New Error Detecting Codes for the Design of Hardware Resistant to Strong Fault Injection Attacks

Zhen Wang and Mark Karpovsky
Reliable Computing Laboratory, Boston University, Boston , USA
wang.zhen.mtk@gmail.com, markkar@bu.edu

Abstract—**Cryptographic devices suffer from fault injection attacks. The security of crypto-systems protected by traditional error detecting codes rely on the assumption that the information bits and the error patterns are not both controllable by the attacker. For applications where the assumption is not valid, the security of systems protected by traditional error detecting codes can be easily compromised. In this paper, we present constructions for algebraic manipulation detection (AMD) codes based on the nonlinear encoding functions. For a (k, m, r) AMD code, a message contains three parts: k-bit information data y, m-bit random data x and r-bit redundancy $f(y, x)$. For any error e and information y, the fraction of x that masks the error e is less than 1. In this paper we describe lower and upper bounds on AMD codes and show that the presented constructions can generate optimal or close to optimal AMD codes in many cases. We presented efficient encoding and decoding methods for AMD codes minimizing the number of multipliers using the multivariate Horner scheme. The proposed codes can provide a guaranteed high error detecting probability even if both the information bits of the code and the non-zero error patterns are controllable by an attacker. These codes can be used for design of secure multipliers, secure memories or secure hardware implementing cryptography algorithms resistant to fault injection attacks.**

Keywords-**Error Detecting Codes, Nonlinear Codes, Secure Hardware, Fault Injection Attacks.**

I. Introduction

Error detecting codes are widely used for communication channels and for computation channels to protect reliable and secure devices against soft errors, hard errors and malicious attacks in applications like Internet, data storage, cryptosystems and wireless communications.

Most of the existing reliable and secure architectures [1], [2], [3], [4], [5], [6] are based on linear codes such as 1-d parity codes, duplication codes, Hamming codes, BCH codes, Reed-Solomon codes, etc. The error detecting capabilities these architectures largely depend on the accuracy of the error model and may not be sufficient if an attacker can control errors distorting the received messages for communication channels or errors distorting outputs of a device protected by an error detecting code for computation channels.

Robust codes based on nonlinear encoding functions were proposed in [7], [8], [9], [10], [11]. A code $C \in GF(2^n)$ is robust if $\{e|c \oplus e \in C, \forall c \in C\} = \{\mathbf{0} \in GF(2^n)\}$. These codes can provide nearly equal protection against all error patterns [7], [8]. The error masking probabilities $Q_C(e) = |C|^{-1}|\{c \in C, c \oplus e \in C\}|$ ($|C|$ is the size of the code) for robust codes are upper-bounded by a number less than 1 for all non-zero errors. Compared to the systems based on linear codes, systems based on robust codes can provide a guaranteed protection regardless of the accuracy of the error model. Variants of robust codes – partially robust and minimum distance robust codes – were proposed in [10], [11], which allow tradeoffs in terms of the robustness and the hardware overhead.

One limitation of robust codes is that these codes assume the information bits of messages or outputs of the device-to-be-protected are uniformly distributed and are not controllable by external forces, e.g. by an attacker during error injection attacks on devices. The reliability and the security of the communication or computation channels protected by robust codes will be largely compromised if both information bits of the messages and the non-zero error patterns can be controlled by the attacker.

Example 1.1: Suppose the 32-bit device is protected by a robust duplication code $C = \{y, f(y)\}$, where $y, f(y) \in GF(2^{32})$, $f(y) = y^3$ and all operations are in $GF(2^{32})$. It is easy to prove that any non-zero error e will be masked by at most two codewords [7], i.e. for any non-zero error $e = (e_y, e_f)$ there exist at most two vectors $y_1, y_2 \in GF(2^{32})$ such that $(y_1 \oplus e_y)^3 = y_1^3 \oplus e_f$ and $(y_2 \oplus e_y)^3 = y_2^3 \oplus e_f$. Assume that an attacker cannot control the fault-free outputs y during attacks and the outputs of the original device are uniformly distributed, then the probability that the attacker conducts a successful attack ($e = (e_y, e_f)$ is not detected) is at most 2^{-31}. If an attacker has the ability to control the inputs of the device (hence the fault-free outputs) and can inject arbitrary error patterns at the output, let (v, y) be an input-output pair, i.e. y is the output of the device when the input to the device is v. Then the attacker can easily derive an error pattern $e^* = (e_y^*, e_f^*)$, $e_y^*, e_f^* \in GF(2^{32}), e_y^* \neq \mathbf{0}$ that will be masked by y, i.e. $(y \oplus e_y^*)^3 \oplus y^3 \oplus e_y^* = \mathbf{0}$. During the attack, the attacker can simply input v to the device and inject the corresponding $e^* = (e_y^*, e_f^*)$ at the output of the device. In this case, the attack will always be successful.

For the situation shown in the above example, all previous protection technologies based on traditional error detecting codes will not be sufficient. A coding technique based on adding to k information bits m random bits and r redundant bits, which can still provide guaranteed reliability and security under the above circumstance, is called **algebraic manipulation detection (AMD) code** . (The formal definition of AMD codes will be given in the next Section, see Definition 2.2). A simple AMD code was first presented in [12]. A much more versatile strong AMD code was introduced in

The first author now works for Mediatek Wireless, Inc. The work of the second author is sponsored by the NSF grant CNR 1012910.

[13], where the construction of optimal AMD codes was presented for $k = br$ information digits and $m = r$ random digits (r is the number of redundant digits). In [14], the authors introduced the concept of AMD codes and put all previous constructions in a unified framework. Compared to the widely used Message Authentication Codes, AMD codes do not require a secret key and have simpler encoding and decoding. Codes combining AMD codes and list-decoding are described in [15]. Applications of AMD codes for the design of non-malleable codes are presented in [16].

The main contributions of this paper are as follows. We present lower bounds for the probability of error masking for systematic AMD codes (Section II) and present several new constructions of systematic AMD codes (Section III), which are generalizations of the construction shown in [14]. Some of the presented codes are optimal or close to be optimal. We showed the relationship between AMD codes and classical codes such as the Generalized Reed-Muller codes and the Reed-Solomon codes (Section II and III). We also describe in Section IV an efficient encoding and decoding algorithm for the presented codes based on the multivariate Horner scheme.

