

1ビット反転によるデータ書換えを用いた Time-Space 制約符号の構成

武山 昌平[†] 鎌部 浩[†][†] 岐阜大学大学院 工学研究科 応用情報学専攻

あらまし 相変化メモリは、記録素子(セル)内部の化合物を結晶状態もしくは非結晶状態に変化させることによってデータを記録する不揮発性のメモリである。セルの状態は熱によって変化させる。したがってデータの書き換えが短時間に狭い領域で集中して生じるとその部分に熱が蓄積し書き換えの操作に時間がかかるなどの影響が生じる。A.Jiangらは熱の発生が集中しないようにデータを符号化するために Time-Space 制約を定義したが、M.Qinらは Time-Space 制約を満す符号の具体的な符号の構成方法を与えた。本稿では、いくつかのパラメータに対して、より符号化率の高い符号の構成方法を与える。

キーワード 相変化メモリ, 熱蓄積, Time-Space 制約符号, 符号化率

Shohei TAKEYAMA[†] and Hiroshi KAMABE[†][†] Information Science Division, Graduate School of Engineering, Gifu University

Abstract Phase-change memory(PCM) is a promising non-volatile solid-state memory. The cell changes between these two states using high temperature. So it is important, when programming cells, to balance the heat both in time and in space. The time-space constraint was introduced by A. Jiang et al. to balance the heat. M. Qin et al. proposed some constructions for codes satisfying the constraint. In this manuscript some new constraint codes for the constraint are proposed. It is shown that the code rates of the proposed code are higher than that of codes by M. Qin et al.

Key words Phase Change Memories, Thermal Accumulation, Time-Space Constrained Codes, Code Rate

1. ま え が き

モバイルデバイスの普及につれて、電源を切ってもデータが消えない不揮発性半導体メモリの普及が拡大している。特にフラッシュメモリは、SSDやメモリカードなどの記録媒体で広く用いられている。

フラッシュメモリよりも書き込み速度が高速で書換え寿命が長い相変化メモリの開発が、1960年代から進められてきた。相変化メモリは、記録素子(セル)内部の化合物を結晶状態もしくは非結晶状態に変化させることによってデータを記録する不揮発性のメモリである。状態を変化させるには熱を利用する。結晶状態から非結晶状態へ変化させるには、600℃を超える熱を約80ns間加える[3]。この操作をRESETと呼ぶ。また非結晶状態から結晶状態へ変化させるには、300~600℃の熱を約10ns間加える[3]。この操作をSETと呼ぶ。RESETとSETの操作は、プログラミングと呼ばれている[1]。

プログラミングを行う場合、セル内部は高温になり隣接するセルに熱が拡散する。したがってプログラミングが短時間に狭い領域で集中して生じると、その部分に熱が蓄積する。この現象は熱蓄積と呼ばれている。熱蓄積が生じた場合には、蓄積さ

れた熱が発散し次の書き込みを行えるようになるまでの待機時間は、熱蓄積が生じていない場合に比べてより長くなる[2]。

熱蓄積を避けるために、熱の発生(データの書き換え)を時間的、空間的になるべく均一化することを目的とした Time-Space 制約が提案されている[1]。この制約は三つのパラメータ α, β, p を持っている。本稿ではこれらのパラメータを明示するときには Time-Space 制約を (α, β, p) 制約と書く。

次に本稿の構成を示す。2節では本稿で使う記号の意味を記す。3節では (α, β, p) 制約の意味を記す。4節では M.Qin らが構成した (α, β, p) 制約符号の構成方法を示す。5節ではデータを書換える際、常に1ビットで書換えが可能な符号を定義する。この符号は6節以降で符号を提案する際に必要となる。6節以降では (α, β, α) 制約符号, $(3, \beta, 5)$ 制約符号, $(3, 4, 3)$ 制約符号を構成し、どの符号も M.Qin らの符号よりも符号化率が高いことを示す。

2. 準 備

本稿では以下の記号を使用する。

- (1) i は書換え回数を表すとす。
- (2) ベクトル x_p^q はベクトル x の p 番目の要素から q 番目

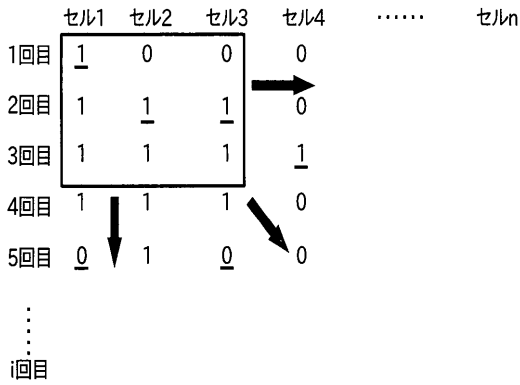


図1 (3,3,3) 制約符号による符号化

の要素までを要素にもつベクトル $(x_p, x_{p+1}, \dots, x_q)$ とする。

(3) 二つの二元ベクトル x, y に対して $x \oplus y$ は座標毎の排他的論理和を表す。

(4) 記号 \leftarrow は右辺から左辺への代入を表す。

(5) F はデータを記録しない状態を表す。

(6) 整数 v, w に対して, $v \bmod w$ は v を w で割った余りを表す。

3. (α, β, p) 制約

n 個のセルの現在の状態を表すベクトルを $u = (u_1, \dots, u_n) \in \{0, 1\}^n$, 1 回書換えた後のセルの状態を表すベクトルを $v = (v_1, \dots, v_n) \in \{0, 1\}^n$ とする。 u と v のハミング距離

$$d_H(u, v) = |\{i \in \{1, \dots, n\} | u_i \neq v_i\}|$$

を書換えコストと呼ぶ。 [1]. つまり書換えコストとは, 1 回の書換えでプログラミングが生じたセルの個数である。

正整数 α, β, p に対して (α, β, p) 制約を以下で定義する。 $n (\geq \beta)$ 個のセルで $i (\geq \alpha)$ 回の書換えを符号 C によって行うと仮定する。 任意の連続する β 個のセルにおける任意の連続する α 回書換えに着目したとき, 書換えコストの総和が高々 p であれば符号 C は (α, β, p) 制約を満たすと言う。 (α, β, p) 制約を満たす符号を (α, β, p) 制約符号と呼ぶ [1].

