SaberDesigner
Release Notes

Release 2001.2

Avant! Corporation

ARN2001.2

Registered in England No. 2372734
Registered Office: Aldwych House, Aldwych, London WC2B 4JP

Avant! Corp.
9205 S.W. Gemini Drive
Beaverton, OR 97008

Table of Contents

Notice to Users: Read Before Using

Disclaimer

The VeriasHDL(TM) and Saber® simulators, and all other software products offered to the customer (Customer) by Avant! Corporation. (Avant! Products) are software programs designed and engineered by Avant! Corp. or its licensors. Avant! Products incorporate the latest ideas and theories in advanced technologies to assist an experienced, highly qualified engineer in the design of Customer's product. Avant! Products have been designed to provide limited analysis and are not intended to replace testing or other analysis of Customer's product. The advanced level engineer must use Avant! Products only as "tools" to help understand the potential performance of Customer's product while continuing to rely upon his or her own expertise.

Avant! Products are based on mathematical abstractions of real world environments or devices. Because Avant! Products work with abstractions, the use of Avant! Products can only provide an estimation of possible future performance of Customer's product. Customer acknowledges these limitations and agrees that it will not rely solely upon the results derived from any usage of Avant! Products in determining the final design, composition, structure, safety, reliability, or performance of any Customer product. Avant! will not be legally liable for any use of Avant! Products by Customer, including Customer misuse, errors in judgment or application of Avant! Products in the development or use of Customer's products.

Avant! does not represent that it has sufficient professional expertise in the application of Customer's product to determine the difference between the performance predicted by Avant! Products and the performance of any real product or device. Therefore, Avant! expects Customer will test Customer's product prior to any actual use. Where the failure of Customer's product may result in bodily harm, Customer must take additional measures, in accordance with the applicable standard of care, to ascertain that Customer's product conforms to Customer's specifications and standard engineering design practices. Regardless of end use application, any Customer product developed with the assistance of Avant! Products must be tested independently to confirm the results obtained from Avant! Products. Where a higher standard of care is required, Avant! recommends that the results of simulation obtained from Avant! Products be confirmed by outside independent professionals, including additional product testing and quality control.

SaberDesigner Release Notes — Release 2001.2

This document describes the new features and enhancements provided in the new 2001.2 SaberDesigner product release, along with important information to help you properly install and configure your system. Please review this material before you begin the installation.

2001.2 is a comprehensive release of all the SaberDesigner products, and replaces the 5.2/5.2.2 versions. Note that SaberDesigner now uses the Avant! release identification numbering standard, which reflects the two annual releases during or following the second and fourth quarter of each calendar year (e.g., 2001.2 will be followed by 2001.4). Interim updates or patches may also be provided.

New Features

iQBus

This release introduces iQBus, an exciting addition to the SaberDesigner family of software products from Avant! Corporation.

iQBus is a sophisticated design, simulation, and analysis package for the development of electrical and mechanical designs and experiments. It provides an electronic mechanism to facilitate coordinated efforts between various designers, and allows review of experiment results and reporting.

SaberScope

SaberScope Release 2001.2 contains several new features which engineers will find very useful and convenient for performing design analysis.

1. Signal Manager

SaberScope now offers improved signal search and display capabilities. This new feature improves finding waveforms within very large signal lists.

Enhancements were also made to signal name filtering to provide an easy way to find the same signal in two different waveform files so they can be compared.

Main Signal Manager Window

New Signal Pulldown menu

This allows you to select a pattern matching language to search for desired signals. Choose one of three "match" options:

`exact – string typed as it appears`

`glob – filename pattern matching using *,?,[abcd] magic characters`

`regular expression – vi/emacs/tcl style patterns using ^,$,.,[,],|,(,) magic characters`

The magic (or wildcard) characters have the following meanings:

Pattern language	Magic character	Meaning
Glob	*	any sequence of characters
Glob	?	any single character
Glob	[abc]	character a, or b, or c.
Regular Expression	.	any character.
Regular Expression	<x>|<y>	anything matching regular expression <x>, or regular expression <y>
Regular Expression	()	parenthesis used for grouping.
Regular Expression	$	end of path component, or full path, depending on settings.
Regular Expression	^	start of path component, or full path, depending on settings.
Regular Expression	[abc]	character a, b, or c
Regular Expression	[^abc]	any character but a, b, or c.

You may also select one of two options target options to specify whether the pattern is matched against single path components, or against full paths. This allows, for example, very specific selections of particular signals for only a given range of plotfile containers. The default target for pattern matching is "`path component`."

New Signal Filter field

Enter your filtering pattern in this field; from the adjoining pulldown, select whether the pattern denotes a set to be **Shown**, **Hidden**, **Selected**, or **Deselected**. Choosing the action causes it to be applied to the signals or containers matching the filter pattern in all plot files.

In addition, you may set a "**Cumulative**" mode. With Cumulative turned off (the default mode), each action on a set of signals or containers denoted by the filter pattern implies that signals or containers not matching the filter pattern have the reverse operation applied to them. For example, if you enter "**v***" as a filter pattern, have "**Show**" as the action, and "**Cumulative**" off, all the **v*** entries appear, and other entries not matching that pattern disappear. Again, if the action had been "**Select**", all of the v* entries would be highlighted (selected) and other entries not matching that pattern, deselected.

Individual Plotfile Signal List Window

New Signals Menu Item

As in the main Signal Manager window, you may select the signal pattern matching language (**exact**, **glob**, or **regular expression**), and the pattern matching target (**full** or **path component**).

New Filter(delim="?") field

This label next to the filter pattern provides the delimiter character used for signal path names for this plotfile. If you wish to match against full path names, separate individual path components with this character **("?"** in this case.). Enter your search pattern, then select the filter attribute from the options in the adjoining pull-down: **Show**, **Hidden**, **Selected**, or **Deselected**. This attribute is retained when you enter subsequent search patterns, until you reset the selected attribute option. And, **s** in the main Signal Manager window, you may set or unset the cumulative mode, with the same meaning.

2. Graph and Measure

SaberScope now provides eye diagram support. This new measure provides the ability to overlay the periods of a periodic waveform on top of one another in order to analyze signal characteristics such as jitter. Another new feature is vertical cursor spanning multiple graphs. This feature allows the user to perform delta measurements on signals on different graph regions.

3. Configuration Files

There is a new feature in SaberScope that allows you to save or clear a configuration. To do this, you must start the application from the directory in which your work will be performed. This new feature is accessed with **File>Configuration>Save**, or **File>Configuration>Clear** menu selections. In the

same menu, there is also a check box that lets you save the configuration on exit.

4. New Hot Keys

Hot keys or shortcuts, have been added to perform specified commonly used features. See the Supplement to Released Documentation section for details.

5. Waveforms Written in ASCII

SaberScope can now import and export ASCII. Any type waveform (including multi-member) can now be written to a file in ASCII format. This textual output format allows for easy transfer of the waveform data into a spread sheet. Waveform data written in this ASCII format can also be imported into SaberScope as a graph.

Cadence Frameway

6. Support for CDS

This release provides support for CDS Version 4.4.6.

7. Support for Inherited Connections

The Cadence Frameway/netlister release for 2001.2 release of Saber supports the inherited Connections feature provided by Cadence. See the Supplement to Released Documentation section for details.

NOTE

The .simrc file is a Cadence file and so can reside in any place along the Cadence search path for this file. A most common place for this file would be one's home directory.

SaberDesigner Applications

8. New Model Characterization Tools

The new Diode Characterization Tool provides support to create diode models for use in power electronic circuits. These models are well suited for examining switching transients and losses in power supplies.

The new Power MOSFET Characterization Tool provides support to generate level-1 MOSFET models for power electronic circuits which are well suited for examining switching transients and losses in power supplies. Models are easily characterizable from datasheet information. The tool allows interactive tuning of parameters.

9. Plotting simulation results in an Experiment using "ScopePlot"

The Experiment Editor for SaberDesigner has a new item called "ScopePlot" for plotting of simulation results data. This item is located on the "**PostProcessing**" menu of the Experiment Editor. ScopePlot allows you to specify a list of signals to display from a specified plotfile or to "capture" an existing Scope graph, including measurements and calculations, to regenerate after an analysis. See the Supplement to Released Documentation section for details.

SaberHarness

10. New types of drawings related to the basic harness design

Generated drawing capability allows generation of new types of drawings related to the basic harness design. Specific drawings include: Harness connectivity diagram; Ground Plane connectivity; Power Plane connectivity.

New shared inline connectors feature allows inlines to be treated as shared symbols.

After CVC check-in, open sheets will restore. This enhancement will automatically reopen the sheets that were open before the check in.

The Cable Element feature provides the capability of representing a cable using a bus-like concept. The cable structure with its assignment methodology remains the same as that of a cable symbol.

Saber Simulator

11. New Simulators

Several new commands have been implemented in the Saber Simulator:

IP_Report Command

Enhanced DIsplay Command

Enhanced List Command

See the Supplement to Released Documentation section for details.

12. Changed Memory Model

The default memory model for the Saber Simulator has been changed to FLAT. This makes it unnecessary for the user to increase the available memory with the saberld command on UNIX or the -size option on Windows NT or Windows 2000 when simulating large designs. As a consequence of this change, the simulator will no longer write a `.tbl` file when it quits, and

therefore it will not be possible to restart a simulation session with the -e option.

Users who want to retain the previous functionality can configure the Saber Simulator accordingly, by following these steps:

1. Start the Saber Simulator with any design
2. At the command prompt, enter the commands

```
CONFIG MEMory_model STANDARD
CONFIG SAVE
```

3. Quit the simulator
4. Copy the file `analogy.scf` that was created in your working directory to a directory in your `SABER_DATA_PATH`. A convenient place is `$HOME/config`, where `$HOME` indicates your home directory. If this directory doesn't exist, create it.

13. NaN and Inf now detected

The simulation algorithms have been improved to detect the out of bound values NaN and Inf defined by IEEE Std. 754/854. Such values can occur for a variety of reasons during the solution of the equations. In some situations the simulator is able to correct the situation and continue the simulation, making the detection transparent to the user. In other situations the simulator will issue an appropriate error message.

14. Deprecated MAST Features

As part of formalizing the definition of the MAST language several language features that were undocumented for a long time have now been marked as deprecated. These features include:

1. the states section

   The states section has been replaced by the much more flexible apparatus of When statements

2. external declarations in the template body

   External declarations should be in the template header

3. val and state declarations in functions

   Such declarations should be replaced by appropriate variable declarations of type number

The simulator will issue a warning for each use of a deprecated language feature if it is started with the -d deprec8 option. Deprecated language features will not be supported in the VeriasHDL Simulator but remain to be available in the Saber Simulator.

Installation and Configuration Notes

This section contains information about your installation, configuration, and operating system requirements. For additional details, see the corresponding installation and configuration guide for this release.

Beginning with release 2001.2, SaberDesigner uses avant daemon rather than saber daemon for licensing. SaberDesigner versions prior to release 2001.2 continue to use saber daemon.

15. Remove Release 5.x of SaberDesigner software before installing Release 2001.2

If you are upgrading from SaberDesigner 5.x to Release 2001.2, you should remove SaberDesigner 5.x using the **Uninstall** tool before installing Release 2001.2.