The proposed codes can be used for many different applications such as robust secret sharing schemes, robust fuzzy extractors [14] and secure cryptographic devices resistant to fault injection attacks. All the codes described in this paper are binary. Generalization to a nonbinary case is straightforward.

II. Definitions and Bounds for Algebraic Manipulation Detection Codes

Throughout the paper we denote by $\oplus$ the addition in $GF(q), q = 2^r$. All the results presented in the paper can be easily generalized to the case where $q = p^r$ (p is a prime). Due to the lack of space, proofs for corollaries are omitted.

A code V with codewords $(y, x, f(y, x))$, where $y \in GF(2^k), x \in GF(2^m)$ and $f(y, x) \in GF(2^r)$, will be referred to as a (k, m, r) code. We will assume that y is a k-bit information, x is an m-bit uniformly distributed random vector (generated by a random number generator) and $f(y, x)$ is an r-bit redundant portion of the message $(y, x, f(y, x))$. The general architecture of using AMD codes for the protection of computation channels is shown in Figure 1.

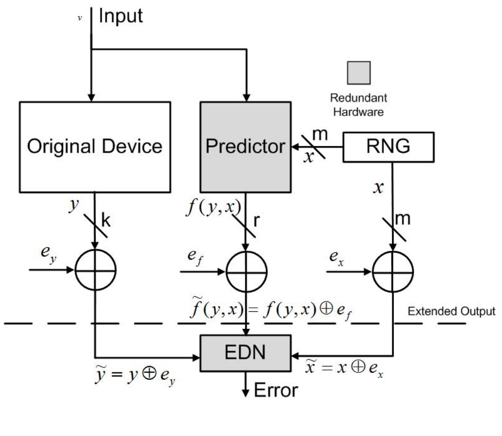

Fig. 1. Computation channel protected by a systematic (k, m, r) AMD code.

Definition 2.1: (**Security Kernel**) For any (k, m, r) error detecting code V with the encoding function $f(y, x)$, where $y \in GF(2^k), x \in GF(2^m)$ and $f(y, x) \in GF(2^r)$, the **security kernel** K_S is the set of errors $e = (e_y, e_x, e_f), e_y \in GF(2^k), e_x \in GF(2^m), e_f \in GF(2^r)$, for which there exists y such that $f(y \oplus e_y, x \oplus e_x) \oplus f(y, x) = e_f$ is satisfied for all x.

$$K_S = \{e | \exists y, f(y \oplus e_y, x \oplus e_x) \oplus f(y, x) \oplus e_f = \mathbf{0}, \forall x\}. \quad (1)$$

We note that in many applications $e_y \neq 0$ is a necessary condition for an attacker to conduct a successful fault injection attack. However, for secure architectures such as the one shown in [9], [17], the integrity of not only the information bits but also the redundant bits of the codes can be critical. Thereby, to conduct a more general analysis, we do not impose $e_y \neq 0$ in the above definition of the security kernel.

Non-zero errors e in the security kernel can be used by an advanced attacker to bypass the protection based on the error detecting code. For the case of communication channels we assume that an attacker can select any k-bit vector y as the information bits of a message $(y, x, f(y, x))$ and any error $e = (e_y, e_x, e_f)$ that distorts the message. For the case of computation channels (Figure 1), we assume the attacker can inject faults that manifest as $e \in K_S$ at the output of the device and select y for which e is always masked. Under the above attacker model for communication or computation channels, the attacker can always mount a successful attack. Thereby an AMD code that can provide a guaranteed error detecting probability under the above strong attacker model should have no errors in the security kernel except for the all zero vector in $GF(2^n)$, where $n = k + m + r$ is the length of the code.

Definition 2.2: A (k, m, r) error detecting code is called Algebraic Manipulation Detection (AMD) code iff $K_S = \{\mathbf{0}\}$, where $\mathbf{0}$ is the all zero vector in $GF(2^n)$, $n = k + m + r$.

Remark 2.1: The original definition of AMD codes in [14] is for both systematic and nonsystematic codes defined in any group. In this paper we consider binary systematic AMD codes, which is the most practical for hardware implementation. The above definition and all other results in this paper can be easily generalized for non-binary cases.

AMD codes $V = \{(y, x, f(y, x))\}$ have no undetectable errors no matter how the attacker select $e = (e_y, e_x, e_f)$ and y. AMD codes for the case $m = r$ and $k = br$ were introduced in [13] and were used in [14] for robust secret sharing schemes and for robust fuzzy extractors.

For a (k, m, r) code V, denote by $Q_V(y, e)$ the probability of missing an error e once y is fixed. Then $Q_V(y, e)$ can be computed as the fraction of random vectors x such that e is masked (see (2)) and $K_S = \{e | \exists y : Q_V(y, e) = 1\}$. The code V is an AMD code if and only if $Q_V(y, e) < 1$ for any y and any $e \neq 0$.

$$\begin{aligned} Q_V(y, e) \quad &= \quad 2^{-m} |\{x \,|\, (y, x, f(y, x)) \in V, \\ & \quad (y \oplus e_y, x \oplus e_x, f(y, x) \oplus e_f) \in V\}|. \quad (2) \end{aligned}$$

For a (k, m, r) AMD code $V = \{(y, x, f(y, x)), y \in GF(2^k), x \in GF(2^m), f(y, x) \in GF(2^r)\}$, for any given $y^* \in GF(2^k)$ and $e^* = (e_y^*, e_x^*, e_f^*)$, $e_y^* \in GF(2^k), e_x^* \in GF(2^m)$,

$e_f^* \in GF(2^r)$, $f(y^* \oplus e_y^*, x \oplus e_x^*) \oplus e_f^*$ considered as functions of $x \in GF(2^m)$ should all be different.

Example 2.1: Let $k = m = tr$, $y = (y_0, y_1, \cdots, y_{t-1})$, $y_i \in GF(2^r)$ be the information digits and $x = (x_0, x_1, \cdots, x_{t-1}), x_i \in GF(2^r)$ be the random digits. Let $f(y, x) = x_0 \cdot y_0 \oplus x_1 \cdot y_1 \oplus \cdots \oplus x_{t-1} \cdot y_{t-1}$ be the encoding function, where all the operations are in $GF(2^r)$.

