# Verdi

## Debug and verification management platform

## Overview

The Synopsys Verdi® debug and verification management platform is an all-encompassing solution designed to streamline and enhance your design entry, debug and verification management. With its robust capabilities and connection into the most popular signal database (FSDB), Verdi empowers you to plan, execute and determine coverage of your simulation regressions. Furthermore, Verdi offers world-class debug capabilities to provide you insight into all design and verification flows. Verdi includes powerful AI technology to automate difficult and tedious debug steps and easily navigate diverse and complicated design environments.

## Powerful, Leading Technologies for High Productivity

The Verdi platform is built on proven technologies and enhanced with AI to enable users to meet verification and tape-out schedules. Productivity is boosted by unified technology that checks design entry in real time, manages regressions, collects data from Synopsys and third-party verification solutions, analyzes that data with AI techniques, and extends analytics with an API.

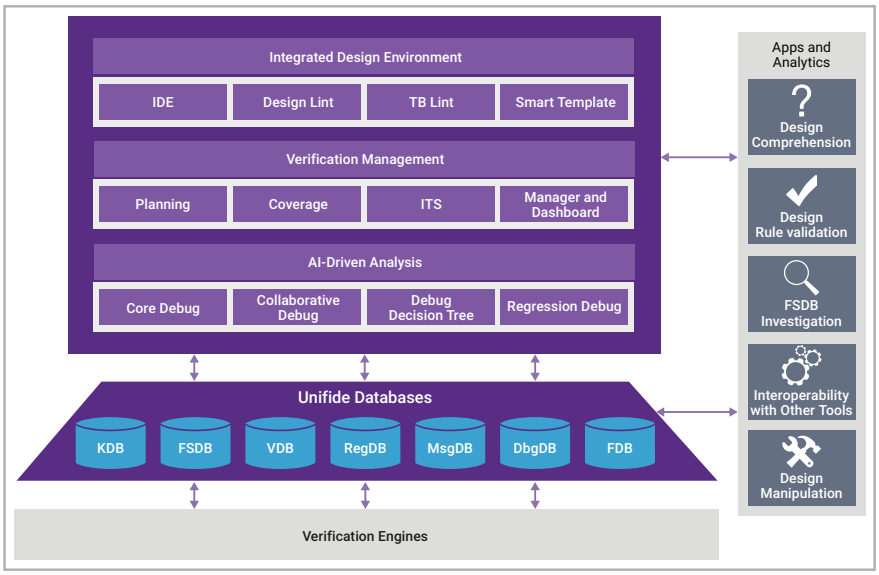


Figure 1: The Synopsys Verdi platform provides AI-driven debug, verification management, and an integrated design environment

## Platform for Debug and Verification Management

The Verdi platform consists of three major components: 1) debug technology, 2) an integrated design environment (IDE), and 3) a verification management system (VMS). The platform reads from, and in some cases writes to, various databases that are unified. Most of the database are managed by servers, some of which operate 24 hours a day,  seven days a week.

The goal of these components are:

- Prevent entry-level bugs
  - By incorporating an IDE, debug is expanded to the design under test (DUT) and testbench created. This is achieved by providing on-the-fly linting and synchronization with the databases.
- Efficiently run regressions
  - Results from regressions drive debug, and often additional tests are need to enhance debug.
  - Regressions are managed by a verification management system which incorporates planning, running or executing the regression tests, aggregating resulting coverage, and making results available for debug.
- Triage bugs using AI
  - The results of regressions, usually in the form of log files, must be categorized (such as origin in the DUT or testbench), then triaged into bugs that should be examined first due to likely hood on being closer to the root cause. AI-based technology is utilized for this purpose.
- Execute root cause analysis (RCA) using AI
  - The step after triaging bugs is to then determine the root cause of the bugs. Several RCA engines, leveraging AI, are run based on the triage results.
- Share with team members
  - After several automated steps, interactive debug can be performed. In addition, results and debug decisions can be shared world-wide via servers.

