

可変レベルキャッシュのマルチプロセッサ環境への適応と評価

城田 幸利

平成 22 年 2 月 23 日

近年、モバイルコンピューティングの分野において低消費エネルギーと高性能の両立が要求されている。そして、高性能化かつプロセッサの微細化、キャッシュ容量の増加に伴いプロセッサ内におけるキャッシュの消費エネルギーが増加している。そこで、キャッシュの消費エネルギーを削減する手法の一つとして可変レベルキャッシュが提案されている。可変レベルキャッシュとはキャッシュの容量を動的に変化させ、必要なときには通常の容量、必要でないときにキャッシュ容量の半分を Sleep モードに移行することによって、低消費エネルギー化を目指す手法である。可変レベルキャッシュはシングルプロセッサにおける消費エネルギー削減に成功している [1]。

しかし、現在主流なのはマルチプロセッサである。そこで、現在主流であるマルチプロセッサにおける可変レベルキャッシュの実装と評価を行った。今回使用したシミュレータはマルチプロセッサ環境に対応していなかったため、シミュレータをマルチプロセッサ環境に改良し、キャッシュは L1 分散 L2 共有として、L2 キャッシュに可変レベルキャッシュを適応した。評価により、従来手法そのままではマルチプロセッサアーキテクチャに適用した場合、性能の劣化する可能性があることが判った。そこで、キャッシュ容量切換え制御の新たな手法を提案し、実装評価を行った。その結果、提案手法はマルチプロセッサ環境において従来手法よりも、電力遅延積において最大 44% の改善された。

Recently, achieving both low energy and high performance are demanded in the field of a mobile computing. And, the consumption energy of cache in the processor is increasing along with high performance, miniaturization of the processor and an increase in cache capacity. Then,

Variable Level cache is proposed as one of the techniques for reducing the consumption energy of cache. Variable Level Cache is a technique for aiming at the low consumption energy by changing dynamically the capacity of cache. The half of cache capacity shifts to the Sleep mode when cache is not necessary. And, it shifts to usual capacity when cache is necessary. Variable Level Cache is that has succeeded in the consumption energy reduction in a single processor.

However, the main style is a multiprocessor now. Then, implementing and the evaluation of Variable Level Cache in the multiprocessor that is the main style now were done. Because the simulator that had been used did not correspond the multiprocessor environment, the simulator has been improved to the multiprocessor environment. And, the L1 cache of the simulator is decentralized cache. The L2 cache is shared cache that introduced Variable Level Cache. It has been understood the performance might deteriorated when applying to the multiprocessor architecture by the evaluation as it is conventional in the technique.

Then, it proposed a new technique to control the cache capacity change, and implementing and the evaluation was done. As a result, the proposal technique was improved from the conventional technique about 44% or less in the energy delay product in the multiprocessor environment.

1 序論

1.1 背景

近年、ノートパソコンやPDA、携帯電話などのモバイル端末の高性能化にともない消費電力が増大し、バッテリーによる駆動時間が短くなってきている。そこで、モバイル端末の性能を落とすことなく低消費電力を実現する事が要求されている。

プロセッサで消費される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に分けられる。動的消費エネルギーはトランジスタのスイッチングによって消費されるエネルギーである。一方、静的消費エネルギーはトランジスタの漏れ電流(リーク電流)によって引き起こされ、トランジスタのスイッチングに関係なく消費されるエネルギーで、リークエネルギーともいう。近年、回路の微細化にともなって、動的消費エネルギーが

削減される一方、リークエネルギーが増加している。リークエネルギーはトランジスタ数に比例するため、プロセッサの高性能化に伴ってキャッシュ容量が増大し、トランジスタが多いキャッシュメモリのリークエネルギーが問題になっている。

そこで、キャッシュメモリのリークエネルギーを削減する様々な手法が提案されている。その手法の一つとして可変レベルキャッシュがある。可変レベルキャッシュとはアプリケーションの負荷が低い場合にキャッシュ容量を小さくし、負荷が高い場合はキャッシュ容量を通常の容量に戻すことによって低消費電力を実現する手法で、シングルプロセッサ環境での低消費電力手法としての有効性は示されている。 [2]

