Introduction Document to STILDirector™

Toshiba Microelectronics
STILDirector Support Group
Contents

1. Overview and Test Environment for STIL

2. Architecture/Features and Development/Support Plan of STILDirector™

3. Environment for STILDirector™

Appendix:

Transition to STIL in Toshiba
STILDirector™ (Supplemental remarks)
Introduction to Pattern Edit Function
Introduction of STILPlanner and tester I/F
Introduction of STILBuilder™
Introduction to Fault Analysis Interface
Introduction to the STIL Learning System (STIL e-Learning)
1. Overview and Test Environment for STIL
What is STIL?

Standard Test Interface Language

- **STIL** is a **test data description language** approved by IEEE as **Std 1450.0** in **1999**. STIL is pronounced as the word “style” in English. (Functional expansion is in progress)
- **STIL** is a **test description language** that can be used on all processes of semiconductor testing such as design, simulation, test generation, testing and failure analysis.

### IEEE STIL Working Group Activity Status

- **1450.0 Test Pattern Specification** (Standardized in Mar 1999)
- **1450.1 Semiconductor Design Environment** (Standardized in Jun 2005)
- **1450.2 DC Level Specification** (Standardized in Dec 2002)
- **1450.3 Target Tester Specification** (Standardized in Sep 2007)
- **P1450.4 Test Flow Specification** (Planned to ballot in 2014~15)
- **P1450.5 Test Method Specification** (Planned to start after the standardization of P.4)
- **1450.6 Core Test Language Support** (Standardized in Nov 2005)
- **1450.6.1 On-Chip Scan Compression** (Standardized in May 2009)
- **1450.6.2 Memory Model in Core Test Language** (Standardized in 2014)
- **P1450.7 Analog and Mixed signal Specification** Has started operation
- **P1450.8 Design Information Test Flow Specification** Has started operation

Test program pattern information
Test program tester control information
Benefits from Using STIL

Problems in using a proprietary language

- The extent of utilization is limited for a language. To fully utilize the tester functions, there is a consistent need for language extensions and functional enhancement.
- Need to acquire languages as an data interface with EDA or ATE vendors, semiconductor manufacturers, and customers.

Benefits of STIL

- Communication with a common language
- Use as a common test interface (easy to reuse test assets)
- Reduce load of data interface at testing

For the common knowledge and usage of STIL, refer to “STIL Usage Guide” issued by STARC SSTAG.
**Purpose of Building a STIL Interface Test Environment**

- Eliminates data correction/conversion between design tools or devices
- Reduces TAT by using a common test language

- STIL is commonly utilized between processes different companies are involved to share data
- Referring STARC STIL Usage Guide is recommended to share understand and usage

**STIL interface test environment**

![Diagram of STIL interface test environment](image)
1-2. STIL Test Environment

(By building a test environment based on STIL....)

- Handles data relevant to a test by STIL
  ⇒ Data conversion/correction in each phase is omitted
- Problematic data is not passed and received by TRC (Tester Restriction Checker)
  ⇒ Prevents iteration
- Enables to reuse test asset easily by keeping test information (test program/pattern) in STIL
- Enables to introduce optimized tools/test devices easily
  ⇒ Development efficiency increases by using cost effective tools/devices promptly.

*Estimated effect on our motif products using STILDirector in STIL test environment
1-2. STIL Test Environment

Test Development Flow with STIL Interface

<<Design phase>>
- STIL1450.1 (fail info)
- STIL1450.2
- STIL1450.3 (TRC)
- STIL P1450.4
- STIL P1450.5

<<Test phase>>
- Tester control information
- Tester conversion (main, pattern)
- Test spec
- Test flow
- Test methods, etc.

<<Analysis phase>>
- Fault Diagnosis
- STIL1450.1 (fail info)
- STDF
- Test data log

STIL
- Tester control info file
- Other files
- Tool/device
- (can select one best suited)

*Excerpted from STARC’s material

STIL1450.3 (TRC) before STIL1450.4/.5 standardization

STIL1450.4/.5 standardization

STDF

STIL

Test Pattern

Test Pattern

Tester Conversion
(main, pattern)

Conventional tester or new STIL driven tester

From Design

Before STIL1450.4/.5 standardization

Tester test

After STIL1450.4/.5 standardization

Test Program

Tester Pattern

Tester control information

Test spec

Test flow

Test methods, etc.

