Installing SiVL

Version Y-2006.12, December 2006

SYNOPSYS®

Copyright Notice and Proprietary Information

Right to Copy Documentation

The license agreement with Synopsys permits licensee to make copies of the documentation for its internal use only. Each copy shall include all copyrights, trademarks, service marks, and proprietary rights notices, if any. Licensee must assign sequential numbers to all copies. These copies shall contain the following legend on the cover page:

> "This document is duplicated with the permission of Synopsys, Inc., for the exclusive use of _____ and its employees. This is copy number _____."

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Registered Trademarks (®)

Synopsys, AMPS, Cadabra, CATS, CRITIC, CSim, Design Compiler, DesignPower, DesignWare, EPIC, Formality, HSIM, HSPICE, iN-Phase, in-Sync, Leda, MAST, ModelTools, NanoSim, OpenVera, PathMill, Photolynx, Physical Compiler, PrimeTime, SiVL, SNUG, SolvNet, System Compiler, TetraMAX, VCS, Vera, and YIELDirector are registered trademarks of Synopsys, Inc.

Trademarks (™)

AFGen, Apollo, Astro, Astro-Rail, Astro-Xtalk, Aurora, AvanWaves, Columbia, Columbia-CE, Cosmos, CosmosEnterprise, CosmosLE, CosmosScope, CosmosSE, DC Expert, DC Professional, DC Ultra, Design Analyzer, Design Vision, DesignerHDL, Direct Silicon Access, Discovery, Encore, Galaxy, HANEX, HDL Compiler, Hercules, Hierarchical Optimization Technology, HSIM^{plus}, HSPICE-Link, iN-Tandem, i-Virtual Stepper, Jupiter, Jupiter-DP, JupiterXT, JupiterXT-ASIC, Liberty, Libra-Passport, Library Compiler, Magellan, Mars, Mars-Xtalk, Milkyway, ModelSource, Module Compiler, Planet, Planet-PL, Polaris, Power Compiler, Raphael, Raphael-NES, Saturn, Scirocco, Scirocco-i, Star-RCXT, Star-SimXT, Taurus, TSUPREM-4, VCS Express, VCSi, VHDL Compiler, VirSim, and VMC are trademarks of Synopsys, Inc.

Service Marks (^{SM})

MAP-in, SVP Café, and TAP-in are service marks of Synopsys, Inc.

SystemC is a trademark of the Open SystemC Initiative and is used under license.
ARM and AMBA are registered trademarks of ARM Limited.
Saber is a registered trademark of SabreMark Limited Partnership and is used under license.
All other product or company names may be trademarks of their respective owners.

Printed in the U.S.A.

Installing SiVL, Y-2006.12

Installing SiVL

This document contains the following sections:

Note:

The installation instructions in this document are the most up-to-date available at the time of production. However, changes might have occurred. For the latest installation information, see the product release notes or documentation.

To run the SiVL tool, the Hercules and IC Workbench tools must be installed and running. See *Installing Synopsys Tools*, available at http://www.synopsys.com/install.

Media Availability and Supported Platforms

The SiVL tool is available by EST or on CD. Obtain the appropriate binary executable files based on the operating system you need. Table 1 shows the supported platforms for this release.

Table 1 Platforms and Keywords

Platform	Operating system	Synopsys platform keyword	Path keyword
AMD Opteron	Red Hat Enterprise Linux v3 (Taroon update 5)	amd64 (64-bit mode)	AMD.64

Table 1 Platforms and Keywords (Continued)

Platform	Operating system	Synopsys platform keyword	Path keyword
IA-32 (X86)	Red Hat Enterprise Linux v3[1] (Taroon update 5)	linux (32-bit mode)	IA.32
Itanium 2	Red Hat Enterprise Linux 2.1 (Derry)	linuxipf (64-bit mode)	IA.64
Sun SPARC	Solaris 9, 10[1]	sparc64 (64-bit mode)	SUN.64

1. Binary-compatible operating system. Note, however, that binary compatibility is not guaranteed.

Note:

The SiVL software is configured so that multiple platforms of this version can be installed in a single installation directory (*install_dir*).

Disk Space Requirements

The disk space requirement varies depending on the platform and the tool selected for installation. Table 2 shows the maximum space required for installing each product on a particular platform. (The total for all platforms is 498 megabytes, including 8 megabytes for the common file.)

Table 2 Disk Space Requirements for SiVL (in Megabytes)

Platform	Megabytes
amd64	64
linux (32-bit mode)	81
linuxipf (64-bit mode)	83
sparc64	92

Table 3 shows the maximum space required for installing Hercules on a particular platform.

Table 3 Disk Space Requirements for Hercules (in Megabytes)

Platform	Megabytes
amd64	91
linux (32-bit mode)	84
linuxipf (64-bit mode)	162
sparc64	92

Installing the Software

SiVL uses the Synopsys Installer tool, which allows you to use a graphical user interface (GUI) or a text script. For information about downloading Synopsys Installer and SiVL, see *Installing Synopsys Tools*.

To install SiVL by EST or from the CD, follow the procedures described in *Installing Synopsys Tools*.

