

マルチコアプロセッサ向けオンチップキャッシュにおける配線遅延の影響

伊藤, 義崇
九州工業大学大学院情報工学研究科

千代延, 昭宏
九州工業大学大学院情報工学研究科

佐藤, 寿倫
九州大学システムLSI研究センター

<https://hdl.handle.net/2324/6380>

出版情報 : 5th Symposium on Advanced Computing Systems and Infrastructures, SACSIS 2007 || ||
p152-153, pp.152-153, 2007-05-24. 先進的計算基盤システムシンポジウム

バージョン :

権利関係 :

マルチコアプロセッサ向けオンチップキャッシュにおける配線遅延の影響

伊藤 義崇[†] 千代延 昭宏[†] 佐藤 寿倫[‡]

Impact of Wire-delay on On-chip Cache Hierarchies in MultiCore Processors

YOSHITAKA ITO[†] AKIHIRO CHIYONOBU[†] TOSHINORI SATO[‡]

1. 背景と目的

近年の消費電力と発熱の問題を解決するために、マルチコアプロセッサが主流となりつつある。低速なコアを採用して消費電力や熱を抑制しながら、チップ内並列処理によってスループットを向上させ、高い性能を実現する。

しかし、コア数を増やしても必ずしもプロセッサ性能が向上するわけではない。この理由として、メモリウォール問題がある。プロセッサの処理性能が高くて、その要求するデータを十分に供給できなければ、性能は向上しない。プロセッサ・メモリ間のデータ供給能力がボトルネックとなる。このボトルネックを解消するために、たいいていのプロセッサではチップ上にキャッシュが搭載されている。しかし、キャッシュが大容量になりコアとキャッシュ間の物理的距離が増加すると、配線遅延によるアクセスレイテンシの増加が問題となる[1, 4]。

マルチコアプロセッサ向けのオンチップキャッシュには、コア間で共有する共有型と共有しない非共有型がある[7]。コア数が増えると、共有型キャッシュは大容量となり、レイテンシが増加する。レイテンシ増を抑えるために非共有型キャッシュの採用が考えられる[6]。しかし、非共有型ではキャッシュミス率が増え、オフチップアクセス回数が増える問題が発生する。本稿の目的は、配線遅延を考慮しつつ、共有型と非共有型キャッシュを比較検討することである。

2. マルチコアプロセッサ向けオンチップキャッシュ

1 次キャッシュは頻繁にアクセスされるため、レイテンシが小さいことが望まれる。このため、ほとんどのマルチコアプロセッサでは、各コアが固有の

1 次キャッシュを持つ。よって本稿では 2 次キャッシュを考察する。3 次キャッシュはチップ外にあるものとする。本稿の検討では、共有型のキャッシュの容量を非共有型キャッシュの総容量とする。つまりコアあたりのキャッシュ容量は同じである。

共有型の利点は、各コアがキャッシュの全領域を利用出来るため、容量性ミスが小さいことである。また、あるコアが参照したデータを他コアが利用出来るため、コア間で同一の命令コードやデータを参照する場合には、キャッシュ内には一つのコピーを持つだけで良い。コヒーレンス制御も不要である。欠点はアクセスレイテンシが大きいことである。コア数に応じて容量が増加すると、配線長が増加してレイテンシも大きくなる。また、複数のコアからの同時アクセスのためにポート数を増やすと、さらに面積と遅延の増大を引き起こす。他の欠点として、あるコアが参照するデータを他コアが追い出すことで、競合性ミスが増すことが考えられる。

一方、非共有型の利点は、コア数に関係なく各コアは一定容量のキャッシュを持つため配線長はほぼ一定であり、アクセスレイテンシが増大する心配が無いことである。欠点は、各コアは自身のキャッシュしか利用出来ないため、共有型と比較して容量性ミスが大きくなることである。また、コア間でデータを共有する場合には、コヒーレンス制御が必要になり、そのためにレイテンシとミス率が増加する。

3. 評価方法

マルチプロセッサシミュレータ M5[2]を用いて評価する。プロセッサとメモリ階層の構成は表 1 に示すとおりである。コア数が 2, 4, 8 の場合を評価する。コアあたりのキャッシュ容量が一定となるように、非共有と共有キャッシュの容量を決定している。アクセスレイテンシの決定には CACTI[5]を参