Pin info

Board connection info

Fault Diagnosis

Copyright © TOSHIBA MICROELECTRONICS CORPORATION 2002—2014, All Rights Reserved.
Building a STIL Interface Test Environment

STIL Standardization Status
- 1450.0 Test Pattern Specification
  - Standardized (1999)
- 1450.1 Semiconductor Design Environment
  - Standardized (2005)
- 1450.2 DC Level Specification
  - Standardized (2002)
- 1450.3 Target Tester Specification
  - Standardized (2007)
- 1450.4 Test Flow Specification (est.)
- 1450.5 Test Method Specification (plan)
- 1450.6 Core Test Language Support (CTL)
  - Standardized (2005)
  - 1450.6.1 On-Chip Scan Compression
    - Standardized (2009)
  - 1450.6.2 Memory Model in Core Test Language
    - Standardized (2014)
- 1450.7 Analog and Mixed signal Specification
- 1450.8 Design Information Test Flow Specification

Toshiba’s move
- Started STILDirector sales (2002/SH-)
- STILDirector Update: e-Learning for the STIL language (2003/FH-)

STARC Activities of STIL-based Semiconductor Test Action Group
- Activity: Creating STIL Usage Guide
- Promoting STIL test programming
- Assurance of STIL data compatibility

Activities of STC STIL WG Japan
- STC STIL WG is restarted
- Building consensus on STIL Usage
- Promoting STIL standardization and support

STIL Usage Guide (Japanese and English) issued
- 1450.0 (Rev5), 1450.1 (Rev4), 1450.6 (Rev4)

STIL Support Tool Rating

STIL Testprogram Generator

STIL Testprogram Generator

STIL Usage Guide (Japanese and English) issued
- 1450.0 (Rev6), 1450.1 (Rev6), 1450.6 (Rev6)

Examination of Test program using STIL

STIL Testprogram Guidebook

Asuka 2 ΦII

New STARC SSTAG

SEMI-CAST STIL WG Japan
2. Architecture/ Features and Development/Support Plan of STILDirector™
What is STILDirector™?

A generalized STIL compliant plug-in system parted from Toshiba Design Kit

STILDirector™

Toshiba Design kit

STIL

Simulator

Tester

Allows you to build a STIL based test environment quickly and surely at a reasonable cost
“plug-in” example of STILDirector™

Interface for Toshiba design environment + STILDirector™

Toshiba STIL DK

Interface for User’s design environment + STILDirector™

User’s STIL DK
2-1. Details of STILDirectors

STILDirector™ Plug-in

- Plug-in for various requirements
- Commands are provided on a program basis and can be built in
- Allows customizing to your environment by using an access interface of the STIL database

Build a STIL environment quickly, surely at a reasonable cost

To build a STIL interface test environment, STILDirector™ can be used as a Standard Plug-in System
Building a test environment based on STIL as a common language, you can shorten TAT from Design to Test and Failure Analysis!

**System Solution**

**2-2. Architecture and Features**

**STIL Data**

**Test Data Description**

**STILDirector™**

- STIL parser
- STIL generator
- Tester rules check
- Include file extraction
- Making simulation stimulus
- DC measurement address extraction
- Simulation result analysis, Cyclize
- Test program generation
- Pin multiplex/pattern multiplex conversion
- Test program generation
- Bit conversion
- Tester/Reversed Tester interface
- Fault simulation/Fault Analysis interface
- Test pattern expansion
- Masking
- Access interface
2-2. Architecture and Features

Edit-rich functions provided:

1. STILDirector Basic Pkg
   - Pattern bit conversion
   - Test pattern expansion
   - Pin multiplex/pattern multiples Conversion
   - Include file extraction
   - Input data
   - Simulator
   - Output result
   - STIL comparing

2. Tester I/F
   - Pattern bit conversion
   - Test pattern expansion
   - Pin multiplex/pattern multiples Conversion
   - Include file extraction
   - I/F to simulator
     - Simulator stimulus generation
     - Result analysis/cyclize
     - DC measurement address extraction

3. STIL Planner
   - I/F to application
     - Access interface
   -Tester I/F
     - Tester/Reverse tester interface
     - Generation of tester control
     - Generation of Test program

