IEC 62526

Edition 1.0 2007-11

INTERNATIONAL STANDARD

IEEE 1450.1™

Standard for Extensions to Standard Test Interface Language (STIL) for Semiconductor Design Environments

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

PRICE CODE XF

ICS 25.040

ISBN 2-8318-9348-8

CONTENTS

FOREWORD....5
IEEE Introduction....8
1. Overview....9
1.1 Scope....10
1.2 Purpose....11
2. Definitions, acronyms, and abbreviations....11
2.1 Definitions....11
2.2 Acronyms and abbreviations....12
3. Structure of this standard....12
4. STIL syntax description....13
4.1 Reserved words....13
4.2 Reserved characters....14
4.3 Reserved UserFunctions....15
4.4 Signal and group name characteristics....15
4.5 STIL name spaces and name resolution....16
5. Expressions....17
5.1 Constant and variable expressions....17
5.2 Expression delimiters—single quotes and parentheses....17
5.3 Arithmetic expressions—integer, real, time, boolean....19
5.4 Pattern data expressions....20
5.5 Expression processing....21
5.6 Boolean—boolean_expr....26
5.7 *Integers—integer_expr*....26
5.8 Logic expressions—logic_expr....27
5.9 Real expressions—real_expr....28
5.10 Addition to timing expressions—time_expr....29
5.11 SignalVariables—sigvar_expr....30
5.12 Formal parameters in procedures and macros....32
5.13 Integer lists—integer_list....32
6. Statement structure and organization of STIL information....33
7. STIL statement....33
7.1 STIL syntax....34
7.2 STIL example....34
8. UserKeywords statement....34
8.1 UserKeywords syntax....34
8.2 UserKeywords example....34

Published by IEC under licence from IEEE. © 2005 IEEE. All rights reserved.

9. Variables block 35

9.1 Variables block syntax 35
9.2 Variables example 37
9.3 Variables scoping 37
9.4 Variables synchronizing 39

10. Signals block 40

10.1 Signals block syntax 40
10.2 Signals example 40
10.3 Bracketed signal notation enhancement 40

11. SignalGroups block 43

11.1 SignalGroups syntax 43
11.2 SignalGroups, WFCMap, and Variables example 43
11.3 Default WFCMap attribute value 44
11.4 Defining indexed signal groups 44

12. PatternBurst block 45

12.1 PatternBurst syntax 45
12.2 PatternBurst example 47
12.3 Tiling and synchronization of patterns 48
12.4 If and While statements 50

13. Timing block and WaveformTable block 51

13.1 Additional domain specification 51
13.2 CompareSubstitute operation—s, S 51

14. ScanStructures block 52

14.1 ScanStructures syntax 52
14.2 Scan cell naming—cell_ref, chain_ref, cell_group, chain_group 55
14.3 Scoping rules for ScanStructure blocks 56
14.4 Example indexed list of scan cells 57
14.5 Example of ScanChainGroups and ActiveScanChain 57
14.6 Scan chain segments and cell groups 59

15. Pattern data 60

15.1 Data content read back—\C, \D, \E, \S, \U, \W 61
15.2 Vector data mapping and joining—\m, \j 63
15.3 Specifying event data in a pattern—\e 65
15.4 Using expressions within pattern data 66

16. Pattern statements 67

16.1 Additional Pattern syntax 68
16.2 Vector data constraints—F, E 69
16.3 Shift and LoopData statements 70
16.4 Loop statement using an integer expression 72
16.5 MergedScan function 73

Published by IEC under licence from IEEE. © 2005 IEEE. All rights reserved.

17. Procedure and macro data substitution 73

17.1 Nested procedure and macro cells 73
17.2 Passing parameters to variables 74
17.3 Default value of formal parameters 75
17.4 Data substitution using WFCConstant and SignalVariable 75

18. Environment block 77

18.1 Environment syntax 77
18.2 MAP_STRING syntax 79
18.3 NameMaps example 79
18.4 Compact scan-cell mapping using InheritNameMap 81

19. Pragma block 82

19.1 Pragma syntax 82

20. PatternFailReport 82

20.1 PatternFailReport syntax 83
20.2 PatternFailReport example 84

Annex A (informative) Glossary 84

Annex B (informative) Signal mapping using SignalVariables 87

Annex C (informative) Using logic expression with signals 91

Annex D (informative) Using boolean expressions in patterns 92

Annex E (informative) Variables and expressions in algorithmic patterns 93

Annex F (informative) Using AllowInterleave 95

Annex G (informative) Vector data mapping using \m 98

Annex H (informative) Vector data joining using \j 101

Annex I (informative) Block data collection 104

Annex J (informative) Using Fixed and Equivalent statements 106

Annex K (informative) Independent parallel patterns 108

Annex L (informative) Applications using new ScanStructures syntax 110

Annex M (informative) BreakPoints using MergedScan() function 114

Annex N (informative) Labels and X statements for diagnostic feedback 117

Annex O (informative) Use of STIL.1 for specific applications 120

Annex P (informative) Bibliography 122

Annex Q (informative) List of participants 123

Published by IEC under licence from IEEE. © 2005 IEEE. All rights reserved.

