

Automating the Generation of DID-Compatible VHDL Models (The SHARP TIREP VHDL Model Generator)

Charles K. Rogers, Author
Naval Air Warfare Center - Aircraft Division
Indianapolis, IN 46219
(317) 306-4797
Email: rogersc1@po4.nawc-ad-indy.navy.mil

Supporting TIREP Team Members

Anthony Hawkins, David Broadhead, Louie Kitcoff
Naval Air Warfare Center - Aircraft Division (NAWC-AD) Indianapolis, IN

John Miles, Gary Hout, Ed Woods, Darin York
Naval Surface Warfare Center (NSWC), Crane, IN

L. J. Ceder
Naval Research Laboratory (NRL), Washington, DC

ABSTRACT

This paper introduces a DOD-developed computer program which transforms a VHDL model, associated stimulus/response vectors and boundary data sheet information into a VHDL deliverable which satisfies key elements of the government VHDL Data Item Description (DID), DI-EGDS-80811. The computer program, developed by the Technology Independent Representation of Electronic Products (TIREP) project under the auspices of the Sustainable Hardware and Affordable Readiness Practices (SHARP) Program, is called the TIREP Model Generator (TMG). The government DID, like rigorous commercial DIDs, can require standalone testbenches for each device, circuit card, assembly or specified logical and physical groupings of interest. In addition, design timing, electrical and physical data may be required in the VHDL deliverable.

The TMG reads a user-generated "core" VHDL model and then prompts the user to enter design electrical, timing and physical parameters. A TMG "core" VHDL model is typically the functional VHDL entity / architecture pair (which can include timing) used for synthesis. The TMG generates a DID-compatible TIREP VHDL model. A TIREP VHDL model includes a simulatable WAVES testbench and a modified EIA-567A electronic data sheet VHDL model which documents electrical, timing and physical data and also performs timing violation checks during simulation. The user need only supply the WAVES vector file which is derived from stimulus/response vectors found in the specification or sampled from a core model simulation run.

This self-documenting approach to VHDL model development yields an executable specification suitable for procurement, hardware design development, evaluating back-annotation synthesis timing results, design archiving and providing life-cycle design documentation. Additionally, DID-compatible VHDL deliverables (with WAVES testbenches) can be generated across physical and logical partitions by structurally connecting design entities and invoking the TMG on this new core design. The TMG provides version control and stylize utilities and supports bi-directional and tri-state I/O. The TMG virtually eliminates the learning curve and engineering expense associated with the manual generation of DID-compatible VHDL models. Whether synthesizing or specifying with VHDL, the TMG provides an automated means to generate a comprehensive VHDL deliverable. This paper reviews the capabilities, operation and applications of the TMG.

INTRODUCTION

The TIREP project was initiated in 1992 under the SHARP program, as a joint effort between the Naval Research Lab (NRL), the Naval Surface Warfare Center, Crane Division (NSWC) and the Naval Air Warfare Center, Aircraft Division, Indianapolis (NAWC-ADI). The primary objective of TIREP was to develop a modeling methodology which would provide a performance based specification in a technology independent format for modules (e.g. electronic assemblies) faced with obsolescence problems. This performance specification could then be utilized to develop a new design for the existing module, employing conventional design techniques and/or VHDL synthesis capabilities to generate an implementation model for the module. The implementation model would then be used by manufacturing engineering to generate the necessary tooling and documentation to support the fabrication of the replacement module.

In order to accomplish these objectives, the TIREP team relied heavily on existing standards and specifications to form the basis of the modeling approach. In this effort, the following standards have been employed:

- IEEE-STD-1076 VHDL Language Reference Manual
- IEEE-STD-1164 Multivalued Logic System for VHDL Model Interoperability
- IEEE-STD-1029.1 Waveform and Vector Exchange Standard (WAVES)
- EIA-567A VHDL Hardware Component Modeling and Interface Standard

Key goals for the developed models included:

- The model shall capture as much information from the module specification as possible.
- The model shall meet the requirements of the VHDL Data Item Description (DI-EGDS-80811).
- The model shall provide information necessary to support the development of manufacturing tooling.

In September 1994, the TIREP team released "A VHDL Modeling Guide", TP-804, Version 1.0 which was based upon the EIA-567 modeling standard. This was followed in January 1995, by Version 2.0 which employed EIA-567A features and capabilities. As a result of the development of this Modeling Guide, a supplemental task has been undertaken by the TIREP team, to automate the generation of TIREP models based upon the approach presented in the Modeling Guide. The result of this effort is the TIREP Model Generator (TMG) which is currently available for Beta site evaluation.

