

一种基于伪 LRU 的新型共享 Cache 划分机制

倪亚路,周晓方

(复旦大学专用集成电路与系统国家重点实验室,上海 201203)

摘要: 本文提出了一种基于伪 LRU 方法的新型共享 Cache 动态划分策略 PLRU-SCP. 本文提出的划分策略在分析电路中给出了基于二叉树的新型分析方法,在划分电路中使用了一种非遍历的划分算法. 并提出了一种新型共享 Cache 结构. 本文提出的新型划分策略比基于 LRU 方法的不划分共享 Cache 策略和效用最优的划分策略的性能分别提高了 11.05% 和 8.66%.

关键词: 共享 Cache; 伪 LRU 替换方法; 二叉树分析; 非遍历划分算法

中图分类号: TP303; TN432 **文献标识码:** A **文章编号:** 0372-2112 (2013)04-0681-04

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2013.04.009

A Novel Pseudo-LRU Based Shared Cache Partitioning Mechanism

NI Ya-lu, ZHOU Xiao-fang

(ASIC & Systems State Key Laboratory, Fudan University, Shanghai 201203, China)

Abstract: A novel dynamic cache partitioning strategy based on pseudo-LRU policies (PLRU-SCP) is proposed in this paper. The analysis circuit for the proposed strategy makes use of a new statistical method based on binary tree, and the partitioning circuit employs the non-exhaustive algorithm. Besides, we presents a novel shared cache structure. According to simulation results, the proposed shared cache with PLRU-SCP outperforms those without the partitioning strategy using LRU policy (LRU-NP) by 11.05% on average performance, and has a better performance than those with the utility-based partitioning strategy using LRU method (LRU-UCP) by 8.66%.

Key words: shared cache; pseudo-LRU policy; binary tree analysis; non-exhaustive partitioning algorithm

1 引言

目前多核处理器已成为当前处理器发展的主要方向^[1,2]. 有效地管理多核共享的最后一级 Cache 对提高系统的性能,降低系统的功耗是十分重要的^[3]. 可以通过划分共享 Cache,减小不同核之间的相互污染. Qureshi^[4]等人提出了利用辅助标签目录实时监控进程运行情况,并以此做出效用最优的共享 Cache 划分. 此外,专门优化系统 IPC^[5]以及优化公平性^[6]的划分方法也被相继提出. 但是,由于这些划分策略过于强调优化系统某一方面的性能,因而在许多情况下反而会造成系统整体性能的下降. 又因为 LRU 替换方法硬件实现过于复杂,所以 Kedziarski^[7]等人进一步提出了基于伪 LRU 替换方法^[8,9]的共享 Cache 动态划分策略. 但该划分策略不能充分抑制由不同伪 LRU 替换方法自身特性造成的不当划分,且系统很难自动纠正错误划分.

因此,本论文提出了一种基于伪 LRU 方法的新型共享 Cache 动态划分策略 (PLRU-SCP). 经测试,本文提出的划分策略,比基于 LRU 方法的不划分共享 Cache 的

策略性能提高了 11.05%,比原效用最优划分策略^[4] (LRU-UCP) 性能提高了 8.66%. 硬件综合结果表明, PLRU-SCP 策略与 LRU-UCP 策略相比,分析电路、划分电路和共享 Cache 的总的面积和功耗分别下降了 13.82% 和 8.30%.

2 基于伪 LRU 的新型共享 Cache 动态划分

2.1 PLRU-SCP 系统结构

本文以双核系统为例,说明基于伪 LRU 的新型共享 Cache 动态划分策略,整体结构如图 1 所示.

2.2 二叉树分析电路

二叉树分析电路是由若干组标签 (tag) 目录和一组计数器构成的. 其中标签目录是从共享 Cache 的每 32 组标签目录中选取 1 组^[10],复制进入分析电路,并采用二叉树伪 LRU 方法^[9]进行更新. 假设采用 8 路组相连的共享 Cache,则在二叉树分析电路中建立的是 8 路组相连的标签目录. 其中每一组均如图 2 所示.

二叉树分析电路还提供一组同共享 Cache 相连度相等数量的计数器以统计各路的命中数,并依次标号.

