PANEL: SYSTEM SYNTHESIS: CAN WE MEET THE CHALLENGES TO COME?

Chair: Robert A. Walker - Rensselaer Polytechnic Inst., Troy, NY
Organizer: Robert A. Walker - Rensselaer Polytechnic Inst., Troy, NY

In the past few years, the field of high-level synthesis (also known as behavioral synthesis) has begun to mature, and to blend with hardware / software codesign, and the design of embedded systems. Thus a new field is evolving -- *system synthesis*. But as we move toward the next millennium, will even system synthesis be enough? Can we develop the necessary tools for the designers of the hottest processor chip in the year 2000, or the next great cellular phone or PDA? Just what tools do those fields need, and how close are we to supplying those tools? *Can we meet the challenges to come*??

Daniel D. Gajksi, Univ. of California at Irvine

No advance position statement.

Raul Camposano, Synopsys, Inc., Mountain View, CA

Electronic systems will be increasingly dominated by so-called "systems on a chip", which integrate digital and analog parts as well as predesigned macros, custom and semicustom blocks. Two major, partly opposite trends, characterize the design of such systems. On the one hand, there is a move towards higher levels of abstraction (mainly to deal with complexity and to meet productivity and time to market requirements). On the other hand, deep submicron technologies make it necessary to take into account more and more physical information in the design of such systems (mainly to meet quality of result requirements such as area, speed and power). Constructing design tools and methodologies that allow to cope with both trends will be the main challenge of the future.

Pierre Paulin, SGS-Thomson Microelectronics, Crolles, France

What is the MSDOS of the consumer electronics future? In the personal computer area, MSDOS and x86 object code have become de facto standards. In embedded systems for high-volume consumer electronics, a different situation is emerging: (1) multiple sub-system standards are emerging (MPEG2 audio, video, Dolby audio, H.263 videophone, digital wireless, etc.), (2) they are invariably described as ANSI C executable specifications, and (3) a large part of the function will be executed on an embedded processor (in most cases, an application-specific instruction-set processor (ASIP)). As a result of this situation, we need (1) computer-aided exploration of ASIP architectures, and (2) high-performance C compilation (ideally, from the executable C specification of the standard). The bottom line is that the winners will be those that can map high-level C descriptions onto a low-cost processor the fastest.

Laurent Bergher, Thomson Consumer Electronic Components, Meylan France

For multimedia design in consumer applications (in particular, the digital audio field), the most important points are time-to-market, low-cost product, and design efficiency. We have developed a methodology for digital audio design where we use a subset of C to write the application code and define the architecture, and use the UNIX profiler to control the real-time capacity of the architecture with embedded software. This methodology has proven very efficient with the previous generation of audio decoders, but now, with the increase in algorithm complexity (especially with surround audio), we need (1) tools to allow the designer to explore the effect of different instruction set selections and/or encoding schemes, and (2) very high performance C compilers that provide all timing profile information and source-level debugging.

Barry Shackleford, Mitsubishi Electric Corp., Yokohama, Japan

The vast majority of designers are not working on products like "the hottest processor chip in the year 2000" (products like that tend to be handcrafted, anyway); they are working on rice cookers (this is Japan, after all), TVs, vacuum cleaners, and washing machines -- all with "fuzzy / neural" controllers. In short, they are building systems. A system designer who is worrying about the bus protocol or the internal intricacies of a microprocessor is diverting precious creative energy away from the primary task of designing a system's function. We need to develop a new generation of design tools aimed at system synthesis, where, for example, a microprocessor is just another function block sized to fit the application at hand, just as adders are by logic synthesis tools today.

Randy Steck, Intel Corp., Hillsboro, OR

Synthesis has made great progress over the last several years, providing better densities, speed, and power optimizations. Unfortunately, it is not enough. Behavioral synthesis is virtually unknown in high performance, highly constrained design, and lower level synthesis is breaking down trying to keep up. The Pentium Pro processor design required more effort per transistor to be expended on synthesized logic than on datapath design. Future microprocessors and necessary support designs will require much higher frequency operation, which in turn drives needs for even better density. In addition, new process technologies require consideration of new factors in circuit design. These needs will require substantial development if they are to be met.