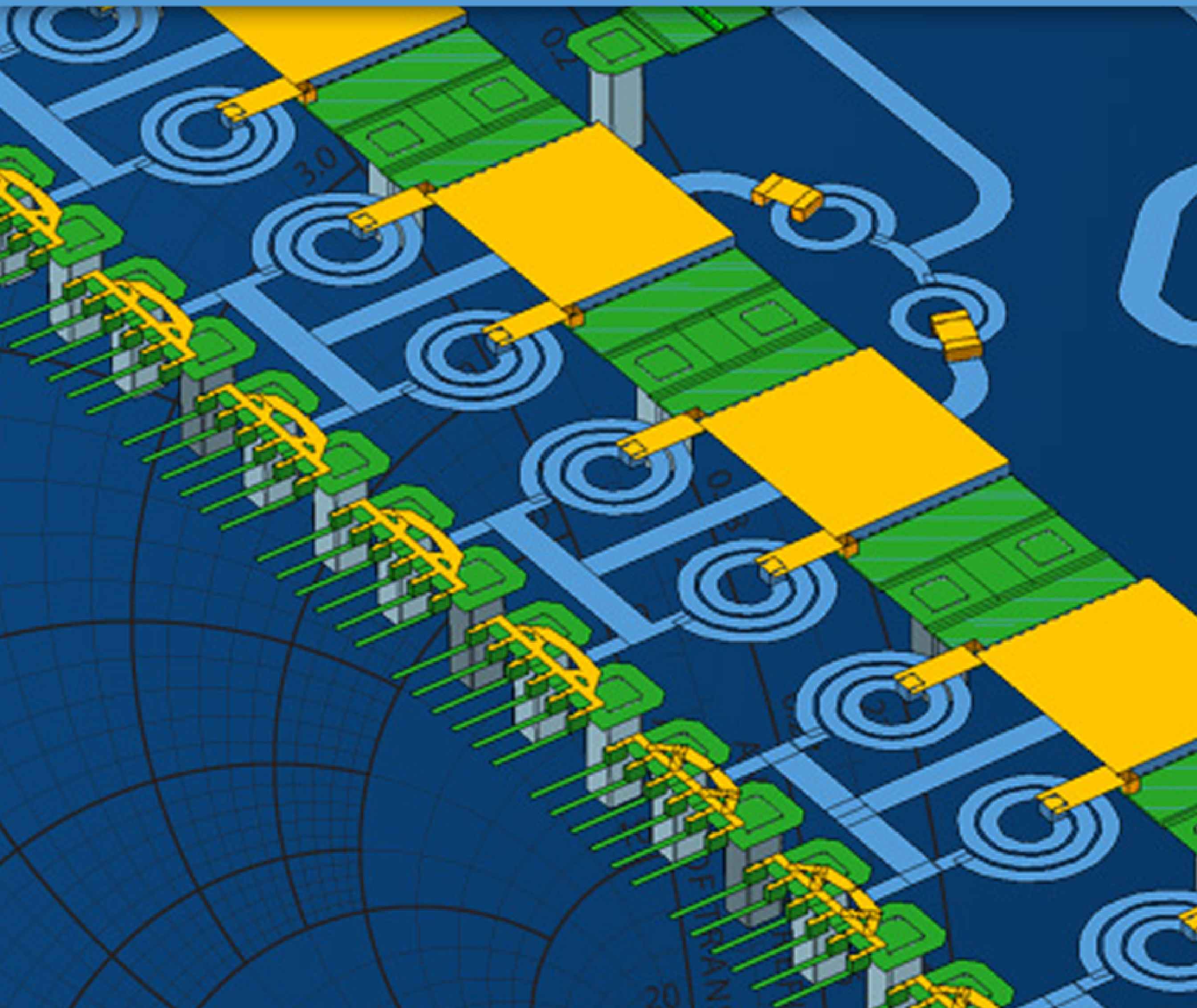
High Frequency Electromagnetic Software

SONNET® 16

Blink



blink. by SONNET®

Blink User's Guide

At Sonnet, we've been developing 3D planar high frequency EM software since 1983, and our software has earned a solid reputation as the world's most accurate commercial planar EM analysis package for single and multi-layer planar circuits, packages and antennas.

Sonnet Software Inc., founded by Dr. James C. Rautio, is a private company, entirely dedicated to the development of commercial EM software. We take great pride in providing quality technical support for our products with timely response--which we believe to be very important for high-end technical software products.

Sonnet is based in Syracuse, NY, USA with representatives across the globe.

BLINK USER'S GUIDE

Published: November 2015

Release 16

Sonnet Software, Inc.
100 Elwood Davis Road
North Syracuse, NY 13212
Phone: (315) 453-3096
Fax: (315) 451-1694
Technical Support: support@sonnetsoftware.com
Sales Information: sales@sonnetsoftware.com
www.sonnetsoftware.com

Rev 16.52

Copyright Notice

Trademarks

TABLE OF CONTENTS

Chapter 1 Cadence Virtuoso Interface

Blink’s plug-in for the Cadence Virtuoso suite enables the RFIC designer to configure and run the EM analysis from a layout cell, extract accurate electrical models, and create a schematic symbol for Analog Design Environment simulation with Spectre, HSPICE and Golden Gate by Keysight. A Broadband Spice extraction module is also available that provides high-order Spice models suitable for Spice simulation. For more information on Broadband Spice, see “SPICE Model Synthesis,” page 329 in the **Native Environment User’s Guide**.

You can take advantage of Sonnet’s accuracy without leaving the Virtuoso environment. Although, for expert users who wish to take advantage of powerful advanced features not presently supported in the integrated environment, the partnership of Cadence and Sonnet has simplified the process of moving EM projects into Sonnet and resultant models back into Virtuoso.

This manual assumes that you are familiar with the basics of using Virtuoso. If this is not true, we recommend referring to the appropriate Cadence Virtuoso documentation.To get started with Blink, we recommend the Cadence Virtuoso Interface tutorial in Chapter 5 "Blink Tutorial" on page 29 of the Getting Started with Blink manual.

Sonnet’s Cadence Virtuoso Interface provides a completely integrated “solver on request” interface between Cadence’s Virtuoso and Sonnet’s native environment, providing the choice of two Sonnet solvers: Fast and Accurate. The interface allows you to stay completely in the Cadence Virtuoso environment using Sonnet's EM analysis engines, or you , when using the Accurate solver, may choose to edit your circuits in the Sonnet native environment before running the EM analysis. Either way, your resulting models are easily integrated back into the Virtuoso environment.

NOTE: The Cadence Virtuoso Interface only supports geometry projects. It is not possible to translate a netlist project.

Design Flow for the Cadence Virtuoso Interface

This section of the manual describes the normal design flow for using the Cadence Virtuoso Interface to set up and run an electromagnetic analysis using Blink's fast and accurate solvers. This provides a general outline, but be aware that every step does not need to be done for every analysis. Oftentimes, the default values are already set correctly and no action needs to be taken for that step. Detailed descriptions of the functions of all the controls are found in Help, accessed by selecting *Sonnet ⇒ Help* from the Sonnet menu in the Cadence Virtuoso CIW, layout window or Sonnet's Cadence Virtuoso Interface main window. You may access specific topics in Help by clicking on the Help button in any dialog box in the Cadence Virtuoso Interface. For a detailed, step by step example, please see the tutorial referred to at the beginning of this chapter.

1 Creating a SonnetEM View

In order to use Sonnet as your electromagnetic simulator you must create a SonnetEM View from a layout view of a cell in Virtuoso.