2 可変レベルキャッシュ

2.1 概要

可変レベルキャッシュとはキャッシュへの要求性能に応じて動的にキャッシュ容量を切り換え、低消費エネルギーと高性能の両立を実現する手法である。概要図を図1に示す。キャッシュ容量が必要なときは通常のキャッシュ、キャッシュ容量が余り必要ないときは半分をスリープモードにする。スリープモードとは、電源の供給を完全に停止するのではなく、データの内容が破壊されない程度に電源電圧を下げた状態の事をいう。電源供給を完全に停止するよりはリークエネルギー効削減率が低くなるが、通常のキャッシュよりは大幅にリークエネルギーを削減できる。しかし、スリープモードのラインへのアクセスには、通常モードに戻した後にアクセスを行うので、通常のアクセスよりアクセス時間がかかってしまう。そこで、主記憶の上位に適用することにより、主記憶にアクセスするよりもアクセス時間は十分に短いため、性能の低下は押さえることができる。また、擬似的に1つ下位レベルの排他的キャッシュとすることでスリープモードでのキャッシュアクセスを極力減らし、消費電力を削減する。

すべてのラインがアクティブの時は「通常モード」、上記のようにキャッシュ容量の半分をスリープモードにし、排他的キャッシュとして動作する時は「低消費エネルギーモード」とする。

スリープモードから通常モードへ移行するにはキャッシュメモリ内のデータ配置が変わってしまうためメモリへの書き戻しが必要になる。

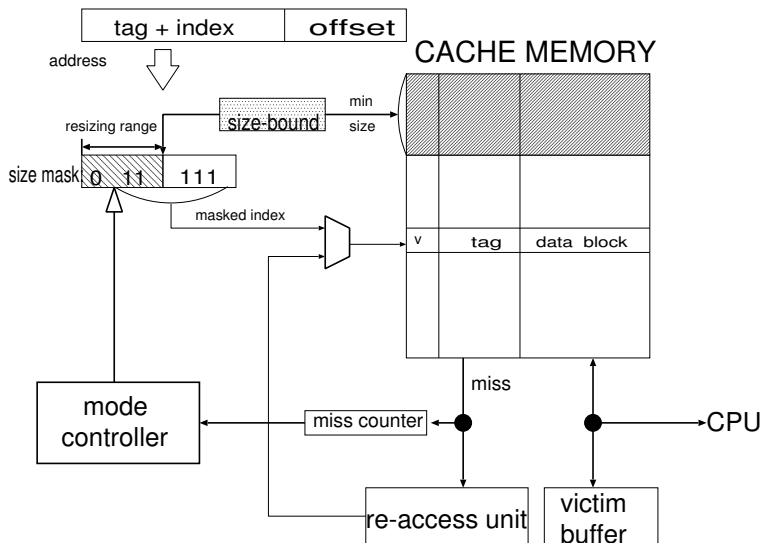


図 1: 可変レベルキャッシュ

2.2 電源供給停止とスリープモード

電源供給を停止する場合とスリープモードの場合の異なる点を上げる。

- 電源供給停止

- リークエネルギー 0
- 以前まで保持していたデータ破壊
- キャッシュミス回数の増加
- 性能低下

- スリープモード

- リークエネルギー微少
- データ保持
- アクセスする際、通常モードへ移行しなければならない
- 性能低下抑制

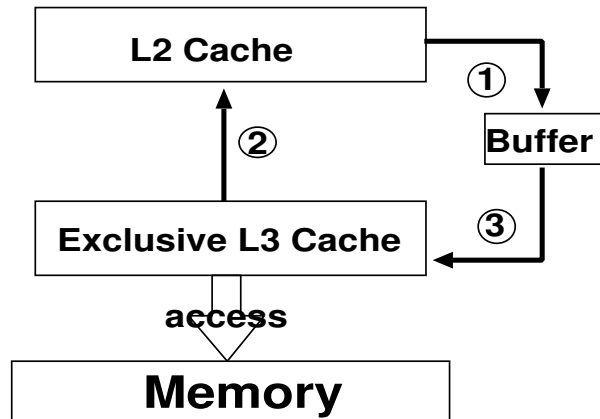


図 2: Exclusive Cache

2.3 排他的キャッシュ(ExclusiveCache)

