Systematic single limited magnitude error correcting codes for Flash Memories

Torleiv Kløve
Department of Informatics
University of Bergen
Pb. 7803, N-5020 Bergen, Norway
Email: Torleiv.Klove@ii.uib.no

Bella Bose
School of Electrical Engineering
and Computer Science
Oregon State University
Corvallis, OR 97331
Email: bose@eecs.orst.edu

Noha Elarief
School of Electrical Engineering
and Computer Science
Oregon State University
Corvallis, OR 97331
Email: elarief@eecs.orst.edu

***Abstract*—A relatively new model of error correction is the limited magnitude error model. That is, it is assumed that the absolute difference between the sent and received symbols is bounded above by a certain value l. In this paper, we propose systematic codes for asymmetric limited magnitude channels that are able to correct a single error. We also show how this construction can be slightly modified to design codes that can correct a single symmetric error of limited magnitude. The designed codes achieve higher code rate than single error correcting codes previously given in the literature.**

I. INTRODUCTION

In the asymmetric error model, a symbol a over an alphabet $Z_q = \{0, 1, \ldots, q-1\}$ may be modified during transmission into b, where $b \leq a$ (assuming that the dominant error type is the decreasing error). For some applications, the error magnitude $b-a$ is not likely to exceed a certain threshold l. One such application is the multi-level flash memory [3]. A multi-level flash cell is electrically programmed into one of q threshold states and thus can be viewed as storing one symbol from the set $\{0, 1, \ldots, q-1\}$. Moreover, errors in this type of memory are typically in one direction (known a priori) and have small magnitudes that may be significantly lower than the size of the alphabet [3]. Therefore, the limited magnitude asymmetric error model is well suited for such application. Systematic and non-systematic codes correcting *all* asymmetric errors with maximum magnitude l are given in [5] and [1] respectively. Although the latter codes are optimal, the code rate is very low since they correct all errors. In practice, only few errors may occur and thus it is more efficient to design codes that correct t or fewer errors. In [3], the authors proposed systematic codes that can correct up to t asymmetric errors of maximum magnitude l. In this paper, we propose systematic codes which correct single limited magnitude asymmetric errors (single l-AEC codes). The proposed codes achieve higher rate than the ones given in [3] for the case where $t = 1$. In the analysis and code design, it is assumed that wrap-around error is possible, i.e. a transmitted digit c_i can be received as $(c_i+e_i) \pmod q$ where $0 \leq e_i \leq l$. We illustrate the main construction by the following examples:

Example 1: Suppose we want to construct a single 1-AEC code over $q = 4$. Consider the following parity check matrix for a code C:

$$H = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 3 & 3 & 3 & 3 \\ 1 & 2 & 3 & 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 \end{bmatrix}.$$

Then C has length 15 and uses 2 check digits. Let $c \in C$ and e be a vector of length 15 with i^{th} component equal to 1 $(1 \leq i \leq n)$ and all other components equal to 0, then

$$(c+e)H^T = cH^T + eH^T = eH^T.$$

Note that $c+e$ is a vector suffering a single asymmetric error of magnitude 1. It is easy to see that the multiplication eH^T gives the transpose of the i^{th}column of H. Since the columns of H are all distinct, the error location can be determined and the error is corrected.

In general, when $l = 1$, the columns of the parity check matrix are all combinations of r column vectors over q (except the all-zero combination). Therefore the length of the code is $n = q^r - 1$. In Section III we will show that this construction is optimal. For higher values of l the construction is less straightforward.

Example 2: Let $q = 5$, $l = 2$ and $r = 2$ check digits. We want the column vectors and two times the column vectors of H to be all distinct $\pmod 5$. This is because the error vector can either be e as in the previous example or $2e$ (up to le in general). We note that 2 is a primitive root modulo 5: $2^0 = 1,\ 2^1 = 2,\ 2^2 = 4,\ 2^3 = 3,\ 2^4 = 1 \pmod 5$ and hence if the leading non-zero elements of the columns of H are taken as the alternating powers of 2 (i.e 1 and 4 or 2 and 3) the desired condition will be satisfied. That is

$$H = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 1 & 4 & 4 & 4 & 4 & 4 \\ 1 & 4 & 0 & 1 & 2 & 3 & 4 & 0 & 1 & 2 & 3 & 4 \end{bmatrix}$$

is a parity check matrix for a single 2-AEC code.

