Beacon-SV-MX

Comprehensive Test Suite for Validating Mixed Language

Key Advantages

- Backed by Interra's field-proven expertise in developing HDL-based test suites for VHDL and System Verilog
- Developed in partnership with significant EDA majors
- Conforming to accepted definition and interpretation of the languages
- Providing an unbiased quality analysis of EDA tools
- Comprehensive validation of mixed language styles

Highlights

- Over 1,500 test cases along with test benches
- Detailed test plans with cross-reference to test cases
- Well organized test cases highlighting testing objectives
- Both positive and negative test cases

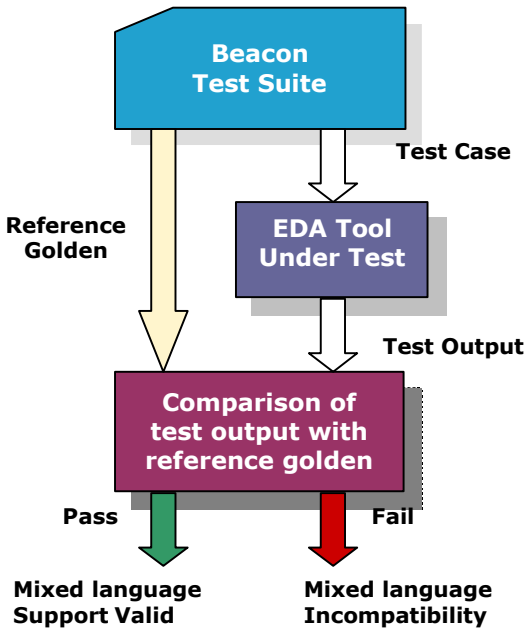
Addressing the needs of EDA tool developers to quickly evaluate the quality of their products, Interra's Beacon-SV-MX delivers a comprehensive test suite to validate mixed System Verilog/VHDL support in EDA tools.

You can use Beacon-SV-MX to validate EDA tools developed using mixed designs: mixed instantiations, data transfer between mixed interfaces, port mapping, and parameter mapping. In addition, you can characterize EDA tools for mixed language support across various VHDL to System Verilog or System Verilog to VHDL interfaces.

Enabling you to discover mixed language incompatibility early in the product development and testing life cycle, Beacon-SV-MX offers:

- Reduced development costs and time-to-market EDA products
- Development of standard-based products
- Precise evaluation of bugs and errors in the product
- Measure of product quality
- Unbiased feedback on product quality
- Regression tests for quality assurance

The test cases and test benches can be applied to the EDA tool under evaluation and results can be verified with the Beacon status provided for each test case with Beacon-SV-MX.



The Beacon-SV-MX Features

Comprehensive Test Suite

Beacon-SV-MX test suite validates mixed language interface. These test cases cover different ways in which System Verilog or VHDL design units can be instantiated in other languages. The test cases also cover transfer of data from one language to another. The test cases check for mapping of data types from one language to another. The test cases also check for various VHDL port map style mapping to that of System Verilog.

Test cases also include various combinations of VHDL generic and System Verilog parameter to pass values. A set of test cases uses defparams and scopevar to modify the parameter/generic or variables value.

Sandwich cases with System Verilog-VHDL-System Verilog and VHDL-System Verilog-VHDL check for correctness of interface data transfer across multiple language boundaries.

Negative test cases check for behavior of the test tool in case of erroneous interfaces. Test cases are distributed as follows.

Language Construct	VHDL Top	Verilog To
Bind Construct		54
Data Type Mapping		
Bit,logic,reg,byte		163
Class Member		170
Interface Data Mapping		170
Shortint,int,integer, longint		99
Structure/Union		99
Enum/String		50
Package Member		141
MisCellaneous Cases	110	83
Parameter Mapping		
NoSandwich		96
Defparam		76
Scopevar		96
Data Type Mapping		
Through Interface	141	
Through Module	223	
Through Program	118	
Generic Mapping		
Through Interface	49	
Through Module	49	
Through Program	54	
DPI in SV	148	
Total	**892**	**1297**

Well Documented Test Plans

The test plans describe all test objectives and are categorized by sections.

Top-design Based Organization

Test cases are organized based on VHDL/System Verilog top design units instantiating a mixed design unit.

Category-A: VHDL Top

The VHDL Top category includes test cases with VHDL as top entity instantiating Verilog modules/interface/ program block. Test cases check for various port map styles, data types, generic values, and configurations.

Category-B: System Verilog Top

The System Verilog Top category includes test cases with System Verilog as top module instantiating VHDL entities. Test cases check for various port sizes, parameter values, and defparams providing VHDL generics values. Sandwich cases include VHDL instances instantiating another level of System Verilog. Test cases also check parameter setting across multiple language boundaries.

Test Benches and Reference Golden

Provides test benches to instantiate test cases and apply vectors on inputs. Outputs are captured after an appropriate interval and written on to a

Sample Test Case

--** Purpose: Byte(signed/unsigned) data type is mapped with std_logic_vector of equal width.
--** TestPlan: Sections 1.1.1.11.5
--** Status: SIMULATION_SHOULD_PASS
--** Assumptions: None

```
//////////////////////////////////////////////
`timescale 1ns/1ns
`include "includeFile.v"

module withSignedUnsigned6_top(input byte
in1,byte signed in2,output byte unsigned
out1, out2);

bottom I1(in1, in2);
assign out1 = I1.b3; assign out2 = I1.b4;
withSignedUnsigned6 vhdInst(I1.b1,I1.b2,
                            I1.b3,I1.b4);
endmodule
```

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