INTERNATIONAL ELECTROTECHNICAL COMMISSION

STANDARD FOR EXTENSIONS TO STANDARD TEST INTERFACE LANGUAGE (STIL) FOR SEMICONDUCTOR DESIGN ENVIRONMENTS

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
6) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC/IEEE 62526 has been processed through Technical Committee 93: Design automation.

The text of this standard is based on the following documents:

IEEE Std	FDIS	Report on voting
1450.1(2005)	93/248/FDIS	93/259/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

The committee has decided that the contents of this publication will remain unchanged until the maintenance result date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

Published by IEC under licence from IEEE. © 2005 IEEE. All rights reserved.

IEC/IEEE Dual Logo International Standards

This Dual Logo International Standard is the result of an agreement between the IEC and the Institute of Electrical and Electronics Engineers, Inc. (IEEE). The original IEEE Standard was submitted to the IEC for consideration under the agreement, and the resulting IEC/IEEE Dual Logo International Standard has been published in accordance with the ISO/IEC Directives.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an IEC/IEEE Dual Logo International Standard is wholly voluntary. The IEC and IEEE disclaim liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEC or IEEE Standard document.

The IEC and IEEE do not warrant or represent the accuracy or content of the material contained herein, and expressly disclaim any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEC/IEEE Dual Logo International Standards documents are supplied “AS IS”.

The existence of an IEC/IEEE Dual Logo International Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEC/IEEE Dual Logo International Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard.

Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

In publishing and making this document available, the IEC and IEEE are not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Neither the IEC nor IEEE is undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEC/IEEE Dual Logo International Standards or IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations – Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments for revision of IEC/IEEE Dual Logo International Standards are welcome from any interested party, regardless of membership affiliation with the IEC or IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA and/or General Secretary, IEC, 3, rue de Varembé, PO Box 131, 1211 Geneva 20, Switzerland.

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

NOTE – Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Published by IEC under licence from IEEE. © 2005 IEEE. All rights reserved.

IEEE Standard for Extensions to Standard Test Interface Language (STIL) (IEEE Std 1450™-1999) for Semiconductor Design Environments

Sponsor

Test Technology Standards Committee
of the
IEEE Computer Society

Approved 9 June 2005

IEEE-SA Standards Board

Abstract: Standard Test Interface Language (STIL) provides an interface between digital test generation tools and test equipment. Extensions to the test interface language (contained in this standard) are defined that (1) facilitate the use of the language in the design environment and (2) facilitate the use of the language for large designs encompassing subdesigns with reusable patterns.
Keywords: advanced scan architecture, core, environment, fail feedback, lockstep, parallel patterns, parameterized data, pattern tiling, pragma, signal variable, system on chip (SoC), test protocol

Published by IEC under licence from IEEE. © 2005 IEEE. All rights reserved.

IEEE Introduction

The Standard Test Interface Language (STIL) was initially developed by an ad hoc consortium of automatic test equipment vendors (ATE), electronic design automation vendors (EDA), and integrated circuit (IC) manufacturers to address the lack of a common solution for transferring digital test data from the generation environment to the test equipment.

The scope of the initial STIL standard was limited to satisfy the basic needs of pattern definition. Additional capabilities are developed as separate projects resulting in separate (dot) extensions to the initial STIL standard. The scope of this extension is defined in 1.1 and is primarily to address design needs.

Whereas the initial STIL standard was developed by reviewing many languages already in existence in the industry, this standard has been developed by inventing new capabilities in support of new device designs. The new language constructs have been added such that they do not alter in any way the initial definition of STIL, yet are syntactically compatible with the initial STIL language.

Much of the work to develop and validate these extensions has been done by prototyping on the part of the contributing companies.

Notice to users

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: http://standards.ieee.org/reading/ieee/updates/errata/index.html. Users are encouraged to check this URL for errata periodically.

Interpretations

Current interpretations can be accessed at the following URL: http://standards.ieee.org/reading/ieee/interp/index.html.

Patents

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents or patent applications for which a license may be required to implement an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Published by IEC under licence from IEEE. © 2005 IEEE. All rights reserved.

STANDARD FOR EXTENSIONS TO STANDARD TEST INTERFACE LANGUAGE (STIL) (IEEE Std 1450™-1999) FOR SEMICONDUCTOR DESIGN ENVIRONMENTS

1. Overview

STIL is an evolving standard being developed in support of various needs for interfacing between test generation tools and test equipment. IEEE Std 1450-1999 (STIL.0) [B3][1] forms the basis for this evolution. New "dot" standards (like this one) are being developed to address specific needs beyond STIL.0.