4. TestPro Generator
   - Pattern bit conversion
   - Test pattern expansion
   - Pin multiplex/pattern multiples Conversion
   - Include file extraction
   - I/F to fault analysis
     - I/F to fault simulator/fault analysis
     - Fault simulator/Fault analysis tool

STIL edit function

STILBuilder on Windows
- STIL Viewer & STIL Editor

Pattern bit conversion
- Test pattern edit
- Test pattern compression
- Test pattern merge
- Pattern insertion

Tester’s constraint check

STIL parser
- STIL generator

STIL DataBase

Option

I/F to STIL data

Customer system

Tester

I/F to STIL data

I/F to application

I/F to tester

Include file extraction

Option

Test pattern edit

Tester/Reverse tester interface

Generation of tester control

Generation of Test program

Fault simulator/Fault analysis tool
2-2. Architecture and Features

Platform for STILDirector™

◆ Platforms
- EWS (SPARC/ UltraSPARC) Sun Solaris 9, 10 (64bit)
- PC (Intel x86) Windows7 (32bit)
- PC (AMD Opteron) RedHat Enterprise Linux v.4/5/6 (64bit)

*The following platforms will be not supported from 2010B ver.
- EWS (SPARC/ UltraSPARC) Sun Solaris 8 (32bit/64bit), 9 (32bit), 10 (32bit)
- RedHat Enterprise Linux v.3 (32bit)

◆ Supported Simulators
- Verilog-XL V11.1, V12.1, V12.2
- NC-Verilog V11.1, V12.1, V12.2
- ModelSim V6.4, V6.5, V6.6
- VCS V2011.12, V2012.09-1, V2013.06
- NC-SIM V11.1, V12.1, V12.2

*Allowed combinations of a simulator and a platform from various vendors are based on support information released by those vendors.
Features of STILDirector™

1. Interface to the STIL data
   - Input and output the STIL data
   - Check violations of the STIL language specification

2. Interface to simulator
   - Generate input for various simulators
   - Input result files of various simulator
   - Analyze simulation results
   - Generate STIL files by automatically extracting the expected values from simulation results
   - Automatically extract DC measurement address

3. Interface to tester
   - Generates test program (test pattern, main program)
   - Check whether pattern description can be used in an objective tester before execution of tester
   - Cyclize simulation results

4. Interface to User’s system
   - Open system allows users to plug-in to their environment
   - Allow customization by using access interface of database

5. Other features
   - STILDirector™ works on EWS and PC
2-2. Architecture and Features

Interface to ATPG

STIL output by the following ATPG tools can be used

*The STIL output might not be used depending on versions or options of the ATPG tools

- Synopsys TetraMAX
- Mentor FastScan
- Cadence EncounterTest
An interface to the following Fault Simulators is allowed:
- Cadence VeriFault
- Syntest TurboFault
- Synopsys TetraMAX IDDq

Allows output of test patterns for the following tester:
- ADVANTEST (T33xx, T6xxx, T2000)
* We plan to support J750 from the 2014A version

Test patterns for testers not supported by STILDirector™ can be easily developed by using access interface.
3. Environment for STILDirector™
3-1. System of Sales

STILDirector™ Various Packages

- **STILDirector™ Time License Sales (with maintenance service for one year)**
  - Basic package sales (for Solaris/Linux platform)
    - STIL parser/STIL generator/Simulator input file generation/Results analysis and cyclization/DC address extraction/Tester rules checking/STIL data editing
  - Enhanced package sales
    1. Database Access Interface (specific platform)
    2. Tester Interface (specific platforms)
    3. Fault Simulator Interface (specific platforms)
- **STIL compliant model package (specific platforms)**
- **STILPlanner (specific platforms)**
- **Test generator (test program generation) (specific platforms)**

- **Customize Service**
  - Consultation to implementation, operational management

- **Other Services**
  - Training
  - e-Learning
3-1. System of Sales

Structure of STILDirector Sales Packages

① STILDirector Basic Pkg
- I/F to STIL data
- STIL parser
- STIL generator
- Test pattern edit
- Test pattern compression
- Test pattern merge
- Pattern insertion
- Tester's constraint check
- Include file extraction