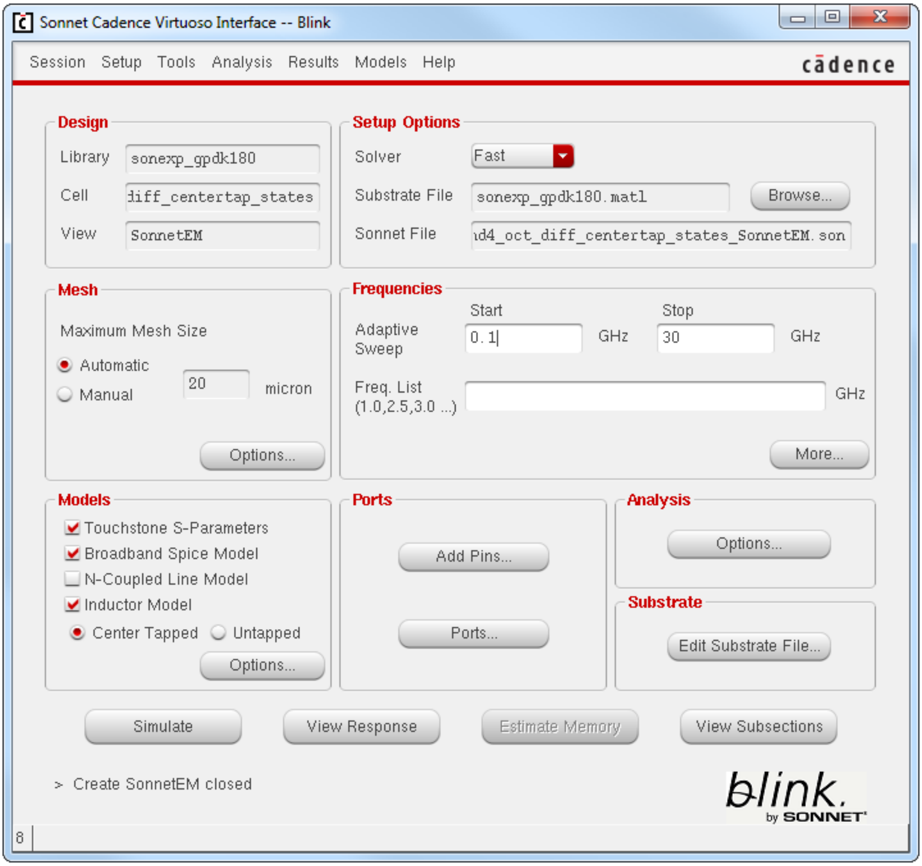
NOTE: The definition of a "cell" in the Cadence Virtuoso environment is equivalent to the "circuit geometry" in Sonnet. When "cell" is used in the Sonnet environment, it is referring to the smallest part of the analysis grid used in subsectioning your circuit.

You create a SonnetEM view by opening the desired cell's layout view in Virtuoso, then loading the Sonnet menu (*Launch ⇒ Sonnet* in IC6 and *Tools ⇒ Sonnet* in IC5). Once the Sonnet menu appears in the layout window, you select *Sonnet ⇒ Create SonnetEM view* to open the Create SonnetEM view from Layout View dialog box which allows you to create the SonnetEM view. For details on the controls in this dialog box, please click on the Help button in the dialog box.

2 Opening the Cadence Virtuoso Interface

Once you have created the SonnetEM View, that view is opened along with the Cadence Virtuoso Interface. To open the interface after the SonnetEM view has been created, open the SonnetEM view in a layout window, load the Sonnet menu (*Launch* ⇒ *Sonnet* in IC6 and *Tools* ⇒ *Sonnet* in IC5), then select *Sonnet* ⇒ *Open Interface Application* to open the interface.

This opens the Cadence Virtuoso Interface which allows you to setup and run an analysis for the circuit shown in the SonnetEM view. The Cadence Virtuoso Interface main window, shown below, is used to perform the rest of the tasks in the design flow

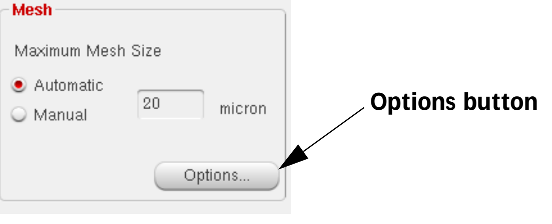


3 Choosing the Solver

Once you have created a SonnetEM View, you need to choose which Sonnet solver you wish to use to perform your analysis: Fast or Accurate. The solver you choose affects which setup tasks you need to perform and which options are applicable. For details on the differences between the solvers, please see "Solver Differences" in Help (see "solver differences" in the index).

4 For the Fast solver, set the mesh parameters.

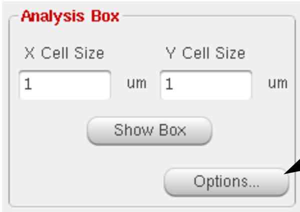
These settings are only available when using the Fast Solver. You control the Fast Solver Mesh Options in the Mesh section of the main window. Clicking on the Options button in the Mesh section of the main interface window opens the Fast Solver Mesh Options dialog box which includes the Maximum Mesh Size settings from the main window and the Shield Minimum Mesh Size settings. Please select *Sonnet* ⇒ *Help* from the main menu of the Cadence Virtuoso Interface, then see the Mesh section or click on the Help button in the Fast Solver Mesh Options dialog box for details on the settings.



5 For the accurate solver, set the Analysis Box.

When using the Accurate solver, you need to specify the box used in Sonnet. The Accurate solver analyzes planar structures inside a six-sided metal shielding box. Port connections are usually made at the box sidewalls. The substrate is on the metal bottom of the box and represents the ground plane. Above the substrate are stacked the dielectric layers and metal levels. The side walls of the box are modeled as lossless metal. The top and bottom of the box may be assigned any metal type defined in the geometry project.

You use the Analysis Box Options dialog box to control these settings. You open the dialog box by clicking on the Options button in the Analysis Box section of the Cadence Virtuoso Interface main window.Note that when you are using the Fast solver, the Analysis Box section is not displayed in the interface.

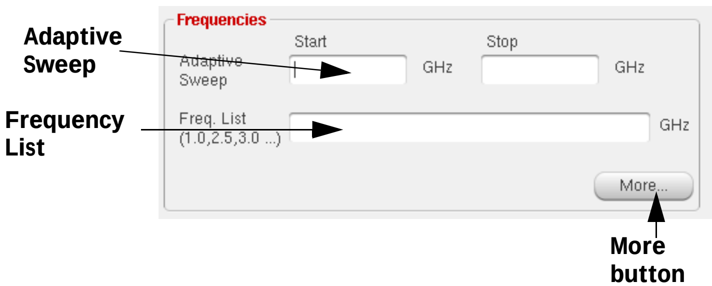


Analysis Box Options button

Please select *Sonnet* ⇒ *Help* from the main menu of the Cadence Virtuoso Interface, then see the Analysis Box section or click on the Help button in the Analysis Box Options dialog box for details on the settings.

6 Set the Analysis Frequencies.

You are able to set up multiple frequency sweeps for your Sonnet analysis using the Frequencies section of the main interface window. You may enter an Adaptive Band Sweep (ABS) or list of frequencies directly in the main interface window, as pictured below. Other types of sweeps must be entered in the Frequencies dialog box opened by clicking on the More button in the Frequencies section of the interface.

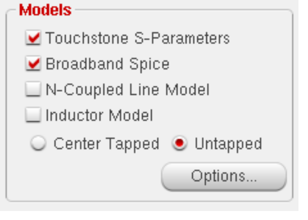


If the only types of frequency sweeps entered are the Adaptive Sweep and/or list of Frequencies, then the sweeps are displayed on the main page. If you enter additional frequency sweeps in the Frequencies dialog box then all of the sweeps defined for your analysis are displayed in the Frequencies dialog box. In that case, a message appears in the main interface directing you to the Frequencies dialog box.

