

RC-006

ウェイアロケーション型共有キャッシュ機構のハードウェア設計に関する研究 Hardware Design of A Way-Allocatable Shared Cache Mechanism

阿部 健太* 小寺 功* 江川 隆輔*† 滝沢 寛之* 小林 広明*†
Kenta Abe Isao Kotera Ryusuke Egawa Hiroyuki Takizawa Hiroaki Kobayashi

1 緒言

近年、一つのチップ上に複数のコアを搭載したチップマルチプロセッサ (Chip Multiprocessor, CMP)[1] がマイクロプロセッサの主流となってきている。CMP では、スレッドレベル並列性を利用し、複数のスレッドをチップ上の異なるコアで並列に処理することによって、プロセッサの実行効率を高めている。一般的な CMP では、搭載された複数のコア間でキャッシュメモリを共有している。そのために、同時実行するスレッド間で共有キャッシュの競合が発生し、スレッドを単独で実行する場合と比較して、各コアの実効性能が不均一に低下するという問題がある。また、半導体加工技術の進歩に伴いプロセッサの消費電力及び電力密度がともに増加傾向にある。将来の半導体テクノロジーでは静的な消費電力が支配的になる事が予想されており、チップ上で大きな面積を占める共有キャッシュメモリは膨大な静的電力を消費する。従って低消費電力、かつ高性能なマイクロプロセッサ設計のためにはキャッシュメモリの消費電力の削減が強く求められている。

共有キャッシュの競合による性能低下と消費電力の問題を同時に解決するための手法として、ウェイアロケーション型共有キャッシュ [2] が提案されている。この手法は、CMP の各コアがアクセスできる共有 L2 キャッシュの範囲を決めることで各コアの性能の不均一な低下を防ぐウェイアロケーション機能と、各コアでスレッドが必要としているキャッシュ容量を適切に予測し、キャッシュへの電力供給をウェイ単位で動的に制御できる電力制御機構 [3] から構成される。小寺ら [2] は、CMP 全体の实効性能を維持しつつ、L2 キャッシュの消費エネルギーの 20% 削減を達成している。

ウェイアロケーション型共有キャッシュ機構の制御を行うためには追加ハードウェアが必要であるが、文献 [2] の結果にはハードウェアの面積及び遅延時間が考慮されていない。また、ウェイアロケーション型共有キャッシュ機構の制御には除算が必要であるが、除算器は一般的に他の演算回路と比べて回路規模と遅延時間が大きい事が知られている。本論文では、ウェイアロケーション型共有キャッシュ機構の実装に必要な追加ハードウェアを設計し、その面積・遅延時間を評価する。また、回路規模の抑制、遅延時間の短縮を目的として、除算器を用いない制御機構を提案する。両手法の処理性能・電力削減効果、および追加ハードウェア量を評価し、従来の手法との比較から、提案手法の長所と短所を明らかにする。

2 ウェイアロケーション型共有キャッシュ機構

2.1 ウェイアロケーション型共有キャッシュ機構の動作

ウェイアロケーション型共有キャッシュ機構の概念図を図 1 に示す。本機構は 2 コアの CMP を想定している。ウェイアロケーション型共有キャッシュ機構は、ウェイアロケーション機能と電力制御機能の 2 つから構成される。ウェイアロケーシ