THE TIREP MODELING APPROACH

The TIREP modeling approach starts with a core VHDL model defined by the user. This core model may be behavioral, representing a single entity, or structural, representing an assembly of components, or a combination thereof. For this core model, a design specification package is developed which contains timing, electrical and physical information for the module being described. The design specification package is then referenced through the Electronic Data Sheet (EDS) Model which forms a timing shell around the core model, providing propagation delays, transition times, synchronous and asynchronous constraint checking for the model. The EDS model is developed based upon guidance provided by EIA-567A.

Due largely to requirements imposed on the TIREP models, the EIA-567A modeling approach has been modified/enhanced to support this project. Key changes to the EIA-567A modeling approach are identified as follows:

- Combined the EIA567EV, TV and PV packages into a single EIA567A package.
- Combined the design electrical, timing and physical view packages into the design specification package.
- Added transition times, "tthl" and "ttlh", to the "DELAY" enumerated type declaration.
- Added "ESD_CLASS" enumerated type declaration.
- Added the "DELAY_PIN" procedure to the TIMING_PACK package to support transition time modeling.
- Converted the "OPERATING_POINT_TYPE" enumerated type declaration to a "NATURAL" subtype and moved it to the design specification package allowing the user to declare as many operating points as necessary for a given model.

- Allowed the use of timing "classes", including a default class, in lieu of fully enumerating timing paths in the model. In most models, this significantly simplifies the generation of timing specifications for the model.
- Moved the "PIN_TO_SIGNAL" declaration from the design physical view or design specification packages to the design electronic data sheet model making it readily available for the development of manufacturing tooling.
- Added "UPPER_TEMPERATURE_LIMIT", "LOWER_TEMPERATURE_LIMIT", "PACKAGE_DESCRIPTION" and "ESD_PROTECTION" constant declarations.
- Added the nominal voltage, "vnom", to the "POWER_LIMIT" record declaration.
- Deleted "PIN_LIST_PV" declaration (duplicate of "PIN_LIST" declaration and no longer needed when packages were combined).
- Added "DC_SIGNAL_LIMITS_TYPE" and "DC_SIGNAL_LIMITS" declarations to link pin electrical characteristics to pins and operating points.
- Incorporated limited power supply modeling and applied it to the outputs in the EDS model.

Once the EDS model is developed, the TIREP modeling approach wraps a WAVES testbench around this model to provide a test environment. The WAVES testbench is based upon a collection of standard WAVES packages and design packages which are integrated into the testbench to automatically stimulate the EDS model and compare received responses against expected responses for accuracy. A few minor variations have been implemented in the WAVES testbenches, as follows:

- The variable "FILEEND" has been added to the waveform generator procedure which allows the end of the vector file to be detected, providing a trigger for a pass/fail report of the testbench.
- A single monitor process is used which monitors all model outputs.
- An "ERRCNT" signal is declared which tracks the number of errors which are detected in the expected versus received responses from the model.

The result of all this is an integrated TIREP model as shown in Figure 1. This model is complete, with physical and electrical information about the module and simulatable using specified timing parameters and constraints through a WAVES Testbench.

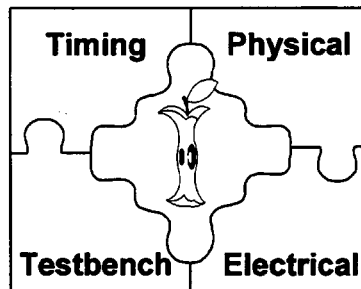


Figure 1. The Integrated TIREP Model

THE TIREP MODEL GENERATOR

The TIREP Model Generator is a DOS based, menu driven program which generates an integrated TIREP model as shown in Figure 2. To do this, the TMG reads port/generic declarations from a core VHDL model and based upon these declarations along with design specification information provided by the user, will generate a design specification package and an Electronic Data Sheet (EDS) shell for the core model. The TMG will also create a WAVES testbench for the generated EDS model and/or the core model, based upon vector information provided by the user.

