

## 常活性ラインに基づく高性能/低リーク・キャッシュ の消費エネルギー評価

小宮, 礼子  
福岡大学大学院工学研究科電子情報工学専攻

井上, 弘士  
科学技術振興機構さきがけ

村上, 和彰  
九州大学大学院システム情報科学研究院 | 財団法人九州システム情報技術研究所

<https://hdl.handle.net/2324/11884>

---

出版情報：電子情報通信学会技術研究報告. ICD, 集積回路. 105 (475), pp.37-42, 2005-12-08. 電子情報通信学会  
バージョン：  
権利関係：

# 常活性ラインに基づく高性能/低リーク・キャッシュの 消費エネルギー評価

小宮 礼子<sup>†1†4</sup> 井上 弘士<sup>†2†3</sup> 村上 和彰<sup>†3†4</sup>

†1 福岡大学大学院 工学研究科 電子情報工学専攻 〒814-0180 福岡市城南区七隈 8-19-1

†2 科学技術振興機構さきがけ 〒332-0012 埼玉県川口市本町 4 丁目 1 番 8 号

†3 九州大学大学院 システム情報科学研究院 〒816-8580 福岡県春日市春日公園 6-1

†4 財団法人 九州システム情報技術研究所 〒814-0001 福岡市早良区百道浜 2-1-22

E-mail: †1†3 arch-ccc-lpc"at"c.scse.kyushu-u.ac.jp

**あらまし** 我々はこれまでに、高性能かつ低リークなキャッシュを提案した。この手法は性能低下を引き起こすラインを常に活性状態として動作させる事でペナルティを低減する。これまではリーク消費エネルギーおよび性能のみの評価であった。そこで本稿では、下位記憶階層へのアクセスに伴う消費エネルギーや、本手法適用に伴うエネルギー・オーバヘッド等も含めた消費エネルギーを詳細に評価した。我々が提案する高性能/低リーク・キャッシュを適用すると、従来の低リーク・キャッシュである Cache decay 方式よりも 8% 程度のエネルギー・オーバヘッドで最大 20% もの性能低下を抑制できた。また、本方式が有効に動作した場合、Cache decay に比べ消費エネルギーの削減効果も得られた。

**キーワード** 高性能, 低リーク, キャッシュ, 消費エネルギー

## Evaluation of energy consumption for High-Performance / Low-Leakage Caches based on Always Active line

Reiko KOMIYA<sup>†1†3</sup> Koji INOUE<sup>†2</sup> and Kazuaki MURAKAMI<sup>†2†3</sup>

†1 Department of electronics engineering and computer Science at Fukuoka University

8-19-1 Nanakuma, Jhonan-ku, Fukuoka, 814-0180 Japan

†2 Department of Informatics at Kyushu University 6-1 Kasuga-koen, Kasuga-shi, Fukuoka, 816-8580 Japan

†3 Institute of Systems & Information Technologies/KYUSHU 2-1-22 Momochihama, Sawara-ku,

Fukuoka, 814-0001 Japan

E-mail: †1†2 arch-ccc-lpc"at"c.scse.kyushu-u.ac.jp

**Abstract** So far we proposed a cache management technique to alleviate the negative effect of low-leakage caches. This technique makes improvements to performance degradation. In our approach, cache lines which caused the performance degradation are forced to stay in the high-speed but high-leakage mode. Before now we evaluated only leakage energy and performance. Therefore, we perform detailed energy evaluation including access energy for low hierarchy memory and overhead on additional circuit. In our evaluation, the proposed scheme brings about 70~80% of energy reduction by only a 5% of performance degradation.

**Keyword** High performance, Low leakage, Cache, Power

### 1. はじめに

携帯電話やノート型 PC などバッテリー駆動型機器の普及に伴い、マイクロプロセッサ・システムの低消費エネルギー化が重要視されるようになった。一般に、CMOS 回路の消費エネルギーは、動的消費エネルギーと静的消費エネルギーの 2 つに大別される。前者は回路負荷容量の充放電によって、また、後者はトランジ

スタの漏れ電流によって消費されるエネルギー(リーク消費エネルギー)である。従来の CMOS 回路では動的消費エネルギーが多くの割合を占めていた。しかしながら、微細化加工技術の進歩に伴い、リーク消費エネルギーによる影響が大きくなってきた。特に、大量のトランジスタで構成されるキャッシュ・メモリにおいては、リーク消費エネルギーの削減が極めて重要となる。例えば、0.07  $\mu$  m プロセスを想定した場合、キ

キャッシュ全消費エネルギーの70%はリークに起因するとの予測もある[2].

