Agilent Technologies

**Advanced Design System 2011**

**September 2011**
**Advanced Design System Quick Start**

**Acknowledgments**

Mentor Graphics is a trademark of Mentor Graphics Corporation in the U.S. and other countries. Mentor products and processes are registered trademarks of Mentor Graphics Corporation. * Calibre is a trademark of Mentor Graphics Corporation in the US and other countries. "Microsoft®, Windows®, MS Windows®, Windows NT®, Windows 2000® and Windows Internet Explorer® are U.S. registered trademarks of Microsoft Corporation. Pentium® is a U.S. registered trademark of Intel Corporation. PostScript® and Acrobat® are trademarks of Adobe Systems Incorporated. UNIX® is a registered trademark of the Open Group. Oracle and Java and registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. SystemC® is a registered trademark of Open SystemC Initiative, Inc. in the United States and other countries and is used with permission. MATLAB® is a U.S. registered trademark of The Math Works, Inc.. HiSIM2 source code, and all copyrights, trade secrets or other intellectual property rights in and to the source code in its entirety, is owned by Hiroshima University and STARC. FLEXlm is a trademark of Globetrotter Software, Incorporated. Layout Boolean Engine by Klaas Holwerda, v1.7 http://www.xs4all.nl/~kholwerd/bool.html . FreeType Project, Copyright (c) 1996-1999 by David Turner, Robert Wilhelm, and Werner Lemberg. QuestAgent search engine (c) 2000-2002, JObjects. Motif is a trademark of the Open Software Foundation. Netscape is a trademark of Netscape Communications Corporation. Netscape Portable Runtime (NSPR), Copyright (c) 1998-2003 The Mozilla Organization. A copy of the Mozilla Public License is at http://www.mozilla.org/MPL/ . FFTW, The Fastest Fourier Transform in the West, Copyright (c) 1997-1999 Massachusetts Institute of Technology. All rights reserved.

The following third-party libraries are used by the NlogN Momentum solver:

"This program includes Metis 4.0, Copyright © 1998, Regents of the University of Minnesota", http://www.cs.umn.edu/~metis , METIS was written by George Karypis (karypis@cs.umn.edu).

Intel@ Math Kernel Library, http://www.intel.com/software/products/mkl

SuperLU_MT version 2.0 - Copyright © 2003, The Regents of the University of California, through Lawrence Berkeley National Laboratory (subject to receipt of any required approvals from U.S. Dept. of Energy). All rights reserved. SuperLU Disclaimer: THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

7-zip - 7-Zip Copyright: Copyright (C) 1999-2009 Igor Pavlov. Licenses for files are: 7z.dll: GNU LGPL + unRAR restriction, All other files: GNU LGPL. 7-zip License: This library

is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful,but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA. unRAR copyright: The decompression engine for RAR archives was developed using source code of unRAR program.All copyrights to original unRAR code are owned by Alexander Roshal. unRAR License: The unRAR sources cannot be used to re-create the RAR compression algorithm, which is proprietary. Distribution of modified unRAR sources in separate form or as a part of other software is permitted, provided that it is clearly stated in the documentation and source comments that the code may not be used to develop a RAR (WinRAR) compatible archiver. 7-zip Availability: http://www.7-zip.org/

AMD Version 2.2 - AMD Notice: The AMD code was modified. Used by permission. AMD copyright: AMD Version 2.2, Copyright © 2007 by Timothy A. Davis, Patrick R. Amestoy, and Iain S. Duff. All Rights Reserved. AMD License: Your use or distribution of AMD or any modified version of AMD implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies.User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. AMD Availability: http://www.cise.ufl.edu/research/sparse/amd

UMFPACK 5.0.2 - UMFPACK Notice: The UMFPACK code was modified. Used by permission. UMFPACK Copyright: UMFPACK Copyright © 1995-2006 by Timothy A. Davis. All Rights Reserved. UMFPACK License: Your use or distribution of UMFPACK or any modified version of UMFPACK implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies. User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included. UMFPACK Availability: http://www.cise.ufl.edu/research/sparse/umfpack UMFPACK (including versions 2.2.1 and earlier, in FORTRAN) is available at

**Errata** The ADS product may contain references to "HP" or "HPEESOF" such as in file names and directory names. The business entity formerly known as "HP EEsof" is now part of Agilent Technologies and is known as "Agilent EEsof". To avoid broken functionality and to maintain backward compatibility for our customers, we did not change all the names and labels that contain "HP" or "HPEESOF" references.

**Warranty** The material contained in this document is provided "as is", and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this documentation and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

**Technology Licenses** The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license. Portions of this product include the SystemC software licensed under Open Source terms, which are available for download at http://systemc.org/ . This software is redistributed by Agilent. The Contributors of the SystemC software provide this software "as is" and offer no warranty of any kind, express or implied, including without limitation warranties or conditions or title and non-infringement, and implied warranties or conditions merchantability and fitness for a particular purpose. Contributors shall not be liable for any damages of any kind including without limitation direct, indirect, special, incidental and consequential damages, such as lost profits. Any provisions that differ from this disclaimer are offered by Agilent only.

**Restricted Rights Legend** U.S. Government Restricted Rights. Software and technical data rights granted to the federal government include only those rights customarily provided to end user customers. Agilent provides this customary commercial license in Software and technical data pursuant to FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for the Department of Defense, DFARS 252.227-7015 (Technical Data - Commercial Items) and DFARS 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation).

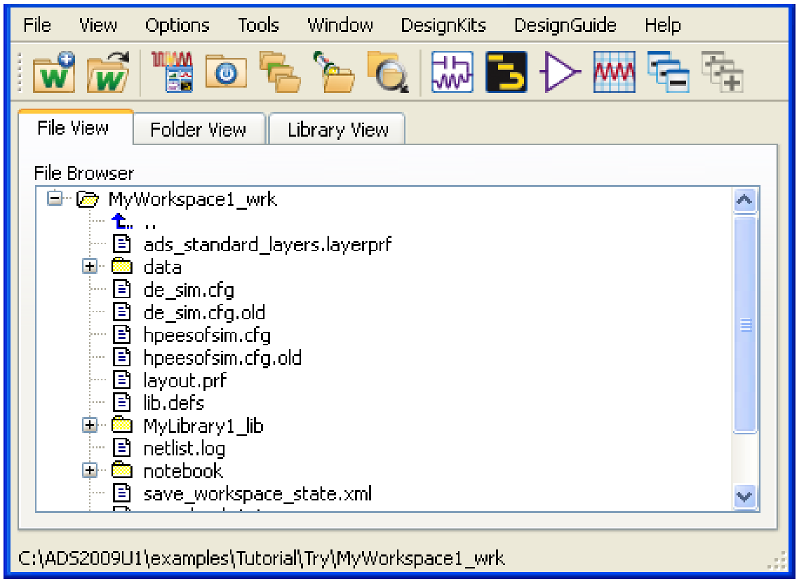
# ADS Design Environment

This section provides a quick introduction to the ADS Design Environment. The ADS design environment has multiple windows. The ADS Main Window (also referred to as Main Window) is your first interface to start using the ADS. This window allows you to access all the features supported by ADS.

## ADS Main Window

The ADS Main Window enables you to create and manage ADS workspaces. A *workspace* is an organizer where you can group everything about a design within— such as libraries, technology, schematic, layout, simulation data, and Momentum data. The entire ADS user interface and simulation operates within a currently opened workspace. The **File View** page of the Main window allows you to traverse to your existing workspaces (located in the *default* folder). To open any of these workspaces, select and right-click on the workspace and choose **Open Workspace** or double-click on the selected workspace.

**Note**
ADS allows you to open only one workspace for a given ADS instance.



Once a workspace is open, ADS automatically switches to the **Folder View** page where it displays all the contents of that workspace, and you can start designing schematic or layout, or perform tasks like simulation of the designs already created. From ADS Main Window, see the **File** menu options for more extensive workspace management commands.

**Note**
After opening a workspace, the toolbar buttons displayed in **File View** and **Folder View** or **Library View** are different.

From the ADS Main window you can:

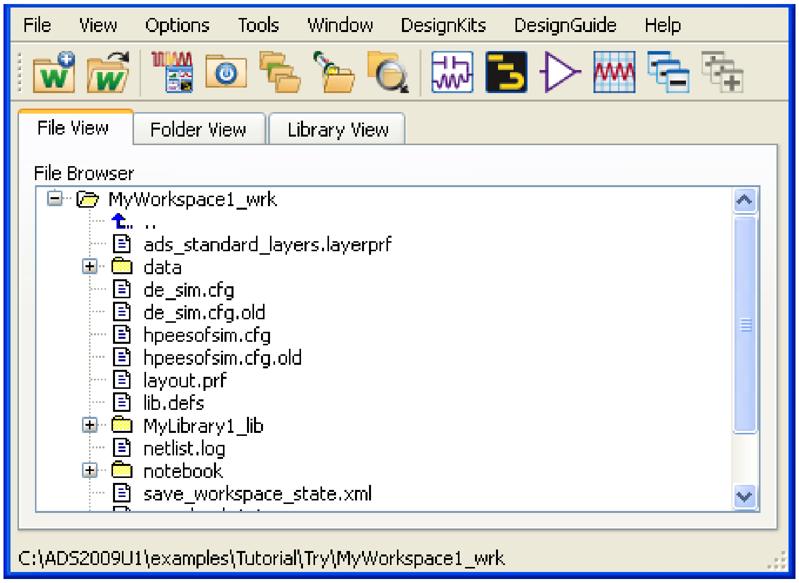
- Create or open a workspace, cell, library, view, substrate, and hierarchy policy
- Upgrade a project to a workspace (for ADS 2009 Update 1 and earlier users)
- Quickly open an example workspace (**File > Open > Example**)
- Archive (**File > Archive Workspace**) or Un-Archive (**File > Unarchive Workspace**

- **or Project**) workspace
- Set program preferences (**Option > Preferences**)
- Change toolbar configuration and keyboard shortcuts (**Tools > Hot Key/Toolbar Configuration**)
- Manage Technology associated with a workspace (**Options > Technology**)
- Record and play macro (from **Tools** menu option)
- Load AEL files/commands from the Command line (**Tools > Command Line**)
- Launch the text editor
- Open data display and Schematic window
- Show/Hide all windows
- Display all types of files and open as required using the context-sensitive menu

## Workspace View Options

From the ADS Main window, the view options are provided for a workspace in three different tabs.
These tabs show a different view of the workspace and the loaded libraries. Changes made in any of these tabs (such as renaming or deleting a file or view) will affect all three tabs.

## File View



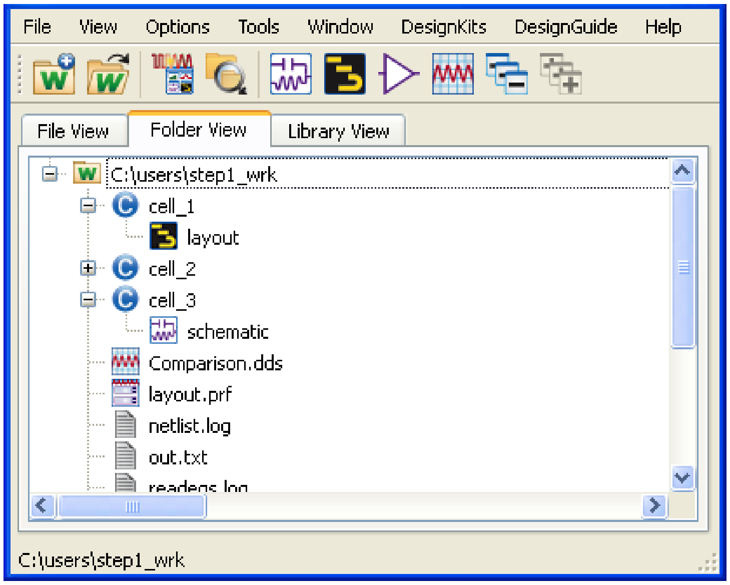
```
MyWorkspace1_wrk
  ..
  ads_standard_layers.layerprf
  data
  de_sim.cfg
  de_sim.cfg.old
  hpeesofsim.cfg
  hpeesofsim.cfg.old
  layout.prf
  lib.defs
  MyLibrary1_lib
  netlist.log
  notebook
  save_workspace_state.xml
```

C:\ADS2009U1\examples\Tutorial\Try\MyWorkspace1_wrk

In the File View, you can:

- Browse to other directories (similar to previous releases)
- See the actual files that are stored in the file system
- View special characters in cell names shown in parenthesis (special characters are used by OpenAccess for cross-platform support). This improves readability.

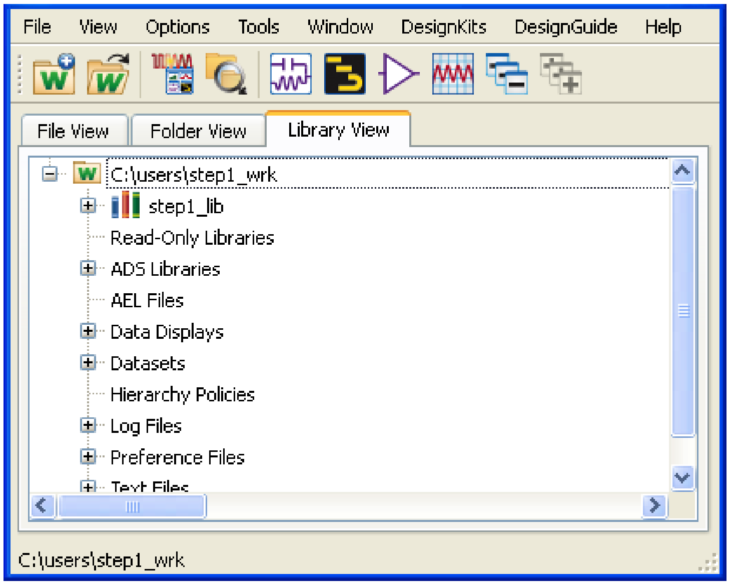
## Folder View



In the Folder View, you can:

- Create virtual folders to group related files (similar to previous releases)

## Library View



In the Library View, you can:

- View the system organization of a workspace
- Find files by type

> **Note**
> Some of the Main window menu and toolbar items will change as you move between Folder View and File View. For details, see Context Menu.

From ADS Main Window, you can create a new workspace or upgrade your ADS project (created using ADS 2009 Update 1 or earlier versions) to workspace. For more details on creating a new workspace, see *Using Workspace* (adstour). If you have been using ADS 2009 Update 1 or earlier version, you will have to upgrade you ADS Project to ADS Workspace. For more details, see *ADS Project Upgrade to ADS Workspace* (oaqkref).

**Note**
To know more about ADS workspaces, see *Workspace* (oaqkref).

## Context Menu

In Folder View each Cell, Library, Workspace, and file contain the context menu. These menus can be different for different file types and workspace. The table below lists the different context menus associated with design or file type. To access these menus, right-click on a file, design, workspace, or folder in the Folder View.

**File Types/Folders and Associated Context Menus**

| Context Menus | Designs | Data Display (*.dds) | Datasets (*.ds) | Workspace and Folders | Ael (*.ael) | Preferences (*.prf) | Layers (*.lay) | Text (*.txt) |
|---|---|---|---|---|---|---|---|---|
| Open (Schematic, Layout, or Symbol) | X | | | | | | | |
| Open Data Display | | X | | | | | | |
| Open in Text Editor | | | | | X | X | X | X |
| Create New Folder | | | | X | | | | |
| Load | | | | | X | | | |
| Copy | X | X | X | X† | X | X | X | X |
| Paste | | | | X | | | | |
| Copy Files | X | X | X | X | X | X | X | X |
| Rename | X | X | X | X | X | X | X | X |
| Delete | X | X | X | X†† | X | X | X | X |
| Filter View | X | X | X | X | X | X | X | X |
| Expand Items in Folder | X | X | X | X | X | X | X | X |
| Collapse Items in Folder | X | X | X | X | X | X | X | X |

† Group copy of files in folder. Invokes the copy/rename wizard.

†† Group delete of files in folder.

## Create New Folder

You can create folders in the Folder View using the context menu of any existing Workspace or folder.

To create a new folder from your Folder View,

1. Right-click on the Workspace or any folder and choose **New Folder** from the pop-up menu. The New Folder dialog box appears.
2. Enter a name for the new folder and click **OK** to create the folder or click **Cancel** to abort the operation.

## Dragging and Dropping Folders and Designs

The Folder View enables you to simply drag and drop a folder or design to a new location/folder.

To drag and drop a folder or design,

1. Locate the folder or design that you want to move in the Folder View.
2. Click and drag folder or design to the new location/folder.
3. Release the mouse button to drop the folder or design in the new location/folder. When you release the mouse button, pop-up menu appears with following options:
    - Move Tree Item: It allows you to drop the folder or design into the new location/folder.
    - Copy Files: For information on Copy Files, see Copying Files.

## Loading AEL Files

You can load Application Extension Language (AEL) files directly from the context menu of any AEL file (`*.ael`) in your Workspace.

To load an AEL file from your Folder View,

1. Ensure the `.ael` file that you want to load is displayed in your Folder View. If you cannot see the `.ael` file in your Folder View, you may need to set the Filter View option to include AEL files.
2. From your Folder View, right-click the `.ael` file that you want to load.
3. Choose **Load**, the AEL file is automatically loaded (executed).

> **Note**
> Only files in the top directory of your Workspace will show up in the Folder View. For more information on the AEL files, refer to the *AEL* (ael) documentation.

## Copy and Paste Files

The *Copy* context menu enables you to copy file(s) in the buffer. You can then use the *Paste* context menu to place a copy of the file(s). If you are pasting the file(s) into the same Workspace then the Copy Files dialog box will open and allow you to specify a different name for the file(s).

## Rename Files

The *Rename* context menu enables you to rename a file that you have selected. Simply right-click a file that you want to rename and choose Rename. Modify the filename and press Enter to change the name. If you are renaming a design, all other designs that reference this design will be modified to use the new design name.

## Copying Files

The Copy Files dialog box enables you to manage copy operations of a single file or multiple selected files. The Copy Files operation is accessible through the Context Menus and will also appear if you attempt to drag and drop a file into another Workspace or folder enabling you to select either *Move File* or *Copy File*.

To copy a file or group of files,

1. Select the file(s) in your Folder View.
2. Right-click and choose *Copy Files* from the context menu. The Copy Files dialog box appears.
3. Select *Include hierarchy* if you want to include sub-networks of the selected designs that you are going to copy.
4. Set your Destination by selecting a Workspace or Directory from the drop-down list, or click **Browse** to access a different Workspace or directory.
5. Click **Choose Folder** to select a different folder.

6. Use the *Auto Rename Rule* options to automatically rename your copied file(s) using specific criteria. The available options include:
   - Filename plus number incremented <filename>_v<number> - Copies the existing filename(s) and appends an _v plus a version number to the filename(s).
   - Filename plus maximum number incremented <filename>_v<number> - Copies the existing filename(s) and appends an _v and the highest version number in all current filename(s), plus one, to the new filename(s).
   - Number incremented plus filename v<number>_<filename> - Copies the existing filename(s) and pre-pends a v plus a version number and an underscore to the filename(s).
   - Maximum number incremented plus filename v<number>_<filename> - Copies the existing filename(s) and pre-pends a v and the highest version number in all current filename(s), plus one, and an underscore to the new filename(s).
   - None - Copies the file(s) using the existing filename(s).
7. After setting the Auto Rename Rule, the new name(s) can be modified manually in the *New Filename* field.
8. Verify that your new names are correct. The existing names (current files) will appear in the *Current Filename* field, while the new names (file copies) are displayed in the *New Filename* field.
9. Click **OK** to copy the file(s) and dismiss the dialog box or click **Cancel** to abort the operation.

## Delete Files

You can delete one or more files in the Folder View by right-clicking the filename(s) and choosing the *Delete* context menu. If you are deleting designs that are referenced by other designs, you will be shown the other designs and asked if you are sure to delete the designs.

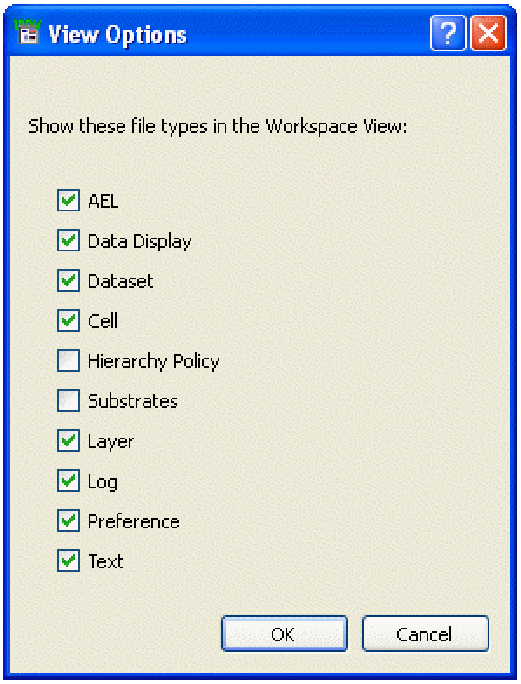
To delete a file or group of files,

1. Right-click the file that you want to delete in the Folder View.
2. If you want to delete more than one file, hold down the **Shift** key to select a group of files and/or use the **Ctrl** key to select multiple individual files.
3. Choose the *Delete* context menu.
4. Click **Yes** to confirm deletion. The files are deleted from disk and memory.

## Filter View

The View Options dialog box enables you to specify the file types that you want display or hide in the ADS Folder View. To start the View Options dialog box and change the display options:

1. Start ADS and open or create a workspace.
2. Select Folder View, if not selected.
3. Right-click anywhere in the blank space and select *Filter View...* menu option to open the View Options dialog box.

3. 

View Options

Show these file types in the Workspace View:

- ☑ AEL
- ☑ Data Display
- ☑ Dataset
- ☑ Cell
- ☐ Hierarchy Policy
- ☐ Substrates
- ☑ Layer
- ☑ Log
- ☑ Preference
- ☑ Text

4. Enable (select) the file types that you want to appear in the Folder View.
5. Disable (deselect) the file types that you do not want to appear in the Folder View.
6. Click **OK** to accept the changes and dismiss the View Options dialog box, or click **Cancel** to abort.

The available file types are:

- *Ael* - AEL files (*.ael) in the Workspace's top directory.
- *Data Display* - Data Display files (*.dds) in the Workspace's top directory.
- *Dataset* - dataset files (*.ds) in the data directory.
- *Cell* - Cells
- *Hierarchy Policy* - The hierarchy policy
- *Substrates* - Substrates
- *Layers* - layers files (*.lay) in the Workspace's top directory.
- *Log* - Log files
- *Preferences* - preferences files (*.prf) in the Workspace's top directory.
- *Text* - text files (*.txt) in the Workspace's top directory.

## Design Windows

ADS allows you to create different design types such as, schematic, symbol, and layout. A design can consist of a number of schematics and layouts embedded as subnetworks within a single design. All designs in a workspace can be displayed and opened directly from the ADS Main window.

A design window is where you create and edit all your designs. You can resize and move these windows in the workspace. You can enlarge one window to fill the entire workspace and you can shrink each window to an icon. ADS supports following design windows:

1. *Schematic Window* (adstour)
2. *Symbol Window* (adstour)
3. *Layout Window* (adstour)

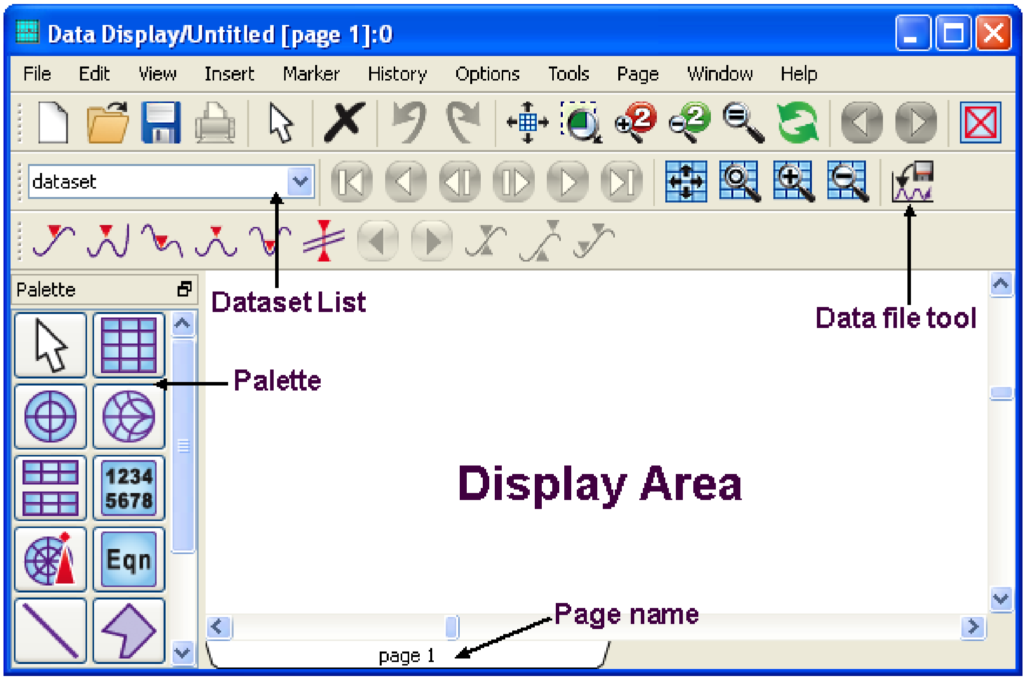
For more details, see *Using Designs* (adstour)

## Substrate Editor

For details, see *Substrate Editor* (adstour)

## Data Display Window

The Data Display window (see figure below) enables you to view and analyze the data generated by simulation, as well as data that has been imported from other sources, such as a network analyzer or CITIfile.



Data Display window allows you to:

- Display data in a variety of plots and formats.
- Create plots with more than two axes.
- Add markers to traces to read specific data points.
- Write mathematical equations to perform complex operations on data, and display the results.
- Add text and drawing objects to enhance your documentation.
- Edit plot titles and axis labels, equations, text, drawing objects, and column headings in lists.

For more details, see *Data Display Basics* (data)

## Manage ADS AEL Addons

Manage ADS AEL Addons dialog box can be used to add customized functionality to ADS. The customized AEL files can be loaded at ADS startup. Also, the AEL Addons can be enabled or disabled. ADS installer provides some example of **ADS Installation Addons**. These examples must be enabled by you.

Addons can be added at three access levels:

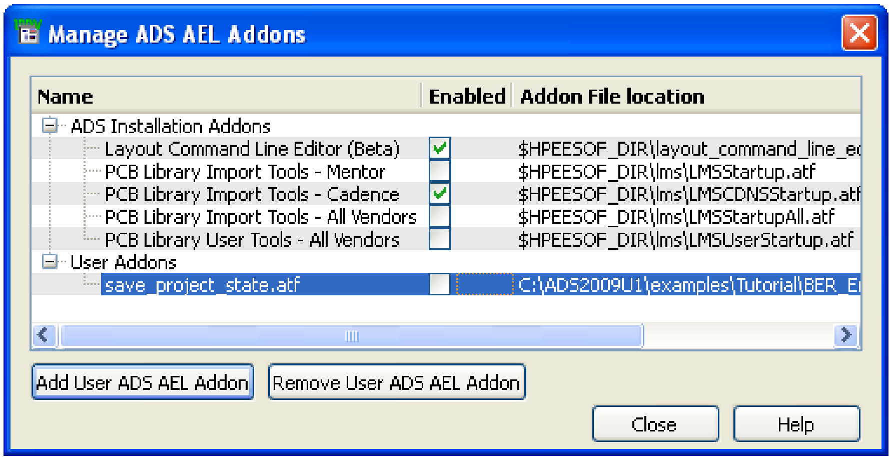
- SITE level: This information is stored at `$HPEESOF_DIR/custom/config/eesof_addons.xml`.
- INSTALL level: This informationn is stored at `$HPEESOF_DIR/config/eesof_addons.xml`.
- USER level: This information is stored at `$HOME/hpeesof/config/eesof_addons.xml`.

> **Note**
> Hand-editing of the USER level *eesof_addons.xml* file is not recommended, the USER level file is managed and controlled by the Manage ADS AEL Addons dialog.

The SITE and INSTALL access level addons are displayed in the dialog box only if the addons are available, and not otherwise.

> **Note**
> A Site Administrator can provide their own custom AEL addons for their site by adding their own customized eesof_addons.xml file to the SITE level `$HPEESOF_DIR/custom/config/eesof_addons.xml` location. The easiest way to do this is for the administrator to set up their user ael addons, then copy their user file to the site level.

To start th Manage ADS AEL Addons dialog box, from ADS Main window select **Tools > Manage ADS AEL addons**.

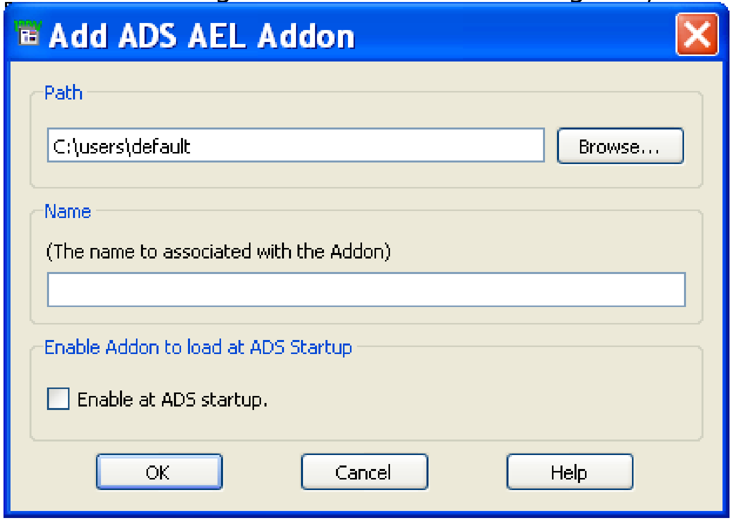


| Name | Enabled | Addon File location |
|---|---|---|
| ADS Installation Addons | | |
| Layout Command Line Editor (Beta) | ☑ | $HPEESOF_DIR\layout_command_line_ec |
| PCB Library Import Tools - Mentor | □ | $HPEESOF_DIR\lms\LMSStartup.atf |
| PCB Library Import Tools - Cadence | ☑ | $HPEESOF_DIR\lms\LMSCDNSStartup.atf |
| PCB Library Import Tools - All Vendors | □ | $HPEESOF_DIR\lms\LMSStartupAll.atf |
| PCB Library User Tools - All Vendors | □ | $HPEESOF_DIR\lms\LMSUserStartup.atf |
| User Addons | | |
| save_project_state.atf | □ | C:\ADS2009U1\examples\Tutorial\BER_E |

## Add ADS AEL Addon

Follow the steps below to add a new user AEL addon and enable the same:

1. From the Manage ADS AEL Addons dialog box, click **Add User ADS AEL Addon**.



Path

C:\users\default

Name

(The name to associated with the Addon)

Enable Addon to load at ADS Startup

□ Enable at ADS startup.