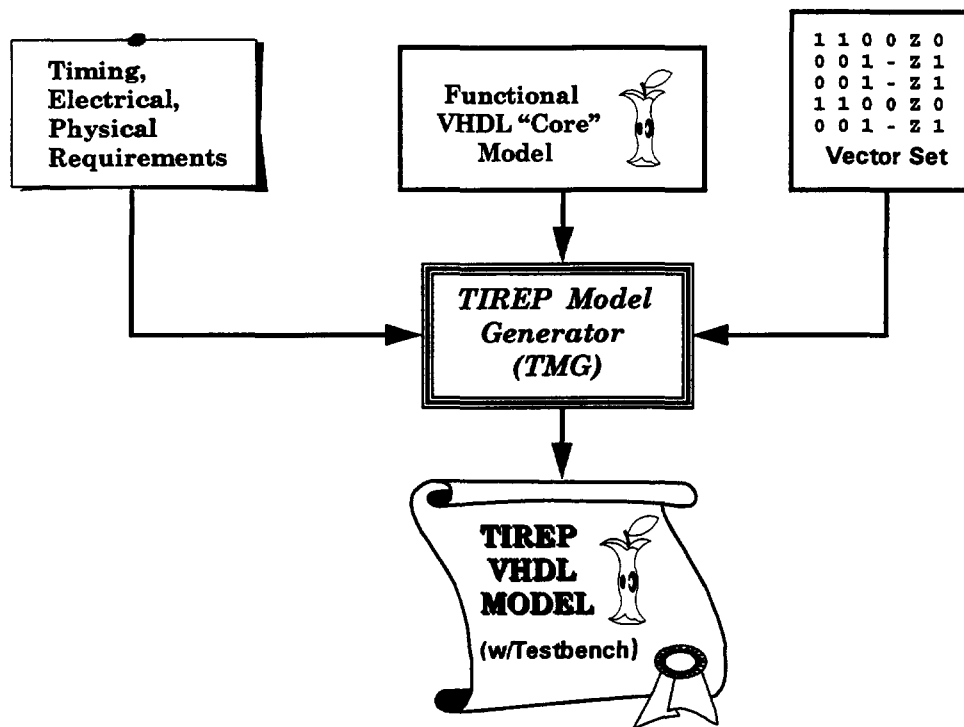


Figure 2. The TIREP Model Generator

Requirements

Although the TMG will work with almost no information about the design (apart from the core model), a lack of this information will result in a model which does not meet the requirements of a TIREP Model and would not be considered DID compatible. The TMG requires the following elements in order to generate a complete TIREP Model:

- A VHDL core model entity declaration. This core model may be a leaf level behavioral model, or a structural assembly level model or a combination thereof, from any level in a design hierarchy. Although the TMG will operate without an architecture for the core model, this architecture will be needed before the resulting TIREP model can be simulated.
- A vector file. A vector file must be provided by the user. The TMG will accept a WAVES vector file named according to the convention <des>_VEC.DAT, where <des> is the 3 to 4 character design designator for the model. Alternatively, the TMG can create a WAVES vector file from a vector table or appropriately formatted list file. The vectors for the vector file may be obtained from specification/test information on the module, or it may be sampled from a core model simulation run. If a WAVES vector file is provided, the TMG features a utility which will check the file to insure that it should work with the testbench.
- Design specifications. The TMG makes use of design specifications in the developed model. Design specifications include elements such as propagation delays, timing constraints, input and output pin characteristics, operating conditions and environmental factors. These elements may be entered manually through the TMG program, or a special capability is provided which will allow most of these elements to be read from a comma delimited spreadsheet file.
- Design designator. A 3 to 4 character design designator is required by the TMG. This designator will be used for naming files, entities and subprograms created by the modeling process.
- Platform. A PC/AT compatible computer with 650Kbytes of hard disk space.

Limitations

Following are the current identified limitations of the Beta version TMG:

- 32,767 pins.
- 250 propagation delay/constraint/IO characteristic classes.
- 250 synchronous/asynchronous constraints.
- 250 propagation delay paths/IO characteristic class assignments (other than the default).
- Vector, integer, Boolean, real, etc. signal types are not supported at the model ports. Only bit, std_logic, std_ulogic and declared subtypes are supported at the model ports.
- Std_logic signals must be used for bi-directional and tri-state port signals.
- 28 characters allowed for signal names, pin names and pin numbers (EIA_STRING limit)
- Core generics are carried to the testbench for the core testbench only. They are carried to the EDS model for the complete model. They are not mapped within the EDS model, however. They may be mapped by the user, if desired.