It is easy to verify that when $e_y = \mathbf{0}$, for any e_x and e_f (e_x, e_f are not both $\mathbf{0}$), there always exist y such that $e = (\mathbf{0}, e_x, e_f), e \neq \mathbf{0}$ will be masked for all x. Thereby, this code is not a AMD code. In this case, K_S contains all vectors $e = (\mathbf{0}, e_x, e_f)$.

Suppose $e_y = (e_{y_0}, e_{y_1}, \cdots, e_{y_{t-1}}), e_{y_i} \in GF(2^r)$ is always non-zero. Without loss of generality, let us assume $e_{y_0} \neq \mathbf{0}$. Then the monomial $e_{y_0} \cdot x_0$ will appear in the error masking equation $f(x \oplus e_x, y \oplus e_y) \oplus f(y, x) \oplus e_f = \mathbf{0}$. Since $e_{y_0} \neq \mathbf{0}$, for every e, y and $x_1, x_2, \cdots, x_{t-1}$, there is a unique solution for x_0. Thereby the error is masked with probability 2^{-r}.

Let C be a q-ary code ($q = 2^r$) of length 2^m with an encoding function $f : GF(2^m) \to GF(2^r)$. Let us define the **orbit of** f by (3). We note that for any $f \in C$, $1 \leq |Orb(f)| \leq q2^m = 2^{m+r}$. If $|Orb(f)| = 2^{m+r}$, then for any e_x and e_f there exists x such that $f(x) \neq f(x \oplus e_x) \oplus e_f$. Moreover, if $\varphi \notin Orb(f)$, then $Orb(\varphi) \bigcap Orb(f) = \emptyset$.

$$Orb(f) = \{\varphi | \varphi(x) = f(x \oplus e_x) \oplus e_f, e_x \in GF(2^m), \quad (3)$$

where $e_f \in GF(2^r)\}$.

Definition 2.3: We will say that a q-ary ($q = 2^r$) code C of length 2^m is a **code with full orbit** if for any $f \in C$, $|Orb(f)| = 2^{m+r}$ and $Orb(f) \subseteq C$.

The notion of codes with full orbit will be used in the lower bound for the probability of error masking (see Theorem 2.1).

Any q-ary code C of length 2^m with full orbit is a union of disjoint orbits of size $q2^m$. The size of C is a multiple of $q2^m$. We note that codes with full orbit are nonlinear and for any code C with full orbit, $\mathbf{0} \in GF(2^m)$ is not a codeword of C.

Example 2.2: Let C be a binary code of length 8 and Hamming distance 2 containing all vectors with an odd number of $1's$. Let $y = (y_0, y_1, y_2), y_i \in GF(2)$ and $f_y(x) = y_0 \cdot x_0 \oplus y_1 \cdot x_1 \oplus y_2 \cdot x_2 \oplus x_0 \cdot x_1 \cdot x_2$. It is easy to verify that for any $y \in GF(2^3)$, $|Orb(f_y)| = 16$. Thus C is a code with full orbit and $|C| = |\cup_{y \in GF(2^3)} Orb(f_y)| = 128$.

The optimal AMD code should minimize $\max_{y, e \neq 0} Q_V(y, e)$ among all codes with the same parameters. Thus, the criterion we use to construct good AMD codes is

$$\min_{V \in V_{k,m,r}} \max_{y, e \neq 0} Q_V(y, e), \quad (4)$$

where $V_{k,m,r}$ is the set of all (k, m, r) error detecting codes.

We note that the optimization criterion selected in this paper is different from the one shown in [14]. The computational complexity of the encoding function for AMD codes is determined by both m and r. In cryptographic applications, the m random digits can be generated by a random number generator (RNG), which is already integrated in most of the modern cryptographic devices. Since the RNG is also used for other purposes such as generating the random mask for countermeasures against power analysis attacks, the number of random digits available for AMD codes in every clock cycle may be limited. The above criterion was selected to maximize the security level of the cryptographic device given the number of available random digits in every clock cycle and the amount of hardware redundancy we can bear.

Let $Q_V = \max_{y, e \neq 0} Q_V(y, e)$ and $Q(k, m, r) = \min_{V \in V_{k,m,r}} Q_V$. Denote by $\hat{d}_q(2^m, M)$ the maximum Hamming distance of a q-ary ($q = 2^r$) code of length 2^m with full orbit containing M codewords. Obviously,

$$\hat{d}_q(2^m, M) \leq d_q(2^m, M), \quad (5)$$

where $d_q(2^m, M)$ is the maximum possible Hamming distance of a q-ary code with length 2^m and M codewords.

We next present a lower bound for $Q(k, m, r)$. The constructions of codes providing tight upper bounds for $Q(k, m, r)$ can be found in Section III.

Theorem 2.1: For any (k, m, r) AMD code, where k is the number of information bits, m is the number of random bits and r is the number of redundant bits,

$$\begin{aligned} Q(k, m, r) &= \min_{V \in V_{k,m,r}} \max_{y, e \neq 0} Q_V(y, e) \\ &\geq 1 - 2^{-m} d_q(2^m, M), \quad (6) \end{aligned}$$

where $d_q(2^m, M)$ is the maximum possible Hamming distance of a (not necessarily systematic) q-ary code C ($q = 2^r$) with length 2^m and $M = |C| = 2^{k+m+r}$ codewords.

Proof: Let V be a (k, m, r) AMD code composed of vectors $(y, x, f(y, x))$, where $y \in GF(2^k), x \in GF(2^m)$ and $f(y, x) \in GF(2^r)$. When y is fixed, f is a function of x. Let us denote this function by f_y. Since V is an AMD code, $f_y(x \oplus e_x) \oplus e_f$ is not the same as $f_{y'}(x \oplus e'_x) \oplus e'_f$ for any $y, y', e_x, e'_x, e_f, e'_f$, assuming that elements of at least one of the pairs (y, y'), (e_x, e'_x) and (e_f, e'_f) are not equal. Thereby, for different y, e_x and e_f, $f_y(x \oplus e_x) \oplus e_f$ corresponds to 2^{k+m+r} different functions.