2. In the Add ADS AEL Addon window, type the system filepath location to the custom `.ael` or `.atf` AEL file into *Path*. You can also click **Browse** to select your custom AEL or ATF file.
3. Type a unique identifier name for your custom user AEL addon into *Name*.
4. Check the *Enable at ADS startup* option to enable loading of AEL file (selected in previous step) every time ADS starts.
5. Click **OK** to dismiss the window.

## Remove ADS AEL Addon

Follow the steps below to remove an user AEL addon:

1. From the Manage ADS AEL Addons dialog box, select the user addon you want to delete.
2. Click **Remove User ADS AEL Addon**.
3. From the confirmation message window, click **Yes** to remove.

## Enable ADS AEL Addon

To enable any AEL addon, you must check the *Enabled* option for the functionality that needs to be enabled.

1. From the Manage ADS AEL Addons dialog box, select the addon you want to enable.
2. Check the *Enabled* check box.
3. From the confirmation message window, click **Yes** to enable. By enabling an addon, it loads the corresponding AEL/ATF file when ADS starts.

## Disable ADS AEL Addon

To disable any addon, you must un-check the *Enabled* option for the functionality that needs to be disabled.

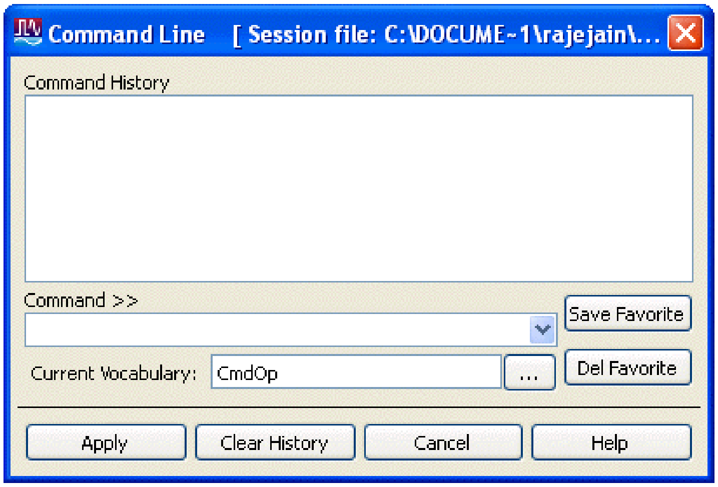
1. From the Manage ADS AEL Addons dialog box, select the addon you want to disable.
2. Un-check the _Enabled" check box.
3. From the confirmation message window, click **Yes** to disable. You must restart ADS to disable the functionality.

## Command Line Window (for AEL Commands)

The *AEL* (ael) commands that are issued in response to your activity in the Main window and the design windows are displayed in the Command Line dialog box. This command summary is updated continuously as you work. You can view this summary any time and you can issue previously executed commands from this list.

To execute the AEL commands:

1. From the Main window, choose **Tools > Command Line** to open the Command Line window.

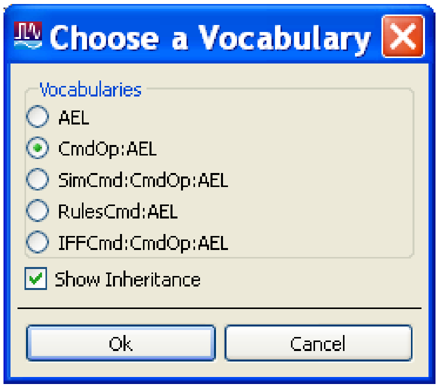


Current Vocabulary: CmdOp

2. Type the command(s) in the *Command >>* field and click **Apply** (or press **Enter** key) after each command to execute the same. As you execute commands, the corresponding AEL functions are displayed.

## Current Vocabulary

The Current Vocabulary option in command line window provides an option to select the AEL vocabulary in which you want the typed/selected AEL command to be executed in. *ComOp* is the default command. To change the vocabulary, click on the button provided next to the *Current Vocabulary* field. You can select the **Show Inheritance** option to see the inheritance hierarchy of AEL vocabularies available in ADS.



○ AEL
● CmdOp:AEL
○ SimCmd:CmdOp:AEL
○ RulesCmd:AEL
○ IFFCmd:CmdOp:AEL
☑ Show Inheritance

For more information on AEL Vocabularies, see *Introduction to AEL* (ael).

**Notes**

1. All commands entered in the *Command >>* field must be in AEL format.
2. You can also select a previously typed command from the list and press **Enter** key or click **Apply** to execute.
3. For configuration details on using AEL, refer to the *AEL* (ael) documentation. For layout artwork and usage, refer to the *Layout Library* (layout) documentation.

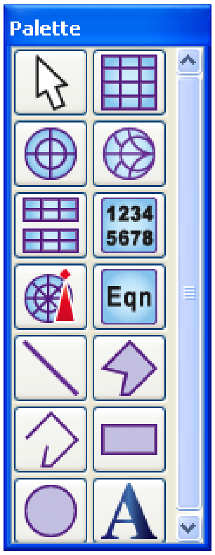
## Component Palette

Different design windows and other windows has a Component Palette which contains buttons that provide a quick method of placing items to create your design. This palette is available in:

1. Schematic Window

2. Data Display (DDS) Window
3. Symbol Window
4. Layout Window

**Component Palette in DDS Window**



> ✓ **Hint**
> All the palette items can also be placed through the Library. While some items are only available through the Library.

## Moving Toolbars (PC Only)

The toolbars can be repositioned anywhere on the screen. You can move them away from the window and use them like floating palettes or you can dock them along the window's edges.

> ✓ **Hint**
> When the title bar of a toolbar is visible, positioning your pointer within the title bar for the drag operation simplifies the docking process. If a title bar is not visible, move the toolbar away from the window's edge and release; when it is not docked, a title bar appears.

To float a toolbar away from the window:

1. Place the cursor on left edge of the toolbar.
2. Drag the toolbar to the desired location and release. When you release the toolbar, a title bar appears at the top of it.

To dock a toolbar on a window border:

1. Place the cursor on left edge of the toolbar.
2. Drag the toolbar toward the desired window border and notice that the ghost image of the toolbar changes as needed to fit in a vertical or horizontal space.
3. When the ghost image reflects the proper orientation, release the mouse button and refine the toolbar's position by dragging as necessary.

To re-attach a toolbar near the top of the window:

1. Place the cursor on left edge of the toolbar.
2. Drag the toolbar toward the top of the window and when your pointer is overlapping

the menu bar, or another toolbar, release.

## Simulation

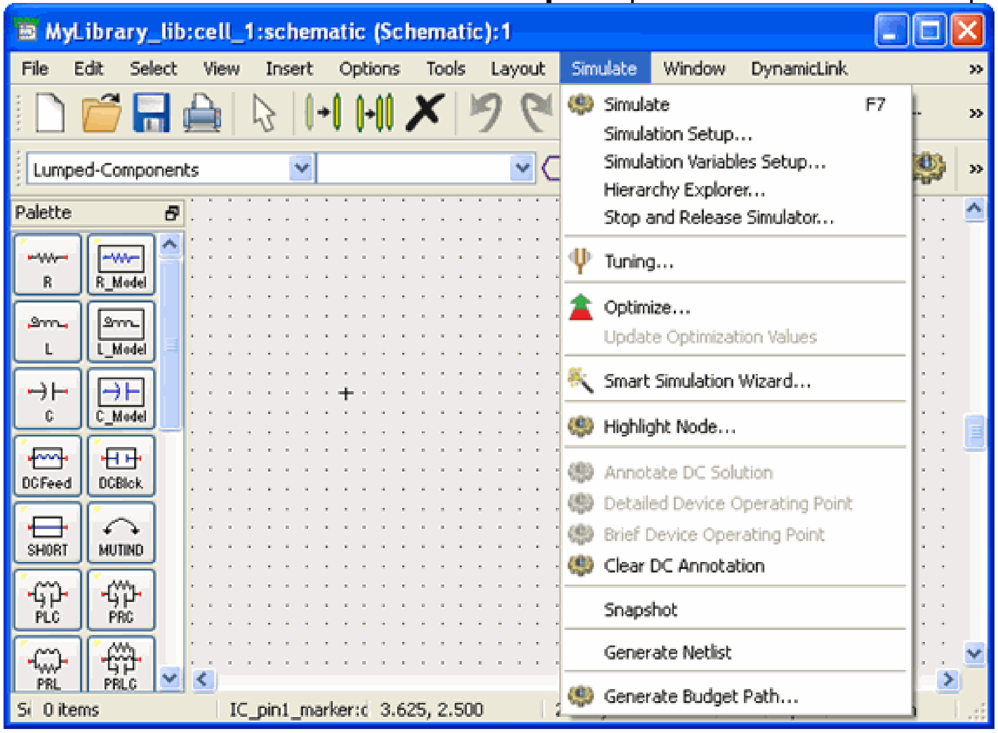
ADS allows you to create your own circuits which you can simulate using simulators provided to simulate the circuits and RF systems designed for specific objectives. You must have a valid ADS License to use these simulators.

There are different templates available to facilitate setting up common simulations. The simulation instrument components provide a method for symbolically connecting your circuit to an instrument. You connect your design to components that represent various instruments and run the simulation. Each simulation output generated with these simulators has a unique id.

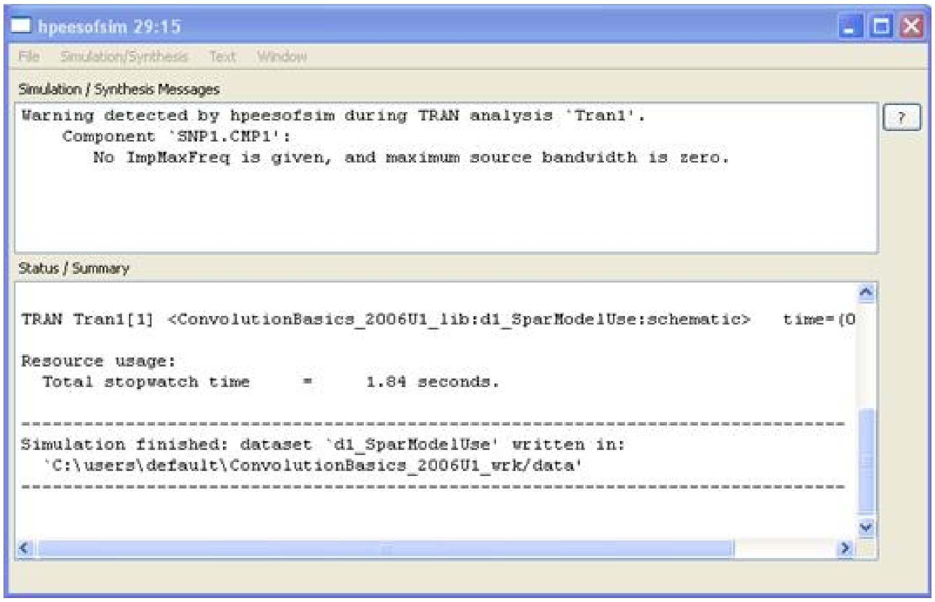
There are several ways to launch a simulation from the Schematic Window:

- Press the F7 key on your keyboard.
- Click the Simulate icon on the toolbar.
- Choose **Simulate > Simulate**.
- Click **Simulate** from the Simulation Setup dialog box.

1. Choose **Simulate > Simulation Setup** to open the Simulation Setup dialog box.



When the simulation begins, a status and error message window appears where you can see the simulation status and all the messages. Once the simulation is complete, **Simulation finished** message is displayed at the bottom of the window confirming that the simulation has run successfully. The location of the dataset where the simulation data is saved is also noted.



Simulation / Synthesis Messages

```
Warning detected by hpeesofsim during TRAN analysis `Tran1'.
    Component `SNP1.CMP1':
       No ImpMaxFreq is given, and maximum source bandwidth is zero.
```

Status / Summary

```
TRAN Tran1[1] <ConvolutionBasics_2006U1_lib:d1_SparModelUse:schematic>   time=(0

Resource usage:
  Total stopwatch time     =     1.84 seconds.

--------------------------------------------------------------------------------
Simulation finished: dataset `d1_SparModelUse' written in:
  `C:\users\default\ConvolutionBasics_2006U1_wrk/data'
--------------------------------------------------------------------------------
```

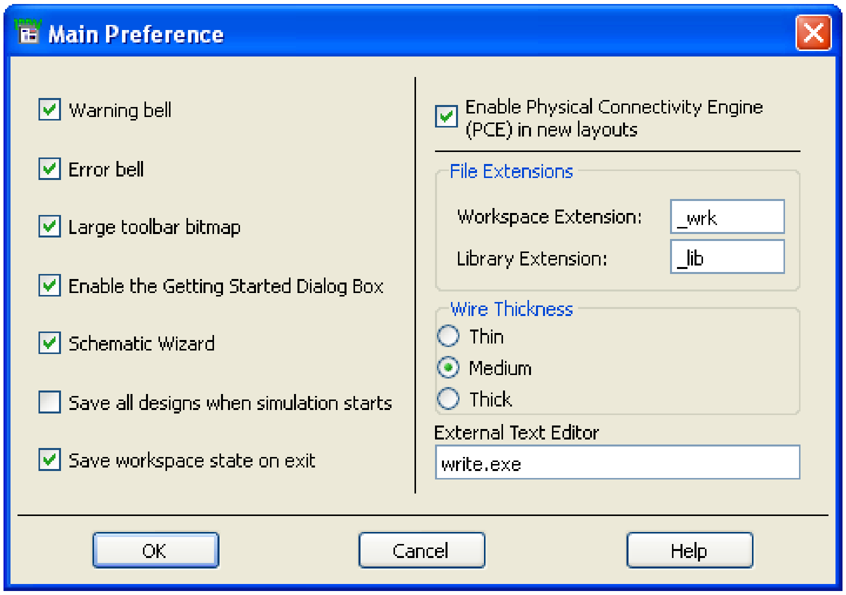
For more details, see *Simulating Designs* (adstour) and *Simulation Basics* (cktsim).

## Setting Preferences for Miscellaneous Options

You can set preferences that affect you throughout the ADS design environment using the Main Preferences dialog box. To open the **Main Preferences** dialog box choose **Options > Preferences...** from the Main window.

To change any of these settings:

1. Change any or all options as desired.
2. Click **OK**. All changes take effect immediately, except as mentioned in the description below.



☑ Warning bell

☑ Error bell

☑ Large toolbar bitmap

☑ Enable the Getting Started Dialog Box

☑ Schematic Wizard

☐ Save all designs when simulation starts

☑ Save workspace state on exit

☑ Enable Physical Connectivity Engine (PCE) in new layouts

File Extensions

Workspace Extension: _wrk

Library Extension: _lib

Wire Thickness

○ Thin

● Medium

○ Thick

External Text Editor

write.exe

- **Warning Bell** - The system beeps anytime you receive a pop-up window with a warning message.
- **Error Bell** - The system beeps anytime you receive a pop-up window with an error

message.

- **Large Toolbar Bitmap** - A set of large bitmaps is placed on the toolbar. Turn this option off to place a set of small bitmaps on the toolbar (better for monitors with lower screen resolution). This change will be evident in any subsequently opened windows. To see the change take effect in a currently open window, open the Hot Key/Toolbar Configuration dialog box, click the Toolbar tab, and click **OK**.
- **Enable the Getting Started Dialog Box** - The Getting Started dialog box is started at ADS startup.
- **Schematic Wizard** - The system automatically launches the Schematic Wizard when a new design is created.
- **Save all designs when simulation starts** – Save all the modified designs before launching the simulation.
- **Save Workspace State on Exit** - The setup of the Workspace you are exiting is saved, including all design windows. The group of windows, and their positions on the screen, are restored the next time you open the Workspace.
- **Enable Physical Connectivity Engine (PCE) in new layout** - Enables layout connectivity features. For more information, refer to *Physical Connectivity Engine* (usrguide)). See also the section on *Disabling Layout Connectivity Features* (usrguide) to understand the consequences of disabling the Physical Connectivity Engine.
- **File Extensions**
    1. **Workspace Extension** - The extension you want appended to workspace names, to clearly identify them as workspace (default is wrk).
    2. **Library Extension** - The extension you want appended to library names, to clearly identify them as library (default is lib).
- **Wire Thickness** - The thickness (Thin, Medium, Thick) of all wires drawn in a Schematic window.
- **External Text Editor** - Specifies the text editor to be launched when you choose **Tools > Text Editor** in the Main window.

# Using Workspace

Advanced Design System (ADS) uses the Workspace to organize and store the data generated when you create, simulate, and analyze designs. An ADS Workspace includes libraries, simulation data, data display files, and other related files.

A Workspace contains:

- **Library**: Zero or more libraries. When first created, a workspace will directly contain one library, plus it will reference other libraries specified during workspace creation (such as PDK's and ADS Libraries). A Library is a directory that includes cells and a definition file, such as `lib.defs`. This file contains a summary of all the libraries selected into the given Workspace and their mode of operation (Read-Only, Non-Shared, or Shared). For more details, see *Library* (oaqkref).
- **Cell**: Libraries contain cells. A cell contains zero or more views. It is somewhat similar to a design file (file with `dsn` extension) of ADS 2009 Update 1 and earlier versions. For more details, see *Cell* (oaqkref).
- **View**: Cells contain views. A View in a cell stores your design work such as schematic, symbol, or layout. For more details, see *View* (oaqkref).

> **Note**
> **For more details about ADS Workspace, see *Workspace* (oaqkref).**

## Working in Workspaces

All design work must be done in a workspace directory. Working in workspace directories enables you to organize related files within a predetermined file structure. This predetermined file structure consists of a set of subdirectories. These subdirectories are used in the following manner:

- *data* is the default directory location for input and output data files used or generated by the simulator
- *synthesis* contains designs created with DSP filter and synthesis tools
- *verification* contains files generated by the Design Rule Checker (DRC), used with Layout

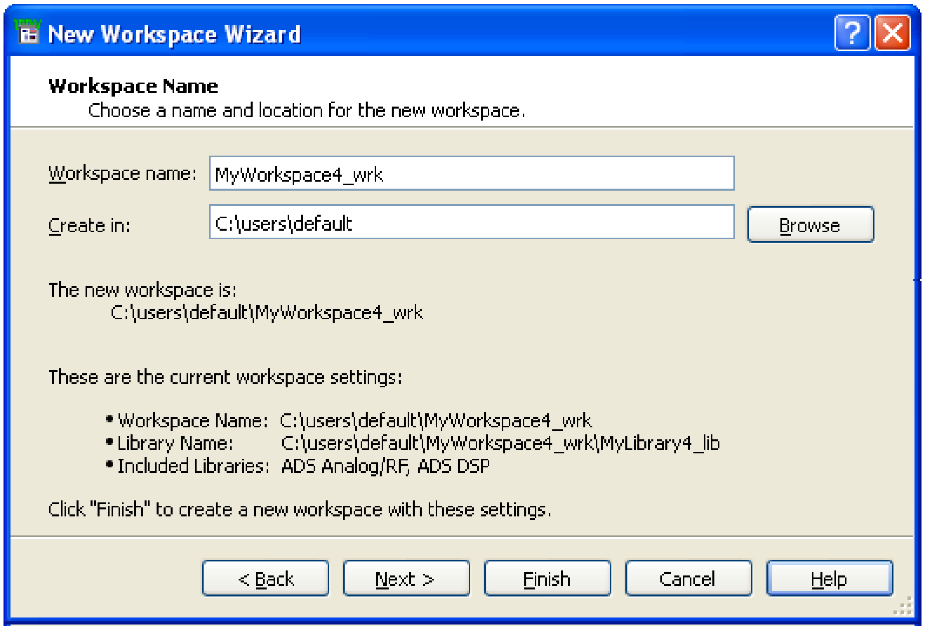
For any workspaces translated from ADS 2009 Update 1 or earlier projects:

- *old_networks* will be present. It contains the designs that have been translated. This directory is not used by ADS 2011 or later.
- *mom_dsn* may be present. It contains designs created with the Agilent EEsof planar electromagnetic simulator, Momentum. This directory is not used by ADS 2011 or later.

## Creating a Workspace

Follow the steps below to create a new Workspace:

1. Start ADS.
2. From the ADS Main window, choose **File > New > Workspace** to open the **New Workspace Wizard** and Click **Next**.



**Workspace Name**
Choose a name and location for the new workspace.

Workspace name: MyWorkspace4_wrk

Create in: C:\users\default

The new workspace is:
C:\users\default\MyWorkspace4_wrk

These are the current workspace settings:

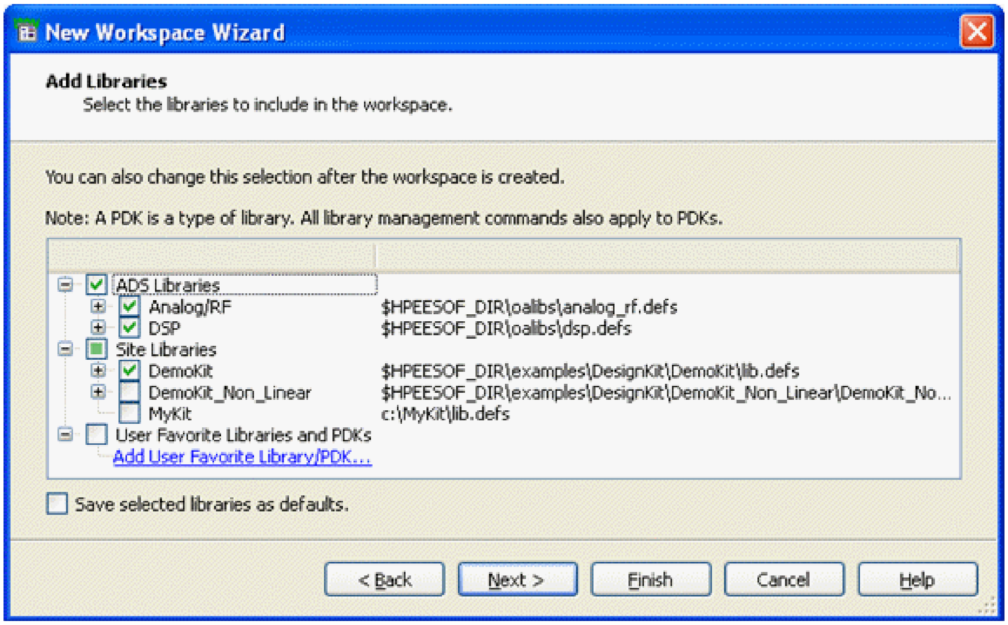
- Workspace Name: C:\users\default\MyWorkspace4_wrk
- Library Name: C:\users\default\MyWorkspace4_wrk\MyLibrary4_lib
- Included Libraries: ADS Analog/RF, ADS DSP

Click "Finish" to create a new workspace with these settings.

3. Enter the Workspace name, e.g. `MyWorkSpace4_wrk`.

> **Note**
> _wrk suffix is added to the Workspace name, which you can change or remove.

4. Enter the desired path in **Create In**. Click **Browse** to select the location.
5. Click **Next**.
6. Under **Add Libraries** select the libraries to be included in the Workspace.
7. Click **Next**.



**Add Libraries**
Select the libraries to include in the workspace.

You can also change this selection after the workspace is created.

Note: A PDK is a type of library. All library management commands also apply to PDKs.

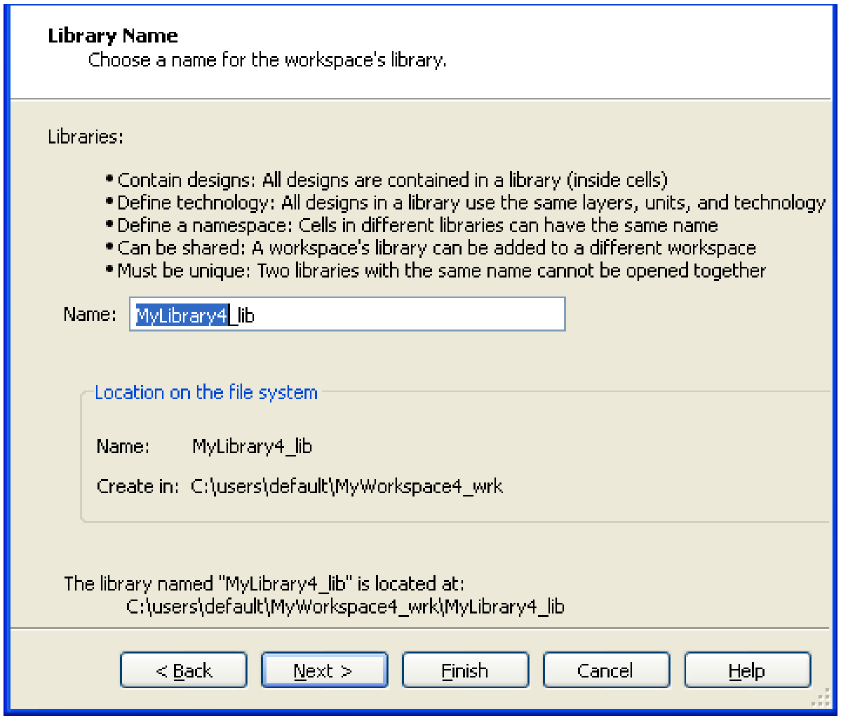
| | |
|---|---|
| ☑ ADS Libraries | |
| ☑ Analog/RF | $HPEESOF_DIR\oalibs\analog_rf.defs |
| ☑ DSP | $HPEESOF_DIR\oalibs\dsp.defs |
| ■ Site Libraries | |
| ☑ DemoKit | $HPEESOF_DIR\examples\DesignKit\DemoKit\lib.defs |
| □ DemoKit_Non_Linear | $HPEESOF_DIR\examples\DesignKit\DemoKit_Non_Linear\DemoKit_No... |
| □ MyKit | c:\MyKit\lib.defs |
| □ User Favorite Libraries and PDKs | |
| Add User Favorite Library/PDK... | |

□ Save selected libraries as defaults.

8. Under **Library Name**, enter a unique library name.

> **Notes**
> 1. The library name must be unique.
> 2. The default library name is `MyLibrary_lib` with _lib suffix, which you can change or remove.

9. Click **Next**.

9. 

**Library Name**
Choose a name for the workspace's library.

Libraries:

- Contain designs: All designs are contained in a library (inside cells)
- Define technology: All designs in a library use the same layers, units, and technology
- Define a namespace: Cells in different libraries can have the same name
- Can be shared: A workspace's library can be added to a different workspace
- Must be unique: Two libraries with the same name cannot be opened together

Name: MyLibrary4_lib

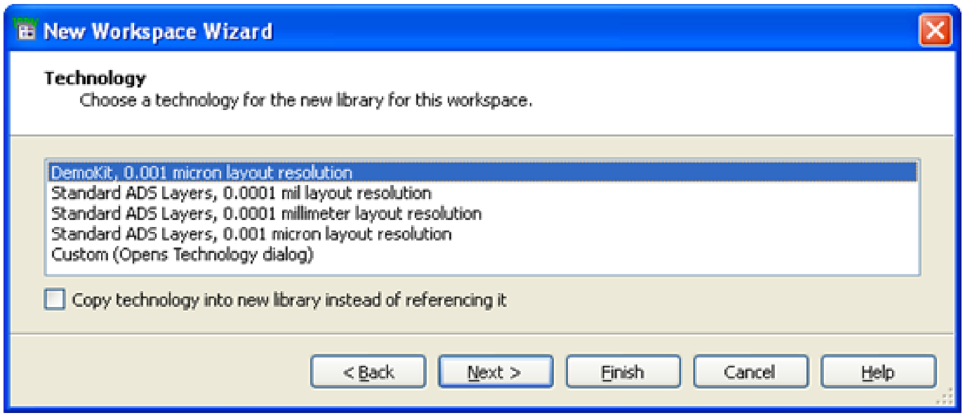
Location on the file system

Name: MyLibrary4_lib

Create in: C:\users\default\MyWorkspace4_wrk

The library named "MyLibrary4_lib" is located at:
C:\users\default\MyWorkspace4_wrk\MyLibrary4_lib

10. Under **Technology**, select the technology (from the list) for the library.
11. Click **Next**.



New Workspace Wizard

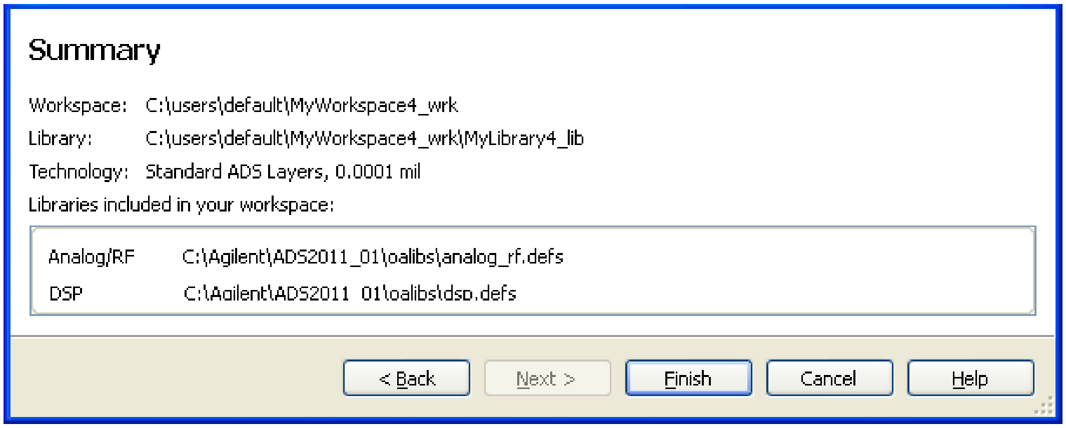
**Technology**
Choose a technology for the new library for this workspace.