Installing Synopsys Tools shows an example Synopsys media installation script for the synthesis tools. SiVL is installed in a similar manner.

SiVL is a stand-alone product and cannot be installed over an existing Synopsys product, including a prior version of SiVL. You must create a new directory for SiVL.

Setting Up the User Environment

To set up the user environment, you must specify the location of the executable file and set the license environment variable.

Specifying the Executable File Location

Add the SiVL directory containing the executable file to the `PATH` environment variable.

If you are using the C shell, add the following line to the .cshrc file:

```
set path=(SIVL_HOME/bin $path)
```

If you are using the Bourne, Korn, or Bash shell, add the following line to the .profile, .kshrc, or .bashrc file:

```
PATH=$SIVL_HOME/bin:$PATH
export PATH
```

where *SIVL_HOME* is the name of the SiVL installation directory.

Setting the LM_LICENSE_FILE or SNPSLMD_LICENSE_FILE Environment Variable

You must install the SCL software and define the `SNPSLMD_LICENSE_FILE`, `LM_LICENSE_FILE`, or another tool-specific variable before you can verify the SiVL installation.

For information about downloading and installing SCL and setting the license variable, see *Installing Synopsys Tools*.

Setting Up SiVL to Use Hercules and IC WorkBench

SiVL requires the Hercules and IC WorkBench tools. For the Y-2006.12 version of SiVL, you must install Hercules version Y-2004.12-SP3 and IC WorkBench version X-2005.09.

Note:

Hercules, IC WorkBench, and SiVL can be installed on different hosts or different platforms. The host names must be specified when the tools call for them.

For download and installation instructions, see the installation instructions available at http://www.synopsys.com/support/installation/install_guide.html.

Note:

The standard installation instructions in the Hercules document also apply to service pack releases (for example, Y-2004.12-SP3). For the latest information about supported platforms, see the product release notes.

To enable SiVL to use Hercules, source the following file:

```
% source install_dir/setup/hercules_setup.csh
```

where *install_dir* is the Hercules installation directory.

The hercules_setup.csh script sets HERCULES_HOME_DIR and other variables and checks whether the appropriate platforms are available. In the hercules_setup.csh script, set the value of HERCULES_HOME_DIR to the actual Hercules installation directory.

To enable SiVL to use IC WorkBench, set the path to the following:

```
set path=(ICWB_HOME/bin $path)
```
where *ICWB_HOME* is the IC Workbench installation directory.

Verifying the SiVL Installation

To verify the SiVL installation, make sure you are in a directory where you have read/write privileges.

Run a small test case to verify that SiVL runs correctly. Create a working directory named sivl_test and perform the following steps:

1. List the available sample xjc files.

   ```
   % runlrc -l
   ```

2. Select a sample xjc file from the list, and copy the file and models to your local machine.

   ```
   % runlrc -i xjc_file [target_xjc]
   ```

   Replace *xjc_file* with the sample file you chose. Supply the name for the destination file. SiVL copies the models directory with the appropriate models for the sample xjc file to the working directory.

3. Open the GUI and edit the parameters of the sample file.

   a. Enter

      ```
      % runlrc -e xjc_file
      ```

      This opens the graphical interface for the script builder and creates a new directory called Run.1. Runsets will be generated in this directory.

   b. Turn on the checks you want to run.

   c. Select Build > Build Now.

d. Exit the script builder.

4. Run SiVL from the newly created Run.1 directory.

```
% runlrc -r
```

This runs Hercules in the Run.1 directory. Results are placed in that directory as well. Note that the checkpoint data is stored in a directory underneath the working directory.

After the run finishes, you should have the following files and directories in the sivl_test directory:

-xjc_file
models/
checkpoint/
run_details_checkpoint/
run.1/

5. Launch IC WorkBench,

```
% icwb -socket port_number &
```

where *port_number* is an arbitrary, unused TCP port (between 1024 and 64000), which will be used to communicate between
IC Workbench and the analysis tool, Error Analyzer. For example,

```
% icwb -socket 5555 port_number &
```

Then load the output GDSII file, lrc_out.gds, located in the Run.1 directory (select the correct top cell, not the one with EV... in the name)

6. Launch Error Analyzer.

```
% analysis &
```

7. In Error Analyzer, select Viewer > Viewer Properties. In the Hostname dialog box, enter the host name where IC Workbench was launched (the default entry is localhost).
8. In the Port dialog box, choose the TCP port specified in step 5 and click OK.
9. In Error Analyzer, create a new database.
   a. Select Data > New Error Database.
   b. In the new window, use the Browse button to locate the Cell Map File in the top box.
   c. Using the Browse button, locate and insert a database name of your choice.

d. Provide “Sampling Trigger” and “Number of Samples” values in the lower boxes. Click OK.

e. Create a new report by choosing Select Data > New Report. In the new window, select any error of interest from the list on the left (for example, pb...).

10. From the newly generated table, select one row (or error) with the left mouse button; then with the right mouse button, open a menu for this error and choose View Error.

    In IC Workbench, the window zooms to the error corresponding to the selection you made in Error Analyzer.

    If the installation was successful, you will be able to complete all steps without any problems.