# 基于热点行搜索的低功耗数据高速缓存

仇 径[1],罗嘉蕙[1],项晓燕[2],陈志坚[1]
(1. 浙江大学超大规模集成电路设计研究所,浙江杭州 310027;2. 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:** 针对数据高速缓存短时间内频繁访问连续区段的特征,该文提出了一种基于热点硬件自搜索和历史访问轨迹的数据高速缓存低功耗方法. 该方法通过动态搜索热点片段,缓存目标热点行在高速缓存中的位置信息,过滤标签存储器和冗余数据存储器的访问. 运行 EEMBC 测试基准的实验结果表明,与基于 MRU(Most Recently Used)的路预测方法相比,该方法 Cache 的动态功耗可降低 30.77%,性能提升 26.21%.

**关键词:** 低功耗; 过滤访问; 热点行搜索
**中图分类号:** TP 302.2; TN 47 **文献标识码:** A **文章编号:** 0372-2112 (2016)01-0110-05
**电子学报 URL:** http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2016.01.016

# Low Power Data Cache Based on Hotspot Initiative Search

QIU Jing[1], LUO Jia-hui[1], XIANG Xiao-yan[2], CHEN Zhi-jian[1]
(1. *Institute of VLSI Design, Zhejiang University, Hangzhou, Zhejiang* 310027, *China*;
2. *State Key Laboratory of ASIC & System, Fudan University, Shanghai* 201203, *China*)

**Abstract:** On account of the characteristics that the data cache will be frequently accessed in a short period of time, this paper proposes a low power data cache access methods based on hotspot initiative search and historical access trace. The method will automatically judge hot process, dynamically buffer the target hotspot location information to filter tag and redundant data memory access. Running EEMBC benchmark results show that compared with MRU prediction method (Most Recently Used) this approach can reduce 30.77% dynamic access power, and get 26.21% performance improvement.

**Key words:** low power; filter access; hotspot initiative search

## 1 引言

高速缓存(Cache)普遍应用于嵌入式处理器中,用以解决存储墙的问题. 随着 Cache 容量和组相连度的不断提升,Cache 的功耗增加明显. 据统计,Cache 功耗已经达到了嵌入式处理器总功耗的 20% ~60%[1,2],降低数据 Cache 功耗对降低整个处理器功耗有重要意义[3].

在降低数据 Cache 功耗方面,目前主流方法包括路预测访问方法和路过滤访问方法两种. 文献[4]中路预测访问方法根据 MRU 表从组内多路数据中猜测选择一路数据,投机完成 Cache 数据访问. 如果预测错误,则在下一个周期内完成剩余候选块的标识比较和数据访问,会明显增加 Cache 的访问时间. 为了提高传统路预测 Cache 预测的准确率,文献[5]提出一种通过引入最小预测位的路预测访问方法,进一步提高路预测算法的预测率. 但这些方法只能针对特定的应用增加预测准确率,无法普遍适用. 因此,文献[6]提出一种基于行为模式预测的访问方法,该方法根据访问 Cache 的历史信息,通过预测器判定当前指令是否适合采用路预测技术以及适用的路预测方法类型. 上述所有路预测方法的缺陷在于无法避免标签路的访问,且一旦预测错误,需要额外访问 Cache,造成性能和功耗损失.

基于路过滤的方法,与路预测方法的本质区别在于,路过滤并非投机访问,而是根据过滤算法排除冗余的 Cache 访问,减少功耗. 文献[7]提出了过滤 Cache 的方案,当 CPU 读取数据时,首先访问过滤 Cache,若缺失,则访问主 Cache. 该方案在过滤失效时会导致较长的访问延时. 为解决这个问题,文献[8]提出了热点