Later in this paper, we will show that the above construction is also optimal. The main essence of the code construction is to find a sequence of numbers that can be used as the leading non-zero term in the columns of the parity check matrix as illustrated in the above examples. These sequences are special cases of modular $B_h(S)$ sequences which will be defined in Section II. Methods to find those sequences are also given in Section II. In Section III, the code construction and the optimality of the code are considered. In Section IV, we show that a similar idea can be used to construct codes correcting single symmetric errors of limited magnitude. Concluding remarks are given in Section V.

II. THE B SEQUENCE

A. *The B_h sequence*

For integers a, b, where $a \leq b$, we let

$$[a,b] = \{a, a+1, a+2, \ldots, b\}.$$

For a set S of integers, a $B_h(S)$ sequence of length m is a sequence of m distinct positive integers $b_0, b_1, \ldots, b_{m-1}$ such that all sums

$$\sum_{j=1}^{h} a_j b_{i_j},$$

where $0 \leq i_1 < i_2 < \cdots < i_h \leq m-1$ and $a_j \in S$, are distinct.

A modular $B_h(S)$ sequence of length m and modulus v is a sequence of m distinct positive integers such that all sums

$$\Big(\sum_{j=1}^{h} a_j b_{i_j}\Big) \pmod v$$

where $0 \leq i_1 < i_2 < \cdots < i_h \leq m-1$ and $a_j \in S$, are distinct. Note that any $B_h(S)$ sequence is trivially a modular sequence modulo v for sufficiently large v. Most known results are on $B_h([0,1])$ sequences, also known as "B_h sequences" and "distinct sum sets". B_2 sequences are known as "Sidon sequences" and also "distinct difference sets". The reason is that if

$$c_{i_1} + c_{i_2} \neq c_{j_1} + c_{j_2}$$

then

$$c_{i_1} - c_{j_1} \neq c_{j_2} - c_{i_2}.$$

Therefore all sums of two elements are distinct if and only if all differences of two elements are distinct. Similar results are also valid under modulo v. Other names for such distinct difference sets are "difference triangle sets" and "Golomb rulers". Famous modular distinct difference sets are the Singer and the Bose-Chowla sets. For extensive litterature on such sets refer to [4] pp. 419-437.

In our future work, we will show that a modular $B_t([0,l])$ sequence can be used to construct l-AEC codes correcting t errors. In this paper, we are interested in modular $B_1(S)$ sequences.

B. *General construction of modular $B_1([0,l])$ sequences*

Given m and l, a modular $B_1([0,l])$ sequence $(b_0, b_1, \ldots, b_{m-1})$ modulo q must have $q \geq ml$. We give a construction where q is not much larger than this.

Theorem 1: Let p be a prime, $p \geq m$ and $p \geq l+1$. Let $q = p(l+1)$ then the sequence $b_i = i(l+1)+1$ for $0 \leq i \leq m-1$ is a $B_1([0,l]) \pmod q$.

Proof: We prove that by contradiction. Suppose that $0 < x_i \leq l, 0 \leq x_j \leq l$, and $b_i x_i \equiv b_j x_j \pmod q$ that is

$$(i(l+1)+1)x_i \equiv (j(l+1)+1)x_j \pmod{p(l+1)}.$$

In particular, this implies that

$$x_i \equiv x_j \pmod{l+1}.$$

Therefore, $x_j = x_i > 0$. And hence

$$i(l+1)x_i \equiv j(l+1)x_i \pmod{p(l+1)},$$

and so

$$i x_i \equiv j x_i \pmod p.$$

But since $0 < x_i \leq l < p$, then $i \equiv j \pmod p$. Finally, since $p \geq m$, this implies that $i = j$. ■

C. *Special cases*

We can make use of the special properties of l and q to construct maximal-length $B_1([0,l])$ sequences. Let q be a prime such that the order α of 2 modulo q is even. Then $B = \{b_i = 2^{2i} \pmod q \mid 0 \leq i \leq \alpha/2 - 1\}$ is a $B([0,2])$ modulo q since $2b_i = 2^{2i+1} \pmod q \notin B$. In particular, the construction is best possible if $l = 2$ is a primitive root of q (that is $\alpha = q-1$).