16. Installation and Configuration Guide and Release Notes available in PDF

The *Installation and Configuration Guides for UNIX* and *Windows* and the *SaberDesigner Release Notes* are located in the EXTRAS directory on the release image. The documents are in PDF format, and require a PDF file reader to view and print them.

The document files have the following file names:

Installation and Configuration Guide-UNIX	*SaberDesignerInstall_UNIX.pdf*
Installation and Configuration Guide-Windows	*SaberDesignerInstall_Win.pdf*

17. Platforms and Operating System availability

Compatible platforms for use with SaberDesigner Release 2001.2 are: SUN Solaris 2.6, 7 and 8; HP9000 HP-UX 10.20, 11.00; IBM AIX 4.3.2.0; Windows 2000 with Service Pack 1 or higher and Windows NT with Service Pack 5 or higher.

18. License Manager Version is FLEXlm 7.2e

FLEXlm 7.2e is the current version of the license manager. If you already have FLEXlm installed, **installation will overwrite it**. You will be prompted first for approval.

It is possible when you use FLEXlm's LMTOOLS that ships with the SaberDesigner 2001.4 release to check for availability of certain license features that you may obtain the following message:

"The license cannot be checked out because: FLEXlm version of vendor daemon is too old."

This message is incorrect, as license features can be successfully served. You can ignore this message. However if you wish to prevent this message from appearing, please read on.

This erroneous message will appear if you are running FLEXlm's lmgrd 7.2 (which ships with SaberDesigner 2001.4) and a version of saber daemon that was built with FLEXlm version 6.2 or older (e.g, the saber daemon that shipped with SaberDesigner 5.2).

To eliminate the erroneous message, you will need to upgrade your saber daemon (6.2 or older version) to a saber daemon built with FLEXlm 7.2. You can find this 7.2 saber daemon (saber-dmn) in the /Extras directory on the SaberDesigner 2001.4 CD.

On Windows systems, you will find the older saber daemon (saber_dmn) that you need to replace in C:\WINNT\system32. IMPORTANT: Save this older saber daemon elsewhere before replacing it in case you need to restore it later.

On UNIX systems, your system administrator can inform you where the older saber_dmn is located. The default location is SABER_HOME/bin.Again, you should save the older saber_dmn before replacing it with the newer one.

19. Supported and Non-Supported files in the Installation Directory

Avant! Corporation only supports files that are installed in the installation directory during the SaberDesigner installation process, and will not support any files that a user adds to the installation directory.

For example, if the user adds MAST template (`.sin`) files to the template directory in the installation directory, these files will not be found by SaberDesigner or any Avant! software tools. (18288)

20. *Uninstalling Release 2001.2*

When uninstalling 2001.2, the entire installation directory will be removed. If you have added files to this directory, they will be deleted as well.

(28000)

21. You must Install using an Administrator login

You must have administrative privileges for the installation to be successful. While the installation software will allow you to install if you do not have administrative privileges, the installation will not be done correctly.

22. Don’t Use License Only Installation Option

The License Server Only installation option in the Windows Installation program should not be used. If you wish to install a license server on a network, you should instead select the stand-alone installation rather than

the license-server-only installation. To minimize the installation disk footprint, select the Custom installation when the choice is presented, and then check only one product to be installed.

23. If you wish to remove SaberDesigner products, use the SaberDesigner Uninstall tool

To remove any SaberDesigner applications use the **Uninstall** tool from the SaberDesigner program directory. (23558)

Using the Add/Remove Programs utility from the Control Panel will not remove all SaberDesigner programs. This will cause new installations of SaberDesigner applications to function improperly.

All SaberDesigner applications must be terminated before running **Uninstall**.

You must wait five minutes before running **Uninstall** after shutting down an application or prepare for persistent dialog boxes which offer options of waiting for shutdown. (23381)

24. Uninstall will not run if `ccdoctor.exe` process is running in the background

If the software product ClearCase is installed on the same PC machine as the SaberDesigner application, it may be running a background process called `ccdoctor`. If this is the case, SaberDesigner applications can not be installed or uninstalled until that process is removed (by using the Task Manager). This only affects the install and uninstall process and will not affect running any SaberDesigner products once they have been installed. (18324)

25. Modify your PATH variable if using Netscape 4.04 or later version

If Netscape 4.04 or a later version is used, Netscape modifies the PATH variable so the AOL instant messenger (`Aim.exe`) is found instead of the SaberDesigner `aim.exe`. This modification may cause problems in using various SaberDesigner applications. If such problems arise, modify the PATH variable to find `/bin in the installation` directory before the AOL `bin` directory. (13631)

26. Upgrading from Release 4.x to Release 2001.2

If you are a SaberSketch user on versions prior to 5.0, DO NOT delete your currently installed Saber software

The Release 2001.2 version of SaberSketch does not read old DesignStar Schematic (`*.sch`) or symbol (`*.sym`) files. [DesignStar was the schematic capture environment prior to release 4.1. SaberSketch users may be unaware they are using DesignStar symbols.]

You will need to use Release 4.x to convert your old DesignStar symbol (`*.sym`) and schematic (`*.sch`) files to be compatible with Release 2001.2.

Make your DesignStar conversions before installing Release 2001.2. To convert and save these files refer to the instructions below.

Any old schematics (`*.sch`) that have not been loaded into a 4.x version of SaberSketch AND saved will have to be converted using SaberSketch version 4.x in order to be readable in SaberSketch version 5.2. Note that reading the schematic in to SaberSketch 4.x and saving it converts the schematic BUT DOES NOT CONVERT THE SYMBOLS. To convert the symbols (`*.sym`), each symbol must be read into the version 4.x SaberSketch symbol editor and then saved.

To facilitate the symbol conversion process, an AIM script is available that automatically converts DesignStar formatted symbols (`*.sym`) to the symbol format (`*.ai_sym`) readable by SaberSketch 4.x and 5.2. The DesignStar symbols will NOT be overwritten since the SaberSketch symbols use a different file extension. Any symbols previously converted to the new format (`*.ai_sym`) will not be overwritten by the AIM conversion script. The script converts all symbols in the current directory and in all sub-directories at all levels below the current directory. This will allow users with DesignStar symbols scattered about in various sub-directories under their home directory to simply run the script once at the top level of their home directory.

To convert your symbols follow the instructions in the `conversionREADME` file located in the `EXTRAS` directory on the distribution CD. If you have any questions please call Customer Support. (18627)

27. Installing SaberDesigner and DesignerHDL then invoking Sketch

For customers with installations of both SaberDesigner and DesignerHDL on the same platform, double clicking on a schematic design file to invoke Sketch may bring up the version of Sketch that was not intended. This is because both SaberSketch and SketchHDL operate on files with the same extensions. This introduces an ambiguity in the file associations. To work around this problem, the intended version of Sketch should be invoked first and then the schematic design file should be opened from within the application. (27129)

28. Online Documentation must be installed to disk

The installation program gives you the option of allowing online help to be accessed from the distribution CD. Due to a defect this option will not work. SaberBook must be installed for the application to access online documentation. (25808)

New Models

Online documentation for some of the new Models is not yet available. To obtain more information about these templates, place the part, activate the **Symbol Menu** (right mouse click), and choose **View Template**.

New Generic Templates for Release 2001.2

29. IQBus Templates

Template Name	Description
dcdc_1	Provides a fully behavioral DC/DC converter model.
dr_cycles	Generates vehicle velocity profiles that can be either one of the common test cycles or a custom-built cycle.
gen_table_1	Provides a Level 1 generator model with the following features: rotational velocity and thermal case pins, percent-usage test point, maximum generation current based on temperature and rotor speed (rpm), efficiency based on rotor speed (rpm), and observed rotor torque dependent on efficiency.
load_cp	Models a constant-power load (constant `i(v(p)-v(m))`product).
load_cp_scale	Implements a power load where the load power may be varied during simulation.
load_cp_switch	Provides a switchable constant-power load. The `i(v(p)-v(m)` product is constant when turned on.
load_s3	Implements a load that may be sequentially switched among three different load values.
load_s3_scale	Implements a load that may be sequentially switched among three different load values. The load value is scaled during simulation with the value of the scale connection point.
load_s3_switch	Implements a load that may be sequentially switched among three different load values.
load_t	Provides three different time-dependent loads determined by the `data_type` provided by the user.

Template Name	Description
load_t_scale	Provides three different time-dependent loads determined by the `data_type` provided by the user. The load value is scaled during simulation with the value of the scale connection point.
load_t_switch	Provides three different time-dependent loads determined by the `data_type` provided by the user. The load may be specified as a resistive load or current load, and can be switched on and off during the simulation by the external control signal.
load_v	Provides three different speed-dependent loads determined by the `data_type` provided by the user. The `disable` parameter is used to enable or disable the load before simulation.
load_v_scale	Provides three different speed-dependent loads determined by the `data_type` provided by the user. The load value is scaled during simulation with the value of the scale connection point.
load_v_switch	Provides three different speed-dependent loads determined by the `data_type` provided by the user. It can be switched on or off during the simulation by the external control signal.
veh_eng_0	A model of a converter that computes an engine power take off shaft angular velocity when a vehicle moves at given speed.
veh_eng_1	A model of a converter that computes an engine power take off shaft angular velocity when a vehicle speed profile is applied to its input. The template also predicts instantaneous and cumulative fuel consumption, based on user supplied fuel consumption data.

30. Pneumatic Templates

Name	Description
flow_acpf_pn	Ideal mass flow rate source. Data is read from the Saber AC plot file
flow_clock_pn	Ideal mass flow rate source. Clock output
flow_dc_pn	Ideal mass flow rate source. Constant output

Name	Description
flow_exp_pn	Ideal mass flow rate source. Exponential output
flow_pls_pn	Ideal mass flow rate source. Pulse form output
flow_pn	Ideal mass flow rate source. General
flow_ppwl_pn	Ideal mass flow rate source. Piece-wise linear. Periodic
flow_pwl_pn	Ideal mass flow rate source. Piece-wise linear
flow_sin_pn	Ideal mass flow rate source. Sinusoidal
flow_trpf_pn	Ideal mass flow rate source. Data is read from the Saber TR plot file
flow_dc_gas	Ideal source of constant mass flow rate of gas at specified temperature
flow_pwl_gas	Ideal source of piece-wise linear mass flow rate of gas at specified temperature
pres_pwl_gas	Ideal source of piece-wise linear specified gaseous pressure with gas at specified temperature
pres_dc_gas	Ideal source of constant gaseous pressure with gas at specified temperature
pres_acpf_pn	Ideal pressure source. Data is read from the Saber AC plot file
pres_dc_pn	Ideal pressure source. Constant output
pres_clock_pn	Ideal pressure source. Clock output
pres_exp_pn	Ideal pressure source. Exponential output
pres_pls_pn	Ideal pressure source. Pulse form output
pressure_pn	Ideal pressure source. General
pres_ppwl_pn	Ideal pressure source. Piece-wise linear. Periodic
pres_pwl_pn	Ideal pressure source. Piece-wise linear
pres_sin_pn	Ideal pressure source. Sinusoidal
pres_trpf_pn	Ideal pressure source. Data is read from the Saber TR plot file
pr_red_pn	Pressure reducing valve. Pressure regulator