Model Completion

Unfortunately, there are a few items which the TMG cannot read from the mind of the user or otherwise extract about the model. It is up to the TMG user to look at these items and fill in the blanks when necessary. Following is a brief description of those items which need to be considered for each model developed using the TMG:

- In the design specification package (<des>_DESP.VHD), test load circuit classes need to be defined, when used. This is a textual description of the load circuits applied to module outputs during test in order to verify proper operation of the outputs. The package body for this file contains a comment for "user defined pin load conditions" identifying the location for this description.
- If the core model contains generic declarations which need to be carried through the hierarchy, it is left to the user to declare and map these generics in the EDS model and in the testbench. Currently, the TMG will carry core generics to the core testbench if generated. Core generics will be carried to the core component declaration in the EDS model, however, they will not be mapped when the component is instantiated. This implies that the generics should contain a default assignment within the core model.
- The TMG automatically implements 4 operating points (0 - zero delay, 1 - minimum, 2 - nominal, 3 - maximum), by default. If operating points have been defined by the user beyond the required 0 to 3 operating points, the user should describe these operating points in the comments of the design specification package (<des>_DESP.VHD), the EDS model (<des>_EDS.VHD) and in the testbench file (<des>_TSTT.VHD). These operating points will be flagged as user defined operating points in these files.
- If TIREP file headers are attached to the core model or to a user defined WAVES vector file, it is necessary that the user edit the headers of these files to record specific model information such as purpose and use, limitations, required packages, etc.

Model Generation

Once the required information is provided, it is read by the TMG and used to assemble the following files and packages. The WAVES Objects package (<des>_OBJ.WAV) is considered to be a standard package, however it does contain design specific library references.

File	Description
<des>_DESP.VHD	Design specification package
<des>_EDS.VHD	Electronic Data Sheet model
<des>_DUTP.WAV	WAVES Device Under Test package
<des>_OBJ.WAV	WAVES Objects package with appropriate library references
<des>_WAVP.WAV	WAVES Waveform Generator package
<des>_TSTT.VHD	Testbench model
<des>_VEC.DAT	WAVES Vector file
<des>_HDR.WAV	WAVES Header file

The TMG also provides copies of common and standard packages which are used by the model, when requested by the user. Standard packages are documented in corresponding IEEE standards. They

are included in the TMG for convenience. Common packages have been developed for use with TIREP models and require no design specific modifications. Following is a listing of the standard and common packages included with the TMG.

File	Description
STDLOGIC.VHD	Standard STD_LOGIC_1164 package
WAVSYSS.WAV	Standard WAVES System package declaration
WAVSYSB.WAV	Standard WAVES System package body
WAVSTDS.WAV	Standard WAVES Standard package declaration
WAVSTDB.WAV	Standard WAVES Standard package body
WAV_INT.WAV	Standard WAVES Interface package
EIA567AP.VHD	This package is common to all TIREP models. It is based upon elements from the EIA567EV, TV and PV packages along with some TIREP specific additions/variations.
TIMPAKA.VHD	This package is common to all TIREP models. It is based upon the TIMPAK.VHD package developed for EIA-567A.
WAV_LOGP.WAV	Common WAVES Logic package
WAV_FRMP.WAV	Common WAVES Frame package

Version Control

The TMG features manual or automatic version control. With automatic version control, each time a model file is generated, the TMG searches the working directory for an existing version of the file. If one is found, the TMG will automatically increment the version number and request version comments from the user. The updated model file will then be generated with the new version number and a complete copy of the version history. With manual version control, the version number will also be requested from the user.

Calculation of Timing Parameters

Generally speaking, it is not necessary that the user enter timing information for all operating points. The TMG has a built-in calculation feature which is based upon the technology factors provided in EIA-567A, Appendix A. If timing information is entered for 1 or 2 of the nominal, minimum or maximum operating points, the TMG will calculate the remaining required operating points based upon the technology selected for the current design. An exception to this occurs for synchronous and asynchronous constraints for which the nominal operating point (2) is the only point specified. Under this condition, the constraint value entered for the nominal operating point will be applied for the minimum (1), nominal (2) and maximum (3) operating points. This allows the user to specify a maximum constraint limit once and have it applied at all operating points.