$(3, 3, 3)$ 制約を満たす符号 C を用いて符号化したときの様子を図 1 で示す。 図 1 において状態遷移が生じた後の状態に下線を付けられている。 符号 C が $(3, 3, 3)$ 制約を満たすとき, 枠がどの位置に移動しても枠内の下線付きの状態は高々 3 個しか存在しない。

4. M.Qin らの (α, β, p) 制約符号の構成

M.Qin らが構成した (α, β, p) 制約符号 ([2] の構成 1) を符号 C' とする。 符号の C' の構成方法を以下に示す。

n 個のセルを β 個のセルから成るブロックに分ける。 n は β の倍数とする。 また n 個のセルの右端に α 個のセルを置く。 このセルをカウンターセルと呼ぶ。 カウンターセルの状態, 書換え回数を数える。 ここで

$$p = \beta(q - 1) + r \tag{1}$$

を満たす $q, r (1 \leq q \leq \alpha, 1 \leq r \leq \beta)$ を定める。 例えば $\alpha = 5, \beta = 5, p = 12$ のとき, 式 (1) を満たすように $q = 3, r = 2$ と定めることができる。

各ブロックにおける符号化規則を以下で定義する。

- $1 \leq i \bmod \alpha < q$ ならば, β 個のセルに長さ β の 2 進系列をデータとして記録する。

- $i \bmod \alpha = q$ ならば, 各ブロックの先頭から r 個のセルに長さ r の 2 進系列をデータとして記録する。 また残りのセルにはデータを記録しない。

- $i \bmod \alpha > q$ ならば, どのセルにもデータを記録しない。 ここでカウンターセルに含まれるセル 1 からセル α の現在の状態を表すベクトルを $l = (l_1, l_2, \dots, l_\alpha) \in \{0, 1\}^\alpha$, 1 回書換えた後の状態を表すベクトルを $l' = (l'_1, l'_2, \dots, l'_\alpha) \in \{0, 1\}^\alpha$ とする。 また $sl = (sl_1, \dots, sl_\alpha)$ を以下で定める。

- $i \bmod \alpha \neq 0$ のとき $sl_{(i \bmod \alpha)} = 1$, それ以外の要素は 0 とする。

- $i \bmod \alpha = 0$ のとき $sl_1 = sl_2 = \dots = sl_{(\alpha-1)} = 0, sl_\alpha = 1$ とする。

カウンターセルにおける状態の書換えを $l' = l \oplus sl$ と定義する。 ここで

$$i' = \sum_{k=1}^{\alpha} |l_k - l'_k| \tag{2}$$

とする。 i' の値が以下の条件を満たす場合, 各ブロックにおいて以下の方法でデータを復号する。

- $1 \leq i' < q$ のとき, β 個のセルに記録されている β ビットのデータを復号する。

- $i' = q$ のとき, ブロックの先頭から r 個のセルに記録されている r ビットのデータを復号する。

- $i' > q$ のとき, どのセルにもデータはないため F を復号する。

[例 1] $\alpha = 3, \beta = 3, p = 5$ のとき符号 C' を用いて $(3, 3, 5)$ 制約符号を図 2 で構成する。 式 (1) より $5 = 3(q - 1) + r$ である。 $1 \leq q \leq 3, 1 \leq r \leq 3$ を満たすように $q = 2, r = 2$ と定める。 3 個のセルから成るブロックを 4 ブロックと右端に 3 個のカウンターセルを配置する。

- $1 \leq i \bmod 3 < 2$ のとき, 各ブロックに 3 ビットの 2 進系列を書き込む。

- $i \bmod 3 = 2$ のとき, 各ブロックの先頭から 2 個のセルに 2 ビットの 2 進系列を書き込む。

- $i \bmod 3 > 2$ ならば, どのセルにもデータを記録しない。 復号は以下で行う。

- $(l_1, l_2, l_3) = (1, 0, 0), (0, 1, 1)$ のとき式 (2) より $i' = 1$ なので, 3 ビットの 2 進系列を復号する。

- $(l_1, l_2, l_3) = (1, 1, 0), (0, 0, 1)$ のとき式 (2) より $i' = 2$ なので, ブロックの先頭から 2 個のセルから 2 ビットの 2 進系列を復号する。

- $(l_1, l_2, l_3) = (1, 1, 1), (0, 0, 0)$ ならば式 (2) より $i' = 3$ なので, データはない。 よって F を復号する。

	000	000	000	000	000
1回目	<u>101</u>	<u>011</u>	<u>111</u>	<u>100</u>	100
2回目	<u>001</u>	<u>101</u>	<u>011</u>	<u>000</u>	110
3回目	001	101	011	000	111
4回目	<u>100</u>	<u>000</u>	<u>010</u>	<u>011</u>	011
5回目	<u>010</u>	<u>100</u>	<u>110</u>	<u>101</u>	001
6回目	010	100	110	101	000

図2 (3,3,5) 制約符号による符号化

符号 C' の符号化率を計算する. α 回の書換えで使うセルの数は $\alpha(n + \alpha) = n\alpha + \alpha^2$ であり, 1ブロックにおいて書き込むことができるデータは p ビットである. よって $\frac{n}{\beta}$ ブロックに書き込むことができるデータは $\frac{pn}{\beta}$ ビットである. したがって符号化率は次式で与えられる.