A corresponding result for $l = 3$ is given below:

Theorem 2: If $q \equiv 1 \pmod 3$, 3 is a primitive root modulo q and $2 \equiv 3^{\beta} \pmod q$ where $\beta \equiv 2 \pmod 3$, then

$$B = \{3^{3i} \pmod q \mid 1 \leq i \leq (q-1)/3\}$$

is a $B_1([0,3]) \pmod q$.

Proof: It can be seen that $2c_i \equiv 3^{3\mu_i+2}$ for some μ_i and $3c_i \equiv 3^{3\nu_i+1}$ for some ν_i. And thus $2c_i$ and $3c_i \notin B$. ■

Examples of q and β meeting the criteria in Theorem 2 with $q < 1000$:

q	139	163	379	571	607	631	751	859
β	101	77	149	545	584	98	416	137

When $l \geq 4$, we get the following:

Theorem 3: Let q be a prime such that the order α of l modulo q is a multiple of l. Let

$$B = \Big\{ c_i = l^{li} \pmod q \;\Big|\; 0 \leq i \leq \frac{\alpha}{l} - 1 \Big\}.$$

If $ab^{-1} \pmod q \notin B$ for $1 \leq a < b \leq l$, then B is a $B_1([0,l]) \pmod q$

Proof: Suppose that $ac_i \equiv bc_j \pmod q$ for some $a, b \in [1, l-1]$ where, w.l.o.g., $j \geq i$. Then

$$ab^{-1} \equiv c_j/c_i \equiv c_{j-i} \pmod q \in B.$$

By assumption, this implies that $a = b$ and so $i = j$. ■

Example 3: For $5 \leq i \leq 12$ there are primes $q \equiv 1 \pmod l$ such that the order of l modulo q is $\alpha = (q-1)/2$. Moreover for $5 \leq i \leq 12$ there are primes $q \equiv 1 \pmod l$ such that the order of l modulo q is $\alpha = (q-1)/2$ and such that the conditions of Theorem 3 are satisfied. The smallest primes are:

l	smallest primes
5	$281, 421, 701, 1051, 1231, 1301, 1471, 1571, 1951$
6	$73, 673, 769$
7	$3557, 3613, 4481$
8	$929, 1697, 2081$
9	$487, 1063, 2539$
10	$4441, 11681, 15881$
11	$8009, 16633$
12	$6217, 6673, 7873$

In this table, the smallest prime is 73, for $l = 6$, and we give a detailed description as an illustration. The order of 6 modulo 73 is $(73-1)/2 = 36$. We have $6^6 \equiv 9 \pmod{73}$,

$$\begin{aligned} B &\equiv \{6^0, 6^6, 6^{12}, 6^{18}, 6^{24}, 6^{30}\} \pmod{73} \\ &= \{1, 9, 8, 72, 64, 65\}, \end{aligned}$$

and

i	0	1	2	3	4	5
$1 \cdot 6^{6i}$	1	9	8	72	64	65
$2 \cdot 6^{6i}$	2	18	16	71	55	57
$3 \cdot 6^{6i}$	3	27	24	70	46	49
$4 \cdot 6^{6i}$	4	36	32	69	37	41
$5 \cdot 6^{6i}$	5	45	40	68	28	33
$6 \cdot 6^{6i}$	6	54	48	67	19	25

D. Examples of modular $B_1([0,2])$ found by a greedy algorithm

We want positive integers $c_0, c_1, \ldots, c_{m-1}$ and a modulus q such that $|\{c_i \mid 0 \le i \le m-1\} \cup \{2c_i \pmod{v} \mid 0 \le i \le m-1\}| = 2m$. Clearly we must have $q \ge 2m$. Some examples found by a greedy algorithm are listed in Table 1. For each length, we give the sequence with smallest modulus found.

