cadence®

Stratus High-Level Synthesis

Industry's first high-level synthesis platform for use across your entire SoC design

Cadence® Stratus™ High-Level Synthesis (HLS) automatically creates high-quality register-transfer level (RTL) design implementations for ASIC, system-on-chip (SoC), and FPGA targets from high-level C++/SystemC descriptions. The proven successes of Stratus HLS in production designs around the world are testament to its consistently high-quality results, mature feature set, and complete design coverage. Products built with Stratus HLS technology can be found in your home, automobile, and pockets.

Overview

Stratus HLS is the next generation of high-level synthesis technology, based on more than 13 years of production high-level synthesis deployment.

With Stratus HLS, engineering teams can quickly design and verify high-quality RTL implementations from abstract SystemC, C, or C++ models. The models can be easily created using the Stratus integrated design environment (IDE), retargeted to new technology platforms, and reused more easily than traditional hand-coded RTL. The Stratus IDE also allows designers to actively make tradeoffs between power, area, and performance from within the high-level synthesis environment.

Stratus HLS users report productivity as high as 2 million verified gates/designer/year, as compared to 200,000 in the traditional RTL flow. At the same time, Stratus HLS users consistently achieve silicon area and power consumption results equal to or better than those achieved with handwritten RTL.

Stratus HLS automates the design and verification flow of hundreds of blocks from transaction-level modeling (TLM) to gates. In addition, Stratus HLS helps with the real-world issues of engineering change orders (ECOs) and routability, both of which normally occur much later in the flow, through tight integration with the full Cadence tool flow.

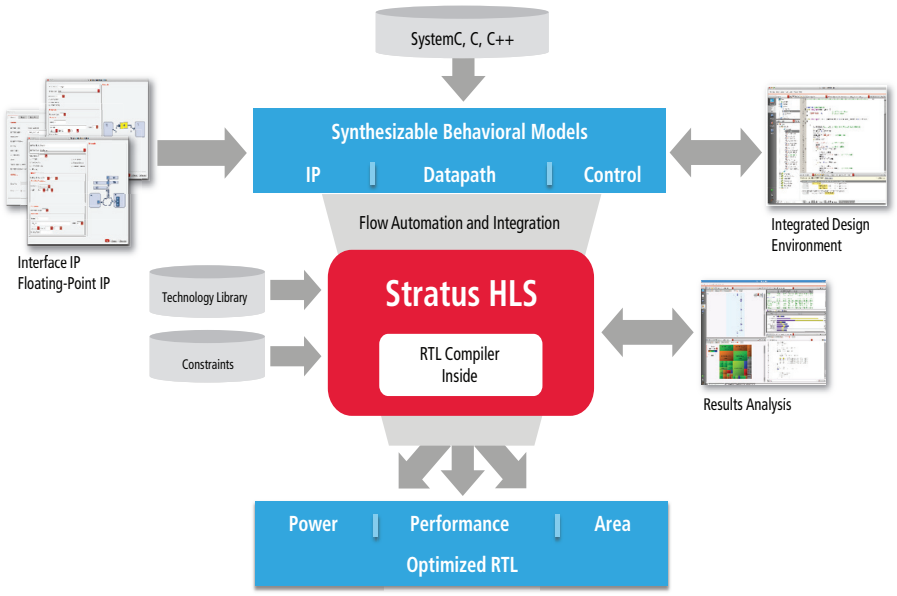

Figure 1: Stratus HLS Flow

Key Benefits

- Industry-leading quality of results (QoR), usability, and scalability via sixth-generation high-level synthesis core engine
- Wide applicability across the full design space, including both control-centric and datapath-centric designs
- Up to 50% power reduction through low-power optimizations
- Automated design and verification of hundreds of blocks with a consistent verification environment from early TLM models through gates
- Top-down and bottom-up support for ECOs
- Reduction of routing congestion via physically aware high-level synthesis
- Floating-point intellectual property (IP) to produce highly optimized signal processing designs.
- I/O interface IP to increase productivity by giving designers synthesizable SystemC building blocks
- Supports industry-standard IEEE 1666 SystemC, C, and C++
- Online knowledge base of over 1,000 articles with design examples, best practices, and application notes
- Full support for designs created with Cadence C-to-Silicon Compiler and Cadence Forte Cynthesizer™ high-level synthesis solution

Features

Behavioral IP Reuse

Stratus HLS enables the creation and adaptation of behavioral IP, delivering on the promise of true IP reuse.

