CMP
United Business Media
EETIMES ONLINE

EE Times:

# Panelists explore system-to-silicon link

Richard Goering
(05/31/2006 10:53 AM EDT)
URL: http://www.eetimes.com/showArticle.jhtml?articleID=188700166

PRIEN, Germany — How can designers produce manufacturable silicon from system-level specifications? Panelists at the Medea+DAC (Design Automation Conference) here had some suggestions Tuesday (May 30), including formal specifications, abstracted hardware-software interfaces and the use of regularity in silicon fabrics.

Panel moderator Joseph Borel, retired vice president of STMicroelectronics, noted that total development costs for 65-nm chips are nearing $40 million, making it more important than ever to get designs right the first time. He called for "applications design platforms" in which formal specs can be reduced to RTL design, and physical layout is aware of design-for-manufacturability (DFM) issues.

Eric Bantegnie, president of Esterel Technologies, presented formal modeling languages as an enabler for electronic system level (ESL) design. "You can automate the ESL to RTL link if you build ESL on solid formal foundations," he said.

Existing modeling languages are inadequate for complex behavior, Bantegnie said. He noted that C can perform sequencing, but describing concurrency or temporal behavior is problematic. HDLs do better with concurrency, but sequencing, state machines and temporal behavior are difficult. What's needed, he said, are usage-oriented languages that provide a direct, hierarchical behavioral description and a native understanding of continuous control, signal processing and state machines.

With a formal modeling language, Bantegnie said, automatic code generation is possible. From a single formal model, he said, designers can get an efficient, synthesizable HDL model, SystemC code for simulation, embeddable C code and formal verification input. His company's Esterel Studio provides formal synthesis and compilation into HDLs, SystemC and C.

Formal modeling languages are mature, apply to both hardware and software design and scale to real-world applications, Bantegnie said. "These languages are not meant to replace things," he said. "You still retain SystemC and HDLs. But now you have a way to express formal specifications early in the design process."

Ahmed Jerraya, research director for system-level synthesis at the TIMA research institute, discussed the transition from formal specs to transaction-level modeling (TLM). He stressed the importance of analyzing multiprocessor systems-on-chip as complete designs, including both hardware and software. "The key problem we need to solve before moving to ESL is how to abstract hardware-software interfaces," Jerraya said. "We know how to code hardware and software, but we don't know how to abstract the interfaces."

Classical design, noted Jerraya, looks at hardware modules and interconnects only. The TLM approach abstracts hardware intellectual property, abstracts hardware channels and uses instruction-set simulation and binary software. The next step, said Jerraya, is to extract not only software and hardware IP, but software-hardware interfaces as well. One challenge for such an abstraction is hardware-dependent software, which includes low-level behavior such as interrupts.

Jerraya noted that there are a number of abstraction levels above TLM. At the lowest level, most everything could be explicit: binary software, an RTL CPU, RTL hardware, and physical memory. At higher levels, software could be native, and there could be a bus-functional model interface and a high-level CPU simulation.

Wolfgang Rosenstiel, professor at the University of Tubingen (Germany), discussed the benefits of TLM design at varying levels of abstraction. He noted that system-level design works only when it's possible to track cost, performance and power throughout the design flow. System-level design tasks must be integrated, he said.

TLM benefits, said Rosenstiel, include the development of hardware-dependent software, architectural exploration and platform management. With platform management, he said, it should be possible to incorporate specification

environments such as Unified Modeling Language (UML) or Matlab, generate virtual prototypes from abstract specifications and finally merge different system perspectives into a joint executable SystemC model with requirements, hardware-dependent software, middleware and hardware.

Hardware-dependent software, said Rosenstiel, requires early software performance evaluation, platform optimization and software refinement. The software refinement tasks include automated target code generation, optimized mapping and scheduling of tasks for multiple cores, power-aware software compilation and software integration.

"We need simulation and debugging for parallel software on multicore hardware, but it will only be possible if we have more abstract models," Rosenstiel said.

Architectural exploration, Rosenstiel said, requires system-level communication analysis and optimization; system-level power optimization with interconnect; and multiple levels of TLM abstraction. He suggested TLM as a new intermediate level between specification and RTL. Current TLM abstraction levels, he noted, include untimed, timed, cycle-approximate, cycle-accurate and RTL.

Marcello Coppola, corporate strategy and system technology manager at STMicroelectronics, wrapped up with a look at the RTL-to-implementation link. One key to success, he said, is design "regularity." That's been an enabler of every quantum step in design productivity, he said, and it both reduces costs and enhances the manufacturability of systems.

Design complexity and growing variability mandate new approaches to computing architectures and design platforms, Coppola said. These will require new architectures that scale at 45 nm and below, communication-centric approaches such as network-on-chip (NOC) and new design tool technologies, he said.

At 65 nm and below, Coppola said, designs will require both manufacturing robustness and system-level predictability. One result will be an increasing move to programmable fabrics. Coppola called for a new approach that will provide a "middle ground" between today's ASICs and FPGAs in terms of performance and implementation cost.

Coppola also said the design productivity gap between silicon and EDA tools can be closed with the addition of synthesized IP assembly, regularity, NoC and silicon implementation platforms. Regularity will also help solve DFM problems and provide higher yields, he added.

The Medea+DAC workshop is an annual event sponsored by Medea Plus, a government-sponsored pan-European initiative that funds advanced research in microelectronics design. Medea Plus recently released a 2005 EDA roadmap.