Let $C_V = \cup_{y \in GF(2^k)} Orb(f_y)$ be a q-ary ($q = 2^r$) code of length 2^m with full orbit. Then $|Orb(f_y)| = 2^{m+r}$, $|C| = 2^{k+m+r}$ and $Q_V = max_{y, e \neq 0} Q(y, e) = 1 - 2^{-m} d(C_V)$, where $d(C_V)$ is the Hamming distance of C_V. By (5) and (6) we have

$$\begin{aligned} Q(k, m, r) &= 1 - 2^{-m} \max_{V \in V_{k,m,r}} d(C_V) \\ &\geq 1 - 2^{-m} \hat{d}_q(2^m, M) \quad (7) \\ &\geq 1 - 2^{-m} d_q(2^m, M). \end{aligned}$$

■

The following Corollary follows directly from Theorem 2.1.

Corollary 2.1: There is no AMD codes V with $k > r2^m - m - r$. ($Q(k, m, r) = 1$ if $k > r2^m - m - r$.)

Remark 2.2: We note that the bound in Theorem 2.1 is much stronger than the trivial bound $Q(k, m, r) \geq 2^{-r}$. In fact, $Q(k, m, r) \geq 2^{-r}$ is equivalent to $d_q(2^m, 2^k + m + r) \geq 2^m - 2^{m-r}$, which is a sub-case of Theorem 2.1.

Theorem 2.1 shows the relationship between the worst case error masking probability Q_V for an AMD code V and the Hamming distance of the corresponding code C_V with full orbit. The exact value of $\hat{d}_q(2^m, M)$ is hard to derive. However, the Hamming distance of C_V should not exceed the maximum possible distance for a q-ary code with length 2^m and 2^{k+m+r} codewords, $q = 2^r$. We note that $d_q(2^m, M)$ can be estimated by classical bounds from coding theory such as the Hamming bound, the Johnson bound, the Singleton bound, the Plotkin bound, etc [18].

When $d_q(2^m, M)$ is estimated by the Singleton bound, the lower bound for $Q(k, m, r)$ can be written in a compact form as it is shown in the following Corollary.

Corollary 2.2: For any (k, m, r) AMD code,

$$Q(k, m, r) \geq \lceil \frac{k+m}{r} \rceil 2^{-m}. \tag{8}$$

Example 2.3: Let $k = m = 3$ and $r = 1$. According to (8), $Q(3, 3, 1) \geq \frac{6}{8}$. Let V be the code composed of all vectors $(y, x, f(y, x))$, where $y, x \in GF(2^3)$ and

$$f(y,x) = x_0 \cdot x_1 \cdot x_2 \oplus x_0 \cdot y_0 \oplus x_1 \cdot y_1 \oplus x_2 \cdot y_2, f(y,x) \in GF(2). \tag{9}$$

The error masking equation is $f(x \oplus e_x, y \oplus e_y) \oplus f(y, x) = e_f$, which is a polynomial of x with degree 2. The function on the left hand side of the error masking equation corresponds to a codeword of the second order binary Reed-Muller code $RM_2(2, 3)$ with 3 variables [18]. Any codeword of $RM_2(2, 3)$ has a Hamming weight of at least 2. Thus the number of solutions for the error masking equation is upper bounded by 6. V is a AMD code with $Q_V = \frac{6}{8}$. It follows from (8) that this code is optimal and $Q(3, 3, 1) = 0.75$.

Optimal (k, m, r) AMD codes attain the equality in (6) and minimize the worst case error masking probability among all codes with the same parameters.

In the next section, we will present several general constructions of AMD codes. Some of the generated codes are optimal with respect to the lower bounds (6) or (8).

III. Constructions of AMD Codes

The codewords of a (k, m, r) AMD code V are in the format $(y, x, f(y, x))$, where $y \in GF(2^k), x \in GF(2^m)$ and $f(y, x) \in GF(2^r)$. When y is fixed, f_y is a function of x. In the proof of Theorem 2.1, we have shown that the necessary condition for V to be an AMD code is that $f_y(x \oplus e_x) \oplus e_f$ cannot be the same function as $f_{y'}(x \oplus e'_x) \oplus e'_f$ for any $y, y', e_x, e'_x, e_f, e'_f$, assuming elements in at least one of the pairs (y, y'), (e_x, e'_x) and (e_f, e'_f) are not equal.

To compute $Q_V(y, e)$ for the code V, the error masking equation $f(x \oplus e_x, y \oplus e_y) \oplus f(y, x) \oplus e_f = \mathbf{0}$ should be evaluated for all 2^m possible $x \in GF(2^m)$.

We will say that an AMD code $V = \{(y, x, f(y, x))\}$ is based on code C if the error masking polynomial $f(y \oplus e_y, x \oplus e_x) \oplus f(y, x) \oplus e_f$ is a codeword of C for all y, e_x, e_y and e_f. Let us re-write $f(y, x)$ as $f(y, x) = A(x) \oplus B(y, x)$, where $A(x)$ is independent of y. We next show that by selecting $A(x)$ and $B(y, x)$ based on different error detecting codes such as the Generalized Reed-Muller codes and the Reed-Solomon codes, we can construct good (and in many cases optimal) AMD codes for different k and different $Q_V = \max_{y, e \neq 0} Q_V(y, e)$ for given m and r.

A. Constructions of AMD codes Based on the Generalized Reed-Muller Codes

Let $x = (x_0, x_1, \cdots, x_{t-1}), x_i \in GF(q)$, $q = 2^r$. A b^{th} order q-ary Generalized Reed-Muller code $GRM_q(b, t)$ [19] with t variables $(1 \leq b \leq t(q-1))$ consists of all codewords $(f(\mathbf{0}), f(\gamma^0), \cdots, f(\gamma^{q^t-2}))$, where $f(x)$ is a polynomial of t variables $x_0, x_1, \cdots x_{t-1}$ and γ is a primitive element of $GF(q^t)$. The degree of $f(x)$ is less or equal to b.

It is shown in [19] that the dimension of $GRM_q(b, t)$ is

$$k_{GRM_q(b,t)} = \sum_{j=0}^{t} (-1)^j \binom{t}{j} \binom{t+b-jq}{b-jq}, \tag{10}$$

where $q = 2^r$, $\binom{i}{j} = 0$ when $j < 0$.. If $b = u(q-1) + v, 0 \leq v \leq q-2$. Then the distance of $GRM_q(b, t)$ is $d_{GRM_q(b,t)} = (q-v)q^{t-u-1}$ [19].