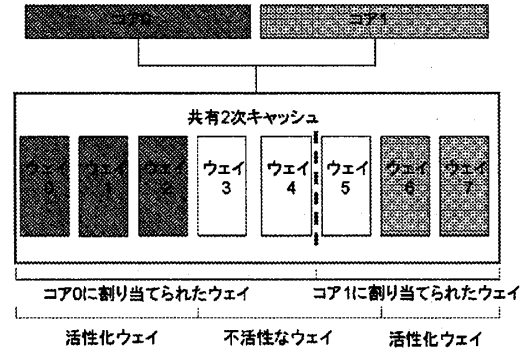


図 1 ウェイアロケーション型共有キャッシュ機構の概念図

ョン型共有キャッシュ機構では、はじめに、キャッシュへのアクセスのサンプリングを行い、キャッシュの利用状況の統計情報を得る。その情報を基に局所性評価量 D を計算し、この D によってキャッシュアクセスの局所性を評価し、ウェイアロケーションと電力制御を行う。ウェイアロケーション機能では、大容量の共有 L2 キャッシュのウェイを各コアに専用のキャッシュメモリとして動的に割り当てる。また、割り当てられた領域の全容量を必要としない場合には、電力制御機能によって不必要なウェイ数を評価し、ウェイへのパワーゲーティングにより電力供給を止める (不活性化)。この 2 つの制御機能により、実効性能を維持したままプロセッサの動的/静的電力の消費を削減する。このような制御を、一定のキャッシュアクセス数ごとに行い、ウェイアロケーション及び電力制御をアプリケーションの振る舞いに動的に対応させる。

2.1.1 局所性評価量

キャッシュアクセスの時間的局所性が高い場合、LRU (Least Recently Used) ブロックへのアクセスは少なく、MRU (Most Recently Used) ブロックへのアクセスは多い。このときはキャッシュの利用範囲が狭く、ウェイ数を減らしても性能の低下が少ないと予測できる。逆に局所性が低い場合、LRU ブロックへのアクセスが多く、キャッシュを広範囲に利用しており、より多くのキャッシュを必要としていると予測できるため、ウェイ数を増加させると高い性能向上が得られると期待できる。そこで、制御の指標である局所性評価量を、MRU ブロックへのアクセス数に対する LRU ブロックへのアクセス数の割合を用いて次の式で定義する [3]。

$$D = \frac{LRUcount}{MRUcount} \quad (1)$$

$LRUcount$ は LRU ブロックへのアクセス数、 $MRUcount$ は MRU ブロックへのアクセス数である。これにより、 D が小さいときには、キャッシュアクセスの時間的局所性が高く、 D が大きいときには、キャッシュアクセスの時間的局所性が低いと判断できる。

* 東北大学大学院情報科学研究科

† 東北大学サイバーサイエンスセンター

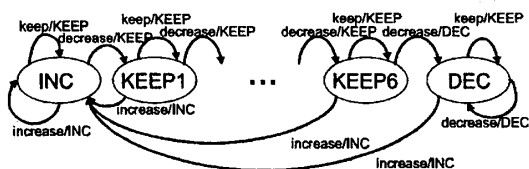


図2 ステートマシンの状態遷移図

2.1.2 ウェイアロケーション機能

ウェイアロケーション機能では、各コア毎に算出された D の大小を比較し、 D が小さいコアに割り当てられるウェイ数を1つ減らし、 D が大きいコアに割り当てられるウェイ数を1つ増やす。 D が等しいときには、キャッシュ要求量が等しいので、現在割り当てられているウェイ数を維持する。

2.1.3 電力制御機能

電力制御機構は、アプリケーションが今必要とするキャッシュ容量を予測する局所的判断と、過渡的な要求に過剰に反応しないよう全体的な傾向を判断する大域的判断からなる。局所的な判断のみでは、例えばアクセスの分布が急激に変化するアプリケーションの場合、キャッシュの活性化ウェイ数が短期間に増減を繰り返すことになり、性能の低下につながる。ステートマシンを用いた大域的判断を導入することで、このような問題を回避し、安定した制御を行うことが可能となる。

局所的判断では、局所的なキャッシュの必要容量を予測するために D を使用する。 D と閾値 $t1, t2$ ($t1 < t2$) の大小を比較してウェイの活性化・不活性化を判断する。 D が $t1$ 以下の場合、キャッシュアクセスの時間的局所性が高く、キャッシュ容量を減らしても性能への影響は小さいため、1ウェイを不活性化すべきと判断する。 D が $t2$ を超えている場合は、キャッシュアクセスの時間的局所性が低く、より多くのキャッシュ容量を必要としているため、不活性化ウェイを1つ活性化すべきと判断する。 D が $t1$ より大きく $t2$ 以下のときには、活性化ウェイ数の増減は行わない。閾値が大きくなるほど、活性化ウェイ数を減らすべきと判断する範囲が大きくなる。逆に閾値が小さいほど、活性化ウェイ数を増やすべきと判断する範囲が大きくなる。そのため、閾値は大きいほど省電力指向の閾値となり、小さいほど性能指向の閾値となる。