- DemoKit, 0.001 micron layout resolution
- Standard ADS Layers, 0.0001 mil layout resolution
- Standard ADS Layers, 0.0001 millimeter layout resolution
- Standard ADS Layers, 0.001 micron layout resolution
- Custom (Opens Technology dialog)

☐ Copy technology into new library instead of referencing it

**Note**
If you select **Custom**, the **Technology Setup** dialog box opens after the **Summary** where you can specify the Type, Layout Units, etc. The technology specified here gets associated with the library name specified in the previous step. To create the manual association between Technology and Library, start the **Technology** dialog box from the ADS main window by selecting **Options > Technology**.

12. Summary window displays the summary of your actions performed in previous steps. You can click **Back** to go back and make relevant changes.

12. 

Summary

Workspace: C:\users\default\MyWorkspace4_wrk

Library: C:\users\default\MyWorkspace4_wrk\MyLibrary4_lib

Technology: Standard ADS Layers, 0.0001 mil

Libraries included in your workspace:

| | |
|---|---|
| Analog/RF | C:\Agilent\ADS2011_01\oalibs\analog_rf.defs |
| DSP | C:\Agilent\ADS2011_01\oalibs\dsp.defs |

13. Click **Finish** to create the `MyWorkSpace4_wrk` Workspace.

> **Note**
> If you have selected **Custom** option under Technology, the **Technology** dialog box opens after this step.

## Opening a Workspace

Follow the steps below to open an ADS Workspace:

1. Start ADS.
2. From the ADS Main window, choose **File > Open > Workspace** and use the Open Workspace dialog box to locate the Workspace to open.
3. Click **Choose** to open the Workspace.

> **Notes**
>
> 1. Before opening any Workspace, ADS prompts you to save the changes (if any) in already open Workspace.
> 2. Only one Workspace can be open at a given point of time. To open more than one Workspace, you need to start another instance of ADS.
> 3. If you try to open ADS 2009 project or an earlier release project, ADS automatically starts the *Convert Project to Workspace* (oaqkref) wizard.

## Deleting a Workspace

Follow the steps below to delete a Workspace:

1. Start ADS.
2. From the ADS Main window, choose **File > Delete Workspace** to open the Delete Workspace dialog box.
3. Select the Workspace to be deleted and click **Choose**.
4. In confirmation window, click **Yes** to delete the Workspace.

> **Note**
> You cannot delete an open Workspace.

## Renaming a Workspace

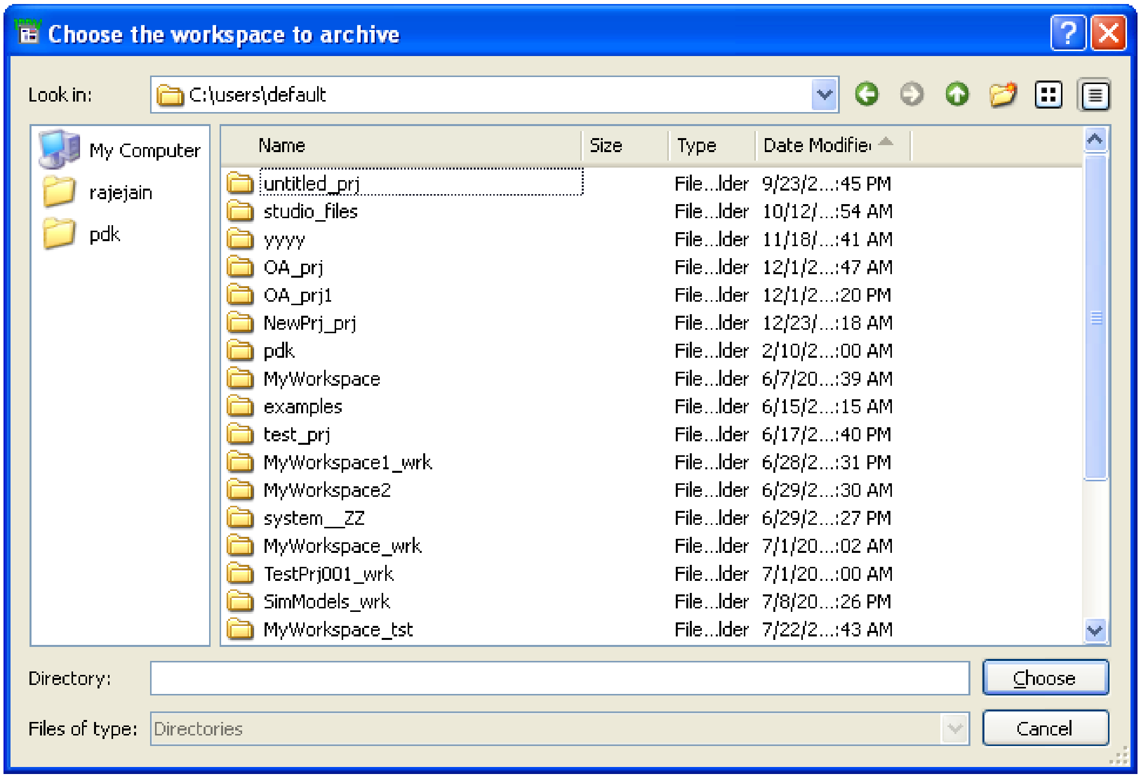
You can rename any workspace just like renaming any folder name using Windows explorer (for Windows version) or any operating system commands (for Linux/Solaris version). Before renaming a workspace, ensure that the workspace is not open in ADS.

## Archiving a Workspace

You can Archive/Unarchive your Workspace to transfer a compact Workspace archive. Creating a single file for a Workspace simplifies transferring Workspace to another file system or to another location on the same file system.

To archive a Workspace,

1. Choose **File > Archive Workspace** to open the Choose the workspace to archive dialog box.
2. Select the workspace to be archived and click **Choose**.



Look in: C:\users\default

| Name | Size | Type | Date Modifie |
|---|---|---|---|
| untitled_prj | | File...lder | 9/23/2...:45 PM |
| studio_files | | File...lder | 10/12/...:54 AM |
| yyyy | | File...lder | 11/18/...:41 AM |
| OA_prj | | File...lder | 12/1/2...:47 AM |
| OA_prj1 | | File...lder | 12/1/2...:20 PM |
| NewPrj_prj | | File...lder | 12/23/...:18 AM |
| pdk | | File...lder | 2/10/2...:00 AM |
| MyWorkspace | | File...lder | 6/7/20...:39 AM |
| examples | | File...lder | 6/15/2...:15 AM |
| test_prj | | File...lder | 6/17/2...:40 PM |
| MyWorkspace1_wrk | | File...lder | 6/28/2...:31 PM |
| MyWorkspace2 | | File...lder | 6/29/2...:30 AM |
| system__ZZ | | File...lder | 6/29/2...:27 PM |
| MyWorkspace_wrk | | File...lder | 7/1/20...:02 AM |
| TestPrj001_wrk | | File...lder | 7/1/20...:00 AM |
| SimModels_wrk | | File...lder | 7/8/20...:26 PM |
| MyWorkspace_tst | | File...lder | 7/22/2...:43 AM |

Directory:

Files of type: Directories

3. Enter the archive filename and select location where you want to save the archive file.

> **Note**
> Workspace is archived in `7zap` format.

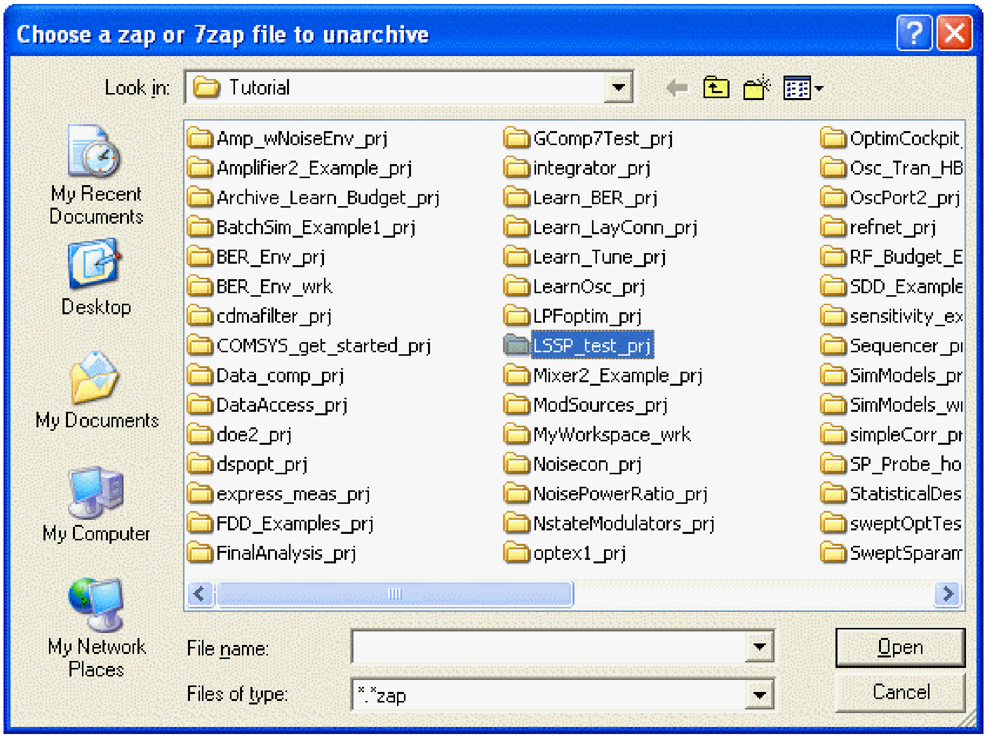
4. Click **Save** to archive the Workspace. After successful archive, a confirmation message is displayed.

## Unarchiving a Workspace

Follow the steps below to unarchive a Workspace:

1. Start ADS.
2. From the ADS Main window, choose **File > Unarchive Workspace** to open the Unarchive Workspace dialog box.

2\.



Choose a zap or 7zap file to unarchive

Look in: Tutorial

File name:

Files of type: *.*zap

3. Select or Enter the file name to be unarchived and click **Open**.
4. Select the directory where you want to unarchive the selected file.
5. Click **Choose** to unarchive the Workspace. After successful unarchive, a confirmation message is displayed with an option to open the unarchived workspace.

## Opening an Example Workspace

Advanced Design System offers an extensive set of example Workspaces that demonstrate designing for various technologies. You can view example Workspace, as well as copy and modify them to create new workspace.

Follow the steps below to open an ADS example:

1. Start ADS.
2. From the ADS Main window, choose **File > Open > Example** and select the example file (file with extension `.7zap`) and click **Open**.
3. Enter location where the example file should be unarchived and click **Choose**.
4. After unarchiving the example file, ADS prompts you to open the example file. Click **Yes** to open the example file, otherwise **No**.

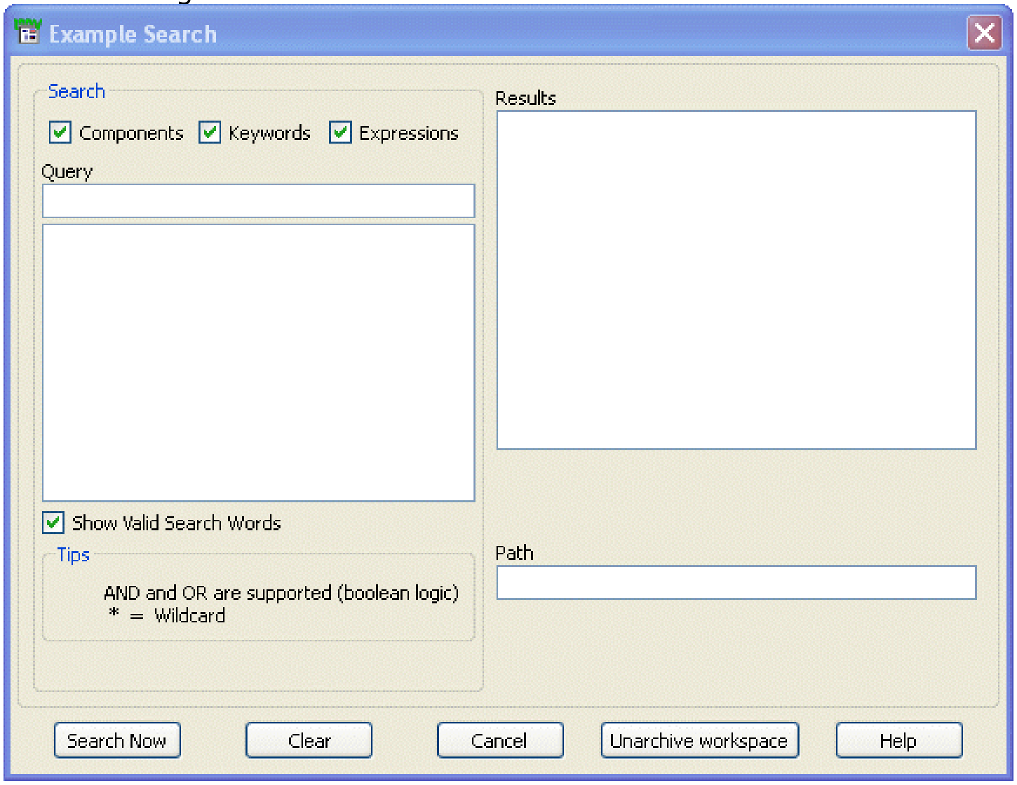
## Finding an Example Workspace or Design

Use the examples search to look for keywords, expressions, or component names in example examples and designs. This search feature looks through the design, layout, and data display files within all example workspaces and displays a list of workspaces that contain the terms you specified.

Following the steps below to search example (from ADS Main Window):

1. From ADS Main Window, choose **Tools > Examples Search** to open the Example

Search dialog.



☑ Components ☑ Keywords ☑ Expressions

☑ Show Valid Search Words

AND and OR are supported (boolean logic)
* = Wildcard

2. Use the *Search* section of the dialog box to define any combination of the following choices to define your search criteria.
   - **Components** - Search for a specific component.
   - **Keywords** - Search for a specific keyword.
   - **Expressions** - Search for a specific expression.
3. Use the Query field to enter the search word or a combination of the search word separated by Boolean operators. The search words are case sensitive. For example searching the word *amplifier* will produce different results than searching for *Amplifier*. This is because *amplifier* is treated as a keyword, while *Amplifier* is treated as a component name. You can use Boolean "OR" operation if you want to search for both *amplifier* and *Amplifier*.
   Use an asterisk (" * ") at either end of the word as a wildcard when entering your search criteria. For example, use " *ing " to look for all words with suffix "ing." When using wildcards, the search is limited to a maximum of one hundred words.
   If you enter two or more words separated by a space, the *AND* operator is implied. You can also specify AND using uppercase letters. For example, Amplifier BPF_Butterworth Attenuator returns the same results as Amplifier AND BPF_Butterworth AND Attenuator.
   An *OR* operator requires an explicit entry using uppercase letters. For example, Amplifier OR BPF_Butterworth OR Attenuator. Note that all multiple word search is limited to a maximum of four words.
4. Select **Show Valid Search Words** to display a list of valid words corresponding to the letters you type. The words appear in the list below the text entry field. You can double-click any word in the scroll-down list to add it to the Query field.
5. Click **Search Now** to begin the search. You can also click **Clear** to clear the search criteria.

Example workspaces that meet the search criteria are listed in the Results section. Use the " + " in the Results field to expand an example workspace hierarchy and view the designs or data display files. A red X across an example in the *Results* field indicates the example is not available for viewing. You may need to install the example from your CD.

The Path field displays the full path to the currently selected example. Double-click a workspace, a design, or data display in the Results field to open the selected item.

## Closing a Workspace

To close any open Workspace, click **File > Close Workspace**.

> **Note**
> Before closing the Workspace, ADS prompts you to save the changes (if any) in open Workspace.

## Exiting from ADS

You can exit the ADS program from the any of the design windows or from the main ADS window.

### Exit from ADS Main Window

To close your workspace and exit the program :

- Choose **File > Exit**.
- Click **Yes** to exit from Advanced Design System.

### Exit from Design Window

To close your workspace and exit the program from any of the open design windows:

- Choose **File > Exit Advance Design System** from any design window (such as Symbol, Schematic, or Layout).
- Click **Yes** to exit Advanced Design System.

# Using Libraries

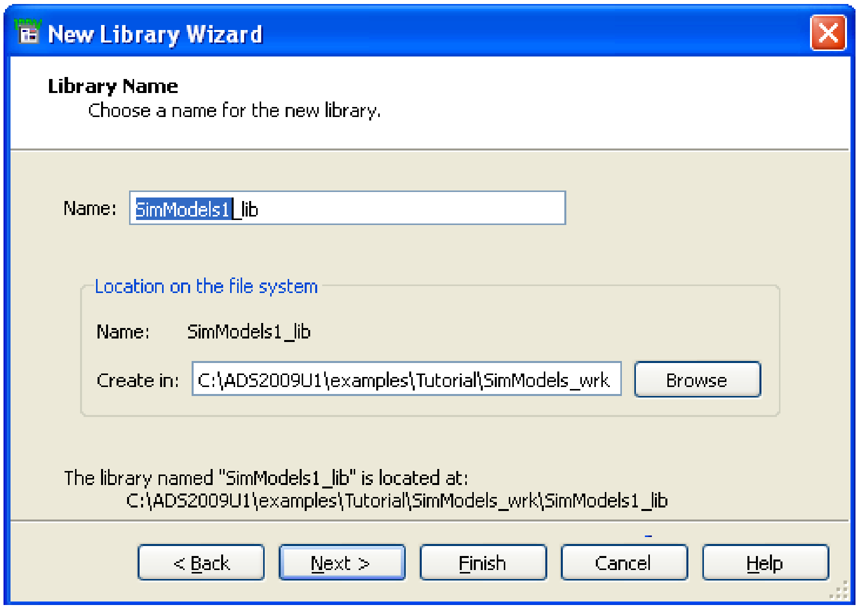
A Library is a collection of cells. It is a directory that holds cells and a definition file, such as `lib.defs`. This file defines the library name associated with workspace and their mode of operation (Read only or Shared). It also defines the technology (layers, resolution, and layout units) to be used by the Views created in that library. You can create multiple libraries within a complete design hierarchy. A library does not have to physically reside in the workspace directory.

> **Note**
> **For more details about Libraries, see *Library* (oaqkref).**

## Creating a New Library

Follow these steps to create a new Library:

1. Start ADS and open or create a workspace.
2. From ADS main window, choose **File > New > Library** to start the New Library Wizard.
3. Click **Next**.



**Library Name**
Choose a name for the new library.

Name: SimModels1_lib

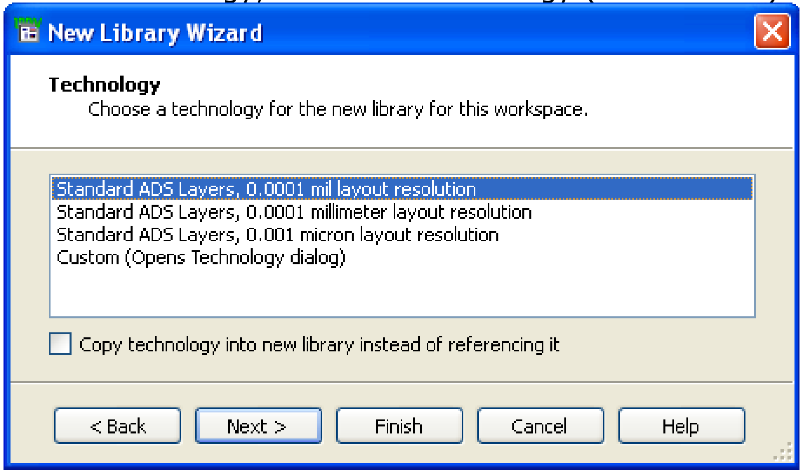
Location on the file system

Name: SimModels1_lib

Create in: C:\ADS2009U1\examples\Tutorial\SimModels_wrk

The library named "SimModels1_lib" is located at:
C:\ADS2009U1\examples\Tutorial\SimModels_wrk\SimModels1_lib

4. Enter Library Name and Location in **Name** and **Create In** respectively and click **Next**.
5. Under Technology, select the technology (from the list) for the library and click **Next**.



**Technology**
Choose a technology for the new library for this workspace.

Standard ADS Layers, 0.0001 mil layout resolution
Standard ADS Layers, 0.0001 millimeter layout resolution
Standard ADS Layers, 0.001 micron layout resolution
Custom (Opens Technology dialog)

□ Copy technology into new library instead of referencing it

> If you select **Custom**, the Technology dialog box opens after the Summary page where you can specify the Type, Layout Units, etc. The technology specified here gets associated with the library created.

6. Summary window displays the summary of your actions performed in previous steps. You can click **Back** to go back and make relevant changes.
7. Click **Finish** to create the library.

> **Notes**
> 1. If you have selected **Custom** option under Technology, the Technology dialog box opens after this step.
> 2. To see the list of all the Libraries, choose **File > Manage Libraries** from the ADS Main window.

## Open Library

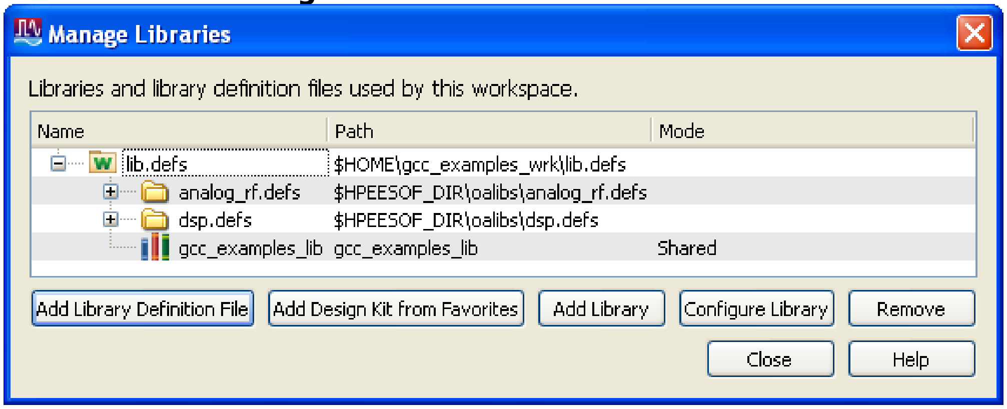
The Open Library option is same as Add Library. You can open a library to add the same in an already open workspace.

To open a library, choose **File > Open Library** from the ADS Main window and follow the steps in Add Library.

## Manage Libraries

Follow the steps below to manage libraries associated with a workspace:

1. Start ADS and open or create a workspace.
2. Choose **File > Manage Libraries** from the ADS Main window.



Manage Libraries

Libraries and library definition files used by this workspace.

| Name | Path | Mode |
|---|---|---|
| lib.defs | $HOME\gcc_examples_wrk\lib.defs | |
| analog_rf.defs | $HPEESOF_DIR\oalibs\analog_rf.defs | |
| dsp.defs | $HPEESOF_DIR\oalibs\dsp.defs | |
| gcc_examples_lib | gcc_examples_lib | Shared |

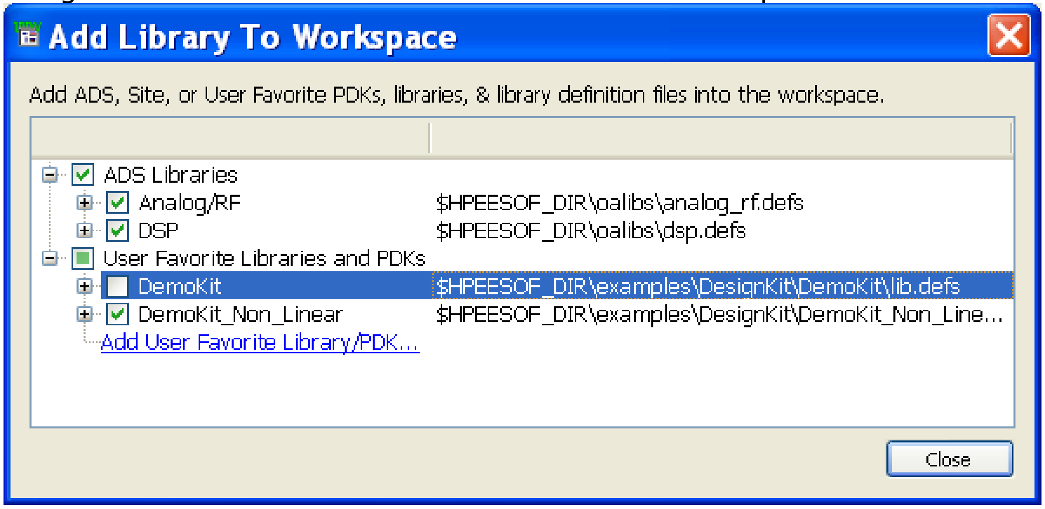
## Add Library Definition File

Library Definition file (lib.defs) is a text file and is similar to the ads.lib file in previous releases of ADS. It lists libraries, their names, path to actual library on disk, and the open mode of the library. ADS uses this file to load the libraries to the current workspace.

Use the **Add Library Definition File** option to browse to an existing lib.defs file. The browsed lib.defs file will be added as an include statement in current workspace's lib.defs file. For more details on syntax and semantics of lib.defs , see lib.defs.

## Add Design Kit from Favorites

This option lets you enable the design kits form a list of favorites. When enabled the design kit's lib.def file is included in the current workspace's lib.defs file.



Add ADS, Site, or User Favorite PDKs, libraries, & library definition files into the workspace.

| | |
|---|---|
| ☑ ADS Libraries | |
| ☑ Analog/RF | $HPEESOF_DIR\oalibs\analog_rf.defs |
| ☑ DSP | $HPEESOF_DIR\oalibs\dsp.defs |
| ■ User Favorite Libraries and PDKs | |
| □ DemoKit | $HPEESOF_DIR\examples\DesignKit\DemoKit\lib.defs |
| ☑ DemoKit_Non_Linear | $HPEESOF_DIR\examples\DesignKit\DemoKit_Non_Line... |
| Add User Favorite Library/PDK... | |

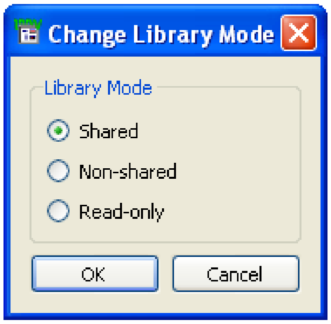
## Library Mode

A library can be opened in any of the following three different modes:

- Shared
- Non-Shared
- Read-Only

To change the library mode,

1. In Manage Libraries window, select the library name.
2. Right click and select **Change Library Mode** option.
3. Choose the desired mode and click **OK**.



Library Mode

● Shared
○ Non-shared
○ Read-only

## Shared

Shared mode means that one library can be shared between two or more users but at a given time only person can edit a particular design. When a design is opened in shared mode, a lock file is created in the cellview. If another user (or another copy of ADS) attempts to open the same design an error is returned because of the lock file. This prevents modification of same design by two or more user at a given point of time.

## Non-Shared

Non-Shared mode means that one library is not shared between two or more users or ADS instances. When a design is opened in non-shared mode, no lock file is created. If same design is opened by two or more users or at two different instances of ADS, it can be edited by any user with no warning or error message that library is already in use.

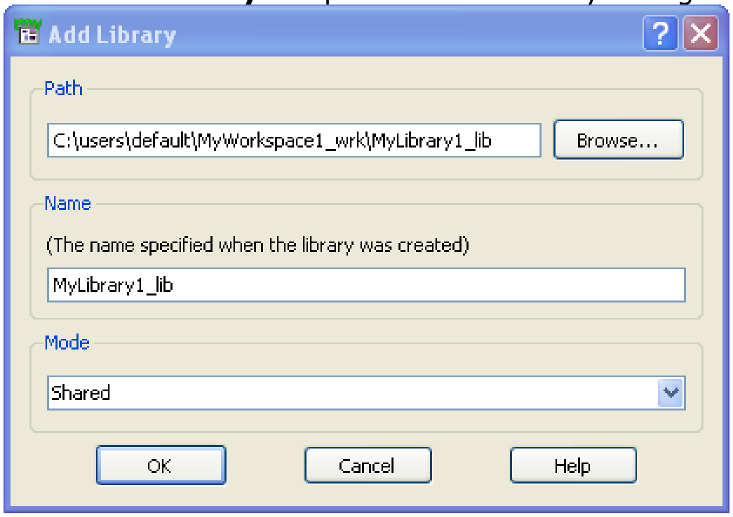
## Read-Only

If library is opened in read-only mode, you can not modify or update the same.

## Add Library

Follow the steps below to add an existing library into the workspace:

1. Start ADS and open or create a workspace.
2. From ADS Main window, choose **File > Manage Libraries...** to open the Manage Libraries dialog box.
3. Click **Add Library** to open the Add Library dialog box.



Path: C:\users\default\MyWorkspace1_wrk\MyLibrary1_lib

Name: (The name specified when the library was created) MyLibrary1_lib

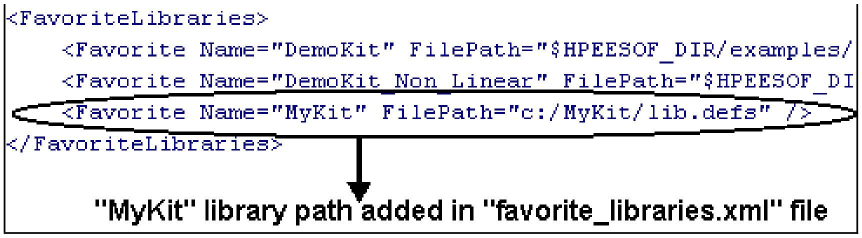
Mode: Shared

4. Click **Browse...** and select the library to be added in workspace. The valid library name is displayed automatically under **Name**.
5. Select the desired mode from the **Mode** drop-down list.
6. Click **OK** to add the library.

