High-Performance Platform FPGAs Now Need Formal Verification

Verplex teams with Xilinx to include ASIC-level equivalency checking in ISE 4.1i for formal verification of large Platform FPGA designs.

by Bassilios C. Petrakis
Business Development Manager
Verplex Systems Inc.
petrak@verplex.com

Historically, FPGAs were small devices enabling ASIC designers to prove designs while being able to experiment with various programming options until they got it right. Design closure was relatively simple.

Today's Platform FPGAs, however, have reached ASIC proportions with as many as 6 million system gates running up to 400 MHz. High performance synthesis engines, floorplanning, and physical synthesis tools are now necessary to achieve timing closure.

These synthesis and physical synthesis tools perform a high number of optimizations that drastically change design structures. Although these drastic changes are necessary to meet the design requirements, it is practically impossible to exhaustively prove with traditional simulation alone that a chip will work under all conditions. It would just take too much time to write and simulate the vast number of vectors required to verify all the possible combinations of states, events, and inputs for a typical design.

The speed, volume, complexity, and size of Xilinx® Platform FPGAs therefore demand an ASIC-level solution to FPGA design validation. That solution is an independent verification engine to audit or validate the changes. The independence of the verification engine is critical. To obtain the highest degree of confidence in the design, the algorithms and methods used in verification must be from a different company with a different tool.

Formal Verification

Formal verification tools address the design intent validation and design implementation functional closure. In support of the growing needs of large FPGA design verification, Xilinx has established a pivotal working relationship with Verplex Systems Inc. for a formal verification solution. Xilinx and Verplex have collaborated to establish a design methodology that guarantees interoperability between the Xilinx Integrated Software Environment (ISE) 4.1.i and the Verplex Conformal™ Logic Equivalence Checker (LEC).

Now in the FPGA flow, instead of completely relying on simulation by applying stimuli to a design and comparing its responses with expected results, formal verification examines the design mathematically, and proves its functional properties without test vectors. Unlike simulation, which will most likely overlook bugs for the lack of appropriate vector patterns or only report a minimal number of bugs based on limited input stimuli, formal verification will exhaustively prove the design while isolating bugs. When errors are found, formal verification can generate counter examples to demonstrate error conditions, allowing designers to complete verification tasks faster.

For design implementation verification, Conformal LEC offers superior verification with speed, capacity, and most important, 100% coverage. It also offers a more productive debugging environment over gate-level simulators. Conformal LEC compares designs in a matter of minutes or hours instead of the days or weeks that is required using traditional gate-level-simulation techniques. Conformal LEC, being an independent verification tool, boosts the designer's confidence that the design has been synthesized and optimized correctly.