This (STIL.1) standard addresses design-related aspects of digital test data. This standard can also be viewed as the addition of advanced features to the STIL.0 baseline to allow for the usage of STIL in more complex applications, while leaving the basic standard unchanged as a vehicle for transmitting basic test data. The following is a brief overview of the new features in STIL.1 to support advanced applications such as (1) embedded cores,[2] (2) families of test patterns, (3) mapping to automated test equipment (ATE) systems,[3] (4) mapping to simulation, and (5) devices with advanced design for test (DFT). Please see Annex O for a list of specific statements for each of these features.

Environment mapping: Data for a device exist in many forms and in many other software environments. Examples include (1) simulation environment, (2) static analysis environment, (3) specific ATE system environment. The STIL Environment block is a new mechanism to relate STIL data to these other environments. No assumptions, expectations, or limitations are imposed on the other environments. It is just a way of relating one to the other.

Parameterized data: Much of STIL data are declarative in nature (i.e., it defines various static attributes of a device or pattern set). The addition of constant declarations, IntegerConstant and WFCConstant, allows a data set to be created that applies to a family of devices.

Complex test protocol definition: Test protocol definitions are usually contained in STIL procedures or MacroDefs and are used to specify the application of a series of data values to a device. STIL.0 supports scan chain data passing and simple WaveformCharacter (WFC) data passing via the # and % characters. STIL.1 enhances this capability by allowing the use of data substitution from SignalVariables and integer-

[1]The numbers in brackets correspond to those of the bibliography in Annex P.

[2]This standard contains syntax in support of embedded cores. See IEEE Std 1450.6™-2005 (Core Test Language) [B5] for the complete specification.

[3]This standard contains syntax in support of ATE systems. See IEEE P1450.3™ (Test Resource Constraints) [B4] for the complete specification.

Published by IEC under licence from IEEE. © 2005 IEEE. All rights reserved.

Complex test protocol definition: Test protocol definitions are usually contained in STIL procedures or MacroDefs and are used to specify the application of a series of data values to a device. STIL.0 supports scan chain data passing and simple WaveformCharacter (WFC) data passing via the # and % characters. STIL.1 enhances this capability by allowing the use of data substitution from SignalVariables and integer-expressions. STIL.1 also enhances the functionality of Loops and Vectors and adds If/While decisions on pattern statements. These capabilities are needed for BIST, embedded cores, and various test access mechanisms.

Advanced scan architecture: Advanced DFT techniques require additional capabilities beyond what is defined in STIL.0, which includes multistate scan cells, reconfigurable scan-chains, and scan-chain indexing.

Run-time pattern decisions: The If, Else, While, and LoopData are new STIL.1 constructs that have been added for specification of pattern activity. These statements are needed in the specification of patterns to be run in the simulation environment. Although there is no standardization among ATE systems on run-time instructions for pattern execution, it is anticipated that restricted versions of these statements will be incorporated into ATE test patterns.

Pattern burst options: New variations on the PatternBurst have been added to allow for patterns running in parallel, patterns running in LockStep, and patterns that can be reordered. For parallel pattern execution, the specification for how the patterns fit together can be specified with the Fixed and Extend statements.

Enhanced user extensibility: The UserKeyword extensibility defined in STIL.0 has been extended to allow keywords to be defined on a per-block-type basis.

Signal relationships: Additional syntax is provided to allow the specification of relationships between signals. This process is preformed via \m to map WFCs to another WFC, \j to join WFCs, Extend to define behavior of signals beyond the bounds of a given pattern, and Fixed to restrict any further changes to signals within a pattern.

Fail feedback: Three new features are added to facilitate the processing of failure data from an ATE system back to design tools. The first is the X or cross-reference statement that allows the specification of where in a pattern/vector sequence a failure occurs. The second is the FailFeedback block for reporting fails. The third is the S/s timing event that allows for the specification of a data capture protocol for the purpose of capturing bulk fail data for processing.

1.1 Scope

Structures are defined in STIL to support usage as semiconductor simulation stimulus, including (1) mapping signal names to equivalent design references, (2) interface between scan and built-in self test (BIST) and the logic simulation, (3) data types to represent unresolved states in a pattern, (4) parallel or asynchronous pattern execution on different design blocks, and (5) expression-based conditional execution of pattern constructs.

Structures are defined in STIL to support the definition of test patterns for sub-blocks of a design[4] (i.e., embedded cores) such that these tests can be incorporated into a complete higher level device test.

Structures are defined in STIL to relate fail information from device testing environments back to original stimulus and design data elements.

---

[4]Syntax in this document that is used in the definition of patterns for sub-blocks is summarized in Annex O.

Published by IEC under licence from IEEE. © 2005 IEEE. All rights reserved.