## Adding Site Libraries

System Administrators can add their own libraries under Site Libraries and set save as default libraries. To add a library under Site Libraries (and save as default) follow the steps below:

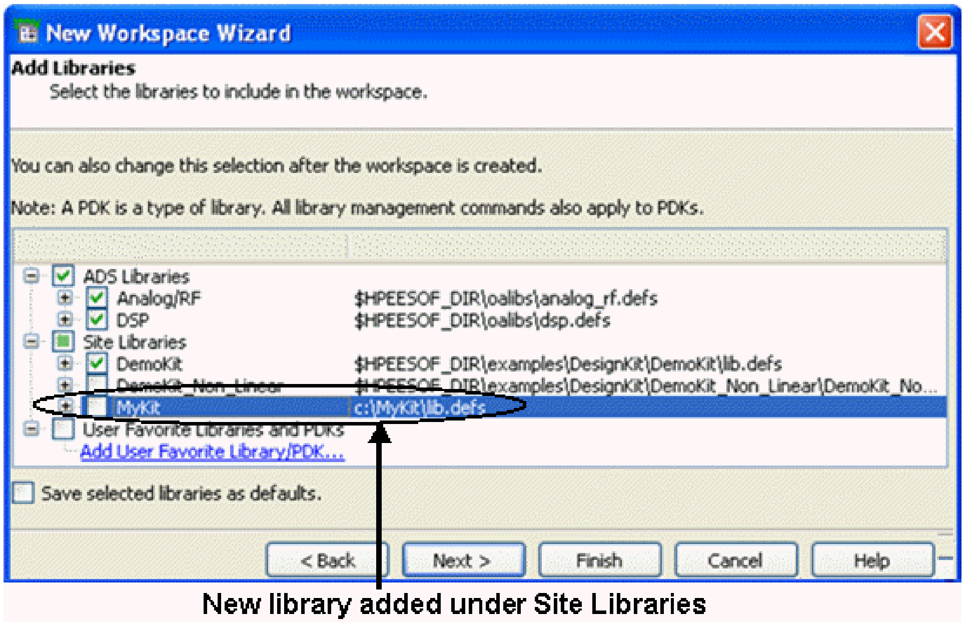
1. Open the favorite_libraries.xml, located in `\custom\config` folder of the ADS installation folder.
2. Under *FavoriteLibraries* tag, add your library (as shows in figure below). Be sure to specify a directory that will be valid on all users' machines.



```
<FavoriteLibraries>
    <Favorite Name="DemoKit" FilePath="$HPEESOF_DIR/examples/
    <Favorite Name="DemoKit_Non_Linear" FilePath="$HPEESOF_DI
    <Favorite Name="MyKit" FilePath="c:/MyKit/lib.defs" />
</FavoriteLibraries>
```

"MyKit" library path added in "favorite_libraries.xml" file

3. Save the XML file and create new workspace. Newly added `MyKit` library is visible under Site Libraries.



New Workspace Wizard

**Add Libraries**

Select the libraries to include in the workspace.

You can also change this selection after the workspace is created.

Note: A PDK is a type of library. All library management commands also apply to PDKs.

```
⊟ ☑ ADS Libraries
    ⊞ ☑ Analog/RF            $HPEESOF_DIR\oalibs\analog_rf.defs
    ⊞ ☑ DSP                  $HPEESOF_DIR\oalibs\dsp.defs
⊟ ▣ Site Libraries
    ⊞ ☑ DemoKit              $HPEESOF_DIR\examples\DesignKit\DemoKit\lib.defs
    ⊞ □ DemoKit_Non_Linear   $HPEESOF_DIR\examples\DesignKit\DemoKit_Non_Linear\DemoKit_No...
    ⊞ □ MyKit                c:\MyKit\lib.defs
⊟ □ User Favorite Libraries and PDKs
      Add User Favorite Library/PDK...
```

□ Save selected libraries as defaults.

New library added under Site Libraries

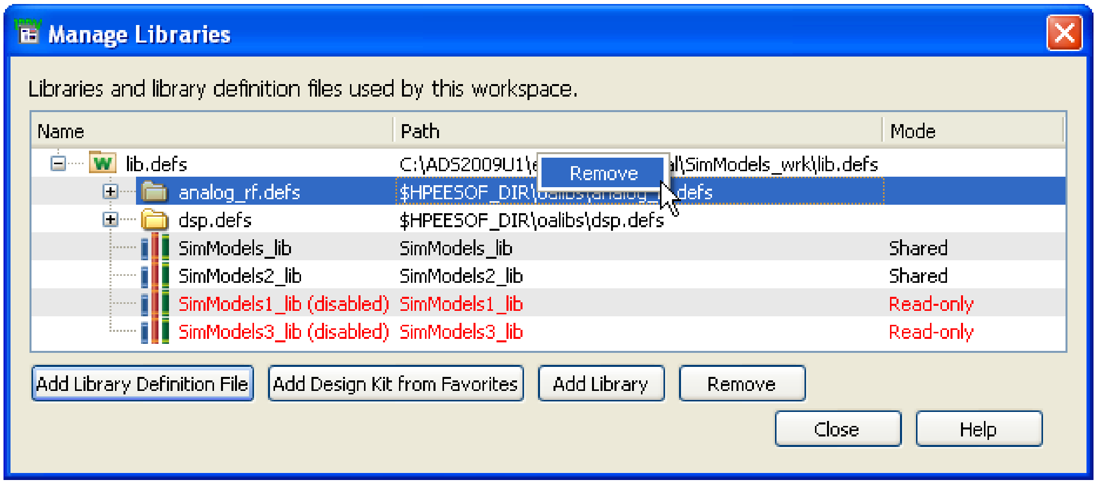
4. Click the checkbox before `MyKit` and click *Save selected libraries as default*. From next workspace creation instance, `MyKit` library will be included by default.

> **Note**
> The easiest way to create this file is create a user favorites library setup as you prefer, then copy favorite_libraries.xml from $HOME/hpeesof/config.

## Remove Library

Follow the steps below to remove a Library, already included in the currently open workspace:

1. Start ADS and open a workspace.
2. From ADS Main window, choose **File > Manage Libraries** to open the Manage Libraries dialog box.
3. Right-click on the Library name and select *Remove* or click the **Remove** button to remove library from the workspace.



Manage Libraries

Libraries and library definition files used by this workspace.

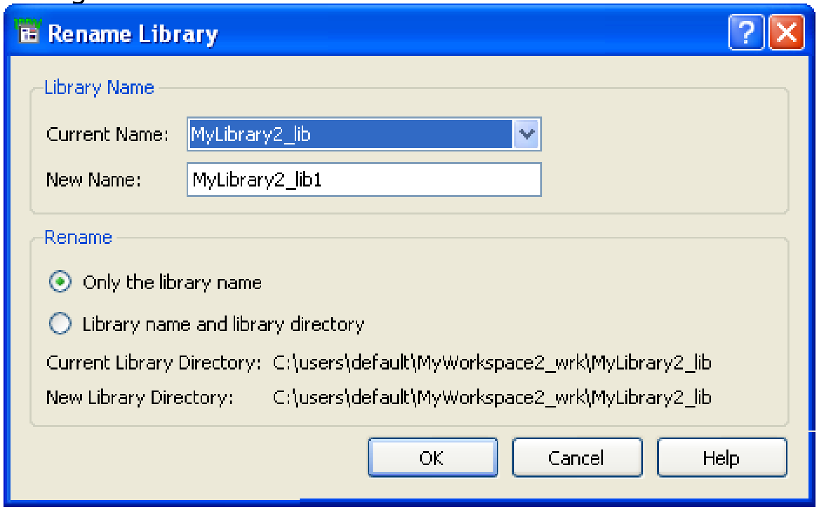
| Name | Path | Mode |
|---|---|---|
| lib.defs | C:\ADS2009U1\e l\SimModels_wrk\lib.defs | |
| analog_rf.defs | $HPEESOF_DIR\ defs | |
| dsp.defs | $HPEESOF_DIR\oalibs\dsp.defs | |
| SimModels_lib | SimModels_lib | Shared |
| SimModels2_lib | SimModels2_lib | Shared |
| SimModels1_lib (disabled) | SimModels1_lib | Read-only |
| SimModels3_lib (disabled) | SimModels3_lib | Read-only |

**Note**
Remove Library option removes the library from the given Workspace and does not mean physical deletion of library.

## Rename Library

Follow these steps to rename a Library:

1. Start ADS and open or create a workspace.
2. From ADS Main window, choose **File > Rename Library** to open the Rename Library dialog box.



Rename Library

Library Name

Current Name: MyLibrary2_lib

New Name: MyLibrary2_lib1

Rename

● Only the library name

○ Library name and library directory

Current Library Directory: C:\users\default\MyWorkspace2_wrk\MyLibrary2_lib

New Library Directory: C:\users\default\MyWorkspace2_wrk\MyLibrary2_lib

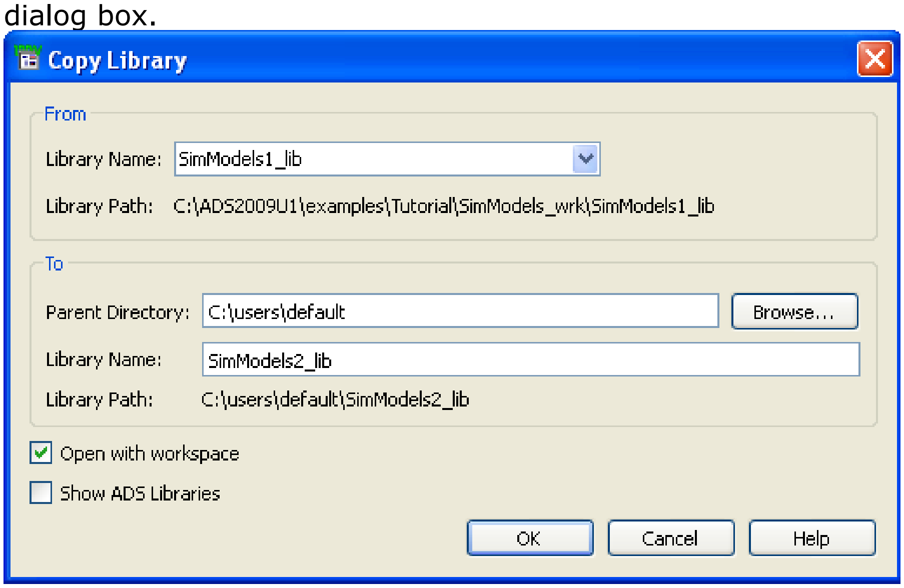
**Note**
All designs must be closed before opening Rename Library dialog box.

3. Under **Library Name**, select the library to be renamed from the **Current Name** drop-down list.
4. Enter the new name for the Library in the **New Name** field.
5. Under **Rename**, select *Only the library name* to rename only the library and select *Library name and library directory* to rename both.
6. Click **OK**.

## Copy Library

Follow these steps to copy a Library:

1. Start ADS and open or create a workspace.
2. From the ADS Main window, select **File > Copy Library** to start "Copy Library" dialog box.



Copy Library

From

Library Name: SimModels1_lib

Library Path: C:\ADS2009U1\examples\Tutorial\SimModels_wrk\SimModels1_lib

To

Parent Directory: C:\users\default

Library Name: SimModels2_lib

Library Path: C:\users\default\SimModels2_lib

☑ Open with workspace

□ Show ADS Libraries

3. Under **From**, select the library name to be copied from the **Library Name** drop-down list.
4. Under **To**, select the directory location where library should be copied. Click **Browse** to select the different location.
5. Enter **Library Directory Name** of the new library.
   - You have **Show ADS Libraries** option to display all ADS default libraries that you can copy to your workspace.
   - **Open with Workspace** opens those copied library with the workspace. If you clear **Open With Workspace**, the copied libraries remain in your directory, but do not open with the workspace.
6. Click **OK** to create copy of the library.

## Related Video

| | | |
|---|---|---|
| | | Video 1: Create a Copy Of a Design Cell |

## lib.defs file

ADS can open existing libraries by reading a library definition file and opening all of the libraries defined in it. ADS reads the library definition file when a workspace is opened.

There are three types of keyword statements that you can use in a lib.defs file:

*DEFINE* statement - Specify a particular library for use in a workspace. Relative paths are interpreted relative to the directory containing the lib.defs file.
*INCLUDE* statement - Use a set of libraries as defined in another lib.defs file. Relative paths are interpreted relative to the directory containing the lib.defs file.
*ASSIGN* statement - Specify attributes for a particular library.

Keywords are case insensitive. An end of line (EOL), end of file (EOF), or comment character terminates an entry in a `lib.defs` file.

## DEFINE Statements

A library is defined in the library definition file by using the DEFINE keyword, followed by a logical name or label for your library, followed by the path to the directory containing the cells. Relative paths are interpreted relative to the directory containing the lib.defs file.

## INCLUDE Statements

You can use the INCLUDE statement to include all the libraries listed in a different lib.defs file, as follows:

```
INCLUDE /usrDir/libs/lib.defs
INCLUDE ../lib.defs
```

## ASSIGN Statements

You can use the ASSIGN statement to assign attributes to a library. The syntax is as follows:

```
ASSIGN libName attrName attrValue
```

| **Attribute Name** | **Attribute Value** |
|---|---|
| *libMode* | *shared*, *nonShared*, *readOnly* |
| *writePath* | is the write path for the library |

## Comments

Use the pound character (#) at the beginning of a line for comments. You can also include inline comments by placing a pound character with a space after it, followed by your comment text.

# Using Designs

Advanced Design System allows you to create different design types such as, schematic, symbol, and layout. A design can consist of one or more schematics and layouts embedded as subnetworks within a single design. All designs in a workspace can be displayed and opened directly from the ADS Main window.

ADS uses the Cell to store these designs. A Cell is a container of one or more views.

The ADS design window can be used to:

- Create and modify circuits and layouts.
- Add variables and equations.
- Place and configure components, shapes, and simulation controllers.
- Specify layer and display preferences.
- Include annotations using text and illustrations.
- Generate layouts from schematics (and schematics from layouts).

## Creating Designs

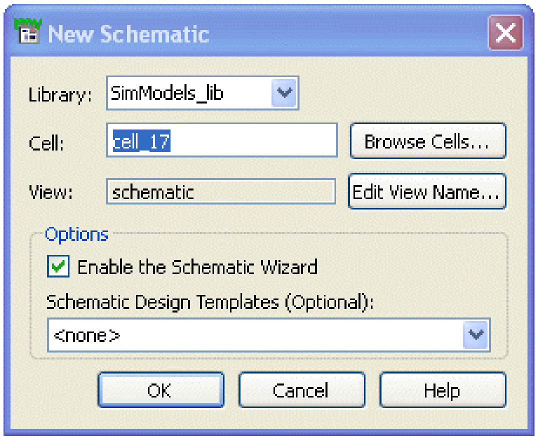
You can create following type of designs in ADS:

- Schematic
- Layout
- Symbol

## Creating a New Schematic

To create a new Schematic, follow the steps below:

1. Start ADS and open an existing workspace, or create a new workspace.
2. From the ADS Main window, choose **File > New > Schematic** to open the New Schematic dialog.



Library: SimModels_lib
Cell: cell_17
View: schematic
☑ Enable the Schematic Wizard
Schematic Design Templates (Optional): <none>

3. From the **Library** drop down list, select the library name where the new schematic will be stored.
4. Enter the new cell name or click **Browse Cells** to select the cell from existing cells of the selected library.

> **Note**
> By default, a view is created in a new cell. To add it to an existing cell, change the cell name. Or, as a shortcut, instead of using **File > New > Schematic**, you can right-click on an existing cell in the Folder View or Library View and select New Schematic.

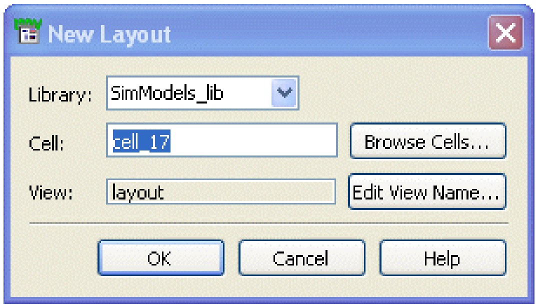
5. Click **Edit View Name** to create a new view.
6. From **Schematic Design Templates** list, you can select the template to be used or check the *Enable the Schematic Wizard* to start the Schematic Wizard.
7. Click **OK** to open the schematic window.

> **Note**
> The Schematic Wizard starts if you have checked the *Enable the Schematic Wizard* check box. For more details on design creation ,see *Creating Designs* (usrguide).

## Creating a New Layout

To create a new Layout, follow the steps below:

1. Start ADS and open an existing workspace, or create a new workspace.
2. From the ADS Main window, choose **File > New > Layout** to open New Layout dialog.



Library: SimModels_lib

Cell: cell_17

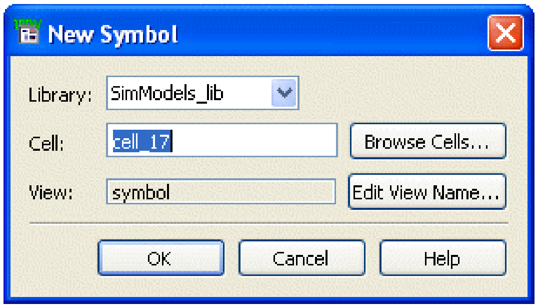
View: layout

3. From the **Library** drop down list, select the library name where the new layout will be stored.
4. Enter the new cell name or click the **Browse Cells** button to select cell from the existing cells of the selected library.
5. Click **Edit View Name** to create a new view.
6. Click **OK** to open the layout window.

## Creating a New Symbol

To create a new Symbol, follow the steps below:

1. Start ADS and open an existing workspace, or create a new workspace.
2. From the ADS Main window, choose **File > New > Symbol** to open the New Symbol dialog.

2. 

Library: SimModels_lib

Cell: cell_17

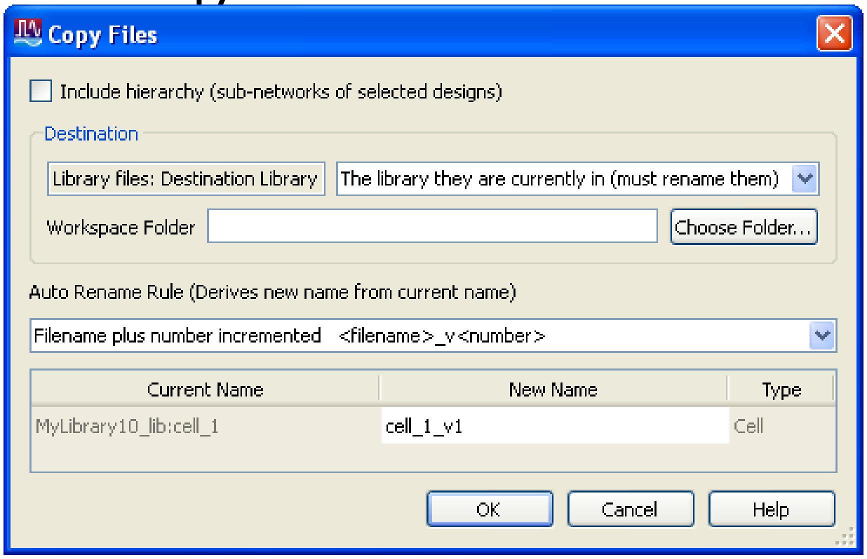
View: symbol

3. From the **Library** drop down list, select the library name where new symbol will be stored.
4. Enter the new cell name or click **Browse Cells** to select cell from the existing cells of the selected library.
5. Click **Edit View Name** to create a new view.
6. Click **OK** to open the symbol window.

## Copying a Design

To copy a Design:

1. Open a Workspace, for example, `WorkspaceA`.
2. From ADS Main Window, click **File > Manage Libraries**.
3. Click **Add Library Definition file** and choose `lib.defs` file of other workspace, say `WorkspaceB`. Click **Close**.
4. From ADS Main Window, select **Folder View** tab. Right click on the cell to be copied and select **Copy Cell**.



☐ Include hierarchy (sub-networks of selected designs)

Library files: Destination Library: The library they are currently in (must rename them)

Workspace Folder:

Auto Rename Rule (Derives new name from current name): Filename plus number incremented <filename>_v<number>

| Current Name | New Name | Type |
|---|---|---|
| MyLibrary10_lib:cell_1 | cell_1_v1 | Cell |

5. In Copy Files dialog box, choose the destination library and click **OK**. This copies the entire cell in selected destination.

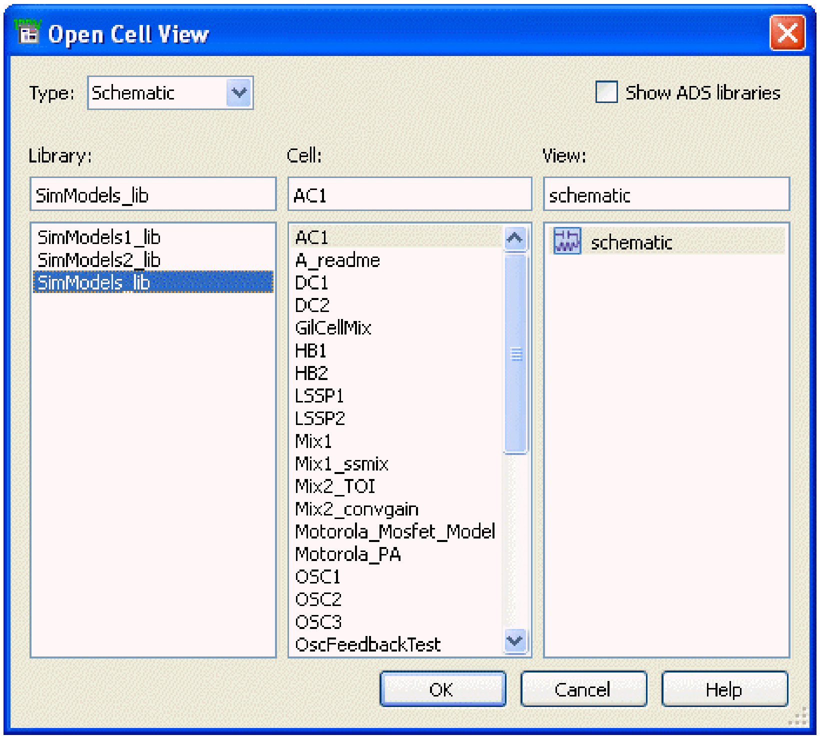
> ℹ After copying the cell, you should remove the `lib.defs` file of `WorkspaceB` through Manage Libraries dialog box. For instructions of how to remove a library, see *Remove Library* (adstour)

## Opening a Design

To open a Design, follow the steps below:

1. Start ADS and open an existing workspace.
2. From the ADS Main window, choose **File > Open > Schematic** to open the Open Cell View dialog box.

> To open a Symbol, Layout, EM Model or EM Setup View, choose **File > Open > Symbol/Layout/EM Model/ or EM Setup View respectively**.



Open Cell View

Type: Schematic

□ Show ADS libraries

Library: SimModels_lib

SimModels1_lib
SimModels2_lib
SimModels_lib

Cell: AC1

AC1
A_readme
DC1
DC2
GilCellMix
HB1
HB2
LSSP1
LSSP2
Mix1
Mix1_ssmix
Mix2_TOI
Mix2_convgain
Motorola_Mosfet_Model
Motorola_PA
OSC1
OSC2
OSC3
OscFeedbackTest

View: schematic

schematic

3. Select the type from drop-down list.
4. If you want to open a built-in ADS design (read-only), check the **Show ADS Libraries** checkbox to display the list of all libraries under **Library**.
5. Under **Library**, select the Library name where the design exists.
6. Under **Cell**, select the cell name.
7. Under **View**, select the view name which could be symbol, schematic, or layout.
8. Click **OK** to open the selected design.

# Substrates in EM Simulation

A substrate in EM simulation describes the media where a circuit exists. An example is the substrate of a multilayer circuit board, which consists of layers of metal traces, insulating material, ground planes, *vias* that connect traces, and *air* that surrounds the board. A substrate definition enables you to specify properties, such as, the number of layers in the substrate, the dielectric constant, and the height of each layer for your circuit.

A substrate consists of the following types of alternating items:

- *Substrate Layer*: This layer defines the dielectric media, ground planes, covers, air, or other layered material.
- *Interface Layer*: This is the conductive layer in between the substrate layers, which is used in conjunction with the layout layers. By mapping layout layers to interface layers, you can position the layout layers that your circuit is drawn on within the substrate.

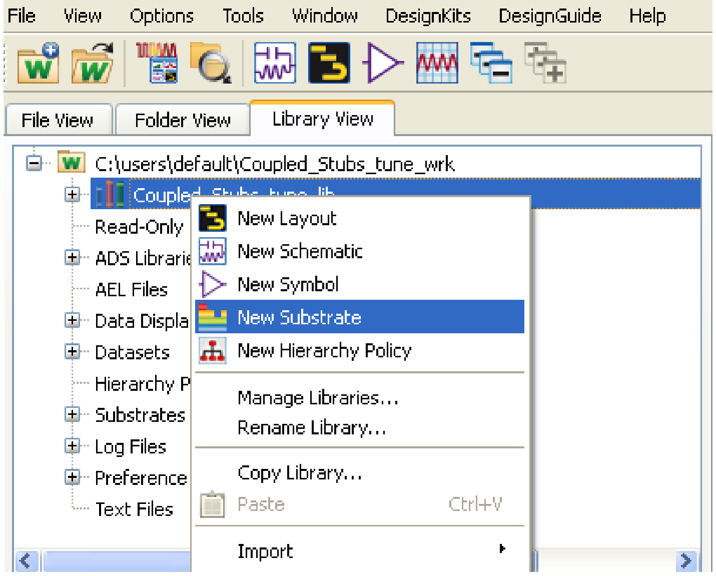
The top and bottom of the substrate either end with a Cover (Interface) or an infinitely thick Substrate Layer.

This section provides information about creating, modifying, and editing a substrate.

## Substrate Editor

You can open the Substrate Editor window in the following ways:

- From the ADS Main Window, choose **File > New > Substrate** and click **OK**.
- From the ADS Main Window, select **Library View** tab. Right-click any library or cell and choose **New Substrate**.

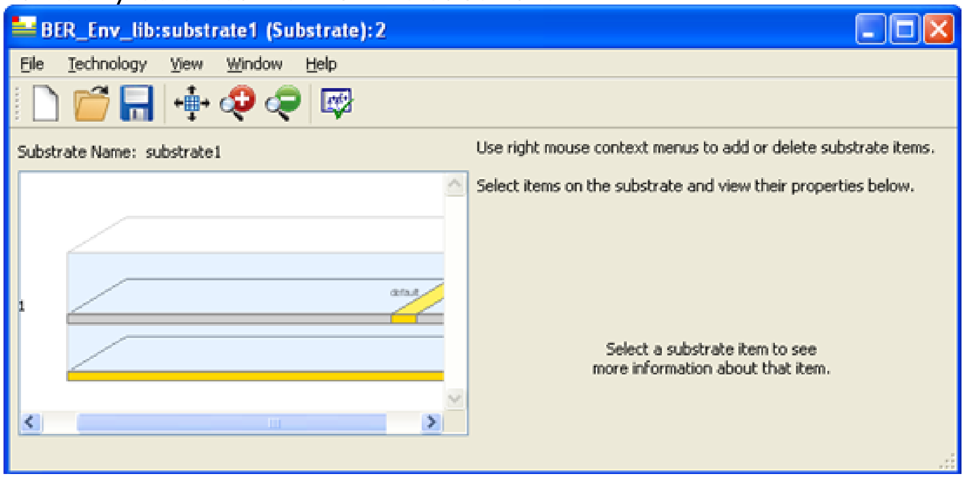


- From the Layout Window, choose **EM > Substrate**.

The key components of substrate editor are listed below:

1. **Main Menu bar**: Contains menu options to edit or create a new substrate.
2. **Toolbar**: Contains the most commonly used buttons.
3. **Substrate view**: Displays 3D cross-section view of substrate stack with mask mappings, it has basic operations to edit the substrate definition.

4. **Status bar**: Notifies about warnings or errors for the substrate.
5. **Properties panel**: This panel, on the right, allows editing the properties of the currently selected item of the substrate.



Substrate Name: substrate1

Use right mouse context menus to add or delete substrate items.

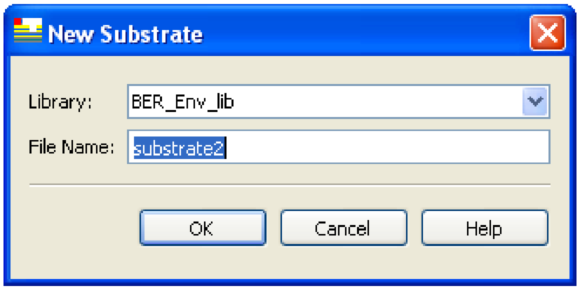
Select items on the substrate and view their properties below.

Select a substrate item to see more information about that item.

## Creating a Substrate

To create a new substrate:

1. Choose **File > New** from the Substrate window or **File > New > Substrate** from the ADS Main window.



Library: BER_Env_lib

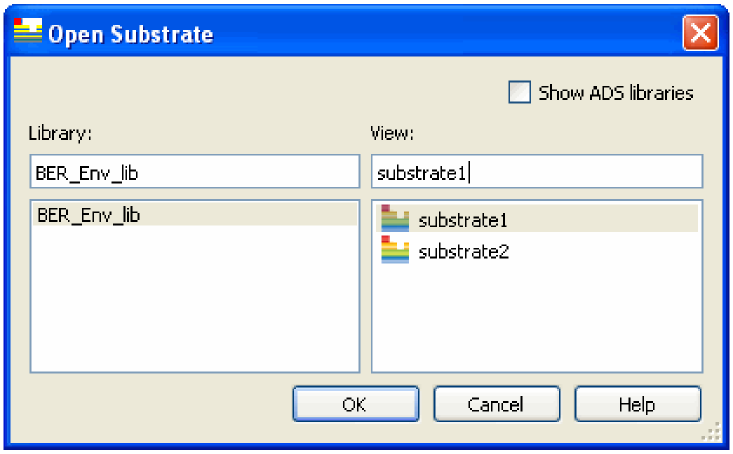
File Name: substrate2

2. From the New Substrate window, select the library where you want to create the substrate.
3. Type the substrate name in *File Name* and click **OK**.