收稿日期:2014-05-06;修回日期:2015-02-17;责任编辑:李勇锋

Cache 方案,仅在热点程序中访问热点 Cache,其他程序访问主 Cache. 热点 Cache 减少了过滤 Cache 方法的访问延时,但仍需额外的 Cache 逻辑,增加静态功耗. 文献[9]提出了一种基于多元信息的路过滤访问方法,不增加额外的 Cache 逻辑,但只能过滤 34.3% 无效 Cache 的访问. 为提高过滤效率,文献[10]通过标志预访问来提前过滤后续指令的冗余访问操作,文献[11] 通过使用邻行信息链接历史访问表,获取指令的路信息,可过滤 96% 的指令 Cache 冗余访问. 但由于数据 Cache 的不确定性和不连续性,上述方法无法进行针对性的过滤访问;文献[12]利用过滤 Cache 和 L1 Cache 访问延时的不同使用字选择和标签提前比较的方式来避免冗余数据路访问,但对于大多数标签路和数据路同时访问的数据 Cache,该方法无效.

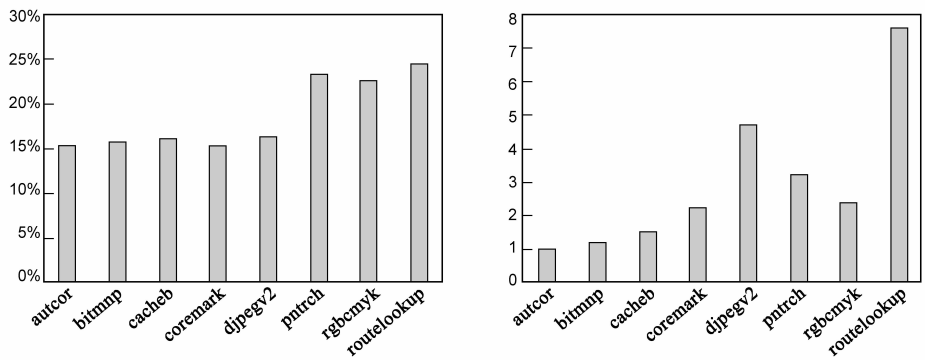
为满足数据 Cache 功耗和性能的双重要求,本文提出了一种热点自搜索的数据 Cache 访问方法. 本文的主要内容包括:(1)研究数据 Cache 访问规律,发现热点行数目平均仅占数据 Cache 的访问次数的 0.4%,却为程序提供了 80.4% 的数据,并有短时间内被频繁访问的规律. (2)针对 Cache 访问热点集中的规律,提出一种热点自搜索数据 Cache 的低功耗访问方法. 通过追踪程序热点行,自动缓存热点程序访问的路选择信息,高效过滤冗余存储器访问. (3)进一步,为同时优化数据 Cache 的性能和时序,复合使用空间信息和时间信息索引历史轨迹,提升索引效率. (4)通过对 EEMBC 测试程序的测试,基于本方法的 Cache 与基于 MRU 的路预测方法的 Cache 相比动态功耗降低 30.77%,性能提升 26.21%.

## 2　数据 Cache 访问规律研究

高速缓存的访问是以路为单位进行的. 通过对主流嵌入式测试基准程序 EEMBC 的分析,发现程序所访问数据高速缓存行数目与存储载入指令的数目的比例小于 2%(如表 1),程序数据访问具有良好的空间局部性,其中热点缓存行(定义同文献[8])数目小于 20%(如图 1(*a*)),说明高速缓存行的访问空间集中在少数热点缓存行. 通过实时监测发现,每百条连续的存储载入指令平均访问的缓存行数目少于 10 行(如图 1(*b*)),热点缓存行在短时间内被频繁访问,Cache 访问的时空局部性明显.

