Expurgated Codes for Detecting Jamming in Multi-level Memories

Yaara Neumeier
Faculty of Engineering
Bar-Ilan University
Email: yaara.neumeier@biu.ac.il

Osnat Keren
Faculty of Engineering
Bar-Ilan University
Email: osnat.keren@biu.ac.il

***Abstract*—Robust q-ary codes can efficiently detect jamming in multilevel memories when q is a power of two. When q is not a power of two, a binary information word has to be converted and encoded into a q-ary codeword. This conversion expurgates the code; some of the q-ary codewords are never used. Unless properly designed, expurgation can significantly degrade the efficiency of the code in terms of its error detection capability. This work presents a q-ary robust Quadratic-Sum code for arbitrary q's and analyzes the error masking probability of the expurgated code when applied to multilevel memories. It is shown that by wisely designing the converter, this degradation can be minimized, and in some cases, the expurgated code's efficiency can be superior to the one of the original code. This work suggests how to construct a converter to optimize code properties.**

***Index Terms*—Robust codes; Multi-level Memories; Jamming attacks; Hardware security.**

I. Introduction

Memory arrays are prone to jamming attacks [1], where an adversary injects faults into the memory to alter a stored value. The injected fault manifests itself as an additive error of an arbitrary multiplicity; i.e., any number of bits may be flipped or distorted. [2]. Fault injection can be executed, for example, using variations on voltage, temperature, white light, laser, ion beams, etc. An attacker can inject faults into the memory to change its content and then acquire information about the system by analyzing its behavior [2].

Several countermeasures to jamming attacks on memories have been proposed [2][3]. For example, one approach to protect memories is to implement intrusion detection mechanisms based on active protection using tamper-proof box and sensors to make the device physically inaccessible. Since different sensors are used against different injection methods, this method becomes expensive and inappropriate for simple, small devices. Moreover, it is powerless against new types of attacks that were not considered by the designers. Furthermore, internal information about the design may help the attacker bypass this protection. An alternative approach is to detect the manifestation of the fault as an error using error detecting codes.

Classic coding theory addresses the problem of the *reliability* of information transmitted over a noisy channel or stored in storage media. In classic coding theory, the errors are assumed to be random with a relatively small probability. Consequently, a reliability oriented code should protect the system from a small number of random errors (small multiplicity). Many known codes designed for reliability (such as the parity bit code, Hamming code, BCH codes, etc) are linear [4], however; in linear codes, all the errors that are codewords are never detected. As a result, reliability oriented codes cannot be used to provide *security* against an attacker that can inject any error.

Jamming can be detected by nonlinear robust codes capable of detecting any non-zero error. The efficiency of these codes is measured in terms of their error masking probability $\overline{Q}_{\mathcal{M}} = \max_{\mathbf{e}\neq 0} Q(\mathbf{e})$ where $Q(\mathbf{e})$ is the probability that an error $\mathbf{e}$ is masked by codewords in $\mathcal{C}$. This probability depends on the probability mass function of the codewords; that is,

$$Q(\mathbf{e}) = \sum_{\mathbf{c},\mathbf{c}+\mathbf{e}\in\mathcal{C}} p(\mathbf{c}),$$

where $p(\mathbf{c})$ is the probability that $\mathbf{c} \in \mathcal{C}$ is used.

The Quadratic-Sum (QS) code [5] is a nonlinear q-ary high-rate robust code of length n and dimension k defined over a finite field, i.e., for a q that is a power of a prime. When all the codewords are equally likely to occur, the code is an optimum code, and its error masking probability equals $\overline{Q}_{\mathcal{C}} = q^{-(n-k)}$ [5]. If these conditions are not fulfilled, the performance of the code may significantly degrade [6].

The encoding complexity of a binary QS code is relatively low with respect to other robust codes (e.g., the codes in [7][8] which involve computations over finite fields of high order); its k information symbols are treated as $2s$ symbols from $\mathbb{F}_2^r$ and its single redundant symbol x_{2s+1} is the sum $\sum_{i=1}^{s} x_{2i-1}x_{2i}$ over $\mathbb{F}_2^r$. This simple structure makes the code an attractive countermeasure to jamming in binary and q-ary multilevel memories, where q is a power of two.

However, in some cases, the code's alphabet size is not a power of two. Note that the number of levels in a multilevel memory, l, may be a power of two. Nevertheless, the code's alphabet size q may be smaller than l. For example, in Write-Once-Memory codes and rank-modulation codes the alphabet size is smaller than the number of levels to enable several write cycles to the same address before block-erasure. As far as we know, all known robust codes ([5][7][8][9]) are defined over a finite field, i.e., where q is a power of a prime, and

 ISBN: 978-1-61208-492-3

cannot be used in the case where the number of states is not a power of a prime.