② STILDirector Enhancement Pkg
- I/F to simulator
  - Simulator stimulus generation
  - Result analysis/cyclize
  - DC measurement address extraction
- STIL comparing
- STIL Planner
- Fault simulator/Fault analysis tool
- Input data
- Simulator
- Output result

③ STIL Planner
- Generation of tester control
- Generation of Test program

④ TestPro Generator
- Option
- Option
- STIL support Model Pkg
- STIL Builder on Windows
- STIL Viewer & STIL Editor

1. System of Sales

Pattern bit conversion
Test pattern expansion
Pin multiplex/pattern multiples Conversion
Include file extraction

STIL

STIL DataBase

Input data

Tester’s constraint check

STIL parser

STIL generator

Test pattern edit

Test pattern compression

Test pattern merge

Pattern insertion

Tester’s constraint check

STIL support Model Pkg

STIL Viewer & STIL Editor

STILBuilder on Windows

I/F to STIL data

I/F to simulator

I/F to tester

I/F to application

Access interface

Customer system

Tester

Tester/Reverse tester interface

Model Pkg

Enhancement Pkg

Basic Pkg

System of Sales

Pattern bit conversion
Test pattern expansion
Pin multiplex/pattern multiples Conversion
Include file extraction

STIL

STIL DataBase

Input data

Simulator

Output result

Fault simulator/Fault analysis tool

Input data

Fault simulator/Fault analysis tool

STIL comparing

① STILDirector Basic Pkg

② STILDirector Enhancement Pkg

③ STIL Planner

④ TestPro Generator

Copyright © TOSHIBA MICROELECTRONICS CORPORATION 2002—2014, All Rights Reserved.
3-1. System of Sales

STILDirector™ Package Deal Models for Test Program Generation

(3) Test Program Generation Package
(2) Test Program Generation Basic Part
(1) Test Pattern Generation Package

➀ STILDirector™ Basic Package
- simulation stimulus data generated by STIL data
- generates STIL data from the simulation output results
- extracts the DC measurement address
- pattern edit function
- generates test patterns for testers
- API interface to the test program generation system
- test programs or test programs by test manufacturer
- creates input data for generation system

For users who want up to test program generation from STIL!

② Tester I/F

③ STIL Planner™ Reg Pkg

④ Test Generator

API

ADVAN
T6XXX
ADVAN
T33XX

Site seer
Test Program
Templ

SIF

Test Pattern

Tester

Sim Bench
STIL

STIL

Test Pattern

Tester

T6XXX

T33XX

(1) Test Pattern Generation Package

(2) Test Program Generation Basic Part

(3) Test Program Generation Package
<table>
<thead>
<tr>
<th>Type</th>
<th>Software Structures</th>
<th>Comments</th>
</tr>
</thead>
</table>
| I    | STIL Parser  
Tester Rules Check | Used to make a tester be compliant with STIL. Used to convert STIL data to customer’s system data. |
|      | Access Interface (Read) | |
|      | * Specific for development/operation of applications read from STIL data. | |
| II   | STIL Generator  
Tester Rules Check | Use to convert tuning result from a tester to STIL data. Use to convert data in a customer’s system to STIL. |
|      | Access Interface (Write) | |
|      | * Specific for development/operation of applications to write to STIL data. | |
| III  | STIL Parser  
Tester Rules Check | Although applications generated from each Read and Write of the model access I/F, which is a combination of Type I and Type II can be executed simultaneously, Read and Write in a single application cannot be operated simultaneously. |
|      | Access Interface (Read) | |
|      | STIL Generator  
Tester Rules Check | |
|      | Access Interface (Write) | |
|      | * Read and Write of the Access I/F cannot be executed simultaneously. | |
| IV   | STIL Parser  
Tester Rules Check  
STIL Generator | The access interface is same as the one in the enhanced package. Used to make a tester be compliant with STIL. |
|      | Access Interface (Read/Write) | |
|      | * Read and Write of the Access I/F can be executed simultaneously. | |

You can easily read and write the STIL data.
3-2. Providing Service

**Services provided for STILDirector™**

- Updated versions (free during a valid license period)
  - Major release: 1~2 times a year
  - Minor release: as needed

