Chapter 10
Tools and Techniques for Integrated Hardware-software Energy Optimizations

N. Vijaykrishnan, M. Kandemir, A. Sivasubramaniam, and M. J. Irwin
Microsystems Design Lab. Pennsylvania State University, University Park, PA 16802

Abstract: With the emergence of a plethora of embedded and portable applications, energy dissipation has joined throughput, area, and accuracy/precision as a major design constraint. Thus, designers must be concerned with both estimating and optimizing the energy consumption of circuits, architectures, and software. Most of the research in energy optimization and/or estimation has focused on single components of the system and has not looked across the interacting spectrum of the hardware and software. This chapter describes the design of energy estimation tools that support both software and architectural experimentation within a single framework. Furthermore, techniques that optimize the hardware-software interaction from an energy perspective are illustrated.

Key words: Simulation tools, energy estimation, kernel energy consumption, compiler optimizations, architectural optimizations

10.1 INTRODUCTION

Performance optimization has long been the goal of different architectural and systems software studies, driving technological innovations to the limits for getting the most out of every cycle. Advancing technology has made it possible to incorporate millions of transistors on a very small die and to clock these transistors at very high speeds. While these innovations and trends have helped provide tremendous performance improvements over the years, they have at the same time created new problems that demand immediate consideration. An important and daunting problem is the power consumption of hardware components and the resulting thermal and reliability concerns that it raises. As power dissipation increases, the cost of power delivery to the increasing number of transistors and the thermal
packaging for cooling the components goes up significantly [1][2]. Cooling systems need to be designed to tackle the peak power consumption of any component. These factors are making power as important a criterion for optimization as performance in commercial high-end systems design. Similarly, energy optimization is of importance for the continued proliferation of low-end battery-operated mobile systems. Unless optimizations are applied at different levels, the capabilities of future mobile systems will be limited by the weight of the battery required for a reasonable duration of operation.

Just as with performance, power optimization requires careful design at several abstraction levels [3]. The design of a system starts from the specification of the system requirements and proceeds through several design levels spanning across architectural design, logic design, and circuit design, finally resulting in a physical implementation. Energy savings can be obtained at all levels of design, ranging from low-level circuit and gate optimizations to algorithmic selection. In earlier design methodologies, cycling through logic synthesis and physical design was used as the main iteration loop to refine and verify a design. This method, however, is not keeping up with the complexity of today’s system-on-chip designs. By the time the design of today’s large and complex designs have been specified to the circuit or gate level, it may be too late or too expensive to go back and deal with excess power consumption problems. Also, various architectural alternatives need to be explored since achieving an optimal design during the first design iteration is very difficult in complex designs. Thus, system designers need advanced techniques and related tools for the early estimation of power dissipation during the design phase in order to quickly explore a variety of design alternatives at the architectural/RT Level.

The increasing need for low-power systems has motivated the EDA industry to introduce power-estimation tools for the various design levels. A number of commercial tools are routinely used to accurately estimate the power of portions of million transistor architectures represented as transistor or gate-level netlists. However, relatively few commercial tools exist to support the RT-level estimation essential for design space exploration. The most mature among these tools is Watt Watcher/Architect from Sente that operates on a RTL description. It uses a gate-level power library and simulation data from an RTL simulation to compute a power estimate for the design. Accuracy is traded for improvements in run time. Only prototype research tools/methodologies exist to support the behavioral or architectural design level. The successful design and evaluation of architectural and software optimization techniques are invariably tied to a broad and accurate set of rich tools that are available for conducting these studies. While tools for analyzing optimizations at the circuit and logic level are fairly mature,