不揮発性メモリを対象とした低書き込みメモリ暗号化手法

多和田 雅師[1,a)]　柳澤 政生[1]　戸川 望[1,b)]

概要：近年，不揮発性メモリの動作メモリとして活用が期待されている．動作メモリから情報を盗み出されることを防ぐためにめメモリ内に格納する情報を暗号化する需要がある．情報は暗号化されると乱雑性が高くなるため書き込み量が多くなり，結果として不揮発性メモリの耐久性が下がる．本稿では不揮発性メモリ暗号化と書き込み量削減を両立させる手法を提案する．データを格納する際にワードレベルで乱数文と排他的論理和をとり暗号化し，セルレベルで冗長化し並び替えにより情報を分散させる．メモリセルへの書き込みは，情報を分散させたことによりエンコーダが書き込み量が少なくなる書き込みを選択する．乱数文と排他的論理和をとった暗号文の暗号の強度を保ったまま書き込み量を削減できるシステムを構築する．情報を保存するためのメモリセルへの書き込みが最も低い書き込み量であることを証明する．

1. はじめに

近年あらゆるモノがインターネットにつながる IoT (Internet of Things) が注目されている．IoT デバイスのような組込みシステムでは供給電力に制約があるため，消費電力を削減できるノーマリオフ技術が研究されている．ノーマリオフ技術では電力供給がないときでもメモリ上に情報を保持できる不揮発性メモリが使われる．不揮発性メモリを使い常にメモリ上に情報を保持すると第三者に情報を盗み出される機会が多くなりセキュリティリスクが高くなる．情報の秘匿性を高めるためにメモリに書き込む情報を暗号化する手法が存在する [1], [2], [3], [4], [5]．メモリ暗号化手法では変更部分が拡散される性質があり，情報を表すビット列の一部を変更するときでもビット列の半数のビットに変更が生じる．不揮発性メモリ，特に PCM (Phase Change Memory) では書き込める回数の上限があり，暗号化メモリは PCM の見かけの書き込み耐性を下げてしまう問題がある．さらにマルチレベルセル不揮発性メモリでは 1 つのセルで複数ビットを表現するため，セルの変更確率が高くなる．暗号化メモリの書き込み量を削減する手法が存在する [1], [4], [5]．既存手法は書き込み量の削減量に対して最適性が保証されていない．メモリ符号化による書き込み削減技術 [6], [7], [8] によりセルレベルで書き込み量を削減する手法を提案する．書き込み量の削減量に対して提案手法の最適性を証明する．

[1] 早稲田大学
Waseda University
[a)] tawada@togawa.cs.waseda.ac.jp
[b)] togawa@togawa.cs.waseda.ac.jp

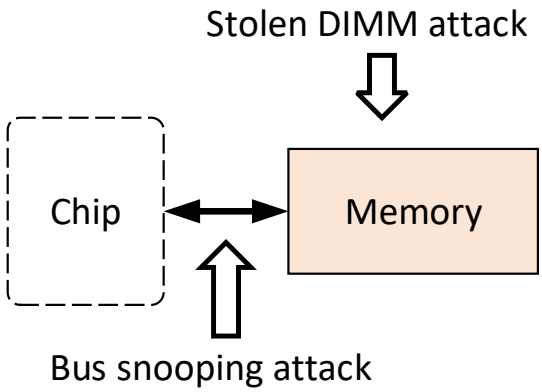

図 1　メモリ上の情報を盗み出す攻撃手法.

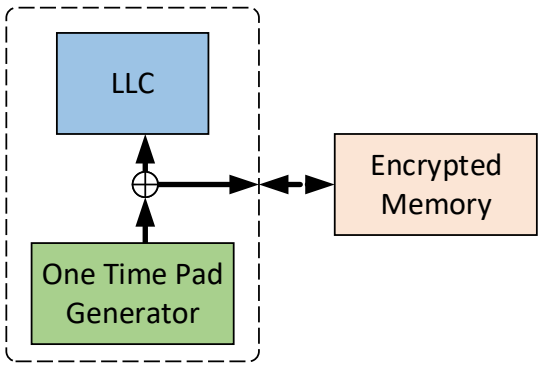

図 2　暗号化メモリの構造.