**表 1　访问数据高速缓存行数目与存储载入指令的数目的比例**

| 测试基准程序名 | Autcor | Bitmnp | Cacheb | Coremark | Djpegv2 | Pntrch | Rgbcmyk | Routelookup |
|---|---|---|---|---|---|---|---|---|
| 比例 | 0.001% | 0.021% | 0.185% | 0.013% | 0.151% | 0.324% | 0.484% | 0.008% |




(*a*) 热点缓存行占比　　(*b*) 百条存储载入指令的平均缓存行

**图1**

## 3　过滤访问方法

基于数据 Cache 热点访问集中的特点,本文提出了一种硬件自搜索热点行,从而低功耗访问数据 Cache 的方法. 该方法的核心思想是:(1)自动缓存存储载入指令的访问地址以及路选择信息,使用缓存的信息过滤后续相同行的 Cache 访问操作,避免标签路以及冗余数据路访问;(2)利于访问频度阈值判定方法,主动跟踪程序,判定所访问数据是否属于热点程序,并自动搜索热点程序后续邻近行的路选择信息;(3)Cache 的过滤与访问操作串行完成,失败的过滤操作正常访问数据 Cache,避免性能和功耗损失;(4)使用时空复合高效索引,过滤操作不增加额外延时. 具体实现方式描述如下:

过滤操作通过历史轨迹单元实现,该单元记录热点存储载入指令所在高速缓存行的地址和路选择信息. 当后续存储载入指令执行时,首先与历史轨迹单元所记录的地址进行匹配,若地址匹配,则根据记录的路选择信息,仅访问所记录的 Cache 数据路,过滤其他组

相连的 Cache 数据路和标签路的访问,省略标签比较过程;若没有匹配,则将其访问地址更新存入历史轨迹单元,正常访问数据 Cache,待访问结束后将此次访问的路选择信息更新回历史轨迹单元(如图 2).

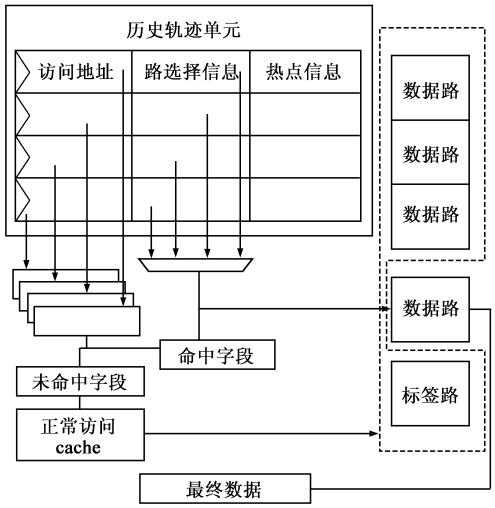



图2 历史轨迹单元示意图

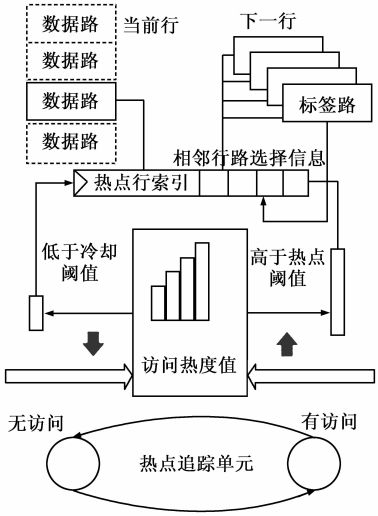



图3 热点追踪单元示意图

热点自搜索操作通过热点追踪单元(如图 3)实现. 热点追踪单元主动监视历史轨迹单元中所有行的访问状态,当某行数据被访问时,热点追踪单元会提升该行的热度值,并按越早访问热度值越低的原则降低其他行的热度值. 热度值超过热点阈值时,热点追踪单元会将该行标示为热点行. 被标示的热点行,热点追踪单元会主动访问数据 Cache,提取其相邻行的路选择信息,缓存在历史轨迹单元中. 当相邻行数据被访问时,直接使用历史轨迹单元中的信息过滤标签路和冗余数据路的访问,并主动触发下一相邻行信息的缓存. 当热度值低于冷却阈值时,该行被取消热点行标记,停止预缓存相邻路的信息. 热点的相邻行在历史轨迹单元中使用热点行的信息进行索引,不耗费冗余资源.