## Opening a Substrate

To open a predefined substrate follow the step below:

1. Choose **File > Open** from the Substrate Window or choose **File > Open > Substrate** from the ADS Main Window.



□ Show ADS libraries

Library: BER_Env_lib

View: substrate1

BER_Env_lib

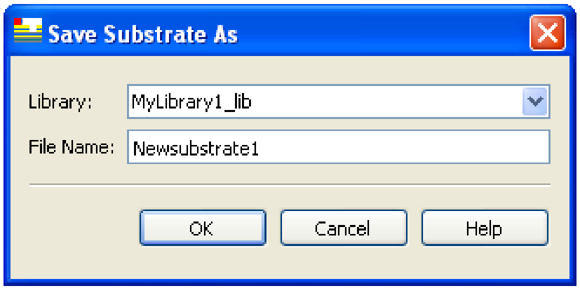
substrate1
substrate2

2. From the Open Substrate window, select the substrate and Click **OK**. Selected substrate opens in a new window.

## Saving a Substrate

Substrate Editor provides three options to save:

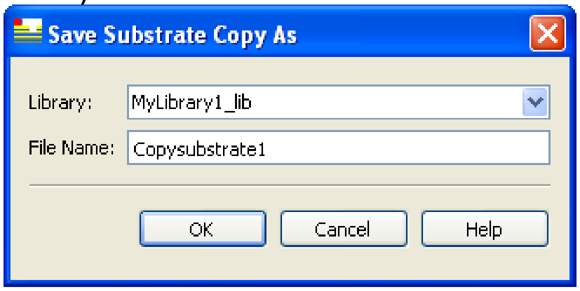
1. Save: This option saves the changes in the current substrate.
2. Save As: The **Save As** command allows you to save the current substrate with a new name. Select the library from the *Library* drop-down list and type the File Name of the substrate. The specified substrate is created in the selected library and displayed in the Substrate Editor.



Library: MyLibrary1_lib

File Name: Newsubstrate1

3. Save a Copy As: The **Save a Copy As** command allows you to save a copy of the current substrate. Select the library from the *Library* drop-down list and type the File Name of the substrate. A copy of the current substrate is created in the specified library.

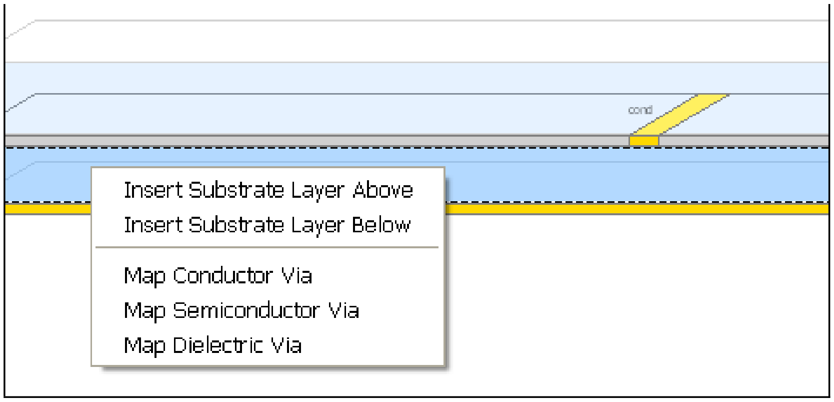


Library: MyLibrary1_lib

File Name: Copysubstrate1

## Inserting, Moving, and Deleting Items

The Substrate View enables you to visualize the substrate stack and do basic editing. To add or delete an item in the substrate, right-click in the substrate view and select from the list of option displayed in the pop-up menu. After selecting the desired action, the properties associated with it are displayed in the right panel of the Substrate Editor.



## Context menus

Right-click on a Substrate Layer and you may see some of the following menus depending on the interface position and properties:

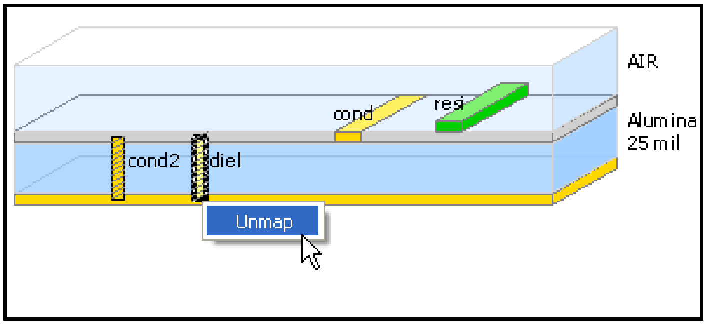
- *Insert Substrate Layer Above* - Inserts a new substrate layer with an interface layer above the selected layer.
- *Insert Substrate Layer Below* - inserts a new substrate layer with an interface layer below the selected layer.
- *Delete with Upper Layer* - Deletes the substrate layer above the selected layer.
- *Delete with Lower Layer* - Deletes the substrate layer below the selected layer.
- *Map Conductor Via* - Inserts a new conductor via in the selected substrate.
- *Map Semiconductor Via* - Inserts a new semiconductor via in the selected substrate.
- *Map Dielectric Via* - Inserts a new dielectric via in the selected layer.
- *Move Up With Upper Interface* - Moves the Substrate Layer and the Interface above it up, along with items on that interface.
- *Move Up With Lower Interface* - Moves the Substrate Layer and the Interface below it up, along with items on that interface.
- *Move Down With Upper Interface* - Moves the Substrate Layer and the Interface above it down, along with items on that interface.
- *Move Down With Lower Interface* - Moves the Substrate Layer and the Interface below it down, along with items on that interface.

Depending on the position of the substrate layer you can move the layer up or down the stack. If the layer is either at the top or bottom, you can add cover above or below the substrate layer, as applicable.

Right-click on a Interface Layer and you may see some of the following menus depending on the interface position and properties:

- *Map Conductor Layer* - Inserts a new conductor layer on the selected interface.
- *Map Semiconductor Layer* - Inserts a new semiconductor layer on the selected interface.
- *Map Dielectric Layer* - Inserts a new dielectric layer on the selected interface.
- *Insert Nested Substrate* - Inserts a new Nested Substrate on the selected interface.
- *Delete Cover* - Deletes the Cover leaving the adjacent Substrate Layer as an infinite thickness layer.

To unmap the already mapped item, right-click on a *Conductor Layer* or *Via* and select **Unmap** option from the pop-up menu.



## Moving Conductor Layers

Use the left mouse button to move a Conductor Layer up or down to a different interface or to drag the item to the new location. This method also works for Semiconductor Layers, Dielectric Layers, and Nested Substrates.

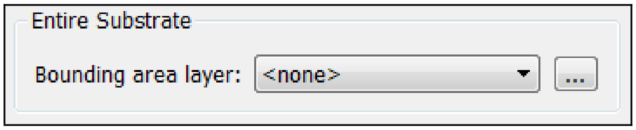
## Moving Vias

Using the left mouse button, drag a via on the upper or lower 1/3 of its body and you will be able to stretch the via so that it goes through more or less Substrate Layers. Dragging it from the middle of its body allows you to move the via up or down without stretching it.

## Editing Substrate Properties

### Editing Properties for the Entire Substrate

Click in the background of the substrate to deselect any specific item. This allows you to edit the properties of the entire substrate on the right panel of the window.



Entire Substrate

Bounding area layer: <none>

The **bounding area layer** is a layout layer specifying an area delimiting the design. It specifies the extent of substrate layers, slot layers and covers for simulators operating in a finite simulation domain. This includes the finite element simulator and exporting to EMPro. It excludes Momentum, which will continue extending these layers to infinity.

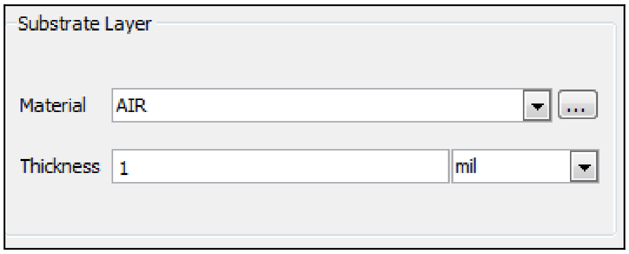
All layout outside the bounding area layer will be discarded.

The bounding area layer can be selectively overridden on substrate and interface layers in their respective property sheet. Note that individual vias can only be discarded as a whole; having them simultaneously inside the bounding area layer of one substrate layer and outside the bounding area layer of another substrate layer in unsupported and leads to undefined behavior.

The top level bounding area layer definition does not descend into nested substrates. However, the shared interface layer between a substrate and a nested substrate uses the Boolean OR of all defined bounding areas. In this respect, the absence of a bounding area layer definition in one of the substrates is being treated as if the entire infinite plane is to be used as bounding area.

### Editing Substrate layer Properties

Select a substrate layer to display and edit the properties listed on the right panel of the window.



Substrate Layer

Material AIR

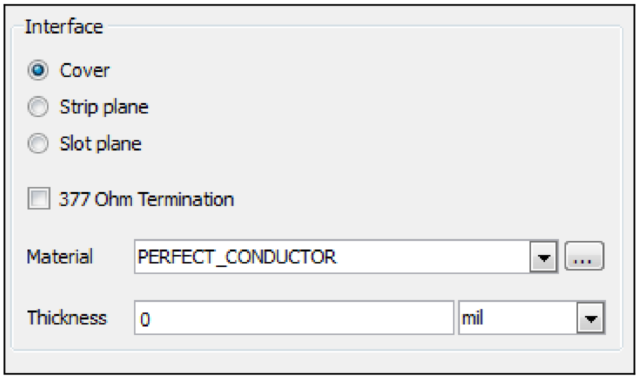
Thickness 1 mil

Following are the properties that can be edited:

- **Material** - This property allows you to select the layer material from the *Material* drop-down list. Materials are defined in the Materials Definition dialog box. Click the … button to open the Material Definition dialog box where you can define a new material. The defined material is added automatically in the *Material* drop-down list.
- **Thickness** - This property allows you to define the thickness of the layer. The units can be selected from the *Thickness* drop-down list.

## Editing Interface Layer Properties

Select an Interface layer to define the interface layer as one of the following:



Interface

● Cover

○ Strip plane

○ Slot plane

□ 377 Ohm Termination

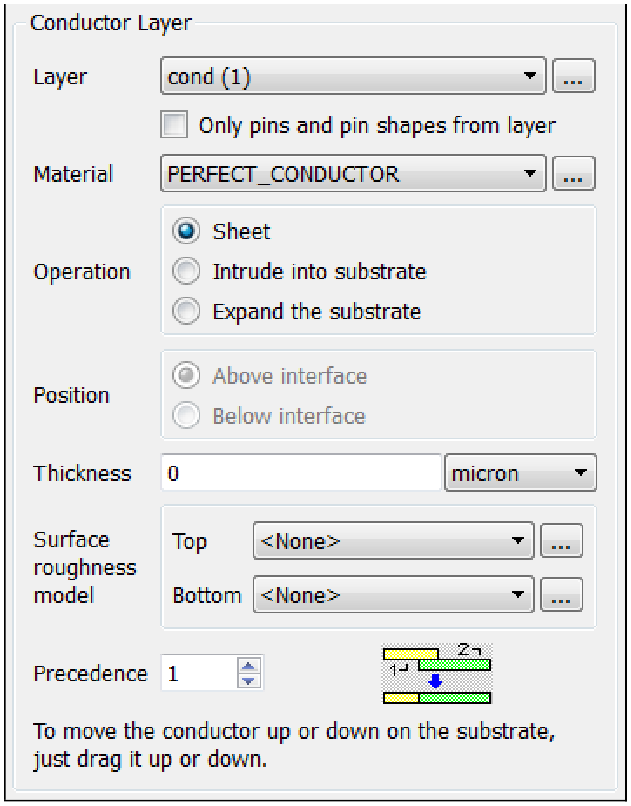
Material PERFECT_CONDUCTOR

Thickness 0 mil

- **Cover** - This option is only available for the top or bottom Interface. If it is enabled, the following properties can also be set:
  - **377 Ohm Termination** - Check the box to enable this termination. If this option is checked, you cannot specify **Material** and **Thickness**.
  - **Material** - This property allows you to select the layer material from the *Material* drop-down list. Materials are defined in the Materials Definition dialog box. Click the … button to open the Material Definition dialog box where you can define a new material. The defined material is added automatically in the *Material* drop-down list.
  - **Thickness** - This property allows you to define the thickness of the layer. The units can be selected from the *Thickness* drop-down list.
- **Strip Plane** - Allows the mapping of Conductor, Semiconductor, and Dielectric Layers, and the insertion of Nested Substrates.
- **Slot Plane** - Allows the mapping of Slot Layers.

## Editing Conductor Layer Properties

Select a Conductor layer to define any of the following properties:



Conductor Layer

Layer: cond (1) ...

□ Only pins and pin shapes from layer

Material: PERFECT_CONDUCTOR ...

Operation:
● Sheet
○ Intrude into substrate
○ Expand the substrate

Position:
● Above interface
○ Below interface

Thickness: 0 micron

Surface roughness model:
Top: <None> ...
Bottom: <None> ...

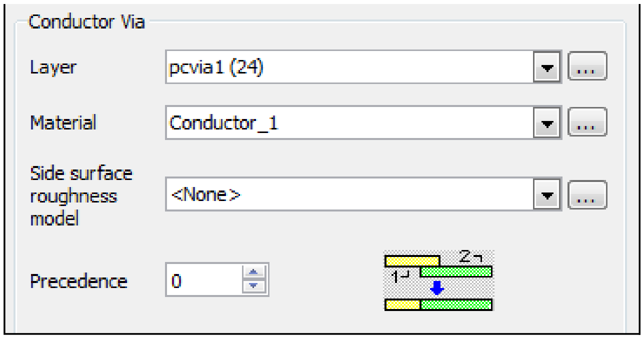
Precedence: 1

To move the conductor up or down on the substrate, just drag it up or down.

- **Layer** - Allows you to map the mask layer with layout layer from the *Layer* drop-down list. To add new layout layer click the ... button (next to *Layer* drop-down list).
- **Only pins and pin shapes from layer** - Allows you to only map the pins and pin shapes into the substrate, but not the geometry.
- **Material** - Defines material property for the mask layer from the *Material* drop-down list. To add new material layer click the ... button (next to *Material* drop-down list).
- **Operation** - The operation transforms 2D shapes drawn on a mask into 3D objects. For example, select the proper expand operation to define the thickness of a conductor mask.
- **Position** - Defines the position of the layer.
- **Thickness** - Defines the thickness of the layer.
- **Surface roughness model** - Allows you to select Surface roughness model at Top and Bottom.
- **Precedence** - Precedence specifies the precedence of a layout layer over another layer, if two or more layout layers are assigned to the same interface or substrate layer and objects overlap. Precedence is used by the mesh maker so that objects on the layer with the greatest precedence number are meshed and any overlap with objects on layers with lesser numbers are logically subtracted from the circuit. If you do not set the precedence, and there are overlapping objects, a mesh will automatically and arbitrarily be created, with no errors reported.

## Editing Via Properties

Select a Conductor Via layer to edit any of the following properties:



Conductor Via

Layer: pcvia1 (24)

Material: Conductor_1

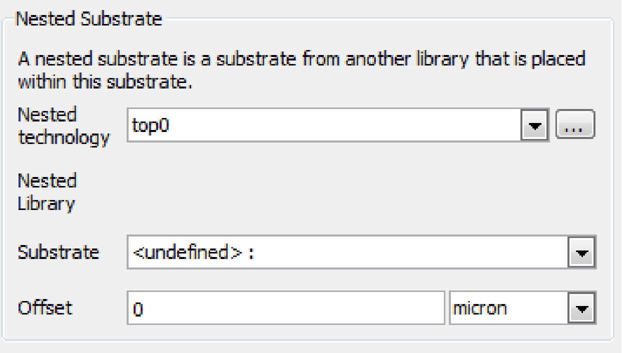
Side surface roughness model: <None>

Precedence: 0

- **Layer** - Allows you to map the mask layer with layout layer from the *Layer* drop-down list. To add new layout layer click the ... button (next to Layer drop-down list).
- **Material** - Defines material property for the mask layer from the *Material* drop-down list. To add new material layer click the ... button (next to Material drop-down list).
- **Surface roughness model** - Allows you to select Surface roughness model for the sides of the via.
- **Precedence** - Precedence specifies which layout layer has precedence over another if two or more layout layers are assigned.

## Editing Nested Substrate Properties

Select a Nested Substrate to edit any of the following properties:



Nested Substrate

A nested substrate is a substrate from another library that is placed within this substrate.

Nested technology: top0

Nested Library

Substrate: <undefined> :

Offset: 0 micron

- **Nested Technology** - Choose a Nested Technology from the *Nested Technology* drop-down list. The chosen Nested Technology determines the position of the Nested Substrate above or below the Interface it is on. It also determines if the substrate appears flipped or not. Click the ... button to create or edit Nested Technologies.
- **Nested Library** - This is the name of the library specified in the chosen Nested Technology. This is not editable.
- **Substrate** - Choose a Substrate from the Nested Library to specify the EM properties of the layouts that will be placed on layouts using this substrate.
- **Offset** - This allows you to move your Nested Substrate up or down relative to the interface it is on.

Normally a Nested Substrate is placed above the top interface or below the bottom interface. See *Nested Technology* (usrguide) and Multi-Technology Design in ADS2011 for more information about using Nested Technologies and Nested Substrates.

## Verifying Substrate Definition

After creating the substrate definition, choose **File > Check** (from the Substrate Editor window) to verify the created substrate definition. The verification result or errors (if any) are displayed in the Check Substrate message window.

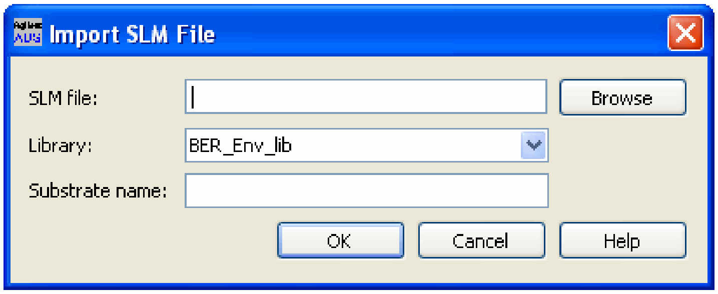
## Importing a Substrate

To import a substrate choose **File > Import** from the Substrate editor window or the ADS Main window. The following import options are available:

- SLM Substrate File
- ltd Substrate File
- Substrate From Database
- Substrate From Schematic

## SLM Substrate File

To import substrate from a slm file, choose **SLM Substrate file**.



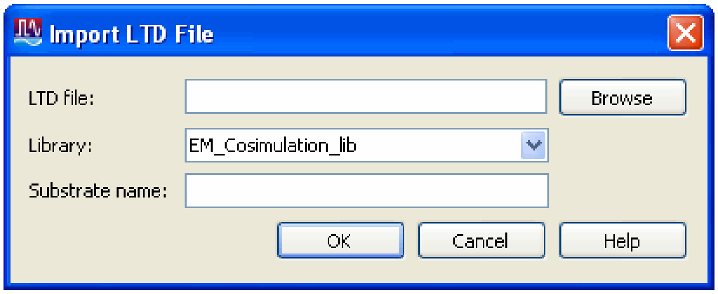
SLM file:

Library: BER_Env_lib

Substrate name:

## ltd Substrate File

To import substrate from an ltd file, choose **ltd Substrate file**.



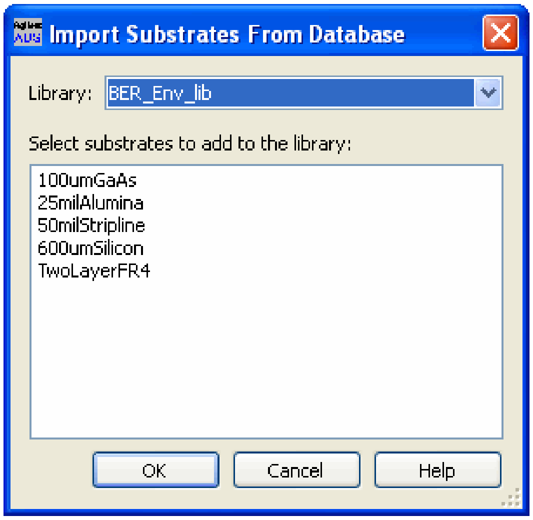
LTD file:

Library: EM_Cosimulation_lib

Substrate name:

## Substrate From Database

To import substrate from a database, choose **Substrate From Database**.



Import Substrates From Database

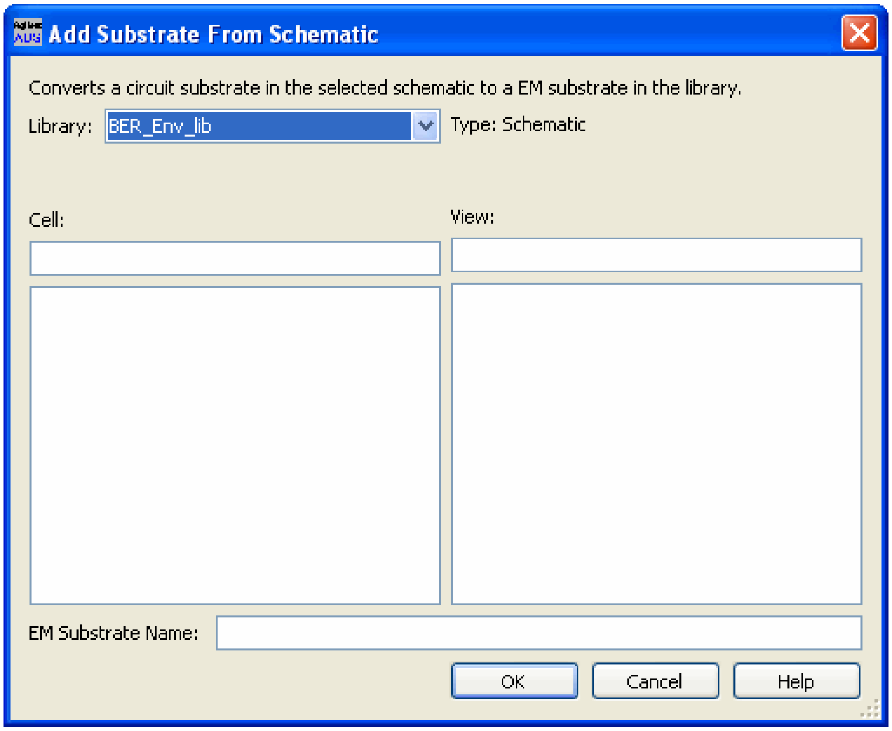
Library: BER_Env_lib

Select substrates to add to the library:

100umGaAs
25milAlumina
50milStripline
600umSilicon
TwoLayerFR4

## Substrate From Schematic

To import a substrate from a schematic, choose **Substrate From Schematic**.



Add Substrate From Schematic

Converts a circuit substrate in the selected schematic to a EM substrate in the library.

Library: BER_Env_lib Type: Schematic

Cell:

View:

EM Substrate Name:

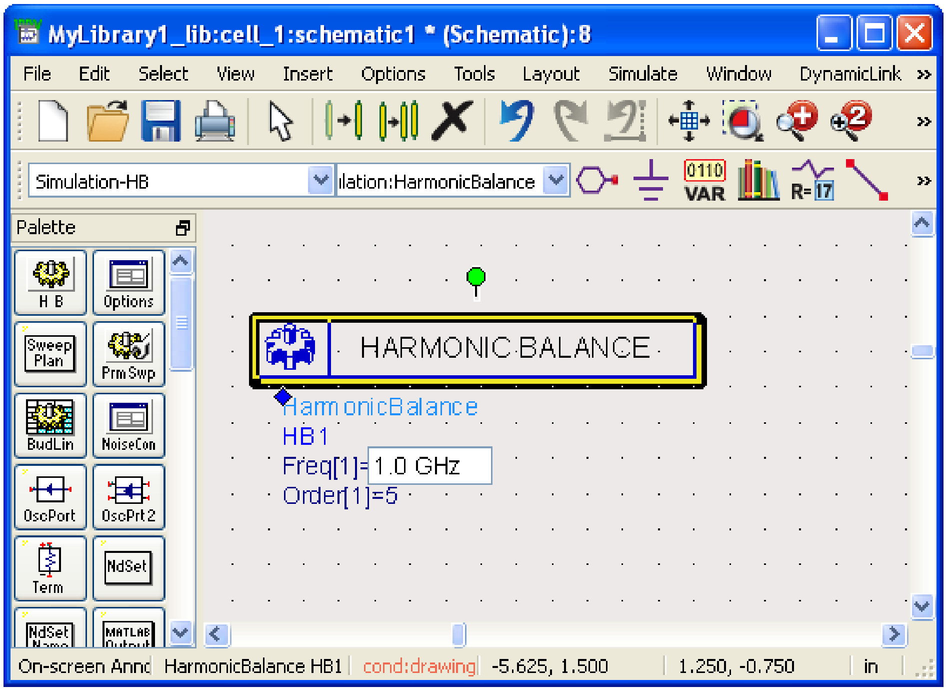
# Simulating Designs

Advanced Design System provides controllers that you can add and configure to simulate, optimize, and test your designs.

A DSP design simulation requires a Data Flow Controller while an Analog/RF design simulation requires one or more of various controllers. You can either add and configure the appropriate controllers or you can insert a template (choose **Insert > Template** from a Schematic window) that contains the appropriate controllers.

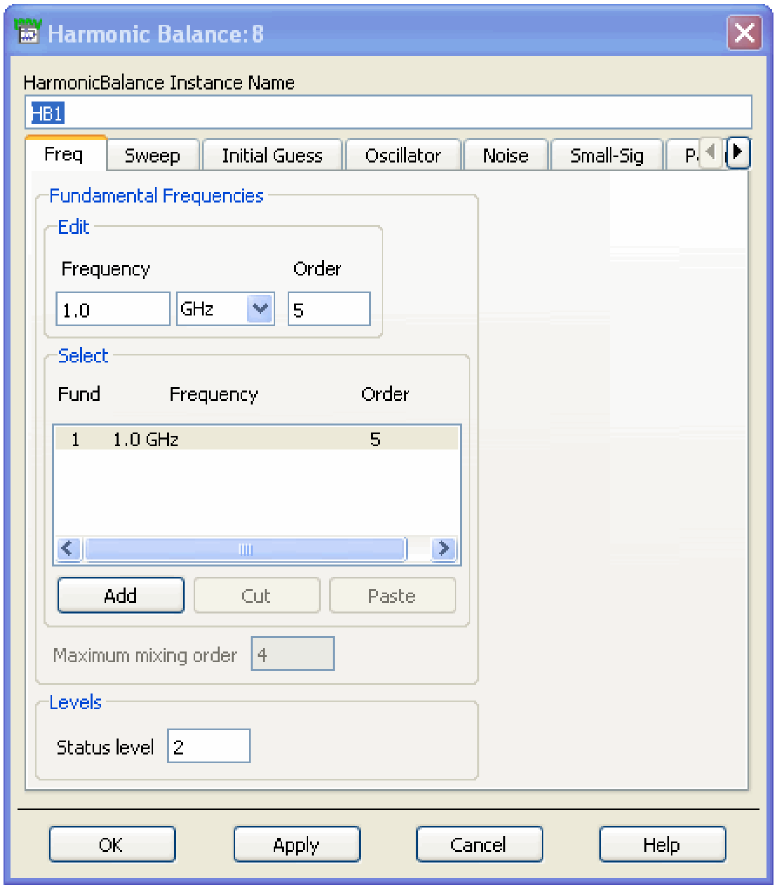
To simulate a design:

1. Click and place the simulation controller.
2. Double-click to edit parameters.



HarmonicBalance
HB1
Freq[1]=1.0 GHz
Order[1]=5

2.



HarmonicBalance Instance Name: HB1

Fundamental Frequencies

Edit

| Frequency | | Order |
|---|---|---|
| 1.0 | GHz | 5 |

Select

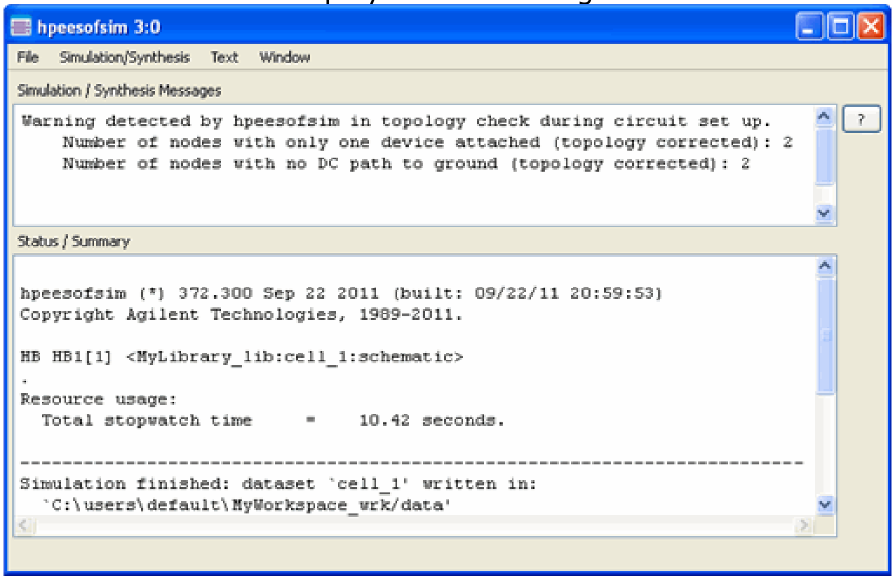
| Fund | Frequency | Order |
|---|---|---|
| 1 | 1.0 GHz | 5 |

Maximum mixing order: 4

Levels

Status level: 2

3. From the schematic window, click the Simulate icon to begin the simulation. The simulation status is displayed in a message window.