Suppose $b + 2 = \alpha(q-1) + \beta \leq t(q-1), 0 \leq \alpha \leq t, 0 \leq \beta \leq q-2$. Assume that b is odd when $t = 1$. Let

$$A(x) = \begin{cases} \bigoplus_{i=0}^{t-1} x_i^{b+2} & \text{if } \alpha = 0, b \text{ is odd;} \\ \bigoplus_{i=1}^{t-1} x_0 x_i^{b+1}, t > 1 & \text{if } \alpha = 0, b \text{ is even;} \\ \bigoplus_{i=0}^{t-1} x_i^{\beta} \prod_{j=1}^{\alpha} x_{|i+j|_t}^{q-1} & \text{if } \alpha \neq 0, \alpha \neq t; \\ \prod_{i=0}^{\alpha-1} x_i^{q-1} & \text{if } \alpha = t; \end{cases} \tag{11}$$

where $x_i \in GF(2^r)$, $|i+j|_t$ is the modulo t addition, $\bigoplus$ is the sum in $GF(2^r)$.

Let

$$B(y, x) = \bigoplus_{1 \leq j_0 + j_1 + \cdots + j_{t-1} \leq b+1} y_{j_0, j_1, \cdots, j_{t-1}} \prod_{i=0}^{t-1} x_i^{j_i}, \tag{12}$$

where $0 \leq j_i \leq q-1$, $y_{j_0, j_1, \cdots, j_{t-1}} \in GF(2^r), x_i \in GF(2^r)$, $\prod_{i=0}^{t-1} x_i^{j_i}$ is a monomial of $x_0, x_1, \cdots, x_{t-1}$ of a degree between 1 and $b+1$ and $\prod_{i=0}^{t-1} x_i^{j_i} \notin \Delta B(y, x)$, where $\Delta B(y, x)$ is defined by (13).

We note it follows from (13) that when $\alpha = t$, $\Delta B(y, x) = \{x_i^{q-2} \prod_{j \neq i} x_j^{q-1}, 0 \leq i \leq t-1\}$.

Example 3.1: Let $r = 3, q = 8, t = 2$ and $b = 10$. Since $b + 2 = 12 = \alpha(q-1) + \beta$, we have $\alpha = 1$ and $\beta = 5$. By (11) and (13), we have $A(x) = x_0^5 x_1^7 \oplus x_0^7 x_1^5$ and $\Delta B(y, x) = \{x_0^5 x_1^6, x_0^6 x_1^5\}$. It is easy to verify that $A(x \oplus e_x) \oplus A(x) \oplus B(y \oplus e_y, x \oplus e_x) \oplus B(y, x)$ is always a non-zero polynomial corresponding to a codeword in $GRM_8(11, 2)$.

AMD codes can be constructed based on $A(x), B(y, x)$ and the Generalized Reed-Muller codes as shown in the next Theorem.

Theorem 3.1: Let $f(y, x) = A(x) \oplus B(y, x)$ be a q-ary polynomial with $y_{j_0, j_1, \cdots, j_{t-1}} \in GF(q)$ as coefficients and $x \in GF(q^t)$ as variables, where $1 \leq b \leq t(q-1) - 2, q = 2^r$ and $A(x), B(y, x)$ are as shown above. Suppose $b+2 = \alpha(q-1)+\beta$ and $b+1 = u(q-1)+v$, $0 \leq \alpha, u \leq t, 0 \leq \beta, v \leq q-2$. Assume $b + 2 \neq t(q-1) - 1$ and b is odd when $t = 1$. Then the code V composed of all vectors $(y, x, f(y, x))$ is an AMD

$$\Delta B(y,x) = \begin{cases} \{x_0^{b+1}, x_1^{b+1}, \cdots, x_{t-1}^{b+1}\} & \text{if } \alpha = 0, b \text{ is odd;} \\ \{x_1^{b+1}, x_0x_1^b, x_0x_2^b, \cdots, x_0x_{t-1}^b, t > 1\} & \text{if } \alpha = 0, b \text{ is even;} \\ \{x_i^{\beta} x_{|i+1|_t}^{q-2} \prod_{j=2}^{\alpha} x_{|i+j|_t}^{q-1}, 0 \le i \le t-1\} & \text{if } \alpha \neq 0. \end{cases} \tag{13}$$

code with $m = tr$,

$$\begin{aligned} k &= (k_{GRM_q(b+1,t)} - t - 1)r \\ &= (\sum_{i=0}^{t} (-1)^i \binom{t}{i} \binom{t+b+1-iq}{b+1-iq} - 1 - t)r, \end{aligned} \tag{14}$$

and

$$\begin{aligned} Q_V &= 1 - d_{GRM_q(b+1,t)} 2^{-m} \\ &= 1 - (2^r - v)2^{-(u+1)r}. \end{aligned} \tag{15}$$

Thus

$$\begin{aligned} & Q((\sum_{i=0}^{t} (-1)^i \binom{t}{i} \binom{t+b+1-iq}{b+1-iq} - 1 - t)r, tr, r) \\ & \le 1 - (2^r - v)2^{-(u+1)r}. \end{aligned} \tag{16}$$

Proof: An error e is masked by V if and only if for all x

$$A(x \oplus e_x) \oplus A(x) \oplus B(y \oplus e_y, x \oplus e_x) \oplus B(y,x) \oplus e_f = \mathbf{0}. \tag{17}$$

1) If $e_x = \mathbf{0}$ and $e_y = \mathbf{0}$, the error is always detected unless e_f is also $\mathbf{0}$. If $e_x = \mathbf{0}$ and $e_y \neq \mathbf{0}$, the left hand side of the error masking equation (17) is a polynomial of degree from 1 to $b+1$, which corresponds to a codeword of a $(b+1)^{th}$ order q-ary Generalized Reed-Muller code. Since $d_{GRM_q(b+1,t)} = (q-v)q^{t-u-1}$, there are at most $q^t - (q-v)q^{t-u-1}$ solutions for the error masking equation.