2. 暗号化メモリ

図 1 に示すように従来よりメモリ内部の情報を盗み出す攻撃やバス通信上の情報を盗み見る攻撃は存在していた．加えて不揮発性メモリでは長期間にわたり情報がメモリ上に保持し続けられるため，物理的に盗み見る機会が多くセ

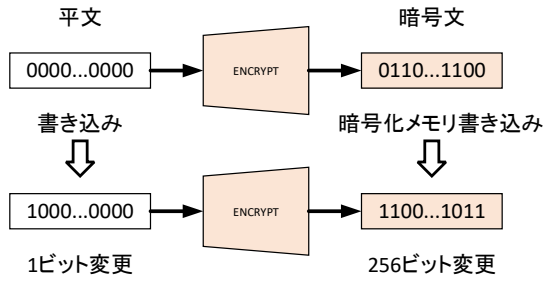

図 3 暗号化メモリによる書き込みビットの拡散.

キュリティ上の懸念事項となっている．これらの攻撃に対処するためにチップ内部ではラストレベルキャッシュに平文として情報を保持し，チップ外部に書き戻すときに情報を暗号化してメモリ上に書き戻す暗号化メモリ手法が存在する．図 2 に暗号化メモリの構造を示す．暗号化メモリでは OTP (One Time Pad) と呼ばれる暗号ベクトルをチップ内部で生成し，チップ内部の LLC (Last Level Cache) からチップ外部へ平文情報が移動するときに OTP を平文情報に XOR することで暗号文に変換している．チップ外部の暗号化メモリからチップ内部の LLC に暗号文情報が移動するときに OTP を XOR して暗号文情報を平文情報に復号している．[5] ではキャッシュラインサイズは 512 ビットとし，64 ビットごとに OTP により暗号化・復号している．OTP はキャッシュラインアドレスや秘匿情報をもとに AES により生成される．

暗号化メモリでは暗号ベクトルを情報に XOR して暗号文を生成するため，暗号文は乱雑さが高く，あるビットが 0 になる確率と 1 になる確率がほぼ等確率となる．つまり平均的に 0.5 の確率でビット書き込みが発生する．実際にはキャッシュラインの一部のデータしか情報を書き込む必要が無い場合でも全体を暗号化すると書き込み量は膨大になる．PCM では書き込み耐性が小さく，暗号化メモリではメモリセルの寿命がさらに短くなってしまう．図 3 に暗号化メモリによる書き込みビットの拡散を示す．LLC 上の平文情報に 1 ビットの変更があった場合でも，暗号化メモリ書き戻す際には暗号化するため多くのビットは反転している可能性がある．キャッシュラインをより細かいワード単位で区切り，1 ビットも変更がない場合は再暗号化をしない手法が研究されている．

不揮発性メモリの 1 つのメモリセルに複数のビットを割り当てるマルチレベルセルやトリプルレベルセルが存在する．マルチレベルセルやトリプルレベルセルではメモリに対する書き込み量はビットレベルではなく，セルレベルで数える必要がある．ビットレベルではなくセルレベルで書き込み量を表現する指標として CHD (Cell Hamming Distance) が存在する [9]．1 つのメモリセルに 1 ビットを割り当てるシングルレベルセルでは平均的に 0.5 の確率で書き込みが発生する場合であっても，マルチレベルセルやトリプルレベルセルではセルに対する書き込みが発生する

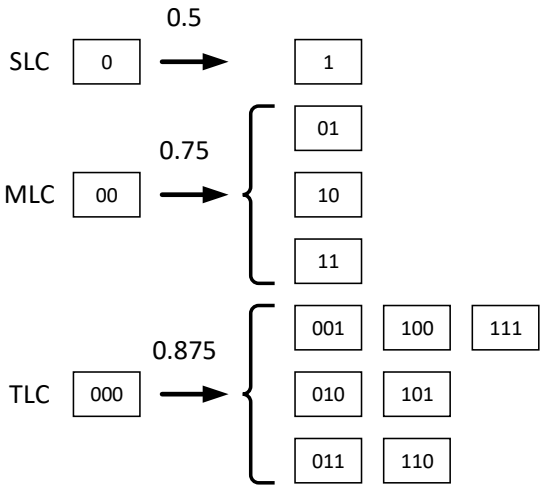

図 4 SLC (Single Level Cell), MLC (Multi Level Cell), TLC (Triple Level Cell) におけるセル書き込み確率.

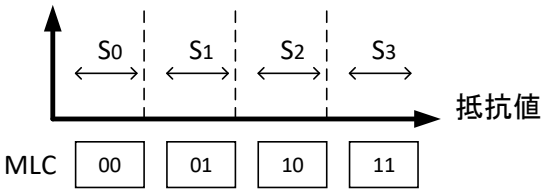

図 5 PCM セルの抵抗値と状態 S_0, S_1, S_2, S_3，ビット列の対応.