Name	Description
relief_pn	Pressure relief valve
orif_pn	Pneumatic orifice. Fixed
orif_var_pn	Pneumatic orifice. Variable
check_pn	Check (No return) valve
shuttle_pn	Shuttle valve
qr_n_valve	Quick release valve
v_2way_pn_mo	2 way, mechanically-operated
v_2way_pn_po	2 way, pneumatically-operated
v_2way_pn_eo1	2 way, electrically-operated
v_3way_pn_mo	3 way, mechanically-operated
v_3way_pn_po	3 way, pneumatically-operated
v_3way_pn_eo1	3 way, electrically-operated
v_5way_pn_mo	5 way, mechanically-operated
v_5_2_pn_eo1	5 way, 2 position, electrically-operated
v_5_3_pn_eo1	5 way, 3 position, electrically-operated
cyl_sa_pn	Pneumatic cylinder. Single-acting
cyl_da_pn	Pneumatic cylinder. Double-acting
br_cham_pn	Breaking chamber
act_cham_pn	Actuating chamber
mass_fl2var	Mass flow rate to Control interface
var2mass_fl	Control to Mass Flow Rate interface
p2var_pn	Pressure to Control interface
var2p_pn	Control to Pressure interface
pn_bottle	Pressurized bottle
pipeline_1_pn	Pipeline. Level 1

31. Hydraulic Templates

Name	Description
Accum_g_1	The model is a level 1 model of a gas-charged accumulator in which the gaseous process is assumed to be adiabatic with no heat exchange between gas, shell body, and atmosphere. The model fits applications with simulated cycle of about 5-10 sec. If the system is observed for more than 10-20 sec, the more accurate model should be built by using hydraulic, pneumatic, thermal, and mechanical building blocks. The accumulator is simulated as a combination of a gaseous and liquid chambers interacting through the bladder. The bladder separates the chambers and interacts with the hard stop as it approaches its lower (empty accumulator) or upper limits.
hyd_chmb_var	This model simulates a variable volume hydraulic chamber as it is seen in various hydraulic machines. The template was developed to be used as a building block in models of cylinders, plungers, pistons, actuators, etc. in which fluid compressibility is believed to be essential and, moreover, the volume of a fluid under pressure changes its volume significantly. Additionally, to the difference of the constant volume chamber model "hyd_chmb_cv", the fluid is considered as a mixture of liquid and entrapped (non dissolved) gas thus making it possible to simulate chambers in which pressure can drop below atmospheric pressure. The fluid bulk modulus is considered as a function of pressure.
Hyd_inertia	The model simulates inertial property of a fluid confined in a round rigid pipe. It is developed mainly as a building block for pipeline models. The template computes the pressure difference across a pipe which it takes to accelerate or decelerate fluid in a pipe.
Chamb_act_0	Actuating chamber (level 0). The model simulates a chamber of a hydraulic cylinder in its simplest form. Developed mainly as a building block, but can represent single acting cylinders especially at the initial design stages.

Name	Description
Chamb_act_1	Actuating chamber (level 1). The model simulates a chamber of a hydraulic cylinder and, in addition to the level 0 model, accounts for fluid compressibility in a variable volume chamber. The fluid is considered as a mixture of entrapped gas and liquid making it possible to simulate cylinder behavior at negative pressures as well. The model is intended to simulate single acting cylinders or to be used as a building block.
Cyl_h_sa_id	Single acting hydraulic cylinder. The model represents a cylinder similar to the level 0 actuating chamber, but treats a cylinder as a component, offering more cylinder-specific output variables and stress measures. The model is intended to be used at initial design stages when correct circuit functionality is of major concern or can represent low velocity cylinders in which fluid compressibility is believed to be insignificant.
Cyl_h_sa	Single acting hydraulic cylinder. The model represents a cylinder with hard stops at the ends of stroke and oil compressibility in a variable volume chamber. The model corresponds to the most popular cylinder configuration and is recommended for practically any application. The model does not account for negative pressures and can not be used in applications in which cavitation can occur.
Cyl_h_da_id	Double acting hydraulic cylinder. The model is the simplest representation of this type of cylinders. It accounts for energy transduction feature only. No motion limitations or fluid compressibility is considered. This simple and robust model can be used to simulate low velocity actuators or be successfully used at initial design stages when main objective is a correct circuit functionality.
Cyl_h_da	Double acting hydraulic cylinder. The model represents a cylinder with hard stops at the ends of stroke and oil compressibility in a variable volume chamber. The model corresponds to the most popular cylinder configuration and is recommended for practically any application. The model does not account for negative pressures and can not be used in applications in which cavitation can occur.

Name	Description
Cyl_h_da_2	Double acting hydraulic cylinder (level 2). The model represents a cylinder with hard stops at the ends of stroke and oil compressibility in a variable volume chamber. To the difference of level 0 and 1 models, fluid properties such as viscosity, bulk modulus, density, and bulk modulus are considered temperature and pressure dependent, thus making it possible to account for possible cavitation. The fluid properties can be specified analytically with a set of coefficients, or provided in a table or file format as an array of points vs. pressure and temperature.
pump_fd	The model is a data sheet based model of a Fixed displacement pump. All the template parameters such as pump displacement, mechanical and volumetric efficiencies, etc. can easily be determined from a catalog or data sheet. The pump behavior is such that the input shaft rotation produces flow rate that is proportional to that rotation angular velocity. Torque at a driving shaft is proportional to pressure drop across the pump. The leakage flow is determined by the Hagen-Poiseuille equation. It is linearly proportional to the pressure drop across the pump and inversely proportional to the fluid absolute viscosity.
Hyd_mot	The model is a data sheet based model of a positive displacement hydraulic motor of any type (piston, vane, screw, gear, etc.). The model accounts for mechanical and hydraulic losses which are estimated with mechanical and volumetric efficiencies whose values are normally provided in the catalog. The hydraulic loss (leakage flow) is assumed to be proportional to the pressure drop across the motor. The leakage coefficient is determined on the basis of given volumetric efficiency, nominal pressure drop, and nominal angular velocity.

Name	Description
Motor_ax_pist	The template is a model of a hydraulic, five-piston, variable displacement axial piston motor or pump. In simulates a machine as a combination of five pistons that are fitted to bores (chambers) in the rotor parallel to the axis of rotor rotation. The rotor is installed between a porting valve and a swash plate. Each bore (chamber) has an orifice at the bottom. As the rotor turns, the swash plate forces the pistons to reciprocate with respect to the rotor and the orifice passes one of two crescent-shaped slots arranged on the porting valve surface. One of the slots is connected to the machine inlet while the other is connected to the outlet. The swash plate and the porting valve are arranged in such a way that each piston passes the inlet as the rotor turns 180˚ and passes the outlet as it turns remaining 180˚ of the revolution. Depending upon the swash plate angular position, which can be either positive or negative, and direction of the rotor rotation, the machine can work as a pump or as a motor.
Swash_plate	The model simulates a swash plate of hydraulic axial piston pumps and motors. The swash plate is a mechanism that forces pistons to reciprocate relative to a rotor as it turns. From energy standpoint, a swash plate is a transducer of a translational piston motion into a rotational motion of the machine rotor. The model simulates interaction between one piston and the swash plate. There must be as many swash plate models in the pump/motor model, as there are pistons in the design of a particular machine. The model represents a swash plate as an ideal converter. In other words, no inertia, friction, or any other loading is considered. If necessary, all these effects can be accounted by attaching a respective building block to the template terminals.

Name	Description
Pipe_ss_0	The model is a level 0 model of a hydraulic pipe which was developed to be used individually to simulate hydraulic pipelines or as a building block for a multi-segmented pipe representation. The model represents a tube used for transmission of hydraulic fluid in one of its simplest form - the model accounts for friction loss only. Neither fluid inertia nor compressibility of both the fluid and the pipe walls are considered in the model. The end effects are also neglected in this round tube model. Another assumption is that the Reynolds Numbers during transition from laminar to turbulent regimes can be determined with the linear interpolation between extreme points of these regimes.
Pipe_ss_1	The model is a level 1 model of a hydraulic pipe which was developed to be used individually to simulate hydraulic pipelines or as a building block for a multi-segmented pipe representation. The model represents a compliant tube used for transmission of hydraulic fluid. The model accounts for viscous friction, fluid inertia, and compressibility of both the fluid and the pipe walls. The fluid is considered as a mixture of liquid and entrapped, non dissolved gas as this is the case with practically all working fluids. As the result, the media bulk modulus is dependent on pressure and model covers not only positive, but also negative gage pressure range. The end effects are not considered in this round tube model, and it is assumed that the Reynolds Numbers during transition from laminar to turbulent regimes can be determined with the linear interpolation between extreme points of these regimes. The model is based on a so-called T-equivalent circuit of a pipe which represents this inherently distributed parameter element as a set of interconnected lumped parameter components. In the T-circuit, a pipe is split evenly into two equal parts and each part is substituted with the hydraulic resistance and hydraulic inductance. The fluid and pipe wall compliance is simulated with the hydraulic capacitance placed between the circuit middle point and the ground.

Name	Description
pipe_ms_1	The model is a level 1 model of a hydraulic pipeline which is based on the concept of lumped parameter representation of long transmission lines. In accordance with the concept, the pipe which is inherently a distributed parameter component is represented as a set of connected in series identical segments. It is assumed that the larger the number of segments, the closer the lumped parameter model becomes to its distributed parameter counterpart. The multiple segmented pipeline model incorporates the MAST "collapsed node" mechanism which allows to vary number of segments in the model from 1 to 9 in accordance with the value assigned to template parameter "elem_numb". The pipe_ss_1 model of a pipeline is used as a model of a pipeline segment.
Elbow_con_h	The model simulates pressure loss associated with the elbow type hydraulic connection as it appears in fittings, manifold channels, etc. The model is built of the generic local hydraulic resistance template "local_h_res".
Grad_enlarg_h	The model represents a local hydraulic resistance in the form of a gradual enlargement/contraction. The model accounts for a pressure loss as a gradual enlargement as liquid flows in positive direction and as a gradual contraction for negative flows. The pressure loss caused by such a resistance is frequently referred to as a minor loss.
Local_h_res	The model simulates a local hydraulic resistance, that is of a loss inducer such as bends, elbows, fittings, sudden changes in flow cross section, filters, etc. The energy loss caused by such a resistance is computed with the semi empirical formulae based on the experimentally determined K-factor. Since the loss could depend on the flow direction, the model allows to provide K-factor for both the direct and the reverse flow. The model distinguishes between the turbulent and accounts for laminar flows. If the Re-K-factor relationship is known, the model allows to specify it directly in the table or file format.

Name	Description
Sud_enlarg_k	The model simulates a local hydraulic resistance in the form of a sudden enlargement/contraction. The model accounts for a pressure loss as a sudden enlargement as liquid flows in positive direction and as a sudden contraction for negative flows.
Tee_con_h	The model represents a tee type hydraulic connection as it appears in fittings, manifold intersecting passages, etc. The model is built of three interconnected models of a local hydraulic resistance (local_h_res).
Porting_vlv	The model simulates a variable orifice as it is seen in porting valves of hydraulic axial piston pumps and motors. In this type of hydraulic machines, each piston chamber has an orifice at the bottom. As the rotor turns, the orifice passes one of crescent-shaped slots arranged on the porting valve plate surface. Since the slots are connected to either high or low pressure ports of the machine, the piston is connected through the orifice to one of these ports depending upon their angular position. The model simulates interaction between the chamber and one slot only. If a chamber interacts with more than one slot, there must be as many orifice models as there are slots on the machine model. The model was developed as a building block to be used in various types of axial piston machines.