Model Analysis and Simulation

The complete model analysis tree is depicted in Figure 3. This figure identifies all files and packages which must be analyzed (including the order of analysis) in order to test/simulate the complete model. Once model analysis is complete, the model may be simulated using the corresponding VHDL simulator. As the model is simulated, any discrepancies between the expected and observed results will be reported. Once the vector file is exhausted, a testbench pass/fail report will be issued.

The WAVES testbench provided through the TMG has proven to be an effective aid in the evaluation of post-synthesis models. When provided by vendor place and route tools, a back-annotated VHDL output file can be analyzed as the EDS model for the WAVES testbench. This model can then be exercised by the testbench to insure that the synthesized design will meet the operational requirements of the model.

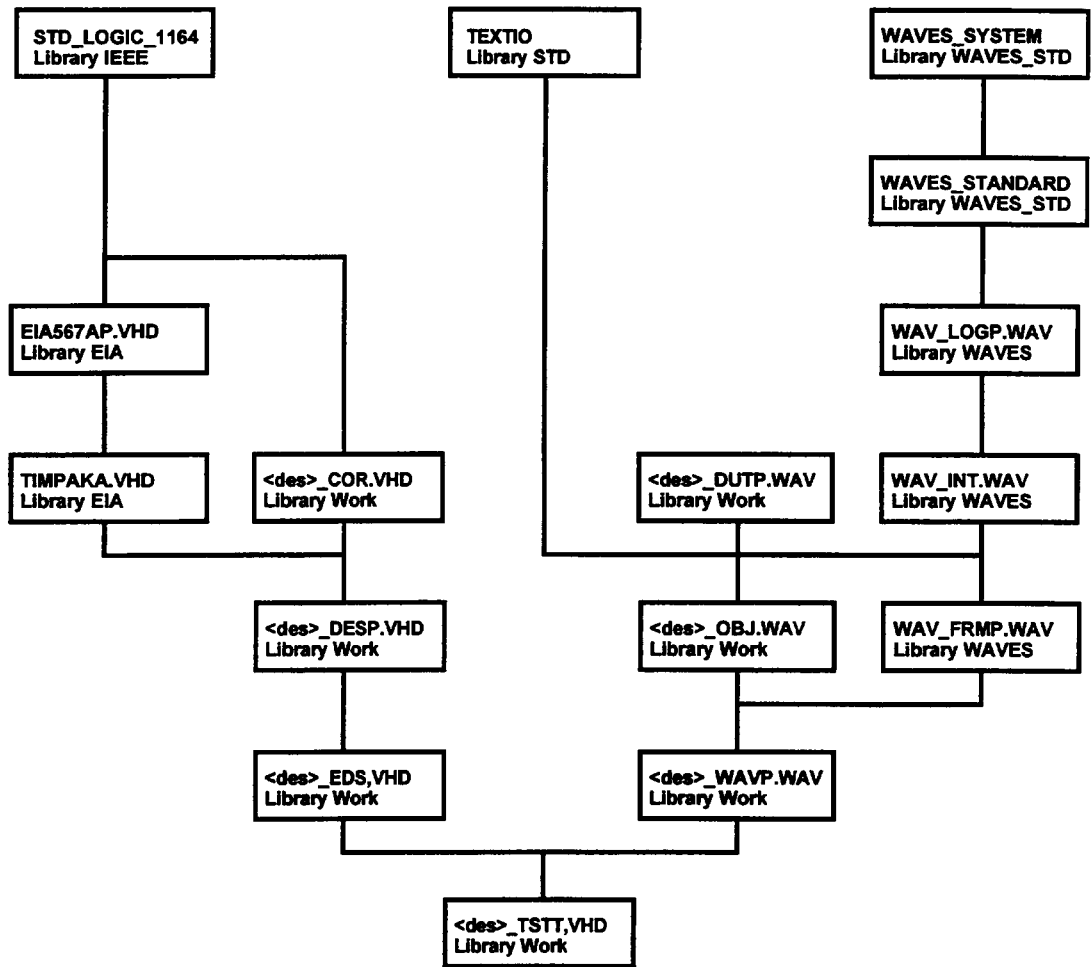


Figure 3. TIREP Model Analysis.

Modeling Style

In addition to its other features, the TMG features a “stylize” utility which may be applied to any or all VHDL files. This utility can be used to give VHDL files a consistent appearance by standardizing indentation levels, capitalizing key words, and moving comments onto their own line. Since this utility does not try to interpret the VHDL code, it does require that the user follow some basic guidelines when generating code. This utility will work with all TMG generated model files.