在分析电路中,通过二叉树所携带的信息分析出命中路于所在组 LRU 链中所处的可能位置,再将相应计数器的值加 1.为此,本文提出了这样一种分析算法.若某二叉树结点为 0,则认为在 LRU 链中其右子树叶结点的各块位于其左子树叶结点各块之前;若二叉树结点为 1,则认为其左子树叶结点的各块位于其右子树叶结点各块之前.

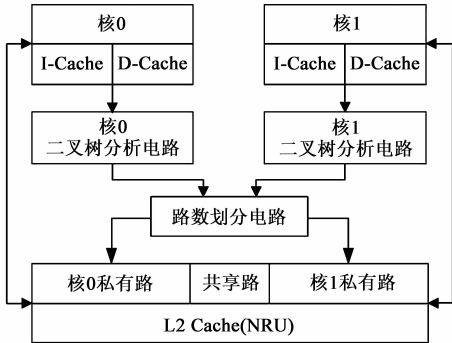


图1 PLRU-SCP系统结构图

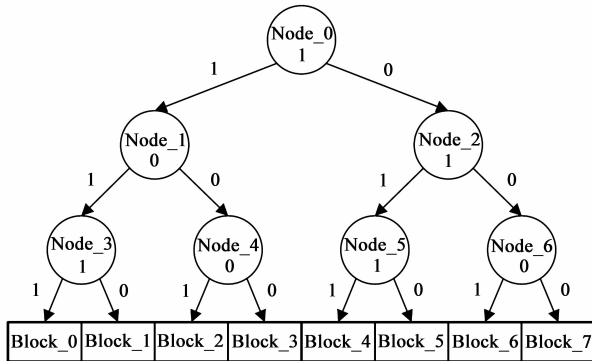


图2 二叉树分析电路8路组相连标签目录

2.3 路数划分电路

在划分电路中,本文提出了一种每次分配 2 路的非遍历划分算法.首先,为每个核分配 0 路.之后,在各核现有路数基础上,根据各核二叉树分析电路中各路命中数,比较为每个核新增 1 路后,各个核会增加的命中数.选取增加命中数最多的核,为其在现有路数的基础上增加 2 路.依此方式,直至所有私有路分配完成.

若需要分配的私有路路数为奇数,则按如上划分算法分配完成后,还需进行一次回溯操作.即假设每个核减少 1 路,选取命中数减少最小的核,从已为该核分配的路数中减少 1 路,至此分配完成.

因划分算法应至少为各核分配 1 路,故若分配完成后,有的核被分配了 0 路,则依旧按照减少命中数最少的原则,在已分得路数大于 1 的核中选取一个核,从其已分得的路数中减少 1 路,分配给未分得路数的核.

经测试,在分析电路采用二叉树分析算法的情况下,本章提出的每次分配 2 路的非遍历划分算法,性能与遍历算法相当,略优于每次分配 1 路的非遍历划分算法.但本章提出的划分算法硬件开销显著小于遍历算

法,所需比较次数也明显少于每次分配 1 路的非遍历划分算法.每隔固定周期执行一次划分,每次划分后将分析电路中各计数器值减半.

2.4 共享 Cache

本文将共享 Cache 每组中的各路,划分成私有路和公有路两大部分.各核可以访问共享 Cache 中的任意一路.所谓私有路就是当发生缺失,需要选取替换路时,该路只允许被其所属核引发的缺失替换,而共享路此时则可被任意核引发的缺失替换.在共享 Cache 中加入共享路,可以使系统在划分算法做出不恰当的划分后,自动纠正不当的划分.

为了减小硬件开销,本文提出了一种新型的划分成不同部分的共享 Cache 结构.以双核为例,在本文提出的共享 Cache 结构中,若划分电路为核 0 分配了 m 路,为核 1 分配了 n 路,此外有 s 路被两核共享.那么共享 Cache 各组中从 0 路至 $m-1$ 路会被分配给核 0, m 路至 $m+s-1$ 路会被分配为共享路, $m+s$ 路至 $m+s+n-1$ 路会被分配给核 1.

