Part of the Cadence® Incisive® functional verification platform, Incisive Enterprise Specman products blend leading-edge process automation technology with the comprehensive Plan-to-Closure Methodology to simplify and speed verification. Specman® products automate your entire verification process, from the individual blocks to the full chip and all the way to the project level. With Specman technology, you benefit from increased productivity and a predictable path to high-quality silicon.
INCISIVE ENTERPRISE SPECMAN ELITE TESTBENCH

Successful verification of today’s multimillion-gate designs requires optimal speed and efficiency. But verification teams often struggle to squeeze in enough cycles to ensure that functional bugs won’t surface in silicon. Incisive Enterprise Specman Elite Testbench (Specman Elite® technology) is a comprehensive environment that accelerates and simplifies all aspects of verification (automatic generation of functional tests, data and assertion checking, and functional coverage analysis) in addition to supporting the time-tested Plan-to-Closure Methodology.

You can extend the functionality of Specman technology with Incisive Enterprise Specman ESL Testbench (Specman ESL), which provides a high-throughput channel between the testbench and the DUT, and enables Plan-to-Closure verification automation of embedded software exactly as if it were another part of your DUT. With other elements from the Incisive platform, including verification IP, hardware acceleration and emulation, analog/mixed-signal/RF verification, and formal assertion verification, Specman products support any testbench, HDL, software, or assertion IP.

You may be thinking, “But we already do testbenches in C, VHDL, and Verilog®,” and you may have invested time and effort in internal solutions. But, realistically, these tools are rewritten project to project without allowing for significant reuse. Nor do they contain the engines already built into Specman products, such as the patented constraint solver that generates stimulus automatically.

With Specman products, you can use the powerful e verification language to capture rules from executable specifications and use the information to automate verification. The Specman methodology finds even the most subtle, corner-case bugs because it eliminates misrepresentations of specifications. And since IP reuse support is built-in, Specman products not only leverage your existing investments, but also produce higher quality products in less time.

**BENEFITS**

- Captures executable specifications and eliminates misrepresentations
- Uses constraints to automate test generation
- Speeds debugging with data and assertion checking
- Increases predictability with functional coverage analysis
- Supports all IEEE-standard languages including e and SystemC®, plus C, C++, VHDL, Verilog, and SystemVerilog
- Enables hardware/software coverification
- Works with all major simulators

**FEATURES**

CONSTRAINT-DRIVEN STIMULUS GENERATION

Specman products provide constraint-driven test generation that automates the process of generating functional verification tests. By specifying constraints, you can target the generator quickly and easily to create any test in your functional test plan. You can even generate tests on-the-fly based on the current design state, making it possible to detect even hard-to-reach corner cases.

**DATA AND ASSERTION CHECKING**

Powerful temporal constructs allow you to capture complex protocols for assertion checking. On-the-fly data checking and generation provides context-specific expected values. With Specman products, you can use any combination of gray-, black-, or white-box checking to speed debugging.

**FUNCTIONAL COVERAGE ANALYSIS**

An executable functional test plan measures the progress of verification, and functional analysis automatically identifies holes in the test coverage. Since functional coverage is a meaningful and direct measure of the completeness of your verification, functional coverage analysis increases predictability in your verification schedules.

HD simulator interfaces

Specman products integrate with all leading HDL simulators and support a high-performance, direct kernel interface to all Incisive simulators. You can sample and drive internal signals of the device under test. With 100% controllability and observability of otherwise inaccessible internal signals, all Specman engines have full access to signal values during simulation.
TRANSACTION-LEVEL MODELING AND SYSTEMC SUPPORT

Specman products provide SystemC interface mechanisms to drive and monitor transaction-level models (TLMs) as well as signal-level models. You can apply Specman verification methodologies to the verification of SystemC architectural models using TLMs and reference models including mixed SystemC/RTL environments, and co-verify SystemC models used for software development.

In addition to supporting Incisive simulators, Specman products provide interface adaptors for SystemC simulators including OSCI® and CoWare ConvergenSC. With Specman technology, you can create a single verification environment to verify your SystemC model and then reuse it throughout the entire downstream flow, from RTL simulation to acceleration and emulation.

HW/SW CO-VERIFICATION

Specman products support all leading hardware/software co-verification tools. They also integrate seamlessly with the Incisive Software Extensions (ISX) in the Specman ESL co-verification environment to enable functional testing of both hardware and software. Early integration and debugging of hardware/software systems eliminates errors and shortens time to market for the combined system.