この問題を解決するため、これまでに様々なキャッシュ・リーク消費エネルギー削減手法が提案された。これらの手法は、以降参照されないと予測されたキャッシュ・ラインを動的に待機状態へ切替えることによってリークを削減する。待機ラインの実現方法には状態破棄([3][7])と状態保存([2][4])の2通りがある。前者は待機状態へと移行するラインのSRAMセルに対して電源電圧の供給を停止することによりリーク消費エネルギーを削減する。そのためSRAMセルに記憶されていた情報は失われ、キャッシュ・ミス回数が増加する。一方、後者はSRAMセルに記憶されたデータを失わない程度まで電源電圧を下げることで実現される。状態破棄と比較して、ライン当たりのリーク消費エネルギーは大きくなるものの、従来型キャッシュと同じヒット率を維持できる。しかしながら、待機ラインを参照する場合、供給電圧を回復する必要があるため、アクセス・ペナルティが発生する。したがって、どちらの待機ライン実現方式を適用したとしても、プロセッサの性能は低下する。

そこで、我々はこれまでに低リークキャッシュにおける性能低下の抑制を目的として、待機ラインへの参照の局所性を活用した動的キャッシュ制御方式を提案し、評価した[5]。本手法では、性能低下の原因である待機状態時の参照が集中して発生するラインを活性状態として動作させる。これにより、待機ライン参照に伴うアクセス・ペナルティを回避し、性能低下を抑制する。本方式が最も効果的に動作した結果、従来の低リークキャッシュと同程度のリーク削減率を維持しつつ、性能低下を10%抑制できた[5]。この評価はリーク消費エネルギーと性能のみの評価であり、下位記憶階層へのアクセスに伴う消費エネルギーや、本手法適用に伴うエネルギー・オーバヘッド等を含んでいない。そこで本稿では、これらも含めた詳細な消費エネルギーの評価を行う。本方式が最も効果的に動作した場合、従来の低リークキャッシュと比較して、性能改善効果に加え、消費エネルギーも削減した。

以下、第2節で性能低下を抑制するキャッシュ制御方式を説明し、その手法の消費エネルギー・モデルを第3節で詳しく述べる。次に、第4節でベンチマーク・プログラムを用いた定量的評価を行い、最後に第5節で簡単にまとめる。

## 2. 低リーク・キャッシュの性能低下抑制方式

待機状態の実現方法に関して、下位記憶階層への参照クロック・サイクルが長いならば状態保存、短いならば状態破棄が適していると文献[6]で報告されている。し

かしながら、トランジスタの微細化加工技術の進歩に伴い、電源電圧値は今後低下していくと予測される。この場合、データが損なわれない程度の低い電源電圧と通常の電源電圧を頻繁に切替えることは困難である。そのため、状態破棄による待機状態の実現方法が今後主流になると予想される。そこで本稿では、状態破棄による低リーク・キャッシュを想定する。また、我々が提案する低リーク・キャッシュの制御方法はどの低リーク・キャッシュに対しても適用可能であるが、本稿では状態破棄により待機ラインを実現する手法として代表的な”Cache decay”を採用する。Cache decayは、アクセスされること無しにキャッシュ内に滞在する期間(非アクセスサイクル数)が閾値期間(decay-interval)以上となった時点で待機状態へと変更し、アクセスが発生したラインに対して活性ラインに変更する事でリーク消費エネルギーを削減する。

状態破棄によって待機ラインを実現した場合、待機ラインのデータは破棄されるため当該ラインへの参照は必ずキャッシュ・ミスとなる。再参照されるデータを破棄した場合、リーク削減手法を用いない通常のキャッシュ・メモリと比較してミス回数が増加する。その結果、プロセッサの性能は低下する。本稿では、従来の低リーク・キャッシュとしてCache decayを採用しているので、リーク削減手法適用に伴い増加するミスの事を”Decay ミス”と呼ぶ。高い性能と低消費エネルギーを両立させるためには、低リーク・キャッシュにおいて、Decay ミスによる性能低下を抑制することが重要となる。