MODELING HIERARCHY

The TIREP Model Generator is not restricted to use with component level models. It works as easily with structural level models as it does with leaf level ones. Hence, the TMG will operate with component, sub-assembly, assembly, sub-system and system level models. Figure 4 depicts how a modeling hierarchy can be developed. In this case, an EDS core component is generated by the TMG using the component core model. From this point, a structural circuit card assembly model is constructed which forms the core for another pass through the TMG. At each level in the hierarchy, the TMG may be used to create an EDS model and testbench. The EDS models may then be used as components in higher level models.

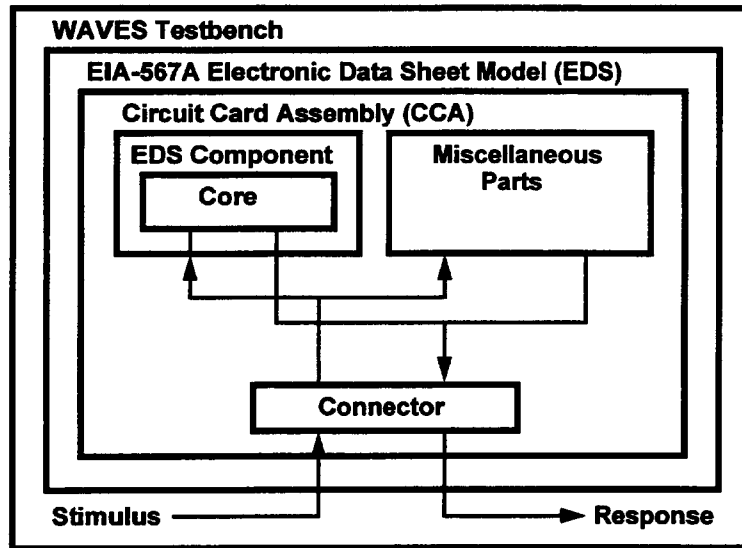


Figure 4. Modeling Hierarchy.

DID COMPLIANCE (DI-EGDS-80811)

During the development of the TIREP modeling process, a significant amount of effort was expended trying to develop a model which would be compliant with the VHDL Data Item Description (DID) DI-EGDS-80811. Pertaining to this, the following notes are made regarding the compliance of TIREP models to this document.

The TIREP models are fully compliant with DI-EGDS-80811 in the following areas:

- The entity declaration for the EDS model contains the port declaration, input wire and output load delay generics, the allowable operating points and supporting comments. (reference DID paragraph 10.2.2)
- The interface declaration in the EDS model contains a "pin_to_signal" correlation table which documents the relationship between pin name and pin number. (reference DID paragraph 10.2.2.1)
- Naming conventions employed by the TMG are consistent from one model to the next. Each model/entity name generated is based upon a 3 to 4 character design designator specified by the user. Note: It is possible that this 3 to 4 character designator may be deemed inadequate to fully meet the entity naming conventions requirement. (reference DID paragraph 10.2.2.4)
- Timing characteristics are detailed in the design specification package, and are modeled in the EDS model. A minimum of 4 operating points are required for the TIREP modeling approach. These include zero (0), minimum (1), nominal (2), and maximum (3) timing. Additional operating points may be declared by the user, if desired. (reference DID paragraph 10.2.3.2)
- VHDL simulation support is provided through the WAVES testbench generated for the EDS model. The WAVES testbench will apply stimuli to the module under test (the EDS model) and compare the module response with the expected output, reporting any discrepancies observed. (reference DID paragraphs 10.2.5 and 10.2.5.1)
- Fairly extensive explanatory comments are included in the TIREP model files. (reference DID paragraph 10.2.7)
- The TMG has a built in revision management feature which retains a revision history for each design unit. Unless otherwise specified in the revision comments, the performing individual is expected to be the author. (reference DID paragraph 10.2.8.1)

The TIREP models include the following information required by DI-EGDS-80811, however it does not reside in the location dictated by the DID:

- Propagation delays, synchronous and asynchronous constraints are detailed in the design specification package. Synchronous and asynchronous constraints are tested in the EDS model

