Cheetah

High Performance System Verilog Analyzer

Highlights

- Latest support of Language Standards
  - System Verilog: IEEE-1800-2012, IEEE-1800-2009, IEEE-1800-2005
  - Verilog: IEEE-1364-2005, IEEE-1365-2001, IEEE-1364-1995, OVI 2.0
  - PSL: IEEE-1850 - support for both embedded and external verification units
- Comprehensive syntax and semantic checks
- Browser and Decompilation utility for easy debugging
- Editable and Extensible object model
- RTL subset semantics checks
- Utility Objects for Expression Evaluation, Elaboration and Partial analysis
- Support for:
  - Static Elaboration
  - Incomplete designs through black box option
  - Complete source information including file name, line/column number for each construct, and user comments
- Highly customizable - error handler, user-defined attributes, user-defined meta comments and more

Addressing the needs of EDA tool developers who need to accelerate deployment of System Verilog products, Interra offers Cheetah—a high performance System Verilog analyzer. Targeted as a customizable front-end for System Verilog based applications, such as simulation, synthesis, formal verification, and code generation, Cheetah is compliant with industry standard simulators and other tools and provides backward compatibility to Verilog.

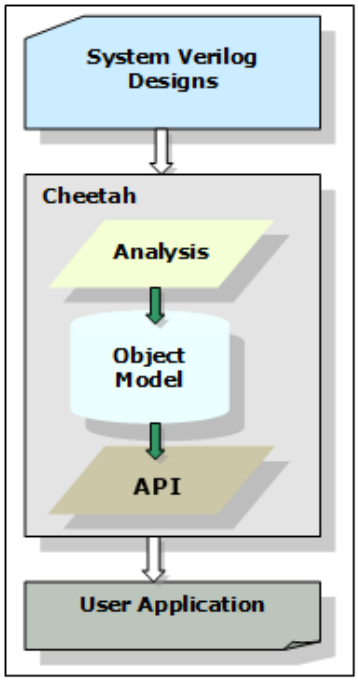
What is New

→ Support for multithreading compliant APIs for the standard IEEE 1800-2012

Cheetah is completely customizable and field proven. Cheetah provides you wide range of APIs to access designs for information, modify designs, evaluate expressions, perform elaboration, and optionally check for RTL subset compatibility.

In addition, Cheetah supports PSL analysis for both embedded PSL and external verification units. Cheetah offers control options enabling applications to work with PSL and System Verilog together.

Cheetah-SV is available on Solaris, HP-Unix, Linux, and Windows platforms. Entertain



API: Application Programming Interface

Key Advantages

- High performance System Verilog analysis
- Option to comply with any of supported standards
- Compile time type-safety helps catch errors early
- Usage of high level design patterns makes application code more concise
- Separation of flow and actions to make EDA tool development more maintainable
- Backed by Interra's field-proven expertise in developing HDL analyzers for VHDL and System Verilog

The Cheetah Features

Complete Language Support

Cheetah completely supports System Verilog IEEE 1800-2012. In addition, Cheetah supports IEEE 1800-2009, IEEE 1800-2005, IEEE 1364-2005, IEEE 1364-2001, IEEE 1364-1995 and OVI 2.0.

Cheetah also supports analysis of simple subset of IEEE 1850, V1.1 and V1.01. After analysis both PSL objects and System Verilog objects co-exist in memory.

Customizable Error Handling

The API functions enable you to customize the messages reported by Cheetah to suit application-specific needs. You can also register an error message handler to meet customized needs along with changing the severity of messages.

Elaboration

API functions enable full static elaboration (via node copy). You can also perform elaboration on interface port, bind, configuration, and instance array. API functions are also available for semantic checks on the fully/partially elaborated design.

Binary Dump & Restore

The binary dump utility enables you to dump the in-memory design and then restore the full design later. The utility supports complex type of dumps, such as package, compilation unit, and more. You can also manage memory consumed by the dumped and analyzed design units.

RTL Subset Semantic Checking

Cheetah provides a run time switch to perform RTL subset checking on a given System Verilog description. Cheetah performs subset checking in compliance with industry standard synthesis policy. If the design is RTL compliant, Cheetah annotates the object model with synthesis-specific information including inferred clock, set and reset signals, and expression sizes.

Cheetah identifies user defined meta-comments. You can register and work with user defined meta-comments.

Expression and Function Evaluation

You can use Cheetah API functions to evaluate static expression and functions. An expression having constant values can be evaluated to static constant value.

You can also determine the expression size in static functions.

Debug Tools for Development

Cheetah allows you to decompile and browse any node in the object model.

The Decompiler utility enables you to write the complete in-memory object model as a System Verilog description.

The Browser utility helps you in better understanding of object model traversal.

The Browser prints out object type and its property names and values. It also allows you to navigate Cheetah object model easily from a given System Verilog example and do partial decompilation of object from any given point of the syntax tree.

You can use the Browser to learn quickly the application programming requirements using Cheetah.

Black Box Analysis

Cheetah allows parsing and the creation of an object model from a design that has missing modules or libraries. A dummy module can be created with a list of inputs and output ports inferred through corresponding instantiations.

User Comments in Object Model

Cheetah stores all the comments, which are a part of the design, in the object model. These comments are then available to user application through the API. File name, line number, and column number information can be used to associate user comments with corresponding System Verilog constructs.

Interra Systems, Inc.
1601 S. De Anza Boulevard, Suite 212, Cupertino, CA 95014
Phone: +1 408 579 2000, Fax: +1 408 579 2050
Email: info@interradesign.com Web: www.interradesign.com