[†] 九州工業大学大学院情報工学研究科 Kyushu Institute of Technology

[‡] 九州大学システム LSI 研究センター Kyushu University

考にした。非共有キャッシュは1ポートであるが、共有キャッシュにはコア数と同数のポートを用意する。そのためコア数の増加に伴って、レイテンシが著しく増大することに注意されたい。8コア時の共有キャッシュでは、オフチップの3次キャッシュよりもレイテンシが大きくなっている。

表 1. プロセッサとメモリ階層の構成

コア周波数	4 GHz
命令幅	8 命令
ROB サイズ	192 エントリ
ISQ サイズ	64 エントリ
LSQ サイズ	32 エントリ
iALU/iMulDiv	4/1
fALU/fMulDiv/Sqrt	4/1
L1\$ (Inst/Data)	32KB,64B,2way,1port,2cycles
L2\$ (非共有)	512KB,64B,8way,1port,6cycles
L2\$ (2 コア共有)	1MB,64B,8way,2ports,11cycles
L2\$ (4 コア共有)	2MB,64B,8way,4ports, 27cycles
L2\$ (8 コア共有)	4MB,64B,8way,8ports,63cycles
パス幅	32B
バス周波数	800 MHz
L3\$ (オフチップ)	8MB,64B,16-way,56 cycles
メモリレイテンシ	400 cycles

本稿ではマルチプログラミング環境で評価する。したがってコヒーレンスの影響は評価の対象外とする。評価に用いたワークロードは表 2 のとおりである。この決定には文献[3]を参考にしている。

表 2. ワークロード

1	bzip2,parser
2	mcf,gcc
3	vpr,twof
4	equake,galgel
5	applu,lucas
6	galgel,swim
1	aspi,art,equak,mesa
2	ammp,mesa,swim,vortex
3	aspi,gzip,mcf,mesa
4	ammp,gzip,vortex,wupwise
5	bzip2,parser,mcf,gcc
6	equak,galgel,applu,lucas
1	bzip2,gcc,mcf,twof,ammp,applu,equak,swim
2	ammp,art,equak,mesa,mcf,gzip,parser,twof
3	bzip2,gcc,mcf,twof,ammp,art,galgel,lucas

4. 結果

図 1 に実行サイクル数の比較を示す。値が正の場合に非共有型が優れていることを意味する。どれかひとつのプログラムが 5 億命令を実行した時点でシミュレーションを終了している。

2 コアでは両者に大差は見られない。4 コアのワークロード 3 と 8 コアのワークロード 1 で非共有型が優位であるが、高々 14% の差である。これら二つのワークロードだけでは、コア数の増加にしたがって非共有型の優位性が増すとは言えない。

バンク分けによるポート数削減などによるレイテンシ削減を考慮すると、共有型が性能において勝る可能性が高いと考えられる。

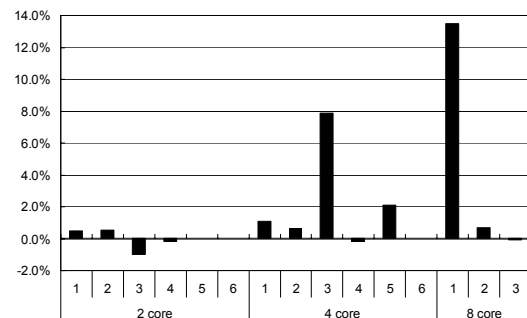


図 1. 実行サイクル数の削減割合

謝辞

本研究は一部、科学研究費補助金 (No. 16300019, No. 176549) の支援を受けた。

参考文献

- [1] B. Beckmann: Managing wire delay in large chip-multiprocessor caches, MICRO-37 (2004)
- [2] N. Binkert: Network-oriented full-system simulation using M5, 6th Workshop on Computer Architecture Evaluation using Commercial Workloads (2003)
- [3] J. Chang: Cooperative caching for chip multiprocessors, ISCA-33 (2006)
- [4] C. Kim: An adaptive, non-uniform cache structure for wire-dominated on-chip caches, ASPLOS-X (2002)
- [5] D. Tarjan: CACTI 4.0, Technical Report HPL-2006-86, HP Lab. (2006)
- [6] 伊藤: キャッシュ非共有型マルチコアプロセッサにおけるキャッシュの性能改善に関する研究, 信学技報 CPSY2006-51 (2006)
- [7] 三原: チップマルチプロセッサにおけるキャッシュメモリの特性解析, 情処研報 2006-ARC-169 (2006)