- and errors are reported if these constraints are violated. (reference DID paragraphs 10.2.2 and 10.2.2.2)
- Electrical requirements are recorded in the design specification package, but are not tested in the EDS model, with the exception of the power supply. In this case, the presence of a declared logic level (either '1' or '0') on the power supply inputs will provide valid power supply operation. (reference DID paragraphs 10.2.2 and 10.2.2.2)
- The core component is referenced in the EDS model header and declared in the EDS model architecture. (reference DID paragraph 10.2.2)
- Operating conditions are recorded in the design specification package. (reference DID paragraph 10.2.2.3)
- Timing constraint errors are reported by the EDS model, expected output versus received output errors are reported by the testbench. There is no specific report in the error message that identifies the module in which the error (assert message) was generated. (reference DID paragraph 10.2.6)

The TIREP models support the following modeling approaches or practices, however compliance with DI-EGDS-80811 is the responsibility of the user:

- Module hierarchy is supported within the TIREP modeling approach. (reference DID paragraph 10.2.1)
- Behavioral bodies are supported within the TIREP modeling approach. Responsibility for the behavioral body and for decomposition of behavioral bodies is the responsibility of the user. (reference DID paragraphs 10.2.3 and 10.2.3.1)
- Structurally dependent signal values are the responsibility of the user. The TMG will accommodate them, when present. (reference DID paragraph 10.2.3.3)
- The TMG will accept structural and/or behavioral cores. The structural body (core) and structural naming conventions are the responsibility of the user. Note: The naming convention for the EDS model is pre-defined by the TMG. (reference DID paragraphs 10.2.4 and 10.2.4.1)
- Test requirement correlation is the responsibility of the user. (reference DID paragraph 10.2.5.2)
- The TMG will generate a WAVES testbench for any and all core models, at whatever level in a VHDL hierarchy they occur. It is the responsibility of the user to generate these models for all applicable modules. (reference DID paragraph 10.2.5.3)
- Explanatory comments for the core models are the responsibility of the user. (reference DID paragraph 10.2.7)
- The list of VHDL modules in a delivery is the responsibility of the user. VHDL modules created by this program are identified in the program documentation. (reference DID paragraph 10.2.8)
- The user is responsible for maintaining the revision history for core models. (reference DID paragraph 10.2.8.1)
- Supporting text files which accompany a VHDL deliverable are the responsibility of the user. (reference DID paragraph 10.3)

SUMMARY

This paper has introduced a software utility christened the TIREP Model Generator. This program is used to semi-automatically build a timing shell and testbench around a user defined core VHDL model eliminating much of the tedious and time consuming effort normally associated with this type of model. This tool is useful for completing the documentation of a VHDL model, and if deemed acceptable by the sponsor, may be used to fulfill requirements for models which meet the specifications of the VHDL Data Item Description (DI-EGDS-80811).

FOR MORE INFORMATION

For additional information on the SHARP TIREP project status and developments, feel free to visit the TIREP home page at "kraken.crane.navy.mil/tirep.htm".

For information or questions about the TIREP Model Generator, contact Chuck Rogers at "rogersc1@po4.nawc-ad-indy.navy.mil" or phone (317) 306-4797.

ACKNOWLEDGMENTS

The work of personnel from NSWC and NAWC was funded by Mr. Mike Grubb, SHARP Program Office. The work of personnel from NRL was supported by Dr. Ingham Mack, Office of Naval Research.

REFERENCES

- [1] "IEEE Standard VHDL Language Reference Manual, IEEE Std 1076-1993", June 6, 1994, The Institute of Electrical and Electronics Engineers, Inc., New York, NY
- [2] "IEEE Standard for Waveform and Vector Exchange (WAVES), IEEE Std 1029.1-1991", September 28, 1992, The Institute of Electrical and Electronics Engineers, Inc., New York, NY
- [3] "IEEE Standard Multivalued Logic System for VHDL Model Interoperability (Std_Logic_1164), IEEE Std 1164-1993", May 26, 1993, The Institute of Electrical and Electronics Engineers, Inc., New York, NY
- [4] "VHDL Hardware Component Modeling and Interface Standard, EIA-567A", May, 1994, Electronic Industries Association, Washington DC
- [5] "VHSIC Hardware Description Language (VHDL) Documentation, DI-EGDS-80811", May 11, 1989, NSPAWAR-324
- [6] "A VHDL Modeling Guide, TP-804, Version 2.0", January 31, 1995, SHARP TIREP Project Team