### 2.1. Decay ミス密度

メモリ参照に局所性が存在するのと同様に、Decay ミスに関しても空間的局所性が存在する[5]。そこで、キャッシュ・ライン*i*におけるDecay ミスの参照密度 $DMD_i$ (Decay Miss Density)を以下の式で定義する。

$$DMD_i = \frac{N_i}{N_{avg}} \quad (1)$$

$N_i$ はライン*i*におけるDecay ミス回数、 $N_{avg}$ は全ラインに関するDecay ミス回数の平均である。例えば、 $DMD_2$ が1以上の場合、ライン2は平均以上のDecay ミスが発生していることになる。

### 2.2. 常活性ライン方式

性能低下を抑制するには、再参照されるラインを活性状態に保つ必要がある。その手段として、集中的にDecay ミスが発生するラインを常に活性ラインとして動作させ、待機状態への切替えを防ぐ制御方式を提案する。待機ラインへの参照局所性の判定には $DMD_i$ の

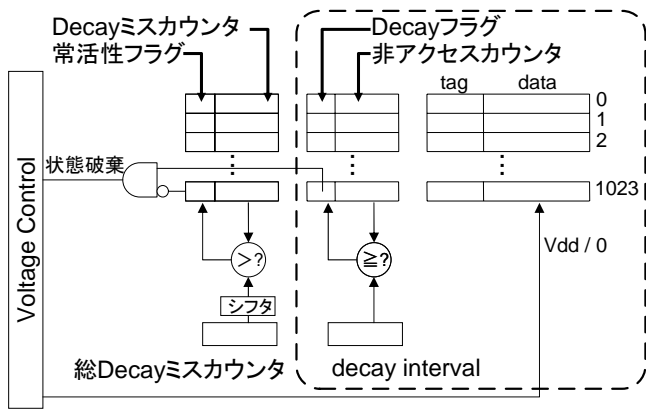


図 1：高性能/低リーク・キャッシュ

値を用いる。  $DMD_i$  がある閾値より大きいならばそのラインは性能低下を引き起こしていると判断し、常に活性ラインとして動作させる。以降このラインを“常活性ライン”と呼ぶ。  $DMD_i$  が閾値以下の場合、当該ラインは性能低下に影響を与えていないと推測されるため通常の低リーク・キャッシュ(本稿では Cache decay)と同様に動作する。この方式を“常活性ライン方式”と呼ぶ。常活性ラインでは Decay ミスが発生しないため、性能改善効果に加え、下位記憶階層への参照エネルギーも削減できる。

例として、常活性ラインによる性能改善手法を支援する 1 ウェイ 1024 ラインのキャッシュを図 1 に示す。内部の点線で囲まれた領域が従来の低リーク・キャッシュ Cache decay である。Cache decay は、非アクセスカウンタにて非アクセスサイクル数をラインごとに保持する。そのカウンタ値が decay-interval 以上となった時点で待機状態への遷移が起きる(Decay フラグ=1, データ領域の供給電圧=0)。その後、アクセスが発生したラインに対して活性ラインへの切替えを行う。

次に、常活性ライン方式を支援するために必要な機構を考える。常活性ラインへの切替えは以下の式が成立した際に行われる。

$$DMD_i > \text{閾値} \quad (2)$$

ここで、式(1)を式(2)に代入する。

$$\frac{N_i}{N_{avg}} > \text{閾値} \quad (3)$$

式(3)より、

$$N_i > N_{avg} \times \text{閾値} \quad (4)$$

式(4)の右辺は、総 Decay ミス回数を  $\{\log_2(\text{ライン数}) - \log_2(\text{閾値})\}$  ビット右シフトする事で求まる。したがって、常活性ライン方式を実現するには各ラインの Decay ミス回数および、総 Decay ミス回数の 2 つが必要となる。そのため、各ラインに常活性フラグと Decay