Using Stratus HLS, the verified source code can be reused without modification for widely different process technologies and clock speeds. Modifications to the algorithm, architecture, or interfaces can be made incrementally at a high level, where previously they required a complete RTL rewrite.

Behavioral IP reuse with Stratus HLS significantly reduces overall design effort and maximizes return on investment (ROI).

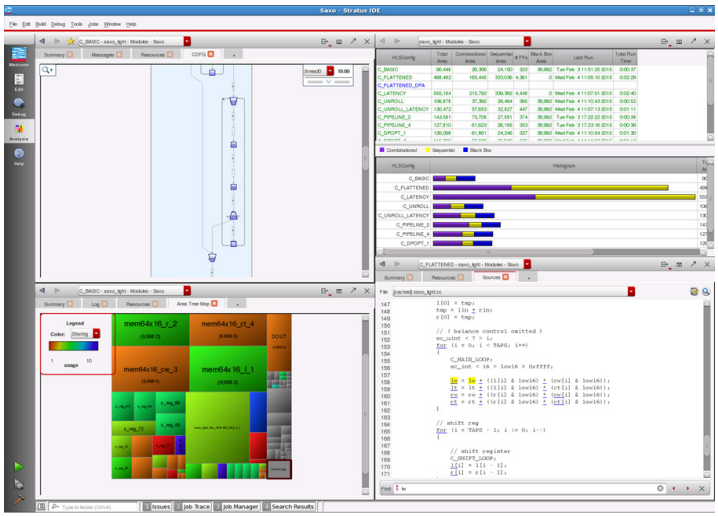

Figure 2: Complete Graphical Analysis with Links to Source Code

SystemC , C, C++ Synthesis

Stratus HLS supports industry-standard IEEE 1666 SystemC as well as C and C++ semantics.

SystemC allows designers to use familiar hierarchical decomposition techniques to manage design complexity. By supporting hierarchy and multiple threads, SystemC allows design and verification of multiple algorithms and their interfaces operating concurrently.

Stratus HLS also allows C or C++ algorithmic code to be automatically put into the context of a hardware module with hardware interfaces without any translation.

Integrated Design Environment

The Stratus IDE graphical user interface (GUI) incorporates a SystemC IDE, making SystemC development easy and intuitive for new users and advanced users alike. In addition to typical IDE features, the Stratus IDE makes it easy to quickly create new models using pre-defined design templates to reduce design and debugging time.

The Stratus IDE analysis environment includes SystemC and RTL source linking, control and dataflow graphs, schematic viewer, pipeline analysis, and QoR reporting and visualization to judge the impact of architectural optimizations.

IP Building Blocks

Stratus HLS includes an IP library and the Stratus interface generator to give designers synthesizable building blocks to jump-start their designs. Because these pre-designed optimized elements are implemented in high-level SystemC code, they can be used without performance or area penalties. The result is truly reusable design IP that accelerates the design and verification process.

The Stratus HLS IP library contains high-speed simulation models and bit-accurate synthesizable models for all IP:

- Floating-point datatypes available in IEEE 754 single and double precision as well as other combinations of exponent and mantissa width defined by the user
- User-configurable connectivity interfaces such as line buffer, circular buffer, trigger-done, streaming data, FIFO-based, and memory interfaces
- Clock domain crossing (CDC) circuitry for multi-clock designs
- Specialized communication interfaces, including frame buffers and bus interfaces for connections using the ARM® AMBA® AXI and AHB interconnects.