確率が高くなる．図 4 に示すように，マルチレベルセルではメモリセルに書き込みをしなくてよいためには，そのメモリセルが表わす 2 ビットが一致しなければならない．メモリセルが表わす 2 ビットのうち 1 ビットでも書き込みが発生する場合にはメモリセルそのものへの書き込みが発生する．マルチレベルセルでメモリセルに書き込みが発生する確率は 0.75 であり，同様にトリプルレベルセルでは 0.875 である．マルチレベルセル，トリプルレベルセルではメモリセルに対する書き込みの発生確率がシングルレベルセルよりも高くなるため，より強く書き込み量削減の需要がある．

表 1 $GF(2^2)$ 上の加法.

	0	1	α	α^2
0	0	1	α	α^2
1	1	0	α^2	α
α	α	α^2	0	1
α^2	α^2	α	1	0

3. 準備

MLC (Multi Level Cell) PCM は図 5 に示すように 1 セルで 4 つの状態を表し，2 ビットの情報を保持できる．メモリセルの抵抗値が低い状態を S_0 としてビット列 00 を表しているとみなす．同様に状態 S_1,S_2,S_3 のときにそれぞれビット列 01,10,11 を表しているとみなす．

MLC 不揮発性メモリにおいて書き込み量はビットレベル

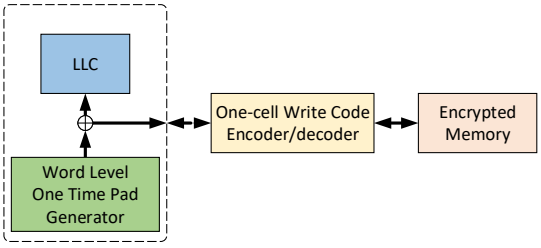

図 6 低書き込みメモリ暗号化手法.

ではなくセルレベルで考える必要がある．セルレベルで考えるために，2 ビットを 1 つのシンボルとして扱う．ガロア体 $GF(2)=\{0,1\}$ に対し，$GF(2)$ 上の方程式 $x^2+x+1=0$ の解 α を元として加えた拡大体 $GF(2^2)=\{0,1,\alpha,\alpha^2\}$ を考える．$GF(2^2)$ の元をシンボルとして用いることでビットレベルの書き込み量ではなく，セルレベルの書き込み量として異なるシンボルの個数を比べれば良い．CHD は $GF(2^2)$ を係数とする n 次線形空間 $GF(2^2)^n$ においてハミング距離と一致する．

元 $0,1,\alpha,\alpha^2$ がそれぞれビット列 00,01,10,11 を表し，すなわちメモリセルの状態 S_0,S_1,S_2,S_3 を表わすことにする．表 1 に $GF(2^2)$ の元における加法の結果を示す．α は $\alpha^2+\alpha+1=0$ を満たすため，元 α^2 と元 α の加法の結果は元 1 となる．XOR 演算においてビット列 00,01,10,11 がなす加法群と $GF(2^2)$ は同型であることがわかる．

シンボルが n 個存在するとき $GF(2^2)$ 上の n 次ベクトルとみなすことができ，n 次線形空間 $GF(2^2)^n$ 上の元とみなすことができる．ビット列で 000110 を表す場合に，メモリ上では状態 S_0,S_1,S_2 のメモリセルが存在することを意味する．本稿ではビット列 000110 を $GF(2^2)^n$ 上の元としてベクトル $(0,1,\alpha)$ と表記する．ベクトル同士の加法は，要素ごとに $GF(2^2)$ 上で加法をとればよい．ベクトル $(0,1,\alpha)$ とベクトル (α,α,α) の加法の結果はベクトル $(\alpha,\alpha^2,0)$ となる．

4. 低書き込みメモリ暗号化手法と 1 セル書き込み符号

暗号化メモリへの書き込みセル数を削減するために暗号ベクトルによる暗号化とは別に書き込み量を削減する符号化処理をするシステムとそのための 1 セル書き込み符号を提案する．さらに 1 ワードをより細かい単位に区切ることで変更がない場合に発生する暗号化メモリの拡散効果を防止する．

