CCIT REPORT #735 June 2009

Performance and Power Aware CMP Thread Allocation Modeling

Yaniv Ben-Itzhak[1], Israel Cidon[2], Avinoam Kolodny[2]
Electrical Engineering Department, Technion, Haifa, Israel
[1]yanivbi@tx.technion.ac.il [2]{cidon, kolodny}@ee.technion.ac.il

Abstract

CMP is the architecture of choice for addressing the performance and power requirements of future computational systems. In this paper we address the problem of mapping software threads to coarse-grain multi threaded cores. Our goal is performance and power-efficient thread allocation in a CMP. To that end, we develop a parameterized performance/power metric that is adjusted according to a preferred tradeoff between performance and power and is based on an analytical model. We introduce an iterative threshold algorithm (ITA) for allocating threads to cores in the case of a single application with symmetric threads. We extend this to a simple and efficient heuristic for the case of multiple applications. We compare the performance/power metric value of the ITA with the results of several standard optimization methods. Our algorithm outperforms the best of these methods, while consuming less than 0.05% of the computational effort. Consequently, the ITA can serve as a basis for practical thread allocation in CMP systems, taking into account the preferred tradeoff between performance and power.

1 Introduction

The demand for more capable microprocessors has led CPU designers to explore various types of parallelism. Many applications are suited to thread level parallelism (TLP), and multiple independent processors can be used to increase a system's overall TLP [1][2]. The combination of increased transistor count due to advanced manufacturing processes and the inherent scalability limitations of a single processor design drove the introduction of chip multi-processors (CMP). CMPs offer higher power efficiency, better performance, and lower design complexity in comparison with other ways to achieve high performance in hardware. However, CMP architectures introduce new challenges associated with massive parallelism. This is emphasized by the fact that CMPs carry parallelism beyond its traditional super-computing and server markets to desktop, notebooks, PDAs and other mobile systems running a rich mix of heterogeneous applications and involving complex power and energy saving requirements. The heterogeneous and pervasive use of CMPs raises the need for different and tunable tradeoffs between system performance and power consumption. There is a clear difference between the requirements of a fan-cooled server in a server farm and a personal laptop computer. Battery operated mobile systems have a completely different operating point as compared to grid-powered equipment. Moreover, the same mobile equipment may have different performance-power settings according to its battery charge level, the set of applications that are currently executed and its desired residual operation time. Application examples can be watching a video during a flight, making a presentation and using the mobile equipment as a video camera or as a surveillance sensor.

In this paper, we address the important problem of efficient allocation of applications and threads to cores in a CMP, taking into account the desired tradeoff between performance and power, unlike works which deal with performance only [5]-[7] or power only [8].

We examine a system where several cores with a shared cache are interconnected by a network on chip (NoC) [2], for example the Piranha CMP [3] or Nahalal [4]. The CMP based system executes several multi-threaded applications. Each core performs coarse-grain multithreading. For reduced power dissipation and battery energy saving, any unused core may be shut down. Varying