INCISIVE ENTERPRISE SPECMAN ESL TESTBENCH

Traditionally, the “ESL” acronym has been used to describe the need for scalability and increased abstraction to accommodate growing HDL designs. Incisive Enterprise Specman ESL Testbench (Specman ESL) builds on this traditional flow to enable systems engineers, logic designers, software engineers, verification engineers, and system validation teams to do “in-system” verification at the block, chip, and full system levels.

This is all made possible by using the components in Specman ESL to leverage the metric-driven Plan-to-Closure Methodology to span design and verification—from an initial system specification and system verification plan to full system-level integration and closure. The result: rapid discovery of deeply buried, cross-domain bugs impossible to find otherwise.

INCISIVE SOFTWARE EXTENSIONS (ISX)

ISX gives your testbench access to your embedded software exactly as if it were another part of your HDL DUT. Using extensions to the familiar Incisive SimVision debug tool, you can simultaneously control and verify software methods, procedures, variables, registers, and other elements with your traditional hardware-centric DUT using the same time-tested coverage-driven verification (CDV) process described in the Plan-to-Closure Methodology. Furthermore, ISX rises above past hardware/software co-verification limitations by supporting processor models in any form: workstation-based host-code execution, ISS, full RTL CPU models, hardware acceleration and emulation—even prototype silicon.

TRANSACTION-BASED ACCELERATED (TBA) SPECMAN TECHNOLOGY

By adding accelerators and emulators equipped with assertions and transaction-level interfaces, the automation environment can achieve the performance needed to drive full system-level validation that includes embedded and application layer software. TBA Specman technology provides the required high-throughput connection between an e language testbench and the acceleration and emulation hardware.

When TBA Specman technology and ISX are combined with vPlan executable specification and Incisive Enterprise Manager as described in the Plan-to-Closure Methodology, the result is a predictable, metric-driven process for both hardware and software development based on an executable plan that enforces adherence to the system specification. In short, this real-time, automatic annotation of metrics against your specification gives you the information you need to confidently argue for additional time to explore more high-risk corner cases, or to ask for a new pen to autograph the signoff form.

SPECIFICATIONS

LANGUAGE SUPPORT

• Testbench
  – e (IEEE 1647)
  – Interface to SystemVerilog (IEEE 1800) testbenches (this is a high-performance, direct kernel interface when using Incisive platform simulators)

• Device under test
  – SystemVerilog (IEEE 1800)

• SystemC (OSCI SystemC v2.01, IEEE 1666)
  – PSL (IEEE 1850)
  – SVA (IEEE 1800)
  – C and C++ models
  – Matlab models
  – Analog models in Verilog-A, VHDL-A, or SPICE formats

• Post-silicon hardware
  – Specman ESL supports embedded software and high-throughput connections to accelerated and emulated DUTs

CADENCE BRAND IP SUPPORT

• Design IP
  – Functional Verification Kit for ARM® processor-based designs

• Verification IP
  – Supports all simulation-based universal verification components (UVCs), transaction-based VIP,
assertion-based VIP, and SpeedBridge® rate adapters used in emulation

– Supports the full portfolio of Cadence UVCs: PCI Express, AHB, AXI, AMBA™, USB, and Ethernet, PCI, SATA, and OCP

– UVCs are designed to use all elements of the Incisive platform and feature a comprehensive “compliance management system” that leverages vPlan executable specifications to exhaustively verify protocol compliance

THIRD-PARTY SUPPORT

• Models
  – Third-party model support through the Cadence Verification IP Partner program

• Software
  – Third-party software support through the Connections® program with more than 30 verification company partners

INTERFACES

• Direct kernel interface to all Incisive platform simulators

• Direct C language interface

• Socket interface

• PLI (IEEE 1364)

• DPI (IEEE 1800)

• VPI (PLI 2.0, IEEE 1364)

• VHPI

PLATFORMS

• Sun Solaris

• HP-UX

• Linux

CADENCE SERVICES AND SUPPORT

• Customer-focused solutions that increase ROI, reduce risk, and achieve your design goals faster

  – Collaborative approach and design infrastructure—virtual teaming

  – Proven methodology and flow tuned to your design environment

  – Design and EDA implementation expertise

• Product and flow training to fit your needs and preferred learning style

  – More than 80 instructor-led courses—certified instructors, real-world experience

  – More than 25 Internet Learning Series (iLS) online courses

• Cadence customer support that keeps your design team productive

  – Cadence applications engineers provide technical assistance

  – SourceLink® online support gives you access to software updates, technical documentation, and more—24 hours a day, 7 days a week