低消費エネルギーモード時にはキャッシュ容量の半分を1つ下位レベルの排他的キャッシュ(ExclusiveCache)として動作させる。排他的キャッシュとはAMDの開発したキャッシュアーキテクチャである。従来のキャッシュの場合、L2キャッシュに格納されてからL1キャッシュへコピーされる。そのため、L1とL2の割り当てがあっても全体的なキャッシュサイズはL2キャッシュのサイズに相当する。しかし、排他的キャッシュでは、図2のように書き戻されるデータを一度バッファに移し、その後L3キャッシュから必要なデータをロードし、最後にバッファからL3キャッシュに書き戻しを行う。このように、データをL2とL3で交換する事でキャッシュの利用効率を高め、スリープモードへの連続アクセスを回避し、性能低下を防いでいる。

2.4 切替制御

可変レベルキャッシュでは、通常モードと低消費エネルギーモードがあり、この2つのモードを切替えることにより高性能と低消費電力の両立図を実現している。このモード切替制御を行うブロック図を図3に示す。

モード切替はある一定間隔毎にキャッシュミス数をカウントし、キャッシュミス率が閾値を下回った場合には低消費エネルギーモードに、上回った場合は通常モードへと遷移させる手法が提案されている [2]。しかし、

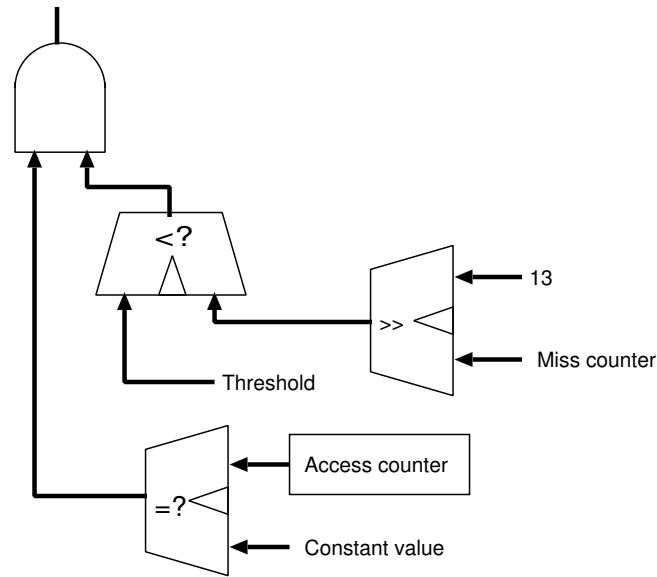


図 3: 切替制御のブロック図

この手法では除算を使用するため、ハードウェアコストが増大してしまう。そこで、同時にハードウェアコストを削減手法が提案されている。削減手法はモード判定を行うタイミングを2のべき乗にすることによってキャッシュミス率を出すとき、除算ではなく定数シフトすることによって求めることができるようになっている。

3 シミュレーターについて

文献 [2] では、可変レベルキャッシュの性能評価に Simple Scalar を用いている。Simple Scalar とは、シングルプロセッサのプロセッサシミュレータであり、実行速度は高速であるが、マルチプロセッサ環境でのシミュレーションを行うことができない。また、速度を求めるためにプログラムが複雑になってしまい、改良を行うことが困難である。そこで、本稿では東工大で開発された SimMips をベースにマルチプロセッサ環境対応に改良し、使用する。

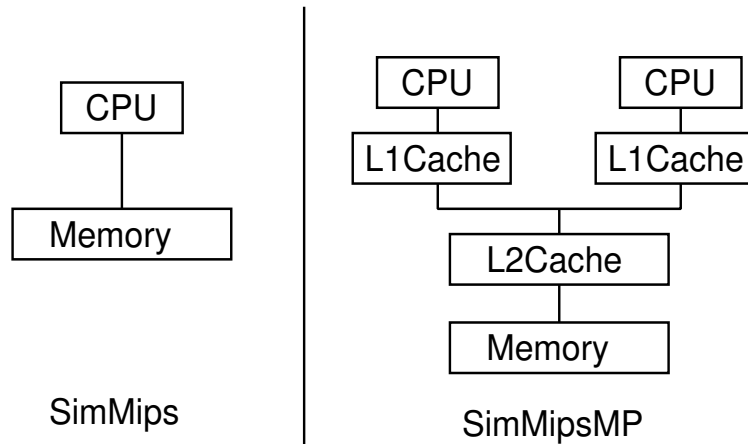


図 4: シミュレーター

3.1 SimMips

本稿で使用するシミュレータのベースとなるシミュレータである, SimMips[3] は東工大で開発された教育・研究において重要となるシンプルさや可読性を重視した MIPS32 命令セットのプロセッサを含むシミュレータである. SimMips はシングルプロセッサのシミュレータであり, マルチプロセッサに対応しておらず, 更にキャッシュも搭載していない.