32. Mechanical Templates

Name	Description
Clutch_w_mo	The model simulates a mechanically operated friction clutch. The friction in a contact between driving and driven shafts is controlled by a force applied to the mechanical control mechanism. The friction torque is simulated as a function of a relative velocity and assumed to be the sum of the Stribeck, Coulomb (dry), and viscous terms. The transition from the breakaway to the Coulomb friction is assumed to be exponential. No dynamic friction terms are accounted in the model.
Gear_box_r	The model is a data sheet based model of a conventional gearbox with angular terminals. The template represents a gearbox as a macro model rather than as a transmission of any particular type. It accounts for a gear ratio and efficiency only. No inertia or friction is considered in the model.
Gear_box_w	The model is a data sheet based model of a conventional gearbox with angular velocity type terminals. The template represents a gearbox as a macro model rather than as a transmission of any particular type. It accounts for a gear ratio and efficiency only. No inertia or friction is considered in the model.
Gear_diff_r	gear_diff_w is a model of a differential gear box with angular terminals. The model represents a gearbox as a macro model accounting for gear ratios and efficiencies only rather than as a differential gear of some particular type. No inertia or friction is considered in the model.
Gear_diff_w	gear_diff_w is a model of a differential gear box with angular velocity type terminals. The model represents a gearbox as a macro model accounting for gear ratios and efficiencies only rather than as a differential gear of some particular type. No inertia or friction is considered in the model.

Name	Description
Lead_screw	The model represents a lead screw – the mechanism that converts rotational motion applied to the screw hub into a translational motion of its nut. The model simulates a mechanism as a data sheet base model accounting for mechanism generic parameters such as a pitch and efficiency. No inertia or friction is considered in the model.
Lever_1	lever_1 is a model of a type 1 lever. The lever of this type is pivoted at the end with two joints located on the same side of the pivot. The model accounts for lever geometrical dimensions. No mass or friction is considered in the template and rotation angle is assumed to be small enough to avoid usage of a hard stop or similar rotation limiting devices.
Lever_2	The model represents a type 2 lever. The lever of this type is pivoted in the middle with two joints located on opposite sides of the pivot The model accounts for lever geometrical dimensions. No mass or friction is considered in the template and rotation angle is assumed to be small enough to avoid usage of a hard stop or similar rotation limiting devices.
Lever_free	lever_free is a model of a free (or summing) lever. The lever of this type sums motions applied to any of its two terminals and generates it at the third terminal. The generated motion is proportional to respective ratios and input displacements. No mass or friction is considered in the template and rotation angle is assumed to be small enough to avoid usage of a hard stop or similar rotation limiting devices.
Spr_t_prld	spr_t_prld is a model of a mechanism consisting of a preloaded spring and a rod. The suspended inside the spring rod can move freely with respect to the spring only until its relative to the case position exceeds a specified limit (clearance). After that, the rod butts against the preloaded spring and causes it to compress. Such mechanisms are widely used, for instance, in directional valves, shock absorbers, single-acting cylinders, servo cylinders, etc.

Name	Description
fr_r_scv_nf	The model represents friction in the contact between two rotating bodies which is acted upon with a squeezing force. The friction torque is simulated as a function of a squeezing force and relative velocity and assumed to be the sum of the Stribeck, Coulomb (dry), and viscous terms. The transition from the breakaway to the Coulomb friction is assumed to be exponential
Shaft_r	The model simulates a cylindrical solid or hollow shaft with rotational velocity terminals. It is a lumped parameter representation of an inherently distributed parameter element with an infinite number of eigenfrequencies. In a lumped model, the shaft is represented as a combination of three connected in series springs and two inertias exhibiting two resonant frequencies. The model was developed for applications in which shaft inertia is comparable to that of a motor and/or load.
Stop_r_pl	stop_r_pl is a model of a two-position mechanical rotational hard stops that restricts rotation of bodies to which the model is connected in both directions. The impact interaction is assumed to be absolutely plastic. It means that the entire energy is assumed to be dissipated while colliding bodies penetrate into each other and there is no bounce. At the same time, the model exerts no restriction to the external torque trying to separate penetrating bodies. The template covers cases of a comparatively low-velocity interactions, interactions through a cushioning layer of fluid between colliding bodies (hydraulic and pneumatic actuators), and applications in which high frequency oscillations caused by a collision can be neglected or must be avoided.
Fric_w_scv	The model simulates friction between a rotating shaft and a frame or between two rotating bodies. The friction torque is simulated as a function of a relative velocity and assumed to be the sum of the Stribeck, Coulomb (dry), and viscous components. The transition from the Stribeck to the Coulomb friction is assumed to be exponential.

Name	Description
Shaft_w	The model simulates a cylindrical solid or hollow shaft with rotational velocity terminals. It is a lumped parameter representation of an inherently distributed parameter element with an infinite number of eigenfrequencies. In a lumped model, the shaft is represented as a combination of three connected in series springs and two inertias exhibiting two resonant frequencies. The model was developed for applications in which shaft inertia is comparable to that of a motor and/or load.
Spring_w	The model simulates a rotational spring with rotational (velocity) terminals. It was mainly developed to simulate stiffness of cylindrical bodies like solid or hollow shafts. Therefore, the spring can be specified by either directly providing the spring rate value or by providing cylindrical body sizes and material properties. In the latter case, the spring rate will be computed inside the model.

33. Semiconductor Templates

Name	Description
mm9	combined Philips mos902 and mos903
hspm13	HSPICE Level 13
hspm54	Hspice level 54, same as BSIM4
hspm3	HSPICE Level 3

34. Berkeley Models

The following table shows the Berkeley BSIM3v3 and BSIM4 models available in Saber with this release. Also shown are some models/ enhancements planned for a subsequent release (p=pending):

	Hspice Enhance-ments	3v3	3v3.1	3v3.2	3v3.2.1	3v3.2.2	4.1.0	4.2.0
mbsim3v3.sin	N	x						
mbsim3v3_1.sin	Y		x					
mbsim3v3_2.sin	Y			x	x	x		

	Hspice Enhance-ments	3v3	3v3.1	3v3.2	3v3.2.1	3v3.2.2	4.1.0	4.2.0
spm49.sin	Y	x	x	x	P	P		
hspm53.sin	N	P	P	P	P	P		
hspm54.sin	N						x	x

It is planned to upgrade hspm49.sin to include BSIM3v3.2.1 and BSIM3v3.2.2. Level 53 is also planned though it can be currently matched with mbsim3v3_1.sin or mbsim3v3_2.sin.

HSpice level 54 is a direct port of BSIM4.1.0 and does not contain any other enhancements aside from the parameter dtemp. hspm54.sin is a port of HSpice MOS level 54 (HSpice version 2000.4) with an upgrade to BSIM4.2.0.

35. Control System Templates

Name	Description
c_chirp	Sinusoidal control source with frequency a linear function of time
c_friction	Control domain non-linear friction
c_memory	1 time step buffer
c_ramp	Linear ramp control source
c_step	Step control source
c_switch2	Two input switch
conv_d2var	logic_4 to var converter
conv_var2d	var to logic_4 converter
memory_d	1 time step digital memory

New Characterized Components for Release 2001.2

36. New components released in the DX component library.

IGBT templates	
cm50dy-12h	cm75dy-12h
cm100dy-12h	cm150dy-12h

cm200dy-12h	cm300dy-12h
cm400dy-12h	cm300ha-12h
cm400ha-12h	cm600ha-12h

Diode templates	
apt2x61d60j	d5kp30a
d5kp40a	dsei30
dsep30	tpsma39a

Power MOS templates	
apt10050lvr	apt40m70jvr
apt8030lvr	buz100
mrf151	spw11n60s5
spw20n60s5	

Opamp templates
AD842
AD8047
AD8057

PWM templates
LT1680 - High Power DC-DC Step-up Controller
UCC2882 - Average Current Mode Sychronous Buck Controller with 5-bit DAC
UCC3882 - Average Current Mode Sychronous Buck Controller with 5-bit DAC

CAN bus controller templates ***The CAN bus controller templates are not in the DX component library but are in the iQBus product.***
PCA82C250
TJA1050

Brushless Motor Controller ICs
MC33033
UC1625

Instrumentation Amplifiers
AD623A
AD623ARM
AD623B

Miscellaneous Parts
D356A Electroluminescent Lamp Driver IC
HCPL-063N - Optocoupler IC

37. Products, Versions and Compatibility for Release 2001.2 Product Matrix for the SaberDesigner Co-Simulator Interfaces and Frameway Integrations

Product/Version	**Solaris 2.6, 7 and 8**	**HP-UX 10.20, 11.00 (HP8000)**	**Windows NT 4.0 and Windows 2000**	**IBM RS6000 AIX 4.3.2**
Co-Simulator Interfaces				
Saber/Verilog Co-Sim Interface w/ Cadence 4.4.6	Verilog 2.7	Verilog 2.7	No	No
Frameway Integrations				

Product/Version	Solaris 2.6, 7 and 8	HP-UX 10.20, 11.00 (HP8000)	Windows NT 4.0 and Windows 2000	IBM RS6000 AIX 4.3.2
Frameway for Cadence (3.5)	CDS 4.4.6	CDS 4.4.6	No	No
Frameway for EDIF (3.4)	Berkeley EDIF 3.2	Berkeley EDIF 3.2	No	No
Frameway for Innoveda (formerly Viewlogic) (3.4)	VWL 6.1	VWL 6.1	Workview Office 7.5.3; ePD 1.0 and 1.1	No

Problems Fixed

38. hspm28.sin and hspm49.sin

An error in some code common to hspm28.sin and hspm49.sin caused the default value of the model parameter tref (tnom) to be set to the circuit temperature (temp) rather than the default value for temp (27C). This has been corrected. (27404)

39. Geometrical Corrections

The geometrical corrections for the narrow width coefficients dvt0w, dvt1w, and dvt2w:

dvt0w = f (ldvtw0, wdvtw0, pdvtw0)

dvt1w = f(ldvtw1, wdvtw1, pdvtw1)

dvt2w = f(ldvtw2, wdvtw2, pdvtw2)

were accidentally omitted in mbsim3v3.sin, mbsim3v3_1.sin, and mbsim3v3_2.sin in prior versions of Saber. dvt0w, dvt1w, and dvt2w will have been used directly as input; no geometrical scaling would occur.

This bug would only result in an error if any of the length, width, and cross terms (ldvtw_x, wdvtw_x, and pdvtw_x) were set in a model card. Otherwise they default to 0 and the scaling equations are irrelevant. The amount of error would be a function other model parameters, instance properties, and circuit topology.