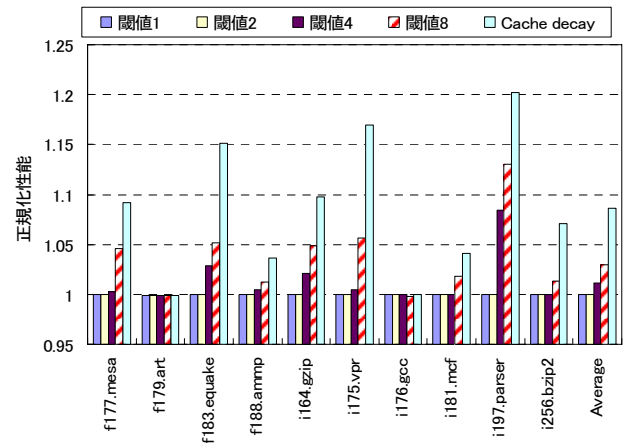


図 2：性能低下

ミスカウンタ、また、キャッシュ全体の Decay ミス回数を保持する総 Decay ミスカウンタおよび比較器を追加する。Decay ミス発生時(タグヒットかつ Decay フラグ 1 の場合)に総 Decay ミスカウンタと当該ラインの Decay ミスカウンタをカウントアップする。当該ラインの Decay ミス回数が総 Decay ミスカウンタ値をシフトした値より大きいならば、そのラインは性能低下を引き起こしていると判断し、常活性ラインに切替える(常活性フラグ=1)。常活性ラインは Decay フラグの状態に関わらず、常に活性ラインとして動作する。逆に、当該ラインの Decay ミス回数が総 Decay ミスカウンタ値をシフトした値よりも小さい場合、そのラインは性能低下に影響を与えていないと推測されるため、Cache decay と同様に動作する。

常活性ライン数が過剰になると、性能低下は改善されるがリーク消費エネルギーの削減効果が得られなくなる。それに対して、常活性ライン数が極端に少ない場合は従来の低リーク・キャッシュと比較して性能が改善されない。

### 2.3. 性能改善効果

常活性ライン方式適用による性能低下改善効果は文献[5]で既に評価した。評価モデルは、常活性ラインへ切替える  $DMD_i$  の閾値が 1, 2, 4, 8 の 4 パタン(それぞれ閾値 1 モデル, 閾値 2 モデル, 閾値 4 モデル, 閾値 8 モデルと呼ぶ)と Cache decay であり、各評価モデルの実行時間を図 2 に示す。各値はリーク削減手法を適用しない通常のキャッシュ・メモリにおけるプログラム実行時間で正規化している。この評価結果から、常活性ライン方式は Cache decay と比較して性能低下を抑制できる事が示せた。全ベンチマークを平均すると、閾値 1, 2, 4, 8 モデルにおいて、それぞれ 8.6%, 8.6%, 7.4%, 5.6% の性能低下抑制効果が得られた。

### 3. 消費エネルギー・モデル

本節では、2.2 節で述べた常活性ライン方式が消費エネルギーに与える影響を調査するため、下位記憶階層も含めた消費エネルギー・モデルを定義する。本稿では、データ L1 キャッシュに対する常活性ライン方式の適用を想定する。そこで、データ L1 キャッシュの消費エネルギー、およびその下位記憶階層である L2 キャッシュ、もしくは主記憶における消費エネルギーを考える。常活性ライン方式を用いた場合の消費エネルギー( $E_{total}$ )は以下のようになる。

$$E_{total} = E_{L1} + E_{下位} \quad (5)$$

$$E_{L1} = LE_{L1} + DE_{L1} \quad (6)$$

$$E_{下位} = LE_{下位} + DE_{下位} \quad (7)$$

ここで、 $E_{L1}$  と  $E_{下位}$  は、それぞれデータ L1 キャッシュおよび下位記憶階層で消費するエネルギーであり、それぞれリーク消費エネルギー( $LE$ )と動的消費エネルギー( $DE$ )に分けられる。

まず、リーク消費エネルギーについて考える。 $LE_{L1}$  は以下の式で与えられる。

$$LE_{L1} = \sum_{i=1}^{cc} LE_{bit} \times N_{active}(i) \quad (8)$$

$CC$  はプログラム実行時間(クロックサイクル数)、 $LE_{bit}$  はクロックサイクル当りの 1 ビット SRAM セル平均リーク消費エネルギー、 $N_{active}(i)$  は  $i$  クロックサイクル時に活性状態で動作しているセルの総ビット数を表す。常活性ライン数の増加に伴い  $N_{active}(i)$  も増え、一方、 $CC$  は減少する。