2) If $e_x \neq \mathbf{0}$, the left hand side of (17) does not contain any monomials of degree $b+2$ due to the fact that $A(x)$ and $A(x \oplus e_x)$ have exactly the same monomials of degree $b+2$. Moreover,

   a) If $\alpha = 0$ and b is odd, x_i^{b+1} appears in (17) iff x_i is distorted, $0 \le i \le t-1$;

   b) If $\alpha = 0$ and b is even, x_1^{b+1} appears in (17) iff x_0 is distorted, $x_0 x_i^b$ appears in (17) iff x_i is distorted $1 \le i \le t-1$;

   c) If $\alpha \neq 0$, since $b + 2 \neq t(q-1) - 1$, $x_i^{\beta} x_{|i+1|_t}^{q-2} \prod_{j=2}^{\alpha} x_{|i+j|_t}^{q-1}$ appears in (17) iff $x_{|i+1|_t}$ is distorted, $0 \le i \le t-1$. (When $\alpha = t$, ${x_i}^{q-2} \prod_{j \neq i} x_j^{q-1}$ appears in (17) if x_i is distorted.)

   Thereby, (17) always contains monomials of degree $b+1$, the left hand side of the error masking equation again is a codeword in $GRM_q(b+1,t)$. Thus the number of solutions for the error masking equation is still upper bounded by $q^t - (q-v)q^{t-u-1}$.

Thus for any fixed y and e, the probability Q_V of error masking is upper bounded by

$$(q^t - (q-v)q^{t-u-1})q^{-t} = 1 - (2^r - v)2^{-(u+1)r}.$$

The left hand side of (17) contains monomials of a degree from 1 to $b+1$ except for the t monomials from $\Delta B(y,x)$. Hence the number of different monomials in $B(y,x)$ is

$$k_{GRM_q(b+1,t)} - 1 - t = \sum_{i=0}^{t} (-1)^i \binom{t}{i} \binom{t+b+1-iq}{b+1-iq} - 1 - t. \tag{18}$$

The number, k, of bits in y is equal to the number of monomials in $B(y,x)$ multiplied by r, which is

$$(\sum_{i=0}^{t} (-1)^i \binom{t}{i} \binom{t+b+1-iq}{b+1-iq} - 1 - t)r. \tag{19}$$

■

Example 3.1 (Continued) For the code shown in Example 3.1, $k = 55 \times 3 = 165$. Since $b = 10 = u(q-1) + v, q = 8$, we have $u = 1$ and $v = 3$. The worst case error masking probability is $Q_V = 1 - 5 \times 2^{-6}$. Thus by (8), $1 - 7 \times 2^{-6} \le Q_V(165, 6, 3) \le 1 - 5 \times 2^{-6}$.

Corollary 3.1: When $b = t(q-1) - 2, q = 2^r$, codes generated by Theorem 3.1 are optimal. We have

$$Q(2^{tr}r - tr - 2r, tr, r) = 1 - 2^{-tr+1}. \tag{20}$$

1) Special Case: $r = 1$: For this case the dimension of a $(b+1)^{th}$ order binary Reed-Muller code of t variables is $k_{RM_2(b+1,t)} = \sum_{i=0}^{b+1} \binom{t}{i}$ $(t = m)$ [18]. The distance of $RM_2(b+1,t)$ is $d_{RM_2(b+1,t)} = 2^{t-b-1}$. As a result, the dimension of the resulting AMD code V constructed by Theorem 3.1 is $k = \sum_{i=0}^{b+1} \binom{t}{i} - t - 1$. The worst case error masking probability of the code is $Q_V = 1 - 2^{-(b+1)}$.

Corollary 3.2: When $q = 2$, the code V generated by Theorem 3.1 is a $(\sum_{i=0}^{b+1} \binom{t}{i} - t - 1, t, 1)$ AMD code with $Q_V = 1 - 2^{-(b+1)}$. The code is optimal when $b = 1$ or $b = t - 2$.

Example 3.2: Suppose $m = 7$ and $r = 1$. Let $b = 1$ and

$$f(y,x) = x_0 \cdot x_1 \cdot x_2 \oplus x_3 \cdot x_4 \cdot x_5 \oplus x_0 \cdot x_3 \cdot x_6 \oplus \sum_{i=0}^{6} x_i \cdot y_i. \tag{21}$$

It is easy to verify that $f(y \oplus e_y, x \oplus e_x) \oplus f(y,x) \oplus e_f$ is a polynomial of degree 2, which is a codeword of $RM_2(2,7)$. The distance of $RM_2(2,7)$ is 32. The worst case error masking probability of the resulting AMD code is $Q_V = 0.75$. This code is optimal.

Remark 3.1: When $q = 2$ and $b = 1$, the code V generated by Theorem 3.1 is an optimal AMD code with $k = \binom{t}{2}$ and $Q = 0.75$. Obviously, for all $k < \binom{t}{2}$, optimal AMD code with the same m and $r = 1$ can be constructed by deleting some codewords from V. The worst case error masking probability for the new code is still 0.75.

2) Special Case: $b \le q - 3$: Another special case of Theorem 3.1 is the case $b \le q-3$. In this case $k_{GRM_q(b+1,t)} = \binom{t+b+1}{t}$ and $d_{GRM_q(b+1,t)} = (q-b-1)q^{t-1}$ [19]. The dimension of the resulting AMD code is $(\binom{t+b+1}{t} - 1 - t)r$.

The worst case error masking probability is $(b+1)2^{-r}$.

Corollary 3.3: Assume b is odd when $t=1$. When $b \leq q-3$, the code V generated by Theorem 3.1 is a $((\binom{t+b+1}{t}-1-t)r, tr, r)$ AMD code with $Q_V=(b+1)2^{-r}$.

When $b=1$ $B(y,x)$ is the quadratic form $x_0 \cdot y_0 \oplus x_1 \cdot y_1 \oplus \cdots \oplus x_{t-1} \cdot y_{t-1}$, where all the operations are in $GF(2^r)$. If $e_y \neq 0$, it is easy to verify that the number of codewords masking the error is upper bounded by q^{t-1}.

3) Special Case: $t=1$ [13], [14]: When $t=1$ and b is odd, $A(x)=x^{b+2}$ and $B(y,x)=x \cdot y_0 \oplus x^2 \cdot y_1 \oplus \cdots \oplus x^b \cdot y_{b-1}$. The code generated by Theorem 3.1 coincides with the construction shown in [13], [14]. For this code, $k \leq r(q-3) = r(2^r-3)$.

