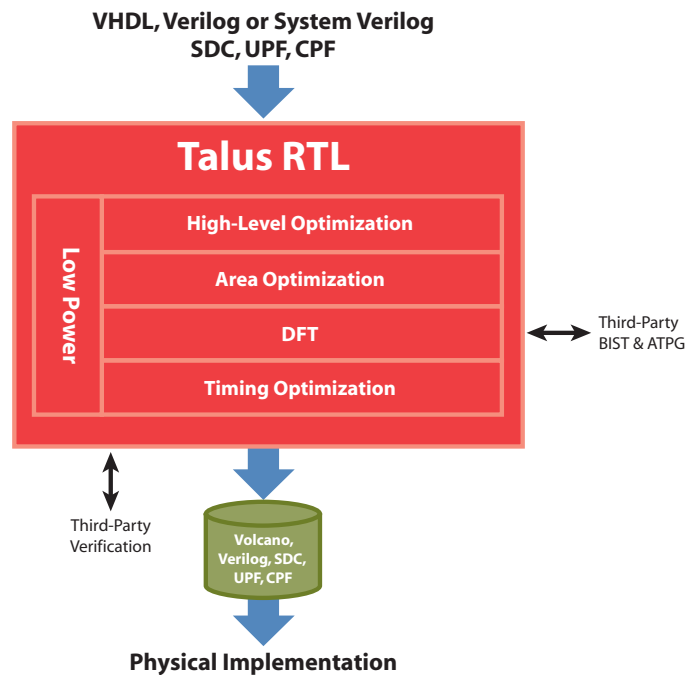


Talus[®] RTL

- Supports VHDL, Verilog and System Verilog with a standard license.
- Full-featured RTL and data-path synthesis minimizes area and maximizes performance.
- Enables a high-performance, high-capacity RTL-to-GDSII flow when integrated with Talus Vortex.
- Supports a range of design methodologies including top-down and bottom-up approaches.
- Comprehensive DFT support including scan insertion and optimization with seamless interfaces to industry leading third-party BIST and ATPG products.
- Includes automatic arithmetic operator inference and extensive building block IP support through LavaWare™ library.
- Features data-path extraction with a unique ability to perform architecture switching in the back-end flow.
- Early RTL, DFT and netlist checks accelerate problem detection and correction, rapidly improving the quality of input.
- High capacity and fast runtimes allow complexity scaling without compromising productivity.
- Volcano™ database support provides a simple, efficient, productive environment to exchange design information with physical implementation teams.
- Timing- and congestion-driven macro legalization removes need for block-level floorplanning.
- Performs comprehensive dynamic and static power optimizations, supports both CPF and UPF power intent formats.

Talus RTL is a comprehensive RTL synthesis solution that can be used standalone or integrated within Magma's Talus IC implementation system. It is a full-chip synthesis environment that enables rapid development of RTL and chip-level constraints throughout the design process without sacrificing design quality or the delivery schedule. When used in conjunction with the Talus Vortex physical implementation tool it dramatically improves the productivity of chip architects and logic designers while shortening turnaround time.



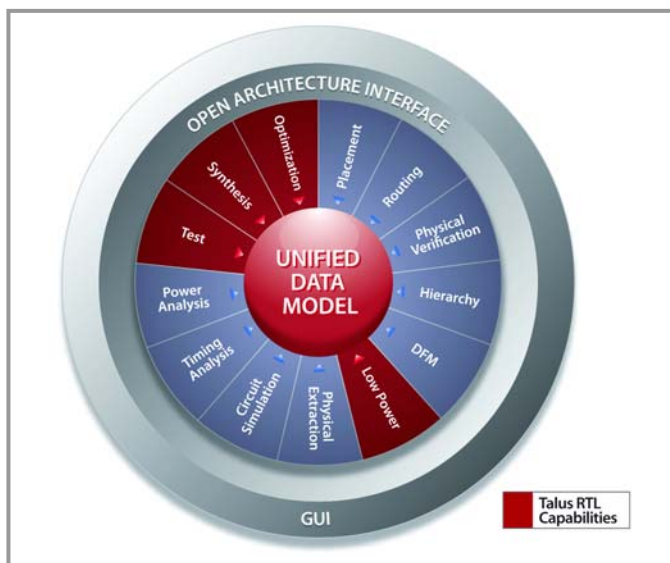
Talus RTL reads in RTL code, cell library, technology information and design constraints, then generates a placed and globally routed netlist consisting of cells from the library.

Talus RTL includes production-proven RTL synthesis and a timing analyzer. Talus RTL synthesizes full-chip RTL for given area, timing and power constraints. Built on Magma's unified data model architecture Talus RTL shares a number of engines with Talus Vortex including a timing analyzer and uses a common script language and constraint formats. With these commonalities, Talus RTL provides logic designers with an integrated, highly productive, predictable flow that reduces the number of iterations, minimizing turnaround time.