Another problem that arises when the code's alphabet size is not a power of two is that each binary information word has to be converted to a q-ary word by a dedicated conversion circuit [10]. A conversion circuit maps a binary vector of length k_2 to a q-ary vector of length k_q. A conversion circuit is constructed from sub-blocks, denoted DCC_i. The input of each sub-block is a binary w_2-bit vector, and its output is a q-ary vector of length w_q. The values w_2 and w_q are chosen such that $w_q = \lceil w_2 \log_q 2 \rceil$. A schematic illustration of a multilevel memory with a conversion circuit is shown in Figure 1. Since $q^{w_q} < 2^{w_2}$, some of the codewords of the (original) q-ary code $\mathcal{C}$ are never used. When these unused words are chosen arbitrarily, the error masking probability of the expurgated code, denoted by $\mathcal{M}$, can become higher than the error masking probability of the original code.

Robust codes over finite fields for a non-uniform distribution of codewords were discussed in [6][11][12]. In [6], the authors showed that when *most of the codewords* appear with low probability, which is the case for some Final State Machines (FSMs), it is possible to avoid the worst case scenario by pre-mapping the information word before the encoding. In [12], a general approach for mapping the most probable codewords to a predefined set was suggested. In [11] the authors dealt with the non-uniform characteristics of FSMs using randomized masking. Another way to cope with a non-uniform distribution of codewords is by embedding randomness [13][14]; these codes are also defined over finite fields. However, since the random symbols are an integral part of the codeword, their rate is lower than the rate of (deterministic-encoding) robust codes such as the QS and the Punctured-Cubic/Square in [7][8]. These solutions are appropriate for applications where a small portion of the states appear with high probability; they are less suitable for applications such as multilevel memories where some words never occur and other words appear with uniform probability.

This paper expands the QS construction to codes over integer rings, proves its robustness, and examines the security related implications of applying expurgated codes on q-ary memory systems with data conversion circuits in cases where q is not a power of a prime. It is shown that by choosing $\mathcal{M}$ properly, the practical error masking probability $\overline{Q}_{\mathcal{M}}$ may be even better than $\overline{Q}_{\mathcal{C}}$. The main ideas and results presented in this paper are the following:

- A QS-based code $\mathcal{C}$ is robust over rings.
- The maximal error masking probability $\overline{Q}_{\mathcal{M}}$ of code $\mathcal{M}$ is bounded by
$$\frac{(2|\mathcal{M}| - |\mathcal{C}|)}{p_1|\mathcal{M}|} \leq \overline{Q}_{\mathcal{M}} \leq \frac{|\mathcal{C}|}{p_1|\mathcal{M}|} < 2\overline{Q}_{\mathcal{C}}, \quad (1)$$
where p_1 is the smallest divisor of q. Since $p_1 \geq 2$, an expurgated code is robust; it can detect any nonzero error with a probability greater than zero.
- If $p_1 = 2$ there exists an expurgated code $\mathcal{M}$, which provides a smaller error masking probability, i.e., $\overline{Q}_{\mathcal{M}} < \overline{Q}_{\mathcal{C}}$. If $p_1 \neq 2$ and
$$(p_1 - 1)k_q + 1 \leq |\mathcal{C}| - |\mathcal{M}| \quad (2)$$
there exists an $\mathcal{M}$ with $\overline{Q}_{\mathcal{M}} < \frac{|\mathcal{C}|}{p_1|\mathcal{M}|}$.
- A code construction for $\mathcal{M}$ which minimizes $\overline{Q}_{\mathcal{M}}$ in cases where $p_1 = 2$ for a given set of parameters, and has
$$\frac{(2|\mathcal{M}| - |\mathcal{C}|)}{p_1|\mathcal{M}|} \leq \overline{Q}_{\mathcal{M}} \leq \overline{Q}_{\mathcal{C}},$$
is presented.

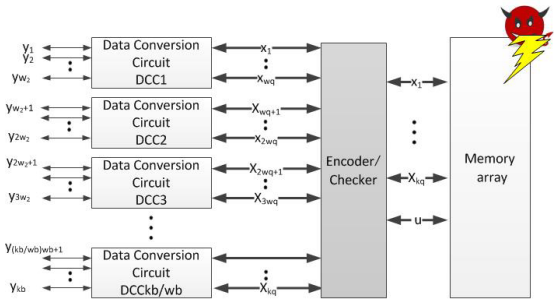

Figure 1. Multilevel memory system with data conversion circuit, protected by encoder and a checker.

The rest of this paper is organized as follows. Section II defines and analyzes the Quadratic-Sum code for a general q. Section III presents the expurgated code and the security problem that arises when applying the codes to q-ary memories where q is not a power of two. Then, lower and upper bounds on the error masking probability are presented. Section IV suggests how to choose $\mathcal{M}$ in cases where $p_1 = 2$ to minimize its error masking probability and Section V concludes the paper.

II. The Extended QS code

Notations: Regular lowercase letters are used to represent scalars. Boldface lowercase letters are used to denote row vectors, e.g., $\mathbf{x} = (x_1, \ldots, x_n)$ is a vector of length n, where $w_H(\mathbf{x})$ denotes the Hamming weight of $\mathbf{x}$. Double stroke capital letters are used to denote algebraic structures, e.g., $\mathbb{F}_q$ is a finite field with q elements. Regular uppercase letters are used to represent sets, e.g., S, where $|S|$ is the number of elements in S. Calligraphic capital letters are used to denote codebooks, e.g., $\mathcal{C}$.