本文进一步提出了适用于这种结构的基于 NRU 的伪 LRU 替换方法.当某核访问共享 Cache 且命中时,则将对路路的 NRU 辅助位置 1.如果在命中路 NRU 辅助位置 1 后,该核的所有私有路和被各核共享的所有共享路的 NRU 辅助位均为 1,则在这些路中除去最近命中的这一路的 NRU 辅助位保持为 1 外,其他所有路的 NRU 辅助位均置 0.

若发生缺失,则在共享路 and 该核的私有路中,按照标号从小到大进行搜索,搜索到的第一个 NRU 辅助位为 0 的路即为替换路.替换完成后,将替换路的 NRU 辅助位置 1.之后,从核 0 开始检查,如果共享路和核 0 所有私有路的 NRU 辅助位均为 1,则将这些路中除这次的替换路外,其它所有路的 NRU 辅助位均置 0.然后,继续依此检查和调整共享路和其他核的所有私有路.

此外,由于划分电路至少会为每个核分配 1 路私有路,所以在本文提出的共享 Cache 结构中,其实可视为每个核均占有 1 路位置固定的私有路.以双核为例,在本文提出的共享 Cache 结构中,每组中的第一路会被固定分配给核 0,而最后一路会被固定分配给核 1.又因为在最优分配方案更新后的一段时间内,某核访问共享 Cache 时,共享路和该核所有私有路的 NRU 辅助位可能均为 1.所以在这种特殊情况下,如果发生缺失,则选取该核在共享 Cache 中固定位置的私有路为替换路.

3 测试结果

3.1 测试环境、用例及评测指标

本文在多核系统仿真器 M-Sim^[11]下构建基于 Alpha 指令集的双核系统.其中,一级指令 Cache 和数据 Cache

为各核私有,二级 Cache 共享,具体配置如表 1 所示.

表 1 测试环境配置

Processor	8 wide, out of order, 48 LSQ, 128 ROB
L1 I-Cache	32KB, 32B line-size, 2-way, NRU
L1 D-Cache	32KB, 32B line-size, 2-way, NRU
L2 Cache	512KB, 32B line-size, 16-way
Memory	300-cycle access latency

本文从 SPEC CPU2000^[12]测试集中选出多道访存敏感的测试用例,见表 2,将其两两一组在 M-Sim 构建的系统中进行测试,每组测试用例平均运行 3 亿条指令.

表 2 测试用例

序号	程序	序号	程序
1	crafty-fma3d	6	crafty-apsi
2	fma3d-applu	7	apsi-applu
3	applu-mcf	8	applu-lucas
4	vpr-fma3d	9	gzip-apsi
5	vpr-mcf	10	apsi-vpr

本文从加权加速比^[13]、吞吐率及基于 IPC 的公平性^[14]三个方面对系统进行评测,三个性能参数依次定义如式(1)至式(3)所示.本文主要以加权加速比表征系统性能.其中 IPC_i 为多个应用共享二级 Cache 时,第 i 个应用的 IPC;而 $Single_IPC_i$ 为第 i 个应用单独运行,独占共享 Cache 情况下的 IPC. N 为总线程数.

$$Weighted-Speedup = \sum (IPC_i / single_IPC_i) \quad (1)$$

$$Throughput = \sum IPC_i \quad (2)$$

$$Fairness = \frac{N}{\sum (single_IPC_i / IPC_i)} \quad (3)$$

3.2 各方法对系统性能影响测试结果及分析

本文在基于伪 LRU 方法的新型动态划分机制下,首先测试了共享 Cache 中共享路的路数对系统性能的影响,测试结果表明在共享 Cache 中设置 1 路共享路对系统性能最为有利.其次,对划分间隔周期进行了测试,结果表明每隔 830 万个时钟周期进行一次划分系统性能最为理想.

本文进一步测试了,在分析电路和共享 Cache 中选择基于不同伪 LRU 方法对系统性能的影响.测试结果表明,在分析电路中选择基于二叉树方法,在共享 Cache 中选择基于 NRU 方法的划分策略比原基于伪 LRU 替换方法的划分策略^[7]中,在分析电路和共享 Cache 中均选择基于 NRU 方法的划分策略,加权加速比、吞吐率和公平性分别提高了 3.17%、6.32% 和 5.58%;比均选择基于二叉树方法的划分策略则分别提高了 8.80%、9.53% 和 11.87%.