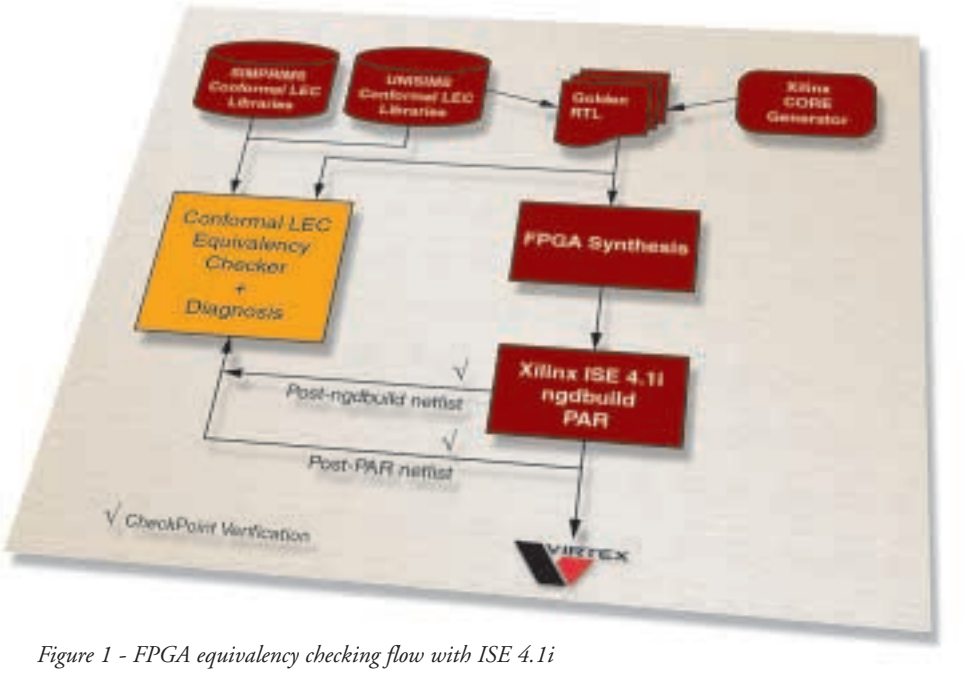

Figure 1 - FPGA equivalency checking flow with ISE 4.1i

Uses of Equivalence Checking in the FPGA Design Flow

Equivalency checking (EC) detects functional inconsistencies between two design representations. One of the two designs is considered to be the "Golden" design, typically described in register transfer-level (RTL) code. The other is the revised gate-level implementation. The goal of running equivalency checking on an FPGA design is to make sure the final design implementation does what the RTL code specifies it should do. EC can be used throughout the FPGA implementation process to:

- Compare a Golden RTL model against a partitioned design. This type of design partitioning occurs in ASIC prototyping using multiple FPGAs.
- Compare a Golden RTL model against a modified version of the model. It is often the case that an RTL model is altered to improve the performance of the design.
- Compare a Golden RTL model to the post-synthesis (post-ngdbuild) netlist. In the Xilinx flow, the netlist from the synthesis tool is first processed by ngdbuild in ISE 4.1i. The netlist generated, which is SIMPRIMS-based, is considered the post-synthesis netlist for the purpose of equivalence checking.
- Compare a Golden RTL model against the final post place-and-route (PAR) netlist (Figure 1). This is the final functional verification step before proceeding with programming the device.

Together with Xilinx ISE Foundation™ 4.1i, Conformal LEC can provide equivalence checking at the RTL and gate level of the design flow to functionally verify designs at every checkpoint.

Running Conformal LEC

It used to be that equivalency checkers were very difficult to set up and run. Typically, it would require a handful of expert verification engineers.

Verplex has set a precedent by creating a tool without the legacy limitations of earlier equivalence checkers. Conformal LEC was designed to tackle large designs (20M+ gates). It was created from the outset with ease of use and high performance in mind. The target user for Conformal LEC is the