$$\lim_{n \rightarrow \infty} \frac{\frac{pn}{\beta}}{n\alpha + \alpha^2} = \lim_{n \rightarrow \infty} \frac{\frac{p}{\beta}}{\alpha + \frac{\alpha^2}{n}} = \frac{p}{\alpha\beta}$$

5. 符号 C_W の定義

四元 $\{0,1,2,3\}$ のデータを書換えるとき, 常に1ビットで書換えが可能な符号 C_W を定義する. $w, w' \in \{0,1,2,3\}$ に対して w を現在記録されているデータ, w' を書き込むデータとし, w に対応する符号語を $c(w)$ とする. 表1には, w' に対する符号語が二つ与えられているが, $d_H(c(w), c(w')) = 1$ となるほうの符号語を選ぶことで, 符号化写像 $\mathcal{E}_{C_W} : \{0,1,2,3\} \times \{0,1\}^3 \rightarrow \{0,1\}^3$ を決める. 復号写像 $\mathcal{D}_{C_W} : \{0,1\}^3 \rightarrow \{0,1,2,3\}$ は表1の対応表にしたがって, 符号語 $c(w)$ をデータ w へ復号する.

データ w	符号語 $c(w)$
0	000,111
1	001,110
2	010,101
3	100,011

表1 符号 C_W の対応表

[例2] データを $0 \rightarrow 2 \rightarrow 1 \rightarrow 3$ の順で書き込むとする. 書換え前と後の符号語間のハミング距離が1になるように書換えを行うため, セルの状態は $000 \rightarrow 010 \rightarrow 110 \rightarrow 100$ と変化する. 現在の状態が100の場合には, 表1に従って3に復号する.

6. (α, β, α) 制約符号の構成

6.1 符号の構成

$\beta \geq 3$ の場合に, (α, β, α) 制約符号を構成する. 構成する

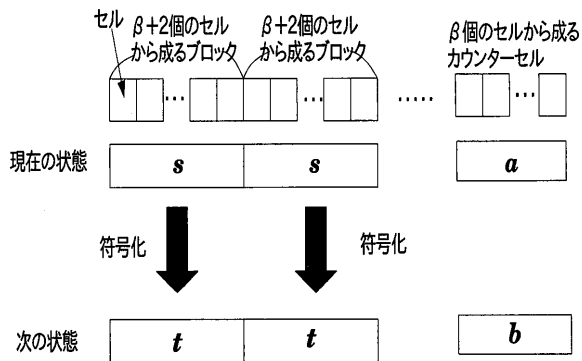


図3 (α, β, α) 制約符号による符号化の様子

(α, β, α) 制約符号を符号 C_1 とする. n をセルの個数とする. n は $\beta+2$ の倍数とする. また n 個のセルの右端に β 個のカウンター用のセルを置く. n 個のセルを $\frac{n}{\beta+2}$ 個のブロックに分ける. 各ブロックに含まれる $\beta+2$ 個のセルの書換え前の状態を表すベクトルを s , 書換え後の状態を表すベクトルを t とする. カウンター用のセルの書換え前の状態を a , 書換え後の状態を b とする.

各ブロックにおいて符号 C_1 による符号化を行う. 符号化写像 \mathcal{E}_{C_1} を以下で定義する. $k = 1, 2, \dots, \beta-2, \beta-1$ に対して

- $i \bmod \beta = k$ の場合
 - $\mathcal{E}_{C_1}(m, s_k^{k+2}) = t_k^{k+2} = \mathcal{E}_{C_W}(m, s_k^{k+2})$
 - $b = a \oplus xs$
- $i \bmod \beta = 0$ の場合
 - $\mathcal{E}_{C_1}(m, s_\beta^{\beta+2}) = t_\beta^{\beta+2} = \mathcal{E}_{C_W}(m, s_\beta^{\beta+2})$
 - $b = a \oplus xs$

と定義する. ただし $m \in \{0,1,2,3\}$ をデータとする. また $xs = (xs_1, \dots, xs_\beta)$ は以下で定める.

- $i \bmod \beta \neq 0$ のとき
 - $xs_{(i \bmod \beta)} = 1$, それ以外の要素は0をとる.
- $i \bmod \beta = 0$ のとき
 - $xs_1 = xs_2 = \dots = xs_{\beta-1} = 0, xs_\beta = 1$ とする.

$a = (a_1, \dots, a_\beta)$ に対して $ct = \sum_{h=1}^{\beta} |a_h - a_\beta|$ とする. 各ブロックに対する復号写像 \mathcal{D}_{C_1} は

- $ct = k$ の場合
 - $\mathcal{D}_{C_1}(s_k^{k+2}) = \mathcal{D}_{C_W}(s_k^{k+2})$
- $ct = 0$ の場合
 - $\mathcal{D}_{C_1}(s_\beta^{\beta+2}) = \mathcal{D}_{C_W}(s_\beta^{\beta+2})$

と表せる.

図4で (α, β, α) 制約符号による符号化を示した. ブロックにおいて枠で囲ったセルが符号 C_W を用いて符号化したセルである. プログラミングが行われた後の状態に下線を付けた.

[例3] $\beta = 3$ のとき2個のブロックを置き, $(\alpha, 3, \alpha)$ 制約符号を構成する.

