SYNOPSYS®

DATASHEET

# Euclide: IDE with Design and Testbench On the Fly Checks

## Design and verification integrated development environment

## Overview

The Synopsys Euclide solution is an integrated development environment intended for chip designers and verification engineers. It helps to cut project time, avoid re-spins, improve code quality, and reduce chip area and power..

## Introduction

Synopsys' Euclide solution is the only tool featuring on-the-fly incremental compilation, elaboration, pseudo-synthesis and rule checking - all of which are integrated into the editor and provide feedback in seconds. With Euclide's toolset you can immediately minimalize implementation bugs in design and testbench, thus substantially improving project convergence rate and eliminating patchy code.

Designed to be simple and intuitive, the tool offers unparalleled user experience, allows immediate and effortless ramp-up, and ensures a fast learning curve for beginners.

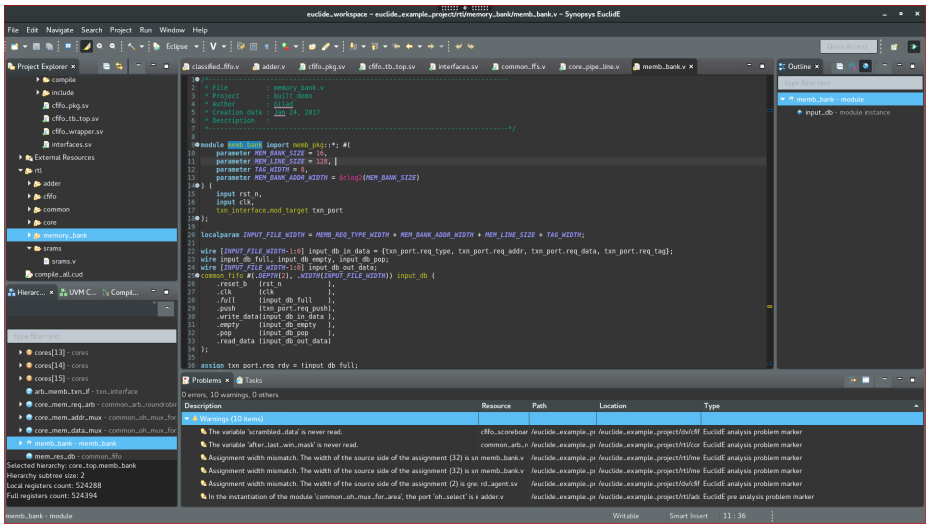

Figure 1: Synopsys' Euclide integrated development environment for SystemVerilog

## Fastest Design Rule Checker and Unrivaled Testbench Linter

Immediately minimalize implementation bugs using a comprehensive rule checks which run on-the-fly while typing code, typically taking seconds to produce feedback. Avoid many unnecessary simulations and lengthy debug sessions. Prevent re-spins by uncovering testbench bugs that were masking corresponding RTL errors, and would otherwise make their way to silicon.

- Runs on-the-fly while typing code, typically takes seconds to produce feedback
- Errors and warnings marked directly on relevant code
- Rules and severities are easily configurable per project, file set, or user
- Flexible and efficient black boxing and waivers mechanisms
- Performs hundreds of design and testbench checks, including:
  - Synthesizability and synthesis results, such as combinational loops, latches, clock/reset related problems, etc.
  - Drives / loads violations
  - Time-0 procedural code analysis, such as detecting unreachable procedural code, 'null' class instance access, dynamic casting failures, out-of-bounds accesses, endless loops/recursions, etc.
  - Width mismatches and other data type mismatches in assignments, port connections, function calls, etc.
  - SVA checks
  - Compliance rules for VCS®, ZeBu®, Design Compiler® NXT
  - Performance rules for VCS simulation and ZeBu emulation
  - UVM rules for non-compliant, incorrect, deprecated or suspicious code
  - Constraint checks
  - Method overrides and implementation inconsistencies or irregularities
  - Powerful custom methodology rules

## Accelerated Coding

Provide your team with an extensive toolset, enabling them to speed up coding and improve code quality.

- Context specific auto-completion and content assistance
  - Reference signals, parameters, and struct/class members
  - Instantiate modules and interfaces with all parameters and ports
  - Call functions and tasks with all arguments
  - Add overrides and provide implementations for class methods
  - Invoke API of external VIPs and of UVM/OVM/VMM libraries
- Configurable code templates
- Code formatter (e.g. auto indentation, vertical alignment, etc.)
- Code refactoring across the project
  - Quick fixes for detected problems (e.g. add missing ports in instantiations, add declarations for implicit wires, etc.

## Code Review and Navigation

Explore and shed light on your code with unparalleled user experience and a wide range of code inspection and navigation tools.

- Design hierarchy tree view
- UVM environment hierarchy tree view
- Quickly navigate to identifier declaration, find all references to identifier, and trace signal drives and loads throughout the design
- View hierarchy-dependent values of data types and parameters
- Semantic coloring
- Advanced semantic search
- File outline (e.g. table of content)

## Integrated Development Environment (IDE)

Enjoy integration and seamless compatibility with version control, bug tracking, task management, EDA tools and more.

- Eclipse-based environment
- Automated integration with version control tools
  - Automatic repository detection
  - Rule checks and other features available while working with version control features such as manual file conflict resolution or comparison between different versions of the same file
- C project integration
  - Automatically set up a unified Verilog+C project using a file list containing both
  - Hyperlinking DPI method declarations and implementation
- Integrates task management capabilities
- Integration with bug tracking tools

Synopsys' Euclide solution is compatible with and integrates into the rest of Synopsys Verification Continuum Platform:

- Immediate Euclide ramp up from setup of other tools
  - Load Euclide from simulation setup
- Rule checking for ensuring tools compatibility and enhancing tools performance
  - VCS compliance and performance
  - ZeBu compliance and performance
  - Design Compiler NXT compliance
- Run other tools from Euclide, run Euclide from other tools
  - Run VCS and Verdi®
  - Debug in Verdi and run Euclide back on same location and hierarchy

**For more information about Synopsys products, support services or training, visit us on the web at www.synopsys.com, contact your local sales representative or call 650.584.5000**

SYNOPSYS®
Silicon to Software™

©2021 Synopsys, Inc. All rights reserved. Synopsys is a trademark of Synopsys, Inc. in the United States and other countries. A list of Synopsys trademarks is available at synopsys.com/copyright.html. All other names mentioned herein are trademarks or registered trademarks of their respective owners.
02/08/21.CS609397483-EuclidE DS.