Please select *Sonnet* ⇒ *Help* from the main menu of the Cadence Virtuoso Interface, then see the Frequencies section or click on the Help button in the Frequencies dialog box for details on the different types of frequency sweeps and settings.

7 **Select the models you wish to output.**

This Models section of the Sonnet Cadence Virtuoso Interface, shown below, allows you to select the type of model(s) you wish to produce when you run your Sonnet EM analysis. There are four different types of models available in the main interface window: S-Parameters in Touchstone format, a Broadband Spice model in Spectre format, a N-Coupled Line model, and an Inductor model. Additionally, you may select S-Parameters in Spectre format in the Advanced Model Options dialog box, opened when you click on the Options button.



You may select to produce any or all of the models by selecting the appropriate checkboxes. You may also choose to create no models at all.

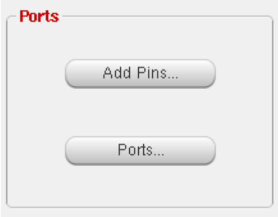
If you do not create models as part of your simulation, it is possible to create the model views afterwards using the Models menu in the Sonnet Cadence Virtuoso Interface, but only if the raw data is available.

Please select *Sonnet* ⇒ *Help* from the main menu of the Cadence Virtuoso Interface, then see the Models section or click on the Help button in the Advanced Model Options dialog box for details on the different types of models and their settings.

8 **Setup ports.**

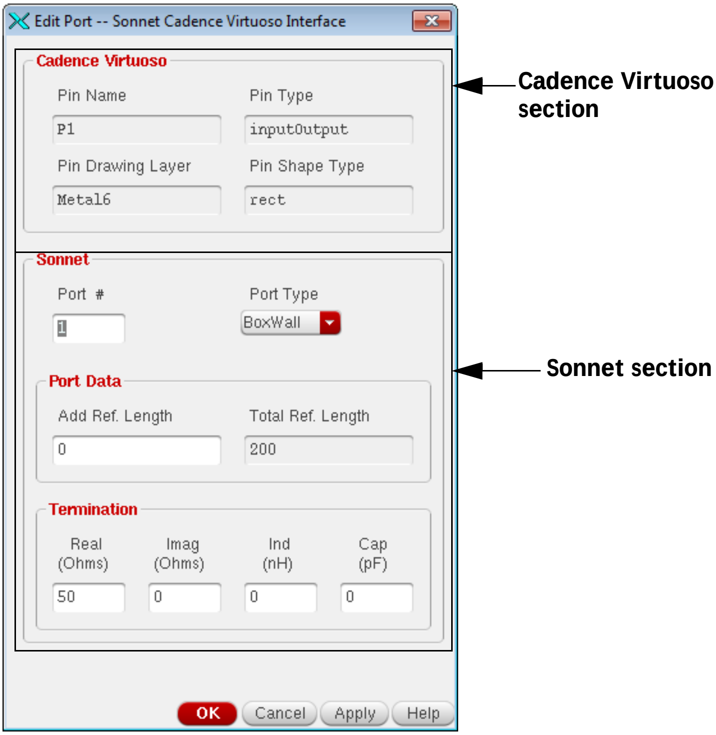
You need to specify the parameters for the ports to be used in Sonnet and the use of reference planes for the Accurate solver. Virtuoso pins are converted to Sonnet ports. All pin types, with the exception of symbolic, may be used in Virtuoso when you are using Sonnet as your electromagnetic solver. Pins can be converted to standard box-wall ports, co-calibrated ports, autogrounded ports, and via ports. Internal ports are supported by the Accurate solver, but not available when using the Fast solver.

The Ports section of the main interface window provides a shortcut to Cadence Virtuoso's Create Shape Pin dialog box and access to the Ports dialog box. Clicking on the Add Pins button opens the Create Shape Pin dialog box and clicking on the Ports button opens the Ports dialog box. When a Pin is added, it is automatically converted to a Sonnet port. You may add the shape pins before creating the SonnetEm View or add them in the SonnetEm View using the Add Pins button. For more information on the Create Shape Pin dialog box, please refer to Cadence Virtuoso's help.



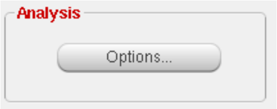
NOTE: **For IC5, the Create Pin dialog box is opened. When you click on the Shape radio button then the Create Shape Pin dialog box is displayed.**

Once the pins have been added, they are automatically converted to Sonnet Ports. To change the parameters of the ports, you click on the Ports button to open the Ports dialog box. For more information on ports and their parameters, please click on the Help button in the Ports dialog box. The Cadence Virtuoso section displays information about the pin in Virtuoso.. The Sonnet section of the dialog box defines how the Virtuoso pin will be translated to a Sonnet port. The fields are discussed below.



9 Setting the Analysis Options

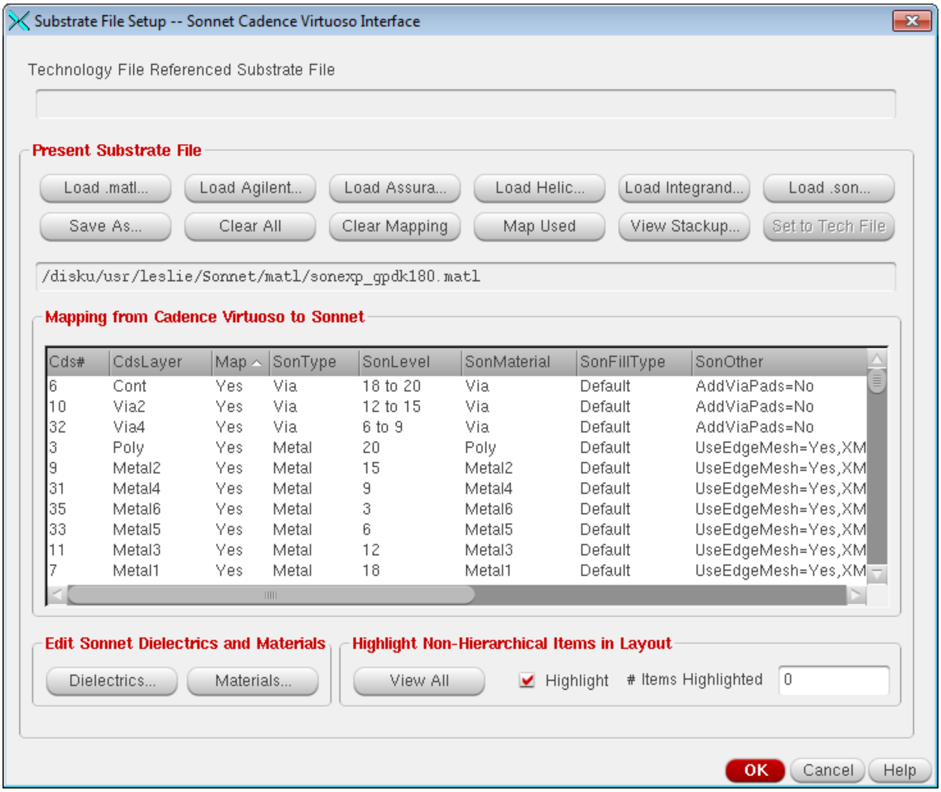
The next section of the main interface window is the Analysis section which allows you to control the settings and run options for the Accurate solver, ***em*** or the Fast solver. Clicking on the Options button opens the Analysis Options dialog box if the Accurate solver is selected and the Fast Solver Options dialog box if the Fast Solver is selected.



For a detailed discussion of the analysis options for the Accurate solver, please click on the Help button in the Analysis Options dialog box. For a detailed discussion of the analysis options for the Fast solver, please click on the Help button in the Fast Solver Options dialog box.