The omitted scaling equations have been added to the above noted models in this release. (27636)

40. Scaling of the Thermal Voltage

An problem with the scaling of the thermal voltage of template hspd has been corrected. In versions 5.2 through 5.2.2, the thermal voltage used a constant 25C, rather than the instance temperature. This caused the template to give incorrect results for any quantities sensitive to thermal voltage if tref=temp, except for temp=25C. (27636)

41. Model Parameter Aliases Available in HSpice

Some model parameter aliases available in HSpice were inadvertently omitted from the new MOSFET models hspm3.sin and hspm13.sin (HSpice level 3 and level 13 model ports). These include aliases for clm, kp, lmlt, wmlt, uo, xl, and xw. If your existing model declaration uses an alias for any of these parameters, please replace it with the corresponding primary name listed here until this is rectified in the next release.

42. The following change has been made to the stripline model

1. Resistance and conductance expressions have been fixed (see below).
2. The symbol has been fixed so that the pins are correctly placed.
3. The error message returned by the template when n is greater than 20 (maximum number of elements allowed) is more meaningful.
4. The parameter pf has been removed from the argument list. tand is now the only way to characterize the dielectric losses. The symbol reflects this change as well.

Below are the equations actually used in the template:

```
math_eo=8.85p

math_uo=1.26u
```

Characteristic impedance:

```
z0 = sqrt(math_uo/math_eo)/(sqrt(er)*(w/h+2))
```

Capacitance per meter:

```
c = (w/h+2)*math_eo*er
```

Inductance per meter:

```
l = c*z0*z0
```

Resistance per meter (no skin effect):

```
r = 1/(sigma*w*t)
```

Conductance per meter (dielectric losses):

```
g = 2*math_pi*er*math_eo*tand*w/h
```

(26860)

43. Model gear box gear_r

The model of a gear box gear_r was developed with respect to three external ports. The port 'case' is unrealistic physically and introduced with a mathematical error. This template is obsoleted in the 2001.2 Release and replaced by 4 new models: gear_box_r, gear_box_w, gear_diff_r, and gear_diff_w. (25869)

Known Problems and Workarounds

SaberDesigner Applications

44. If you are running SaberDesigner applications using Exceed, your fonts may not appear as they should

Windows NT or Windows 2000 with Exceed

If you are running SaberDesigner applications using Exceed, your fonts can be unreadable. Exceed by default comes in a single font mode. To allow additional fonts, select the following:

Start >Programs > Exceed > XConfig > Font > FontDataBase > Automatic Font Substitution

Automatic Font Substitution is a check button that should NOT be checked. It should be off. All fonts will then be available in SaberDesigner applications through Exceed. (16741)

45. The Aim Macro Recorder will not work correctly if a subdirectory exists beneath the `.aimMacro` directory

The Aim Macro Recorder will not work correctly if a subdirectory exists beneath the `.aimMacro` directory. In such a case, you will see a Tcl/Tk error.

The workaround is to move the subdirectory to another location outside of the `.aimMacro` directory. (17578)

46. If you are running `saber -b` on an SaberDesigner example, the results of the simulation may not be displayed

If you are running `saber -b` on a SaberDesigner example that plots the results of that example, SaberScope should open and display the results of the simulation. This does not always occur due to a race condition in the `.scs` file.

The workaround is to open SaberScope before running `saber -b` from your system command line. (16715)

47. Plotter users must set a “Roll Feed Printer” Option

A new option has been added to the SaberDesigner UNIX Add Printer dialogue box when adding an HPGL printer. The name of this option is Roll Feed Printer. If the printer you are adding is a roll feed printer, e.g. the Hewlett Packard 600 or 750c plotter, make sure you select this option. Failure to do so may result in printouts with an incorrect orientation. (22574)

48. Remove conflicting `dformd.dll` files in WINNT/ System 32 directory

The main FORTRAN library, `dformd.dll`, must match the file installed with DEC Visual FORTRAN 5.0 with patch C. If there are any other files named `dformd.dll` in your system, remove or rename them. Conflicts between versions of this library will result in application crashes. (20691)

49. SaberDesigner software on Windows 2000 and Windows NT does not load files or path names with spaces

For SaberDesigner purposes, do not use spaces in file or folder names on Windows 2000 or Windows NT Systems. We suggest you use the underscore character instead. For example:

`XYZ Design` is NOT ALLOWED

`XYZ_Design` is ALLOWED

Using spaces in folder or file names can cause start menu shortcuts to not work, design files to be “lost” and other problems with unrecognized files and folders. (18868), (18536)

50. Upgrading From AIM 1.0 to AIM 2.+ for Release 4.X users

SaberDesigner’s AIM scripting language was upgraded in the SaberDesigner 5.1 and subsequent releases to support the current versions of its base languages, Tcl 8.0/Tk 8.0. AIM macros that you may have written in versions prior to 5.1 may need to be modified to work with the version of AIM provided in this release. For information on the differences between AIM 1.0 and AIM 2.2, refer to the AIM 1.0 to 2.2 Conversion Guide available online in SaberBook.

51. Mouse Buttons in Windows 2000 on Laptops

The SaberDesigner middle mouse button (MMB) paste and pan operations will not work with the default Windows 2000 mouse driver settings on laptop systems.The MMB operation needs to be set to "`Middle Button`" in the mouse driver software for correct operation in SaberDesigner. Since the Windows 2000 mouse driver software does not offer the "`Middle Button`" option, users must obtain the latest release of their specific mouse driver software and select the "`Middle Button`" setting option. (27163)

52. Plotter users must set a “Roll Feed Printer” option

A new option has been added to the SaberDesigner UNIX Add Printer dialogue box when adding an HPGL printer. The name of this option is **Roll Feed Printer**. If the printer you are adding is a roll feed printer, e.g. the Hewlett Packard 600 or 750c plotter, make sure you select this option. Failure to do so may result in incorrect orientation of printouts. (22574)

Saber Simulator

53. The pf _ option does not work for noise or distortion analyses

There is no workaround for this problem. (9014)

The `alter` command can fail in certain cases

The `alter` command fails with requests of the form:

```
alter /.../par=1
```

if the parameter `par` appears in different templates.

The workaround is to run separate `alter` commands for each template, for example:

```
alter /.../a.*/par=1,then
alter /.../b.*/par=1,etc.)
```

(9646)

54. Saber may occasionally run out of memory

Saber may occasionally run out of memory if the standard memory model

is used. If the standard memory model is used the Saber simulator may occasionally run out of memory (`ERR_NOSTORE`) when you attempt to run a transient analysis after a DC analysis.

The workaround is to use the `dctr` command, or to change the memory model by using the command:

```
config memory_model flat.
```

(9917)

55. The sigset command does not support setting digital signals

The workaround for this problem is to use node values in the sigset command of 1, 2, 3, or 4, corresponding to the state_l4 digital values _0, _1, _x, _z, respectively.

Example - using a circuit with a digital node called "`inv_out`". The initial point file is called "`ip`".

Sigset command (at the SaberGuide Transcript Window command line):

```
sigset inv_out 3 (ip dc)
```

Results of a Display Initial Point command:

```
di(siglist /
                    dc
time                0
```

```
----------------------

i(v_sin.v_sin1)      0

inv_in               1

0                    0

inv_inv_l4_1         0

inv_out              x
```

Note that the node "`inv_out`" is displayed with the value of "x". This value "x" will appear in SaberScope as having the state_l4 value "_x". (10035)

56. The display command may compute some vals incorrectly if an alter command was executed after the initial point was created

There is no workaround for this problem. (15721)

57. Accelerating netlister speed

An Accelerate Netlister option has been added to the **Edit > Saber/Netlister Settings**, Netlister tab.

The default setting is **No**, which is the slowest setting, but which brings up a window indicating netlister activity. The **Yes** setting increases netlisting speed, but no window showing netlister activity is displayed. When the **Yes** option is selected you should be aware that larger designs may appear as if they are not netlisting when they are in fact netlisting. (28106)

58. Older designs must be saved before netlisting or simulating

Designs created with previous versions of supplied symbols must be saved before netlisting and simulating. If not, some of the SaberDesigner symbols, such as supplies, may netlist incorrectly. (18627)

SaberSketch

59. Do Not Use Spaces in Symbol Port Names

If a symbol port name has a space in it, a stack trace occurs when ending the wire connection at that port. Also, a stack trace occurs if you bring up the Port Property form at that port. To avoid problems, do not use spaces in the port name. If you wish to show a port with a specific name containing a space, use a port property instead. Turn off the port name visibility in the attribute form, then create a port property with the specific name. Then, turn on the port property and place it as desired. This will prevent the occurrence of a stack trace while making connections to the port or when bringing up the port property form.

60. The Aim Macro Recorder will not work correctly if a subdirectory exists beneath the .aimMacro directory

The Aim Macro Recorder will not work correctly if a subdirectory exists beneath the `.aimMacro` directory. This applies to Windows NT platforms. In such a case, you will see a Tcl/Tk error.

The workaround is to move the subdirectory to another location outside of the `.aimMacro` directory. (17578)

61. Initial Display of Probe Signals is Zoomed-in

When a probe is placed on a node in Sketch, the signal on that node is displayed in a probe panel. The initial display of the node signal is zoomed-in so that only a limited portion of the x-axis is visible. To see the full signal, use the right mouse button to select Zoom to Fit. (26052), (26503)

62. Saving SaberSketch schematics created in Release 5.2

SaberSketch schematics containing busses which are created in Release 5.2 format cannot be saved in 5.0 format. When an attempt is made to save a 5.2 schematic in 5.0 format, an error message is generated stating that border information will not be saved. (25120)

63. SaberSketch symbols may display inconsistent fonts

HP laser printers

It is possible that your printer has installed FontSmart fonts onto your computer via your network that are incompatible with SaberSketch fonts (in particular, Arial).

The workaround is to open the control panel, select the font folder, select Arial, then view the properties. This action will activate the font for programs that might not otherwise recognize and use them (19913).

64. In SaberSketch, when Windows NT or 2000 is used as an X-terminal with Exceed, the text associated with a symbol does not scale relative to the symbol graphics

Windows NT or Windows 2000

When Windows NT or 2000 is used as an X-terminal with Exceed the text associated with a symbol does not scale relative to the symbol graphics. Text may overlap other text or graphics.

The workaround is as follows:

1. Start Exceed.
2. On the Exceed icon in the task bar, display the right mouse button menu. Choose **Tools->Configuration...**
3. On the form that appears, double click on the **Font** icon.

4. In the resulting form, click on **Font Database...** and at the bottom of that form, turn off the **Automatic Font Substitution** option. (17940)

SaberScope

65. When highlighting a signal in SaberScope, the signal disappears from the graph and shows up outside of the SaberScope window.

This is a video card related problem on Windows NT and Windows 2000.

The workaround is to change your display settings to the following:

```
Color palette=256 colors

Font size=large fonts

Frequency=70 Hertz

Desktop area=1280x1024
```
(13023)

66. When SaberScope is running in the background (invoked with “scope &”), exiting MATLAB can “hang” SaberScope

The workaround if you wish to run SaberScope in the background is to redirect output to a file. For example, type:

```
scope >& scope.out &
```

67. Setting Matlab format to compact

If the Matlab format 'compact' is set, SaberLink data translations from Matlab to Aim will not be performed properly. This happens because setting Matlab format to compact changes the format of the output of various Matlab commands, including those used to accomplish the transfer of data from Matlab to Aim applications. As a result, the transfer of data from Matlab to Aim is misunderstood as having failed. A select and paste data transfer operation is instead understood as a string paste operation. (Data translations from Aim to Matlab are not affected; theses transfers proceed as expected.)

