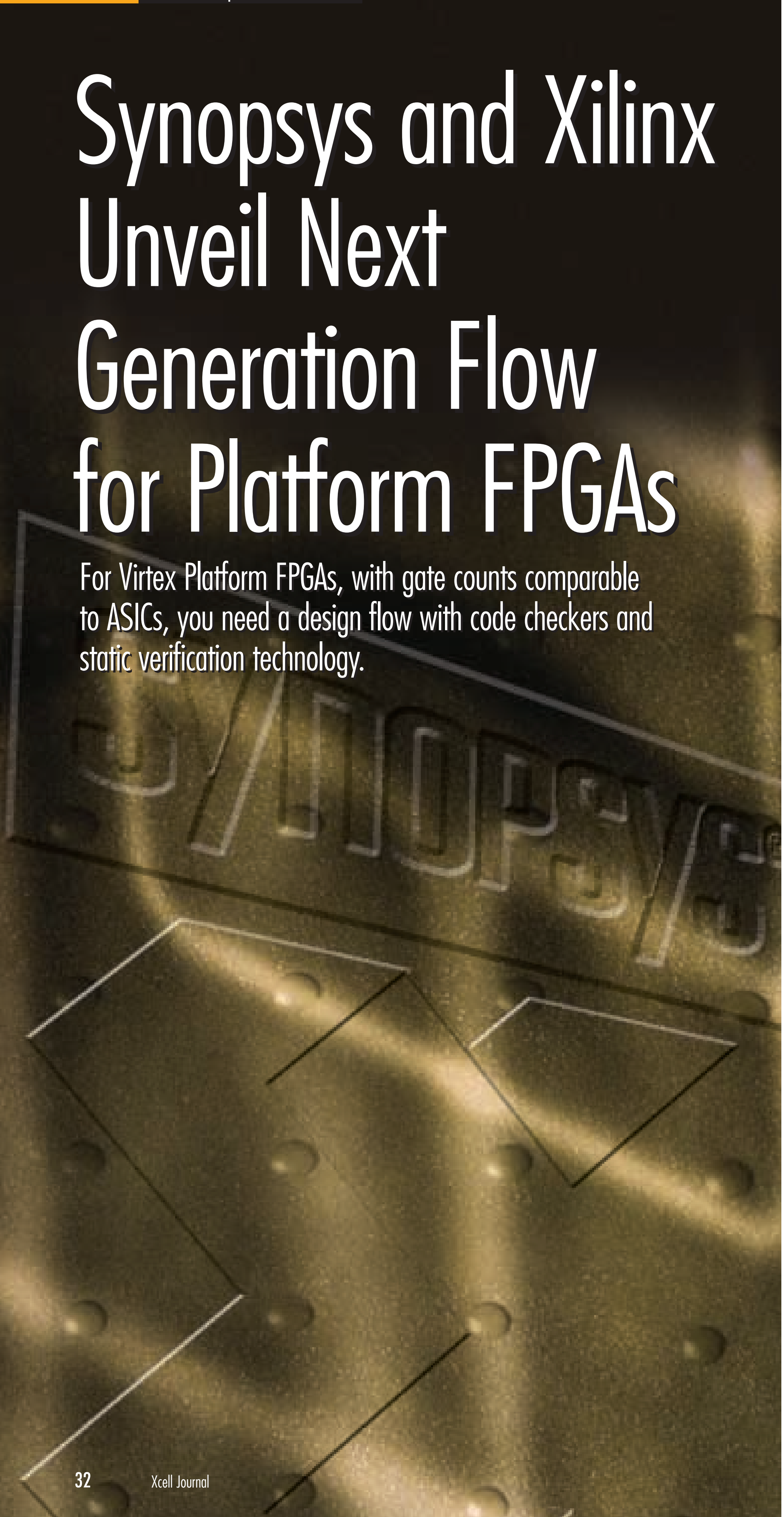
Synopsys and Xilinx Unveil Next Generation Flow for Platform FPGAs