Corollary 3.4: [13], [14] When $b \leq q-3$ is an odd number, the code V composed of all vectors $(y,x,f(y,x))$, where $y \in GF(q^{bt}), x \in GF(q), q=2^r$ and $f(y,x)=x^{b+2} \oplus x \cdot y_0 \oplus x^2 \cdot y_1 \oplus \cdots x^b \cdot y_{b-1}, f(y,x) \in GF(q)$, is an optimal (br,r,r) AMD code with $Q_V = \max_{y,e \neq 0} Q_V(y,e) = (b+1)2^{-r}$. Thereby, $Q(br,r,r)=(b+1)2^{-r}$.

Remark 3.2: One limitation of Corollary 3.4 is that b can only be an odd number when the characteristic of the field $GF(q)$ is 2. Otherwise, $A(x \oplus e_x)$ for $A(x)=x^{b+2}$ and $e_x \neq \mathbf{0}$ does not contain any monomial of degree $b+1$. The resulting code is not a secure AMD code as pointed out in [14]. When b is even, $A(x)$ can be chosen as x^{b+3}. In this case, $Q_V=(b+2)2^{-r}$.

Remark 3.3: When $t=1$, the left hand side of the error masking equation $f(y \oplus e_y, x \oplus e_x) \oplus f(y,x) \oplus e_f = \mathbf{0}$ is a codeword of an extended q-ary $(q, b+2, q-b-1)$ Reed-Solomon code, $q=2^r$ [18].

When $t>1$, codes V generated by Theorem 3.1 may have larger number of codewords than codes generated by Corollary 3.4 ($t=1$), assuming the two codes have the same Q_V and the same r.

Example 3.3: Suppose $r=16$, $Q_V=2^{-14}$. Then for $t=1$ and $b=3$, for codes generated by Corollary 3.4, the maximum number of codewords is $2^{br}=2^{48}$. When $t>1$, the maximum number of codewords for codes generated by Theorem 3.1 depends not only on b but also on t. When $t=2$, for example, the number of codewords of codes generated by Theorem 3.1 can be $2^{(\binom{t+b+1}{t}-1-t)r}=2^{192}$.

To end the section, we summarize cases the when codes constructed by Theorem 3.1 are optimal in the Table I.

IV. Encoding and Decoding Complexity for AMD Codes

In this section, we estimate the hardware complexity for the encoders and decoders for AMD codes based on q-ary Generalized Reed-Muller codes (Theorem 3.1). The hardware complexity for the encoders and decoders for AMD codes based on the product of GRM codes can be estimated in a similar way.

It is well known that a multivariate polynomial of t variables $x_i, 0 \leq i \leq t-1, x_i \in GF(2^r)$ can be efficiently computed using the multivariate Horner scheme [20]. When $t=1$, any polynomial of degree $b+1$ defined over $GF(2^r)$ can be represented as

$$f(x)=a_0 \oplus x(a_1 \oplus x(\cdots(a_b+a_{b+1}x))),$$

where $a_i \in GF(2^r), x \in GF(2^r)$. The computation of the polynomial requires $b+1$ multipliers and $b+1$ adders in $GF(2^r)$.

When $t>1$, we can first apply Horner scheme as if x_0 is the variable and $x_1, x_2, \cdots, x_{t-1}$ are coefficients. In this case coefficients will be polynomials of $t-1$ variables $x_1, x_2, \cdots, x_{t-1}$. To compute these polynomials, we can select one of the remaining $x_i, 1 \leq i \leq t-1$ as variable and apply the Horner scheme again. We repeat the procedure until all $x_i, 0 \leq i \leq t-1$ are factored out.

Example 4.1: In Theorem 3.1, let $t=b=2$ and assume r is large enough. Then the resulting code is a $(7r, 2r, r)$ AMD code. At most 8 multipliers and 7 adders in $GF(2^r)$ are required for the encoding or the decoding. The corresponding encoding network is shown in Figure 2. The critical path of the encoder contains 4 multipliers and 4 adders in $GF(2^r)$.

It is easy to verify that for the encoder of a (k,m,r) AMD code generated by Theorem 3.1 using the multivariate Horner scheme shown above, the number of multipliers and adders is upper bounded by $\lceil \frac{k+m}{r} \rceil$. The latency of the encoder will be $(b+1)(T_M+T_A)$, where T_M and T_A are the latency for a multiplier and an adder in $GF(2^r)$. We note that the actual number of multipliers in the encoder may be smaller than $\lceil \frac{k+m}{r} \rceil$ due to the fact that the power operation can be simplified. For example, in the normal base Galois field, the square operation can be implemented by cyclic shifting [21]. In this case, the multiplier marked in Figure 2, which is used to compute x_1^2, is not needed and the total number of multipliers in the encoder becomes 7.

An estimation of the overhead for secure Galois field multipliers based on AMD codes in $GF(2^{239})$ and $GF(2^{409})$ for the elliptic curve cryptographic devices can be found in [22]. We showed in [22] that the area overhead for the protection architectures based on AMD codes is between 110% and 160%. Moreover, when the encoder is pipelined, the protected multiplier has no latency penalty and can achieve the same performance as the original device. (The work in [22] only considered the special case of AMD codes with $b \leq q-3$ (see Section III-A2).)

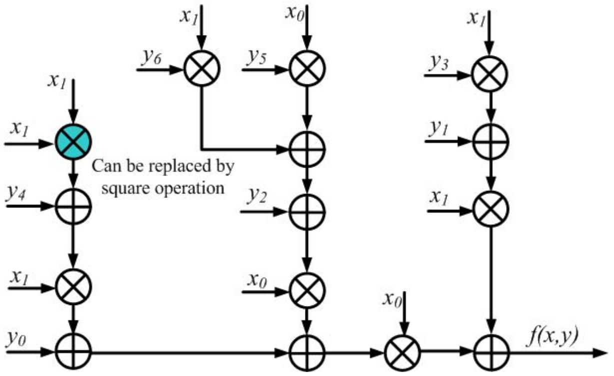

Fig. 2. Encoder Architecture for the $(7r, 2r, r)$ AMD Code Based on $GRM_q(3,2)$ code

TABLE I
OPTIMALITY OF (k, m, r) AMD CODES CONSTRUCTED BY THEOREM 3.1

k	m	r	Q_V	Optimality
$2^{tr}r - tr - 2r$	tr	r	$1 - 2^{-tr+1}$	Optimal (Corollary 3.1)
$\sum_{i=0}^{b+1} \binom{t}{i} - t - 1$	t	1	$1 - 2^{-(b+1)}$	Optimal when $b = 1$ or $b = t - 2$ (Corollary 3.2)
$(\binom{t+b+1}{t} - t - 1)r$	tr	r	$(b+1)2^{-r}$	Optimal when $t = 1$ (Corollary 3.4)

V. CONCLUSIONS

In this paper, we presented bounds, general constructions and encoding/decoding procedures for algebraic manipulation detection (AMD) codes based on q-ary Generalized Reed-Muller codes and their products. Some of the presented codes are optimal. These codes can provide a guaranteed level of reliability and security even if both the information bits and the non-zero error patterns are controllable by external forces. The same characteristic cannot be achieved by any previously known reliable and secure architectures based on error detecting codes. These codes can be applied for many different applications such as robust secret sharing scheme, robust fuzzy extractors and secure cryptographic devices resistant to fault injection attacks. An efficient encoding and decoding method minimizing the number of required multipliers are given for the presented AMD codes.