3.3 不同划分策略系统总性能测试结果及分析

最后本文对基于 LRU 方法的不划分共享 Cache 的策略(LRU-NP)、基于 LRU 方法的原效用最优划分策略(LRU-UCP)和基于伪 LRU 方法的新型划分策略(PLRU-SCP)进行了综合的比较.

图 3 给出了 10 组测试用例下,各策略的加权加速比.可以看出,本文提出的基于伪 LRU 方法的新型共享 Cache 划分策略,一方面在前 4 组中有效抑制了不恰当划分对系统性能的危害,另一方面在其余几组中基本保持了恰当划分的优势,因此使系统的平均性能获得了提升.测试结果表明,在加权加速比方面,采用 PLRU-SCP 的系统比采用 LRU-NP 的系统平均提高了 11.05%,比采用 LRU-UCP 的系统平均提高了 8.66%.

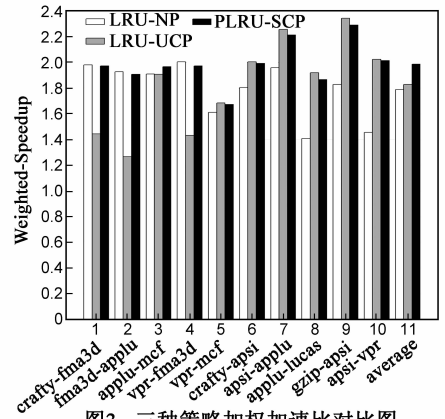


图3 三种策略加权加速比对比图

经测试,在系统吞吐率方面,PLRU-SCP 比 LRU-NP 平均提高了 14.59%,比 LRU-UCP 平均提高了 15.77%.在系统 IPC 公平性方面,PLRU-SCP 比 LRU-NP 平均提高了 20.60%,比 LRU-UCP 平均提高了 12.58%.

本文进一步硬件实现了原效用最优划分策略和基于伪 LRU 方法的新型共享 Cache 划分策略,并在 0.13 μm 工艺下综合.为了可在 FPGA 上进行验证,在硬件实现的上述两种策略中,一级 Cache 均仅拥有 32 组,二级 Cache 均仅拥有 128 组,其余配置则均同表 1.综合后,两种策略的面积和动态功耗对比见表 3,其中分析电路和共享 Cache 均包含各自的标签电路和控制电路.表 3 中三个模块在 PLRU-SCP 策略下总的面积和动态功耗,比在 LRU-UCP 策略下分别下降了 13.82% 和 8.30%.

表 3 PLRU-SCP 比 LRU-UCP 面积和动态功耗减小百分比及各模块面积和动态功耗在三个模块中所占百分比

模块名称	各模块面积减小百分比	各模块面积所占百分比	各模块动态功耗减小百分比	各模块动态功耗所占百分比
分析电路	17.46%	15.23%	11.25%	6.96%
划分电路	27.83%	2.90%	27.31%	0.69%
共享 Cache	12.54%	81.87%	7.90%	92.35%

4 结论

本文提出并实现了一种基于伪 LRU 方法的新型共享 Cache 动态划分策略. 在该策略中本文主要提出了, 一种基于二叉树的分析统计方法, 以及适应该方法统计特性的非遍历划分算法; 一种新型的共享 Cache 结构, 以及应用于该结构的基于 NRU 的替换方法.

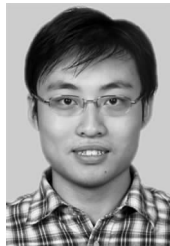
测试结果表明, 本文提出的基于伪 LRU 方法的新型共享 Cache 划分策略, 比基于 LRU 方法的不划分共享 Cache 策略和以效用最优为目标的划分策略的性能分别提高了 11.05% 和 8.66%. 综合结果表明, 本文提出的策略与基于 LRU 方法的原效用最优策略相比, 分析电路、划分电路和共享 Cache 的总的面积和功耗分别下降了 13.82% 和 8.30%.