使用存储载入指令的物理地址索引历史轨迹单元理论上可以获得最高的索引效率. 但由于存储载入指令的物理地址需经过基址和偏移量运算,通过内存管理单元转换虚实地址获得,难以保证过滤操作在 Cache 访问之前完成. 本文使用低位虚拟地址进行索引,避免全地址计算和虚实地址转换引入的关键路径. 但若仅引入低位地址等空间信息索引,缺乏时间信息,会导致特定程序的索引效率低,具体表现为:当不同程序段的存储载入指令的低位地址相同出现叠名时,历史轨迹单元记录的信息会在叠名的两条指令中反复抖动. 为了降低该损失,索引机制须同时引入时间信息和空间信息进行索引. 本文提出利用时间信息(PC、PID)和空间信息(地址)的哈希索引算法,使用高位 PC 拼接低位地址和 PID 的方式索引历史轨迹单元. 使用这种时空复合索引的机制可以大幅减少索引历史轨迹单元所需延时,索引效率和使用存储载入指令的物理地址的传统索引方式相差仅为 3%.

实现硬件电路如图 4 所示.

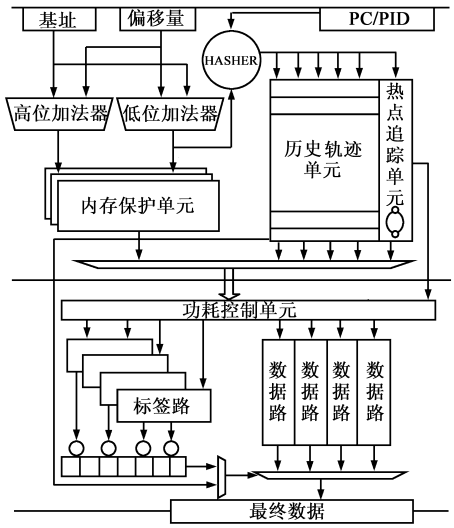



图4 热点自搜索的低功耗数据cache实现

## 4 实验探究

实验平台使用 CK807 效能优先嵌入式处理器. 该处理器使用自主研发的 C-SKY 16/32 位混编 RISC 指令集,支持指令双发乱序执行,数据 Cache 大小 16KB,四路组相连,每路数据 256bit,采用 LRU 替换策略,在 tsmc65lp 工艺实测频率达到 666MHz. 实验部分基于 EEMBC 测试基准的典型程序,在实现基于热点行搜索的低功耗数据高速缓存的基础上,进一步搜索了历史轨迹单元的深度对过滤效率的影响.

实验结果表明深度为 4 的历史轨迹单元命中比例在 77% 左右;深度为 8 时则大幅提升至 86%;深度为 16 时进一步提升为 93%;但由于热点行访问的时空局部性明显,深度为 32 的历史轨迹单元命中比例提升不明显,仅提升 1% 至 94% 左右(如图 5).

根据文献[13]的结论,Cache 的动态访问功耗同比于访问的位宽和次数之积. 实验平台的 Cache 为四路组相连结构,每次访问 64 位数据路,88 位标签路,单次过滤成功可减少 81% 的动态访问功耗. 针对 EEMBC 测试程序,深度不同的历史轨迹单元平均可以减少 58% 到 73% 的动态访问功耗(如图 6).

为了达到频率要求(tsmc65lp 工艺达到 600MHZ 以上),Cache 命中的载入指令通常需经过地址准备、Cache 访问、数据选择三个时钟周期. 过滤成功时,数据选择的过程可省略,载入指令可在两个周期内完成,后续数据依赖的程序能够更早获得操作数,提速程序运行. 该过滤方法针对 CK807 这个 CPU 可以提升 6.2% 的处理器性能(如图 7).