3.2 SimMipsMP

本稿で使用するシミュレータで SimMips をマルチプロセッサ対応に改造した上で, 独立型 L1 キャッシュと共有型可変レベル L2 キャッシュを実装した. 概要図を図 4 に示す. データの書き戻しには L1 にライトスルー, L2 にはライトバックとし, コヒーレンスは有効無効の 2 状態のみで制御を行う.

4 マルチプロセッサへの可変レベルキャッシュの適用

前述の通り, 可変レベルキャッシュはこれまでシングルプロセッサ環境でのみ評価されてきた. そこで, まず可変レベルキャッシュをマルチプロ

キャッシュ容量	
L1 命令-cache	32KB(64B/entry, 1way, 512entry)
L1 データ-cache	32KB(64B/entry, 2way, 256entry)
L2 cache	512KB(64B/entry, 4way, 2048entry)
ヒット・レイテンシ	
L1 cache	1 cycle
L2 cache	16 cycle
主記憶	250 cycle
モード切換のオーバーヘッド	
レイテンシ	5cycle

表 1: SimMipsMP のキャッシュに関するパラメータ

セッサ環境に適用し、その有効性を調査する。可変レベルキャッシュについて、性能の評価を行うプロセッサ構成は、表 1 に示す。

可変レベルキャッシュは L2 共有型キャッシュに実装し、512KB の L2 キャッシュとして扱う通常モードと、256KB の L2 キャッシュと 256KB の L3 排他的キャッシュとして扱う低消費エネルギーモードの 2 種類を動的に切替える。モード切換は 8192 回のキャッシュアクセス毎にキャッシュミス回数をシフト演算を用いてキャッシュミス率を測定し、閾値より下だったら低消費エネルギーモード、閾値より上だったら通常モードで動作する。閾値は文献 [2] と同じパラメータである 25% を用いた。

4.1 第一次評価

通常キャッシュと可変レベルキャッシュについて実行時間の評価を行う。ベンチマークプログラムは SPEC2000 より、SPECint2000 から 164.gzip, 175.vpr, 181.mcf, 256.bzip2 の 4 種類、SPECfp2000 から 183.quake, 188.ammp の 2 種類、計 6 種類を使用し SimMipsMP で全パターンシミュレーションを行った。結果を図 5 に示す。

「normal」が通常キャッシュ、「level」が可変レベルキャッシュである。結果、ベンチマーク全体の半分において性能の悪化が見られた。これは、可変レベルキャッシュは、キャッシュアクセスの局所性が高ければ、キャッシュヒット率が高くなるので低消費エネルギーモードで動作することが多く、局所性が低ければ通常モードで動作することが多くなるという特