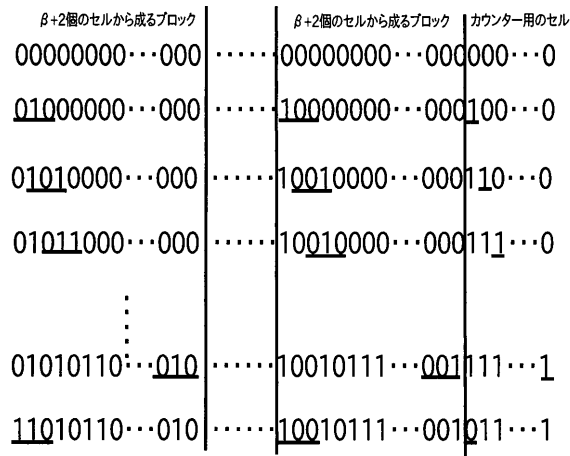


図4 (α, β, α) 制約符号

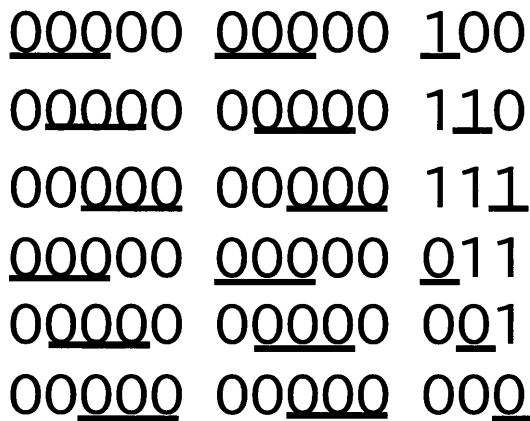


図5 $(\alpha, 3, \alpha)$ 制約符号

- $i \bmod 3 = 1$ の場合
 - $\mathcal{E}_{C_1}(m, s_1^3) = t_1^3 = \mathcal{E}_{C_W}(m, s_1^3)$
 - $\mathbf{b} = \mathbf{a} \oplus (1, 0, 0)$
- $i \bmod 3 = 2$ の場合
 - $\mathcal{E}_{C_1}(m, s_2^4) = t_2^4 = \mathcal{E}_{C_W}(m, s_2^4)$
 - $\mathbf{b} = \mathbf{a} \oplus (0, 1, 0)$
- $i \bmod 3 = 0$ の場合
 - $\mathcal{E}_{C_1}(m, s_3^5) = t_3^5 = \mathcal{E}_{C_W}(m, s_3^5)$
 - $\mathbf{b} = \mathbf{a} \oplus (0, 0, 1)$

図5で $(\alpha, 3, \alpha)$ 制約符号による符号化の様子を示す。

6.2 (α, β, α) 制約の確認

構成した符号が (α, β, α) 制約を満たすことを確認する。図6, 7において、符号 C_W によって符号化されたセルとカウンター用のセルでプログラミングが行われたセルの状態を下線で示し、連続する β 個のセルの状態を枠で囲った。この符号は枠を任意の場所に置いたとき、枠の中で下線が1本だけ含まれる構成となっている。したがって任意の連続する β 個のセルにお

いて、高々1回プログラミングが行われている。よって $(1, \beta, 1)$ 制約を満たしている。

$(1, \beta, 1)$ 制約を満たすため、枠を書換え方向に α 下に移動させると、どの枠内でも下線が1本だけ含まれる。よって任意の連続する β 個のセルにおける任意の連続する α 回の書換えでは、高々 α 回プログラミングが行われることになる。つまりこの符号は (α, β, α) 制約を満たす。

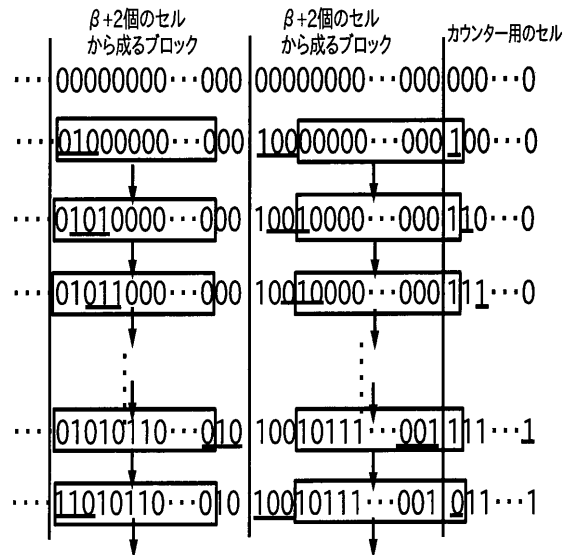


図6 (α, β, α) 制約符号

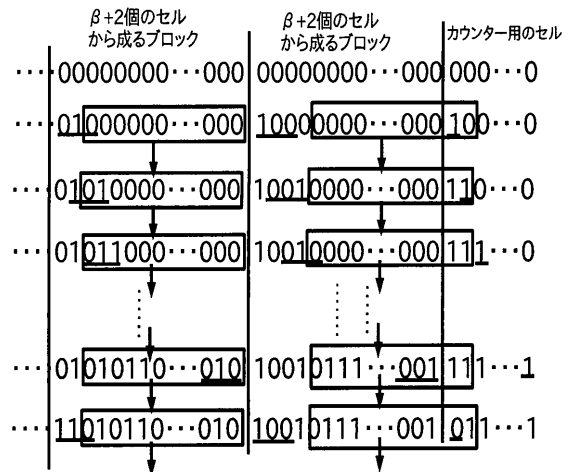


図7 (α, β, α) 制約符号

6.3 符号化率の評価

(α, β, α) 制約符号の符号化率を計算する。1回の書換えで使うセルの数は $n + \beta$ である。また1ブロックに4通りのデータを記録できるので、 $\frac{n}{\beta+2}$ ブロックで記録できるデータは $\log_2 4^{\frac{n}{\beta+2}} = \frac{2n}{\beta+2}$ (ビット) である。よって符号化率 R は $n \rightarrow \infty$