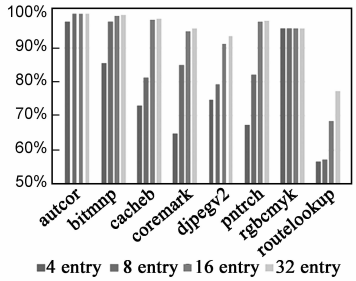


图5　历史轨迹单元命中比例

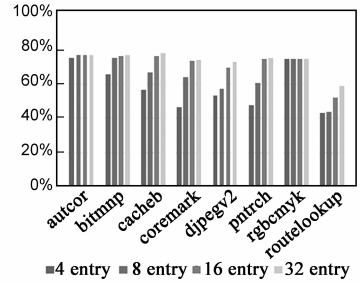


图6　动态访问功耗减少比例

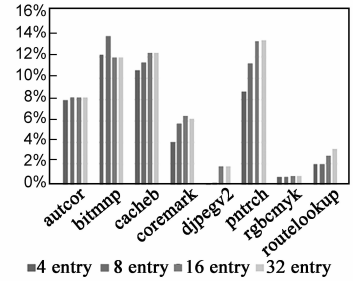
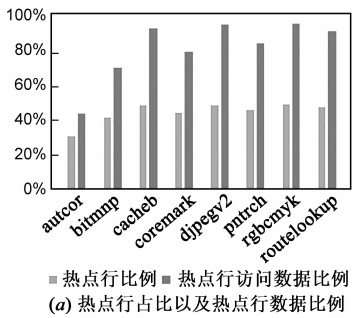


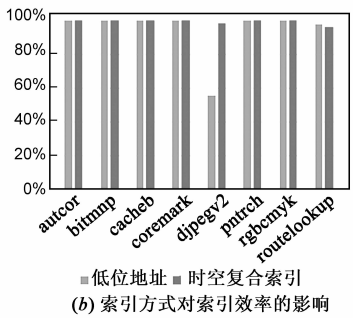
图7　处理器性能提升比例

在 EEMBC 测试程序中,使用深度为 16 的历史轨迹单元,热点阈值设置为 4 时,热点行预取的效能最高. 其中被自动标识为热点行的程序占比为 22.93%,平均 84.83% 的数据通过热点行以及其预取的相邻行实现数据 Cache 的过滤访问. 与不使用热点自搜索的方法相比,提升了历史轨迹单元 39.97% 的命中率(如图 8(*a*)). 本文还提出了空间时间信息复合索引机制. 实验表明,PC 信息的引入可以很好的离散索引信息,提升历史轨迹单元的索引效率(如图 8(*b*)).

为了降低硬件的实现复杂度和历史轨迹单元本身的能耗开销,针对 EEMBC 基准测试程序,选择深度为 16 的历史轨迹单元能最好地实现数据 Cache 的过滤访问. 在资源相当的情况下,与 MRU 算法[4] 和与字选择方法[12] 本文的方法可以提升过滤准确率,且不会造成性能损失. 实验表明数据 Cache 的动态功耗可降低 30.77%(与 MRU 相比)5.1%(与字选择相比),性能提升 26.21%(与 MRU 相比)11.88%(与字选择相比)(如图 9).



(*a*) 热点行占比以及热点行数据比例



(*b*) 索引方式对索引效率的影响

图 8

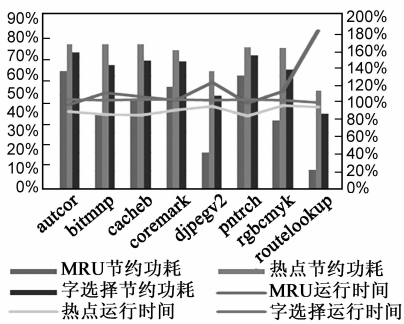


图9　Cache访问功耗与时间

## 5　总结

本文提出了一种基于热点行搜索的数据 Cache 过滤访问方法. 自动搜索热点程序,缓存热点行路选择信息,使用时空信息复合索引被缓存信息,过滤数据 Cache 的访问. 实验表明,在成本略微增加的前提下,与传统 Cache 访问方法相比,实际动态访问功耗降低 70.1%,性能提升 6.2%;与基于 MRU 的路预测方法相比,动态功耗降低 30.77%,性能提升 26.21%;与基于字选择的路过滤方法相比,动态功耗降低 5.1%,性能提升 11.8%.

