Invited Talks

Is IP Business Hype or Reality?

D. D. Gajski
University of California at Irvine, USA

Abstract: Selling and using IPs seems to be the perfect solution for improving design productivity. However, IP business models may not have adequate infrastructures with which to flourish. This talk will elaborate on new design styles, modeling guidelines, methodologies, and CAD tools needed for successful reuse.

Issues in Embedded DRAM Development and Applications

Doris Keitel-Schulz
Siemens Research and Development, Germany

Abstract: The term system-on-silicon has been used to denote the integration of random logic, processor cores, SRAMs, ROMs, and analog components on the same die. Until recently, however, one major component had been missing: high-density memories such as DRAMs. Today's integration densities are beginning to allow the integration of significant amounts of DRAM memory. Embedding DRAM can therefore be considered a key technology for certain applications such as data buffering, picture storage, and program/data storage. Fast on-chip buses between DRAM and logic, low-power dissipation, fewer pins, higher integration density and optimized memory structures are among its major advantages. In this talk we discuss some of the challenges associated with embedded DRAM design in greater detail.

Processor Architectures and Compilers for Systems on Chips

Monica Lam
Stanford University, USA

Abstract: As it becomes possible to integrate an entire system on a chip, the processor architect is presented with an unprecedented opportunity to tailor the processor to the application at hand. To fully realize the potential of this technology, it is critical for us to be able to quickly produce an optimized processor design and associated programming tools. Languages and compilers have an important role in the revolution of processor IP in ASIC designs. Many of the advanced architectural ideas (parallelism, memory subsystem optimizations) originally developed for high-end processors are directly applicable to this environment. They often require nontrivial compiler support. Opportunities for application-specific optimizations can be exposed by means of more expressive programming languages. Finally, program analyses can extract program characteristics that can be used directly in guiding the architecture customization process.