For Virtex Platform FPGAs, with gate counts comparable to ASICs, you need a design flow with code checkers and static verification technology.



by Jackie Patterson
Director of Marketing Programs
Synopsys Inc.
jackiep@synopsys.com

Hamid R. Agah
Senior Technical Marketing Director
System Level and Alliance Marketing Group
hamid.agah@xilinx.com

As FPGAs grow in speed, size, and complexity, you need EDA tools and design flows that are similar to those used for ASICs. With 10-million system gates, 300+ MHz internal clock speeds, and gigabit serial I/O technology, the Virtex-II Platform FPGAs give you ASIC-like specifications and performance. And, because of far lower FPGA development costs, many designers are turning to complex FPGAs instead of ASICs for system-on-chip (SoC) designs. In fact, nearly one in two FPGA designs are SoCs.

While this FPGA complexity is good news for your SoC designs, you can face difficult design challenges unless you adopt some of the coding techniques and static verification methodologies used for developing ASICs. For high-density Xilinx® Virtex™-II Platform FPGAs, you can no longer rely on ad hoc methods and the simple design flows that are adequate for smaller FPGAs.

To meet this need, Synopsys and Xilinx recently announced an enhanced design flow that combines advanced Synopsys tools (LEDA, VCS, Scirocco, FPGA Compiler II, PrimeTime, and Formality) with the Xilinx ISE Alliance Series™ software, to reduce development time for Xilinx Virtex-II Platform FPGAs.

Enhanced Design Flow

The Synopsys tools give you an ASIC-like design flow that makes high density FPGA designs much easier and faster to develop. It begins with LEDA.

LEDA

The LEDA ProVHDL and ProVerilog programmable checkers enable you to create and automate the enforcement of design and coding-style guidelines, thereby improving performance and results. You can create a specification in VHDL or Verilog, and LEDA checks your HDL code for correctness and performance, applying a Virtex-II Platform FPGA rule set that Synopsys and Xilinx jointly developed. This gives you the opportunity to fix problems up front where they are least expensive. Synopsys will also prepackage LEDA checkers with policies to improve performance and results of tools like VCS, Design Compiler, Formality, Scirocco, and so on.

VCS

The register transfer level (RTL) design is then simulated using a high-speed simulator such as VCS or Scirocco. VCS, the Synopsys Verilog simulator, uses special algorithms that boost performance to provide simulation run times that often outperform today's commercial cycle simulators. The simulator accurately analyzes negative timing constraints, which are critical in verifying skew between high-speed signals in deep submicron designs.