下位記憶階層は我々が提案する常活性ライン方式適用に関わらず、常に活性状態として動作している。そこで、本評価では下位記憶階層におけるリーク消費エネルギーは考慮しない。

$$LE_{下位} \doteq 0 \quad (9)$$

次に、動的消費エネルギーを定義する。データ L1 キャッシュの回路構成は大きく 3 つに分けられる。1 つ目はリーク削減手法を用いない通常のキャッシュメモリ実現に必要な回路、2 つ目は Cache decay 実現に伴い追加される回路(Decay フラグ、非アクセスカウンタや比較器等)、最後に 3 つ目は常活性ライン方式の実現に必要な追加回路(常活性フラグ、Decay ミスカウンタ、シフタや比較器等)である。これらの回路の動作に伴い、データ L1 キャッシュはエネルギーを消費する。したがって、 $DE_{L1}$  は以下のように表される。

$$DE_{L1} = DE_{L1\_org} \times N_{L1\_org} + DE_{L1\_decay} \times N_{L1\_decay} + DE_{L1\_aa} \times N_{L1\_aa} \quad (10)$$

表 1: プロセッサ構成

|                         |   |
|-------------------------|---|
| 命令発行方式                  | インオーダー                                  |
| 分岐予測器                   |   |
| 種類                      | bimodal                                 |
| サイズ                     | テーブル:128 エントリ, 1 ウェイ                    |
| 命令デコード幅                 | 2 命令/サイクル                               |
| 命令発行幅                   | 2 命令/サイクル                               |
| IFQ サイズ                 | 2 エントリ                                  |
| RUU サイズ                 | 16 エントリ                                 |
| LSQ サイズ                 | 16 エントリ                                 |
| キャッシュ・メモリ               |   |
| L1 データ                  | 32KB (32B/エントリ, 32 ウェイ, 1K エントリ)        |
| L1 命令                   | 32KB (32B/エントリ, 32 ウェイ, 1K エントリ)        |
| L2 共有                   | なし                                      |
| ヒットレイテンシ                |   |
| L1 キャッシュ                | 1 クロック                                  |
| L2 キャッシュ                | なし                                      |
| 主記憶                     | 32 クロック                                 |
| メモリ・バンド幅                | 8B                                      |
| メモリ・ポート数                | 1                                       |
| ITLB, DTLB              |   |
| エントリ数                   | 1M エントリ (4KB/エントリ, 32 ウェイ, 32 エントリ/ウェイ) |
| ミスペナルティ                 | 30 サイクル                                 |
| 整数演算器                   |   |
| (装置数, 実行レイテンシ, 発行レイテンシ) |   |
| ALU                     | 1, 1 クロック, 1 クロック                       |
| 乗算器                     | 1, 3 クロック, 1 クロック                       |
| 除算器                     | 1, 20 クロック, 19 クロック                     |
| 浮動小数点演算器                |   |
| (装置数, 実行レイテンシ, 発行レイテンシ) |   |
| ALU                     | 1, 2 クロック, 1 クロック                       |
| 乗算器                     | 1, 4 クロック, 1 クロック                       |
| 除算器                     | 1, 12 クロック, 12 クロック                     |

ここで、 $DE_{L1\_org}$  はリーク削減手法を用いないキャッシュにおける 1 参照あたりの平均動的消費エネルギー、 $DE_{L1\_decay}$ 、 $DE_{L1\_aa}$  は順に、非アクセスカウンタ活性時、および、Decay ミスカウンタ活性時に消費する平均動的エネルギーである。また、 $N_{L1\_org}$ 、 $N_{L1\_decay}$ 、 $N_{L1\_aa}$  は各回路の動作回数であり、それぞれ、データ L1 キャッシュ参照回数、非アクセスサイクル数、Decay ミス発生回数である。

最後に、 $DE_{下位}$  は下位記憶階層における 1 参照あたりの平均動的消費エネルギー( $DE_{下位\_org}$ )と下位記憶階層参照回数( $N_{下位}$ )によって求められる。

$$DE_{下位} = DE_{下位\_org} \times N_{下位} \quad (11)$$

常活性ライン方式を適用すると、 $N_{下位}$  が削減される。

### 4. 評価

本節では、常活性ライン方式を適用した低リーク・キャッシュの消費エネルギー評価を行う。具体的にはリーク削減手法を適用しない通常のキャッシュ・メモリの消費エネルギーで正規化した値を求め、従来の低