- Customized service by contract (paid service)
  - We provide support to “plug-in” STILDirector to your environment
  - We provide support to develop specific applications using an access interface

- Customized service examples
  - Developing applications to convert the STIL data to dedicated test data for various makers
  - Developing applications to convert the STIL data to test programs in tester environments of various makers
3-2. Providing Service

Services provided for STILDirector™

■ Contact information

Web: http://www.tosmec-web.toshiba.co.jp/stildirector/
E-mail: STILDirector@tosmec.toshiba.co.jp
TEL: Toshiba Microelectronics Corporation
STILDirector Support Center: +81-44-548-2127
  Open: Mon-Fri 10:00～17:00
Closed: Sat, Sun, national and corporate holidays
*For our corporate holidays,
  please check the Website
FAX: +81-44-548-8986

STILDirector Website
3-2. Providing Service

Services provided for STILDirector™

Distributors: STILDirector™ is available from the following distributor

ATE Service Corp. (Technical support is offered by ATE Service)

■ Contact

TEL: 042-795-8602
E-Mail: sd_support@ate.co.jp
POC: Mr. Kato

■ Company Profile: ATE Service Corp.
Corporate Headquarters: 1887-2 Tsuruma Machida-shi, Tokyo, Japan
President: Kenji Suga
Founded: Feb 1st, 1983
Capital: 47,000,000 yen

*Excerpted from the company’s website (as of Sep 2014)
Appendix
Transition to STIL in Toshiba
Transition to STIL in Toshiba

Provides a tool to convert the conventional language to STIL

TSTL2

STIL

TSTL2STIL

STIL

STIL Director

Tester

Simulator

The language conversion tool, TSTL2STIL, is included for smooth transition
Initially TSTL2 D.K., STIL D.K. and the tool (TSTL2STIL) to convert TSTL2 to STIL are provided as a complete set of D.K.

With the progress of transition to STIL, the D.K. is changed to contain only STIL D.K. and conversion tool, TSTL2STIL.
STILDirector™ (Supplemental remarks)
Flow of STILDirector™

1. System simulation
2. Generate test data
3. Generate input pattern for logic verification
4. Logic verification
5. Analyze simulation results
6. Extract simulation expected values
7. Convert patterns for tester
8. SIM2STIL
9. STIL2SIM
10. Execute Simulation
11. SIM2STIL
12. VTEST
13. STILDirector™

- Extract test patterns at strobe points from RTL simulation results to create STIL test data.
- Create input data for various simulators (logic verification) from STIL test data.
- Create STIL test data from simulation result of logic verification.
- Generates test pattern from logically-verified STIL test data
Structure and Flow of STILDirector™

STIL DK

STIL

Simulator I/F
(T2SIM, SIM2T)

STIL parser
(S2T)

STIL DB

STIL generator
(T2S)

Tester Rule
checker (TRC)

Bit conversion
(SCNV)

Test pattern expansion
(PATEXP)

Pin multiplex / pattern
multiplex conversion
(STILMUX)

Masking
(PATCHG)

Include File Extraction
(STILDIV)

Tester
I/F

Fault simulator Interface

Enhancement Package

Rule description

① STIL2SIM
② SIM2STIL
Commands in STILDirector™

Main Commands

(1) **STIL2SIM**
Composed of three commands, S2T, TRC and T2SIM. Checks the description of STIL test data file and changes it to signals to be applied during simulation. Also, checks violation against tester restrictions.

(2) **SIM2STIL**
Composed of three commands, T2S, TRC and SIM2T. Converts the value change of I/O pins during a simulation into a STIL test data file with expected values. Analyzes and cyclizes the results. Extracts DC measurement address. Also, checks for violations against tester restrictions.

**I/F to STIL data**
- **S2T**: Checks description of STIL test data file and converts it into a STIL database file.
- **T2S**: Converts a STIL database file into a STIL test data file in the STIL format.

**I/F to simulator**
- **T2SIM**: Generates a simulator input file from a STIL database file.
- **SIM2T**: Converts simulation results file to a STIL database file. Analyzes and cyclizes results, and extracts DC measurement address.

**I/F to tester**
- **TRC**: Checks for violations against tester restrictions.
Logic verification based on tester behavior

Simulation verification closer to the actual operation can be conducted

(a) Specification of the DNRZ waveform Initial value
(During creation of input patterns for logical verification)

The initial state varies according to the load order of test patterns.