10 Setting up the substrate.

You need to define the layer mapping from Virtuoso to Sonnet, and the dielectric layers and materials to be used in your Sonnet analysis. You use the Substrate File Setup dialog box, shown below, to control these settings. You open the dialog box by clicking on the Edit Substrate button in the Substrate section of the Cadence Virtuoso Interface. For a detailed explanation of the controls and settings, please click on the Help button in the Substrate File Setup dialog box



11 Pre-check your circuit before simulating.

There are three commands available in the Cadence Virtuoso Interface that allow you to check your circuit before running an analysis: Estimate Memory, Estimate Box Resonances and Check Connectivity.

NOTE: **The Estimate Memory and Estimate Box Resonances commands are only available for the Accurate Solver.**

Analysis ⇒ Estimate memory subsections your circuit and estimates the memory required for the analysis. *Analysis ⇒ Estimate Box Resonances* allows you to detect possible box resonances before running your simulation. *Tools ⇒ Check Connectivity* opens your circuit in the project editor so that you can check for any opens or shorts in your circuit that might have been created during the translation. For more information about these commands, open Help by selecting *Sonnet ⇒ Help* from the main menu of the Cadence Virtuoso Interface window, then look for the Analysis menu and Tools menu in the Table of Contents under Cadence Virtuoso Interface. For the Connectivity checker, you may also refer to "Connectivity Checker," page 29.

12 Run the simulation.

You may run your analysis by clicking on the Simulate button in the main interface window or by selecting *Analysis ⇒ Tools* from the main menu of the interface window.

For longer analysis runs, you may want to make the license for the Sonnet Cadence Virtuoso interface available in order to work on another project or to allow another user access to the interface. In those cases, you can use the command *Analysis ⇒ Simulate and Release* to run your analysis.

For more information about these commands, open Help by selecting *Sonnet ⇒ Help* from the main menu of the Cadence Virtuoso Interface window, then look for the Analysis menu in the Table of Contents under "Cadence Virtuoso Interface."

13 View your simulation results.

Blink provides two post processing applications, the response viewer, and the current density viewer, that allow you to observe your data. There is also a complete log of your analysis run, as well as the ability to create automatic documentation for your analysis. For more information, open Help by selecting *Sonnet ⇒ Help* from the main menu of the Cadence Virtuoso Interface window, then see the Results menu in the Table of Contents under "Cadence Virtuoso Interface."

Sonnet's Component Feature

When you are using the Accurate solver, you may use all Component types from Sonnet in the interface, but if you wish to connect your Virtuoso Models inside the Virtuoso environment with your Sonnet simulated results, we would recommend using the Ports Only type of Component. For a detailed discussion of Components as used in Sonnet, please refer to Chapter 5, "De-embedding" in the **Native Environment User's Guide**. Components are not supported by the Fast solver.

Creating the Models and Model Views

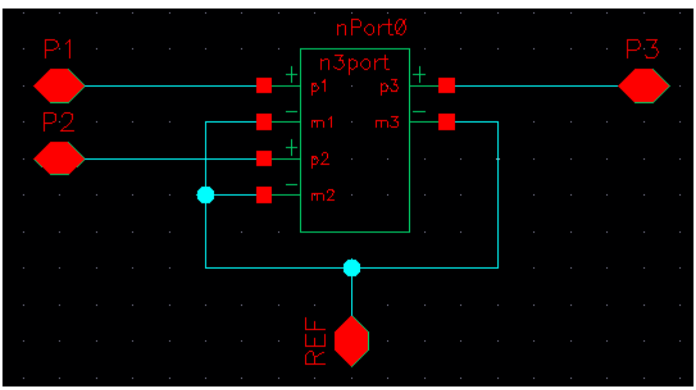
You may choose to create models and model views as part of your analysis but you may also create the model views after the analysis, using the model data generated during the analysis which is stored in your translated Sonnet project. If you have used the Simulate and Release command to run your analysis, then you must create the model views manually.

To create the model views after the completion of an analysis, you use the Models menu in the Sonnet Cadence Virtuoso Interface. For more information, open Help by selecting *Sonnet ⇒ Help* from the main menu of the Cadence Virtuoso Interface window, then see the Models menu in the Table of Contents under "Cadence Virtuoso Interface."

Using the Models

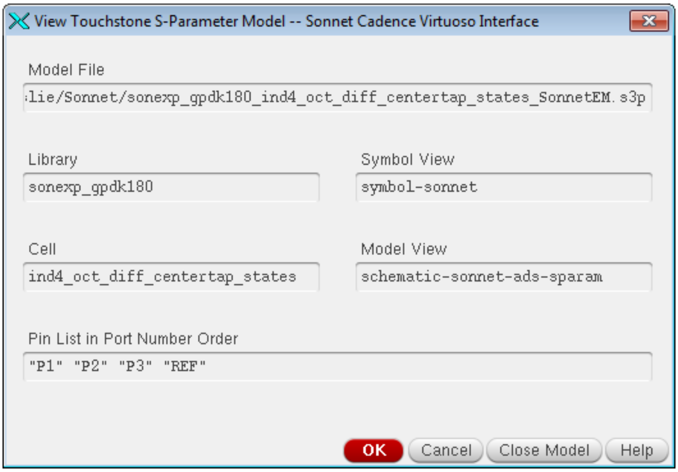
There are five models available: Touchstone S-Parameters, Spectre S-Parameters, Broadband Spice, N-Coupled Line and Inductor. The five types of models are handled in a very similar manner so that the ensuing discussion applies to all types of models.

When you specify any model, a schematic view of the model and a symbol for the model are created. The symbol can be a layout look-alike (default) or a black box. To view the model, select *Models ⇒ <Model Type> ⇒ Show View* to open a view of the schematic, the symbol and the View <Model Type> Model dialog box. An example is shown below. The "Add Ref. Pin" option was selected in the Model Symbol Options dialog box, so that a reference pin appears in the schematic.

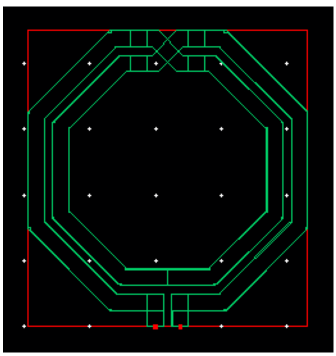


Schematic View of a Spectre S-Parameter Model

The View Model dialog box, which appears when you view the model, provides information about the model including the name of the symbol view and the model view.



For all models, the Symbol view is automatically created along with the model. You use the symbol view in your schematic design. You plug in the symbol created by the Cadence Virtuoso Interface so that the S-parameter model or Broadband Spice model will be used in the Cadence simulation.

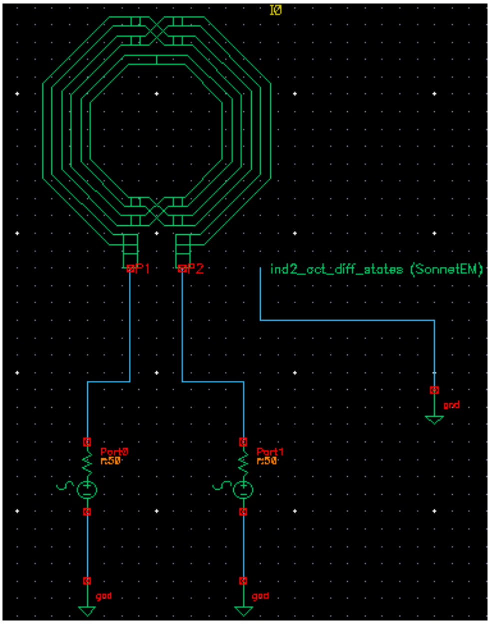


A layout look-alike symbol for an S-Parameter Model or Broadband Spice model. This is used in your larger design to "plug in" the desired model at the appropriate location(s).