WORKAROUND:

To re-enable the correct operation of SaberLink data translation, enter the following command in the SaberLink Matlab window:

```
format loose
```

This restores the default output mode for Matlab and allows the SaberLink translations to function properly. (19391)

68. Analysis in the Experiment Editor

The Saber analysis forms have a field to specify to load a plotfile into SaberScope after an analysis. However, this does not work when the analysis is inserted in the Experiment Editor. To achieve this functionality in an experiment, create a 'plot' item in the experiment by selecting "**PostProcessing > View Plot Files in Scope**". Also, you may select the item 'ScopePlot' for more advanced plotting of simulation results. (27703)

69. Fonts are corrupted in SaberScope on systems with the Matrox Millennium graphics card

Fonts are corrupted in SaberScope on systems with the Matrox Millennium graphics card. The specifications for the problem chip set and drivers are:

```
Matrox Graphics Version 2.20 4.0.0

MGA Millennium (String)

Memory 2 MB

MGA-2064WB2R2 (Chip Set)

TI TVP3026 @175MHz (DAC)
```

(13748)

70. UNIX plot file data may be corrupted on Windows NT and 2000 machines

If SaberScope cannot read a plot file that was created on a UNIX machine, or if the plot file data produces erroneous results or crashes SaberScope, you may need to alter the way you created the original file.

Depending on how you got the UNIX file over to the Windows NT or 2000 machine, try one of the following solutions:

1. Using WinZip to untar the UNIX plot files.

   If you are using a DOS floppy disk to transfer plot files, the file names must be in standard DOS 8 by 3 format (eight characters, a period, then a three character extension).

   - When in WinZip, select the **Options > Configuration** menu item. You should find a selection titled:

   `TAR File Smart CR/LF Conversion`

   This option is On by default (an x appears in the box next to this option).

   - Click on the box to turn the x off. Your data should now be converted correctly.

2. Using FTP to transfer the UNIX plot files:

- If your Windows NT or Windows 2000 machine is attached to the same network as your UNIX system, you can use FTP to transfer files. When using FTP, be aware of options to transfer binary files, which you may need to set when transferring data files. This depends on whether your FTP program does an "auto detect" of data type. (20694)

iQBus

71. Saving changes to a design without a project file returns an error message

If the design you are simulating does not have a project file (*design_name*.ai_prj), then changing any of the settings under **Edit>Saber/ Netlister Settings** (Output Filename, Additional Options, etc.) will result in the error message "Cannot write null-named file."

The workaround is to create an empty file named *design_name*.ai_prj in the same directory where the design file *design_name*.sin is located. (25860)

72. Save As... results in Stack trace failure

The Save As feature will fail with a stack trace when using a relative pathname in the form. Use the file browser form or type an absolute pathname in the 'Save iQBus Project As' form. If the stack trace does occur, it is necessary to close all active designs before using the Save As feature again. (28111)

SaberHarness

73. Do Not Use Spaces in Symbol Port Names

If a symbol port name has a space in it, a stack trace occurs when ending the wire connection at that port. Also, a stack trace occurs if you bring up the Port Property form at that port. To avoid problems, do not use spaces in the port name. If you wish to show a port with a specific name containing a space, use a port property instead. Turn off the port name visibility in the attribute form, then create a port property with the specific name. Then, turn on the port property and place it as desired. This will prevent the occurrence of a stack trace while making connections to the port or when bringing up the port property form.

74. Version Control and Generated Drawings

Issue: Version Control error message will be presented to the user if multiple Generated Drawings use the same named assembly.

It is recommended that if a user plans to use Version Control that they restrict the use of an assembly name to a single Generated Drawing.

If multiple Generated Drawings use the same assembly name the user will be presented with the following error message upon Check in, or Update:

`Merge Error: The assembly specification` *assembly_name* `for the harness` *harness_name* `is already in use. Used by generated drawing Name:` *generated_drawing_name* `File name:` *generated_drawing_filename*. `Not adding this generated drawing.`

To correct this error, a user must delete the generated drawing that uses the same named assembly.

To prevent this problem the user should create a separate assembly name for each generate drawing.

Example:

```
Assembly1_Front - Used by the bundle drawing
Assembly1_Front_Conn - Assembly used for the connectivity drawing
Assembly1_Front_Gnd - Assembly used for the ground distribution drawing
Assembly1_Front_Pwr  - Assembly used for the power distribution drawing
```

75. Transient windows hide behind main window in SaberHarness

If you have problems in the Common Desktop Environment (CDE) with windows that are supposed to be transient hiding behind the main window, go to **Programs > Style Manager > Window**, and unselect the checkbutton that says **Allow Primary Windows on Top**. (24267)

76. Append description value to ref value during Catia generation

You may implement customizing to support the appending of the description value to the ref value during Catia generation. This is controlled by setting:

```
set CATIAExp(AddDes2Ref) 1
```

in the harness rc file. By default this is 0 so the ref does NOT have the description appended. The description, when appended, has carriage returns replaced with spaces. (28086)

Cadence Frameway

77. If symbol property values are not listed in the tables section, the netlister fails

The netlister checks to see if values listed in the `tables` section of the mapping file are present before mapping the schematic. If there are symbol property values that are not listed in the `tables` section, the netlister fails for

all Frameways Integrations. Even if these values are not used by an instance of the symbol, they must appear in the `tables` section of the mapping file.

Each table should have a wildcard entry [*], which will match if none of the other entries match.

As a workaround, you can either map the required values that are not used by the instance to something that is obviously wrong, or keep them unchanged. For example, if a symbol has a property that needs to be `0` or `1` in the simulation but you enter the values `ZERO` or `ONE`, then the netlister would fail on the following table entries that map this property:

```
zero_one[ZERO]->0,[ONE]->1
```

A successful alternative is to use this table (keep all code on one line):

```
zero_one[ZERO]->0,[ONE]->1,[*]->ThisPropertyMustHaveAValue
```

To keep values other than `ZERO` or `ONE` unchanged, use this table:

```
zero_one[ZERO]->0,[ONE]->1,[*]->*
```

This problem applies to the Frameway for Cadence, Frameway for Innoveda (formerly Viewlogic), Frameway for Mentor Graphics, and the Frameway for EDIF. (4781)

78. The netlister fails to insert power pins when inserting a technology-specific Hypermodel interface

The workaround is to make a default entry in the `.shm` file being used. For example, in *install_home*`/template/hypermod/mhc.shm`, the default entry is the first entry after `"pins {"`. This entry refers to model `mhc` defined at the top of the file.

This problem applies to the Frameway for Cadence, Frameway for Mentor Graphics, and the Frameway for EDIF. (7527)

79. Constant identifiers are causing errors from the simulator

Constant identifiers are defined in an include file, but when they are used for a parameter to an instance in the Saber netlist, the netlister inserts external statements for them, causing errors from the simulator. This is caused by the netlister's hierarchical parameter passing feature being applied when it isn't expected.

This problem applies to all Frameways Integrations. There are two solutions:

1. This problem only occurs when the value of a template parameter is a simple constant (that is, with no arithmetic performed on it). Making the constant a part of a formula will solve the problem. For example, if the problematic parameter value is `dmax`, change it to `dmax+0` in the design.

2. To avoid changing the design, you can prevent the parameter passing with a mapping file. To do so, create a specific entry in the undetermined section of a mapping file. Field 1 of the entry should contain the name of the hierarchical cell where the external statements are being inserted. Field 18 should contain a comma-separated list of the parameter names which *should not be* added to this hierarchical cell. (15255)

80. Verilog text view names in the Frameway for Cadence

When creating Verilog text views in the Frameway for Cadence (view type `text.v`), a view name other than `verilog`, `abel`, `pld_verilog` and `minc` should be used. If any of these four view names are used, the resulting view that is created will not be of type `cdba` and will not be treated as a valid stopping view by the netlister. This is a Cadence bug (#270819, #236729, #266748) and is scheduled to be fixed in Cadence release 4.4.6. (21598)

81. Problem with nets written into the Verilog portion of the design in the Frameway for Cadence

The **Show Selected** feature does not work for nets written into the Verilog portion of the design. There is no workaround for this problem. (4948)

82. Cross-probing is not functional for signals generated outside the schematic from which netlists are generated in the Frameway for Cadence

Cross-probing is not functional for signals generated outside the schematic from which netlists are generated. For example, Verilog signals cannot be cross-probed.

The workaround is to include stimulus signals to be cross-probed by incorporating them into the schematic or by editing the `verilog.ntf` file or the netlist. Or, view these signals by selecting them from the signal list displayed in SaberScope. (14730)

83. Support for Inherited Connections

To support this feature, the Cadence SKILL portion of the netlister has been changed to run in incremental mode.

Note that the SKILL netlister can run in both incremental and non incremental mode. If the user does not want the SKILL netlister in incremental mode, then he/she needs to set `simNotIncremental = t` in their `.simrc` file. The default will be incremental mode (i.e. `simNotIncremental = nil`).

Also, anytime the CDF properties for a cell has been changed, or the design has changed, the old run directories need to be purged, `simReNetlistAll` variable needs to be set to `t in .simrc` file to generate a correct netlist. There is no workaround. (25758)

Model Architect Tools

84. Stability Path Setting for Model Architect Tools

Double clicking on a symbol associated with a model created by a characterization tool brings a model selection form that contains models of the same type found along the SABER_DATA_PATH. Adding a directory to the SABER_DATA_PATH should be done outside the application by editing the SABER_DATA_PATH environment variable. Using the **Search Path** tab form from the **Edit>"Saber/Netlister Settings...**" menu will not work.

Model Libraries

85. dp1 Template Problems

The template description for **dp1** (Diode Model with Reverse Recovery) has known incorrect entries in the Example Input fields. The values in the Example Input fields for the Symbol Properties **ifo**, **dirdt**, **irrm**, **trr**, and **qrr** are incorrect. These fields must only be populated with a number and a magnitude modifier, if required.

86. Stability Path Setting for Model Architect Tools

Double clicking on a symbol associated with a model created by a characterization tool brings a model selection form that contains models of the same type found along the SABER_DATA_PATH. Adding a directory to the SABER_DATA_PATH should be done outside the application by editing the SABER_DATA_PATH environment variable. Using the Search Path tab form from the **Edit>"Saber/Netlister Settings...**" menu will not work.

87. Changes to lead-acid battery template batt_pb_1

Changes have been made to the lead acid battery template, batt_pb_1, to improve the convergence of the model, and to give more realistic behavior compared to automotive SLI batteries at default parameter settings. The parameter defaults that have been changed are fc (was 0.1, is now 0.3) and rnom (was 0, is now 1m.) Instances of batt_pb_1 in existing designs that used the default values of fc and rnom will change their behavior, generally showing higher capacity at higher rate discharges than before. The behavior of the template may be restored to that of Release 5.2.2 and earlier by explicitly setting fc=0.1 and rnom=0.