Simulation / Synthesis Messages

```
Warning detected by hpeesofsim in topology check during circuit set up.
    Number of nodes with only one device attached (topology corrected): 2
    Number of nodes with no DC path to ground (topology corrected): 2
```

Status / Summary

```
hpeesofsim (*) 372.300 Sep 22 2011 (built: 09/22/11 20:59:53)
Copyright Agilent Technologies, 1989-2011.

HB HB1[1] <MyLibrary_lib:cell_1:schematic>
.
Resource usage:
  Total stopwatch time     =    10.42 seconds.

--------------------------------------------------------------------------
Simulation finished: dataset `cell_1' written in:
  `C:\users\default\MyWorkspace_wrk/data'
```

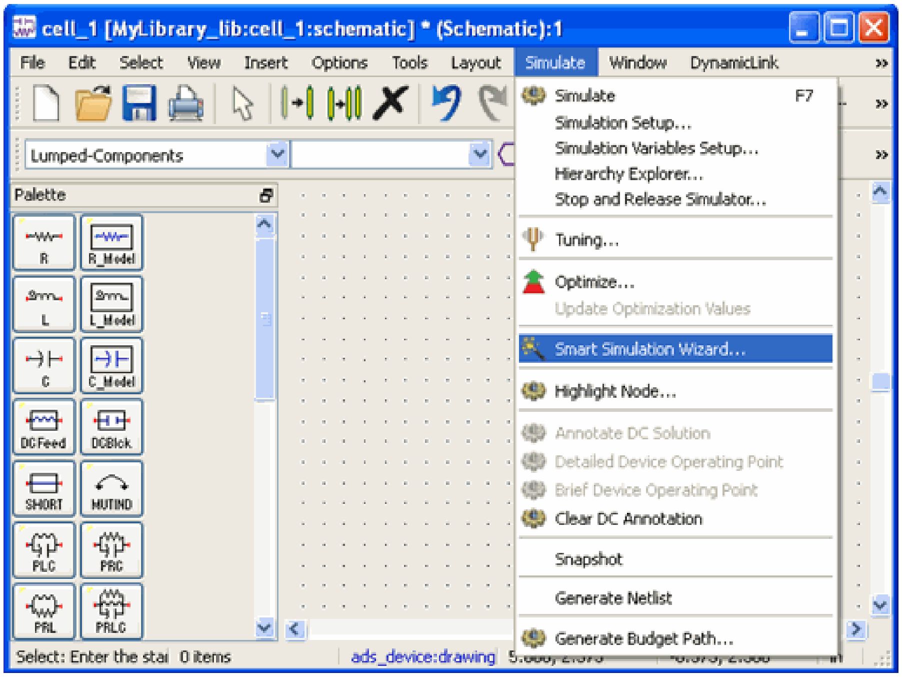
## Simulation Wizard

Advanced Design System also provides a step-by-step interface for circuit simulation. The *Smart Simulation Wizard* can be used to:

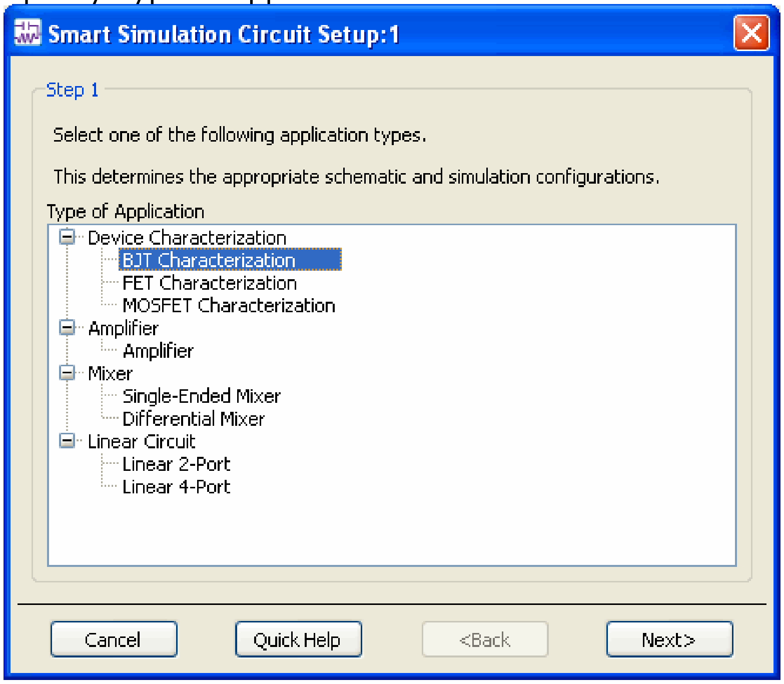
- Create Analog/RF designs
- Set up and run simulations
- Display simulation results

### To smart simulate a design...

1. Choose **Simulate > Smart Simulation Wizard**.



2. Specify Type of Application.



Select one of the following application types.

This determines the appropriate schematic and simulation configurations.

3. Specify simulation options.
4. Display results.

## Signal Processing Simulation

ADS provides an integrated environment for the design and validation of RF/analog/DSP system designs to the implementation level using the *ADS Ptolemy simulator* (ptolemy).

The ADS signal processing environment enables:

- Accurate RF system models for faster development of system specifications

- Extensive behavioral model set for RF system and DSP system modeling that helps engineers rapidly create and optimize larger designs
- Co-design between DSP, analog and RF portions of the signal path
- Hundreds of DSP and analog models for development of algorithms
- Propagation and matrix models that allow modeling of complete wireless systems
- Data export and import capability to measurement instrumentation to verify designs using virtual prototyping concepts
- IP reuse of MATLAB, HDL, & C++ models

The systems designer can architect a communications system using behavioral models to validate a concept. The designer can then design and substitute lower levels of abstraction to verify the RF/mixed signal design down to the circuit level, and export the design to a variety of manufacturing tools. Available statistical design capability allows the user to make difficult trade-offs during the design process in order to optimize performance or manufacturing yield.

A large array of behavioral RF/analog/DSP models works with the ADS Ptolemy simulator to provide leading-edge simulation accuracy during the design process. The inclusion of propagation and matrix models, facilitate modeling of the complete wireless system. ADS communications library modules support the latest communications standards such as WLAN, 3GPP, and EDGE. These libraries can be used at the front end of the design process when the system architecture is conceptualized, during the design and implementation process, or at the back end of the design process during the final verification.

Instrument links to Agilent Technologies test and measurement instrumentation products provide virtual prototyping verification for designs prior to final implementation or tape out. For example, a new RF/analog/DSP transmitter design modeled in the signal processing schematic can be verified by linking the output of the simulation with one of the Agilent ESG signal generator products. The resulting real world signal produced in a virtual environment will include all of the signal distortions, noise, and propagation effects modeled into the design. This signal can then be fed into an Agilent signal analysis component or real-world receiver circuit to provide virtual prototyping capability, and the ability to tune the design using real-world hardware and analysis.

ADS Ptolemy simulation is controlled using a *Data Flow Simulation Controller* (adstour), sources, and sinks placed on the design. There must be at least one source or sink that is controlling the simulation. Controlling sinks and sources keep the simulation running; non-controlling sinks and sources do not.

## Sources

Sources are components with no inputs. Sources can read data from files, instruments, and datasets. When a source is controlling the simulation, it will keep the simulation running long enough to output all its data.

## Sinks

Sinks are components with no outputs. When a sink controls the simulation, it will keep the simulation running long enough to satisfy its start and stop times. When a sink is not controlling the simulation, it will start collecting data at Start, then collect as much data as the simulation produces.

## Components

There are two basic types of Ptolemy components *Timed* and *Numeric*. Timed components have a notion of sampling rate, carrier frequency, and envelope. Numeric components process integers, matrixes, floats, fixed point numbers and model the DSP portions of a design.
For more information on:

- Cosimulation with analog/RF designs, refer to the Cosimulation section of *ADS Ptolemy Simulation* (ptolemy).
- Connecting to instruments, refer to *Connection Manager* (connectmui).
- Cosimulation with MATLAB IP import, refer to *MATLAB Cosimulation Introduction* (ptolemy).
- Cosimulation and HDL IP import, refer to *HDL Cosimulation* (hdlcosim).
- C++ IP import, refer to *User-Defined Models* (modbuild).

## Analog/RF Simulation and Convergence

Analog/RF simulation computes the response of a circuit to a particular stimulus by formulating a system of circuit equations and then solving them numerically. Each simulation technology accomplishes this analysis as follows.

## DC Analysis

- Solves a system of nonlinear ordinary differential equations (ODEs)
- Solves for an equilibrium point
- All time-derivatives are constant (zero)
- System of nonlinear algebraic equations

## Transient Analysis

- Solves a system of nonlinear ordinary differential equations (ODEs)
- Time-derivatives replaced with a finite-difference approximation (integration method)
- Sequence of systems of nonlinear algebraic equations (one system at each timepoint)

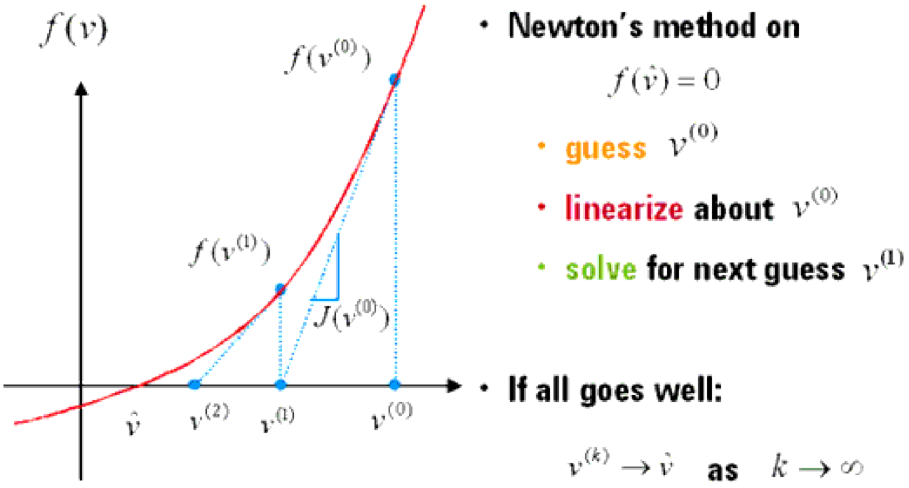
## Harmonic Balance (HB)

- Solves a system of nonlinear ordinary differential equations (ODEs)
- Steady-state method
- Solution approximated by truncated Fourier series
- System of nonlinear ODEs becomes a system of nonlinear algebraic equations in the frequency domain

## Solving Nonlinear Algebraic Equations

Nonlinear algebraic equations are solved using the Newton-Raphson algorithm (Newton's method) as follows.

- Convert the problem to a sequence of systems of linear equations
- Quadratic convergence near the solution (error squared at each iteration)



- Newton's method on $f(\hat{v}) = 0$
  - guess $v^{(0)}$
  - linearize about $v^{(0)}$
  - solve for next guess $v^{(1)}$
- If all goes well:

  $v^{(k)} \to \hat{v}$ as $k \to \infty$

## S-parameter Test Lab

An S-parameter test lab enables you to calculate the S-parameters of multiple N-port networks in a single simulation run.

An S-parameter test lab is a schematic that contains one S-parameter test lab component and one or more test benches. A test bench is a schematic that contains an N-port network and terminations for each port of the network. Its use is best illustrated in multiple stage circuit designs where viewing the inter-stage circuit behavior of all stages simultaneously is desired. In such situations the S-parameter test lab can be used to terminate each stage in the applicable input/output impedances of adjacent stages rather than in the standard 50 ohms.

RefNets can also be used in conjunction with the S-parameter test lab feature.

## Design Sequencer

A Design Sequencer controller enables you to sequence multiple simulations in a single simulation run using a test bench that includes all the desired simulation controllers and the top-level design file.
Some typical applications for a Sequencer controller are as follows.

- Optimizing a variable across multiple simulations
- Enabling complex instrument control in Ptolemy
- Running a series of verifications tests on a design

### Differences Between S-parameter Test Labs and Sequencer

| Sequencer | Test Lab |
|---|---|
| DC, SP, AC, HB, Tran, ENV, Ptolemy | SP only |
| Utilizes Test Bench Controllers | Utilizes Test Lab Controller |
| Different temps per test bench possible | One simulation temp for all |
| Opt/Stat/ParamSwp at top level | |
| RefNets supported | |

## RefNets

A RefNet (reference network) component enables the port impedance from another design in the workspace (the referenced network) to be referenced as a terminating impedance for the current design under test.

Two typical applications for RefNets are as follows.

1. **Inter-stage circuit analysis and design**: In some design applications it is desirable to simultaneously evaluate the performance of individual circuit stages terminated in the input and output impedances of adjacent stages. To accomplish the termination of an individual stage referenced to a specific port of other stages in the design chain, the RefNet is utilized in the S-parameter test lab.
2. **Design specific termination**: For some top level DC, AC, or S-parameter design files, it may be desired to terminate a port whose impedance is characterized by data, from an external file (e.g. S-parameters, Z-parameters, Y-parameters) or some other network.

The two RefNet components, *RefNetTB* and *RefNetDesign*, have the same functionality and are supported under DC, AC and S-parameter analysis, with two differences:

- RefNetTB supports nested network referencing while RefNetDesign does not.
- RefNetTB uses a test bench as the reference design while RefNetDesign uses a standard (non-test-bench) schematic design.

## Common Circuit Simulation Methods

## Backward Euler

- First order method that assumes the solution waveform is linear over one time step
- One-step method (needs one previous time point solution only)
- Adapts faster to abrupt signal changes
- Stable on all stable differential equations and some unstable ones.
- Exhibits heavy numerical damping, increases loss
- Require smaller time step to maintain accuracy

## Trapezoidal Rule

- Second-order method, assumes the solution waveform is quadratic over one time step
- One-step method

- May exhibit point-to-point ringing on circuits that have very small time constant comparing to time step (stiff circuit)
- Stable only on stable differential equations
- Exhibits no artificial numerical damping

## Backward Difference Formulas (Gear's methods)

- Multiple order polynomial over one time step
- Only the first six orders are available in ADS
- First order method is identical to backward Euler
- Higher-order polynomials allow a larger time step without sacrificing accuracy, are efficient for smooth waveforms
- Higher order methods (order > 2) may exhibit stability problems on lightly damped circuits
- Second-order backward difference formula (Gear 2)
- Two-step method
- Stable on all stable differential equations and some unstable ones.
- Exhibit some numerical damping

## Truncation Error

The error made by replacing the time derivatives with a discrete-time approximation. This error is difficult to estimate and depends on the type of circuits and the time steps.

## Local Truncation Error (LTE)

The truncation error made on a single step

## Global Truncation Error (GTE)

- Maximum accumulated truncation error
- The circuit with long time constant is sensitive to these errors
- Logic and bias circuits are not sensitive to these errors

## Convergence Criteria

Newton's iteration is converged if the approximate solution first satisfies the Residue criteria at the end of each Newton iteration and the Update criteria once the residue criteria are satisfied.

## Residue Criterion

KCL satisfied to a given tolerance. This is enforced at each node and is important when impedance at a node is small.

## Update Criteria

Difference between the last two iterations must be small. This is important when impedance at a node is large.

## Using Continuation Methods

Use continuation methods to provide a sequence of initial guesses that are sufficiently close to the solution to assure Newton's method convergence.

- Choose a natural or contrived continuation parameter which controls a modification of the circuit
- Step the continuation parameter from 0 to 1 (the original circuit configuration), using the solution from the previous step as the starting point

As long as the solution changes continuously as a function of the continuation parameter and the steps are small enough, Newton's method will converge. Keep in mind though that the first two methods, Source and gmin stepping, will fail if the continuation path contains a limit point.

## Source Stepping

Uses a fraction of the source voltages and currents applied to the circuit as the continuation parameter.

- Turn off all sources when the continuation parameter equals 0
- Raise source levels to their final levels slowly, generating a sequence of circuit configurations
- Use the solution from the previous configuration as an initial guess for the current configuration

## Gmin Stepping

Uses the continuation parameter to control the value of the gmin resistors.

- Start with a large value of gmin for an easy to compute solution, because nonlinear device behavior is muted by the presence of the small resistors
- End with a very small value of gmin for resistors that are so large that they no longer affect the circuit
- Remove the gmin values to compute the final solution

## Arc-length Continuation

Works best for complicated continuation paths and limit points using a continuation parameter that is a function of the arc-length parameter.

- Travel same distance at each step, as specified by the arc-length
- Increase or decrease the continuation parameter along the path in each step

## Preventing Convergence Problems

Convergence problems usually arise as a result of errors in circuit connectivity or unreasonable (out of range) model or component values. Some of the steps you can take are as follows.

- Turn on the topology checker
- Turn on warnings
- Act upon the messages in the ADS Status Server window
- Eliminate small floating resistors (or increase I_AbsTol) because any error in computed voltages for nodes with small resistors results in large error currents
- Avoid very large and very small resistances connected to a node because large resistances are lost during Jacobian construction due to numerical round-offs

## Momentum Simulation, Optimization, and Visualization

Momentum includes simulation, optimization, and visualization tools for predicting the performance of multilayer high-frequency circuit boards, antennas, hybrids, multichip modules, and integrated circuits.

Momentum enables you to:

- Simulate when a circuit model range is exceeded or the model does not exist
- Identify parasitic coupling between components
- Go beyond simple analysis and verification to design automation of circuit performance
- Visualize current flow and 3-dimensional displays of far-field radiation
  *Momentum* is an electromagnetic simulator that computes S-parameters for general planar circuits, including microstrip, slotline, stripline, coplanar waveguide, and other topologies.
  *Momentum Optimization* varies geometry parameters automatically to help you achieve the optimal structure that meets the circuit or device performance goals.
  *Momentum Visualization* provides a 3-dimensional perspective of simulation results, enabling you to view and animate current flow in conductors and slots, and view both 2D and 3D representations of far-field radiation patterns.

## Instrument Connectivity

Connection Manager enables the sharing of signals, measurements, algorithms, and data between ADS simulations and Agilent instruments (signal generators and signal analyzers).

Using Connection Manager, you can:

- Access and control instruments from ADS dialogs
- Measure devices and construct ADS datasets from the measurement data
- Create simulation models based on measured data
- Use real-time instrument-generated stimulus and measurement during simulations

## Simulation Controllers

Add one or more simulation controllers to the design based upon the type of design to be simulated and the kinds of analyses desired.

| **Description** | **Typical Use** |
|---|---|
| *Data Flow Simulation Controller* (adstour)<br>Controls the flow of mixed numeric and timed signals for digital signal processing simulations using the ADS Ptolemy simulator. | All signal processing designs |
| *DC Simulation Controller* (adstour)<br>Fundamental to all RF/Analog simulations. It performs a topology check and an analysis of the DC operating point. | All RF/Analog designs |
| *AC Simulation Controller* (adstour)<br>Obtains small-signal transfer parameters like voltage gain, current gain, and linear noise voltage and currents. | Filter<br>Amplifier |
| *S-Parameter Simulation Controller* (adstour)<br>Provides linear S-parameter, linear noise parameters, transimpedance, and transadmittance. Can be used to achieve many goals of the AC simulator. | Filter<br>Oscillator<br>Amplifier |
| *Harmonic Balance Simulation Controller* (adstour)<br>Uses nonlinear harmonic-balance techniques to find the steady-state solution in the frequency domain. | Mixer<br>Oscillator<br>Power Amplifier<br>Transceiver |
| *Circuit Envelope Simulation Controller* (adstour)<br>Uses a combination of frequency- and time-domain analysis techniques to yield a fast and complete analysis of complex signals such as digitally modulated RF signals. | Mixer<br>Oscillator<br>Power Amplifier<br>Transceiver<br>Phase-locked Loop |
| *LSSP Simulation Controller* (adstour)<br>Performs large-signal S-parameter analyses to represent nonlinear behavior. The accompanying P2D simulator can be used to speed up subsequent analyses. | Power Amplifier |
| *XDB Simulation Controller* (adstour)<br>Seeks a user-defined gain-compression point at which an actual power curve deviates from an idealized linear power curve. | Power Amplifier<br>Mixer |
| *Transient/Conv. Simulation Controller* (adstour)<br>Solves a nonlinear circuit entirely in the time domain using simplified models to account for the frequency-dependent behavior of distributed elements. | Mixer<br>Power Amplifier<br>Switching Circuits |
| *RF Budget Controller* (adstour)<br>Determines the linear and nonlinear characteristics of an RF system made up of a cascade of two-port, two-pin linear or nonlinear components. | Mixer<br>Nonlinear<br>Amplifier |

## Optimization & Statistical Design Controllers

Optimization and statistical design controllers are used in conjunction with RF/Analog and signal processing simulation controllers to:

- Characterize and improve an unknown process such as the response of a design
- Identify variables that contribute significantly to variations in performance
- Vary parameter values to identify combinations that deliver the desired yields
  Some of their design applications include:
- Optimizing gain and matching
- Filter response optimization
- Pulse-rise time tuning
- Carrier lock time and residual loop error optimization
- Fixed-point bit-width optimization
- Maximize manufacturing yield

Advanced Design System includes the optimization and statistical design controllers shown below. For more detailed information on optimization and statistical design, refer to the *Tuning, Optimization, and Statistical Design* (optstat) documentation.

| **Description** | **Used With** |
|---|---|
| *Nominal Optimization Controller* (adstour)<br>Used to compare computed and desired responses and modify parameter nominal values to bring the computed response closer to the desired optimization goals. | Goal Component (required) A Goal component is used in conjunction to specify the optimization goals. |
| *Monte Carlo Controller* (adstour)<br>Uses the Monte Carlo method to simulate a design over a given number of trials in which the statistical variables have values that vary randomly about their nominal values with specified probability distribution functions. | Yield Specification Component (optional) A Yield Specification component is used in conjunction to specify the desired yields. Statistical Correlation Component (optional) A Statistical Correlation component is used to specify statistical correlation between statistical design variables. |
| *Yield Analysis Controller* (adstour)<br>Uses the Monte Carlo method described above to determine the manufacturing yield. For each trial, the computed response is compared to the corresponding yield specification, and a pass/fail decision is made. | Yield Specification Component (required) A Yield Specification component is used in conjunction to specify the acceptable performance. Statistical Correlation Component (optional) A Statistical Correlation component is used to specify statistical correlation between statistical design variables. |
| *Yield Optimization Controller* (adstour)<br>Used to analyze multiple yield analyses and adjust the nominal values to maximize the yield estimate of the statistical design variables. | Yield Specification Component (required) A Yield Specification component is used in conjunction to specify the acceptable performance. |
| *Design of Experiments Controller* (adstour)<br>Used to sequentially and iteratively improve the statistical performance of a design by identifying variables that contribute significantly to performance variation and honing in on the target statistical response. | DOE Goal Component (required) A DOE Goal component is used in conjunction to specify the desired goals. |

# Analyzing Results

Advanced Design System uses the datasets to store the simulation information you generate when analyzing designs. You can display this information for analysis using the Data Display window. A Data Display window can also be used to display data imported from other sources.

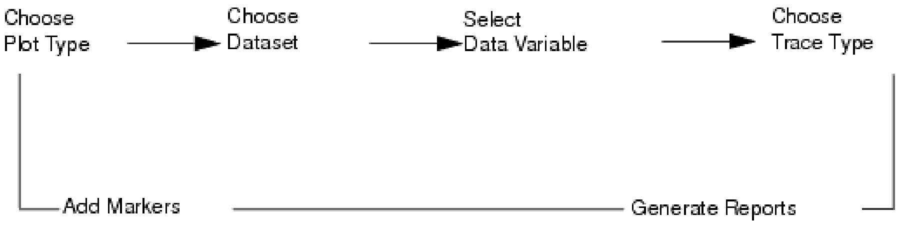
In a Data Display window you can:

- Display data in a variety of plots and formats
- Use markers to read specific data points on traces
- Use equations to perform operations on data
- Annotate results using text and illustrations
  Once simulation is complete, the data is displayed automatically if you did one of the following (a blank Data Display window is opened if you did none of them):
    - Specified a dataset and display before simulation
    - Use a schematic template for an Analog/RF simulation
    - Specified Rectangular in the Plot parameter in a sink for a Signal Processing simulation

## Creating Data Displays

The basic process of creating a data display is as illustrated:

1. Choose a plot type for the display
2. Choose the dataset that contains the data you want to display
3. Select the data variable to be displayed
4. Choose a trace type for the display



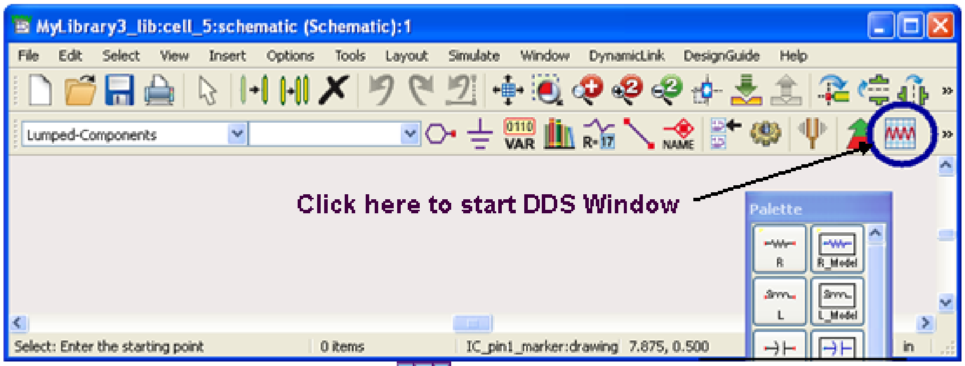
To enhance the display you can also add:

- Markers to identify specific data points
- Annotations using text and illustrations
- Legends to help identify specific traces

If you used a template to create the design you have simulated, the initial setup and configuration to create displays for data analysis is done for you automatically.

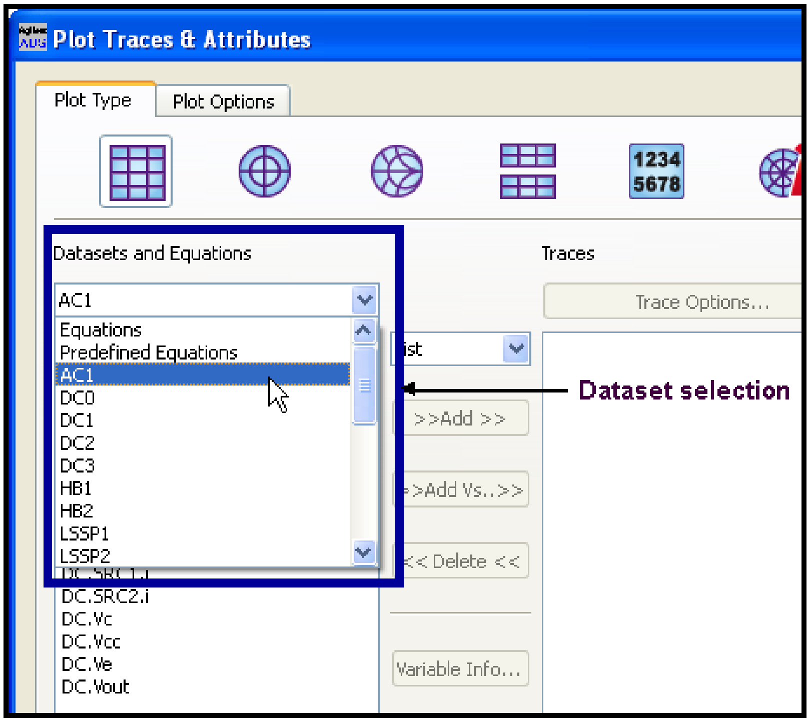
Follow the steps below to create a data display:

1. Click the New Data Display Window icon () button to open the Data Display window.

1. 
MyLibrary3_lib:cell_5:schematic (Schematic):1
File Edit Select View Insert Options Tools Layout Simulate Window DynamicLink DesignGuide Help
Lumped-Components
VAR
NAME
Palette
R
R_Model
L
L_Model
Select: Enter the starting point
0 items
IC_pin1_marker:drawing
7.875, 0.500
in

Click here to start DDS Window

2. Click Rectangular Plot icon (▦) from the Palette to open the Plot Traces & Attributes dialog box.
3. Select the dataset from **Datasets and Equations** drop-down list.



Dataset selection

4. After selecting the dataset, select plot and trace options as per your requirements. For more details, see *Plots and Lists* (data).
5. Click **OK** to return to DDS window.

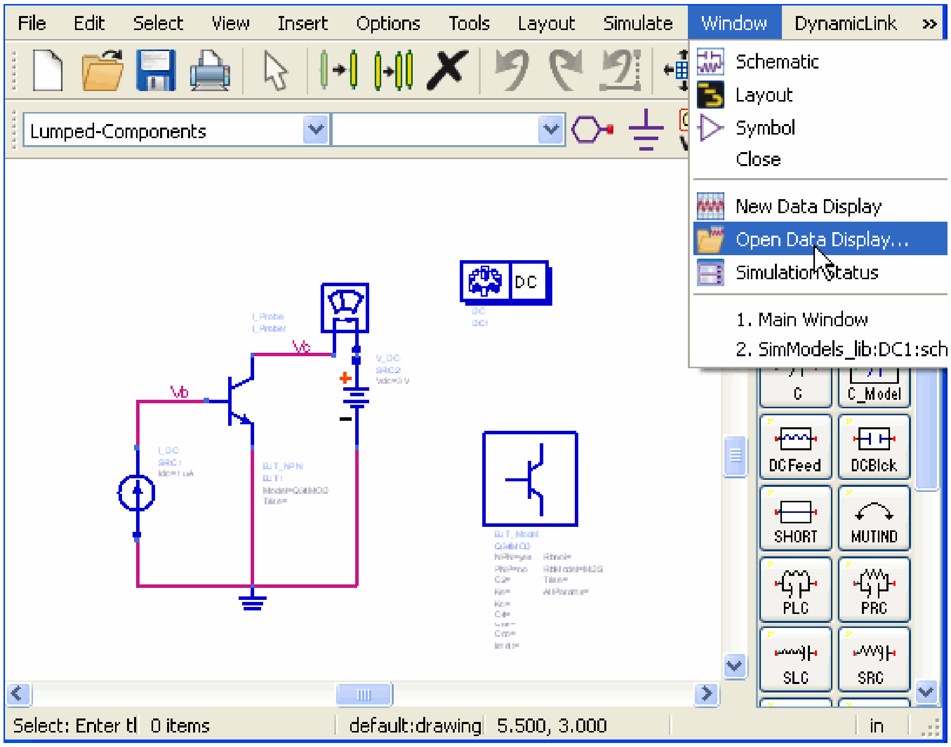
## Viewing Results

To view simulation results from the Main, Schematic, or Layout window choose **Window > Open Data Display** and use the dialog box to locate and open the results.