For any prime p and integer a, let $v_p(a)$ be the exact power of p dividing a (this is known as the valuation). A general construction for $l = 2$ is then: given q, consider the sequence of integers c such that $1 \le c \le q/2$ and $v_2(c)$ is even (that is, $c = 4^\alpha \gamma$ where γ is odd). This gives the elements less than or equal to $q/2$ in the greedy construction above. For example, for $q = 35$, we get the following sequence:

$$1, 3, 4, 5, 7, 9, 11, 12, 13, 15, 16, 17.$$

The greedy algorithm gives

$$1, 3, 4, 5, 7, 9, 11, 12, 13, 15, 16, 17, 27, 28, 29, 33.$$

The length of the sequence is

$$m = \left\lfloor \frac{q}{4} \right\rfloor + \left\lfloor \frac{q}{4^2} \right\rfloor + \left\lfloor \frac{q}{4^3} \right\rfloor + \cdots.$$

We see that

$$m < \frac{q}{4} + \frac{q}{4^2} + \frac{q}{4^3} + \cdots = \frac{q}{3}.$$

On the other hand, if $4^k \le v < 4^{k+1}$, then

$$m \ge \frac{q}{4} + \frac{q}{4^2} + \cdots + \frac{q}{4^k} - k = \frac{q}{3}\left(1 - \frac{1}{4^{k-1}}\right) - k.$$

For example, for $q = 100$, the bounds are $28.25 < m < 33.3$, that is $29 \le m \le 33$. Direct computation shows that $m = 32$.

E. Examples of $B_1([0,3])$ found by a greedy algorithm

Some examples of $B_1([0,3])$ are given:

q	m	$[c_i]$
3	1	$[1]$
7	2	$[1, 6]$
8	3	$[1, 4, 7]$
15	5	$[1, 4, 5, 7, 13]$
20	6	$[1, 4, 5, 9, 13, 17]$
26	7	$[1, 4, 5, 7, 13, 16, 23]$
28	9	$[1, 4, 5, 7, 9, 13, 17, 24, 25]$
34	10	$[1, 4, 5, 7, 9, 11, 17, 20, 29, 31]$
40	13	$[1, 4, 5, 7, 9, 11, 13, 19, 20, 23, 32, 35, 37]$
50	14	$[1, 4, 5, 7, 9, 11, 13, 16, 23, 25, 28, 40, 43, 47]$

Similar to the construction in II-D, a general construction for $l = 3$ is: consider the sequence of integers $c \le q/3$ such that both $v_2(c)$ and $v_3(c)$ are even, that is, $c = 4^\alpha 9^\beta \gamma$ where $\gcd(\gamma, 6) = 1$. The length of the sequence will be approximately $q/6$. For example, for $q = 50$ we get the sequence

$$1, 4, 5, 7, 9, 11, 13, 16.$$

III. CODE CONSTRUCTION

Using the modular $B_1[0,l]$ sequence $B = (b_0, b_1, \ldots, b_{m-1})$, we can construct the following linear single l-AEC code. Let H be the $r \times n$ parity check matrix whose columns are all possible vectors in Z_q^r whose first non-zero element belongs to B. Let C be the null space of H^T. Then C has the following properties:

Theorem 4: C can correct a single asymmetric error of limited magnitude l.

Proof: Let $x \in C$ be the sent codeword and x' be the received word such that $x' = x + e$ where $e = (e_1, e_2, \ldots, e_n)$, $\exists i$, $1 \le i \le n$, such that $0 < e_i \le l$ and $\forall j \ne i$, $e_j = 0$. Then, we have

$$\begin{aligned} x'H^T &= xH^T + eH^T \\ &= eH^T = e_i h_i^T \end{aligned}$$

where h_i^T is the transpose of the i^{th} column of H. Let z be the first non-zero element in h_i. Then, by construction, we know that

$$z \equiv e_i b \,(mod\, q)$$

where $b \in B$. By properties of B, z is unique and b can be identified. Hence, the error magnitude e_i and its position i can be determined and x can be recovered from x'. ■

Theorem 5: The length of the code, n, is $m\frac{q^r-1}{q-1}$.