とすると

$$\lim_{n \rightarrow \infty} R = \lim_{n \rightarrow \infty} \frac{2n}{\beta+2} = \lim_{n \rightarrow \infty} \frac{2}{1 + \frac{\beta}{n}} = \frac{2}{\beta+2}$$

である。一方、M.Qin らの符号の符号化率は $p = \alpha$ の場合

$$\frac{p}{\alpha\beta} = \frac{\alpha}{\alpha\beta} = \frac{1}{\beta}$$

である。提案する (α, β, α) 制約符号の符号化率と M.Qin らが構成した (α, β, α) 制約符号の $p = \alpha$ の場合の符号化率の比較表を表 2 で示した。 β が表中のどの値でも、提案する符号の符号化率の方が高いことが確認できた。

β の値	3	4	5	6	7	8
提案する符号	0.4	0.333...	0.285...	0.25	.222...	0.2
M.Qin らの符号	0.333...	0.25	0.2	0.166...	0.142...	0.125

7. $(3, \beta, 5)$ 制約符号

7.1 符号の構成

$\beta \geq 4$ の場合の $(3, \beta, 5)$ 制約符号を構成する。構成する符号を符号 C_2 とおく。 n 個のセルを $\beta+2$ 個のセルから成るブロックに分け、その右端に 3 個のカウンタセルと $\beta-3$ 個の検出用のセルを置く。 n を $\beta+2$ の倍数とする。各ブロックに含まれる $\beta+2$ 個のセルの書換え前の状態を表すベクトルを s' 、書換え後の状態を表すベクトルを t' とする。カウンタ用のセルの書換え前の状態を a' 、書換え後の状態を b' とする。検出用のセルの書換え前の状態を表すベクトルを c 、書換え後の状態を表すベクトルを d とする。セルの状態とそれに対応するベクトルを図 8 で示す。

ここで $\lfloor \frac{i}{3} \rfloor \bmod (\beta-3)$ を ik とする。

また $\mathbf{x}l = (xl_1, \dots, xl_{\beta-3})$ を $ik+1$ 番目の要素が 1、他の要素はすべての 0 となるベクトルとする。

符号化写像 \mathcal{E}_{C_2} は以下で定義する。

- $i \bmod 3 = 1$ のとき
 - $\mathcal{E}_{C_2}(s'_1, m) = t'_1 = \mathcal{E}_{C_W}(s'_1, m)$
 - $\mathcal{E}_{C_2}(s'_4, m) = t'_4 = \mathcal{E}_{C_W}(s'_4, m)$
 - $b' = a' \oplus (1, 0, 0)$
- $i \bmod 3 = 2$ のとき
 - $\mathcal{E}_{C_2}(s'_1, m) = t'_1 = \mathcal{E}_{C_W}(s'_1, m)$
 - $\mathcal{E}_{C_2}(s'_{4+ik}, m) = t'_{4+ik} = \mathcal{E}_{C_W}(s'_{4+ik}, m)$
 - $b' = a' \oplus (0, 1, 0)$
 - $d = c \oplus \mathbf{x}l$
- $i \bmod 3 = 0$ のとき
 - $\mathcal{E}_{C_2}(s'_1, m) = t'_1 = \mathcal{E}_{C_W}(s'_1, m)$
 - $b' = a' \oplus (0, 0, 1)$

符号化写像 \mathcal{E}_{C_2} による符号化の様子を図 9 で示す。符号化された状態を下線で示した。

[例 4] $\beta = 6$ のとき 2 個のブロックを配置する。 $(3, \beta, 5)$ 制約符号による符号化の様子を図 10 で示した。

復号写像 D_{C_2} を以下で定義する。ただし $a' = (a'_1, a'_2, a'_3)$ に

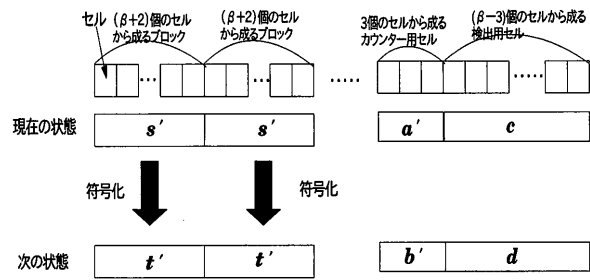


図 8 符号化の様子

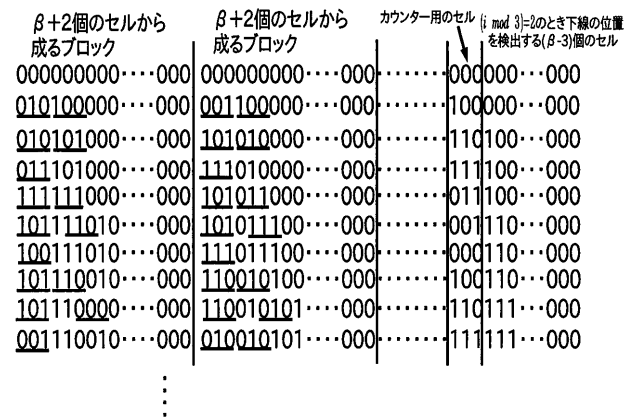


図 9 符号 C_2 の符号化の様子

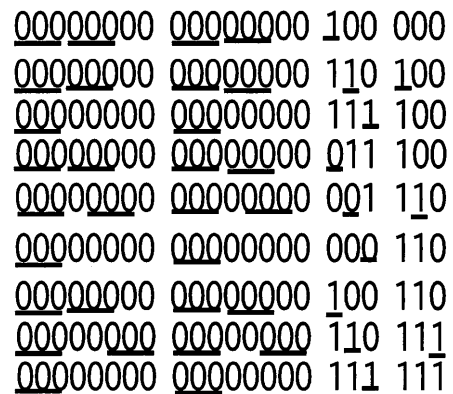


図 10 符号 C_2 の符号化の様子

対して $cn = |a'_1 - a'_3| + |a'_2 - a'_3|$, $c = (c_1, c_2, \dots, c_{\beta-3})$ に対して $cnt = \sum_{j=1}^{\beta-3} |c_j - c_{\beta-3}| - 1$ とする。ただし $cnt = -1$ のとき $cnt \leftarrow \beta - 4$ とする。