Design and Verification Flow Automation

Stratus HLS comes with a fully integrated automation system. One short Tcl file can configure and automate all of the following:

- C++ compilation and linking
- Synthesis directives
- SystemC and RTL verification
- Logic synthesis
- Formal equivalence checking between RTL and gates
- Power analysis
- RTL analysis and debugging
- Design exploration for multiple configurations

This automation allows designers to use Stratus HLS to explore multiple architectures and implementations, while improving verification by providing a consistent environment from early TLM models through gates.

ECO Support

Changes to the design can become necessary at any point during the design flow. When those changes come late, the ECO mode of Stratus HLS can enable you to still meet your deadlines.

Rather than creating new RTL and completely redoing the entire RTL-GDSII implementation for the design, Stratus HLS creates an ECO patch to minimize the impact of the change on the design and thus the schedule.

Working in conjunction with Cadence Encounter® Conformal® ECO Designer, the patch can be applied downstream on the netlist, the placed design, the routed design, or even the post-mask netlist if a metal-only ECO is possible. Encounter Conformal ECO Designer combines Cadence Encounter Conformal Equivalence Checker with Cadence Encounter RTL Compiler synthesis optimization to ensure the patch is correct and that the design still meets its quality of results goals.

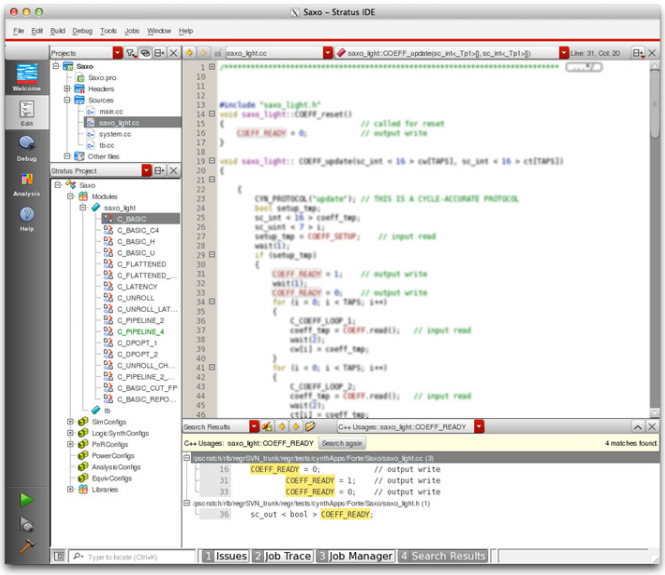

Figure 3: Stratus Integrated Design Environment (IDE)

Low-Power Optimization

With Stratus HLS, design teams can automate complex low-power optimizations that are often difficult or impossible to realize with hand-coded RTL, such as fine-grained and block-level clock gating, finite state machine (FSM) optimizations, and power-aware scheduling (re-ordering) of operations.

These optimizations can yield power reductions of more than 50% depending on the design. Designers can use the exploration capabilities to quickly trade off design area, performance, and power.

Timing Closure

Stratus HLS ensures easy timing closure for the generated RTL by exhaustively analyzing each path and scheduling operations so they easily fit in the given clock period.

Stratus HLS uses patented datapath optimization technology and the embedded Cadence Encounter RTL Compiler to build all operations, multiplexers, and registers in the specified technology library to get accurate timing and area models.

The user can control how aggressively Stratus HLS packs these operations into each clock period. Creating designs with Stratus HLS can save months of backend effort by preventing timing closure problems.

Cadence Services and Support

- Cadence application engineers can answer your technical questions by telephone, email, or Internet—they can also provide technical assistance and custom training
- Cadence certified instructors teach more than 70 courses and bring their real-world experience into the classroom
- More than 25 Internet Learning Series (iLS) online courses allow you the flexibility of training at your own computer via the Internet
- Cadence Online Support gives you 24x7 online access to a knowledgebase of the latest solutions, technical documentation, software downloads, and more

cadence®

Cadence Design Systems enables global electronic design innovation and plays an essential role in the creation of today's electronics. Customers use Cadence software, hardware, IP, and expertise to design and verify today's mobile, cloud, and connectivity applications. **www.cadence.com**