ARTICLE IN PRESS

J. Parallel Distrib. Comput. ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

J. Parallel Distrib. Comput.

journal homepage: www.elsevier.com/locate/jpdc

The Tag Filter Architecture: An energy-efficient cache and directory design

Joan J. Valls [a,*], Alberto Ros [b], María E. Gómez [a], Julio Sahuquillo [a]

[a] Universitat Politècnica de València, Spain

[b] Universidad de Murcia, Spain

HIGHLIGHTS

- Homogeneous distribution of the less significant bits of the tag across ways of cache sets.
- A single bit of the address is enough to guarantee a tag mismatch.
- We propose a mechanism to filter the ways accessed using the less significant bits of the address tag.
- Dynamic power consumption in the processor cache is reduced up to 85.9%.
- Dynamic power consumption in the directory cache is reduced up to 84.5%.

ARTICLE INFO

Article history:
Received 21 August 2015
Received in revised form
22 April 2016
Accepted 28 April 2016
Available online xxxx

Keywords:
Multicore processors
Cache
Directory
Dynamic consumption

ABSTRACT

Power consumption in current high-performance chip multiprocessors (CMPs) has become a major design concern that aggravates with the current trend of increasing the core count. A significant fraction of the total power budget is consumed by on-chip caches which are usually deployed with a high associativity degree (even L1 caches are being implemented with eight ways) to enhance the system performance. On a cache access, each way in the corresponding set is accessed in parallel, which is costly in terms of energy. On the other hand, coherence protocols also must implement efficient directory caches that scale in terms of power consumption. Most of the state-of-the-art techniques that reduce the energy consumption of directories are at the cost of performance, which may become unacceptable for high-performance CMPs.

In this paper, we propose an energy-efficient architectural design that can be effectively applied to any kind of cache memory. The proposed approach, called the Tag Filter (TF) Architecture, filters the ways accessed in the target cache set, and just a few ways are searched in the tag and data arrays. This allows the approach to reduce the dynamic energy consumption of caches without hurting their access time. For this purpose, the proposed architecture holds the *X* least significant bits of each tag in a small auxiliary *X-bit*-wide array. These bits are used to filter the ways where the least significant bits of the tag do not match with the bits in the *X-bit* array. Experimental results show that, on average, the TF Architecture reduces the dynamic power consumption across the studied applications up to 74.9%, 85.9%, and 84.5% when applied to L1 caches, L2 caches, and directory caches, respectively.

1. Introduction

As silicon resources become increasingly abundant, core counts grow rapidly in successive chip-multiprocessors (CMPs) generations. These CMPs usually improve their memory latency by using a multi-level cache hierarchy that occupies a significant percentage of the overall CMP die area [3] and consumes an important fraction of the overall power budget. Most of the cache power consumption is due to dynamic power while a small fraction is due to static power. Dynamic power consumption comes from switching activity of transistors during cache accesses while static power consumption comes from current leaking, even when the cache is not being accessed. Cache designers must reach a compromise among performance, cost, size, and power/energy dissipation.

Dynamic energy consumption is dominated by first-level (L1) caches since they are much more frequently accessed than last

* Corresponding author.
E-mail addresses: joavalmo@fiv.upv.es (J.J. Valls), aros@ditec.um.es (A. Ros), megomez@disca.upv.es (M.E. Gómez), jsahuqui@disca.upv.es (J. Sahuquillo).

http://dx.doi.org/10.1016/j.jpdc.2016.04.016
0743-7315/

level caches (LLC), *e.g.* L3 caches. Moreover, the high consumption of caches is due to the parallel access of tag and data arrays, a common characteristic in L1 caches since they have a strong influence on the overall processor performance. This concern is even more important in CMPs than in monolithic processors since L1 caches can be accessed both from the processor side and from the interconnection network side due to coherence requests, which increases the number of accesses.

Due to performance reasons, processor caches are being deployed with a high associativity degree. In high-performance microprocessors, all the cache ways in the target set are accessed concurrently on a cache lookup. Thus, the associativity degree defines the number of accessed tags. In addition, caches include one comparator per way and compare as many tags as the number of ways. As a consequence, the dynamic energy dissipated per access increases along with the cache associativity.

Apart from the cache hierarchy, other cache structures are implemented in typical multiprocessors. In current manycores, like the Xeon Phi [13], directory-based coherence is the commonly preferred approach. Cache memories, referred to as directory caches or sparse directories, are used to keep track of the cached blocks. In this approach, when a directory entry is evicted, all copies of the tracked block in the processor caches are invalidated, even if the block is being used by the processor, thus rising the so-called *coverage* misses [25]. Mainly due to power reasons (*i.e.* lower associativity degree in comparison to duplicate tags directories [21,23,24]), sparse directories are the preferred design choice for a medium to high number of cores.