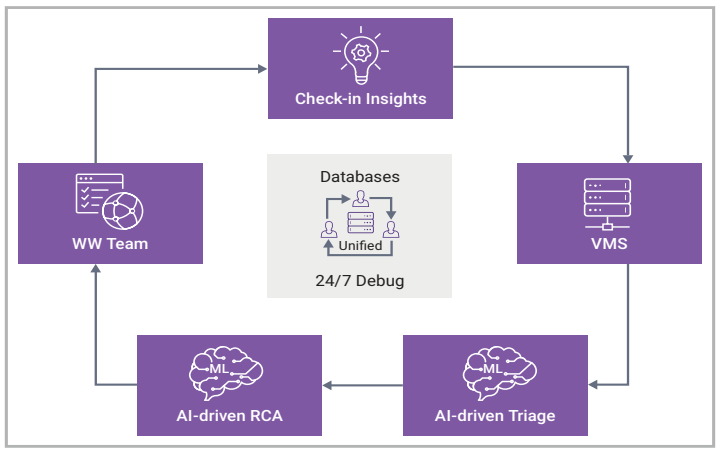


Figure 2: Synopsys Verdi platform technology and goals

## Core Debug

The Verdi system incorporates all technology and capabilities to exceed expectations for debug. It includes a full-featured waveform viewer, powerful waveform comparison engine, source code browser, state machine diagram viewer, simulator-independent protocol analysis, low-power analysis, and assertion analysis.

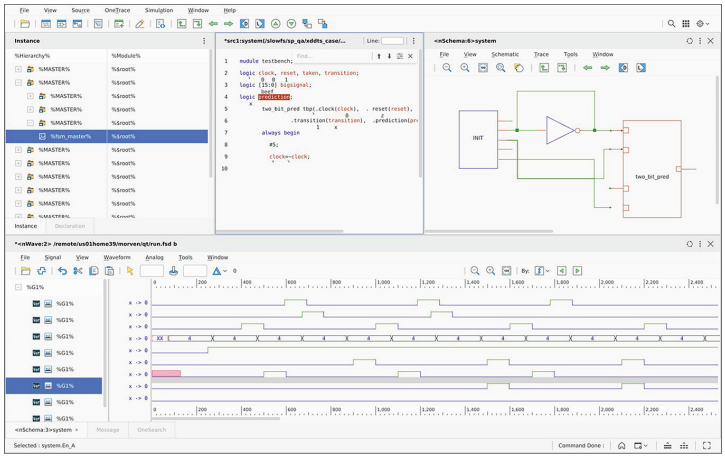

Figure 3: Powerful debug capabilities are accessed via the Synopsys Verdi platform graphical user interface

## Powerful Visualization and User Experience

The Verdi platform provides the following fundamental viewers:

- Full-featured waveform viewer to display and analyze activity over time
- Flexible schematics and block diagrams give the ability to display logic and connectivity using familiar symbols
- Intuitive bubble diagrams help to reveal the operation of finite state machines

The viewers can be personalized to include the following attributes:

- Welcome page: information, debug modes, documentation, predefined modes, previously save sessions
- Docking/undocking from major windows
- Standalone window configuration
- One-click access to multiple source files within a single window
- Customizable toolbar, menu and hot keys for natural fit with debug tasks
- Spotlight search for fast, efficient navigation of commands, manuals and preference settings
- Skin color setup, including a dark mode

## Advanced Debug Features

Synopsys Verdi also includes the following advanced debug features:

- Automatic tracing of signal activity to quickly trace activity across many clock cycles with powerful behavior analysis technology
- Temporal flow views provide a combined display of time and structure to help rapidly understand cause-and-effect relationships
- Assertion-based debug with built-in support for assertions facilitates quick traversal from assertion failure to related design activity
- Various abstraction views, including transaction level
- Clock tree and clock domain analysis
- Built-in message logging and automated testbench transaction recording capabilities
- Decision tree for debug that can be visually built and manually executed
- AI-based regression debug automation (RDA) to determine root cause of regression bugs
- Intelligent debug acceleration (IDX) to enable reuse of waveforms, accelerating time to debug
- Applications (apps) via an API

In addition, there are specialized technologies for analysis of the following: power-aware, performance of system-level protocols, analog-mixed signal, hardware-software simulations, emulation and prototyping solutions, and fault simulation.

## Integrated Design Environment

The Verdi platform provides an integrated design environment (IDE). Based on Synopsys Euclide, the IDE allows users to find bugs earlier and optimizes code for design and verification flows by identifying complex design and testbench compliance checks during SystemVerilog and Universal Verification Methodology (UVM) development.

The IDE provides correct-by-construction code development through context specific autocompletion and content assistance that is tuned for Synopsys VCS® simulation, Verdi debug, and ZeBu® emulation, to improve code quality during the entire project development cycle.

The IDE features on-the-fly incremental compilation, elaboration, pseudo-synthesis and rule checking, all of which are integrated into the editor and provide feedback in seconds. Additionally, it helps to minimalize implementation bugs in RTL and testbench, improving project convergence rate and eliminating patchy code.