In order to use a model in the Analog Design Environment, it is recommended that you create a config view which references the Sonnet symbol. This is because when you create multiple models from an analysis, only one symbol is created and shared among all of the models. The configuration view allows you to easily switch which model the symbol is representing. To set up the environment to use your model, do the following:

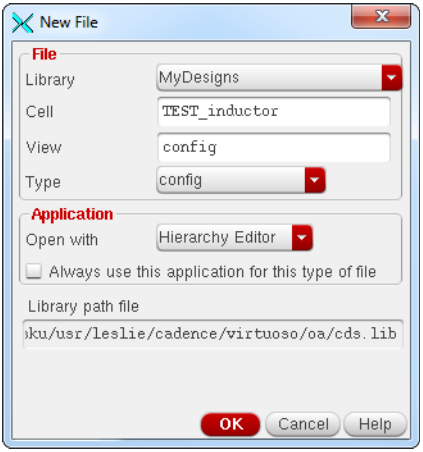
1 **Create a test bench schematic using your Sonnet symbol.**

An example is shown below, using a layout look-alike symbol.



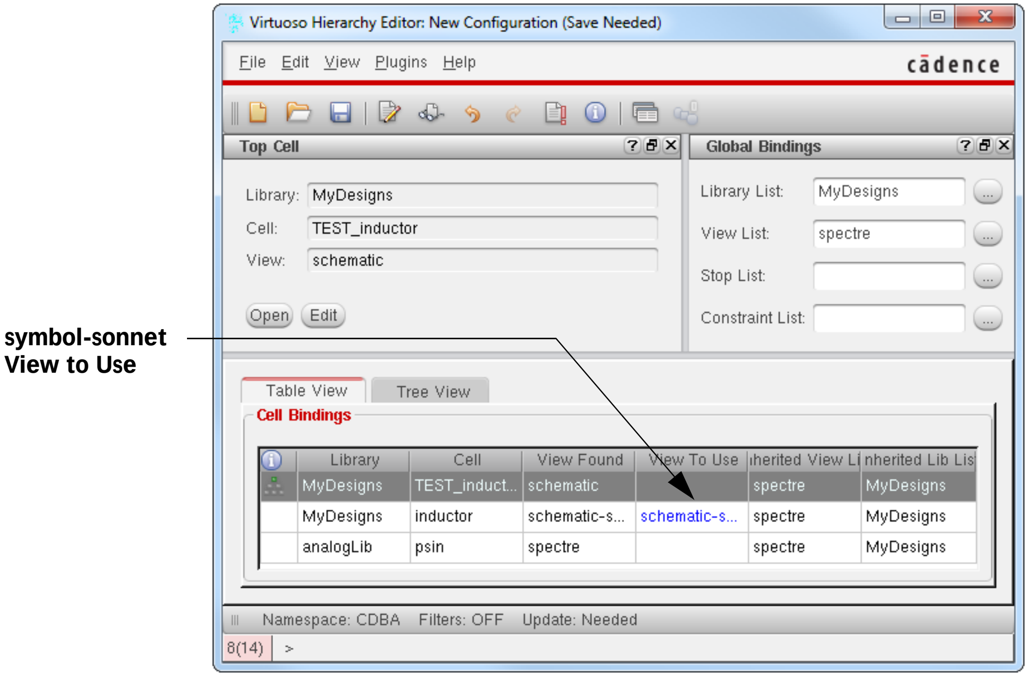
2 **Create a Config view based on the test bench schematic.**

When you create the Config cell use “config” as the view.



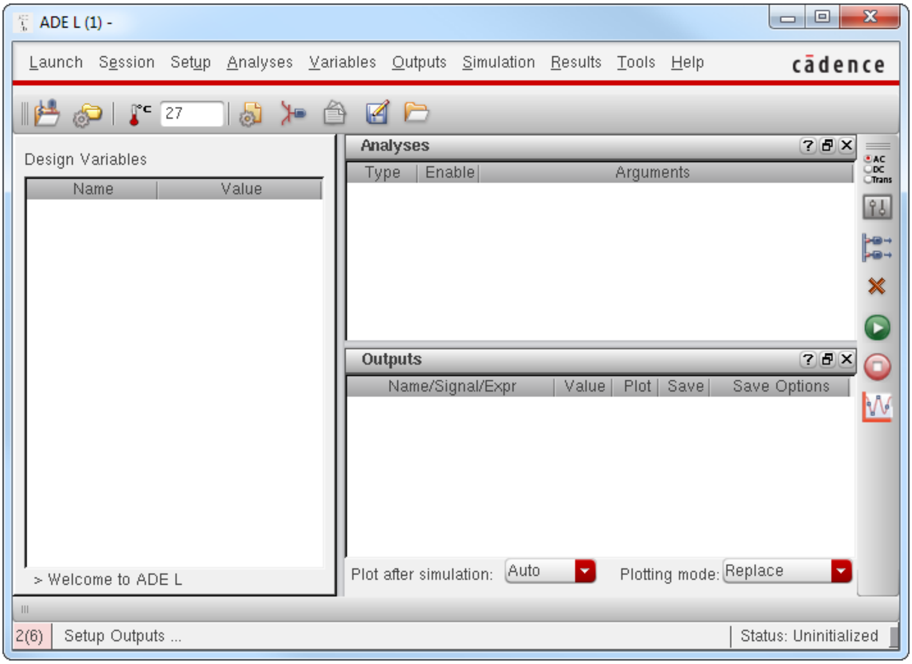
3 **In the Hierarchy Editor for the Config view, enter the model you wish the symbol to point to under View to Use for the cell view from which the test bench was created, then save the Config cell.**

An example, with schematic-sonnet-cds-param entered as the model, is pictured below. To change what model the symbol references, you would simply edit the config view.



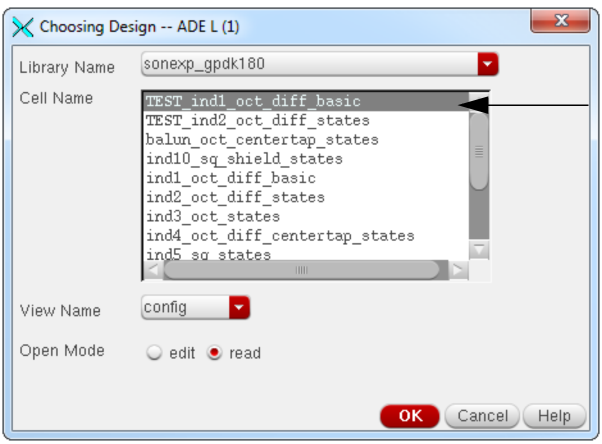
4 Select *Tools* ⇒ *ADE L* from the main menu in the CIW.

The Virtuoso Analog Design Environment window appears on your display.



5 Select *Setup* ⇒ *Design* from the main menu of the Analog Design Environment.

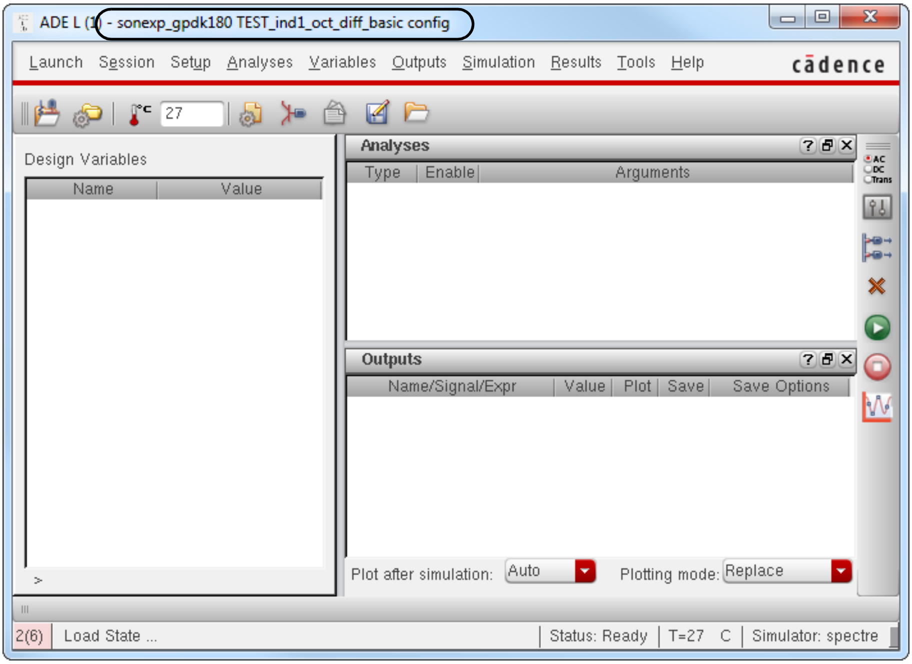
The Choosing Design dialog box appears on your display.



