global sources
EE Times India
India's fortnightly focus on electronics design
October 16-31, 2008

Tools manage verification data reports

By Rahul V. Shah
Project Manager, ASIC
eInfoChips Ltd

Rapid advancements in technology have increased the functionality of SoCs and reduced die size and time-to-market. While it may sound exciting to consumers, the impact of this growth is antagonistic in the verification domain. The challenges posed by the new verification environments are far more complex than what they were couple of years ago.

Until recently, most verification environments were created using traditional Verilog or VHDL based HDL. A good mix of Perl and basic randomisation used to be sufficient for the requirements of chip level verification. As verification code size grew and chip dimensions shrunk, the industry made the wise decision to adopt some good practices from the software domain.

Aspect Oriented Programming (AOP) was adopted not only to enable reusability of code, but also to learn from and leverage numerous software concepts for managing the growing code size of verification environments. Constrained random environment was introduced, opening up a completely new world of reducing code size. We started utilising controlled random stimulus to get maximum meaningful random stimulus for good verification confidence. More HVL languages like SystemVerilog were launched using Object Oriented Programming (OOP) concepts, once again aligning the verification industry towards good practices followed by the software industry. Code reusability is the new "mantra" backed up by methodologies like VMM, OVM, and the like.

Shah: We try to control verification code with given methodologies and languages, but even our reduced code gives birth to additional verification data to be analysed and managed.

The focus is constantly on reducing code size, but it does not imply a reduction in the stimuli required to verify the chip. We try to control verification code with given methodologies and languages, but even our reduced and more efficient code gives birth to additional verification data to be analysed and managed as a necessary part of the chip verification cycle.

Other challenges

There is, furthermore, the increasing number of third party IP components in today's SoCs. Reduced time-to-market and reusability are the major drivers for semiconductor companies to integrate third party proven IP into their system. While it saves cost and valuable effort,

continued on page 3 ▶

'Safari' through SoC verification

By Amit Sharma
Manager
Corporate Applications Engineering
Srinivasan Venkataramanan
Senior Staff CAE
Synopsys India

To state the obvious, verification of an advanced multi-million-gate system on chip (SoC) is a complex activity. Today's SoCs typically use multi-core architectures with numerous peripherals. Verification environments for such SoCs are inherently heterogeneous in nature due to the involvement of different abstraction level models (such as TLM, RTL, gate-level, etc.) and different languages (such as C, C++, SystemC, Verilog, VHDL and SystemVerilog). Gone are the days when a plain HDL simulator with a robust PLI interface as an optional plug-in was all that was needed for verification. Given the different characteristics of various parts of a SoC, orthogonal verification technologies (such as coverage, assertions, PLI applications, and debug) are needed to achieve comprehensive verification. A comprehensive verification platform must encompass all these technologies natively to minimise the cost of ownership for SoC teams.

The sheer size of these SoCs can throttle the memory requirements of a verification platform. To keep up, a verification platform should provide smallest memory footprint along with fastest run time. A platform must also pro-

continued on page 7 ▶

Inside