Consider a multilevel memory whose levels are mapped into symbols in an alphabet of size q. In this paper, we refer to such a memory as a q-ary memory. The set of q symbols with addition and multiplication form an algebraic structure. If $q = p^t$ and p is prime, the algebraic structure is a finite field $\mathbb{F}_q$; otherwise, it is a ring $\mathbb{R}_q$ in which operations are computed modulo q. To simplify the text, when it is clear from the context, we denote the algebraic structure by $\mathbb{Z}_q$, and denote addition and subtraction by the symbols $\oplus$ and $\ominus$, respectively.

 ISBN: 978-1-61208-492-3

Known robust codes are defined over a finite field, i.e., the size of the alphabet, q, is a power of a prime. If the number of different states (voltage levels) that a memory cell can have in each write cycle is not a power of a prime, a robust code over a ring is required. Note that the computation of all these known robust codes over finite fields involves multiplication. However, in the case of a ring, there are elements in the ring with no multiplicative inverse, which may affect the analysis and the resulting error masking probability. In this section we introduce an extension of the QS code. The resulting code is a robust code over a ring. The maximal error masking probability of the extended code is different (higher) than the maximal error masking probability of the original QS code over a finite field.

The QS code is defined in [5] for the case where q is a power of a prime (PoP). The number of redundancy symbols in [5] is $r \leq k$. The code can be extended for q's which are not necessarily PoPs as follows:

Construction 1. *Let $q = \prod_{i=1}^{v} {p_i}^{t_i}$ where $p_i < p_{i+1}$. Let $k = 2sr$, where $r = 1$ if q is not a PoP. Let $\mathbf{x} = (x_1, x_2, \ldots, x_{2s})$ where $x_i \in \mathbb{Z}_q^r$ for $1 \leq i \leq 2s$. The code QS code is*

$$\mathcal{C} = \{(\mathbf{x}, \mathbf{u}) : \mathbf{x} \in \mathbb{Z}_q^{2s}, \mathbf{u} = \sum_{i=1}^{s} x_{2i-1}x_{2i} \in \mathbb{Z}_q^r\}.$$

Note that when q is not a PoP, we take $r = 1$ since a larger r cannot improve the code's efficiency. To simplify the notation, from here on, unless otherwise stated, q is not a PoP. The case where q is a PoP can be viewed as subcase of the general case with $p_1 = q$.

Let $\mathbf{e} = (\mathbf{e}_x, e_u)$ be an error vector, where $\mathbf{e}_x = (e_{x_1}, \ldots e_{x_{2s}}) \in \mathbb{R}_q^{2s}$ and $e_u \in \mathbb{R}_q$. The error is masked by a codeword $\mathbf{c}$ if $(\mathbf{x} \oplus \mathbf{e}_x, u \oplus e_u) \in \mathcal{C}$. In other words, the error masking equation of the code is

$$\sum_{i=1}^{s}(x_{2i-1} \oplus e_{x_{2i-1}})(x_{2i} \oplus e_{x_{2i}}) = \sum_{i=1}^{s} x_{2i-1}x_{2i} \oplus e_u. \quad (3)$$

Equivalently,

$$\mathbf{a}\mathbf{x}^T = b \quad (4)$$

where $\mathbf{a} \in \mathbb{R}_q^{2s}$ and $b \in \mathbb{R}_q$ are

$$a_i = \begin{cases} e_{x_{i+1}} & \text{if } i \text{ is odd} \\ e_{x_{i-1}} & \text{if } i \text{ is even} \end{cases}, \text{ and } b = e_u \ominus \sum_{i=1}^{s} e_{x_{2i-1}}e_{x_{2i}}.$$

Let $B(\mathbf{a})$ be the set

$$B(\mathbf{a}) = \{b | \exists \mathbf{x} : \mathbf{a}\mathbf{x}^T = b\}.$$

Clearly, $B(\mathbf{a}) = B(\mathbf{e}_x)$. To analyze which elements are in $B(\mathbf{a})$, it is convenient to use the greatest common divisor (gcd) over a set of nonzero integers; define $g(\mathbf{a}) \in \mathbb{R}_q$ as

$$g(\mathbf{a}) = \gcd(\{a_i | a_i \neq 0\} \cup \{q\}).$$

The set $B(\mathbf{a})$ contains all the multiples of $g(\mathbf{a})$ modulo q. Therefore $|B(\mathbf{a})| = \frac{q}{g(\mathbf{a})}$. In addition, for all $1 \leq i \leq 2s$, a_i is also in $B(\mathbf{a})$. For example, if $q = 6$, $r = 1$, $k = 2$ and $\mathbf{a} = (0, 4)$, then $g(\mathbf{a}) = 2$, $B(\mathbf{a}) = \{0, 2, 4\}$ and $|B(\mathbf{a})| = 3$.