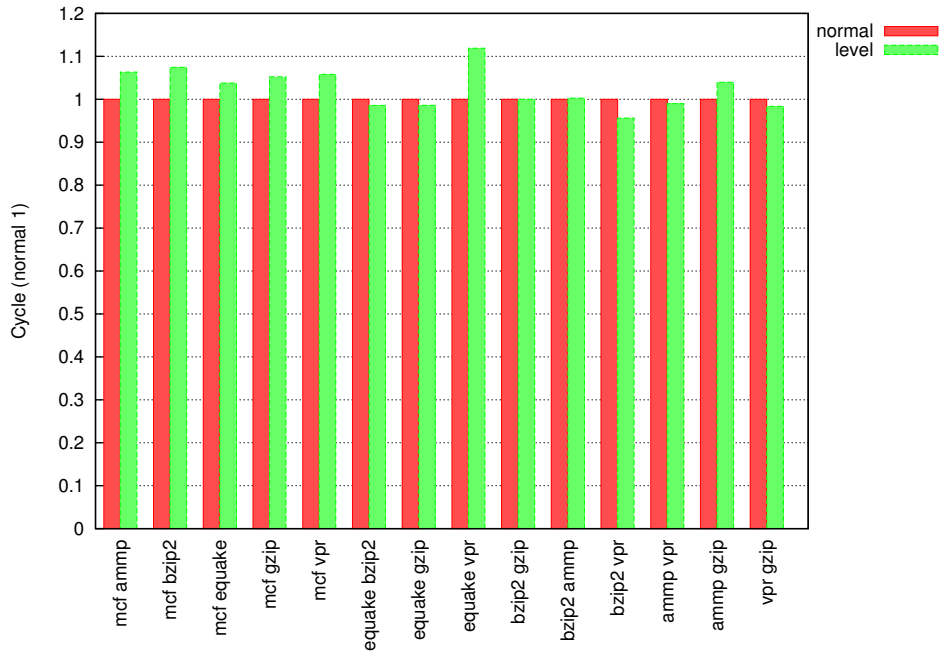


図 5: 実行時間

性をもつ。マルチプロセッサ環境の場合、各プロセッサで実行中のプログラムが必要とするキャッシュ必要容量が違うため、L2 キャッシュにアクセスが多い方のプログラムが局所性が高い場合、もう一方の局所性が低い場合でも低消費エネルギーモードになってしまう。また、それでミス率が高くなるとモード切替が起こり、モードの切替が多発してしまう。上記のことが悪化の原因だと考えられる。

5 マルチプロセッサ向けモード切替手法の提案

本節では、マルチプロセッサ向けのモード切替手法を提案を行う。

5.1 モード切替手法の問題点

モード切替は可変レベルキャッシュの性能を左右する重要な部分である。上記の通り、可変レベルキャッシュのモード切替はある一定間隔毎にキャッシュミス率によって切替判定を行っている。よって、マルチプロ

	Core2	25%以下	26%以上
Core1			
25%以下		低消費エネルギーモード	状態維持
26%以上		状態維持	通常モード

表 2: 状態遷移

セッサに適応した場合，L2 共有に可変レベルキャッシュを適応しているため，CPU 全体のキャッシュミス率でモード切替を判定してしまう．しかし，各プロセッサに走るプログラムによってキャッシュ必要容量が異なるため，アクセスが多いプログラムに比重が偏ってしまい柔軟に対応できない．

5.2 改良型モード切替手法の提案

そこで，マルチプロセッサ環境においても各プロセッサでキャッシュミス率を求めることにより性能悪化を低減する手法を，モード切替判定を行う間隔が異なる 2 種類提案する．

上記の通り，性能が悪化する原因は各プロセッサのプログラムのキャッシュ必要容量に対応できていないためである．よって，各プロセッサのキャッシュ必要容量に対応できるようにキャッシュミス率を各プロセッサでとる．表 2 のように全プロセッサでキャッシュミス率が閾値を下回った場合のみ低消費エネルギーモードに切替，逆に全プロセッサで上回れば通常モードに切替を行うことで性能悪化を低減させる．なお片方が上回り，片方が下回る場合は状態を維持する．モード切替間隔は，図 6 のように各プロセッサの可変レベルキャッシュに対する一定アクセス毎、別々に閾値判定を行うものと，図 7 のように従来の可変レベルキャッシュと同じように可変レベルキャッシュに対する一定アクセス毎に判定を行い，同時に各プロセッサの閾値判定を行うもの 2 種類の手法で性能評価を行った．なおこれ以降提案手法 1 を「separate」，提案手法 2 を「integrate」とする．

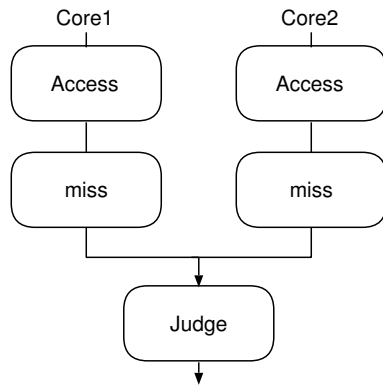


図 6: separatea フローチャート

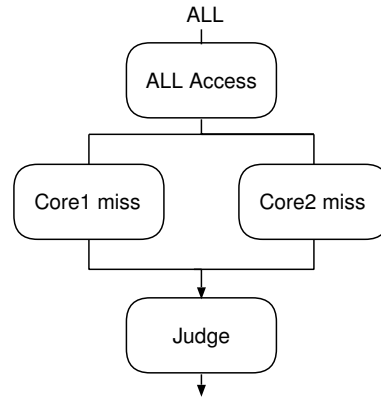


図 7: integrate フローチャート

6 消費エネルギーと電力遅延積

6.1 電力遅延積

本稿では、性能の維持と低消費エネルギーの両立を目的としているため、性能の指標として電力遅延積を使用する。電力遅延積とは消費エネルギー E と実行時間 T を掛けることによって求めることができる。電力遅延積が小さいほど低消費電力と高性能の両立が達成できていることを示している。