The initial value based on the initial state at the test can be specified optionally.
init=[X|0|1]

(b) Specification of the NRZ waveform status value after I/O change
(during creation of input patterns for logical verification)

The status value varies according to the tester model.

The status value immediately after I/C change can be specified according to the tester model.

setoi=prev_in …Value is determined by the input pattern value immediately before the output state
prev_out …Value is determined by the expected output value
X …Delay section is performed as X
(c) Strobe margin checking (during analysis of simulation results)

The stable regions before and after the output strobe waveform are displayed as a map

```
PIN_NAME:IO10

---******************************************************************************
STB_TIME = 30, 10
22.000 + 60.000
LEFT:8.000 RIGHT:100.000
(CYCLE_START_TIME:0.000) (CYCLE_START_TIME:0.000)
```

(d) Conflict/floating checking (during analysis of simulation results)

The regions where a conflict/floating for a bidirectional pin occurs is displayed as a map

```
PIN_NAME:IO11
XXXXX---------------------------------------------

LEFT:0.000 RIGHT:10.000
(CYCLE_START_TIME:0.000) (CYCLE_START_TIME:0.000)
```
Introduction of Pattern Edit Function
STIL edit function

Pattern bit conversion
Test pattern expansion
Pin multiplex/pattern multiples Conversion
Include file extraction

STIL parser
STIL generator
Tester’s constraint check

I/F to STIL data

STIL DataBase

Simulator stimulus generation
Result analysis/cyclize
DC measurement address extraction

I/F to simulator

STIL comparing

STILDirector edit commands

1. Pattern bit conversion (SCNV)
2. Test pattern development tool (PATEXP)
3. Pin multiplex/pattern multiplex conversion (STILMUX)
4. Include file extraction (STILDIV)
5. Test pattern edit (PATCHG)
6. Test pattern compression (PATCOMP)
7. Test pattern merge (STILMERGE)
8. STIL pattern insertion (STILINSERT)
9. STIL comparing (STILCMP)
STILDirector Pattern Edit Functions

Edit-rich functions

To support all processes, such as searching of the pattern to match, data indication, edit, and review of equivalence

◆ Using multiple conditional expression combining “()”, “&&”, and “||”, in addition to single conditional expression.
◆ Specifies the edit range of pattern.
◆ Enables to describe wildcard for pin name and pin group name in conditional expression.
  *: Zero or more arbitrary letter(s)
  ?: One arbitrary letter
◆ Inserts label/comment/STIL pattern to the pattern which matches the conditional expression.
◆ Outputs the report of the pattern before/after the edit.
◆ Edits the expected value, etc. in accordance with the compare information file output with SIM2STIL (also supports PLS)
◆ Converts the tester double speed mode (pin multiplex/pattern multiplex) pattern into the flat pattern.
◆ Splits STIL data for include, and merges the STIL data split into include files.
◆ Enables loop (multiple vectors) compression/procedure compression of STIL data.
◆ Merges the pattern in the direction of pin/address of STIL data.
◆ Reviews equivalence of two STIL data.
Introduction of STILPlanner and Tester I/F
Tester I/F Enhanced Package and Test Program Generation Tools

Generates test programs and test patterns for a LSI tester using STIL data from STILDirector as input.

**Tester I/F enhanced package**

- Generates a test pattern source for a tester
- Supported testers: Advantest T6500/6600 series
  - T33xx series, T2000

  Supported platforms: Solaris 9/10, RedHat Enterprise Linux 4/5/6

**STILPlanner**

- Generates a test program source for a tester utilizing test pattern information generated by the above tester I/F
  - (Note: T6500/6600 series generates SIF)

  Supported testers: Advantest T6500/6600 series
  - T33xx series, (T2000)

  Supported platform: Solaris 9/10
Flows for Generating Tester Pattern and Test Program

- STIL
  - STIL parser processing
  - STIL DataBase
    - Enhanced SDB
      - Pattern output for tester
        - Test pattern
        - Pattern information
  - Pin information file
  - Spec information file
  - Test order file
  - Template
  - Generating tester control information
    - SIF
  - Generating Test program
  - Siteseeer @ ADVANTEST
  - STILPlanner
  - Tester I/F
    - Pattern information
    - Test program
    - Test pattern

Introduction of STILBuilder™
STILBuilder™

• Using this STIL editor tool, which operates on Windows, you can create and edit a STIL file, search information, and check description. STILBuilder™ is compliant with STIL1450.0, 1450.2, 1450.1 (partially).