Property 1. *Let $\mathbf{a} \in \mathbb{R}_q^{2s} \setminus \{\mathbf{0}\}$. Then, (4) has $q^k|B(\mathbf{a})|^{-1}$ solutions if $b \in B(\mathbf{a})$ and 0 solutions otherwise.*

For uniformly distributed codewords, the number of solutions of (4) for a given $\mathbf{e}$ defines the error masking probability;

Theorem 1. *Let q not be a PoP. Let $\mathcal{C}$ be a QS code where the codewords in $\mathcal{C}$ are uniformly distributed. The error masking probability of $\mathcal{C}$ for any nonzero error $\mathbf{e}$ is $Q(\mathbf{e}) = \frac{1}{|B(\mathbf{a})|}$ if $b \in B(\mathbf{a})$, and $Q(\mathbf{e}) = 0$ otherwise. In particular, the maximal error masking probability of the QS code is $\overline{Q}_{\mathcal{C}} = 1/p_1$.*

The set of all $\mathbf{e}_x$'s can be divided into subsets according to their error masking probability. Let $\overline{E}_x$ be the set of $\mathbf{e}_x$'s that have the maximal error masking probability. Any $\mathbf{e}_x \in \overline{E}_x$ can be written as $\mathbf{e}_x = \frac{q}{p_1}\tilde{\mathbf{e}}_x$ where $\tilde{\mathbf{e}}_x \in \mathbb{Z}_{p_1}^k$. Since $B(\mathbf{e}_x) = B(\mathbf{e'}_x)$ for all $\mathbf{e}_x, \mathbf{e'}_x \in \overline{E}_x$, there are $({p_1}^k - 1)p_1$ distinct error vectors $\mathbf{e}$ that maximize the error masking probability.

Example 1. *Consider the case where $q = 6$, $k = 2$, and $r = 1$. In this case, the set $\overline{E}_x = \{03, 30, 33\}$ and $B(\overline{E}_x) = \{0, 3\}$. Each one of the errors $030, 033, 300, 303, 330, 333$ has an error masking probability $Q(e) = 0.5 = \overline{Q}$.*

III. THE EXPURGATED CODE

Consider a k_2-bit binary word to be stored in a q-ary memory array where q is not a power of two. For converting the binary word into a q-ary word of length k_q, the k_2 bits are divided into blocks of w_2 bits which are then mapped into blocks of w_q q-ary symbols; whereas,

$$\lfloor \frac{k_2}{w_2} \rfloor w_q + \lceil (k_2 \mod w_2) \log_q 2 \rceil \leq k_q \leq \lceil \frac{k_2}{w_2} \rceil w_q.$$

For simplicity, we assume that $\frac{k_2}{w_2}$ is an integer (however, our results equally apply to non integers). Since q is not a power of two, some of the q-ary vectors are never used; denote by D_w the set of the combinations over in $\mathbb{F}_q^{w_q}$ that are never used,

$$|D_w| = q^{w_q} - 2^{w_2} < 2^{w_2}.$$

Denote by $D \subseteq \mathbb{Z}_q^{k_q}$ the set of q-ary vectors of length k_q that never occur at the output of the converter, and by $M = \mathbb{Z}_q^{k_q} \setminus D$ the set of vectors that can appear at the output of the converter. Each vector in D corresponds to a codeword in $\mathcal{C}$ that is never used; denote by $\mathcal{D} \in \mathcal{C}$ the set of unused codewords, and by $\mathcal{M} = \mathcal{C} \setminus \mathcal{D}$ the expurgated code,

$$\begin{aligned} |\mathcal{M}| &= |M| = 2^{k_2} > |\mathcal{C}|/2, \\ |\mathcal{D}| &= |D| = (q^{k_q} - 2^{k_2}) < |\mathcal{C}|/2. \end{aligned} \quad (5)$$

It is assumed that the codewords in $\mathcal{M}$ are uniformly distributed.

Clearly the error masking probability of $\mathcal{M}$ may be different from the error masking probability of $\mathcal{C}$. Denote by $R(\mathbf{e})$ the

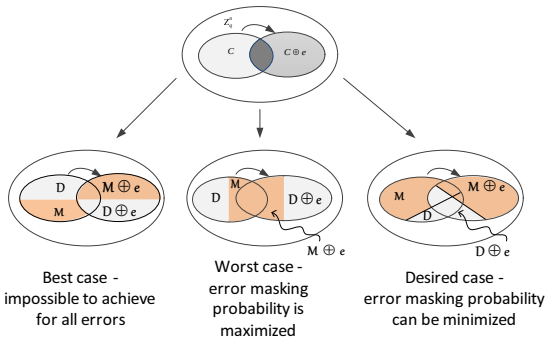

Figure 2. Three types of errors in expurgated codes.

number of codewords that mask the error vector $\mathbf{e}$,

$$R_{\mathcal{C}}(\mathbf{e}) = |\{\mathbf{c}|\mathbf{c} \in \mathcal{C} \text{ and } \mathbf{c} \oplus \mathbf{e} \in \mathcal{C}\}|.$$

For uniformly distributed codewords we have, $Q(\mathbf{e}) = R_{\mathcal{C}}(\mathbf{e})/|\mathcal{C}|$. Denote by $\Lambda_{\mathcal{C}_1,\mathcal{C}_2}(\mathbf{e})$ the cross-correlation from a code $\mathcal{C}_1$ to a code $\mathcal{C}_2$; i.e,

$$\Lambda_{\mathcal{C}_1,\mathcal{C}_2}(\mathbf{e}) = |\{\mathbf{c}|\mathbf{c} \in \mathcal{C}_1 \text{ and } \mathbf{c} \oplus \mathbf{e} \in \mathcal{C}_2\}|.$$

