Simulating a Reconfigurable Cache System for Teaching Purposes

Ezequiel Herruzo, J. Ignacio Benavides
Dept. Computer Architecture and Electronics
University of Córdoba
eze@uco.es, ellbebej@uco.es

Ricardo Quislant, Emilio L. Zapata, Oscar Plata
Dept. Computer Architecture
University of Málaga
ezapata@ac.uma.es, oscar@ac.uma.es

Abstract

This paper presents a tool that simulates a reconfigurable cache whose parameters can be changed at runtime through a special instruction at the ISA level. The tool was developed through a series of laboratory exercises in Computer Architecture. The proposed tool simulates a cache system that can be reconfigured within a variety of 298 combinations of C, W and L (cache capacity, block size and number of blocks per set) without changing its architecture. The students are introduced to reconfigurable hardware architecture while refreshing their knowledge on Computer Architecture issues like Digital Design, Register Transfer Level and Computer System Level.

1. Introduction

The growing disparity between processor and memory speeds makes the cache memory and its efficient utilization a critical factor in determining program performance. There are many ways to improve the cache performance; one of them consists in the integration of a number of cache levels inside the processor chip, although it may cause a significant increment in energy consumption and problems with heat dissipation. Other solutions are based on the modification of the cache organization [2], optimization of cache algorithms [4] or improvement of data locality [1].

In this paper we propose a simulator tool of a reconfigurable cache that improves the efficiency by reducing the number of cache misses depending on the code running, and minimizes the energy consumption by using only the required modules in every moment.

The tool allows the students to study cache behavior statistics (number of read/write misses and hits) of a set of memory references for different cache organizations (direct-mapped, n-way set-associative) and block and cache sizes. That is to say, we are able to study the cache performance resulting from running the same code for different cache configurations.

2. Educational Objectives

The reconfigurable cache simulator has been used in several courses on Reconfigurable Architectures and Computer Design and Architecture in the University of Cordoba, Spain. There are three subjects in which this software is used: Computer Architecture and Engineering, Processor Design and Application Specific Architecture Design, all of them in the curriculum of Computer Engineer. Several lab exercises are prepared for the students on these subjects. This simulator is a helpful tool to reach some important aims in these courses:

- Introducing reconfigurable hardware as a way to improve the throughput and energy saving.
- Performing some practical exercises with reconfigurable hardware simulators.
- Studying in depth the cache memory parameters, such as, associativity, block size, replacement policies, writing policies…
- Reinforcing the knowledge acquired in previous related courses.
- Motivating the students on these subjects.

The IEEE-ACM Computer Science Curricula 2001 lists Computer Architecture (AR) as one of the subjects that should be in the curriculum of Computer Engineering, which should include several core materials. A reconfigurable cache simulator is a powerful tool to enhance the teaching of at least three of them, including digital logic and digital systems (AR1), memory system organization and architecture (AR4) and multiprocessing and alternative architectures (AR7).
3. Reconfigurable cache design

Our simulator tool assumes a reconfigurable cache embedded in a system with a 256 Mb 32-bit word main memory. The cache capacity (C) ranges from 256 bytes to 256 Kb. The block size (W) can be 1, 2, 4, 8, 16, 32 or 64 words. The number of blocks per set (L) can be 1, 2, 4 or 8. Every combination of C, W and L is permitted. The cache is unified, and the writing policy chosen is copy-back so there is a dirty bit in tag RAM to specify a modified block in cache. The reconfigurable cache is an inline cache. It means that a memory access causes a cache look-up and, if a miss occurs, a main memory look-up begins. In a look-aside cache the look-up process begins in the cache and in the main memory at the same time. It is faster in case of a miss, but the system bus stays occupied for a longer time. The replacement policy considered is a Pseudo-LRU [3], which is a variant of the LRU (Least Recently Used) algorithm that requires less memory use. The reconfigurable cache implements it in the LRU module depending on the associativity supplied by the CRR (Cache Reconfigurable Register) and other signals. The reconfigurable cache includes a dynamic mapping module that maps the address issued by the CPU dynamically depending on the content of the CRR. This can be set by a reconfiguration instruction and holds the current parameters of the cache: C, W and L. There is also a cache-main-memory interface that allows burst line transfers between cache and main memory that are faster than individual word transfers.

A more detailed description of the reconfigurable cache is in [5].

4. Reconfigurable cache simulator

The reconfigurable cache simulator includes a main user interface that consists of a Menu Bar, that provides access to every functionality of the application, a Toolbar, with the buttons to play and stop the simulation, a References Table, that shows the current set of memory addresses to simulate, a Statistics Chart, that shows the hit and miss rates, a Reconfigurable Cache Block Chart, that shows the reconfigurable cache structure in RTL, and a Status Bar. There are several windows to show the contents of the cache data and tag sections, as well as the state of the micro-programmed cache controller and the dynamic mapping module.

To run a simulation, a references file is needed containing the set of memory addresses issued by a program execution. A special instruction has been defined to reconfigure the cache system by setting up the parameters of the cache via a reconfiguration word. The “Cache Configurations Table” in the Help menu visualizes the reconfiguration word corresponding to the cache parameters C, W and L.

We may design different laboratory exercises for the students, for instance, finding the cache memory configuration that minimizes the read and write misses, finding the cache configuration that reduces the power consumption keeping a miss rate constant, and so on.

The simulator also includes a user’s manual that contains a comprehensive explanation of its functionality and the digital design of the modules in the reconfigurable cache block chart. With this information the student will be able to understand the complete operation of our reconfigurable cache simulation tool and will acquire a comprehensive view of an essential area of Computer Architecture. The simulator, called SRC, is free and available in the URL site http://www.uco.es/~el1hegoe/download/SRC.

5. Conclusions

In general, the main aims of the courses were achieved successfully, proving that our simulator is a motivating tool to introduce reconfigurable architectures and to reinforce the students’ knowledge in Computer Architecture.

Theoretical tests passed by the students in those courses concerning the memory system organization and architecture obtained higher scores than in previous years.

6. References