### 参考文献

[1] Zang W,Gordon-Ross A. A survey on cache tuning from a power/energy perspective[J]. ACM Computing Surveys (CSUR),2013,45(3):32.

[2] ZHENG Z, Zhiying W, Li S. Region-based way-partitioning on L1 data cache for low power[J]. IEICE Transactions on Information and Systems, 2013, 96(11): 2466 – 2469.

[3] Xiangyun Z, Lianfeng Z, Dong B. Research on the low power design method for the embedded multi-core processor[A]. 2013 Fourth International Conference on Digital Manufacturing and Automation (ICDMA) [C]. IEEE, 2013. 1141 – 1144.

[4] Inoue K, Ishihara T, Murakami K. Way-predicting set-associative cache for high performance and low energy consumption[A]. Proceedings of the 1999 International Symposium on Low Power Electronics and Design [C]. ACM, 1999. 273 – 275.

[5] Chen H C. Design of a low-power way-predicting cache using valid-bit pre-decision strategy[J]. Journal of the Chinese Institute of Engineers, 2008, 31(5): 805 – 814.

[6] Ye J, Ding H, Hu Y, et al. A behavior-based adaptive access-mode for low-power set-associative caches in embedded systems[J]. Journal of Information Processing, 2012, 20(1): 26 – 36.

[7] Kin J, Gupta M, Mangione-Smith W H. The filter cache: an energy efficient memory structure [A]. Proceedings of the 30th Annual ACM/IEEE International Symposium on Microarchitecture[C]. IEEE Computer Society, 1997. 184 – 193.

[8] Yang C L, Lee C H. HotSpot cache: joint temporal and spatial locality exploitation for i-cache energy reduction [A]. Proceedings of the 2004 International Symposium on Low Power Electronics and Design[C]. IEEE, 2004. 114 – 119.

[9] Fan L, Wang S, Zheng Y, et al. Low power cache architectures with hybrid approach of filtering unnecessary way accesses[A]. Proceedings of the 2013 International Workshop on Programming Models and Applications for Multicores and Manycores[C]. ACM, 2013. 93 – 99.

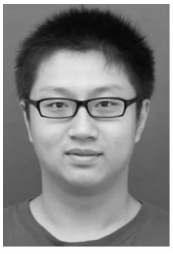
[10] 张宇弘,王界兵,严晓浪,等. 标志预访问和组选择历史相结合的低功耗指令 cache[J]. 电子学报, 2004, 32(8): 1286 – 1289.
Zhang Y, Wang J. Pre-visiting tag and keeping way history to reduce power in instruction cache[J]. Acta Electronica Sinica, 2004, 32(8): 1286 – 1289. (in Chinese)

[11] 项晓燕,陈志坚,孟建熠,等. 基于邻行链接访问的低功耗指令高速缓存[J]. 浙江大学学报(工学版), 2013, 7: 011.
Xiang X, Chen Z. Low power instruction cache based on adjacent line linking access[J]. Journal of Zhejiang University(Engineering Science), 2013, 7: 011. (in Chinese)

[12] Choi J H, Kwak J W, Jhang S T, et al. Data filter cache with word selection cache for low power embedded processor[A]. Proceedings of the 2013 Research in Adaptive and Convergent Systems[C]. ACM, 2013. 422 – 427.

[13] Kamble M B, Ghose K. Analytical energy dissipation models for low power caches[A]. Proceedings of International Symposium on Low Power Electronics and Design [C]. IEEE, 1997. 143 – 148.

## 作者简介



**仇 径** 男, 1988 年出生于江西省, 2010 年获浙江大学电气工程学院学士学位. 现为浙江大学超大规模集成电路设计研究所博士研究生, 主要研究方向为低功耗处理器设计与研究.
E-mail: qiujing@ vlsi. zju. edu. cn

**罗嘉蕙** 1989 年出生于湖南省, 2012 年获浙江大学电气工程学院学士学位. 现为浙江大学超大规模集成电路设计研究所博士研究生, 主要研究方向为处理器体系结构设计与研究.