cadence™

Interview: SiP16.0 extends RFSiP Implementation to Parasitics/Simulation

From cdnusers.org on 5 December 2007

Closing in on Profitability with Leading-Edge Verification Practices
Design and verification engineers use a variety of methods to verify advanced designs and build verification environments. The question is whether there's a best way to do it, or are there a number of acceptable ways? To answer this question, Cadence® engineers have developed technology that includes best practices and guidelines that close the gap on predictability and profitability for complex SoC and full-chip designs.

This technology—the Incisive® Plan-to-Closure Methodology—is included in Incisive Enterprise Simulator, Design Team Simulator, Specman Elite® Testbench, and Specman ESL solutions. However, not everyone who has the methodology is using it, so cdnusers talked to Cadence Engineering Group Directors Mike Stellfox and Erik Panu to find out when and how to deploy the new technology.

cdnusers: Why should someone use the Plan-to-Closure Methodology for verification?
Erik: The goal of the Plan-to-Closure Methodology is to capture the best-known verification practices that, in our experience with our customers, have been effective over time. One of the key benefits of the methodology is that it addresses verification reuse, which is critical in today's designs. As a designer, I can't accurately verify a design on a tight schedule if I have to start from scratch every time. Teams want to share and reuse their components with each other. If design environments are not built in the same way, you can't do that effectively—you have to rewrite, which is very costly in terms of time.

Mike: Also, the number of different tools and techniques available can be overwhelming. How do you decide which is the best tool for your process—formal vs. simulation vs. emulation—to tackle a specific problem?

With the Plan-to-Closure Methodology, guidance is provided as to when and how to apply each technology with the associated methodology in order to optimize the effort of finding and correcting design bugs. We have combined the best methods to help designers and verification engineers capture and plan appropriately so they can leverage the right technology for the right type of problem at the right phase in the project based on what type of design they are implementing.

cdnusers: How do you decide what constitutes a best practice?
Mike: By working directly with many customers across the world as they do functional verification, we are able to see what methodologies work and use that information to influence the tool flow, which results in an optimized approach. When customers develop a methodology, they do not have the luxury to impact the tool flow to match the methodology as efficiently as possible. We believe that, in order to provide the best verification technology, we must first have a deep understanding of verification methodology best practices to ensure our tool flow provides the easiest path for engineers to successfully apply that methodology.

cdnusers: You have included the Plan-to-Closure Methodology in Cadence Incisive technologies. How many designers and engineers are using it?
Erik: To answer that, I have to explain that the Plan-to-Closure Methodology spans the entire verification process, from planning to closure. There are four core components: verification planning and management; assertion-based verification; testbench automation and reuse; and system verification.

Many engineers who are working with complex SoC and full-chip designs are using the testbench

and reuse methodology. However, they are not yet taking advantage of the entire breadth of the solution. There are a lot of benefits for designers and engineers who are ready and willing to expand into these other component methodologies.

cdnusers: Why are more people adopting the testbench component?
Mike: Until now, the main workhorse of verification has been testbench simulation; in effect, it is still the main workhorse in the flow. So designers are comfortable with a testbench simulation approach. But today's large, complex designs are so challenging that everyone is asking, how do we further optimize the process?

The answer is to adopt other aspects of Plan-to-Closure into your verification flow—assertions, for example. Before the testbenches are even built, designers can write assertions and leverage our formal verification technology to quickly validate their blocks and find the bugs very early—the earlier you can find bugs, the easier it is to debug and move on.

Erik: I agree. I would add that anyone just starting to use the Plan-to-Closure Methodology should start not just with testbench creation, but also with the verification planning and management component, which gives you guidance on how to plan and set up your verification environments and define your goals. This is why we say, "To be successful, begin with the end in mind."

cdnusers: How easy would it be for me to move from using the testbench piece of the methodology to using verification planning or the other components?
Mike: Good question. Truth is, no one goes out and adopts every aspect of the Plan-to-Closure Methodology right from the get go. So, to help designers understand the benefits of the entire solution and show them how it applies to their specific designs, we offer workshops about Plan-to-Closure and how to decide when you should adopt new pieces of the methodology.

Typically, our application engineers will sit down with your project team and look at how you are doing verification today: how the team is organized (designers vs. verification engineers), what kind of verification capability you have, and where you want to improve—is it in the area of quality, better predictability, higher productivity? Additionally, to give you a realistic example to follow, we have built the SoC Functional Verification Kit. It is a full-blown system-on-chip design to which we have applied all of the aspects of the Plan-to-Closure Methodology for learning and demonstration purposes. The chip is verified from the block level through the unit level to the system level with the different techniques we discussed.

Using the Kit, you will be able to see how to best apply assertion-based verification early at the block level or how to move from a test-driven simulation approach to a constrained random, coverage-driven testbench simulation approach. Or you can look at the system verification component of the Plan-to-Closure Methodology, which provides guidance on leveraging hardware acceleration and emulation for increasing run-time performance, doing automated, coverage-driven HW/SW verification, or leveraging SystemC TLM as a means for early architecture exploration and validation.

At the end of the workshop, you decide what will make the most sense for you to adopt—where will you get the biggest bang for your buck? You can then make an incremental step. Keeping in mind that, say next year, when you come up against a problematic hardware/software issue and need more capability, you know you have the knowledge to adopt more of the methodology. All of the components of the methodology can be applied incrementally, but they have also been designed to fit together as part of a complete flow, so the approaches you adopt today will fit into the next set of approaches you adopt in the future.

cdnusers: If you could leave cdnusers with a final takeaway point, what would it be?
Erik: The Plan-to-Closure Methodology can save you time and money. Each component is a well-tested, well thought out, thorough set of guidelines plus examples that exemplify those guidelines. These guidelines were conceived and made into products after application in many different types of verification applications.

With Plan-to-Closure Methodology, you have a way to approach verification that can span different application spaces. We have tried to make the methodology into a science where designers can track their plan via metrics regardless of what technique is being used. Whether it be formal, testbench, simulation, or emulation, there are various metrics designers and engineers can use

along the way to track their plan, from conception through to closure.

Finally, we have a unique way of delivering the methodology—the Knowledge System. The Plan-to-Closure Methodology is available on Cadence downloads. Documentation and more information is available in our Web-based portal— My Plan-to-Closure.—where users can customize the methodology and the VIP. All Cadence customers with an active Sourcelink® account can use My Plan-to-Closure. It is a great way to learn more.

About the author

Erik Panu Biography
Erik serves as Group Director for Cadence verification IP and methodology. He is responsible for the development and productization of Cadence Universal Verification Components (UVCs), which include simulation, formal, and acceleratable verification IP and the Incisive Plan-to-Closure Methodology. Erik has more than 15 years of professional experience in verification and holds a B.S. and M.S. in Computer Science from the University of Georgia. He can be reached at erik@cadence.com.

Mike Stellfox Biography
Mike joined Cadence in 2004 as the Group Director of the Verification Technical Field Organization. He came to Cadence with the Verisity acquisition. He was with Verisity since 1998 in a variety of technical management positions. Prior to joining Verisity, he worked at Viewlogic as an Applications Engineer. He began his career at IBM, where he was an ASIC Design Engineer responsible for the design and verification of 2D and 3D graphics chips. Mike holds a Bachelor of Science degree in Electrical Engineering from the University of North Carolina at Charlotte. Mike can be reached at stellfox@cadence.com.