky and P. Nagvajara, "Optimal Codes for the Minimax Criterion on Error Detection," IEEE Trans. on Information Theory, Vol. 35, No. 6, November 1989, pp. 1299–1305.

[6] M.G.Karpovsky and A. Taubin, "A New Class of Nonlinear Systematic Error Detecting Codes," IEEE Trans. Info. Theory, Vol 50, No.8, 2004, pp.1818–1820.

[7] Y. Neumeier and O. Keren, "Punctured Karpovsky-Taubin Binary Robust Error Detecting Codes for Cryptographic Devices," IEEE International On-Line Testing Symposium, March 2012, pp.156–161.

[8] S. P. Skorobogatov, "Semi-Invasive Attacks - a New Approach to Hardware Security Analysis" Technical Report, University of Cambridge. Number 630.

[9] I. Shumsky and O. Keren, "Security-Oriented State Assignment", TRUDEVICE, 1'st Workshop on trustworthy manufacturing and utilization of secure devices, 2013.

[10] I. M.R. Verbauwhede(Ed.), Secure Integrated Circuits and Systems, Springer, 2010.

[11] Z. Wang, M. G. Karpovsky, and K. Kulikowski, "Design of Memories with Concurrent Error Detection and Correction by Non-Linear SEC-DED Codes" Journal of Electronic Testing, Vol. 26, No. 5, Oct 2010, pp.559–580.