6.2 消費エネルギー

消費エネルギーについては、文献 [4][6] を参考に以下のように近似した。

まず、キャッシュアクセスによる動的エネルギーの総和 DE_{total} はラインアクセス当りの平均動的エネルギー DE_{line} とアクセス回数 $Access$ の和で求められる。よって近似式は

$$DE_{total} = DE_{line} \times Access \quad (1)$$

となる。

次にリークエネルギー LE_{total} は 1 クロックサイクルで消費するライン当りの平均リークエネルギー LE_{line} に総ライン数 $CSize$ とプログラム実行クロックサイクル数 CC の積で求まる。よって近似式は

$$LE_{total} = CC \times LE_{line} \times CSize \quad (2)$$

式(2)の LE_{line} は、キャッシュメモリ全体のうちスリープモードとなっているラインの割合 SR とスリープモードのライン当たりのリークエネルギー LE_{sline} の積に、通常モードのラインの割合 $(1 - SR)$ と通常モードのライン当たりのリークエネルギー LE_{aline} の積を加えたものであるため、以下のように

$$LE_{line} = SR \times LE_{sline} + (1 - SR) \times LE_{aline} \quad (3)$$

となる。スリープモード時に L3 キャッシュにアクセスする場合、通常モードに切り替える必要がある、L3 キャッシュアクセスの際、キャッシュラインをスリープモードから通常モードに切り替えるエネルギーは

$$CE_{total} = CE_{line} \times BSize \times Access_{sline} \quad (4)$$

と表される。 CE_{line} はキャッシュライン当たりのモード切替エネルギーであり、 $BSize$ はモードを切り替えるライン数(バンクの大きさ)、 $Access_{sline}$ は通常モードへの切替回数、つまり可変レベルキャッシュにおけるスリープモード時の L3 キャッシュに当たる部分へのアクセスの回数である。

よって、キャッシュ全体の消費エネルギー E_{total} は、式(1)、式(2)、式(4)の和、すなわち、

$$E_{total} = DE_{total} + LE_{total} + CE_{total} \quad (5)$$

となる。

尚、通常キャッシュや DRI キャッシュでは、スリープモードのラインは存在しないため、 $CE_{total} = 0$ となる。

キャッシュのエネルギーを評価する値は Cacti[7] を用い、32nm プロセスを想定し、ラインサイズを 64B として求めた、求めた値を以下に示す。

$$DE_{line} = 2.20E - 10(J) \quad (6)$$

$$LE_{aline} = 5.39E - 15(J) \quad (7)$$

$$LE_{sline} = 8.61E - 16(J) \quad (8)$$

$$CE_{line} = 1.92E - 16(J) \quad (9)$$

以上の式で求めた、実行サイクル数と消費エネルギーの積を求め比較を行う。

7 性能評価

提案した手法を SimMipsMP に実装し、先ほどと同じ SPEC のベンチマークプログラムをシミュレーションを行った。評価指標は前述で示した電力遅延積で行う。

7.1 評価結果

実験によって得られた結果を得られた結果を図 8 に示す。「normal」が通常キャッシュ、「level」が従来の可変レベルキャッシュ、「separate」と「integrate」が提案手法である。結果を見ると、separate において従来の可変レベルキャッシュよりも結果が悪いケースがある。これは、従来手法の時には通常モードの動作よりも、低消費エネルギーモードの方が割合が高かったが、separate では、通常モードのほうが動作の割合が高くなっているからだと考えられる。また、separate より integrate の方が結果がいいのは、全体のアクセスによってモード切換を判定しているため、separate よりも細粒度にモード切換を行っているためである。