Proof: By construction n is the number of all possible vectors over Z_q^r whose first element belongs to B and thus

$$n = \sum_{j=1}^{r} |B| q^{r-j} = m \frac{q^r - 1}{q - 1}.$$

■

q	m	$[c_i]$	$2c_i \pmod q$
2	1	$[1]$	$[0]$
5	2	$[1,4]$	$[2,3]$
6	3	$[1,3,5]$	$[2,0,4]$
9	4	$[1,3,4,7][2,6,8,10,7]$	$[2,6,8,5]$
11	5	$[1,3,4,5,9]$	$[2,6,8,10,7]$
14	6	$[1,3,4,5,7,13]$	$[2,6,8,10,0,12]$
15	7	$[1,3,4,5,7,12,13]$	$[2,6,8,10,14,9,11]$
18	8	$[1,3,4,5,7,9,15,17]$	$[2,6,8,10,14,0,12,16]$
21	9	$[1,3,4,5,7,9,16,17,20]$	$[2,6,8,10,14,18,11,13,19]$
22	10	$[1,3,4,5,7,9,11,17,19,21]$	$[2,6,8,10,14,18,0,12,16,20]$
25	11	$[1,3,4,5,7,9,11,12,19,20,21]$	$[2,6,8,10,14,18,22,24,13,15,17]$
29	12	$[1,3,4,5,7,9,11,12,13,23,25,28]$	$[2,6,8,10,14,18,22,24,26,17,21,27]$
30	13	$[1,3,4,5,7,9,11,12,13,15,23,25,29]$	$[2,6,8,10,14,18,22,24,26,0,16,20,28]$

Table I
$B_1([0,2])$ FOUND BY A GREEDY ALGORITHM

Theorem 6: The $B_1([0,1])$ sequence $1, 2, \dots, q-1$ can be used to construct an optimal single 1-AEC code. Moreover, the $B_1([0,2])$ and $B_1([0,3])$ sequence given in Section II-C yields an optimal code for $l=2$ and $l=3$ respectively.

Proof: The bound on the size of any single l-AEC code, C', is given in Theorem 8 in [3]:

$$|C'| \sum_{i=0}^{1} \binom{n}{i} l^i \leq q^n.$$

Hence, when $l = 1$

$$|C'|(1+n) \leq q^n.$$

Furthermore, the sequence $1, 2, \dots, q-1$ has length $q-1$. Therefore, by Theorem 5 the length of the designed code C is

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

and thus

$$\begin{aligned} |C|(1+n) &= |C|(1+q^r-1) \\ &= |C|(q^r) = q^{n-r}q^r = q^n. \end{aligned}$$

Therefore the code is optimal. Similarly it can easily be seen that the code constructions for $l=2$ and $l=3$ with the B sequence defined in Section II-C are also optimal by observing that the length of the sequences are $q/2$ and $q/3$ respectively and that single 2-AEC and 3-AEC codes C' and C'' are such that

$$|C'|(1+2n) \leq q^n,$$

and

$$|C''|(1+3n) \leq q^n.$$

■

Now, we compare our codes with the ones given in [3]. For $l=1$, it shown in Theorem 6 that our construction is optimal. On the other hand, the construction given in [3] is not optimal in general. For example, starting with a $(7,4)$ Hamming code, the given construction over $q=4$ has length at most 6 when 2 check digits are used whereas the optimal length is $q^2-1 = 15$. For $l > 1$, according to Theorem 5 we have that

$$n \geq m\frac{q^r-1}{q-1} \geq \frac{q^r-1}{q-1}.$$

In [3], the code given has length n' where

$$n' \leq \frac{(l+1)^{rs}-1}{l}$$

such that

$$s = \log_{l+1} \frac{q}{2}.$$

Hence

$$n' \leq \frac{(\frac{q}{2})^r - 1}{l}.$$

Therefore, for large enough q we have $n \geq n'$, and thus, using the same number of check digits, our construction can be used to encode more information digits than the construction given in [3].