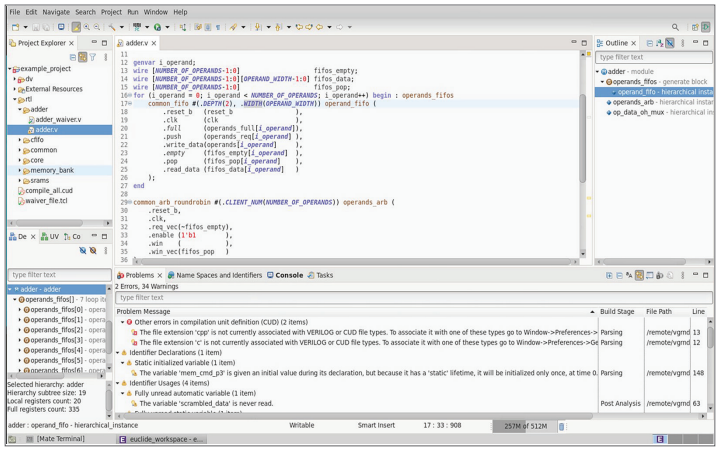

Figure 4: The Synopsy Verdi platform IDE

### IDE Features

The Verdi platform includes the following advanced IDE features:

- Runs on-the-fly checking while typing code, typically takes seconds to produce feedback
- Use context specific autocompletion and content assistance
- Reference signals, parameters, and structure/class members
- Instantiate modules and interfaces with all parameters and ports
- View, review and navigate the design and testbench hierarchy, various data types
- Search semantics, easily see with coloring

## Verification Management System

The Verdi platform includes a verification management system (VMS) as well as integrates with Synopsys VC Execution Manager. The VMS is a powerful and flexible system for managing compilation, regression test execution, data collection, reporting and tracking of the design verification process. It automates coverage-driven SoC functional verification flows, tracks and collects regression results data in a relational database, and supports annotation of coverage results in the Verdi planner.

The VMS is preconfigured to work with VCS flows, and is natively integrated with Verdi's planner and coverage features. The Verdi VMS can also scale to support any number and any size of projects and users, is fully extensible and customizable, and provides full reporting capabilities, including API access for custom reports.

## Broad Language Support

The Verdi platform supports the following verification languages and methodologies:

- Verilog, SystemVerilog, and SystemVerilog assertions (SVA)
- VHDL
- Portable Stimulus
- Universal Verification Methodology (UVM)
- Unified Power Format (UPF)
- Synopsys Design Constraints (SDC)
- Mixed-signal simulation support

### Ecosystem and Interoperability Support

The Verdi platform supports a vast array of solutions available from Synopsys as well as third-party vendors including HDL software simulators, emulators, prototyping, formal verification and more. For many of these solutions, the connection is provided via application programming interfaces (APIs) to the FSDB database. APIs provide open access to both databases and command-and-control mechanisms, enabling easy integration with the Verdi system with other verification tools and design environments.

The APIs also enable a rich array of Apps to tailor Verdi deployment for user flows. The customizable menu/toolbars enable the Apps functions in the Verdi user interface. In addition, direct launch of Apps-enabled third-party tools/scripts is possible from the debug environment.

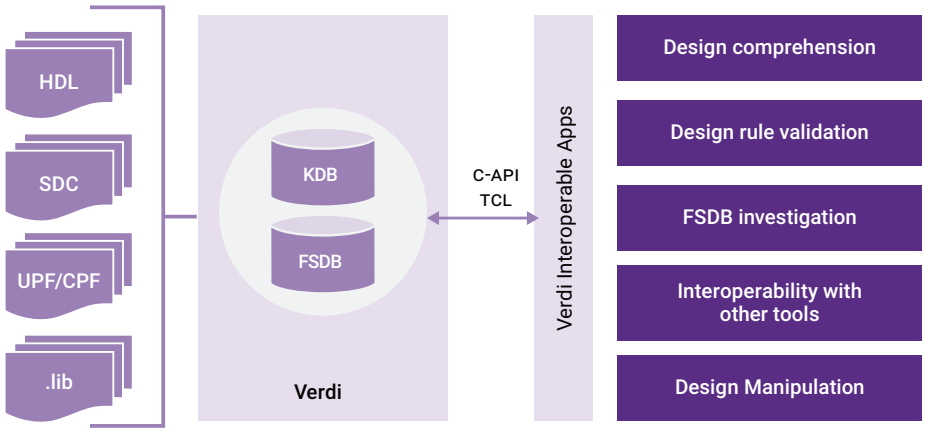


Figure 5: Apps further extend Synopsys Verdi functionality

**For more information about Synopsys products, support services or training, contact your local sales representative or visit us on the web at: www.synopsys.com**