Since $\mathcal{C} = \mathcal{M} \cup \mathcal{D}$ and $\mathcal{M} \cap \mathcal{D} = \emptyset$ the autocorrelation of the code $\mathcal{C}$ can be rewritten as

$$R_{\mathcal{C}}(\mathbf{e}) = R_{\mathcal{M}}(\mathbf{e}) + \Lambda_{\mathcal{M},\mathcal{D}}(\mathbf{e}) + \Lambda_{\mathcal{D},\mathcal{M}}(\mathbf{e}) + R_{\mathcal{D}}(\mathbf{e}). \quad (6)$$

Figure 2 illustrates the contribution of each component in (6) to $R_{\mathcal{C}}(\mathbf{e})$ for three types of errors. The codewords of expurgated code $\mathcal{M}$ and its shifted set $(\mathbf{e}+\mathcal{M})$ are shown in red, and the codewords that correspond to $\mathcal{D}$ appear in light gray. $R_{\mathcal{M}}(\mathbf{e})$ is the number of codewords in the intersection of the two red areas. The best case is shown on the left hand side of the figure. Since $R_{\mathcal{M}}(\mathbf{e})$ is the autocorrelation of code $\mathcal{M}$, the best case is where $R_{\mathcal{M}}(\mathbf{e}) = 0$, however, it is impossible to achieve this for all errors, since for each two codewords $\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{M}$ there is an error vector $\mathbf{e}$ such that $\mathbf{c}_1 \oplus \mathbf{e} = \mathbf{c}_2$. The worst case is where $R_{\mathcal{M}}(\mathbf{e})$ is maximized. The desired case is where the maximal value of $R_{\mathcal{M}}(\mathbf{e})$ is minimized.

A standard checker of a separable code uses the k_q information symbols that are read from the memory to compute the expected redundant symbols. If the computed value matches the value of the r_q symbols stored in memory, the checker declares that no error has occurred, otherwise, it raises a flag. Such a checker masks an error $\mathbf{e}$ with a probability

$$\frac{R_{\mathcal{M}}(\mathbf{e}) + \Lambda_{\mathcal{M},\mathcal{D}}(\mathbf{e})}{|\mathcal{M}|}. \quad (7)$$

which may be higher than $\overline{Q}_{\mathcal{M}}$. This problem can be avoided if the checker also verifies that the received codeword belongs to M (i.e., it verifies that the information vector is a legal output of the converter). Now the error masking probability is reduced to the true error masking probability of $\mathcal{M}$, i,e,

$$Q_{\mathcal{M}}(\mathbf{e}) = \frac{R_{\mathcal{M}}(\mathbf{e})}{|\mathcal{M}|}.$$

In what follows we assume that the latter checker is used.

A. An upper bound on the error masking probability

Theorem 2. *The expurgated code $\mathcal{M}$ is robust. Its error masking probability is upper bounded by*

$$\overline{Q}_{\mathcal{M}} \leq \frac{|\mathcal{C}|}{p_1|\mathcal{M}|} < \frac{2}{p_1} = 2\overline{Q}_{\mathcal{C}}. \quad (8)$$

The error masking probability of the expurgated code depends on the choice of the set $\mathcal{M}$; in particular, $Q_{\mathcal{M}}(\mathbf{e})$ may be larger, smaller, or identical to the error masking probability of the original code. The following example demonstrates how sensitive the error masking probability is to the choice of $\mathcal{M}$.

Example 2. *Consider the case where $k_2 = w_2 = 6$ bits of information are converted to $k_q = w_q = 2$ symbols over alphabet $q = 10$, and are protected by a single redundant symbol. In this case, $|\mathcal{C}| = 10^2$, $|\mathcal{M}| = 2^6$ and $|\mathcal{D}| = |\mathcal{C}| - |\mathcal{M}| = 36$. The maximal error masking probability of the original QS code $\mathcal{C}$ is $\overline{Q}_{\mathcal{C}} = 0.5$.*

Let $D = \{10-19, 30-39, 50, 51-59, 70-75\}$. Consider the error vector $\mathbf{e} = 050$; the corresponding parameters are $\mathbf{a} = 50$ and $b = 0$. Note that for each $x \in D$, $a\mathbf{x}^T = 5x_1 \oplus 0x_2 \neq 0$; namely, all the vectors in D are not in $X_{\mathcal{C}}(050)$. Therefore, $R_{\mathcal{M}}(050) = R_{\mathcal{C}}(050) = 50$, and the error masking probability is $Q_{\mathcal{M}}(050) = \overline{Q}_{\mathcal{M}} \sim 0.78$. In the following section we introduce a method to choose a D which provides a $\overline{Q}_{\mathcal{M}}$ of 0.3125; this D consists of the following vectors:

$$\{02-09, 12-19, 20, 21, 24, 25, 30, 31, 34, 35, 40, 41, 50, 51, 60, 61, 70, 71, 80, 81, 90, 91\}.$$

B. A lower bound on the error masking probability

Denote by $X_{\mathcal{C}}(\mathbf{e})$ the set of the information words that mask an error $\mathbf{e}$,

$$X_{\mathcal{C}}(\mathbf{e}) = \{\mathbf{x}|(\mathbf{x}, \mathbf{u}(\mathbf{x})) \in \mathcal{C} \text{ and } (\mathbf{x}, \mathbf{u}(\mathbf{x})) \oplus \mathbf{e} \in \mathcal{C}\}.$$

Note that $|X_{\mathcal{C}}(\mathbf{e})| = R_{\mathcal{C}}(\mathbf{e})$.