You should select your Test bench schematic and choose “config” for the View name.

6 **Select the Library and Cell you wish to use and select "config" for the view, then click OK.**

When you choose the Test bench Cell name, the View Name should be set to config, as shown above. When you close the dialog box, the title bar of the Analog Design Environment window is updated with the Cell you have chosen.



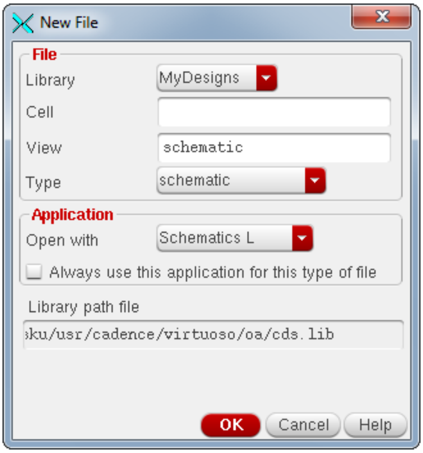
The model is loaded and ready to use in your simulation.

Using the Symbol Model Utility

This utility allows you to input a Broadband Spice model in the Spectre format (.scs) from Sonnet and use it in your symbol library in Virtuoso. To create the symbol, you perform the following:

7 Select *File* ⇒ *New* ⇒ *Cellview* from the main menu in the Cadence CIW.

The Create New File dialog box appears on your display.



8 Select your Library name from the Library name drop list.

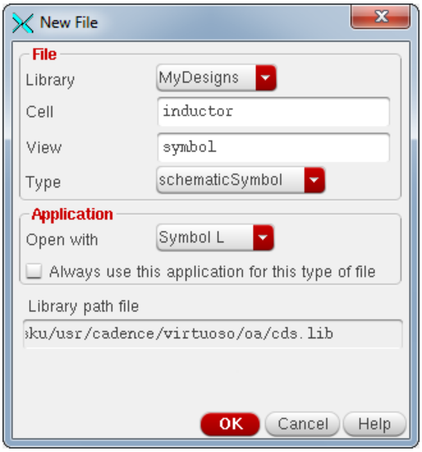
This should be the working library in which you wish to use the imported Broadband Spice model.

9 Enter the desired cell name in the Cell Name text entry box.

This will be the name of the symbol in your Virtuoso library.

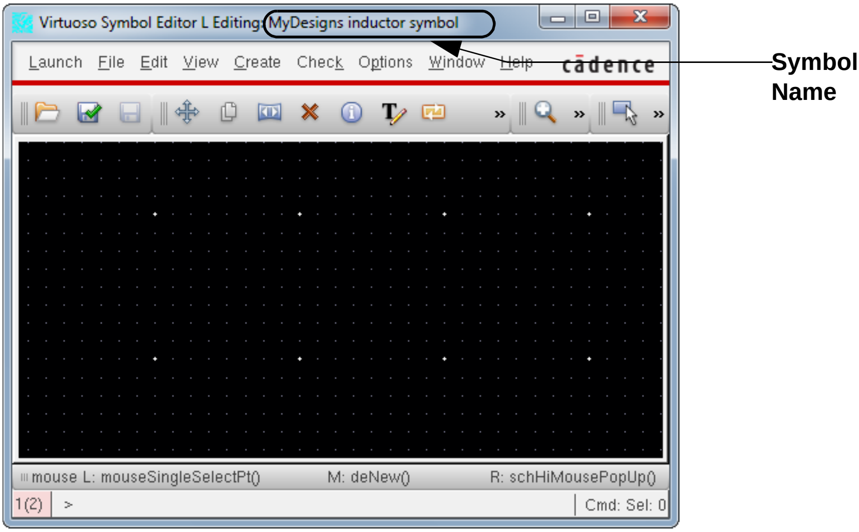
10 Select “schematicSymbol” from the Type drop list.

The View Name will be updated to “symbol” after this selection. The dialog box should appear similar to that shown below.



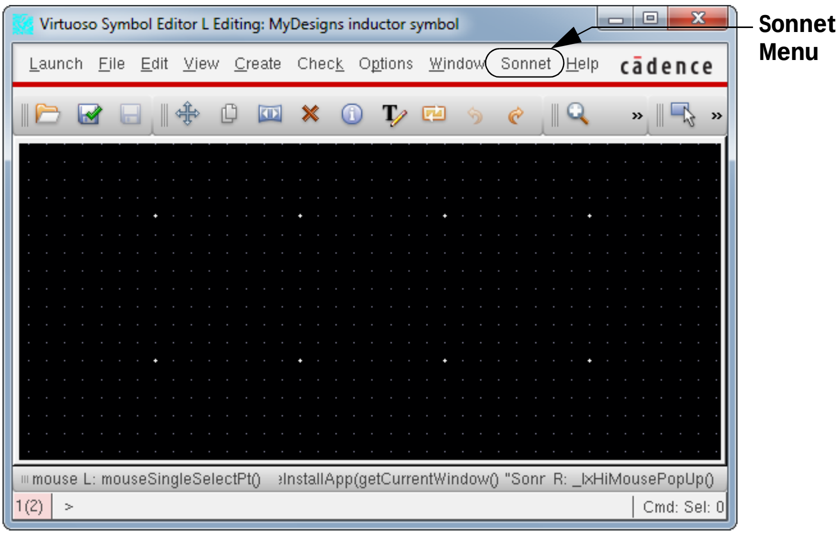
11 Click on the OK button.

The Virtuoso Symbol Editing window appears on your display with your library and symbol name displayed in the title bar.



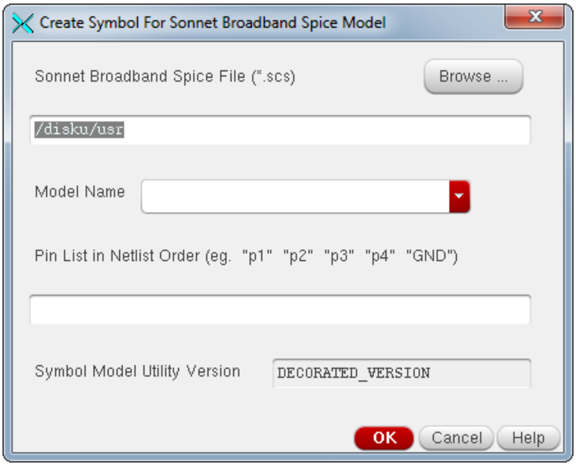
12 Select *Launch* ⇒ *Sonnet* from the menu in the Virtuoso Symbol Editing window.

This loads the Sonnet menu in the Editing window.



13 Select *Sonnet* ⇒ *Create Symbol for Sonnet Broadband Spice Model* from the menu in the Virtuoso Symbol Editing window.