大域的な判断に用いるステートマシンの状態遷移図を図2に示す。INCを増加状態、DECを減少状態、KEEPを維持状態とする。また、 D と閾値の比較によるウェイ数の増減判定結果 increase, keep, decrease を入力とする。例えば、減少判定 (decrease) を入力とした状態遷移後の出力が KEEP の場合、decrease/KEEP と表記する。局所的判断において活性化ウェイ数を増やすべきと判断された場合、現在の状態に関わらず状態は INC に遷移し、活性化ウェイ数を増加させる。逆に減らすべきと判断された場合、次の状態に遷移し、状態 DEC に移動するときのみ活性化ウェイ数を減らし、それ以外の場合は活性化ウェイ数を変更しない。局所的判断において増減を行うべきでないと判断された場合は、現在の状態を維持し、活性化ウェイ数を変更しない。

2.1.4 ハードウェア設計

ウェイアロケーション型共有キャッシュ機構の制御では、 D の計算、ウェイアロケーション、電力制御の局所的判断、大域的判断を行う。 D の計算を行うためには、除算器が必要である。ウェイアロケーションを行うには、各コアの D の大小を

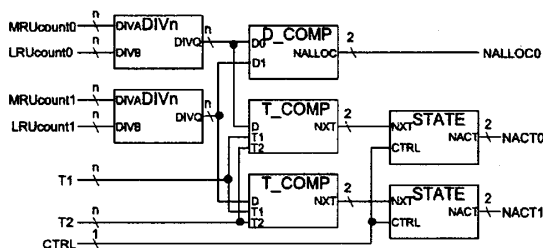


図3 追加ハードウェアのブロック図

比較し、各コアに新たに割り当てられるキャッシュのウェイ数の増減を出力する比較器が必要である。電力制御の局所的判断を行うには、 D と閾値の大小を比較し、活性化ウェイ数の増減の局所的判断を出力する比較器が必要である。電力制御の大域的判断には、ステートマシンが必要である。

本論文で設計するウェイアロケーション型共有キャッシュ機構のために、従来の L2 キャッシュに追加するハードウェアのブロック図を図3に示す。DIVは除算器、D_COMP及びT_COMPは比較器、STATEはステートマシンを示す。DIVでは D の計算を、D_COMPではウェイアロケーションを、T_COMPでは電力制御の局所的判断を、STATEでは大域的判断を行う。また各演算器の左側が入力、右側が出力となる。

ハードウェア全体への入力は、各コアの $MRUcount$ と $LRUcount$ 、閾値 $t1, t2$ 、ステートマシンの制御信号 CTRL であり、出力は、コア0に割り当てられるウェイ数の増減を示す NALLOC0 と、各コアの活性化ウェイ数の増減を示す NACT0 及び NACT1 である。まず各コアの $MRUcount$ と $LRUcount$ を DIV に入力し、各コアの D を計算する。次にコア0とコア1の D を D_COMP にて比較し NALLOC0 を計算する。同時に各コアごとに D と閾値 $t1, t2$ を T_COMP にて比較しその結果を STATE へ入力する。最後に STATE にて NACT を決定し、出力する。

ウェイアロケーション型共有キャッシュ機構では、L2 キャッシュへのアクセス数が $A(= 2n)$ に達するまで MRU ブロックと LRU ブロックへのアクセスを数え、アクセス数が A に達した時点でウェイアロケーションと電力制御を行う。

3 LRU アクセス数に基づくウェイアロケーション機構

除算を用いる D の導出では、ハードウェア量の増加が予想される。そこで、本論文では除算器を用いずに、 D 値の代わりに LRU アクセス数を用いるウェイアロケーション手法を提案する。