マルチレベルセル不揮発性メモリの 1 ワードを n セルで構成する．n セルのワードは，符号化されていないならば $2n$ ビットのビット列となる．本稿では 1 ワードは k ビット情報を持つように符号化する．1 セルで 2 ビットを表現するマルチレベルセル不揮発性メモリでは，符号化されていない場合に $\frac{3}{4}$ の確率でセル書き込みが発生する．1 ワードあたり 1 セル書き込みで情報を保持できる 1 セル書き込み符号を提案する．提案符号はパラメータとしてメモリセル数 n，情報量 k ビットを用いて (n,k)-1 セル書き込み符号と表現する．m を非負整数のパラメータ変数として，n,k は式 1,2 とする．

$$k = 2m \tag{1}$$

$$n = \frac{2^k-1}{3} \tag{2}$$

n メモリセルが表現する n 次線形空間 $GF(2^2)^n = \{0,1,\alpha,\alpha^2\}^n$ に対し，式 3 を満たす，$GF(2^2)^n$ の部分空間となる n 次線形空間 A を考える．

$$A \subset GF(2^2)^n \tag{3}$$

このとき $GF(2^2)^n$ の A による商空間 $GF(2^2)^n/A$ が存在する．X を式 4 と定義する．$GF(2^2)^n$ と A は線形空間であり，A を同値関係とみなしたとき X は線形空間となる．

$$X = GF(2^2)^n/A \tag{4}$$

商空間 X は式 5 を満たすベクトル s,t に対して，式 6 を満たすとき s,t は同値であると呼び．同値であるような集合（同値類）によって X を生成できる．特に本稿では同値類の中で最もハミング重みの小さいベクトルを同値類の代表として X の元とみなす．

$$s,t \in GF(2^2)^n \tag{5}$$

$$s-t \in A \tag{6}$$

式 7 に示すように A と X の直積 $\otimes$ は $GF(2^2)^n$ となる．よって $GF(2^2)^n$ 上のベクトル $v \in GF(2^2)^n$ は A 上のベクトル $a \in A$ と X 上のベクトル $x \in X$ を用いて，一意に $v = a \oplus x$ と表現できる．

$$A \otimes X = GF(2^2)^n \tag{7}$$

4.1 1 セル書き込み符号構成法

4.1.1 Step 1:G_A を構成

A は線形空間（線形符号）であるため生成行列 G_A により構成できる．G_A を構成することを考える．$GF(2^2)$ を要素とするハミング重みが 2 以上の m 次元ベクトルを考える．$GF(2^2)$ を要素とする m 次元ベクトルは $4^m = 1+3n$ 個存在する．ハミング重みが 0 の m 次元ベクトルは 1 個，ハミング重みが 1 の m 次元ベクトルは $3\times m$ 個存在する．よって $GF(2^2)$ を要素とするハミング重みが 2 以上の m 次元ベクトルは 3(n-m) 個存在する．スカラー倍には 1 倍，α 倍，α^2 倍の 3 通りあるため，互いに独立ではないベクトルの組は 3 通りある．この $GF(2^2)$ を要素とするハミング重みが 2 以上の m 次元ベクトルから互いに独立なベクトル

を $n-m$ 個選ぶ．選んだ m 次元ベクトルを横ベクトルとみなし，$n-m$ 個を縦に並べて $n-m \times m$ 行列 P を作る．I を $n-m \times n-m$ 単位行列とする．I，P を並べることで A の生成行列 G_A が構成できる．生成行列 G_A を式 8 に示す．G_A は $(n-m) \times n$ 行列である．

$$G_A = \begin{pmatrix} I & P \end{pmatrix} \tag{8}$$

4.1.2 Step 2:A を構成

A の要素数は $|A| = 2^{n-m}$ である．よって $GF(2^2)^{n-m}$ の元であるベクトル $u \in GF(2^2)^{n-m}$ を生成行列 G_A にかけることで生成できる．式 9 に生成される A を示す．

$$uG_A \in A \tag{9}$$

4.1.3 Step 3:X を構成

式 10 に示すように，$GF(2^2)^n$ 上のハミング重みが 0 または 1 となるベクトルで X を構成する．このとき式 4 が成立する．

$$\begin{aligned} X = \{ & (0,0,0,\ldots,0), \\ & (1,0,0,\ldots,0), \\ & (\alpha,0,0,\ldots,0), \\ & (\alpha^2,0,0,\ldots,0), \\ & (0,1,0,\ldots,0),\ldots\} \end{aligned} \tag{10}$$

X の要素数は $|X| = 1+3n = 2^k$ となる．X の要素がハミング重み 1 以下となる A を生成できれば 1 セル書き込み符号を構成できる．

4.1.4 Step 4:情報と X の元を対応

1 セル書き込み符号は $k = 2m$ ビット情報の符号である．m 次線形空間 $GF(2^2)^m$ の元を X の元に対応付ける．対応関係は任意に対応付けて良い．