IV. SYMMETRIC LIMITED MAGNITUDE ERROR CORRECTION

In this section, we give codes correcting a single symmetric error of maximum magnitude l (l-SEC). In this error model, a symbol $a \in Z_q$ can be changed into $a \pm e$ where $0 \leq e \leq l$.

For $l = 1$, single 1-SEC codes are equivalent to single symmetric error correcting codes in the Lee metric and for these optimal codes are known [2]: Let H be the parity check matrix whose columns are all vectors in Z_q^r whose first non-zero elements are in $\{1, 2, \dots, \frac{q-1}{2}\}$ (for odd q). Then the corresponding code can correct any symmetric error of limited magnitude 1. Moreover, in the general symmetric case where $l = q-1$, Hamming codes can be used in order to correct a single error. However, when $2 \leq l \leq q-2$, using a $B_1([-l,l])$ sequence, we can apply the same construction given in the previous section in order to design single l-SEC codes achieving higher rate than the Hamming code. We want a set $c_0, c_1, \dots, c_{m-1}$ and a modulus q such that all $c_i x_i \pmod q$ for $-l \leq x_i \leq l$ are distinct. Let p be a prime, $p \geq m$ and $p \geq 2l+1$. Let $q = p(2l+1)$ and let $c_i = i(2l+1)+1$ for $0 \leq i \leq m-1$. Similar to the proof for the $B_1([0,l])$ sequence in Section II-B, we show that this is indeed a modular $B_1([-l,l])$. Suppose that $c_ix_i \equiv c_jx_j \pmod{q}$, where $x_i \neq 0$, that is

$$(i(2l+1)+1)x_i \equiv (j(2l+1)+1)x_j \pmod{p(2l+1)}.$$

In particular, this implies that

$$x_i \equiv x_j \pmod{2l+1}.$$

But since $x_i, x_j \in \{-l, -l+1, \ldots, l-1, l\}$ and $x_i \neq 0$, this implies that $x_j = x_i \neq 0$. Hence

$$i(2l+1)x_i \equiv j(2l+1)x_i \pmod{p(2l+1)}$$

and so

$$ix_i \equiv jx_i \pmod{p}.$$

Since $x_i \neq 0$, we have $\gcd(x_i, p) = 1$ and so $i \equiv j \pmod{p}$. Finally, since $p \geq m$, then $i = j$.

V. CONCLUSION

Errors in Multi-level Flash memories are asymmetric in nature. Moreover, cell values are most likely to change, in case of error, to neighboring values. Thus the most suitable error model for this application is one where symbols can only change in one direction with a limited error magnitude. We proposed error correcting codes that can correct single asymmetric error of maximum magnitude l. We also showed how the code construction can be slightly modified to design code correcting symmetric errors of limited magnitude. Both codes are Hamming-like codes that depend on special sequences which we defined as the B sequences. Not only that those sequences can be used in the construction of single l-AEC and l-SEC codes, but also they can be used to construct codes correcting higher number of errors. This will be further explored in future work.

REFERENCES

[1] R. Ahlswede, H. Aydinian, L. H. Khachatrian, and L. M. G. M. Tolhuizen, "On q-ary codes correcting all unidirectional errors of a limited magnitude", Proceedings of Ninth International Workshop on Algebraic and Combinatorial Coding Theory, Kranevo, Bulgaria, pp. 20-26, Jun. 2004.

[2] E. R. Berlekamp, Algebraic Coding Theory, McGraw-Hill Book, 1968.

[3] Y. Cassuto, M. Schwartz, V. Bohossian, and J. Bruck, "Codes for Asymmetric Limited-Magnitude Errors with Application to Multi-Level Flash Memories", California Institute of Technology, Pasadena (CA), Tech. Rep. CaltechPARADISE:2008.ETR088, 2008

[4] C. J. Colbourn and J. H. Dinitz eds., Handbook of Combinatorial Designs, 2ed. Chapman and Hall.

[5] N. Elarief and B. Bose, "Optimal systematic q-ary codes correcting all asymmetric errors of limited magnitude", Proceedings of the IEEE International Symposium on Information Theory, Seoul, Korea, June 2009.