

Guest Editors' Introduction: Designing Real-Time Embedded Multimedia Systems

Radu Marculescu

Carnegie Mellon University

Petru Eles

Linköping University

■ **EMBEDDED MULTIMEDIA SYSTEMS** represent an important segment of today's electronic industry. Although there has been a notable growth in the use of such systems, their design process has become remarkably difficult because of increasing design complexity and shorter time to market. Without significant changes in the design methodology, designers will be able to exploit less and less from the potential that emerging technologies have to offer.

Among the most promising solutions to the design complexity problem are design platforms consisting of hardware and software resources that are shareable across multiple multimedia applications. Such generic platforms consist of dedicated processing resources (such as ASICs) and programmable processors (such as general-purpose or DSP processors) that can operate together and run the target application (for example, MPEG-2 audio/video, e-mail, or Web browsing). For modern multimedia systems with many heterogeneous components that interact and communicate over a network, early estimation of complex performance metrics and quality-of-service (QoS) based management are the critical steps for judicious allocation of the available resources.

This special issue addresses some fundamental problems in the design and optimization of modern real-time multimedia systems and illustrates the potential design trade-offs and their impact on media quality. Over the years, the design community has witnessed a constant transition from stand-alone (or desktop) multimedia to distributed multimedia systems. Most notably, with the transition from desktop to portable multimedia, communication and power consumption become crucial

to system-level analysis and optimization. As such, the design issues change significantly: Whereas desktop-based systems are mainly optimized based on performance constraints, power consumption becomes the key design constraint for multimedia devices that draw their energy from batteries. Consequently, it is crucial to maximize the operating time between successive recharge cycles and thus efficiently use the available energy. Examples include mobile personal multimedia systems such as MP3 players, PDAs with built-in cameras and gaming features, and cell phones; but also include nonpersonal devices such as sensor networks.

Multimedia systems represent a very special class of complex computing systems. Consequently, a multimedia system's design process should start by considering its unique characteristics, which are dominated by the huge amount of data it must continuously process and transmit. Another important characteristic is that QoS embraces all the nonfunctional properties (such as power consumption, timeliness, jitter, and cost). In multimedia systems, QoS requirements vary considerably from one media type to another. For example, because of the large amounts of data they must process, video streams require consistently high throughput but can tolerate reasonable levels of jitter and packet errors. In contrast, audio applications manipulate a much smaller volume of data (and therefore do not require such a high bandwidth), but place tighter constraints on jitter and error rates.

This special issue

Simply speaking, designing a multimedia system involves mapping the target application onto a given

implementation architecture while satisfying a prescribed set of design constraints (for example, power, performance, and cost). Finding the best match between the application and the underlying architecture implies several specific actions with direct impact on the end-to-end latency, power consumption, achievable throughput rate, and so forth. To start with, providing enough buffering space is crucial for media quality. As the article by Im and Ha shows (pp. 358-366), frame skipping and buffering can exploit the characteristics of video applications, particularly the tolerance to latency and video quality, to increase slack times. Im and Ha further demonstrate how to employ these slack times to save energy, using dynamic voltage scaling (DVS) techniques.

Choosing the appropriate scheduling technique is another crucial issue in satisfying the high computational and real-time constraints of processing multimedia streams. Maxiaguine et al. (pp. 368-377) present an analytical framework for the evaluation of schedulers designed for SoC multimedia platforms. The modeling technique the authors propose is general enough to subsume standard event models used in traditional real-time scheduling and quite accurate in capturing the variability in task execution requirements. This analytical framework is more accurate and orders of magnitude faster than traditional simulation-based approaches.

The importance of scheduling policies and memory allocation in the context of energy-efficient embedded systems has received wide recognition. However, designers have yet to explore the interplay between these two aspects. The article by Marchal et al. (pp. 378-387) proposes a runtime data assignment technique for synchronous dynamic random access memories (SDRAM) in dynamic multithreaded multimedia applications. This technique, combined with task scheduling, can minimize the energy cost and the number of deadline violations.

Although significant research efforts have focused on optimizing power consumption at application, network, and processor levels, much less work has focused on reducing backlight power consumption. Two articles in this special issue address this problem. Shim, Chang, and Pedram (pp. 388-396) introduce a backlight power management framework for color thin-film transistor (TFT) liquid-crystal display (LCD) panels that is useful in battery-operated multimedia applications. This framework extends dynamic luminance scaling to cope with transfective LCD panels that operate both with and without a backlight, depending on the remaining bat-

tery energy and the ambient luminance. The authors explore the extended dynamic luminance scaling (EDLS) design space in which the application transparency and hardware-software partitioning exhibit trade-offs in terms of energy reduction, energy overhead, performance penalty, and image quality. Taking a more network-centric perspective, Pasricha et al. (pp. 398-405) propose an adaptive middleware-based approach to optimize the backlight power consumption for mobile handheld devices when playing streaming video. This approach simultaneously reduces the backlight power consumption and minimizes any negative impact on the perceived video quality.