The choice of D_w determines D, and hence $\mathcal{D}$. Denote by $\triangle(\mathbf{e})$ the difference between the number of codewords that mask $\mathbf{e}$ in $\mathcal{C}$ and the number of codewords that mask it in $\mathcal{M}$,

$$\triangle(\mathbf{e}) = R_{\mathcal{C}}(\mathbf{e}) - R_{\mathcal{M}}(\mathbf{e}).$$

If $\triangle(\mathbf{e})$ equals zero, $R_{\mathcal{C}}(\mathbf{e}) = R_{\mathcal{M}}(\mathbf{e})$ and $Q_{\mathcal{M}}(\mathbf{e})$ is maximized. If $\triangle(\mathbf{e}) > 0$ then $R_{\mathcal{M}}(\mathbf{e}) < R_{\mathcal{C}}(\mathbf{e})$ and $Q_{\mathcal{M}}(\mathbf{e})$ is smaller than its upper bound. From (6) it follows that

$$\triangle(\mathbf{e}) = \Lambda_{\mathcal{M},\mathcal{D}}(\mathbf{e}) + \Lambda_{\mathcal{D},\mathcal{M}}(\mathbf{e}) + R_{\mathcal{D}}(\mathbf{e}).$$

The sum $\Lambda_{\mathcal{D},\mathcal{M}}(\mathbf{e}) + R_{\mathcal{D}}(\mathbf{e})$ is the number of codewords that mask $\mathbf{e}$ in $\mathcal{C}$ and are in $\mathcal{D}$ and therefore are not in $\mathcal{M}$. In fact, it equals the size of the intersection between the set of codewords that mask $\mathbf{e}$ and the set of deleted codewords, that is,

$$\Lambda_{\mathcal{D},\mathcal{M}}(\mathbf{e}) + R_{\mathcal{D}}(\mathbf{e}) = |D \cap X_{\mathcal{C}}(\mathbf{e})|.$$

 ISBN: 978-1-61208-492-3

Similarly, denote by $(D - \mathbf{e}_x) = \{\mathbf{x} \ominus \mathbf{e}_x | \mathbf{x} \in D\}$, then

$$\Lambda_{\mathcal{M},\mathcal{D}}(\mathbf{e}) + R_{\mathcal{D}}(\mathbf{e}) = |(D - \mathbf{e}_x) \cap X_{\mathcal{C}}(\mathbf{e})|.$$

Therefore,

$$\triangle(\mathbf{e}) \leq |D \cap X_{\mathcal{C}}(\mathbf{e})| + |(D - \mathbf{e}_x) \cap X_{\mathcal{C}}(\mathbf{e})|. \quad (9)$$

Thus,

$$\triangle(\mathbf{e}) \leq 2|D|.$$

The rationale behind the choice of M (and hence, the choice of D), is to decrease $\overline{Q}_{\mathcal{M}}$ by decreasing the error masking probability of the errors that maximize it in $\mathcal{C}$; these errors form the set $\overline{E}_x$. In other words, denote by

$$\underline{\triangle} = \min_{\mathbf{e} \in \overline{E}_x, b \in B(\mathbf{e}_x)} \triangle(\mathbf{e}),$$

the minimal difference of the error masking probabilities over all the errors that maximize $Q_{\mathcal{C}}(\mathbf{e})$. The goal is to maximize $\underline{\triangle}$ so as to minimize $\overline{Q}_{\mathcal{M}}$.

Theorem 3. *The error masking probability of $\mathcal{M}$ is*

$$\overline{Q}_{\mathcal{M}} \geq \frac{(|\mathcal{M}| - |\mathcal{D}|)}{p_1 |\mathcal{M}|}. \quad (10)$$

Proof. Let $\mathbf{e}_x \in \overline{E}_x$. For all $b_i \neq b_j \in B(\mathbf{e}_x)$, we have,

$$\{\mathbf{x} | \mathbf{a}\mathbf{x}^T = b_i\} \cap \{\mathbf{x} | \mathbf{a}\mathbf{x}^T = b_j\} = \emptyset.$$

Recall that the size of $B(\mathbf{e}_x)$ is p_1. Therefore, for each $\mathbf{e}_x \in \overline{E}_x$ there are p_1 distinct non-empty and disjoint sets $X_{\mathcal{C}}(\mathbf{e})$.