4.2 符号構成例

$n = 5$，$k = 4$ となる $(5,4)$-1 セル書き込み符号の構成例を示す．$k = 2m$ より $m = 2$ である．

Step 1:G_A を構成 $GF(2^2)$ を要素とするハミング重みが 2 以上の $m = 2$ 次元ベクトルは $(1,1)$, $(1,\alpha)$, $(1,\alpha^2)$, $(\alpha,1)$, (α,α), (α,α^2), $(\alpha^2,1)$, (α^2,α), (α^2,α^2) の 9 個存在する．そのうち互いに独立なベクトルを $n-m = 3$ 個選ぶ．互いに独立とは一次独立を意味しない．任意の 2 ベクトルが互いに独立であればよい．ベクトル $(1,1)$,$(1,\alpha)$,$(1,\alpha^2)$ を選び，式 11 に示すように $(n-m) \times m$ 行列 P を構成する．

$$P = \begin{pmatrix} 1 & 1 \\ 1 & \alpha \\ 1 & \alpha^2 \end{pmatrix} \tag{11}$$

式 8，式 11 より生成行列 G_A は式 12 と表せる．

$$G_A = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & \alpha \\ 0 & 0 & 1 & 1 & \alpha^2 \end{pmatrix} \tag{12}$$

Step 2:A を構成 $GF(2^2)^3$ の元を G_A にかけることで A の元を生成できる．

$$\begin{aligned} A = \{ & (0,0,0,0,0), \\ & (1,0,0,1,1), \\ & (\alpha,0,0,\alpha,\alpha), \\ & (\alpha^2,0,0,\alpha^2,\alpha^2), \\ & (0,1,0,1,\alpha),\ldots\} \end{aligned} \tag{13}$$

Step 3:X を構成 $GF(2^2)^5$ の元のうちハミング重みが 1 以下の元を取り出すことで X を生成できる．

$$\begin{aligned} X = \{ & (0,0,0,0,0), \\ & (1,0,0,0,0), \\ & (\alpha,0,0,0,0), \\ & (\alpha^2,0,0,0,0), \\ & (0,1,0,0,0),\ldots\} \end{aligned} \tag{14}$$

Step 4:情報と X の元を対応 $GF(2^2)^2$ の元を表 2 のように X の元に対応付ける．

表 2 $GF(2^2)^2$ の元と X の元の対応].

$d \in GF(2^2)^2$	$x \in X$
$(0,0)$	$(0,0,0,0,0)$
$(1,0)$	$(1,0,0,0,0)$
$(\alpha,0)$	$(\alpha,0,0,0,0)$
$(\alpha^2,0)$	$(\alpha^2,0,0,0,0)$
$(0,1)$	$(0,1,0,0,0)$
$(1,1)$	$(0,\alpha,0,0,0)$
$(\alpha,1)$	$(0,\alpha^2,0,0,0)$
$(\alpha^2,1)$	$(0,0,1,0,0)$
$(0,\alpha)$	$(0,0,\alpha,0,0)$
$(1,\alpha)$	$(0,0,\alpha^2,0,0)$
(α,α)	$(0,0,0,1,0)$
(α^2,α)	$(0,0,0,\alpha,0)$
$(0,\alpha^2)$	$(0,0,0,\alpha^2,0)$
$(1,\alpha^2)$	$(0,0,0,0,1)$
(α,α^2)	$(0,0,0,0,\alpha)$
(α^2,α^2)	$(0,0,0,0,\alpha^2)$

5. 1 セル書き込み符号の性質

1 セル書き込み符号は情報を線形空間 X の元に割り当て，線形空間 A の元は書き込み量削減のためのベクトルのバイアスに費やす．A と X の直積が線形空間 $GF(2^2)^{n-m}$ 全体

となるため，任意のベクトル $v \in GF(2^2)^{n-m}$ は一意に A の元 a と X の元 x の和として表現できる．いまメモリセルに入っているワードが $a_0 + x_1 = (0,0,0,0,0) \oplus (1,0,0,0,0)$ だとする．表 2 よりこのワードは情報 $d = (1,0)$ を表わす．メモリセルの情報を $d' = (0,\alpha)$ に書き換えたいとする．X の元のいずれかを書き込むと任意の情報を書き込める．エンコーダにより最適なベクトルを選び，$W = (0,0,0,\alpha^2,0)$ を書き込むと，$a_0 + x_1 + W = (1,0,0,\alpha^2,0) = (1,0,\alpha,\alpha^2,0) + (0,0,0,\alpha^2,0) = a_{33} + x_8$ となり，d' に対応するベクトル $(0,0,0,\alpha^2,0)$ を書き込めた．

