PANEL: THE ROAD AHEAD IN CPLD & FPGA DESIGN METHODOLOGY

Chair: Rhondalee Rohleder, PACE Technologies, Scottsdale, AZ
Organizer: John Birkner, QuickLogic, Sunnyvale, CA

Large capacity CPLD and FPGA design opportunities and challenges are explored by this panel. As densities grow beyond 20,000 gates, there is a disconnect between the cutting-edge EDA tools featured in trade shows and the more ordinary tools used by the typical programmable logic designer as s/he moves to a higher level design methodology. A three-to-five year roadmap will guide this discussion of the future of synthesis and simulation, the impact of HDLs on complex devices, and the role of intellectual property and universal tools in the CPLD & FPGA designscape.

Don Faria, Synopsys, Inc., Mountain View, CA
PlD design is beginning the same paradigm shift that drove the success of logic synthesis within ASIC design, namely the move from schematics to HDL-based design tools and methodologies. HDL designers will grow from 12,000 in 1996 to over 70,000 by the year 2000, driven by a 'second wave' of mainstream programmable logic designers adopting HDL methodologies. The demands of synthesis technology for PLDs are quite different than for ASICs. PLD designers demand very quick design cycles and are constantly trying to squeeze every drop of logic capacity and performance from the devices. The power of high-level design must be customized to the needs of this broad design community. Ease-of-use must be addressed through a graphical user interface that allows the designer to focus on the design. High quality results must be obtained through detailed architecture knowledge built into the optimization technology. Addressing these needs concurrently will provide designers the power necessary to continue to take advantage of the benefits of PLDs.

Mike Dini, The Dini Group, La Jolla, CA
Gone are the days when PLD designers can simply plug in a PAL and debug it with a scope. High density PLDs require HDL entry and synthesis which brings a new set of problems. Synthesis run time, for example can impact the design approach when the time exceeds 1000 gates per minute. Formal verification may be required to prove correctness of lock-up-sensitive state machines. Synthesis changes and/or loses signal names, complicating the debug and trace process. EDA vendors must provide better feedback into the synthesis process.

Steve Golson, Trilobyte Systems, Carlisle, MA
The Road Ahead for PLD design flow has been paved by ASIC design tools. The artificial distinction between PLD and ASIC design flow will vanish as PLD tools change to conform to ASIC methods. It will be one world of seamless design tools, where the decision to target either PLD or ASIC is a simple switch for the entire design flow. PLD designers will begin to use simulation (kicking and screaming). Counting gates and squirming around with CLB fitting will vanish as PLD tools get on to the ASIC bandwagon.

Bob Donaldson, Annapolis Micro Systems, Annapolis, MD
Reconfigurable processing is quickly moving from a niche to a commercially mainstream technology. Reconfigurable processing development systems are now available and utilize HDLs, replacing schematic design, which has become obsolete as FPGAs have grown in size. Reconfigurable processing is suitable for many applications such as bit manipulation, correlation and DSP. In one example, 20 DSPs on a board were replaced by one FPGA. Current FPGA architecture directions are improving, but still diverse. EDA place and route tool run times need to move from hours to minutes. Synthesis tools need to mature to a more ASIC like design flow.

Dave Kohlmeier, Synario Design Automation, Redmond, WA
Silicon vendors are delivering massive numbers of gates in their new CPLD and FPGA devices. To fully utilize and efficiently implement these new devices, a robust integrated set of tools is needed that supports top-down design, including advanced synthesis, behavioral simulation, timing analysis and more. While many companies offer point tools, engineers today are busier than ever, and don't have the time to integrate new and complex software products together. What they need are "Ready-to-use" solutions -- complete with best-in-class tools, built-in design flows, design examples, templates, and whatever utility programs or translators are necessary to stitch together a complete design environment. "Ready-to-use" solution providers need to have experienced designers on the hardware development team, acting as champions for the hardware designer.