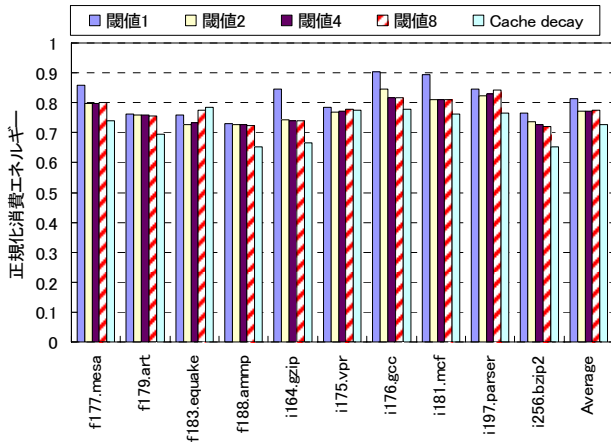


図 3: 消費エネルギー削減効果

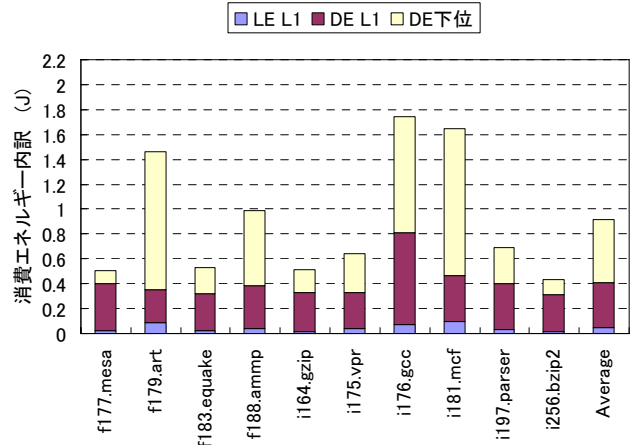


図 4: 消費エネルギー内訳(Cache decay)

リーク・キャッシュである Cache decay と比較することで本方式の有効性を議論する。

#### 4.1. 評価環境

本実験で想定したプロセッサ構成を表 1 に示す。評価対象アプリケーションは SPEC CPU 2000 ベンチマーク・セット[9]より 4 個の浮動小数点プログラムと 6 個の整数プログラムを用いた。なお、本実験では各ベンチマークにおいて、実行開始時から 10 億命令をフォワードし、続く 5 億命令を評価対象とした。

第 3 節で定義した消費エネルギー・モデル式(8)~(11)の、 $CC$ ,  $N_{active}(i)$ ,  $N_{L1\_org}$ ,  $N_{L1\_decay}$ ,  $N_{L1\_aa}$  および  $N_{下位}$  を測定するため、マイクロプロセッサ・シミュレータである SimpleScalar[8]を利用した。それ以外の 5 つの変数  $LE_{bit}$ ,  $DE_{L1\_org}$ ,  $DE_{L1\_decay}$ ,  $DE_{L1\_aa}$  および  $DE_{下位\_org}$  は固定値を与える。まず、 $DE_{L1\_org}$  はキャッシュ・アクセス時間およびエネルギーを算出するシミュレータである hp 社の CACTI3.0[1]を用いて決定した。具体的には、 $DE_{L1\_org}=1.90197(\text{nJ})$ であった。この値を元に、 $DE_{L1\_aa}$  と  $LE_{bit}$  を求める。 $DE_{L1\_aa}$  は  $DE_{L1\_org}$  に対してビット数の比で求める。リーク削減手法を用いないキャッシュ・メモリは 1 エントリあたり、有効フラグが 1 ビット、タグ 25 ビット、データ領域 32B の計 9088 ビットから成っている。一方、Decay ミスカウンタは 20 ビット、常活性フラグは 1 ビットである。したがって、 $DE_{L1\_aa}=4.20536(\text{pJ})$ となる。また、文献[2]において  $LE_{bit}$  と  $DE_{L1\_org}$  の値が示されている。そこで、本稿が想定するキャッシュ構成を考慮すると、 $LE_{bit}=0.1295 \times 10^{-2} (\text{pJ})$ となる。次に、 $DE_{L1\_decay}$  を考える。非アクセスカウンタは階層型カウンタを用いる事により、非常に小さな回路規模で実現される[3]。具体的には、キャッシュ全体に対して 1 つのグローバルカウンタ、また、各ラインに 2 ビットカウンタを保持するだけでよい。この階層型カウンタはグローバルカウンタにより一定サイクルをカウントし、そのカウンタ