3.1 LRUcountによる時間的局所性予測

$LRUcount$ が小さく $MRUcount$ が大きい場合には時間的局所性が高く、 $LRUcount$ が小さく $MRUcount$ も小さい場合は、キャッシュへのアクセス数が少ないと考えられる。従って、 $LRUcount$ が小さい場合、キャッシュのウェイ数を減らしても性能に与える影響は小さいといえる。次に、 $LRUcount$ が大きい場合は、 $MRUcount$ の大小に関わらず、キャッシュを広範囲に利用しており、ウェイ数を増やすことで性能の向上が期待できる。以上より、 D の代わりに $LRUcount$ に基づいて、キャッシュアクセスの時間的局所性を予測可能といえる。

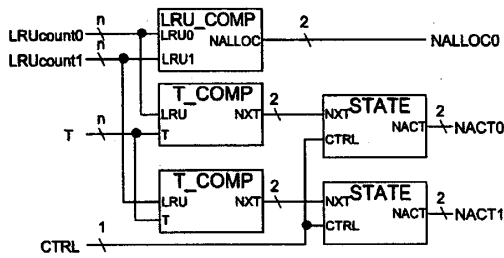


図4 LRUcountを用いたハードウェアブロック図

3.2 提案手法の動作

提案手法では、アクセスサンプリングの後、 D の計算を行わず、 $LRUcount$ によりウェアラケーションと電力制御を行う。ウェアラケーション機能では、各コアの $LRUcount$ の大きさを比較し、 $LRUcount$ が大きいほうのコアのウェイ数を1つ増やす。電力制御の局所的判断では、 $LRUcount$ と閾値 T の大きさを比較を行い、 $LRUcount$ が T 以上の場合、活性化ウェイ数を1つ増やすべきと判断し、 T よりも小さい場合、活性化ウェイ数を1つ減らすべきと判断する。

閾値は経験的に決定されるものであり、本論文の従来手法の閾値として、文献 [2] で使用されている閾値を用いている。提案手法では、ハードウェアコスト削減のため、閾値を1つとした。閾値を1つとすることで、T_COMPとSTATEのハードウェアの削減が可能となる。

3.3 ハードウェア設計

図4に、 D の代わりに $LRUcount$ を用いたハードウェアのブロック図を示す。従来手法と比較して、除算器がなくなり、D_COMPの代わりにLRU_COMPが用いられている。ハードウェアへの入力は $LRUcount_0$ 、 $LRUcount_1$ 、閾値 T 、ステートマシンの制御信号CTRLであり、出力はNALLOC0、NACT0、NACT1である。LRU_COMPでは、ウェアラケーションするために、各コアの $LRUcount$ の大きさを比較する。T_COMPでは、 $LRUcount$ と閾値の大きさを比較することで電力制御の局所的判断を行う。その結果をSTATEに入力し、電力制御の大域的判断を行いNACTを決定する。

本論文では、従来の手法をD法、提案手法をLRU法と呼ぶことにする。

4 評価

4.1 ハードウェア量と遅延時間

$MRUcount$ および $LRUcount$ のカウンタのサイズをそれぞれ8から20ビットまで、すなわち、 A を 2^8 から 2^{20} まで変化させたときのD法と、 A が 2^{20} のときのLRU法の実装に必要なハードウェア量、遅延時間を評価し、ウェアラケーション型共有キャッシュ機構の有効性を評価する。

ハードウェアの記述にはVHDLを用いる。また、シミュレータとしてSynopsys Scirocco、論理合成にはSynopsys Design Compiler、配置・配線にはSynopsys Astroを用いる。論理合成にはRohm社から提供されている $0.18\mu\text{m}$ CMOSテクノロジーのセルライブラリを使用し、特に論理合成時の制約条件は与えていない。

設計したD法のハードウェアの遅延時間及び面積を図5に示す。図5の縦軸は遅延時間と面積、横軸はアクセスサンプリング数 A のビット幅である。アクセスサンプリングにかかる時間とハードウェアの遅延時間を比較すると、ハードウェアの

