Skewed-associative Caches *

André Seznec, François Bodin
IRISA, Campus de Beaulieu
35042 Rennes Cedex
FRANCE
tel: (33) 99 84 73 36
FAX: (33) 99 38 38 32
e-mail: seznec@irisa.irisa.fr

Abstract. During the past decade, microprocessor peak performance has increased at a tremendous rate using RISC concept, higher and higher clock frequencies and parallel/pipelined instruction issuing. As the gap between the main memory access time and the potential average instruction time is always increasing, it has become very important to improve the behavior of the caches, particularly when no secondary cache is used (i.e. on all low cost microprocessor systems). In order to improve cache hit ratios, set-associative caches are used in some of the new superscalar microprocessors.

In this paper, we present a new organization for a multi-bank cache: the skewed-associative cache. Skewed-associative caches have a better behavior than set-associative caches: typically a two-way skewed-associative cache has the hardware complexity of a two-way set-associative cache, yet simulations show that it exhibits approximately the same hit ratio as a four-way set associative cache of the same size.

Keywords: microprocessors, cache, set-associative cache, skewed-associative cache.

1 Introduction

Performance of commercial microprocessors is increasing at a tremendous rate: 100 MHz clocks were announced on the MIPS R4000 processor in September 1991, a 200 MHz clock processor is offered by the end of 1992 by DEC, etc. At the same time, the architecture complexity of microprocessors is also increasing. First generation RISC microprocessors such as the MIPS R2000 or the first SUN Sparc microprocessors rapidly become obsolete. In association with technological advances which have allowed faster and faster clock speeds and larger and larger transistor counts, two major architectural techniques have been used in order to improve performance of the microprocessors: superscalar issuing of the instructions (IBM Power, SUN SuperSparc, Intel i860, etc) and “superpipelining” (i.e. increasing clock frequency by deepening the pipeline as in the MIPS R4000) [6, 15].

Most of the newly introduced commercial microprocessors have been designed to address a very large segment of the market: constructors generally claimed to address low cost embedded systems as well as high end file servers or workstations with the same basic microprocessor architecture.

In order to feed these new microprocessors with both instructions and data, a large memory is needed, and the effective performance of the whole system highly depends of the performance of the memory system. Unfortunately, over the last ten years, the main memory access cycle time has not decreased at the same rate as the processor cycle time. On a superscalar microprocessor such as the IBM Power e.g., the demand on memory instruction throughput may be up to 4 instructions per cycle and the demand on memory data throughput may be very close to one word of data per cycle. When the penalty for a cache miss is about 20 cycles, the performance may dramatically decreased when the number of cache misses increases even very smoothly (Amdahl’s law).

Increasing the hit ratio of both data and instruction cache is a key issue in order to improve the effective performance of microprocessors and particularly for low-end systems.

* This work was partially supported by CNRS (PRC-ANM)
In section 2, we propose a new data mapping on a partially associative cache: the skewed-associative cache. Then some properties wished on skewing functions are characterized and a family of "good" skewing functions is presented.

Trace driven simulations results presented in section 3 shows that two-way skewed associative exhibits the same hit ratio as a four-way set associative cache of the same size, but at the hardware cost of a two-way set-associative cache.

2 Skewed-associative caches

2.1 Skewing on caches: principle

A set-associative cache is illustrated in Figure 1: a X way set-associative cache is built with X distinct banks of $\frac{1}{X}$ cache lines. Then a line of data with base address $D$ may be physically mapped on physical line $f(D)$ in any of the distinct banks.

We propose a very slight modification to this cache organization illustrated in Figure 2:

Different mapping functions are used for the distinct cache banks i.e., line of data with base address $D$ may be mapped on physical line $f_0(D)$ in cache bank 0, in $f_1(D)$ in cache bank 1, etc. We call a multi-bank cache with such a mapping of the lines onto the distinct banks: a skewed-associative cache.

This hardware modification is very slight. It will induce a very marginal hardware upper cost when designing a new microprocessor with on-chip caches: we may choose $f_i$ which may be implemented with a very few gates. But we shall see that this may help to increase the hit ratio of caches and then to increase the overall performance of a microprocessor using a multi-bank cache structure.

Note that skewed-associative caches may be used for internal primary caches as well as for external caches (primary or secondary).

Related works In 1977, Smith [12] considered set-associative caches and proposed to select the set by hashing the main memory address; this approach corresponds to figure 1: a single hashing function is used.

More recently Agarwal [2] (Chapter 6.7.2) studied hash-rehash caches.