- $cn = 1$ のとき
 - $D_{C_2}(s'_1) = D_{C_W}(s'_1)$
 - $D_{C_2}(s'_4) = D_{C_W}(s'_4)$
- $cn = 2$ のとき
 - $D_{C_2}(s'_1) = D_{C_W}(s'_1)$
 - $D_{C_2}(s'_{4+cnt}) = D_{C_W}(s'_{4+cnt})$

- $cn = 0$ のとき

- $D_{C_2}(s_1^3) = D_{C_W}(s_1^3)$

[例5] $\beta = 10$ でブロックの数が1とする。現在の状態が 101111010000011100000 の場合、復号写像 D_{C_2} を用いてデータを復号する。ここで 101111010000 がブロック内のセルの状態、001 がカウンターセルの状態、1100000 が検出用セルの状態である。よって $cn = |0 - 1| + |0 - 1| = 2$, $cnt = |1 - 0| + |1 - 0| + |0 - 0| + |0 - 0| + |0 - 0| + |0 - 0| + |0 - 0| + |0 - 0| - 1 = 1$ となる。 $cn = 2$ なので

- $D_{C_2}(101) = D_{C_W}(101) = 2$
- $D_{C_2}(s_5^7) = D_{C_2}(110) = D_{C_W}(110) = 1$

と復号する。

7.2 (3, β , 5) 制約の確認

構成した符号が (3, β , 5) 制約を満たすことを確認する。図 11~14 の枠は連続する β 個のセルの状態を表す。ブロック内で下線が付いた状態は符号 C_W による符号化を行った後の状態を表す。符号 C_W による符号化の際には高々1ビットの反転でデータを書き込むことが可能なので、この下線が付いた3個のセルのうち高々1個がプログラミングされる。またカウンターセルと検出用セルにおいて下線が付いた状態はプログラミングが行われたことを表す。

$i \bmod 3 = 1$ のとき枠が移動する途中の例を図 11 で示した。枠がどの位置に移動しても枠に含まれる下線は高々2本しか含まれない。すなわち任意の連続する β 個のセルにおいて1回の書換えでは高々2回プログラミングが行われる。

$i \bmod 3 = 2$ のとき枠が移動する途中の例を図 12, 13 で示した。枠がどの位置に移動しても枠に含まれる下線は2本である。すなわち任意の連続する β 個のセルにおいて1回の書換えでは高々2回プログラミングが行われる。

$i \bmod 3 = 0$ のとき枠が移動する途中の例を図 14 で示した。枠がどの位置に移動しても枠に含まれる下線は1本である。すなわち任意の連続する β 個のセルにおいて1回の書換えでは高々1回プログラミングが行われる。

以上をまとめると任意の連続する β 個のセルにおける1回の書換えでは、 $i \bmod 3 = 1$ と $i \bmod 3 = 2$ のとき高々2回のプログラミングが生じ、 $i \bmod 3 = 0$ のとき高々1回のプログラミングが生じる。よって任意の連続する β 個のセルにおける任意の連続する3回の書換えでは、高々5回プログラミングが行われるため (3, β , 5) 制約を満たす。

7.3 符号化率の評価

(3, β , 5) 制約符号の符号化率を計算する。3回の書換えで使うセルの数は $3\{n + 3 + (\beta - 3)\} = 3(n + \beta)$ である。1ブロックにおいて3回書換える場合、データは 4^5 通り記録できる。 $\frac{n}{\beta+2}$ ブロックにデータを記録する場合 $\log_2(4^5)^{\frac{n}{\beta+2}} = \frac{10n}{\beta+2}$ (ビット) 記録できる。よって符号化率 R は $n \rightarrow \infty$ とすると

$$\lim_{n \rightarrow \infty} R = \lim_{n \rightarrow \infty} \frac{\frac{10n}{\beta+2}}{3(n + \beta)} = \lim_{n \rightarrow \infty} \frac{\frac{10}{\beta+2}}{3 + \frac{3\beta}{n}} = \frac{10}{3(\beta + 2)}$$

である。一方、M.Qinらの符号の符号化率は $\alpha = 3, p = 5$ の場合

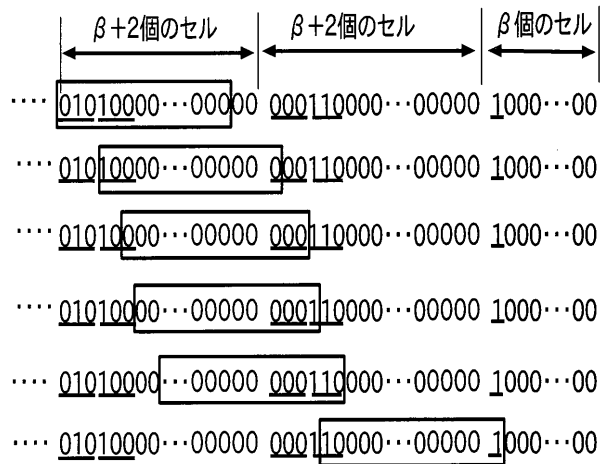
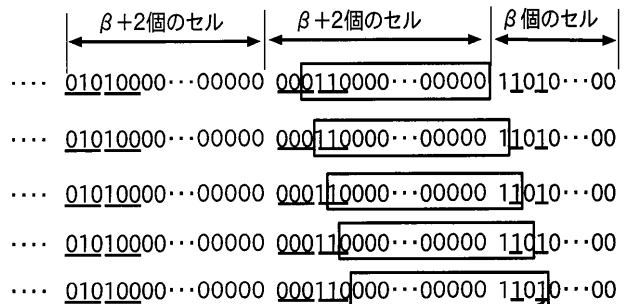