Talus[®] RTL

Fast, High-Capacity RTL, DFT and Data-Path Synthesis

Leveraging the high capacity of Magma's exclusive unified data model, Talus RTL synthesizes several-million-gate RTL designs without hierarchical partitioning or guard-banding-related timing constraints. Synthesizing the entire chip or very large logic partitions, instead of numerous smaller sub-modules, produces superior optimization results because the flow is not limited by arbitrary boundaries. An incremental elaboration capability allows small changes to the design's RTL to be propagated quickly into the implementation flow without the time-consuming and error-prone manual process of recompiling the entire design.



Magma's unified data model provides unmatched integration that delivers superior optimization.

Talus RTL includes a built-in dynamic arithmetic module generator that infers data-path elements within the RTL source and considers timing constraints to automatically implement the best micro-architecture that achieves required performance while minimizing area and power. Unlike traditional synthesis, Magma's module generator globally synthesizes entire arithmetic expressions using operator merging and by sharing common operations across hierarchical boundaries to generate the best architecture for the entire data-path module. Taking advantage of its fast synthesis capability, Talus RTL implements data-path modules dynamically, without creating large caches of alternative architectures. Talus RTL also has the unique ability to swap architectures during physical synthesis to achieve the desired performance. If, for example, a particular architecture of an arithmetic function is selected during synthesis to meet a timing or area goal but later, during physical synthesis or placement, prevents design closure, the Talus RTL-to-GDSII system will automatically replace it with an alternate faster or smaller architecture, if available.

To ease adoption of this powerful, automated design environment, Talus RTL synthesizes and optimizes RTL input in all of the industry-standard HDL formats, including System Verilog (IEEE 1800), Verilog (IEEE 1364-1995 and 1364-2001) and VHDL (IEEE 1076-1987, 1076-1993). Talus RTL is compatible with widely-used coding styles for synthesis and reuse. Talus RTL also supports synthesis pragmas and instantiated arithmetic data-path components used in legacy commercial synthesis tools.

Strength-Based Delay Model for Optimal Timing

With Talus RTL, Magma introduces optimization capabilities based on strength-based delay models. This delay model accurately considers the effects of buffering and sizing at an early stage of optimization. Each cell is assigned a continuous drive strength that abstracts actual sizes available in the technology library. Rather than using fixed cells from a library, Talus RTL replaces each logic function with automatically abstracted HyperCell™ models. These are functional placeholder cells with variable drive strengths.

Initial placement and routing is done with the HyperCell model to determine the final optimal timing for all paths in the design. Layout optimization is performed by continuously adjusting the strength of each HyperCell as load and timing change throughout the optimization process, allowing optimal delay to be achieved at any stage of optimization. Finally, the HyperCells are mapped to actual library cells with a discrete size. By employing optimal continuous sizing combined with adaptive buffering, Talus RTL is able to quickly deliver designs that consume less power and area while meeting timing requirements.

Built-in Low-Power Design and Optimization

Talus RTL provides an integrated power optimization flow, providing better power reduction than conventional standalone synthesis tools while also delivering high performance. Power optimization capabilities deliver lower dynamic power consumption than conventional synthesis. Only optimal cell sizes are used to drive known loads, avoiding unnecessary power dissipation by cells. Balancing input slews to cells through optimal sizing is used to reduce total switching power. Additional power optimization capabilities such as multi-Vt library-based optimization, DFT-aware automatic clock gating, use of integrated clock-gating cells in the standard-cell library, detection of synchronously enabled registers and hierarchical insertion of clock-gating logic minimize power and improve testability.

Advanced, low-power design capabilities are available using the Talus Power Pro option. These include voltage island support, automatic MTCMOS switch insertion and support for the Common Power Format (CPF), Unified Power Format (UPF) or native commands to define power intent.

Comprehensive and Configurable DFT Rule Check Engine

To improve testability of the design, Talus RTL supports top-down and bottom-up hierarchical scan insertion throughout the synthesis and physical design flow. This capability is included with a standard license. Talus RTL has a comprehensive and configurable DFT flow check engine that allows the user to analyze and debug testability issues. A repair mechanism automatically adds test logic and optionally adds test points to resolve problems and help improve test coverage. During scan insertion, the scan chains are properly and safely balanced to help reduce test time. The chains can be verified with post-scan flow checks, and the tool can quickly produce a fault coverage estimate within 1 percent of the final results from commercial ATPG tools. Support files for leading ATPG tools are automatically generated for a seamless handoff.

Additional DFT strategies such as on-chip test vector compression, logic BIST, memory BIST, and boundary scan insertion are supported in the form of RTL insertion as well as through interfaces to tools from leading DFT vendors. These interfaces are developed in collaboration with the MagmaTies DFT partners to help ensure a smooth integration of Talus RTL and third-party DFT tools.

Single Timing Analyzer and Constraints

Talus RTL eliminates timing mismatches between synthesis and physical design using a single static timing analyzer throughout the RTL-to-GDSII flow. Design optimization constraints are set once at the RT level and are used throughout synthesis and physical design. The optimization flow performs push-down operations on timing constraints to support top-down design flows, and pull-up operations to support hierarchical, IP- or black-box-based bottom-up design flows.