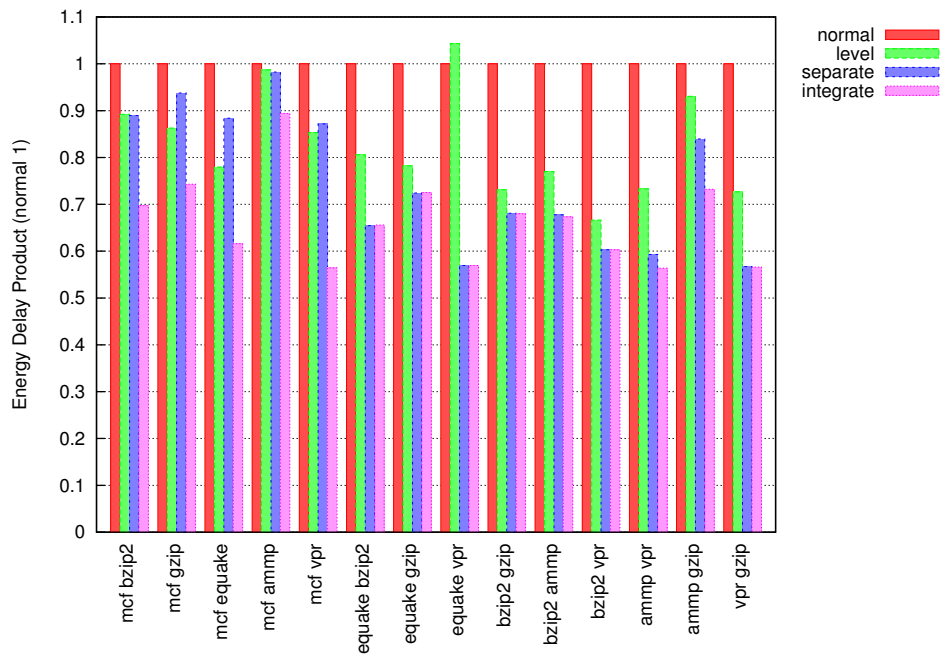


図 8: 電力遅延積

8 結論

本稿では、可変レベルキャッシュをマルチプロセッサ環境へ適応し評価を行った。結果、可変レベルキャッシュの従来のモード切替では性能悪化が見られたが、モード切替を各プロセッサでキャッシュミス率とることにより、電力遅延積において separate で従来の可変レベルキャッシュのモード切替よりも平均10%程度、integrate では平均15%程度改善が見られた。

今後の課題として、当研究室で研究されている VSP (Variable Stages Pipeline) と合わせたシミュレーション評価、または、更にプロセッサ数を増やした場合の評価などが考えられる。

9 謝辞

本研究を行うにあたり、ご指導、ご助言いただき下さいました佐々木敬泰助教、並びに多くの助言をいただきました近藤利夫教授、大野和彦講師に深く感謝いたします。また、様々な局面にてお世話になりました計算機アーキテクチャ研究室の皆様にも心より感謝いたします。

参考文献

- [1] 恩賀琢也, 佐々木 敬泰, 大野 和彦, 近藤 利夫, “キャッシュ階層動的切り替えによる低消費電力化”, 情処学研報, 2007-ARC-174, pp.115-120. August 2007
- [2] 松原 伸幸, 佐々木 敬泰, 大野 和彦, 近藤 利夫, “高性能かつ低消費電力を実現する可変レベルキャッシュのモード切替アルゴリズムの改良と評価”, 信学会技報, CPSY2009-44, pp.7-12, December 2009
- [3] 藤枝直輝, 渡邊伸平, 吉瀬謙二, ” S i m M i p s : 教育・研究に有用なL i n u xが動く5000行のM I P Sシステムシミュレータ”, 情報処理学会シンポジウム論文集, Y0978B, pp.143-150, December 2009
- [4] 小宮礼子, 井上弘士, モシニヤガ・ワシリー, 村上和彰, “キャッシュ・リーク電力削減アルゴリズムに関する定量的評価”, 第17回回路とシステム軽井沢ワークショップ論文集, pp.235-240, April 2004.
- [5] “SPEC -Standard Performance Evaluation Corporation-,” URL: <http://www.spec.org/>.
- [6] 関子純平, 富山宏之, 高田広章, 井上弘士, “Drowsy キャッシュにおけるモード切替アルゴリズムの評価”, 情処学研報, 2006-ARC-170, pp.37-41, December 2006.
- [7] CACTI 5.1 Shyamkumar Thoziyoor, Naveen Muralimanohar, Jung Ho Ahn, and Norman P. Jouppi