ACKNOWLEDGEMENT

We would like to thank Prof. Leonid Reyzin from the Computer Science Department of Boston University in the United States and Prof. Keren Osnat from the Bar Ilan university in Israel for their important comments and suggestions for the paper.

REFERENCES

[1] T. Malkin, F.-X. Standaert, and M. Yung, "A comparative cost/security analysis of fault attack countermeasures," in *Fault Diagnosis and Tolerance in Cryptography*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2006, vol. 4236, pp. 159–172.

[2] G. Bertoni, L. Breveglieri, I. Koren, P. Maistri, and V. Piuri, "Error analysis and detection procedures for a hardware implementation of the advanced encryption standard," *IEEE Transactions on Computers*, vol. 52, no. 4, pp. 492–505, 2003.

[3] G. Bertoni, L. Breveglieri, I. Koren, and V. Piuri, "Fault detection in the advanced encryption standard," in *Proc. of the International Conference on Massively Parallel Computing Systems (MPCS 2002)*, Ischia, Italy, Apr. 2002, pp. 92–97.

[4] N. Joshi, K. Wu, and R. Karri, "Concurrent error detection schemes for involution ciphers," in *Cryptographic Hardware and Embedded Systems - CHES 2004*, ser. Lecture Notes in Computer Science, vol. 3156. Springer Berlin / Heidelberg, 2004, pp. 153–160.

[5] R. Karri, K. Wu, P. Mishra, and Y. Kim, "Concurrent error detection of fault-based side-channel cryptanalysis of 128-bit symmetric block ciphers," in *38th Design Automation Conference (DAC 2001)*. ACM Press, 2001, pp. 579–585.

[6] ——, "Concurrent error detection schemes for fault-based side-channel cryptanalysis of symmetric block ciphers," *IEEE Transactions on CAD of Integrated Circuits and Systems*, vol. 21, no. 12, pp. 1509–1517, 2002.

[7] M. G. Karpovsky and A. Taubin, "New class of nonlinear systematic error detecting codes," *IEEE Transactions on Information Theory*, vol. 50, no. 8, pp. 1818–1820, 2004.

[8] M. Karpovsky., K. Kulikowski, and A.Taubin, "Robust protection against fault-injection attacks on smart cards implementing the advanced encryption standard," ser. Proc. Int. Conference on Dependable Systems and Networks (DNS 2004), July 2004.

[9] G. Gaubatz, B. Sunar, and M. G. Karpovsky, "Non-linear residue codes for robust public-key arithmetic," in *Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC '06)*, 2006.

[10] K.Kulikowski, Z.Wang, and M.G.Karpovsky, "Comparative analysis of fault attack resistant architectures for private and public key cryptosystems," in *Proc of Int. Workshop on Fault-tolerant Cryptographic Devices*, 2008.

[11] Z. Wang, M. Karpovsky, and K. Kulikowski, "Design of memories with concurrent error detection and correction by nonlinear SEC-DED codes," *Journal of Electronic Testing*, pp. 1–22, 2010, 10.1007/s10836-010-5168-5. [Online]. Available: http://dx.doi.org/10.1007/s10836-010-5168-5

[12] S. Cabello, C. Padr, and G. Sez, "Secret sharing schemes with detection of cheaters for a general access structure," *Designs, Codes and Cryptography*, vol. 25, pp. 175–188, 2002, 10.1023/A:1013856431727. [Online]. Available: http://dx.doi.org/10.1023/A:1013856431727

[13] Y. Dodis, B. Kanukurthi, J. Katz, L. Reyzin, and A. Smith, "Robust fuzzy extractors and authenticated key agreement from close secrets," in *In Advances in Cryptology CRYPTO 6*. Springer, 2006, pp. 232–250.

[14] 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 *Proceedings of the theory and applications of cryptographic techniques, 27th annual international conference on Advances in cryptology*, ser. EUROCRYPT'08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 471–488. [Online]. Available: http://portal.acm.org/citation.cfm?id=1788414.1788441

[15] V. Guruswami and A. Smith, "Codes for computationally simple channels: Explicit constructions with optimal rate," in *51st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 10 2010.

[16] S. Dziembowski, K. Pietrzak, and D. Wichs, "Non-malleable codes," in *Innovation in Computer Science, Cryptology ePrint Archive: Report 2009/608*.

[17] Z. Wang, M. Karpovsky, B. Sunar, and A. Joshi, "Design of reliable and secure multipliers by multilinear arithmetic codes," in *Information and Communications Security*, ser. Lecture Notes in Computer Science, 2009, vol. 5927, pp. 47–62.

[18] F. MacWilliams and N. Sloane, *The Theory of Error-Correcting Codes*. North-Holland, 1998.

[19] E. F. Assmus, Jr, and J. D. Key, "Polynomial codes and finite geometries, Manuscript," 1995.

[20] M. Ceberio and V. Kreinovich, "Greedy algorithms for optimizing multivariate horner schemes," *SIGSAM Bull.*, vol. 38, pp. 8–15, March 2004. [Online]. Available: http://doi.acm.org/10.1145/980175.980179

[21] S. Gao, "Normal bases over finite fields," Ph.D. dissertation, University of Waterloo, 1993.

[22] Z. Wang and M. Karpovsky, "Algebraic manipulation detection codes and their applications for design of secure cryptographic devices," in *IEEE 17th International On-Line Testing Symposium (IOLTS)*, 2011, pp. 234–239.