Consider the intersection of an arbitrary set $S \subseteq \mathbb{Z}_{q^r}^{k_q}$ with all the sets $X_{\mathcal{C}}(\mathbf{e})$ where $\mathbf{e}_x \in \overline{E}_x$. The minimal size of the intersection is smaller or equal to the average; that is,

$$\min_{\mathbf{e}, \mathbf{e}_x \in \overline{E}_x \text{ and } b \in B(\mathbf{e}_x)} |S \cap X_{\mathcal{C}}(\mathbf{e})| \leq \frac{|S|}{p_1}.$$

By applying this upper bound to the sets D and $(D - \mathbf{e}_x)$, we get,

$$\underline{\triangle} \leq \min_{\mathbf{e} \in \overline{E}_x} |D \cap X_{\mathcal{C}}(\mathbf{e})| + |(D - \mathbf{e}_x) \cap X_{\mathcal{C}}(\mathbf{e})| \leq \frac{2|D|}{p_1}.$$

Therefore, the minimal difference of the errors in $\overline{E}_x$ is upper bounded by $\underline{\triangle} \leq 2|D|/p_1$ for any D, and

$$\overline{Q}_{\mathcal{M}} = \frac{\max_{\mathbf{e} \neq 0} R_{\mathcal{C}}(\mathbf{e}) - \underline{\Delta}}{|\mathcal{M}|} \geq \frac{(|\mathcal{M}| - |\mathcal{D}|)}{p_1 |\mathcal{M}|}.$$

□

C. The impact of the size of $\mathcal{M}$ on its error masking probability

Before we address the question of how to choose D_w (and hence, D), we need to relate to cases where the choice of D has no impact. In such cases, regardless of the choice of D, the error masking probability coincides with the worst case given in Th. 2; that is, $\underline{\triangle} = 0$.

Theorem 4. *If $p_1 = 2$, it is always possible to choose D such that $\underline{\triangle} > 0$. If $p_1 \neq 2$ and*

$$|D| \geq (p_1 - 1)k_q + 1,$$

it is possible to choose D such that $\underline{\triangle} > 0$; and similarly, if

$$|D_w| \geq (p_1 - 1)w_q + 1$$

then it is possible to choose D_w such that $\underline{\Delta} > 0$.

Proof ommitted.

IV. Converter structure

Usually, a converter is built from identical sub-blocks. Hence, it is sufficient to determine the set D_w of unused vectors for a single sub-block. In this section, it is assumed that $p_1 = 2$; however, with a small modification the results can be applied to other cases. Recall that we assume that $\frac{k_2}{w_2}$ is an integer, and that $k_q = 2s$. The case where $\frac{k_2}{w_2}$ is not an integer can be viewed as a subcase of this case. The output of the converter is a q-ary vector of length w_q, $\mathbf{x} = (x_1, \ldots x_{w_q})$.

We define a Hamming ball of dimension w_q and radius p_1 as the set

$$H^{w_q, p_1} = \Big\{ \mathbf{z}_j = \sum_{i=1}^{w_q} h_{j_i} \mathbf{v}_i \mid h_{j_i} \in \{0, \ldots, p_1 - 1\} \Big\},$$

where $\mathbf{v}_i = 0^{i-1} 1 0^{w_q - i - 1}$ is a unite vector of Hamming weight one.

The size of a Hamming ball is $p_1^{w_q}$. If $p_1 = 2$ then $|H^{w_q, p_1}|$ divides $|D_w|$. Hence, D_w can be a union of shifted disjoint Hamming balls.

The following construction is designed to maximize $\underline{\Delta}$, hence to minimize the error masking probability.

We start by defining a set of offset vectors Γ^{w_q},

$$\Gamma^{w_q} = \{\theta \in \mathbb{Z}_q^{w_q} \mid \theta = \sum_{i=1}^{k_q} \mathbf{v}_i p_1 t_i, \quad t_i \in \{0, \ldots, \frac{q}{p_1} - 1\}\}.$$

Notice that the symbols of θ are multiples of p_1. Therefore, for any two vectors $\theta_1 \neq \theta_2$, the intersection $(\theta_1 \oplus H^{w_q, p_1}) \cap (\theta_2 \oplus H^{w_q, p_1})$ is empty.

Construction 2 (Disjoint Hamming Balls)**.** *Define the set D_w as*

$$D_w = \bigcup_{\theta_i \in \Theta} \theta_i \oplus H^{w_q, p_1}. \quad (11)$$

where $\Theta \subseteq \Gamma^{w_q}$ is an arbitrary subset of offsets vectors, $|\Theta| = |D_w| / {p_1}^{w_q}$.

Recall that each information word is a concatenation of k_q/w_q vectors of length w_q. If one of these vectors is in D_w, the resulting information word is in D; if none of them is in D_w, the resulting information word is in M. In other words, the set of unused information words is

$$D = \bigcup_{\psi \in \Psi} \psi \oplus H^{k_q, p_1},$$

where a vector ψ is in Ψ if at least one of its k_q/w_q portions is a vector in Θ and the others are in Γ^{w_q}.

Theorem 5. *If D is chosen according to Const. 2 and $p_1 = 2$ then*

$$\frac{(|\mathcal{M}| - |\mathcal{D}|)}{p_1|M|} \leq Q_{\mathcal{M}}(\mathbf{e}) \leq \frac{1}{p_1} = \overline{Q}_{\mathcal{C}}.$$

Proof ommitted.