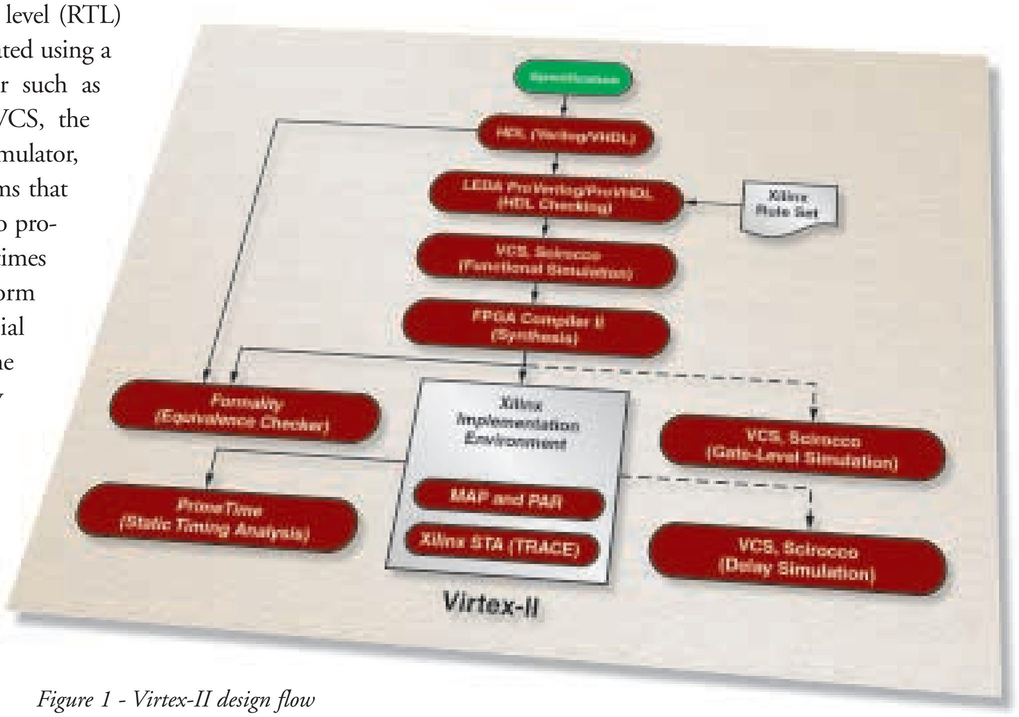

Specification
HDL (Verilog/VHDL)
LEDA ProVerilog/ProVHDL (HDL Checking)
Xilinx Rule Set
VCS, Scirocco (Functional Simulation)
FPGA Compiler II (Synthesis)
Formality (Equivalence Checker)
Xilinx Implementation Environment
MAP and PAR
Xilinx STA (TRACE)
VCS, Scirocco (Gate-Level Simulation)
PrimeTime (Static Timing Analysis)
VCS, Scirocco (Delay Simulation)
Virtex-II

Figure 1 - Virtex-II design flow

Scirocco

Use the Synopsys Scirocco simulator for VHDL designs; it unites the performance potential of cycle-based optimization techniques with the flexibility of event-driven simulation in a single simulator. Its optimized VHDL language compiler generates memory-efficient simulation executables, enabling acceleration of complete system verification, providing capacity in excess of 10 million gates on a single workstation. This level of performance and capacity is essential if you are using the high density Xilinx FPGAs.

FPGA Compiler-II

Next, your design goes to synthesis with the Synopsys FPGA Compiler II, which optimizes the high-level logic description into the unique Xilinx Virtex-II Platform FPGA architecture. FPGA Compiler II delivers these architecture-specific synthesis capabilities with automatic register retiming, pipelining, and Block-Level Incremental Synthesis (BLIS). The fast Xilinx ISE Alliance software completes the implementation. The Xilinx ISE tools deliver the industry's fastest runtimes and highest clock rates for the Virtex-II Platform FPGAs.

PrimeTime and Formality

Once your design is implemented, you can save valuable development time by verifying and debugging your design before trying it in the lab. Xilinx supplies scripts, libraries, and application notes to enable the Synopsys PrimeTime and Formality verification tools. PrimeTime performs a comprehensive post-layout static timing analysis of the Virtex-II device, which enables a fast, accurate, and exhaustive timing verification for the design. You can also use PrimeTime's rich set of analysis features to easily analyze and debug any critical timing issues.

The Formality equivalence checker allows you to quickly confirm that the implemented design matches the RTL specification. Formality uniquely addresses the need for functional verification of large, SoC designs. Many design engineers report that verification tasks consume 60% to 80% of design resources. Formality offers a much faster (10-100X) and more thorough alternative. With this tool, you compare the functional equivalency of the RTL source to the post-synthesis (or the final post-place-and-route) gate-level netlist. You can employ Formality after each design step to maintain complete functional equivalence at the gate level during every stage of the flow. With Formality, the need to perform time-consuming gate-level simulations at intervals in the flow, or one massive simulation run at the end, is virtually eliminated. Thus, you save time and also verify your Virtex- II Platform FPGAs much more comprehensively than ever before.

Conclusion

With the addition of LEDA, PrimeTime, and Formality, Synopsys and Xilinx have created design flows that were previously only available to ASIC designers. As a result of these improvements, ASIC designers who want to use the Xilinx Platform FPGAs will have the tools they need and currently use. This means shorter learning curves and faster time to market. These improvements also mean that ASIC and SoC designers have an alternative to expensive silicon fabrication technology.