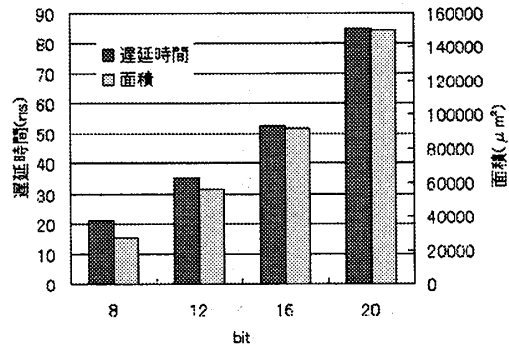


図5 D法を設計したハードウェアの遅延時間及びハードウェア量

遅延時間は、20ビットでも85nsであり、十分に小さいといえる。また、CACTI[4]によって得られたキャッシュ面積(1MB, 32ウェイ)は 152.95mm^2 である。この値とハードウェアの面積を比較すると、20ビットでもハードウェア面積はキャッシュ面積の0.1%未満と十分に小さい。

また、LRU法のハードウェアの場合、 A が 2^{20} のときの遅延時間は6.17ns、面積は $7147\mu\text{m}^2$ で、これらの値はD法のハードウェアのそれぞれ7.26%、4.77%に相当する。これにより、 $LRUcount$ に基づくウェアラケーション機構によって、ハードウェア量と遅延時間を大幅に削減できることが明らかになった。

4.2 実効性能

4.2.1 実験方法

本実験では、最初にLRU法とD法に基づくウェアラケーション機能によって得られる実効性能を評価する。次に、LRU法およびD法における実効性能と消費電力の関係から、両者の電力制御機構の有効性を明らかにする。

以下の実験では、CMPの実効性能を、Weighted-Speedup(WS)により評価する。CMPで複数のアプリケーションを同時実行するとき、コア i の1クロックサイクルあたりの実行命令数(Instructions Per Cycle, IPC)を IPC_i とする。また、同じCMPでアプリケーションを単独で実行するときのIPCを $SingleIPC_i$ とする。このときWSは次式で定義される。

$$WeightedSpeedup = \sum \frac{IPC_i}{SingleIPC_i} \quad (2)$$

また、プロセッサの実効性能の評価にはM5シミュレータ [5]を用い、消費電力にはM5の結果を基にCACTIを用いて評価する。主なシミュレーションパラメータを表1に示す。2コアのCMPを用い、ベンチマークとしてSPEC CPU2006[6]から5つのアプリケーションを選び、各コアで1アプリケーションずつ実行する。

また、アクセスサンプリング数 $A = 10^5$ とする。D法では、閾値($t1, t2$)として(0.001, 0.005), (0.01, 0.05), (0.1, 0.5)を用いる。(0.001, 0.005)が高性能指向の閾値、(0.1, 0.5)が省電力指向の閾値である。D法とLRU法の省電力能力の比較のため、LRU法では、D法の実効性能に最も近いWSが得られる閾値として $T = 100$ を選択し、そのときのLRU法とD法の消費電力を比較する。

4.2.2 実験結果

LRU法およびD法によって達成されたWSを図6に示す。図6のLRUがLRU法、DがD法の値である。値はウェイア

表1 シミュレーションパラメータ

パラメータ	値
フェッチ幅	8 insts
デコード幅	8 insts
発行幅	8 insts
コミット幅	8 insts
命令キュー	64 insts
LSQ サイズ	32 entries
L1 命令キャッシュ	32kB, 2-way, 32B-line 1 cycle latency
L1 データキャッシュ	32kB, 2-way, 32B-line 1 cycle latency
L2 共有キャッシュ	1024kB, 32-way, 64B-line 14 cycle latency
主記憶	100 cycle latency
動作周波数	1GHz
テクノロジー	70nm
Vdd	0.9V