The Create Symbol for Sonnet Broadband Spice Model dialog box appears on your display.

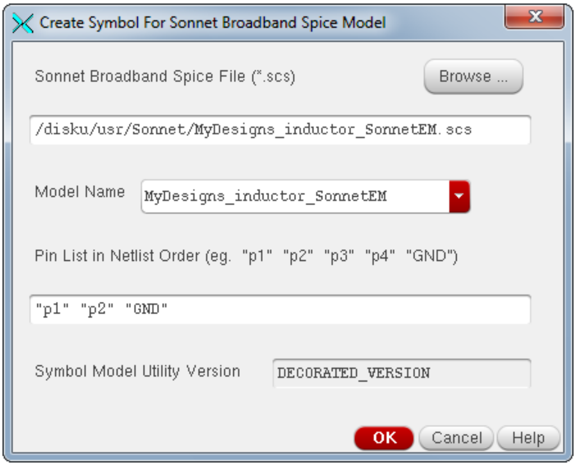


14 Enter the pathname of the Broadband Spice model file (.scs) you wish to convert in the text entry box.

You may also click on the Browse button which opens a browse window which allows you to select the desired Spice model file. When you enter the filename the model name is updated with the basename of the Spice Model file.

15 Enter the Pin List for the symbol in Netlist order in the text entry box provided.

Each pin represents a port in your Broadband Spice model. You will also need to add a grounding port, so that the total number of pins will always be equal to the number of Sonnet ports plus one. The dialog box should appear similar to that shown below.



16 Click on the OK button to close the dialog box and create the symbol.

If the symbol is created successfully, the message shown below appears on your display.

17 Click on the Close button.

The symbol appears in the Virtuoso Symbol Editing window. This completes the creation of the symbol which is now available in your library for your use. For a discussion of how to use your Broadband Spice model, please see "Using the Models" on page 17.

States and Properties of the Interface

Saving a State

Once you have completed the setup of your Sonnet simulation environment, specified your analysis frequencies and specified which types of models you wish to create, you may wish to save all these settings for use with this or other SonnetEM views. In order to save all the settings, you select *Session ⇒ Save State* from the main menu in the Cadence Virtuoso Interface. The Save State dialog box appears on your display. For details on the controls in this dialog box, please click on the Help button in the dialog box.

Loading a State

Whenever you start a session in the Cadence Virtuoso Interface, all the options in the interface revert to their default states. To restore previously supplied settings, you must use the Load State command. To load a state, you select *Session ⇒ Load State* from the main menu in the interface. The Load State dialog box appears on your display. For details on the controls in this dialog box, please click on the Help button in the dialog box.

Similar to the Save State dialog box, this dialog box allows you to select a saved state and which components of that state you wish to load. You can use the Library and Cell drop lists to navigate to other states. When you click on the OK button, the property sets of the state you selected are loaded into the interface.

Translating the Cell to a Sonnet Project

If you wish to translate your Virtuoso SonnetEM View into a Sonnet project but not execute a simulation, you may use the *Tools ⇒ Translate Only* command, available in the main menu of the Sonnet Cadence Virtuoso Interface. When you select the command, the Save As Sonnet dialog box appears to display the project to which the translation will be written. When you click on the OK button the Virtuoso SonnetEM View is translated and written to the specified Sonnet project.

Sonnet Features Not Available in Cadence Virtuoso

If you edit the project in Sonnet be aware that there are Sonnet features which may not be used in the Virtuoso Environment. Major features that do not translate are listed below:

Metal Types: Metal types in Sonnet may be assigned on a polygon basis when using the Accurate solver. You may have two polygons on a metal level using a different metal type. In Virtuoso, and when using the Fast solver, the metal type applies to a whole metal level. All polygons on a metal level must use the same metal type.

Parameterization and Optimization: Geometry parameters are available in Sonnet’s native environment, but may not be used in a Virtuoso SonnetEM View. Therefore, parameterization or optimization of a circuit may not be done from within the Virtuoso environment. However, it is possible to translate the Virtuoso SonnetEM View to Sonnet and run a parameterization or optimization in the Sonnet environment, then use the results to change the geometry in Virtuoso to the optimal values. **Accurate solver only**.

Ports: Virtuoso symbolic pins may not be used in translation; all other Virtuoso pin types may be used.

Reference Planes: You may set a fixed reference plane in the Cadence Virtuoso Interface for the Accurate solver, but you may not use a linked reference plane where the length of the reference plane is linked to a vertex of a polygon.

The following Sonnet native environment features are not available in Virtuoso: dimensions, subdividers, dielectric bricks, calibration lengths, and parallel subsections.

Using Sonnet's Project Editor

If you have access to Sonnet's native environment project editor from the same computer on which you are running Virtuoso, it is possible to open the Virtuoso SonnetEM View in the project viewer/editor for viewing or editing (Accurate solver only). This allows you to tweak the settings or take advantage of features not available in the Cadence Virtuoso Interface. If you do use features not available, it is important to run your simulation and create models which may then be imported into Virtuoso before you translate the cell from Virtuoso again. The only solver available for your simulation is the Accurate solver. Translating the circuit from Virtuoso either by using the Translate Only command or the Simulate command will overwrite any features not available in the Cadence Virtuoso Interface.

If you are using the Fast Solver, you may only view your project, you may not edit your layout.

For more information, open Help by selecting *Sonnet ⇒ Help* from the main menu of the Cadence Virtuoso Interface window, then see the Tools menu in the Table of Contents under “Cadence Virtuoso Interface.”

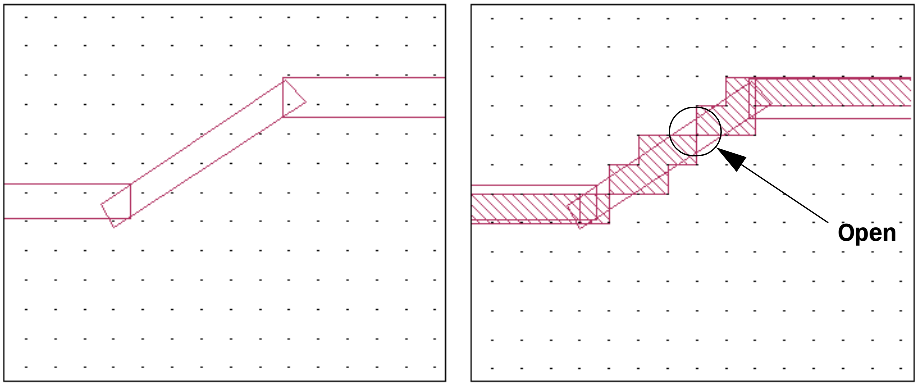
Connectivity Checker

Once you have completed the setup of your Sonnet simulation environment, but before you run your analysis, you may wish to check your circuit to ensure that no opens or shorts are created during the translation. The Connectivity Checker is a tool available in the project editor which allows you to visually check for opens or shorts in your circuit.

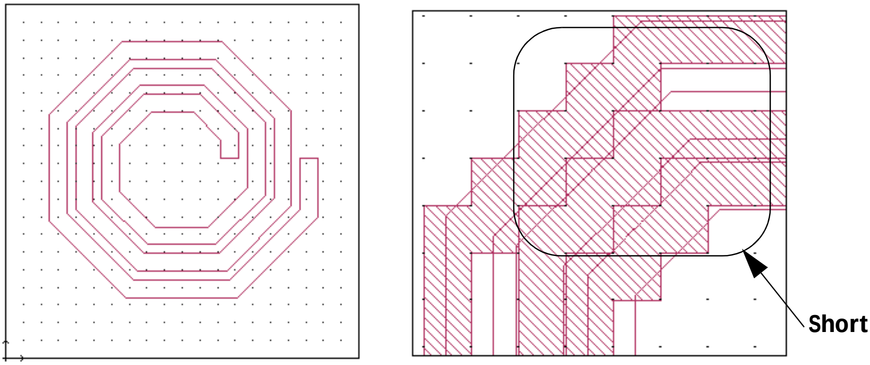
To use the Connectivity Checker, you select *Tools ⇒ Check Connectivity* in the main window of the Cadence Virtuoso Interface. Your layout is translated and the project editor (native editor) is opened in view only mode with the Connectivity Checker turned on. Once the checker is invoked you can click on any metal or via polygon in the circuit to see the polygons to which it is electrically connected or any shorts created by the polygon. The process is detailed below.