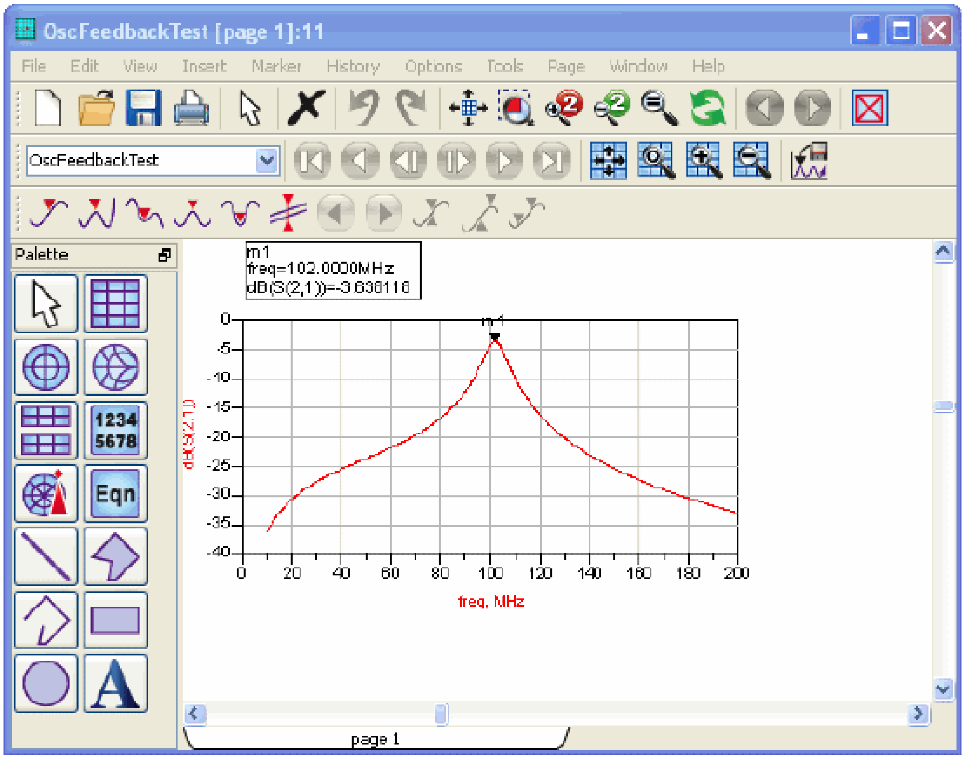
> **Note**
> To display a list of data display files in the **File Browser** page of the ADS Main window, select **View** > **Show All Files** from the ADS Main window.

To display the simulation results:

1. From any of the design window, choose **Window** > **Open Data Display**.

1. 

2. Select the Data Display File (`.dds` extension) from the Open Data Display dialog to display the simulation results (see below).



## Display Options

The following plot, trace, and data options can be used to display data for analysis:

| ***Plot Type*** | | | | ***Trace*** | ***Type*** | | | |
|---|---|---|---|---|---|---|---|---|
| | **Bus** | **Linear** | **Scatter** | **Spectral** | **Histogram** | **Digital** | **Sampled** | **Density** |
| **Rectangular** | **X** | X | X | X | X | X | X | X |
| **Stacked** | X | X | X | X | X | X | X | X |
| **Polar** | | X | X | | | | | |
| **Smith** | | X | X | | | | | |
| **Antenna** | | X | X | | | | | |

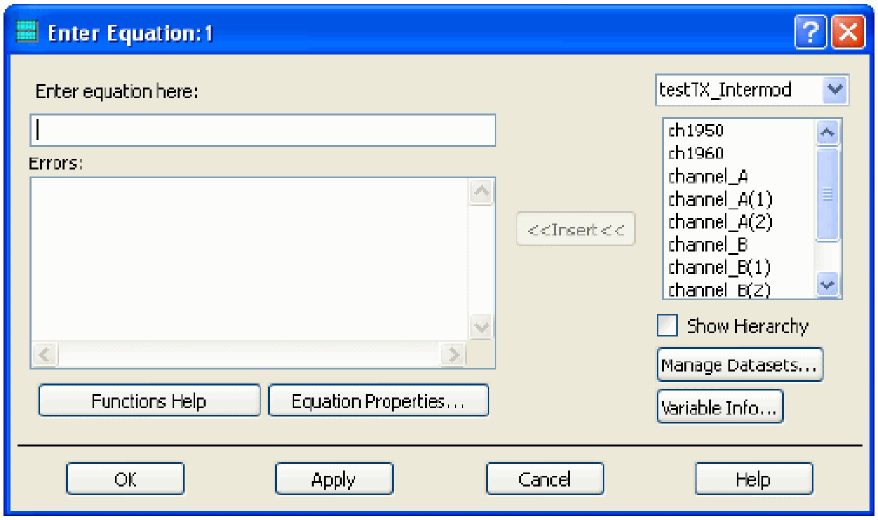
## Using Functions

You can use Measurement Equations to perform operations on data generated during a simulation. These equations are created using functions that are based on AEL, the Application Extension Language.

> **Note** Data from a marker can also be used as part of an equation. To insert a marker, choose *Marker* > *New* and click the trace where you want to insert it.

To create and insert a function...

1. Click the *Equation* icon (Eqn), and select a spot on the Display window to open the Enter Equation dialog box.



2. Enter the equation name and add the equal to (=) symbol to the equation.
3. Add the data that you want to assign to the equation name.
4. Click **Ok**.

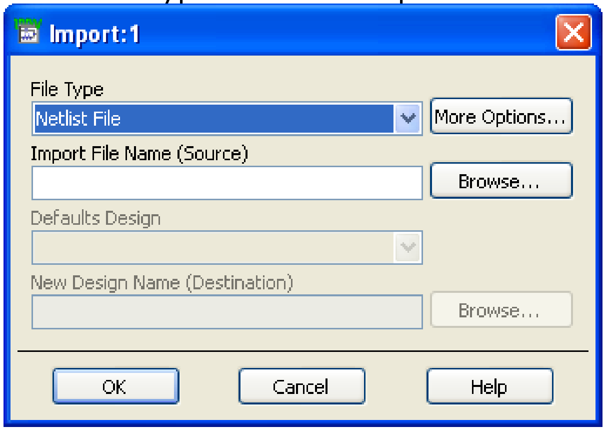
> **Note**
> For more details on inserting a new equation, see *Inserting Equations* (data).

# Inputs and Outputs

## Importing and Exporting Design

ADS allows you to import and export designs as well data. To import or export a design (schematic or layout):

- From Schematic window, choose **File** > **Import** (or **Export** ).
- Select file type from the drop down list.



File Type

Netlist File

Import File Name (Source)

Defaults Design

New Design Name (Destination)

- Enter file name to be imported (along with path). You can click **Preview** button to preview the file, before importing.
- Click **OK** to import (or export) the file.

To export ADS Ptolemy designs:

- From Schematic window, choose **Tools** > **Export ADS Ptolemy Design** > **As GoldenGate VTB** (or **As GoldenGate Model**)
- Enter the output location and click **OK**.

For more details, see *ADS Ptolemy Simulation* (ptolemy).

To import or export data:

- Choose **Tools** > **Data File Tool** from a Schematic window (for Touchstone, MDIF, CITI, and IC-CAP files)
- Choose **Tools** > **Connection Manager Client** from a Schematic window (for data from connected instruments)
- Choose **Tools** > **Instrument Server** (Windows only) (read and write data from various legacy instrument sources in a variety of file formats)

## Formats for Design Exchange

| Format | Import | Export |
|---|---|---|
| DXF (.dxf, .dwg) | Layout | Layout |
| EGS Archive Format (_a) | Layout | Layout |
| EGS Generate Format (_g) | Layout<br>Schematic | Layout |
| GDSII Stream Format (.gds) | Layout | Layout |
| Gerber (.gbr) | Layout | Layout |
| HPGL/2 (.hpg) | Layout<br>Schematic | Layout |
| HP IFF (.iff) | Layout<br>Schematic | Layout<br>Schematic |
| IGES (.igs) | Layout | Layout |
| Mask File (.msk) | Layout<br>Schematic | Layout |
| MGC/PCB (.iff) | | Layout |
| Spice (.cir, .cki, .iff, .net) | Schematic | |
| ODB++ | Layout (dir and file format) | Layout (file format) |

## Drawing Exchange Format (DXF)

This format was developed by Autodesk for its AutoCAD product to transfer geometric data between systems. Like the mask file format, it provides a simple geometric representation of data. DXF files can be transferred between PC-based or UNIX-based systems.

## Engineering Graphics System (EGS)

This format is a general graphics format used for capturing manually entered designs. EGS has been applied to ICs, Micro-circuits, Hybrids, and PC Board design applications. Using this format, you can easily exchange data with other programs using EGS formats. In addition, EGS facilitates better artwork translation with Advanced Design System.

- The Generate format is a flattened list of EGS primitives specified in the user-defined unit space.
- The Archive Format is a hierarchically organized list of EGS primitives specified in the user-defined unit space. Information such as drawing shapes, layout units, database precision, and grid spacing is included.

## GDSII Stream Format

This format is an industry standard for translating final mask data to foundries. Unlike other data formats, GDSII stream format is binary. You cannot easily view or edit a stream format file using a text editor. This format is easily translated between different CAD systems because it represents a highly restrictive data type.

## Gerber

This format refers to various data input formats that Gerber Scientific uses to drive its photoplotters. The Gerber format is used by photoplotters produced by other manufacturers also. RS274X and MDA formats are supported.

### HPGL/2

This format is a subset of the HPGL/2 printer/plotter language. When creating a graph or chart in another tool, you can write the graphics data to an HPGL/2 output file, then import the file into Advanced Design System. In Advanced Design System, the HPGL data is transformed into forms and shapes that can be edited and manipulated like any other drawing. Additional text, annotation, scaling or editing may be added.

### Intermediate File Format (IFF)

This format is an ASCII file with a simple, line-oriented command structure and a fairly rich set of constructs. This format is machine- and application-independent, thus simplifying design data transfer. IFF files are used as the exchange mechanism when transferring designs between Advanced Design System and third-party EDA tools such as Mentor Graphics Design Architect and Cadence Analog Artist.

### Initial Graphics Exchange Specification (IGES)

This format is an approved ANSI standard that is used extensively throughout the computer-aided design and manufacturing world. It can represent both mechanical and electrical design data in two and three dimensions. The IGES standard for the transfer of electrical design data is known as CALS specification. Advanced Design System supports version 4.0 and 5.0 IGES formats. It reads and writes IGES CALS Level 1 (technical illustration) and Level 3 (electrical/electronic applications) files.

### Mask

This format is a simple flat (non-hierarchical) geometric description. The format facilitates the transfer of simple geometric data for final mask processing. Only geometric forms are described in a mask file; simulation data, element parameters, substrate definitions, and hierarchy are not included.

### MGC/PCB

These files are IFF files that are used exclusively for Mentor Graphics design transfers. MGC/PCB files write to a specific location each and every time. When you select this format, the filename and location of the IFF transport is determined automatically.

### Spice

Simulation Program with Integrated Circuit Emphasis (Spice) has become a simulation tool used by engineers throughout the world for simulating circuits of all types. After its development at the University of California Berkeley, Spice has been commercialized and modified by a large number of vendors and also adopted and modified by electronics companies for their own in-house use.

## ODB++

ODB++ is widely accepted as a practical de-facto standard within the electronics industry as an efficient way to move printed circuit bareboard, assembly and test data on the manufacturing-engineering level within design/manufacturing supply chains.

## Supported Data Transfer Format

The following data transfer formats are supported:

| **Touchstone (SnP) Format** | **Description** | **Usage** |
|---|---|---|
| SnP | Small signal S, H, Y, Z, or G-parameters. May also include optional noise data (2 port data only). Where n is the number of ports from 1 to 99. | n-Port S-parameter file (SnP) components in the Data Items Library. |

When writing data from a dataset to a file, the variable names are limited to S,H,Y,Z or G, for example, S[1,1], S[1,2], G[1,1], G[1,2]. The variable name is used to determine the type of data.
The first set of data in the dataset that matches the data type (name) will be output. It is not possible to arbitrarily select which data will be output.

| **CITIfile Format** | **Description** | **Usage** |
|---|---|---|
| CITI | A general data format supported by network analyzers. Capable of storing multiple packages of multi-dimensional data. | S#P #-Port S-parameter file components in the Data Items Library. |

There are some specific problems with the current version in writing and/or reading this data format. Refer to the release notes or on the Agilent EEsof support Web site for more information and workarounds.

| **Agilent IC-CAP Formats** | **Description** | **Usage** |
|---|---|---|
| DUT, MDL, SET | Device under test (DUT), model (MDL), and setup (SET) files from the Agilent IC-CAP software. These files can contain Measured, Simulated, and/or Transformed data. | Once the data is read into a dataset, it can be used with any component (for example, a VtDataset source) that can read data from a dataset. |

You can read in IC-CAP data only.
Only simple, scaled expressions with numbers or variables and one operator (either +, -, \<B>, or /) are supported for start, stop, step, and number of points parameters, for example, start= 1 GHZ or stop=icmax/10.

| **MDIF Formats** | **Description** | **Usage** |
|---|---|---|
| DSCR | Discrete (indexed) tabular and possibly statistical density data. | DAC |
| GCOMP | Gain compression data | Amplifier and Mixer items in the System - Amps & Mixers library. |
| GEN_MDIF | Generalized multi-dimensional tables unifying other MDIF formats. | DAC |
| IMT | Intermodulation product table of mixer intermodulation products between the LO and signal that relates the mixer IM output level to signal input level. | MixerIMT in the System - Amps & Mixers library. |
| MODEL_MDIF | Nonlinear model parameters | EEFET1, BJTAP, etc. |
| P2D | Large-signal, power-dependent, 2-port S, H, Y, Z, or G - parameters. | AmplifierP2D item in the System - Amps & Mixers library. |
| PDF | User defined, piece-wise linear probability density function data. | With expressions in the Statistics tab. |

The PDF format is not yet fully supported.

| | | |
|---|---|---|
| S2D | 2-port S, H, Y, Z, or G-parameters with gain compression and optional noise and intermodulation data. | Amplifier S2D, Amplifier, and Mixer items in the System - Amps & Mixers library. |
| S2PMDIF | Multi-dimensional 2-port, S, Y, Z, H, G signal and optional 2-port noise parameter (Fmin, Gopt, Rn) data. | With S2PMDIF and DAC |
| SDF | Type #1:Newer/More Used by customers: 89600 series VSA's encrypted sdf data file | Use the Ptolemy "Instruments" library SDFWrite and SDFRead files to work with this type of data. The resulting files can be played back in Agilent 89601A VSA software as well. For more details, refer to the software documentation of the Agilent 89600. |
| | Type #2:Older/legacy/obsoleted 89440 VSA's and encrypted sdf data file. | You can use ADS design's /Tools/Data File Tool (write/read) with 'File format…= MDIF and 'mdif subtype' = SDF to write this type of file. For more details see, http://www.home.agilent.com/agilent/editorial.jspx?cc=US&lc=eng&ckey=456587&nid=-11143.0.00&id=456587&pselect=SR.GENERAL |
| SPW | Time-domain voltage data file in Cadence Alta Group SPW format | TimeFile item in Timed Sources and OutFile item in Sinks library. |
| TIM | Time-domain data | TimeFile item in Timed Sources and OutFile item in Sinks library. |

When writing data from a dataset to a file, the variable names are limited to S,H,Y,Z or G, for example, S[1,1], S[1,2], G[1,1], G[1,2]. The variable name is used to determine the type of data.

The first set of data in the dataset that matches the data type (name) will be output. It is not possible to arbitrarily select which data will be output.

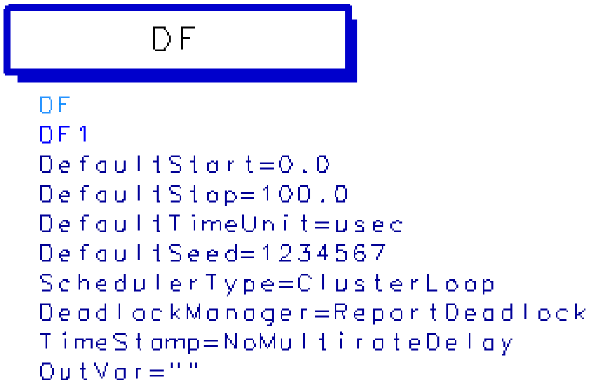
There are some specific problems with the current version in writing and/or reading this data format. For more information and workarounds, refer to the release notes at the Agilent EEsof support Web site.

**Obsolete Formats:** COD, FIR, LAS, SPE, LIST2, and T2D.

# Simulation and Optimization Controllers

## Data Flow Simulation Controller

The Data Flow controller is used to control the flow of mixed numeric and timed signals for all digital signal processing simulations within ADS. This controller works with the sink components to provide you flexibility to control the duration of the simulation globally or locally.

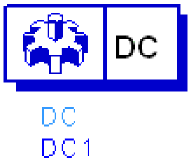


```
DF
DF1
DefaultStart=0.0
DefaultStop=100.0
DefaultTimeUnit=usec
DefaultSeed=1234567
SchedulerType=ClusterLoop
DeadlockManager=ReportDeadlock
TimeStamp=NoMultirateDelay
OutVar=""
```

> **Note**
> You cannot place the multiple controllers on the schematic to simulate the same design with different controller parameters. To achieve the same functionality by using the single-point sweeps on the parameter you are interested in varying.

## DC Simulation Controller

The DC controller provides for both single-point and swept simulations. Swept variables can be related to voltage or current source values, or to other component parameter values. By performing a DC swept bias or a swept variable simulation, you can check the operating point of the circuit against a swept parameter such as temperature or bias supply voltage.



```
DC
DC1
```

Use the DC controller to:

- Verify the proper DC operating characteristics of the design under test.
- Determine the power consumption of your circuit.
- Verify model parameters by comparing the DC transfer characteristics (I-V curves) of the model with actual measurements.
- Display voltages and currents after a simulation.

A DC simulation is the first analysis for most other analyses. It uses a system of nonlinear ordinary differential equations (ODEs) to solve for an equilibrium point in the linear/nonlinear algebraic equations that describe a circuit once:

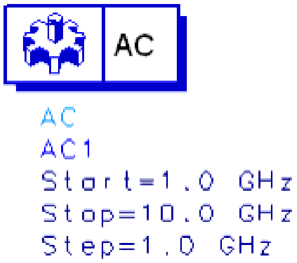
- Independent sources are constant valued

- Capacitors and similar items are replaced with open circuits
- Inductors and similar items are replaced with short circuits
- Time-derivatives are constant (zero)

Linear elements are replaced by their conductance at zero frequency.

## AC Simulation Controller

A linear AC analysis is a small-signal analysis. For this analysis the DC operating point is found first and then the nonlinear devices are linearized around that operating point. Small-signal AC simulation is also performed before a harmonic-balance (spectral) simulation to generate an initial guess at the final solution.



AC
AC1
Start=1.0 GHz
Stop=10.0 GHz
Step=1.0 GHz

Use the AC controller to:

- Perform a swept-frequency or swept-variable small-signal linear A simulation.
- Obtain small-signal transfer parameters, such as voltage gain, current gain, transimpedance, transadmittance, and linear noise.

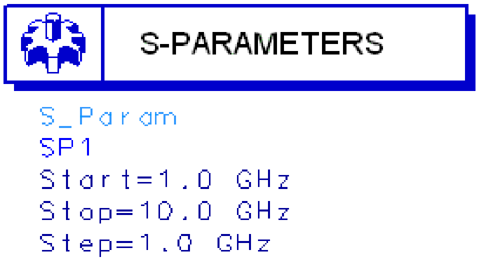
An AC simulation also offers a linear noise simulation option that can include the following noise contributions in its simulation:

- Temperature-dependent thermal noise from lossy passive elements, including those specified by data files.
- Temperature and bias-dependent noise from nonlinear devices.
- Noise from linear active devices specified by two-port data files that include noise parameters.
- Noise from noise source elements.

The noise simulation computes the noise generated by each element, and then determines how that noise affects the noise properties of the network.

## S-Parameter Simulation Controller

The S-Parameter controller is used to define the signal-wave response of an n-port electrical element at a given frequency. It is a type of small-signal AC simulation that is most commonly used to characterize a passive RF component and establish the small-signal characteristics of a device at a specific bias and temperature.



```
S_Param
SP1
Start=1.0 GHz
Stop=10.0 GHz
Step=1.0 GHz
```

Use the S-Parameter controller to:

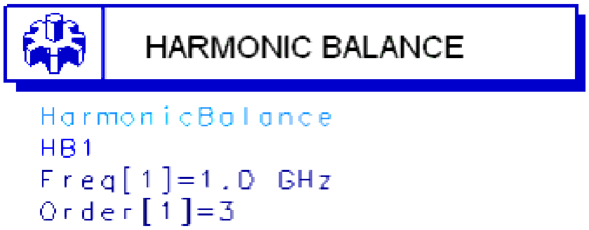
- Obtain the scattering parameters (S-parameters) of a component or circuit, and convert the parameters to Y- or Z-parameters.
- Plot, for example, the variations in swept-frequency S-parameters with respect to another changing variable.
- Simulate group delay.
- Simulate linear noise.
- Simulate the effects of frequency conversion on small-signal
- S-parameters in a circuit employing a mixer.

S-parameter simulation normally considers only the source frequency in a noise analysis. Use the Enable AC Frequency Conversion option if you also want to consider the frequency from a mixer's upper or lower sideband.

## Harmonic Balance Simulation Controller

The Harmonic Balance controller is best suited for simulating analog RF and microwave circuits. It is a frequency-domain analysis technique for simulating distortion in nonlinear circuits and systems. Within the context of high-frequency circuit and system simulation, harmonic balance offers the following benefits over conventional time-domain transient analysis:

- It captures the steady-state spectral response directly.
- Many linear models are best represented in the frequency domain at high frequencies.
- The frequency integration required for transient analysis is prohibitive in many practical cases.



```
HarmonicBalance
HB1
Freq[1]=1.0 GHz
Order[1]=3
```

Use the Harmonic Balance controller to:

- Determine the spectral content of voltages or currents.
- Compute quantities such as third-order intercept points, total harmonic distortion, and intermodulation distortion components.
- Perform power amplifier load-pull contour analyses.
- Perform nonlinear noise analysis.

Harmonic Balance enables the multitone simulation of circuits that exhibit intermodulation frequency conversion, including frequency conversion between harmonics. It is an iterative method that assumes that for a given sinusoidal excitation there exists a steady-state solution that can be approximated to a satisfactory accuracy.

## Simulation Overview

Harmonic balance is a frequency-domain analysis technique for simulating distortion in nonlinear circuits and systems. It obtains the frequency-domain voltages and currents to calculate the spectral content of voltages or currents in the circuit. The harmonic balance method is iterative. It is based on the assumption that for a given sinusoidal excitation there exists a steady-state solution that can be approximated to satisfactory accuracy by means of a finite Fourier series.

The Harmonic Balance solution is approximated by truncated Fourier series and this method is inherently incapable of representing transient behavior. The time-derivative can be computed exactly with boundary conditions, v(0)=v(t), automatically satisfied for all iterates.

The truncated Fourier approximation + N circuit equations results in a residual function that is minimized.
N x M nonlinear algebraic equations are solved for the Fourier coefficients using Newton's method and the inner linear problem is solved by:

- Direct method (Gaussian elimination) for small problems
- Krylov-subspace method (e.g. GMRES) for larger problems

Nonlinear devices (transistors, diodes, etc.) in Harmonic Balance are evaluated (sampled) in the time-domain and converted to frequency-domain via the FFT.

## Advantages

- Harmonic balance captures the steady-state spectral response directly while conventional transient methods need to integrate over many periods of the lowest-frequency sinusoid to reach steady state.
- Harmonic balance is faster at solving typical high-frequency problems that transient analysis can't solve accurately or can only do so at prohibitive costs.
- Harmonic balance is more accurate at solving high frequencies where many linear models are best represented in the frequency domain.

## Convergence

Nonconvergence is a numerical problem encountered by the harmonic balance simulator when it cannot reach a solution, within a given tolerance, after a given number of numerical iterations. There is no one specific solution for solving convergence problems. However, consider the following guidelines:

- Increase the Order (or other harmonic controls); this is the most basic technique for solving convergence problems, if the time penalty for doing so is acceptable.
- Use the Status server window as the main tool in solving convergence problems (set StatusLevel=4). For each Newton iteration the L-1 norm of the residuals throughout the circuit is printed: a "*" indicates a full Newton step (vs. a Samanskii step).
- Convergence criteria are controlled by Voltage relative tolerance, and Current relative

tolerance (in the Options component, under the Convergence tab). In general, convergence speed is improved by increasing these values, but at the expense of accuracy. Similarly, the smaller these values are, the more accurate the results but the slower the convergence.

- Newton convergence issues with Krylov methods (because linear problem solutions can only approximate) can be improved by using better preconditioners.
- Set the Oversample parameter to a value greater than 1.0, such as 2.0 or 4.0. However, remember that although this can often solve convergence problems, it does so at the cost of computer memory and simulation time. For multiple-tone harmonic balance simulations, make sure that the largest signal in the circuit is assigned to Freq[1]. The simulator's FFT algorithm is set up so that aliasing errors are much less likely to affect Freq[1] than any other tone.
- When using a direct linear solver, the blocks of the Harmonic Balance Jacobian inherit the Jacobian matrix ordering from the DC solution process. This matrix ordering can greatly affect the efficiency of the Harmonic Balance Jacobian factorization, and in some circuits show noticeable simulation slowdown. To circumvent this issue, use a DC convergence mode that hasn't changed, e.g. DC_ConvMode=3.
- For non-convergence due to tight tolerances, monitor the residuals in the Status Server window.
  - Increase I_AbsTol if the circuit is converging to within a few pA but not quite to I_AbsTol=1pA
  - Increase I_RelTol if the problem is with nodes associated with large currents
  - Increase I_AbsTol if the small current nodes are the issue
  - Relax voltage tolerances for failure in the Newton update criterion
- The internal circuit simulator engine in ADS (Gemini) runs from a netlist. ADS writes a netlist file (netlist.log) before invoking Gemini. The order of the components and model definitions in the netlist determine the initial Jacobian matrix ordering. This matrix ordering can affect the efficiency of the Jacobian factorization and cause either a simulation slow down or non-convergence.
- For convergence problems due to errors in the component model equations (incorrect derivatives, etc.) make sure ancient Berkeley MOSFET Level 1, 2, 3 are not the culprit and that the latest model version is used (especially BSIM3 models). Model problems can cause the Newton residual to hit a threshold (greater than the convergence criteria tolerances) and stale the convergence process or even exhibit random jumps (sudden increase in value). Set the device's Xqc parameter to a nonzero value to allow the simulator to use a charge-based model for the gate capacitance. This often enables convergence, but at the cost of extracting an extra SPICE model parameter.

## Sweeps as Convergence Tools

Continuation methods provide a sequence of initial guesses that are sufficiently close to the solution to assure Newton's method convergence in Harmonic Balance. Sweeps can be used to formulate a specialized continuation method geared towards the particular circuit problem.

Sweep a circuit element that, when set to some different value, makes the circuit more linear. For instance, in an amplifier circuit there may be a resistor that can be used to lower the amplifier's gain. The simulator may be able to find a solution to the circuit under a low-gain condition. Then, if the component's value is swept toward the desired value, the simulator may be able to find a final solution. Start with a value that works, and stop with the desired value. Also, select Restart, under the Params tab. Usually, a better initial guess at each step helps the simulator to converge.

The two main ways to perform sweeps are:

- HB sweep within the HB controller. This is preferred for most sweeps, except frequency.
- Parameter sweep using a separate sweep controller.

## Convergence and Samanskii Steps

The Samanskii steps can significantly speed up the solution process. However, using an approximate Jacobian, particularly for a larger number of iterations, may result in poor or even no convergence. The constant is used in two ways. First, it becomes a more absolute measure when it is smaller. It then approaches the requirement that each iteration reduces the relevant norm by one-third.

Decreasing the Samanskii constant beyond a certain point (which in turn depends on the quality of the most recent Newton step) will make no difference. However, setting the Samanskii constant to zero will effectively disable any Samanskii steps altogether.

Increasing the Samanskii constant relaxes this requirements in general, but the condition becomes more dependent on the quality of the standard most recent Newton iteration. In other words, a more rapid convergence of the Newton step would also require better convergence of the Samanskii steps.

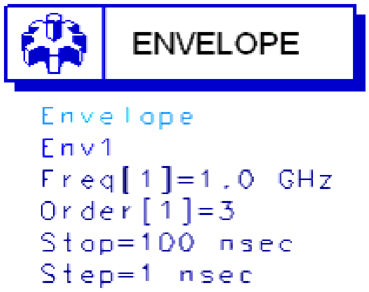
## Convergence and Arc-Length Continuation

Arc-length continuation is an extremely robust algorithm. If it fails, try all other convergence remedies first before adjusting arc-length parameters

- MaxStepRatio controls the maximum number of continuation steps (default 100)
- MaxShrinkage controls the minimum size of the arc-length step (default 1e-5)
- ArcMaxStep limits the maximum size of the arc-length step (default is 0, i.e. no limiting)
- ArcMinValue & ArcMaxValue define the allowed range for the variation of the continuation parameter

## Circuit Envelope Simulation Controller

The Circuit Envelope controller is best suited for a fast and complete analysis of complex signals such as digitally modulated RF signals. It combines features of time and frequency-domain representation by permitting input waveforms to be represented in the frequency domain as RF carriers, with modulation "envelopes" that are represented in the time domain.



Envelope
Env1
Freq[1]=1.0 GHz
Order[1]=3
Stop=100 nsec
Step=1 nsec

Circuit Envelope is highly efficient in analyzing circuits with digitally modulated signals, because the transient simulation takes place only around the carrier and its harmonics. In addition, its calculations are not made where the spectrum is empty.

- It is faster than Harmonic Balance, for a given complex signal Spice, assuming most of the frequency spectrum is empty
- It does not compromise in Signal complexity, unlike time-varying HB or Shooting Method Component accuracy, unlike Spice, Shooting Method, or DSP
- It adds physical analog/RF performance to DSP/system simulation with real-time co-simulation with ADS Ptolemy
- It is integrated in same design environment as RF, Spice, DSP, electromagnetic, instrument links, and physical design tools

## Advantages over Harmonic Balance

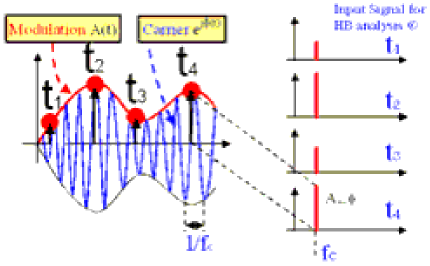
- In Harmonic Balance, if you add nodes or more spectral frequencies, the RAM and CPU requirements increase geometrically. Krylov improved this, but it's still a limitation of Harmonic Balance because the signals are inherently periodic.
- Conversely the penalty for more spectral density in Circuit Envelope is linear: just add more time points by increasing TSTOP. The longer you simulate, the finer your resolution bandwidth.
- Doing a large number of simple 1-tone HB simulations is effectively faster and less RAM intensive than one huge HB simulation.
- With a circuit envelope simulation the amplitude and phase at each spectral frequency can vary with time, so the signal representing the harmonic is no longer limited to a constant, as it is with harmonic balance.