が飽和すると各ラインの 2 ビットカウンタを遷移させる。また、ライン参照時に当該ラインの 2 ビットカウンタをリセットする。2 ビットカウンタが飽和したならば、非アクセスサイクル数が一定サイクルを超えたと判断し、当該ラインを待機状態へ切替える。本機構の採用により、非アクセスカウンタにおける消費エネルギーは、その影響を無視できるほど小さくなる。したがって、 $DE_{L1\_decay}=0 (\text{J})$ とする。最後に、下位記憶階層参照時の消費エネルギーは  $DE_{下位\_org}=\alpha DE_{L1\_org}$  とし、 $\alpha=5, 10, 20$  の 3 パターンで評価を行った。本稿では紙面の都合上、 $\alpha=10$  の結果を用いて議論する。

#### 4.2. 評価結果

本提案方式が消費エネルギーに与える影響について評価する。評価モデルは文献[5]と同様に、閾値 1 モデル、閾値 2 モデル、閾値 4 モデルそして閾値 8 モデルの 4 パターンである。各モデルと Cache decay の正規化リーク消費エネルギーを図 3 に示す。

常活性ライン方式は全ベンチマーク、また全パターンにおいて、リーク削減手法を用いない通常のキャッシュと比較して消費エネルギーを削減している。特に、f183.earthquake と i175.vpr は Cache decay と同等もしくはそれ以上の消費エネルギー削減効果を上げている。これは、常活性ライン方式適用に伴う下位記憶階層参照時の動的消費エネルギー( $DE_{下位}$ )の削減効果が L1 キャッシュにおけるリーク消費エネルギー( $LE_{L1}$ )の増加を上回ったからである。図 4, 5 に Cache decay および常活性ライン方式の閾値 2 モデルにおける消費エネルギーの内訳を示す。f183.earthquake の場合、常活性ライン方式は Cache decay と比べて 56.2(mJ)  $LE_{L1}$  が増加したが、95.1(mJ)  $DE_{下位}$  を削減した。この時、性能低下(図 2)は 15.1%改善した。

一方、他のベンチマーク・プログラムは  $LE_{L1}$  増加の

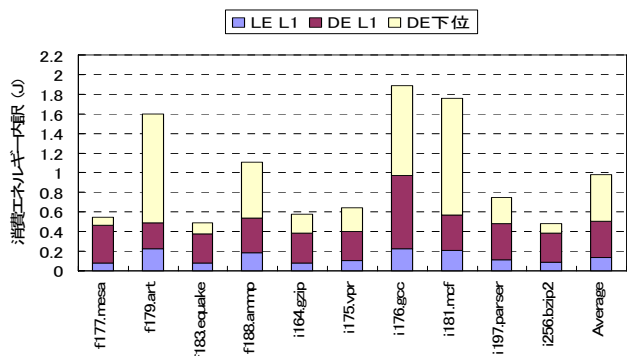


図 5：消費エネルギー内訳  
(常活性ライン方式，閾値 2 モデル)

影響が  $DE_{\text{下位}}$  削減効果よりも大きかったため，Cache decay と比較して消費エネルギーが大きい．特に，閾値 1 モデルはこの特徴を顕著に表す．図 6 に常活性ライン方式における閾値 1 モデルの消費エネルギー内訳を示す．図 5 と比較して，閾値 1 モデルは  $LE_{L1}$  が大きいことが分かる．他のモデルと比較して，閾値 1 モデルは常活性ライン数が多い．つまり，常活性ラインの増大は消費エネルギーの増加を導く．しかしながら，閾値 2, 4, 8 モデルは最大 8.3% の消費エネルギー増加に抑えている．また，いずれのモデルも性能低下抑制効果は高く，最大 20.1% の改善効果が得られる．

これらの結果から，常活性ライン方式は適切なライン選択を行う事により従来の低リーク・キャッシュに対する僅かな消費エネルギー・オーバーヘッドで高い性能改善効果を得られる手法になると言える．