Is it possible to reach the lower bound on $\overline{Q}_{\mathcal{M}}$? Recall the proof of Theorem 3; in the proof, an upper bound on $\underline{\Delta}$ was obtained by adding the sizes of two sets. If these two sets are disjoint, an equality holds. In other words, it is possible to reach the lower bound on $\overline{Q}_{\mathcal{M}}$ if for all $\mathbf{x} \in D$, $\mathbf{x} \notin D - \mathbf{e}_x$. The following example shows that in some cases this situation cannot be avoided.

Example 3. *Consider the case where $q = 6, k_q = w_q = 3$ and $k_2 = w_2 = 7$. In this case D is a union of 11 shifted Hamming balls $|D| = 11 \cdot |H^{3,2}| = 88$, here $|\Psi| = 11$. In fact, there are $(q/2)^3 = 27$ possible vectors out of which Ψ is chosen. Therefore, there must be at least two linearly dependent vectors in Ψ. Without loss of generality, assume that $\psi_1 = 002$ and $\psi_2 = 004$ are in Ψ. In this case, for $\mathbf{e}_x = 003 \in \overline{E}_x$ and $\mathbf{x} = 005$ we have,*

$$\begin{aligned} \mathbf{x} &= (\psi_2 \oplus 001) \in D \\ \mathbf{x} &= (\psi_1 \oplus 000) \ominus 003 \in D - \mathbf{e}_x \end{aligned}$$

That is, $\mathbf{x}$ is both in D and $D - \mathbf{e}_x$.

V. Conclusion

This work analyzed the efficiency of robust codes when used to protect multilevel memories. When the code's alphabet size, q, is not a power of two, the binary information must be converted into a q-ary word. It was shown that this conversion can significantly degrade the error masking probability of the codes. However, by wisely designing the converter, the degradation of the code properties can be minimized. Bounds on the practical error masking probability were given. A construction for the converter in cases where the QS code is applied to the multilevel memory was provided. It was shown that this construction indeed reduces the error masking probability of the resulting code.

Acknowledgment

This research was supported by the ISRAEL SCIENCE FOUNDATION (grant No. 1200/12).

References

[1] S. Skorobogatov and R. Anderson, "Optical fault induction attacks," in *Cryptographic Hardware and Embedded Systems-CHES 2002*. Springer, 2003, pp. 2–12.

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

[3] H. Bar-El, H. Choukri, D. Naccache, M. Tunstall, and C. Whelan, "The sorcerer's apprentice guide to fault attacks," *Proceedings of the IEEE*, vol. 94, no. 2, pp. 370–382, 2006.

[4] R. E. Blahut, "Theory and practice of error control codes," *Reading*, 1985.

[5] M. Karpovsky, K. Kulikowski, and Z. Wang, "Robust error detection in communication and computational channels," *Spectral Methods and Multirate Signal Processing. SMMSP'2007. International Workshop on.*

[6] I. Shumsky, O. Keren, and M. Karpovsky, "Robustness of security-oriented binary codes under non-uniform distribution of codewords," in *DEPEND 2013, The Sixth International Conference on Dependability*, 2013, pp. 25–30.

[7] Y. Neumeier and O. Keren, "Robust generalized punctured cubic codes," *IEEE Transactions on Information Theory*, vol. 60, pp. 2813–2822, 2014.

[8] N. Admaty, S. Litsyn, and O. Keren, "Puncturing, expurgating and expanding the q-ary bch based robust codes," in *Electrical Electronics Engineers in Israel (IEEEI), 2012 IEEE 27th Convention of*, Nov 2012, pp. 1–5.

[9] M. Karpovsky and A. Taubin, "New class of nonlinear systematic error detecting codes," *Information Theory, IEEE Transactions on*, vol. 50, no. 8, pp. 1818–1819, 2004.

[10] M. Bauer, "Data path for multi-level cell memory, methods for storing and methods for utilizing a memory array," Mar. 24 2011, uS Patent App. 12/956,977. [Online]. Available: http://www.google.com/patents/US20110069548 [accessed: 2016-5-22]

[11] K. D. Akdemir, G. Hammouri, and B. Sunar, "Non-linear error detection for finite state machines," in *Information Security Applications*. Springer, 2009, pp. 226–238.

[12] A. Levina and S. Taranov, "Spline-wavelet robust code under non-uniform codeword distribution," in *Computer, Communication, Control and Information Technology (C3IT), 2015 Third International Conference on*. IEEE, 2015, pp. 1–5.

[13] R. Cramer, Y. Dodis, S. Fehr, C. Padró, and D. Wichs, "Detection of algebraic manipulation with applications to robust secret sharing and fuzzy extractors," in *Advances in Cryptology–EUROCRYPT 2008*. Springer, 2008, pp. 471–488.

[14] Z. Wang and M. Karpovsky, "Robust fsms for cryptographic devices resilient to strong fault injection attacks," in *On-Line Testing Symposium (IOLTS), 2010 IEEE 16th International*. IEEE, 2010, pp. 240–245.

 ISBN: 978-1-61208-492-3