88. mbsin3v3_h template to be discontinued

mbsim3v3_h is the BSIM3v3 model with enhancements to provide a close match with HSpice level 49 (version=3.0). Avant! more recently implemented in HSpice enhancements to Saber models mbsim3v3_1 and mbsim3v3_2.

mbsim3v3_1 and mbsim3v3_2 yield excellent HSpice compatibility with level's 49 and 53, with the exception of an optional charge model available in level 49. Also, the HSpice level 49 MOS model has been ported to Saber (hspm49) and is the recommended replacement for customers requiring direct HSpice compatibility for the BSIM3v3 series.

As a result of these new models/enhancements, Avant! will be phasing out mbsimv3_h. The symbols will be removed from Sketch in this release followed by the eventual elimination of mbsim3v3_h.

89. Models of hydraulic cylinder: actr_d,actr_e, and actr_r have been obsoleted and replaced

3 existing models of a hydraulic cylinder: actr_d,actr_e, and actr_r did not account for important cylinder features like initial piston position, dead volume, and cylinder orientation. For these reasons, these 3 models were obsoleted and replaced with 5 new models: cyl_h_da, cyl_h_da_id, cyl_h_da_2, cyl_h_sa, cyl_h_sa_id. (18245)

90. Default value of f1 and sr in the `op1type.sin` file has been changed

The default value of a, f1 and sr in the op1type.sin has been changed to the following:

```
a = 100k, f1 = 100meg.
```

This may affect the simulation results of any design that op1 models used default value defined in op1type.sin. To obtain the original simulation results, the user may restore the original default value by inserting the following statement in the netlist:

```
op1type..model  origin_model = (a=10k, f1=undef)
```

and assign origin_model to the op1 instances in the netlist that use default model value.

For example, if the original entry in a netlist was

```
op1.op1_ref ......
```

that there is no model assignment. Add a new line at the beginning of the netlist

```
op1type..model  origin_model = (a=10k, f1=undef)
```

and change the entry mentioned above to

```
op1.op1_ref ..... model = origin_model
```

In this way, the simulation results will be identical to the previous. (26316)

91. Template gas_vol replace with the accum_gas_1 model

Template gas_vol did not account for an energy associated with the change of internal energy which is an essential portion of the volume energy balance. For this reason, this template has been replaced with the accum_gas_1 model. The templates gas_vol and based on it accumulator model accum_g are obsoleted and not recommended for future use.

Template fluid_cham was developed as a building block for various configurations of hydraulic cylinders. But such important cylinder parameters as: piston initial position, dead volume, and cylinder orientation were not included in the model. For these reasons, 3 new models of an actuating chamber were developed and included in the Release 2001.2: chamb_qact_0, chamb_act_1, and chamb_act_2.

Template ft_elem was supposed to be a building block for pipe line models, but was based on an inaccurate approximation of the Reynolds Number. Therefore it was replaced in the Release 2001.2 with pipe_ss_0 module which uses considerably more accurate Haaland formulae. The models of a pipeline line_f and line_r which are based on the ft_elem module are also obsoleted and replaced with more accurate models pipe_ss_1 and pipe_ms_1. (17382)

92. Generic diode template **dp** may give erroneous results

The generic diode template dp may give erroneous results under the following conditions:

1. The BV parameter is set to 1000 or greater.
2. Default simulator settings are used.

To overcome this problem the DENsity simulator option should be set to 20 or higher (default is 1).

This problem will be addressed in a subsequent release. (25174)

93. Comparator components in the component library may not work appropriately

It is a known problem that the comparator components in the component library using **cmp1** or **cmp2** building blocks may not work appropriately under all conditions. That is why we developed the **cmp3** model and upgraded all comparator models to level 3 which can be identified by *_3 in the part names.

We strongly suggest using level 3 comparator models in the design. Level 1 or level 2 comparator models are obsolete. They are there for backward compatibility. (27858)

94. Hspice diode model

There is an error in the `Hspice` diode model, `hspd.sin`, in v5.2 and v5.2.update. The thermal voltage used to compute or scale some model parameters contains the wrong temperature: `25C` instead of the instance temperature. This results in the wrong output when `tref=temp` except when `tref=25`. (27402)

95. Control option parameter VNTOL in Hspice

The control option parameter `VNTOL` in Hspice 'sets the absolute minimum voltage for DC and transient analysis'. In some cases `VNTOL` also serves as an offset to avoid numerical divergence of some scaled model or internal parameters. Relative to the former usage, there is no correspondence with Saber and `VNTOL` had been set to 0 in the ports of MOS level 28 and level 49 (hspm28 and hspm49). Under very rare circumstances this could have caused numerical divergence of some parameters. To correct this, `VNTOL` has now been modified to Hspice's default value of 50uv. As now implemented, the user cannot modify this term, though there should be rarely if ever the requirement to do so. In any event, it is planned in a future release to make this parameter a term the user can modify. (27409)

96. Model parameter “level” in hspm28.sin compatible with Hspice

Though the model parameter 'level' in hspm28.sin is effectively a dummy parameter for compatibility with Hspice, it does serve as a placeholder in the model code. Spurious output will result if it is not set per: level=28. It currently has no default value. It must either be specified in the model card (i.e. model library) used with a netlist or it can be permanently set it the model structure in `$SABER_HOME/bin/template/hspice/hspm28.sin`. This will be rectified in the next release. (26303)

97. Older designs may have symbols slightly out of position

If user opens an old design with one of the following symbols in it, the symbol may have moved a bit from its old position. All of the symbols relocate about +/ -0.05 inches.

batt_pb_1, boostavg, buckavg, elm_da_dv1, elm_pv1, elm_sa_dv1, frwdavg, op3d, orif_var1, p_amp01, p_cmp06, p_dac16, p_dac17, p_driver02, p_mult01, p_mux05, p_mux06, p_mux08, p_sh01, p_vreg12, p_vreg14, sqz_cyl1, sqz_cyl2, t_snk_gem1, v_2way_eo1, v_2way_rh, v_3way_mo1, v_4way_mo1, var2i, var2mmf, var2th_pwr, var2v, var2w_radps (18649)

98. On HP Systems, placing Switch Components into a schematic can cause the operating system to log out the user

Hewlett-Packard systems

On HP9000 with PA-RISC2.0 and the 10.2 operating system, placing Switch

Components from the Parts Gallery into a schematic can cause the HP operating system to log out the user.

The workaround is to obtain patch PHSS_11398, titled "S700-800 10.20 X Server Cumulative Patch" from HP and add it to your operating system. This patch may be superseded by a later patch. (14313)

99. Symbols with the same name in different libraries are no longer confused

In prior releases, the netlister would search all library sections of a mapping file for a matching symbol name. This sometimes caused incorrect mappings if there were two symbols with the same name from different libraries. This has been changed so that the library name of a symbol must match field 1 of the generic section for the mapping to be used. You can still map all symbols with the same name (regardless of what library they are in) by leaving field 1 of the generic section empty. Most mapping files should not need to be updated. However, if some mappings are not performed in Release 5.0, the mapping files may need to be fixed.

In SaberSketch, the library name should always be empty, because symbols don't have libraries.

In the Frameway for Innoveda (formerly Viewlogic), the library name should always be empty, because the library name is a part of the symbol name, such as `SBR_ELECTRIC:r`.

In the Frameway for Cadence, the library name in the mapping file is the same as the library the symbol is in. For example, all provided symbols are in `SaberLib`.

In the Frameway for Mentor Graphics, the library name is the leaf name of the directory the component which contains the symbol is in, such as `sbr_electric_lib` or `sbr_hydraulc_lib`. (11428)

100. A compound statement in a template cannot have an empty body

A compound statement in a template cannot have an empty body. For example, the following is not allowed:

```
values {
# comment
}
```

There is no workaround for this problem. (8980)

101. A foreign routine in a template that does not return a value can cause errors

If you include a foreign routine in a template and it does not return a value, it can cause excessive simulation time or other unpredictable results with no error reported.

The workaround is to make sure any foreign routine returns a value even if it is a “dummy” value. (10118)

102.You cannot use an enumerated type parameter (`enum`) as an external variable in a template

The workaround is to use an argdef (..) operator to pass the `enum` in from another template. (10272)

103.Duty cycle for `boostavg_cd.sin` model.

The problem was caused when the duty cycle control voltage at the `vc` input of the `boostavg_cd` template is zero, i.e. `d1 = 0`, that the model becomes singular in the sense that there is no energy conversion from the input to the output. In reality, the boost circuit can never reach a desired solution when `duty cycle = 0`. As a practical restriction, `d1` is limited to 0.25m, internally. (27102)

Online Documentation

104.Subscripted text in SaberBook is not being displayed correctly

Subscripted text in SaberBook is not being displayed correctly. Subscripts appear too high relative to the normal text line, approximately centered vertically.

There is no workaround. Subscripts are not supported by the Microsoft WinHelp viewer used by SaberBook on Windows 2000 systems. (8421)

105.SaberBook Online Documentation

During installation there is an option to install SaberBook online documentation. An alternative is to run the documentation from the CD, however for Windows systems, this will result in a error. To access the help system, it must be installed with your software.

106.When using the Find search option, expected topic not displayed

UNIX system

When using the Find search option, the expected topic is not displayed or a message is displayed stating that the topic is not found.

The workaround is to delete any `*.fts` files found in the directory `.hh` in your home directory. These files override the `*.fts` files provided with SaberBook and may be outdated. When they are removed, SaberBook will then use the `*.fts` files provided with the current release of the SaberDesigner software. (12098)

107.X-Terminal display options mis-labeled

Windows NT and Windows 2000 Systems used as x-terminals

In the Find->Options dialog box, the label for the panel containing the top set of options displays as **Label 1** rather than **Search for topics containing**. (12138)

There is no workaround for this problem.

108.SaberBook window is too small

If your system setting for Desktop Area is not at least 1152x882, you may need to enlarge the SaberBook Help window to enable you to view the entire line of text.

The workaround is to set the Desktop Area system setting to at least 1152x882 if possible. (15013)

109.Fatal error message on SaberBook start-up

UNIX system

The following message is displayed when SaberBook is invoked:

```
ld.so.1: hyperhelp: fatal: libXm.so.3: can't open file:errnon=2
```

The workaround is that SaberBook requires the Motif shared libraries provided with the X11R5/Motif 1.2 windowing environment. This is a part of the standard installation for all UNIX operating systems on which SaberDesigner software is supported. If you see this error message, these libraries may have been moved or deleted from your system. Replace these libraries in their standard location. (17817)

110.Color warning messages on SaberBook start-up

UNIX system

If you run multiple color intensive applications and invoke SaberBook in stand-alone mode by specifying the following command in a shell:

```
saberbook
```

You may get the following warning messages:

```
Warning: Cannot allocate colormap entry for '#bebebe'
Warning: Cannot allocate colormap entry for default background
Warning: Cannot allocate colormap entry for '#6e6e6e
```

If you get similar warning messages, then the colormap of SaberBook will be affected. You might see strange color mapping issues (such as the hypertext links appearing black instead of the typical green).

The workaround if you have color map problems is to try invoking SaberBook from within a SaberDesigner application instead of stand-alone. This process allows the two applications to share colors instead of competing for the colors. (17928)

111.A Help window appears but has no text in it

The following sequence of actions may cause the **Help** window to appear without text in it:

- Bring up the Help Topics window by selecting **Help > SaberBook** from within a SaberDesigner application (DO NOT select a topic).
- Select context-sensitive help from within a SaberDesigner application (such as help on the properties of a symbol, help on the data in a template, help on any of the application forms, or help on the application)
- A blank Help window will appear.

