PANEL: CORE-BASED DESIGN FOR SYSTEM-LEVEL ASICs - WHOSE JOB IS IT?

Chair: Lynn Watson - In-System Design, Meridian, ID
Organizer: Rich Goldman - Synopsys, Inc., Mountain View, CA

As the shift from simple synthesis to full-scale leverage of large intellectual property blocks (cores) continues to gain momentum, the boundaries of design responsibility begin to be blurred. Silicon quickly becomes a product of very diversified lineage. ASIC vendors' cores, CAE companies' macro-cells, third-party contributed IP, and, of course, the implementer's HDL must all now play together seamlessly. Often the bulk of the design work moves away from original contribution to interfacing - trying to smooth the uneven joints of these juxtaposed blocks.

As both sides look at an incompatible interface boundary, the natural question arises: "*Whose responsibility is it to change*?" While either or both sides can change, more often than not the differences are bridged by additional logic introduced by the implementing designer. This often is not the optimal solution for either area or timing constraints.

Even more imposing, the question is further complicated by the issues of validation strategy, simulation environments, format of the property, etc. With large portions of the functionality of the part now being instantly provided, a great portion of the implementer's time is shifted from original design work to validation of someone else's design -- more time spent on portions of a design that potentially look like a "black-box".

Some of the questions to be addressed by the panel:

- How are cores effectively designed-in today? Where have responsibility lines been traditionally drawn?
- How can architectural exploration be done in the early stages of design with large cores? Are multiple representations (C-language, behavioral, RTL, gate) of the IP a requirement for an efficient design process?
- Who provides and modifies vectors associated with a dropped-in core to assure coverage goals in-situ? As validation efforts push into the unknowns of cores, how are questions resolved? Whose responsibility is it to reconcile differences that show up in the course of stitching these cores together?
- Is intellectual property protection an issue? Are there sufficient mechanisms in place with the current set of EDA tools?
- Is it reasonable to expect that the core provider (typically an ASIC vendor) work with each application to modify interfaces, add or subtract features in order to arrive at an optimal implementation? Could cores be provided in a customer-modifiable format (synthesizable or otherwise)? If so, what then is the core provider's support responsibility?
- For µP, DSP, and other cores that have intense real-time debug requirements, who should address the issues associated with observability/controlability of the core in the silicon?

Panel Members:

Kim Asal - Texas Instruments, Austin, TX
Andreas Danuser - Ascom Tech., Berne, Switzerland
Chris King - IBM Corp., Waltham, MA
Susan Mason - Information Architects, Los Altos Hills, CA
Jim Pena - LSI Logic, Milpitas, CA
Scott Runner - Synopsys, Inc., Beaverton, OR