参考文献

- [1] 朱允敏, 张丽伟, 等. 面向多核处理器的低级并行程序验证[J]. 电子学报, 2009, 37(21): 1-6.
ZHU Yun-min, ZHANG Li-wei, et al. Verifying parallel low-level programs for multi-core processor[J]. Acta Electronica Sinica, 2009, 37(21): 1-6. (in Chinese)
- [2] 杨际祥, 谭国真, 等. 多核软件的几个关键问题及其研究进展[J]. 电子学报, 2010, 38(9): 2140-2146.
YANG Ji-xiang, TAN Guo-zhen, et al. Some key issues and their research progress in multicore software[J]. Acta Electronica Sinica, 2010, 38(9): 2140-2146. (in Chinese)
- [3] 晏沛湘, 杨先炬, 张民选. 一种面向 CMP 的可变相联度混合 Cache 结构[J]. 电子学报, 2011, 39(3): 656-659.
YAN Pei-xian, YANG Xian-ju, ZHANG Min-xuan. A CMP oriented variable-way hybrid cache[J]. Acta Electronica Sinica, 2011, 39(3): 656-659. (in Chinese)
- [4] M K Qureshi, Y N Patt. Utility-based cache partitioning: A low-overhead, high-performance, runtime mechanism to partition shared caches[A]. The 39th Annual IEEE/ACM International Symposium on Microarchitecture[C]. Washington: IEEE Computer Society Press, 2006. 423-432.
- [5] Suo Guang, Yan Xunjun, Liu Guanghui, et al. IPC-based cache partitioning: an IPC-oriented dynamic shared cache partitioning mechanism[A]. International Conference on Convergence and Hybrid Information Technology[C]. Washington: IEEE Computer Society Press, 2008. 399-406.
- [6] S Kim, D Chandra, Y Solihin. Fair cache sharing and partitioning in a chip multiprocessor architecture[A]. The 13th International Conference on Parallel Architecture and Compilation Techniques[C]. Washington: IEEE Computer Society Press, 2004. 111-122.

- [7] K Kedzierski, M Moreto, F J Cazorla, et al. Adapting cache partitioning algorithms to pseudo-LRU replacement policies [A]. 2010 IEEE International Symposium on Parallel & Distributed Processing[C]. Piscataway: IEEE Press, 2010. 1-12.
- [8] Sun Microsystems, Inc. UltraSPARC T2 supplement to the UltraSPARC architecture 2007, Draft D1. 4. 3 [OL]. <http://www.opensparc.net/docs/UST2-UASuppl-current-draft-HP-EXT.pdf>, 2007.
- [9] T Chen, P Liu, et al. Implementation of a pseudo-LRU algorithm in a partitioned cache [P]. US Patent: 7069390, 2006-06.
- [10] M K Qureshi, D N Lynch, O Mutlu, et al. A case for MLP-aware cache replacement[A]. The 33th Annual International Symposium on Computer Architecture[C]. Washington: IEEE Computer Society Press, 2006. 167-178.
- [11] J Sharke. M-Sim: A flexible, multithreaded architectural simulation environment [R]. Department of Computer Science, State University of New York at Binghamton, Tech Report CS-TR-05-DP01, 2005.
- [12] SPEC CPU2000[OL]. <http://www.specbench.org/>.
- [13] A Snively, et al. Symbiotic jobscheduling for a simultaneous multithreading processor[A]. The International Conference on Architectural Support for Programming Languages and Operating Systems[C]. New York: ACM Press, 2000. 234-244.
- [14] K Luo, J Gummaraju, et al. Balancing throughput and fairness in SMT processors[A]. The 21st International Symposium on Performance Analysis of Systems and Software[C]. Washington: IEEE Computer Society Press, 2001. 164-171.

作者简介



倪亚路 男, 1987 年生于天津, 硕士研究生, 主要研究方向为专用集成电路设计.



周晓方 男, 1970 年生于上海, 1988 年复旦大学理学博士毕业, 现任教于复旦大学微电子学系, 高级工程师、硕士生导师. 研究方向: 嵌入式处理器架构和可重构数字信号处理, 数字集成电路 SoC 设计.

E-mail: xiaofangzhou@fudan.edu.cn