- Enables to edit data as STIL data viewer, in addition to the tabular display and waveform indication
- Enables to use as the STILDirector user interface (on PC)
Features of "STILBuilder™"

Useful features of STILBuilder™ include:

- Functions of displaying patterns in waveform and tabular formats
- Function of searching for patterns corresponding to the specified time or scan flip-flop
- Function of supporting GZIP files
- Pattern masking
- Function of editing multiple STIL files by switching tabs
- Capability to create STIL 1450.0/STIL 1450.2-compliant STIL
- Functions of outputting to Include files, and expanding easily from Include files
- Capability to jump to one of blocks which make up a STIL file, and to a specified line
- Easy insertion of statements by using the dedicated statement TreeView
- Capability to open STIL files easily from the file view screen
- Functions of coloring keywords and automatically inserting linefeed codes
- Capability to use functions of STILDirector™ as external commands
### Displaying patterns in waveform and tabular formats

**Pattern Display in Tabular Format**

Patterns edited in a waveform image can be output in STIL description.

Complicated pattern information by STIL description can be displayed and edited easily and plainly in a tabular format.

<table>
<thead>
<tr>
<th>Col No.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern Block</td>
<td>PATT1</td>
<td>PATT1</td>
<td>PATT1</td>
<td>PATT1</td>
<td>PATT1</td>
<td>PATT1</td>
<td></td>
</tr>
<tr>
<td>Cycle</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>Timing Set</td>
<td>T51</td>
<td>T51</td>
<td>T51</td>
<td>T51</td>
<td>T51</td>
<td>T51</td>
<td></td>
</tr>
<tr>
<td>Cycle Time</td>
<td>0ns</td>
<td>100ns</td>
<td>200ns</td>
<td>300ns</td>
<td>400ns</td>
<td>500ns</td>
<td>600ns</td>
</tr>
<tr>
<td>Pin Name</td>
<td>PIN1</td>
<td>PIN2</td>
<td>PIN3</td>
<td>PIN4</td>
<td>PIN5</td>
<td>PIN6</td>
<td>PIN7</td>
</tr>
<tr>
<td>Loop Count</td>
<td>x1</td>
<td>x1</td>
<td>x1</td>
<td>x1</td>
<td>x1</td>
<td>x1</td>
<td>x1</td>
</tr>
</tbody>
</table>

- PIN1
- PIN2
- PIN3
- PIN4
- PIN5
- PIN6
- PIN7

Pattern Display in Waveform Image
Searching for patterns corresponding to a specified time or scan flip-flop

Function of searching for a STIL pattern based on a specified time (e.g., time when an expected value discrepancy in simulation occurs)

Function of searching among shift patterns for a STIL pattern that corresponds to the selected scan flip-flop
Introduction of Failure Analysis Interface
Building a Failure Analysis Interface for STIL

Operational Flow of Failure Analysis

Connection data → ATPG

TetraMAX FastScan TestBench

Possible Failure Node → Failure Diagnosis

To Failure Diagnosis Device

STIL → Files for Failure Analysis Tool

Tester → Tester Fail Log

Newly Developed

Failure Analysis Interface
- common fail information generation portion
- pattern information generation portion
- individual fail information generation portion
Failure Analysis Interface

Tester Fail Log

- Pattern name
- Sample No.
- DUT No.
- Failed address
- Failed pin name
- Expected value
- Actual measured value

STIL

- Coordinating ATPG pattern addresses for all primary patterns with expanded addresses from the top
- Coordinating ATPG pattern addresses for the top pattern of each scan pattern with expanded addresses from the top

Common Failure File

Converting Tool 1

Pattern Information File

Converting Tool 2

Files for Failure Analysis Tool

Converting Tool 3

- Failed pattern address
- Failed pin name
- Failed flip-flop location
- Actual measured value, et al.

