Panel Discussion: Is There a Crisis in Hardware Verification?

(Date and time: October 18, 1997, 3:00 - 4:30 p.m.)

Chairman: Francisco Corella (Hewlett-Packard)
Members: Bogdan Boruslawski (Nortel),
Jerry R. Burch (Cadence),
Roger Hughes (Abstract Inc.),
Isadore Katz (Chrysalis Design),
Carlos M. Roman (Lucent Technologies).

Technology, architecture, and application trends are clear for all to see. Many have observed the increasing complexity of hardware designs, and the presence of higher-level functionality on chip. Design (i.e. product) cycles have become much shorter. Both for reasons of avoiding catastrophic failure, and because the cost of fabrication is so high, an increasing level of a priori and a posteriori confidence in hardware designs is required. What resources are being expended by industry towards this goal (for example, how much money is HP/Intel spending to verify the Merced processor?). More critically, are the formal tools and methods that are available to industry adequate to this task?

Software systems are debugged by the user community over time. Every software product is followed by a potentially infinite sequence of patches. The same is true of hardware products, including processors, although sometimes compilers help us with the work arounds. It seems clear that hardware designers will be using formal verification both within and between levels, and test vectors, for the foreseeable future.

There are perhaps two problems that industry has. First, hardware is described at too low a level to support algorithm correctness even in "high-level" HDLs. For example, consider the way in which concurrent state machines ("processes") are expressed in Verilog or VHDL. Still, this description is better than gates for large systems.

Second, the semantic gap between high-level HDL, even RTL, and netlists is already too wide for today's synthesis tools, due chiefly to the overwhelming importance of locality in chip layout. The compiler technology incorporated in the synthesis tools has been weak in certain areas, vectorization for example, and that hasn't helped. The result is that high-performance systems such as microprocessors have been using low-level descriptions and "data-path compilers".

So industry lowers high-level design specifications by hand, and formal verification's chief role is to make sure that it has been done right.

Many designers now feel that current hardware-description languages cannot capture the hardware requirements specification (i.e. the hardware designer's abstract machine) because of inadequate expressive power in the description languages. It is not that it is hard for us to verify the implementation of what we can say in VHDL or Verilog. Rather, it is hard for us to say (i.e. specify) what we need to say in either language. The consensus is rapidly growing that industry needs a new hardware-description language.

Disclaimer: Since these lines were written before the panel was convened, they do not necessarily represent the views of individual panel members. Panels are more lively and more focused when they have a clear position paper to react to.