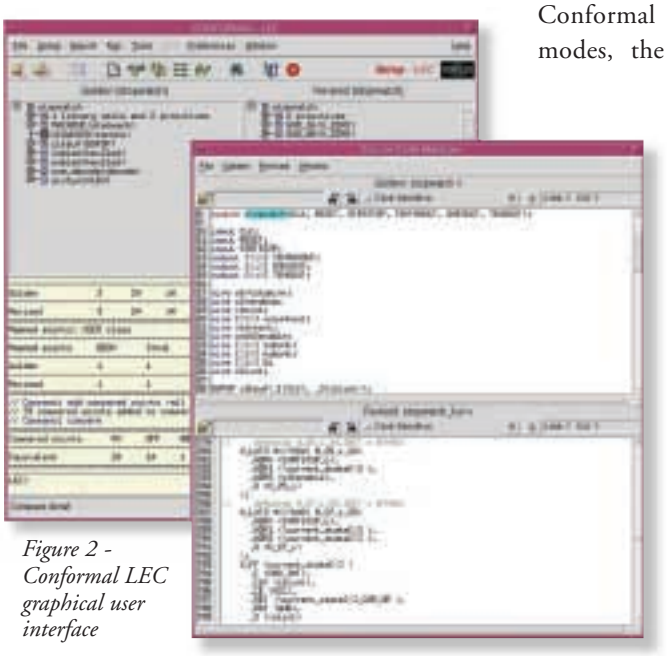

Figure 2 - Conformal LEC graphical user interface

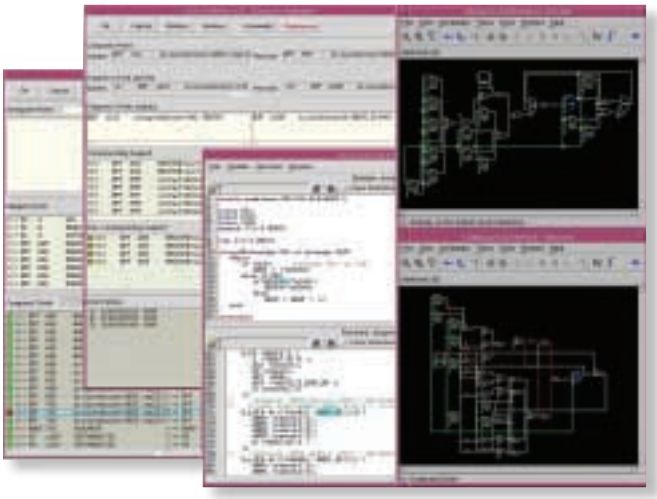

Figure 3 - Conformal LEC Diagnosis Manager

designer who is coding the entire design or a subset of a design. Conformal LEC requires minimal user setup. It runs in GUI interactive mode (Figure 2), command text mode, as well as in batch mode (using a command file).

Conformal LEC accepts Verilog, VHDL, and mixed-language RTL designs. Conformal LEC reads Verilog, VHDL, and EDIF gate netlists and supports Verilog simulation and Synopsys Liberty™ library models. Xilinx UNISIMS and SIMPRIMS libraries for Conformal LEC are now available in ISE 4.i software.

Conformal LEC operates in two modes, the "setup" mode and the "LEC" mode.

Setup Mode

In the setup mode, the Golden RTL and the revised (post-ngdbuild or post-PAR) designs are read into Conformal LEC along with the Xilinx UNISIMS and SIMPRIMS Conformal LEC libraries. The post-ngdbuild and post-PAR netlists can be generated by ndg2ver in ISE 4.1i

LEC Mode

In the LEC mode, the tool performs design flattening, remodeling, key-point mapping and compare.

Key-point mapping involves partitioning the two designs respectively by state element, input, output, tri-state, combinational feedback loop cuts, and black box instances, then automatically pairing up (mapping) the corresponding instances.

Once the design is fully mapped, Conformal LEC compares the corresponding logic cones of each map point. When all mapped points are found equivalent, then the whole design is equivalent. If one or more map or key-points is non-equivalent, the tool offers an intuitive debugging environment to help you understand the source of the mismatch (Figure 3).

The three RTL-to-Xilinx equivalent gate designs in Table 1 highlight the run time performance of Conformal LEC. The gates size for each design does not include memories.

Conclusion

RTL-to-gate equivalency checking has been used successfully in ASIC design verification for many years. It has been widely adopted as the mainstream verification step in the ASIC design flow. With the growth of FPGA devices for system applications, the need for verification for FPGA-to-ASIC and ASIC-to-FPGA conversions has become critical. The low cost of FPGA design will push the device capacities. "Reprogrammable" prototyping has reached it limits. Expanding into formal verification has now become an essential factor in the FPGA design flow.

Common uses for both ASIC and FPGA designs are growing. Commonality in both design flows is merging. One of the mutual meeting points in the design flows will be formal verification, and in particular, equivalence checking. Conformal LEC is available today to meet that challenge.

As Xilinx continues to meet the future demands in FPGA design, Verplex will provide independent verification tools to meet those same demands. For more information on formal verification and equivalency checking, go to *www.verplex.com.*

Test Case	Type	Design Size (gates)	CPU Time (sec.)	Memory (Mbytes)
1	RTL vs. Post-Route	50K	19.08	67.22
2	RTL vs. Post-Route (networking design)	720K	305.47	395.73
3	RTL vs. Post-Route	241K	297.89	180.16

Work Station: Ultra 80 Sun Solaris machine with 2G RAM

Table 1 - Test case examples and results