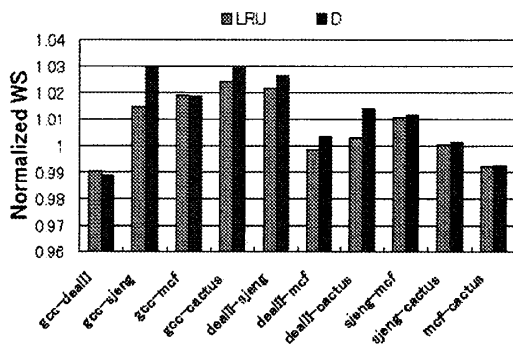


図6 LRU法とD法のWSの比較

ロケーション機能を持たないキャッシュの値で正規化されている。図6の結果より、多くのアプリケーションの組み合わせでLRU法はD法と同等のWSを達成できている。この結果より、LRU法でもキャッシュアクセスの局所性を評価できていることがわかる。ただし、いくつかの組で、LRU法の性能がD法と比較して大きく低下している。これは、LRU法の閾値の設定が、その組に対して適切ではないためである。D法で用いられているDはLRUcountとMRUcountの比であるのに対して、LRU法ではLRUcountを評価量として用いており、適切な閾値はアプリケーションの組み合わせに大きく依存する。従って、LRU法ではハードウェアコストを大幅に削減できる一方、適切な閾値の設定方法にはさらなる改善が必要であるといえる。

次に、gccとmcfを同時に実行した際の、WS及び消費電力を表2に示す。表2の値はウェイアロケーション機能を持たないキャッシュの値で正規化されている。各閾値において、D法とLRU法の双方で、同等のWSに対し同等の消費電力を得られた。また、他のアプリケーションの組み合わせにおいても、D法とLRU法の双方で、同等のWSに対し同等の消費電力を得られた。このことから、LRUcountを用いる手法でも、閾値を適切に設定すればDを用いる手法と同等のウェイアロケーションが可能であるといえる。

表2 D法, LRU法による性能と消費電力 (gcc-mcf)

		省電力指向 ←→ 高性能指向		
D法	WS	0.8609	0.9958	1.0185
	Energy	0.5099	0.6531	0.9323
LRU法	WS	0.8485	0.9986	1.0188
	Energy	0.4694	0.6388	0.9122

5 結言

本論文では、ウェイアロケーション型共有キャッシュ機構の実装に必要な追加ハードウェアを設計し、そのハードウェア量が十分小さいことを示した。その結果、ウェイアロケーション型共有キャッシュ機構はハードウェアの面積及び遅延時間を考慮しても有効な制御が可能であることが明らかになった。また、ウェイアロケーション型共有キャッシュ機構の更なる高性能化を目的として除算器を必要としない局所性評価量を提案し、ハードウェア量とプロセッサの実行性能を評価した。提案するLRU法に必要なハードウェア量は、D法と比べ大幅に削減できることを示した。また、LRU法は適切な閾値を設定することでD法と同等のウェイアロケーション性能を得られる可能性を示した。しかし、閾値の設定はD法と比較して困難である。今後の課題として、LRU法で用いる閾値の定量的な評価が挙げられる。

謝辞

本論文の執筆にあたり、ご協力頂いた東北大学船矢祐介氏、佐藤雅之氏に感謝致します。本研究の一部は、文部科学省科学研究補助金基盤研究(B)(課題番号18300011)による。本研究は東京大学大規模集積システム設計教育研究センターを通じ、シノプシス株式会社の協力で行われたものである。

参考文献

- [1] Nayfeh B., Olukotun K., "A single-chip multiprocessor," IEEE Computer, Vol.30, No.9, pp.79-85, Sept 1997.
- [2] Kotera I., Egawa R., Takizawa H. and Kobayashi H., "A Power-Aware Shared Cache Mechanism Based of Locality Assessment of Memory Reference for CMPs," Presented at ACM PACT07 Workshop on Memory Performance, Dealing with Applications, Systems and Architectures (MEDEA '07), Session IV-4, in Brasov Romania, September 16, 2007.
- [3] Kobayashi H., Kotera I., Takizawa H., "Locality analysis to control dynamically way-adaptable caches" SIGARCH Comput. Archit. News, Vol.33, No.3, pp.25-32, 2005.
- [4] Wilton S., Jouppi N., "Cacti: an enhanced cache access and cycle time model," IEEE Journal of Solid-State Circuits, Vol.31, No.5, pp.677-688, May 1996.
- [5] Binkert N., Dreslinski R., Hsu L., Lim K., Saidi A., Reinhardt S., "The m5 simulator: Modeling networked systems," IEEE Micro Vol.26, No.4, pp.52-60, July-Aug 2006.
- [6] "The Standard Performance Evaluation Corporation," <http://www.spec.org/>.