図 11 $i \bmod 3 = 1$ のとき



検出用セルでプログラミングされたセル

図 12 $i \bmod 3 = 2$ かつ $c = (1, 0, 0, \dots)$ のとき



検出用セルでプログラミングされたセル

図 13 $i \bmod 3 = 2$ かつ $c = (1, 1, 0, \dots)$ のとき

$$\frac{p}{\alpha\beta} = \frac{5}{3\beta}$$

である。提案する (3, β , 5) 制約符号の符号化率と M.Qinらが

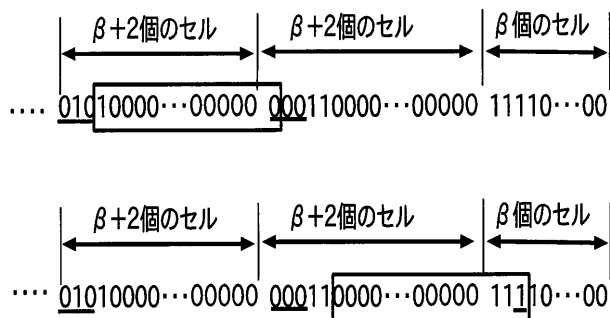


図 14 $i \bmod 3 = 0$ のとき

β の値	3	4	5	6	7	8
提案する符号	0.666...	0.555...	0.476...	0.416...	0.370...	0.333...
M.Qin らの符号	0.555...	0.416...	0.333...	0.277...	0.238...	0.208...

表 2 符号化率の比較表

構成した $(3, \beta, 5)$ 制約符号の $\alpha = 3, p = 5$ の場合の符号化率の比較表を表 2 で示した. β が表中のどの値でも, 提案する符号の符号化率の方が高いことが確認できた.

8. $(3, 4, 3)$ 制約符号

8.1 符号の構成

6 節で構成した (α, β, α) 制約符号の $\alpha = 3, \beta = 4, p = 3$ の場合とは別の方法で $(3, 4, 3)$ 制約符号を構成する. この符号を C_3 と呼ぶ. n をセルの個数とする. n は 6 の倍数とする. n 個のセルを $\frac{n}{6}$ 個のブロックに分ける. 各ブロックについて最初の 3 個のセルの現在の状態を表すベクトルを u , 1 回書換えた後の状態を表すベクトルを v , 次の 3 個のセルの現在の状態を表すベクトルを u' , 1 回書換えた後の状態を表すベクトルを v' とする.

n 個のセルの右端に 4 個のセルを置き, 右端 3 個のセルにおける現在の状態を表すベクトルを as , 1 回書換えた後の状態を表すベクトルを bs とする.

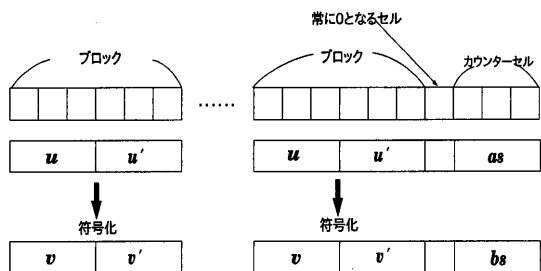


図 15 $(3, 4, 3)$ 制約符号

1 回目	<u>010</u>	<u>001</u>	<u>001</u>	<u>100</u>	0	<u>100</u>
2 回目	<u>000</u>	001	<u>011</u>	100	0	<u>110</u>
3 回目	000	001	011	100	0	<u>111</u>
4 回目	<u>010</u>	<u>011</u>	<u>001</u>	<u>110</u>	0	<u>011</u>
5 回目	<u>000</u>	011	<u>001</u>	110	0	<u>001</u>
6 回目	000	011	001	110	0	<u>000</u>

表 3 符号 C_3 による符号化

各ブロックに対して符号化写像 \mathcal{E}_{C_3} を以下で定義する.

- $(i \bmod 3) = 1$ の場合
 - $\mathcal{E}_{C_3}(m) = v = \mathcal{E}_{C_W}(m)$
 - $\mathcal{E}_{C_3}(m') = v' = \mathcal{E}_{C_W}(m')$
 - $bs = as \oplus (1, 0, 0)$
- $(i \bmod 3) = 2$ の場合
 - $\mathcal{E}_{C_3}(m) = v = \mathcal{E}_{C_W}(m)$
 - $\mathcal{E}_{C_3}(F) = v' = u'$
 - $bs = as \oplus (0, 1, 0)$
- $(i \bmod 3) = 0$ の場合
 - $\mathcal{E}_{C_3}(F) = v = u$
 - $\mathcal{E}_{C_3}(F) = v' = u'$
 - $bs = as \oplus (0, 0, 1)$

ただし $m, m' \in \{0, 1, 2, 3\}$ をデータとする.

ここで $as = (as_1, as_2, as_3)$ に対して $cs = |as_1 - as_3| + |as_2 - as_3|$ とする. 復号写像 \mathcal{D}_{C_3} は以下で表せる.