また， $\alpha$  が大きいほど  $DE_{\text{下位}}$  が消費エネルギーに占める割合も増すため，常活性ライン方式適用に伴う Decay ミス回数削減の効果も大きい．よって，下位記憶階層の消費エネルギーが大きいほど，本手法は有効な手法になると言える．

## 5. おわりに

我々は以前，待機状態時の参照局所性が高いラインに対して常に活性状態で動作させることによって性能低下を抑制する方式を提案し，性能およびリーク消費エネルギーの評価を行った．そこで，本稿では下位記憶階層へのアクセスに伴う消費エネルギーや，本手法適用に伴うエネルギー・オーバーヘッド等も含めた詳細な消費エネルギー評価を行った．

その結果，本方式が有効に動作した場合，Cache decay よりも消費エネルギーを削減し，性能改善できることがわかった．しかしながら，常に活性状態で動作させるライン(常活性ライン)数が増えすぎると性能低下は改善されるが消費エネルギーも増加することがわかった．

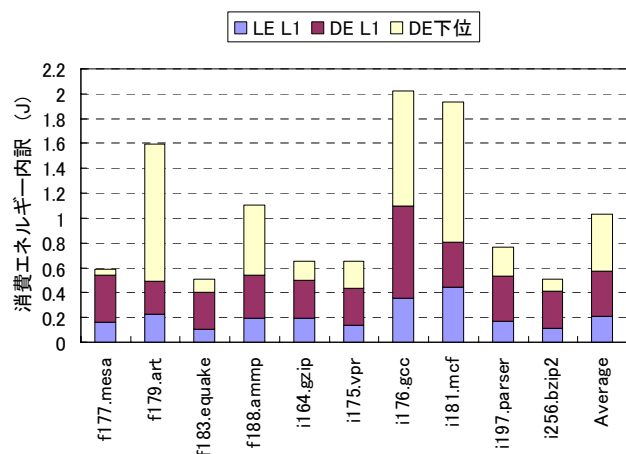


図 6：消費エネルギー内訳  
(常化成ライン方式，閾値 1 モデル)

## 謝辞

本研究を進めるにあたり，多くのご指導を頂いた富士通株式会社の池田正幸氏，丸山拓巳氏，富士通研究所の勝野昭氏，坂本真理子氏に深く感謝致します．なお，本研究は一部，文部省科学研究費補助金(課題番号：14GS0218, 17680005)，ならびに，JST さきがけ研究による．

## 文 献

- [1] CACTI, <http://research.compaq.com/wrl/people/jouppi/CACTI.html>
- [2] K.Flautner, N.S.Kim, S.Martin, D.Blaauw, and T.Mudge, "Drowsy Caches: Simple Techniques for Reducing Leakage Power," Proc. of the 29th Int. Symp. on Computer Architecture, pp.148-157, May 2002.
- [3] S.Kaxiras, Z.Hu, and M.Martonosi, "Cache Decay: Exploiting Generational Behavior to Reduce Cache Leakage Power," Proc. of the 28th Int. Symp. on Computer Architecture, pp.240-251, June 2001.
- [4] N.S.Kim, K.Flautner, D.Blaauw, and T.Mudge, "Drowsy Instruction Caches; Leakage Power Reduction using Dynamic Voltage Scaling and Cache Sub-bank Prediction," Proc. of the Int. Symp. on Microarchitecture, pp.219-230, Nov. 2002.
- [5] 小宮礼子, 井上弘士, 村上和彰 "待機ラインへの参照密度に基づく低リーク・キャッシュの動的制御", 情報処理学会研究報告, 2005-ARC-164, pp.73-78, Aug. 2005.
- [6] D. Parikh, Y. Zhang, K. Sankaranarayanan, K. Skadron, and M. Stan, "Comparison of State-Preserving vs. Non-State-Preserving Leakage Control in Caches," Workshop on Duplicating, Deconstructing and Debunking, pp.14-25, June 2003.
- [7] M. Powell, S. Yang, B. Falsafi, K. Roy, and T. N. Vijaykumar, "Gated-Vdd: A Circuit Technique to Reduce Leakage in Deep-Submicron Cache Memories," Int. Symp. on Low Power Electronic and Design, pp.90-95, July 2000.
- [8] SimpleScalarLLC, <http://www.simplescalar.com>
- [9] SPEC –Standard Performance Evaluation Corporation, <http://www.spec.org/>