The constraints are set in widely adopted SDC format and support simultaneous multi-mode case analysis. Customized timing reports are available throughout the chip implementation flow. The built-in timer enables fast, incremental timing analysis following design changes during prototyping. Early insights into block- and chip-level timing help identify and fix potential timing problems from sub-optimal RTL code or incorrect design constraints in the early design stages.

The optional advanced timing and variability engines allow for concurrent, correct-by-construction timing results and optimizations across all modes and process, voltage and temperature (PVT) corners, while also considering both on-chip variation (OCV) and crosstalk effects. This implementation-level capability removes the need to iterate through a single-mode sign-off timer, greatly reducing turnaround time.

Predictability Improves Productivity

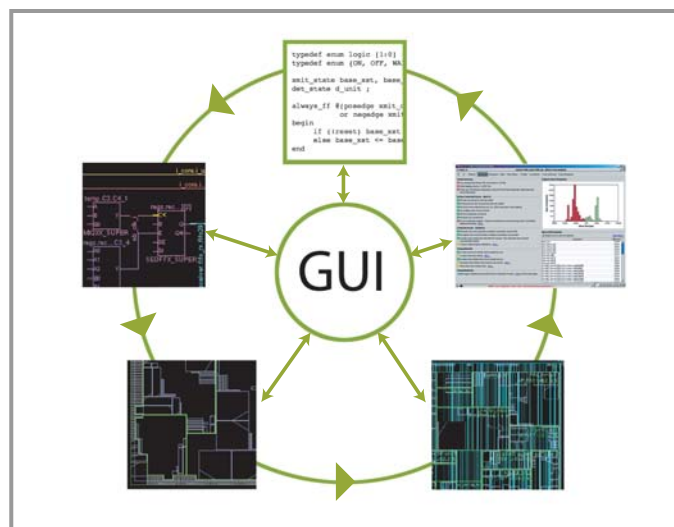
With Talus RTL and Talus Vortex sharing common design and analysis engines, implementation is no longer the bottleneck in the design process. Electronic System Level (ESL) designers can rapidly assess the impact of alternate system architectures on physical metrics such as area, performance, power, routability, testability, manufacturability and yield. Any late-arriving specification, RTL or constraint changes can be easily accommodated without affecting the schedule or the productivity of the engineering team.

Talus RTL enables designers to easily check the correctness of synthesis and optimizations by providing seamless interfaces to industry leading third-party formal verification tools. Verifying advanced optimizations can be time consuming. To speed this process, Talus RTL provides verification hints to the formal verification tool such as identification of swept or cloned flip-flops, scan enables and integrated clock gates. In addition, for advanced optimizations that are difficult to verify or provide hints for, these optimizations can be disabled, allowing designers to trade off small degradations in design results to ease verification.

Talus also supports handoffs in industry-standard formats such as Verilog netlist, SDC constraints, CPF and UPF power intent, and DEF placement for use with third-party physical design tools.

Powerful GUI Speeds Design Debug and Exploration

Using the Talus RTL schematic- or layout-based visualizations, logic designers can examine the functionality of RTL code and view logic levels, physical partitions, pin placement, clock distributions and timing paths throughout the flow. Cross-probing throughout the flow from the schematic, floorplan or layout into the RTL code allows logic designers to quickly identify, locate and fix problems due to timing constraints or



Cross-probing accelerates quality improvement.

Talus[®] RTL

improper RTL code structures. With this unique debug capability, designers can browse the logical hierarchy and guide partitioning decisions for floorplanning. Connectivity-driven visualizations such as fly-lines and clock-domain distribution provide valuable architecture and constraint improvement information. Slack-based timing histograms of critical paths in the built-in timing visualizer allow designers to quickly locate

timing problems through direct cross-probing of the RTL, schematic, floorplan or layout. Such analysis readily leads to identification of missing constraints or exceptions such as false paths or multi-cycle paths. Detailed power reports and maps provide power consumption and distribution information early in the design flow, saving back-end packaging and design re-spin costs.

TECHNOLOGY FEATURES:

Full-Featured HDL Synthesis

- IEEE LRM-compliant Verilog, System Verilog and VHDL
- Top-down and bottom-up flow support
- Dynamic data-path module generation
- High-performance, high-capacity RTL-to-GDSII flow when integrated with Talus Vortex
- DFT scan checks, repair, insertion and optimization
- Distributed synthesis
- High-capacity support for large, complex, multimillion-gate designs
- Interfaces to third-party formal verification tools
- Interfaces to third-party ATPG and BIST tools

Low-Power Optimization

- Hierarchical clock gating
- Integrated clock-gate cells
- Vector or vectorless driven advanced clock gating
- Support for Multi-Vt, Multi-Vdd, MTCMOS libraries
- UPF and CPF support

Fast, Accurate Static Timing Analysis

- Hierarchical timing constraints
- Timing reports and graphical timing analyzer
- Multi-mode analysis

Inputs

- Verilog, System Verilog, VHDL, SDC, .lib, UPF, CPF

Outputs

- Verilog, SDC, SDF, SPEF, DEF, UPF, CPF, Volcano

Platforms

- Linux, Sun Solaris