- $as = 1$ の場合
 - $\mathcal{D}_{C_3}(u) = \mathcal{D}_{C_W}(u)$
 - $\mathcal{D}_{C_3}(u') = \mathcal{D}_{C_W}(u')$
- $as = 2$ の場合
 - $\mathcal{D}_{C_3}(u) = \mathcal{D}_{C_W}(u)$
 - $\mathcal{D}_{C_3}(u') = F$
- $as = 0$ の場合
 - $\mathcal{D}_{C_3}(u) = F$
 - $\mathcal{D}_{C_3}(u') = F$

[例 6] 12 個のセルを 4 個のブロックに分けて, 符号 C_3 を用いて符号化を行う. データを $(2, 1, 1, 3) \rightarrow (0, F, 3, F) \rightarrow (F, F, F, F) \rightarrow (2, 3, 1, 1) \rightarrow (0, F, 1, F) \rightarrow (F, F, F, F)$ と与えると符号化の様子は表 3 のようになる. 下線で示した状態がデータが書き込まれている状態である.

8.2 符号化率の評価

符号化率を計算し, Qin らが構成した符号 ([2] の構成 1) の符号化率と比較する. 3 回の書換えで使うセルの数は $3(n+4)$ である. 1 ブロックで 3 回の書換えで記録できるデータは 4^3 通りである. $\frac{n}{6}$ ブロックにおいて 3 回の書換えで記録できるデータは

$$\log_2(4^3)^{\frac{n}{6}} = n(\text{ビット})$$

となる. よって符号化率 R は

$$R = \frac{n}{3n+12} = \frac{1}{3 + \frac{12}{n}}$$

なので, $n \rightarrow \infty$ とすると

$$R \rightarrow \frac{1}{3} = 0.333\dots$$

となる。一方, M.Qin らの符号の符号化率は

$$\frac{p}{\alpha\beta} = \frac{3}{3 \cdot 4} = 0.25$$

となる。よって提案した (3, 4, 3) 制約符号の方が, 符号化率が高くなる。

8.3 (3, 4, 3) 制約の確認

図 16 から図 18 で構成した符号が (3, 4, 3) 制約を満たすことを確認する。図のブロックの下線は符号 C_W による符号化が行われたセルである。カウンターセルにおける下線はプログラミングが行われたセルである。連続する 3 個のセルにおける連続する 3 回の書換えを枠で表している。図では枠が自由に移動する場合のいくつかの例を表している。

すべての枠に着目したとき, どの枠の中でも下線の本数は高々 3 本となっている。符号化が行われた下線では, 3 個のセルのうち高々 1 個がプログラミングされている。したがって連続する 3 個のセルにおける連続する 3 回の書換えでは, 高々 3 回プログラミングが行われている。

9. ま と め

本研究では, α, β, p の値を任意に変え (α, β, p) 制約符号を構成した。構成した符号は (α, β, α) 制約符号, $(3, \beta, 5)$ 制約符号, $(3, 4, 3)$ 制約符号である。 $(3, 4, 3)$ 制約符号は (α, β, α) 制約符号とは違う方法で構成した。そして同じ α, β, p の値に対して, これらの符号と M.Qin らが構成した (α, β, p) 制約符号について符号化率に関する比較を行った。その結果, 提案する符号の方が符号化率が高いことが分かった。

今後の課題は, $\alpha = 4, 5$ に対して β, p が一般の場合, 高い符号化率をもつ符号を構成することである。

謝辞

この研究は科研費(基盤研究(C) 課題番号 23560439)の助成を受けている。

文 献

- [1] A. Jiang, J. Bruck and H. Li, "Constrained codes for phase-change memories," in *Proc. IEEE Inf. Theory Workshop*, Dublin, Ireland, pp. 1-5, Aug. 30 - Sep. 3, 2010.
- [2] M. Qin, E. Yaakobi, P. H. Siegel, "Time-space constrained codes for phase-change memories," *IEEE Trans. Inf. Theory*, vol. 59, no. 8, pp. 5102-5114, Aug., 2013.
- [3] Y. B. Liao, J. T. Lin, M. H. Chiang, and W. C. Hsu, "Assessment of novel phase change memory programming techniques," *Electron Devices and Solid-State Circuits, 2008. EDSSC 2008. IEEE International Conference on*, pp. 1-4, Dec. 2008.

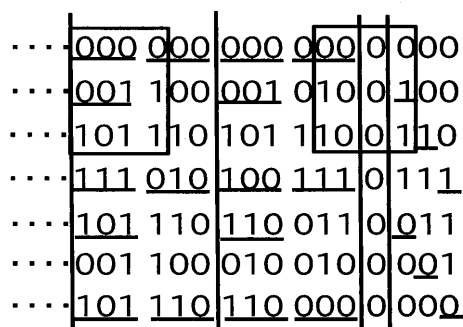


図 16 (3, 4, 3) 制約

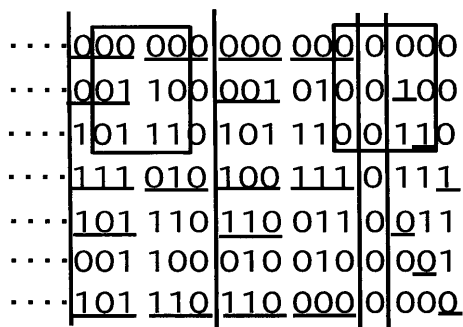


図 17 (3, 4, 3) 制約

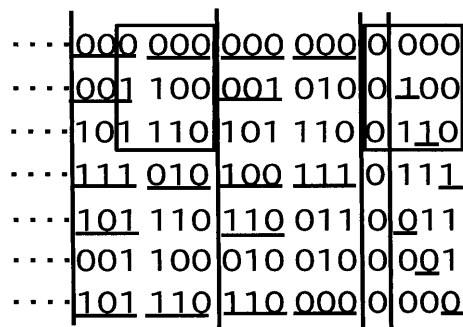


図 18 (3, 4, 3) 制約