The workaround to avoid problems is to always display a help topic after you bring up the Help Topics window.

If a blank Help topic window is displayed, do one of the following:

- Always display a help topic when you bring up the Help Topics window.
- Exit out of the blank window and then re-invoke the context-sensitive help.

or

- Click on the Help Topics button at the top of the blank Help window.

  In the Help Topics window bring up a topic from the Contents, Index, or Find tabs. (19359)

112.Warning message on the Solaris platform

When you start SaberBook on the Solaris platform, you may get the following message:

```
Warning: Can't load codeset file 'C', using internal fallback
```

This message is not generated by SaberBook or the Bristol HyperHelp application on which SaberBook is based. You will get this warning whenever you try to run any Motif application, such as an xterm, if your environment is not configured appropriately. To get rid of the message, set the `XLIBI18N_PATH` environment variable to *install_home*`/lib`. (14011)

113.Subscripted text in SaberBook is not being displayed correctly

Windows NT and Windows 2000 systems

Subscripted text in SaberBook is not being displayed correctly. Subscripts appear too high relative to the normal text line, approximately centered vertically.

There is no workaround. Subscripts are not supported by the Microsoft WinHelp viewer used by SaberBook on Windows NT and 2000 systems. (8421)

114. SaberBook closes but leaves a stray HyperHelp background process running

UNIX system

If you invoke SaberBook and click on the **Cancel** button in the Help Topics window without ever viewing a topic, SaberBook closes but leaves a stray HyperHelp background process running. However, when you log off, all the stray HyperHelp background processes stop.

There is no workaround for this problem. (11524)

115. A multiple word search in the SaberBook Find tool operates differently in UNIX, Windows NT and Windows 2000

In Release 5.1 and later, the Find tool in SaberBook has changed the Find Option to "All the words you typed in any order." When you type multiple words in field 1 of the Find tool, the topics associated with those words appear in field 3 at the bottom of the Find dialog box.

In UNIX, a list of words may or may not appear in field 2. Selecting a word from this list will not display a topic in field 3. Ignore field 2 when performing a multiple word search.

In Windows NT and 2000, a list of words will appear in field 2. Selecting a word will change the list of topics in field 3. (24597, 24704)

Documentation Available with this Release

The following reference material is provided with this release:

116.After Installation

Directory	File Name	Document Title	Doc ID	Description	Size
	SaberDesigner Install_UNIX. pdf	Installation and Configuration Guide, SaberDesigner 2001.2, UNIX	ISDU2001.2	Instructions for UNIX systems	58 pgs
	SaberDesigner Install_Win.pdf	Installation and Configuration Guide, SaberDesigner 2001.2, Windows NT and 2000	ISDN2001.2	Instructions for Windows NT and 2000 systems	184 pgs
	SaberDesignerRel Notes.pdf	SaberDesigner Release Notes, Release 2001.2	ARN2001.2	Selected release information for SaberDesigner, Release 2001.2	62 pgs
eng_notes	Calibration HOWTO.pdf	Calibration HOWTO, Engineering Note No. 130	EN130-1	Background on the calibration utility in the Saber simulator.	14 pgs
eng_notes	conversion. aim	AIM script, no document title	-	AIM script for converting older symbol files to the newer format. See the companion document conversonREADME.wri	-
eng_notes	conversion README	-	-	Describes the conversion script (conversion.aim, also included) that converts older symbol files to the newer format (For UNIX platforms)	4 KB
eng_notes	conversion README.wri	-	-	Describes the conversion script (conversion.aim, also included) that converts older symbol files to the newer format (For Windows platforms)	4 KB

eng_notes	en127-4.pdf	Engineering Note No. 127: External Waveform Database API	EN127-4	Describes the External Waveform Database API for creating dynamically loadable database access packages (dll) for reading non-SaberDesigner data formats into SaberScope	89 KB, 14 pgs
eng_notes	en128-3.pdf	Engineering Note No. 128: Library Encryption Tool	EN128-2	Describes the ai-encryp program provided with the Library Encryption tool for encrypting templates	36 KB, 3 pgs
pdf_docs	iQBusMan.pdf	IQBus Reference Manual	ADI2001.2	Introduction to iQBus 2001.2	68 pgs
pdf_docs	HarnessGS.pdf	Getting Started with Saber Harness	GSH2001.2	Simple, step-by-step first-time tutorial for SaberHarness.	44 pgs
pdf_docs	HarnessRef.pdf	SaberHarness Reference Manual	ADH2001.2	Users' and Reference manual for SaberHarness.	232 pgs
pdf_docs	SaberDesignerGS.pdf	Getting Started with SaberDesigner	GSS51-1	Simple, step-by-step first-time tutorial for SaberDesigner.	46 pgs
pdf_docs	SaberDesignerAD.pdf	Analyzing Designs Using SaberDesigner	ADS51-1	Tutorial for the experienced user on how to analyze a design using real-world examples including a brake system and power converter.	250 pgs
pdf_docs	SDDesign Examples.pdf	SaberDesigner Design Examples	XG51-1	Tutorial for the experienced user on how to analyze a design using real-world examples including a brake system and an audio system.	192 pgs
pdf_docs	AIM.pdf	-	AIMR2001.2	Alphabetical listing of all AIM commands	5.292 MB, 1,450 pgs
pdf_docs	power_tools.ppt	-	-	Windows Power Point presentation on the iQBus	-
TclTk_docs	README.wri	-	-	Instructions on how to access the TclTk documentation in HTML form	-

Supplement to Released Documentation

New Simulator Commands

117.New IP_Report Command

A new command has been implemented in the Saber Simulator to report initial point data of transistor circuits. The command creates a two part report of the initial point data in a form similar to HSPICE. The first part includes the values of all selected nets. The second part includes information about selected instances. Either part can be selected or omitted. The command requires the use of cooperating templates in your design. It has the following format:

IP_Report [name] [(variable value, ...)]

where:
IP_Report the name of the command

name	the name of an initial point. Default: DC
variable value	a command variable and its value.

Available variables, their meanings and their defaults are:

variable meaning	default
REPort	An indication of which parts of an ALL initial report to print. ALL, NETs, or INSTances
NETs	The list of nets to include in the net /.../* report. The value can be a list of system variables (pins and vars/refs) and event-driven states
NETClass	The class of the nets to include in Unique Nodes States Vars the net report. Used in the interpretation of * in the NETs variable. Can be any combination of:

	Unique Nodes States Vars (vars/refs) If Unique is specified, each net selected by the NETs variable is listed once using the name of the net highest in the hierarchy; otherwise, all aliases for the net at different levels of the hierarchy are listed
SIGList	Specifies the instances and the dynamic /.../hsp variables (i.e. nodes, vars/refs, vals, states) of each instance to include in the instance report
PINs	The kind of information to print if THROUGH the SIGList selects a pin. One of ACROSS, THROUGH, or ALL
OFile	The name of the file where a copy of_ (means to write to the printed output is stored transcript)
FORMat	The format in which numbers are SCALEfactor printed. One of SCALEfactor or EXPonential
SIGDigits	The number of significant digits in a 4 number

118.Enhanced DIsplay Command

The DIsplay command has been enhanced with a new command variable SIGClass. The purpose of this variable is to specify the meaning of the * in a SIGList. The value of the variable can be any combination of the identifiers in the first column of the following table:

Unique	if specified, each net specified by the SIGList variable is listed once using the name of the net highest in the hierarchy; otherwise, all aliases for the net at different levels of hierarchy are listed
Nodes	include nodes
Estates	include event-driven states
Astates	include assigned states
Vars	include vars and refs
Locals	include vals

States	shorthand for Astates Estates

The default for the SIGClass variable is Nodes States Vars Locals.

119.Enhanced List Command

The List command has been enhanced with two new options:

The command

List Instances <instance_path>

lists the instances nested underneath the instance specified by <instance_path> in a format that indicates the nesting level. If no <instance_path> is specified, the current working block (usually /) is used. Multiple <instance_paths> can be specified.

The command

List Nets

lists each net in the design, including all connection points connected to the net. Each net includes the kind of the net (node, var/ref, or state), and additionally the following information:

var/ref nets:	v	indicates a var connected to the net
	r	indicates a ref connected to the net
state nets:	d	indicates that this state drives the net (scheduled with schedule_event)
	o	indicates that this state is observed (appears in event_on())

The List Nets command can also be used in the following form:

List Nets <waveform_path>

where <waveform_path> specifies a pin, event-driven state, var or ref. For this form of the command the net to which the specified pin, state, or var/ref is connected is listed.

New Hot Keys

Hot Key	Function
`Ctrl+o`	Open design.
`Ctrl+a`	Open new graph window
`Ctrl+Del`	Close current graph window.
`Ctrl+1`	Iconify current graph window
`Ctrl+f`	Zoom to fit
`Ctrl+z`	Zoom in
`Ctrl+5`	Zoom out

Plotting simulation results in an Experiment using "ScopePlot"

To plot a specified list of signals from a single plotfile, open the ScopePlot form (by clicking on the ScopePlot button once it is placed in the experiment) and set "**Specify Scope Graph/Outline**" to "No". You can then select a plotfile and list of signals to plot. Setting the plotfile name to "Last" will select the most recent plotfile created by Saber (e.g. by the preceding analysis item in the experiment). You can select signals to display either by getting the current selected schematic signals, by browsing the list of signals in the Saber design, or by browsing an existing plotfile. It is important to make sure that the signals which are selected for the specified plotfile name will be available in the plotfile when the experiment is run. In other words, make sure that the signal list specified for the analysis that creates the plotfile contains the signals that you wish to plot.

The second option is to use Scope to create the desired graph and import the data necessary to recreate the graph into the experiment. The data is transferred through a Scope file known as an "`outline`". An outline saves all information needed to format signals and measurements in the current graph including those signals created with the Scope calculator. An outline may be used to create a new graph with a different set of signals, different plotfiles, and different simulation end time from those which were used in the original graph.

To use this option in an experiment, set "**Specify Scope Graph/Outline** to "Yes". You may then choose a Scope outline by selecting "**Get From Scope**" or "**Get From File**" using the down arrow next to the field named "Outline" in the ScopePlot form. "**Get From Scope**" will automatically import the data necessary for recreating the graph which is currently active in Scope. Once

again it is important to make sure that the signals and plotfiles will be available at the time that the ScopePlot is executed in the Experiment. By default, the signal and plotfile names are set the same as those used in the original Scope graph. You may select different signals or plotfiles by pressing mouse button one in the "Signals" field in the ScopePlot form.

The "**Get From File**" option is similar to "**Get From Scope**", but reads the outline data from a file instead of directly from Scope. To use "**Get From File**" you will need to select the item "**Save>Outline**" from within Scope to save an outline of the graph that you wish to create in the experiment. Then use the "**Get From File**" item in the ScopePlot form to read the outline into the experiment. When using "**Get From File**" the default signal names are the same as those in the original graph, and the default plotfile is set to "Last" (which selects the most recently generated plotfile). Once again you may modify the plotfiles and or signals by clicking on the signal field in the ScopePlot form.