In a network of battery-powered multimedia systems, designers must account for both computation energy for processing the data stream and communication energy for transmitting or receiving the data. The computation energy is usually a strong function of the clock frequency of the multimedia system, which can be varied by using dynamic voltage and frequency scaling (DVFS). The communication energy, on the other hand, strongly affects the bit error rate (BER) and, thereby, the received data quality. The article by Delaney, Simunic, and Jayant investigates the energy consumption (spent in computation and communication) of a distributed speech recognition embedded system. (This article will appear in the Jan.-Feb. 2005 issue of *IEEE Design & Test*.) The authors propose concrete optimization techniques—at application and network layers—that reduce the overall energy consumption while maintaining adequate QoS for the end user.

Along with the issue of energy optimization, copyright protection of sensitive multimedia plays a significant part in the design of multimedia systems. Lekatsas et al. (pp. 406-415) introduce a hardware platform capable of performing both data/code compression and encryption in a unified architecture. Besides being a parameterizable hardware architecture, this platform features a suite of software tools that allows for evaluating and optimizing specific multimedia applications.

Future multimedia systems will become increasingly complex as the demand for functionality increases steadily. Toward the end of the decade, when the complexity of SoCs will reach more than 1 billion transistors on a single die, hundreds of heterogeneous processors might be integrated on the same chip. Using a design methodology that allows extensive design exploration for hardware-software codesign and improving the overall use of reconfigurable platforms for multimedia might turn out to be key factors in cutting down the produc-

tion costs, as La Rosa, Lavagno, and Passerone discuss in their article. (This article will appear in the Jan.-Feb. 2005 issue of *IEEE Design & Test*.) However, at this level of complexity, communication becomes a major concern as the inability to scale in terms of performance and power consumption makes traditional bus-based architectures ineffective. Several researchers recently proposed a network-on-chip approach based on regular architectures as a possible solution to mitigating the complex on-chip communication problems. Such a chip consists of regular tiles, where each tile can be a general-purpose processor, a DSP, a memory subsystem, and so forth. A router embedded within each tile has the objective of connecting the tile to its neighboring tiles. Thus, instead of routing design-specific, global on-chip wires, the intertile communication operates through routing packets. Lv et al. present a methodology for the architectural design of such multiprocessor SoCs; they illustrate its benefits with an architectural analysis of a generic implementation for a real-time gesture recognition application. (This article will appear in the Jan.-Feb. 2005 issue of *IEEE Design & Test*.) The proposed methodology covers all the important architectural aspects, from processor allocation to the memory and network design.

WE HOPE that the selection of articles in this special issue provides good coverage and a representative set of results on design issues of modern multimedia systems. It was a great pleasure and privilege for us to act as guest editors for this special issue and to collaborate with the always helpful and efficient *D&T* editorial team. In particular, we would like to acknowledge the support we received from Editor-in-Chief Rajesh Gupta, Editorial

Assistant Anna Kim, and Magazine Assistant Kimberly Merritt. Last but not least, we would like to thank the authors and the reviewers for their contribution and help in making this special issue possible. ■



Radu Marculescu is an associate professor of electrical and computer engineering at Carnegie Mellon University. His research interests include system-level design methodologies and software tools for SoC design, on-chip communication, multimedia, and ambient intelligence. Marculescu has a PhD in electrical engineering from the University of Southern California. He received the National Science Foundation's Career Award for design automation of electronic systems in 2001 and the Carnegie Institute of Technology's Ladd Research Award in 2002. He also received best-paper awards from the 2001 and 2003 Design, Automation and Test in Europe (DATE) Conferences and the 2003 Asia South Pacific Design Automation Conference (ASP-DAC). He is a member of the IEEE and the ACM.



Petru Eles is a professor in the Department of Computer and Information Science at Linköping University, Sweden. His research interests include the design of embedded systems, hardware-software codesign, real-time systems, system specification and testing, and CAD for digital systems. He has published extensively and coauthored several books in these areas; he received the best presentation award from the 2003 IEEE/ACM/IFIP International Conference on Hardware-Software Codesign and System Synthesis. Eles has a PhD in computer science from the Politehnica University of Bucharest, Romania. He is a member of the IEEE and the ACM.

■ Direct questions and comments about this special issue to Radu Marculescu, Dept. of Electrical and Computer Eng., Carnegie Mellon Univ., Hamerschlag Hall 2108, 5000 Forbes Ave., Pittsburgh, PA 15213-3890; radum@cmu.edu.

For further information on this or any other computing topic, visit our Digital Library at <http://www.computer.org/publications/dlib>.

**Members
save 25%**
on all conferences sponsored by
the IEEE Computer Society.
Not a member?
Join online today!
computer.org/join/