Building a Failure Analysis Interface for STIL (contd.)
Introduction to the STIL Learning System (STIL e-Learning)
“STIL Learning System” (STIL e-Learning)

*STIL language e-Learning course has started (since Apr 7, 2003)

- User Management
- Progress Management
- Comprehension Management

Top page of each chapter

Details

Exercises

User
How to use STIL e-Learning

http://www.tosmec-web.toshiba.co.jp/stildirector/

STIL Learning System

User

Demo course

Apply for Enrollment

TOSHIBA

Obtain ID

User Registration

Registration not required

Registration completed

Start course

Start course

In the demo course, users can learn about STIL and how to operate web-based courses.

STIL e-Learning Top Page:
https://www.tosmec-web.toshiba.co.jp:8443/stildirector/training/e-learn/index.html

Copyright © TOSHIBA MICROELECTRONICS CORPORATION 2002—2014, All Rights Reserved.
User Support for STIL e-Learning

- **History**
  Shows results of Comprehension Tests

- **User Info**
  Check user information

- **Contact Us**
  Questions and inquiries to course instructors

- **Page Save**
  Exit from program and enter back to saved page

- **Progress**
  Color-coded scheme indicates progress
  - Finished chapter
  - Chapter in progress
  - Not yet studied

- **License Period**
  Indicates license period and days remaining

**Program Description**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Time</td>
<td>5 hours (300 minutes)</td>
</tr>
<tr>
<td>Objective</td>
<td>To accumulate knowledge about STIL that is required for the actual operation. Note that STIL introduced in this program is within the IEEE std 1450 D - 1999 specification range, and it is a description confined to the Toshiba supported STIL.</td>
</tr>
<tr>
<td>Required Knowledge</td>
<td>General knowledge of test data (e.g. R/WL, TSTL).</td>
</tr>
<tr>
<td>Audience</td>
<td>Those who completed STIL Language Course (STIL Introductory Course), or have equivalent knowledge (e.g. basic knowledge of what STIL is, what can be done with STIL, what STIL is used for).</td>
</tr>
</tbody>
</table>
| Program Contents   | Chapter 1: STIL Syntax [92 min]  
|                    | Chapter 2: STIL Declaration, and Comments and Ann Statements [29 min]  
|                    | Chapter 3: Defining Signal Names [29 min]  
|                    | Chapter 4: Defining Timing [48 min]  
|                    | Chapter 5: Defining Test Pattern [20 min]  
|                    | Chapter 6: Defining Data, Data Information [50 min]  
|                    | Chapter 7: Advanced Course Information [50 min]  
|                    | Final Test [50 min]  
|                    | License Expiration Date [31 Dec 2004]  
|                    | User Information [50 min]  
|                    | User Information [50 min]  
|                    | Page Save [50 min]  
|                    | Top [50 min]  
|                    | Top [50 min]  
|                    | Top [50 min]  
|                    | Top [50 min]  
|                    | Top [50 min]  
|                    | Top [50 min]  

Copyright © TOSHIBA MICROELECTRONICS CORPORATION 2002—2014, All Rights Reserved.
Courses
- “STIL1450.0 Basic Course” - in service
  To understand the STIL language’s basic specifications, learn basics of test pattern description, such as signals, timing, and scan
- “STIL1450.0 Advanced Course” - in service
  To learn advanced description of STIL, not covered by the Basic Course, such as timing definition by variables, event description and Hex value notation of patterns, and flexible timing definition by the inheritance of definition information or the reuse of partial definition
- “STIL1450.1 Basic Course” - in service
  To learn how to describe algorithmic test patterns using description function of scan patterns output from ATPG tools, and operators such as Boolean expressions
- “STIL1450.2 Course” - in service
  To understand the STIL description for DC level, learn voltage and current values as test control information for test programs

e-Learning Courses for STIL Extensions
- STIL1450.1 Advanced Course – in preparation
- STIL1450.3 Course – in preparation
- STIL1450.4 Course
- STIL1450.5 Course
- STIL1450.6 Course – in preparation
The information contained herein is subject to change without notice. The information contained herein is presented only as a guide for the applications of our products. No responsibility is assumed by TOSHIBA for any infringements of patents or other rights of the third parties which may result from its use. No license is grated by implication or otherwise under any patent rights of TOSHIBA MICROELECTRONICS CORPORATION or others.