定理 1. 1 セル書き込み符号はどのようなワードに対しても 1 セルの書き込みで情報を保持できる．

証明. 1 セル書き込み符号は任意のワードに対して $a_i \in A$, $x_j \in X$ となる $a_i + x_j$ に一意に分解できる．このとき W を書き込んで情報 d に対応するベクトル $x_d \in X$ を書き込みたい．$a_s \in A$ に対して式 15 と書ける．

$$a_i + x_j + W = a_s + x_d \tag{15}$$

式 15 を式変形すると式 16 となる．

$$a_i + x_j + x_d = a_s + W \tag{16}$$

ここで式 16 の左辺は既知である．右辺は A と X の元の和として一意に表せるので，W は X の元となるものを求めることができる．X の元はハミング重み 1 以下なので，1 セル書き込み符号はどのようなワードに対しても 1 セルの書き込みで情報を保持できる． □

定理 2. 1 セル書き込み符号は書き込みのハミング距離の最大値に対してセルの長さをこれ以上短く出来ない最適符号である．

証明. $GF(2^2)^n$ 上の n 次ベクトルにおいてハミング距離が 0 のものは 1 個，ハミング距離が 1 のものは $3n$ 個ある．このベクトルにハミング距離 1 以下でシンボル変更するとき，与えられる情報量の最大値が $\log_2(1+3n)$ である．1 セル書き込み符号は k ビット情報を保持し，$1 + 3n = 2^k$ であるので 1 ワードの書き込みに情報量の最大値を与えることができる．よってこれ以上セルを短くすると与えられる情報が減ってしまうため，最適符号である． □

6. おわりに

本稿では不揮発性メモリを対象として，乱数文と排他的論理和をとった暗号文の暗号の強度を保ったまま書き込み量を削減できるシステムを提案した．さらに 1 セル書き込み符号を提案し，1 セル以下で書き込みが行えることを証明した．今後はハードウェア実装により回路面積やオーバヘッドの評価をする．

謝辞 本研究は JSPS 科研費 16K16028 の助成による．

参考文献

[1] J. Kong and H. Zhou, “Improving privacy and lifetime of PCM-based main memory,” Proceedings of the International Conference on Dependable Systems and Networks, pp.333–342, 2010.
[2] J. Yang, L. Gao, and Y. Zhang, “Improving memory encryption performance in secure processors,” *IEEE Transactions on Computers*, vol.54, no.5, pp.630–640, 2005.
[3] A. Awad, P. Manadhata, S. Haber, Y. Solihin, and W. Horne, “Silent Shredder: Zero-Cost Shredding for Secure Non-Volatile Main Memory Controllers Amro,” in *Proc. Asplos*, pp.263–276, 2016.
[4] V. Young, P.J. Nair, and M.K. Qureshi, “DEUCE: Write-Efficient Encryption for Non-Volatile Memories,” SIGARCH Comput. Archit. News, vol.43, no.1, pp.33–44, 2015.
[5] S. Swami, J. Rakshit, and K. Mohanram, “SECRET: Smartly EnCRypted Energy EfficienT Non-Volatile Memories,” in *Proc. DAC 2016*, pp.1–1, 2016.
[6] M. Tawada, S. Kimura, M. Yanagisawa, and N. Togawa, “Ecc-based bit-write reduction code generation for non-volatile memory,” *IEICE Transactions*, vol.98-A, no.12, pp.2494–2504, 2015.
[7] M. Tawada, S. Kimura, M. Yanagisawa, and N. Togawa, “A bit-write reduction method based on error-correcting codes for non-volatile memories,” in *ProcThe 20th Asia and South Pacific Design Automation Conference, ASP-DAC 2015*, pp.496–501, 2015.
[8] S. Cho and H. Lee, “Flip-N-Write: a simple deterministic technique to improve PRAM write performance, energy and endurance,” in *Proc.MICRO-42*, pp.347–357, 2009.
[9] A. Alsuwaiyan and K. Mohanram, “MFNW: A flip-n-write architecture for multi-level cell non-volatile memories,” in *Proc. NANOARCH 2015*, pp.13–18, 2015.