Previous research on the design of energy-efficient caches addressing dynamic power has focused on minimizing the internal transistor activity during a cache access. That activity comes from reading and comparing tags in the tag array, and from reading and writing data in the data array. Ideally, on a cache hit, the cache would read and compare a single tag, and access a single data entry without losing performance. Furthermore, on a miss, the cache would have no need to access neither the tag array nor the data array.

Many cache energy reduction approaches have focused on monolithic processors in the past such as Cache Decay [14], Drowsy Caches [8], and Way Guard [10]. Some of them, *e.g.* [5], were originally developed to reduce cache access time, but subsequent research has proven that these schemes provide important energy savings. However, since these schemes are not directly applied to CMPs, recent research has dealt with energy savings on CMPs when running parallel workloads.

In this work, we propose the Tag Filter (TF) Architecture, a technique that can be applied to any set-associative cache in the system, ranging from processor caches that store memory blocks to directory caches holding coherence information. The TF Architecture reduces the number of tags and data entries checked when accessing a cache, which leads to reducing the dynamic power consumption. The TF Architecture is based on using the X least significant bits (LSB) of the address tag ($X = 1, 2, 3, 4$) to discern which ways of a cache may contain the searched block. Only those ways that may potentially contain the block are accessed, thus saving the energy required to access the other ways. This allows caches to implement high-associativity with a similar energy consumption as that of conventional low-associativity caches, while reducing the number of conflict misses.

The TF Architecture is deployed with minimal hardware complexity, while allowing tag and data arrays to be accessed in parallel, so no performance degradation rises with respect to conventional caches. In addition, unlike other approaches [30], no way-alignment across sets must be done, which simplifies its implementation.

Experimental results show that the TF Architecture with $X = 4$ can reduce dynamic power consumption up to 74.9% when applied to the L1 cache and up to 85.9% when applied to the L2 cache. Additionally, when applied to the directory caches, the TF Architecture can reduce the dynamic energy dissipated up to 84.2% for a single-level directory cache and 84.5% for a multi-level directory cache.

The remainder of this paper is organized as follows. Section 2 describes the main reasons that motivate us to carry out this research. Section 3 discusses the related work. Section 4 presents the proposal. Section 5 describes the simulation framework. Section 6 presents how the number of ways accessed is reduced and energy results. Finally, Section 7 draws some concluding remarks.

2. Motivation

Cache memories, especially first- and second-level caches, are frequently accessed, since memory reference instructions represent a significant percentage of the executed instructions. Fig. 1 shows the fraction of memory reference instructions in each individual benchmark executed on a 16-core CMP system (experimental environment, system parameters, protocols and cache hierarchy are described in Section 5). This value is roughly the same, around 16%, across the different benchmarks.

Therefore, a significant fraction of the total power budget is often consumed by on-chip caches. For example, in the Niagara2 processor [29], where 44% of the chip power is consumed by the L2 cache [17]. Reducing dynamic power consumption in caches of CMPs is an actual problem that is being under research [30,16]. To deal with this problem, this paper proposes an architectural approach based on the hypothesis of the homogeneous distribution of the least significant bits of the address tag across the ways of a set-associative cache [26].

We launched experiments to verify this hypothesis in the considered experimental scenario. Fig. 2 shows the average distribution of the blocks across an 8-way L1 cache and a 16-way L2 cache on a 16-core CMP system. In Fig. 2(a) and (b) on average there are 1 and 2 ways in *invalid state*, under the implemented MOESI protocol, in the L1 and L2 cache, respectively. Meanwhile, the remaining ways share a quite homogeneous distribution considering the lowest order tag bits of the allocated blocks. Therefore, there is no need to access all the ways in a set if there is some mechanism able to filter accesses for a subset of ways that might potentially allocate the requested block. The homogeneous distribution of the lowest order tag bits makes our approach a perfect method for filtering accesses.

Directory caches are typically built as set-associative caches and, as experimental results will show, the least significant bits in the address tag follow a similar homogeneous distribution. A low associativity of the directory caches causes frequent evictions of directory entries that lead to extra coverage misses in the processor caches, thus as in any kind of cache, a higher associativity would improve the performance. However, their associativity is also limited due to power constraints. This power limitation can be attacked using the TF Architecture that allows the same performance as highly-associative directories with similar dynamic energy consumption as low-associative directories.

In summary, the aim of this paper is to save dynamic energy by reducing the number of lookups on each cache or directory access while keeping the performance of high-associative caches.

3. Related work

An important amount of research has focused on reducing the energy consumption of cache memories. Most of this work has mainly focused on processor caches which already was a concern in monolithic processors. Other work has focused only on directory caches in manycore CMPs where the problem mainly lies on scalability issues. Below, we summarize these two lines of research.

Please cite this article in press as: J.J. Valls, et al., The Tag Filter Architecture: An energy-efficient cache and directory design, J. Parallel Distrib. Comput. (2016), http://dx.doi.org/10.1016/j.jpdc.2016.04.016