## Limitations

1. More occupied spectrum than unoccupied spectrum.
   You're carrying more overhead with frequency-domain assumptions and harmonics than necessary. Use SPICE.
2. Everything baseband. Depends.
   - If everything linear, use AC/S-parameter (for noise or budget)
   - If everything nonlinear or digital, use SPICE.
   - If everything logic/behavioral, use PTOLEMY.
3. Occupied spectrum is relatively sparse.
   If you can do what you want using Harmonic Balance, you should. Post-processing, optimization, and yield are simpler and faster.

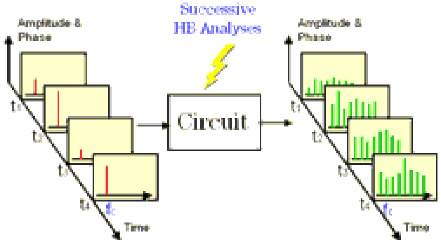
## Simulation Process

1. Transform input signal



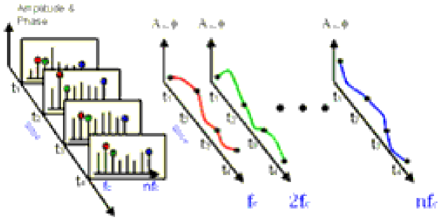
Each modulated signal can be represented as a carrier modulated by an envelope - A(t)*ejf(t). The values of amplitude and phase of the sampled envelope are used as input signals for Harmonic Balance analyses.

2. Frequency Domain Analysis



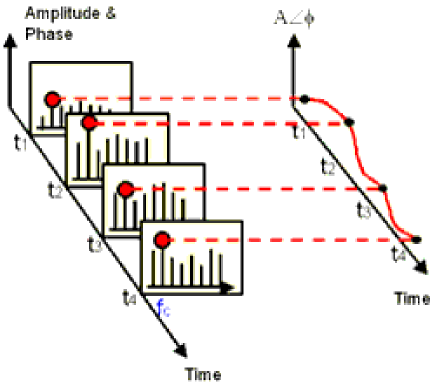
Harmonic Balance analysis is performed at each time step. This process creates a succession of spectra that characterize the response of the circuit at the different time steps.

3. Time Domain Analysis



Circuit Envelope provides a complete non steady-state solution of the circuit through a Fourier series with time-varying coefficients.

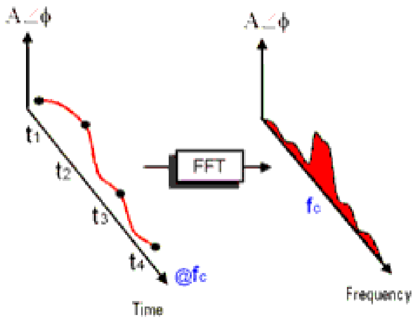
4. Extract Data from Time Domain



Selecting the desired harmonic spectral line (fc in this case), it is possible to analyze:

- Amplitude vs. Time (Oscillator start up, Pulsed RF response, AGC transients)
- Phase (f) vs. Time (t) (VCO instantaneous frequency (df/dt), PLL lock time)
- Amplitude & Phase vs. Time (Constellation plots, EVM, BER)

5. Extract Data from Frequency Domain



By applying FFT to the selected time-varying spectral line it is possible to analyze:

- Adjacent Channel Power Ratio (ACPR)
- Noise Power Ratio (NPR)
- Power Added Efficiency
- Reference frequency feedthrough in PLL
- Higher order intermods (3rd, 5th, 7th, 9th)

## Simulation Steps

| | | |
|---|---|---|
| 1. Define baseband signal modulation<br>• Predefined sources<br>• Equations<br>• I & Q data vs. time data from DSP simulation<br>2. Define RF carrier frequencies, time step and duration of the simulation<br>3. Compute time-varying Fourier coefficients<br>4. Post-process and display results | OR | 1. Define input signal(s) with modulation - amplitude, phase, frequency, I/Q, etc.<br>2. Define the time step<br>3. Simulator computes Fourier coefficients versus time:<br>4. Fourier transforms are computed to display frequency spectrum around any tone (if necessary) |

## Typical Analyses

- Intermodulation distortion.
- Amplifier spectral regrowth and adjacent channel power leakage.
- Oscillator turn-on transients and frequency output versus time in response to a transient control voltage.
- PLL transient responses.
- AGC and ALC transient responses.
- Circuit effects on signals having transient amplitude, phase, or frequency modulation.
- Amplifier harmonics in the time domain.
- Subsystems using modulation signals such as multilevel FSK, CDMA, or TDMA.
- Third-order-intercept and higher-order intercept analyses of amplifiers and mixers.
- Time-domain optimization of transient responses.

## Typical Applications

## Time Domain Data Extraction

Selecting the desired harmonic spectral line it is possible to analyze:

- Amplitude vs. Time
  Oscillator start up

Pulsed RF response
AGC transients

- Phase vs. Time
  VCO instantaneous frequency, PLL lock time
- Amplitude & phase vs. time
  Constellation plots
  EVM, BER

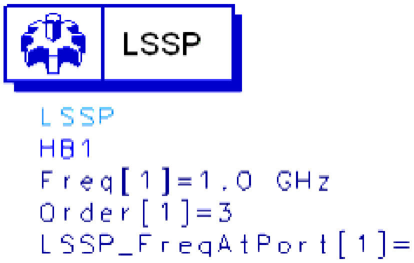
## Frequency Domain Data Extraction

By applying FFT to the selected time-varying spectral line it is possible to analyze:

- Adjacent Channel Power Ratio (ACPR)
- Noise Power Ratio (NPR)
- Power added efficiency
- Reference frequency feedthrough in PLL
- Higher order intermods (3rd, 5th, 7th, 9th)

## LSSP Simulation Controller

The large-signal S-parameter simulation controller facilitates the computation of large-signal S-parameters in nonlinear circuits.

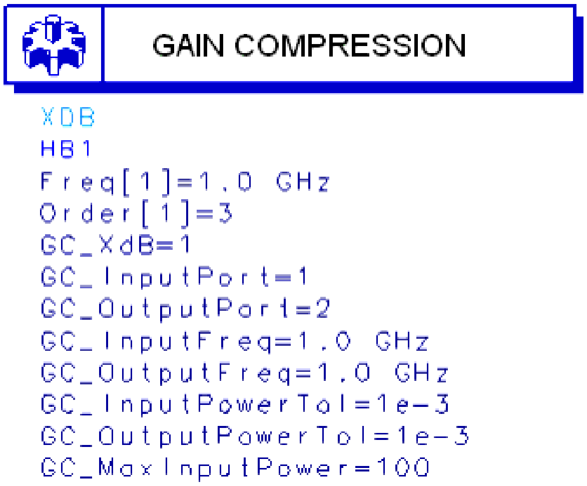


LSSP
HB1
Freq[1]=1.0 GHz
Order[1]=3
LSSP_FreqAtPort[1]=

Large-signal S-parameters are based on a harmonic balance simulation of the full nonlinear circuit. Unlike S-parameters, large signal S-parameters can change as power levels are varied because the harmonic balance simulation includes nonlinear effects such as compression.

## XDB Simulation Controller

The XDB simulation controller computes the gain compression point of an amplifier or mixer. It sweeps the input power upward from a small value, stopping when the required amount of gain compression is seen at the output.

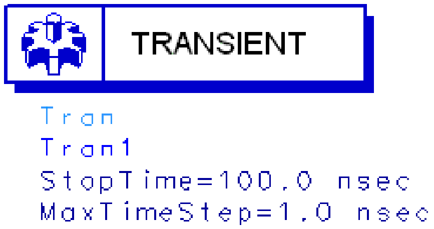


GAIN COMPRESSION

```
XDB
HB1
Freq[1]=1.0 GHz
Order[1]=3
GC_XdB=1
GC_InputPort=1
GC_OutputPort=2
GC_InputFreq=1.0 GHz
GC_OutputFreq=1.0 GHz
GC_InputPowerTol=1e-3
GC_OutputPowerTol=1e-3
GC_MaxInputPower=100
```

## Transient/Conv. Simulation Controller

The transient and convolution simulation controllers solve a set of integro-differential equations that express the time dependence of the currents and voltages of the circuit. The result of such an analysis is nonlinear with respect to time and, possibly, a swept variable.



TRANSIENT

```
Tran
Tran1
StopTime=100.0 nsec
MaxTimeStep=1.0 nsec
```

Use the Transient/Convolution controller to perform:

- SPICE-type transient time-domain analysis.
- Nonlinear transient analysis on circuits that include the frequency-dependent loss and dispersion effects of linear models, or Convolution analysis.

A transient analysis is performed entirely in the time-domain. It does not account for the frequency-dependent behavior of distributed elements.

A convolution analysis represents distributed elements in the frequency domain to account for their frequency-dependent behavior.

## Transient Simulation and Convergence

In Transient analysis a numerical integration algorithm is employed at each time point to approximate the differential equations into algebraic equations. Integration methods are used to replace the time derivative with a discrete-time approximation

## Time Step Control Characteristics

Local Truncation Error

- Estimates the LTE made on every capacitor and inductor
- Determines the time step size to ensure the largest LTE remains within the accepted tolerance
- The estimated LTE is inversely proportional to TruncTol
- The accepted tolerance depends upon the relative and truncation tolerances set for the current and voltage. It is proportional to I_RelTol x TruncTol and V_RelTol x TruncTol

Iteration-Count

- Determines the time step size based on the number of Newton iterations required for previous time point
- No direct relationship between iterations and LTE
- Effectively controlled by Max time step (for linear circuits)

Fixed

- The time step is fixed and equal to Max time step

Break Points

- Generated by built-in independent sources whenever an abrupt change in slop occurs
- Ensure that corners in waveforms are not missed
- ADS always places time points on a break point (except fixed time step)
- Backward Euler is used on time points that are the first time step after break points
- The step size is reduced when time point is close to a break point

## Transient Convergence Tips

1. For initial Transient analysis, try to use I_RelTol = V_RelTol = 1e-3, and tighten these values only when higher accuracy is needed. Simulation will run much faster with these setting compared to 1e-6.
2. Transient analysis convergence problems are often caused by jumps in the solution. This most often occurs in circuits with overly simplified models that exhibit positive feedback, or when the circuit contains nodes that do not have a capacitive path to ground. Add a small capacitor from the troublesome node to ground and give a complete capacitance model when specifying the nonlinear device model parameters.
3. Generally analog circuits are sensitive to truncation error due to their relative long time constants. Use LTE time step control to ensure the accuracy of the results.
4. Backward Euler (Gear1 or Mu=0 in Trapezoidal) and Gear2 are stable for all stable and some unstable differential equations. However, trapezoidal rule are stable only on stable differential equations. Switch to Gear1 or Gear2 when trapezoidal rule fails on unstable differential equations.

## Typical Convergence Problems

## Capacitor model problems

- Use simplified device models that do not include capacitance model or incomplete capacitance model give a complete capacitance model when specifying nonlinear device model parameters, in junction capacitance, include both depletion (at least) and diffusion capacitances
- Discontinuous jumps in waveforms when circuit contains nodes have no capacitive path to ground add small capacitor to ground or specify Cmin
- Capacitance model does not conserve charge GaAsFET Statz's, MOSFET Meyer's capacitance models switch to charge based model
- Large floating capacitors that are similar to the small-floating resistor problem in DC (finite precision problem) check capacitance unit, use smaller capacitance
- Discontinuous capacitance models in user defined model, SDD device fix the model

### Slow Transient analysis

- Make sure I_RelTol and V_RelTol are set to 1e-3 or not set at all
- Decrease these values when higher accuracy is needed

### Oscillator circuit does not oscillate

- Apply a short pulse at the beginning of the simulation
- Avoid using Gear2 or backward Euler

### Circuit exhibits ringing or divergence

- Reduce Mu value from 0.5 toward 0 if trapezoidal rule is used
- Use Gear1 or Gear2

### Circuit does not converge at first time point

- Reduce Min time step

### Convergence Hints

### Add break points

Use piecewise linear source to add break points to the region where the waveform changes abruptly

### Reduce max time step

Ensure enough time points for sharp edges

### Increase Max iterations per time step

Increase to 50 or more to increase the possible number of Newton iterations on each time step

### Increase I_AbsTol

Try 1e-10 instead of the default 1e-12

### Relax TruncTol

Increase this value 10 times or more to relax LTE tolerance

### Relax I_Reltol and V_Reltol

Increase to 1e-3 to relax Newton convergence tolerance as well as LTE tolerance

### Try different integration methods

Switch from trapezoidal to Gear's method

### Using Convolution

- Don't set any convolution parameters (let the adaptive algorithm figure it out)
- Set ImpMaxFreq first (larger than signal bandwidth)
- Set convolution parameters on component, not controller, when possible
- Don't allowed measured data to be extrapolated (either set ImpMaxFreq or provide more data)

### Convolution Modeling for Time-Domain Simulation

- In time-domain simulation, simulate devices that can only be defined in the frequency domain
    - Transmission lines with dispersion
    - Devices with frequency-dependent loss
    - Measured frequency-domain data
- Convolution is the key
    - Inverse Fourier transform of frequency-domain data produces the impulse response h(t)
    - The impulse response is convolved with time-domain signal

## Time and Frequency Range

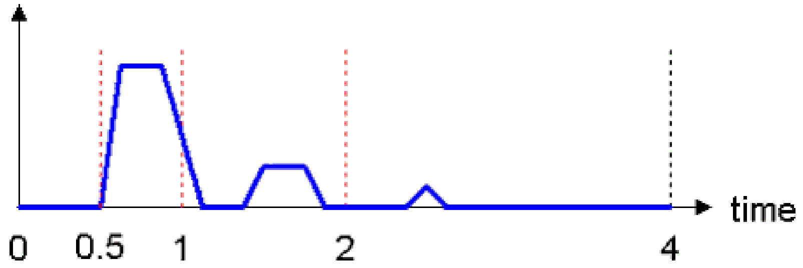
- Impulse response is computed from the inverse Fourier transform of frequency-domain response frequency is uniformly sampled from 0 to some upper value
- Upper frequency sets the time-domain spacing of the impulse response
- Frequency spacing sets the length of the impulse response

## Adaptive Impulse Response Calculation

- Estimate of system bandwidth is made from source frequencies and rise times - initial guess at fmax
- Build a trial impulse response with 32 timepoints
  very coarse frequency spacing
- Build a second impulse response with 64 timepoints
  less coarse frequency spacing
- Keep doubling the number of timepoints until a good impulse response is obtained
  increase fmax, decrease Df
- y11 and y12 may be sampled with different fmax and Df
- Adaptive calculation is only done if ImpDeltaFreq is not specified
  don't set ImpDeltaFreq if you don't have to

## Good Impulse Responses

- Compare impulse responses with N and 2N points. The second impulse response is twice as long in time domain and has half the frequency spacing.
- An impulse is considered "good" when no appreciable energy is present in the second half of the impulse response if energy is present in the second half, implies either that the impulse is not long enough or it is noncausal
- If not good, Controller keeps doubling the length
- Controller also tries doubling the maximum frequency, giving smaller impulse timesteps



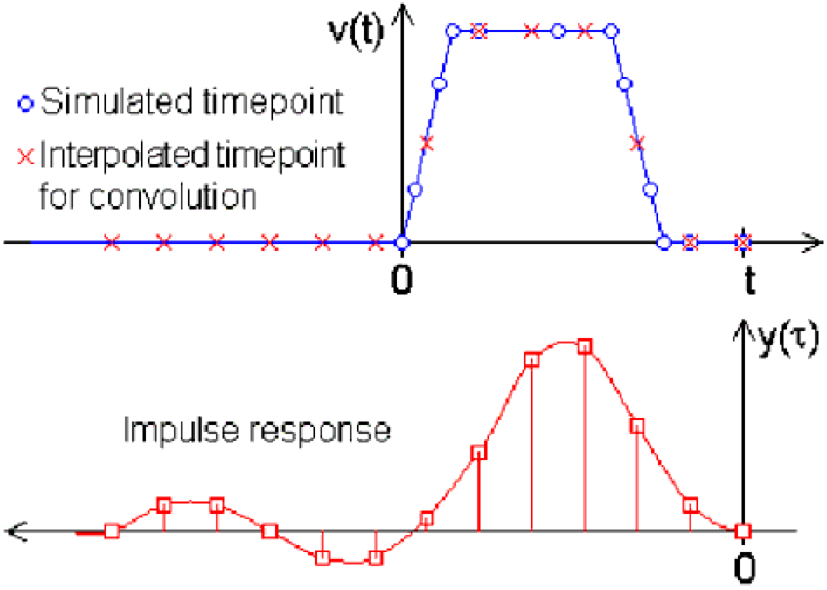
## Interpolation

- The impulse response is sampled with a uniform timestep, but is not guaranteed to match the simulation timestep. The simulation may even be using a variable timestep.
- Interpolate the signal v(t) to match the timepoints in the impulse response
- Don't interpolate the impulse response because the Fourier transform of the interpolated impulse response would no longer match the original frequency response

## Impulse Evaluation

- Signal response at time zero extends back to minus infinity
- Evaluate the integral as a sum

$$i(t) = \sum_{0}^{n} v(t - \tau_n)\, y(\tau_n)$$



## Solving an Invalid Impulse Response

This is the most commonly encountered problem during convolution. It does not necessarily imply noncausality but means that significant energy is present in the second half of the impulse response. In addition, simulation results may or may not be valid.

- Set ImpMaxFreq or ImpDeltaFreq. Set ImpMaxFreq first, typically only for measured data.
- For every component that generates this message, fix each component one at a time to simplify the design.

## Viewing an Impulse Response

- In an S-parameter simulation, analyze over the given frequency spacing and maximum frequency
  inverse Fourier transform the response by plotting ts (x)
- In the time domain, apply an impulse and simulate
  plot the transient result
  the pulse risetime is used to set fmax and thus can influence the impulse response

## Setting ImpMaxFreq and ImpDeltaFreq

Generally a good impulse response can be found without manually setting ImpMaxFreq and ImpDeltaFreq

- If ImpMaxFreq is set, the adaptive algorithm tries different lengths but doesn't modify fmax
- If ImpDeltaFreq is set, the adaptive algorithm is disabled and the impulse is computed from ImpDeltaFreq and ImpMaxFreq
- Set ImpMaxFreq on the component, then set ImpDeltaFreq on component if necessary, and finally, set ImpMaxFreq on the transient controller if necessary
- For transmission lines, set ImpMaxFreq to at least n/td, where td is the delay time and n is a small integer (2-3)
- For lowpass and bandpass filters, set ImpMaxFreq to at least twice the upper passband edge

## Measured Data with S2P Component

- The algorithm that computes the impulse response has no special knowledge of the component it's working on and assumes data is available at any desired frequency. It has no knowledge of flow and fhigh or frequency spacing of measured data
- S2P interpolates and extrapolates data as needed
- Be sure to supply good data to prevent dangerous extrapolation extends down to DC and up to fmax
- Set ImpMaxFreq on S2P component to match frequency limits in datafile (avoid extrapolation)
- Typically there is not enough frequency-domain data in the S2P file for use in the simulation

Given a pulse with a risetime of tr, the equivalent bandwidth is 2.2/tr (0.1 ns risetime represents a 22 GHz bandwidth)

Package models typically must be measured up to 10x higher than the signal frequency to represent transmission line effects well

## Solving a Noncausal Impulse Response

This is the second most commonly encountered problem during convolution. The Time-domain simulation starts at time zero and moves forward in time, computing the value of next timepoint from all previous timepoints. And the Controller deals with this by introducing a delay to force causality.

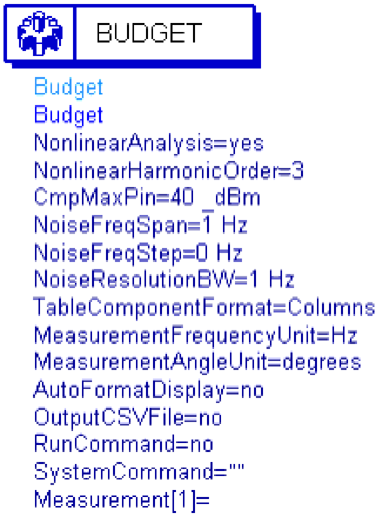
Length of delay set to ImpNoncausalLength (default=32) with timestep set by default ImpMaxFreq
Simulation results will not be accurate because of the added delay, especially if the delay is added in a critical timing or phase path.

All physically realizable devices are causal (the output is dependent only on past states and not any future states) while noncausal devices are nonphysical. Some ADS components, user-defined data or equations may be noncausal.

- Frequency-dependent real part with constant imaginary part, for example resistance as a function of frequency without any reactance
- Constant real and constant non-zero imaginary part
- Negative time delays
- INDQ, CAPQ, PLCQ, SLCQ have problems in some modes

## RF Budget Controller

Use the Budget controller for budget analysis of an RF system. This RF system budget analysis enables you to determine the linear and nonlinear characteristics of an RF system comprising a cascade of two-port, two-pin linear or nonlinear components.
The Budget controller includes a large number of built-in budget measurements and improved budget noise measurements.



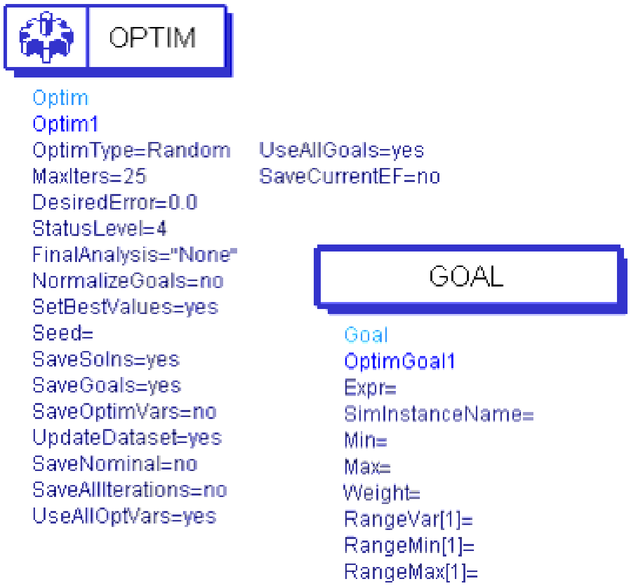
```
Budget
Budget
NonlinearAnalysis=yes
NonlinearHarmonicOrder=3
CmpMaxPin=40 _dBm
NoiseFreqSpan=1 Hz
NoiseFreqStep=0 Hz
NoiseResolutionBW=1 Hz
TableComponentFormat=Columns
MeasurementFrequencyUnit=Hz
MeasurementAngleUnit=degrees
AutoFormatDisplay=no
OutputCSVFile=no
RunCommand=no
SystemCommand=""
Measurement[1]=
```

Use the RF Budget Controller to

- Modify simulations using tuning, parameter sweeps, optimization, yield analysis, etc.
- Include AGC loops to control gain and set power levels at specific points in the RF system.
- Select alternate budget paths.

## Nominal Optimization Controller

Use the Nominal Optimization controller in combination with Goal components to satisfy predetermined performance goals.  Optimizers that compare computed and desired responses and modify design parameter nominal values to bring the computed response closer to that desired can be selected from within the Nominal Optimization controller setup.

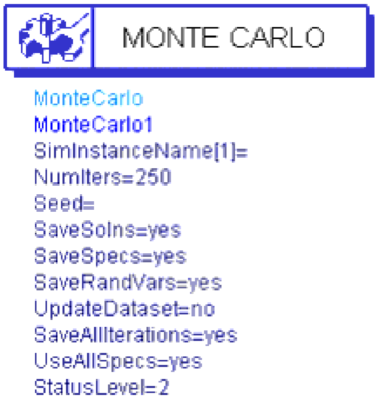


Optim
Optim1
OptimType=Random
MaxIters=25
DesiredError=0.0
StatusLevel=4
FinalAnalysis="None"
NormalizeGoals=no
SetBestValues=yes
Seed=
SaveSolns=yes
SaveGoals=yes
SaveOptimVars=no
UpdateDataset=yes
SaveNominal=no
SaveAllIterations=no
UseAllOptVars=yes
UseAllGoals=yes
SaveCurrentEF=no

Goal
OptimGoal1
Expr=
SimInstanceName=
Min=
Max=
Weight=
RangeVar[1]=
RangeMin[1]=
RangeMax[1]=

## Monte Carlo Controller

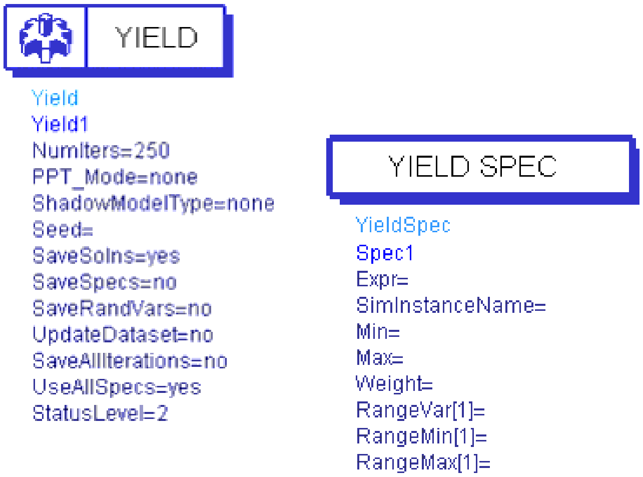
Use the Monte Carlo analysis controller to randomly vary network statistical parameter values according to statistical distributions to get the overall performance variation. This process involves simulating the design over a given number of trials in which the statistical variables have values that vary randomly about their nominal values with specified probability distribution functions.



MonteCarlo
MonteCarlo1
SimInstanceName[1]=
NumIters=250
Seed=
SaveSolns=yes
SaveSpecs=yes
SaveRandVars=yes
UpdateDataset=no
SaveAllIterations=yes
UseAllSpecs=yes
StatusLevel=2

## Yield Analysis Controller

Use the Yield Analysis controller in combination with a Yield Specification component to vary a set of statistical parameter values, using specified probability distributions, to determine how many possible combinations result in satisfying predetermined performance requirements. This process involves simulating the design over a given number of trials in which the statistical variables have values that vary randomly about their nominal values with specified probability distribution functions. The numbers of passing and failing trials are recorded and these numbers are used to compute an estimate of the yield.



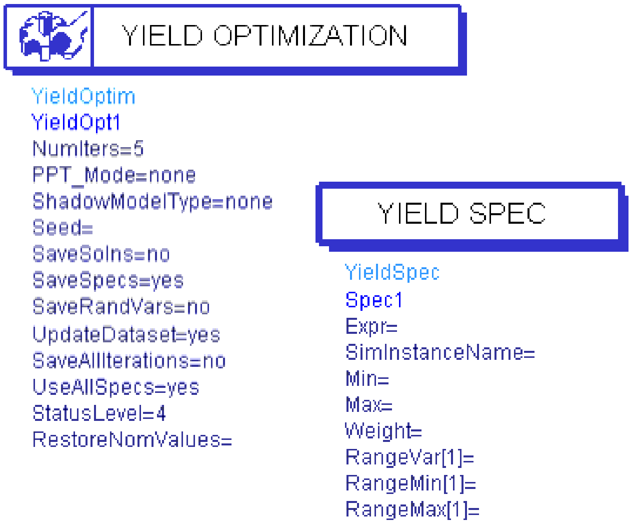
Yield
Yield1
NumIters=250
PPT_Mode=none
ShadowModelType=none
Seed=
SaveSolns=yes
SaveSpecs=no
SaveRandVars=no
UpdateDataset=no
SaveAllIterations=no
UseAllSpecs=yes
StatusLevel=2

YieldSpec
Spec1
Expr=
SimInstanceName=
Min=
Max=
Weight=
RangeVar[1]=
RangeMin[1]=
RangeMax[1]=

## Yield Optimization Controller

Use the Yield Optimization controller in combination with a Yield Specification component to perform multiple yield analyses with the goal of adjusting the nominal values of the statistical variables to maximize the yield estimate.

During yield optimization, each yield improvement is referred to as a design iteration.

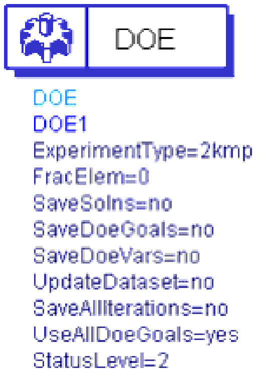


YieldOptim
YieldOpt1
NumIters=5
PPT_Mode=none
ShadowModelType=none
Seed=
SaveSolns=no
SaveSpecs=yes
SaveRandVars=no
UpdateDataset=yes
SaveAllIterations=no
UseAllSpecs=yes
StatusLevel=4
RestoreNomValues=

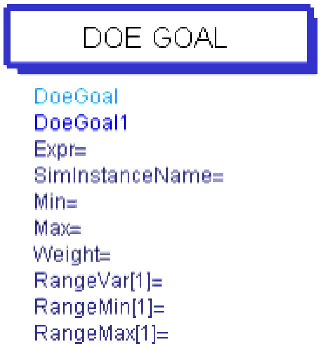
YieldSpec
Spec1
Expr=
SimInstanceName=
Min=
Max=
Weight=
RangeVar[1]=
RangeMin[1]=
RangeMax[1]=

## Design of Experiments Controller

Use the Design of Experiments (DOE) controller in combination with DOE Goal components to perform an experiment and collect response data. You can then analyze the data using statistical methods. Sequential application of this methodology can be used to improve the statistical performance of a given circuit or system. Because of an inherent compromise between statistical performance prediction accuracy and the number of input variables, a *screening* experiment is used to identify variables that contribute significantly to performance variation. Next a *refining* experiment can be used to *hone in* on the target statistical response.



DOE
DOE1
ExperimentType=2kmp
FracElem=0
SaveSolns=no
SaveDoeGoals=no
SaveDoeVars=no
UpdateDataset=no
SaveAllIterations=no
UseAllDoeGoals=yes
StatusLevel=2



DoeGoal
DoeGoal1
Expr=
SimInstanceName=
Min=
Max=
Weight=
RangeVar[1]=
RangeMin[1]=
RangeMax[1]=