NOTE: **For via polygons, you must click on the top level of the via.**

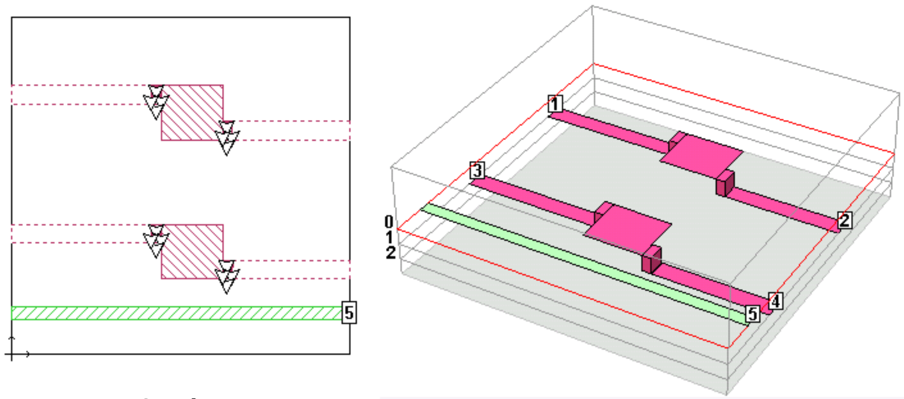
The Connectivity checker uses the cell fill when checking the connectivity. The cell fill represents the actual metalization analyzed by the Accurate solver, ***em,*** and may differ from the polygons input by the user, or translated from another tool. Using the Connectivity Checker can identify problems created by the differences between the input polygons and the cell fill. An example where the cell fill creates an open is shown below. The circuit is shown on the left with the cell fill turned off so you see only the input polygons. On the right is a view of the input polygons with the cell fill turned on. The input polygons provide a connected electrical path, but because of the staircase fill, the solver would treat this as an open circuit.



Cell fill can also create an undesired short as shown below.

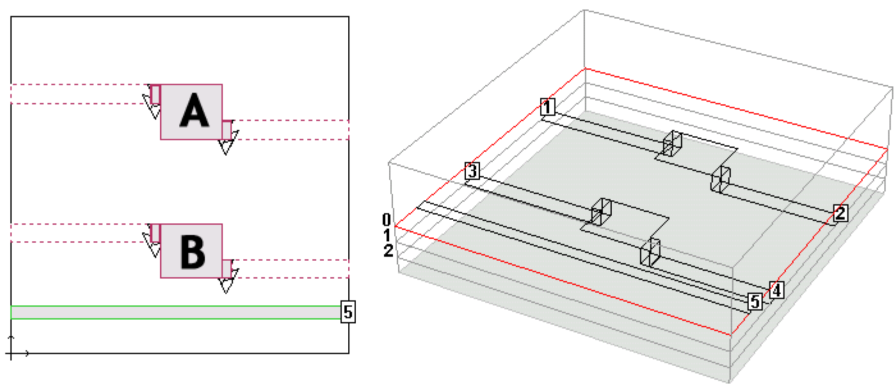


An example circuit, contained in the Connectivity example, will be used to explain the operation of the Connectivity Checker and is shown below. This example file can be found in the Sonnet example files. You may access the Sonnet Example browser by selecting *Help ⇒ Browse Examples* from the menu of any native environment application. For instructions on using the Example Browser, please click on the Help button in the Example Browser window.

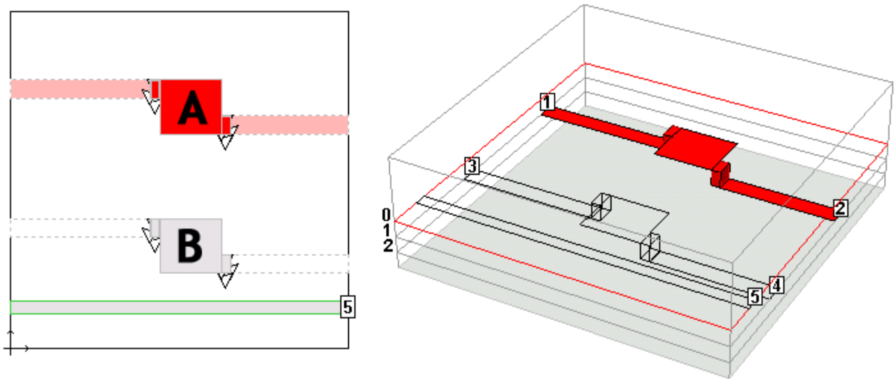


2D View　　3D View

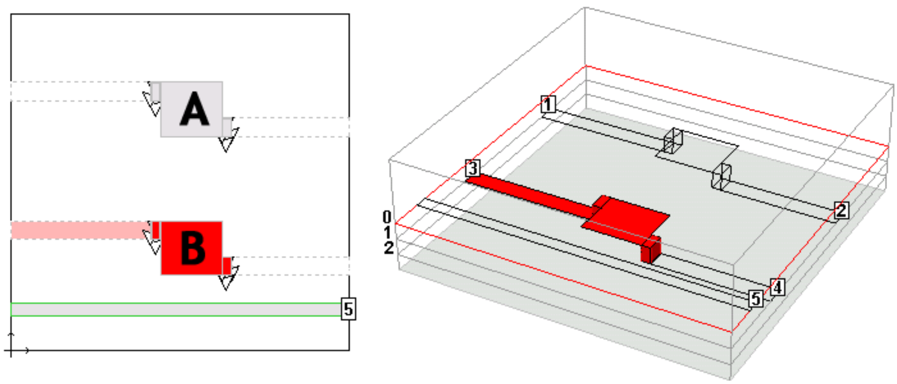
When you invoke the Connectivity Checker, all the polygons and vias in the 2D view are displayed with grey fill while the circuit in the 3D view is displayed as a wire frame as illustrated below.



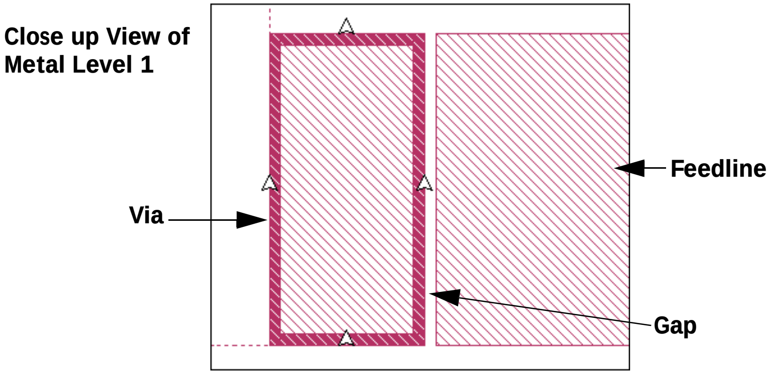
When you click on a polygon, it and all the polygons to which it is electrically connected are highlighted in red. If there are polygons or vias that are connected on another level, they are highlighted in a lighter red. In the graphic below, polygon A was selected. The path starting at Port 1 and extending to Port 2 was highlighted. As noted earlier, the feedlines are on another level from the center polygon so in the 2D view they are light red. The whole path is highlighted in red in the 3D view.



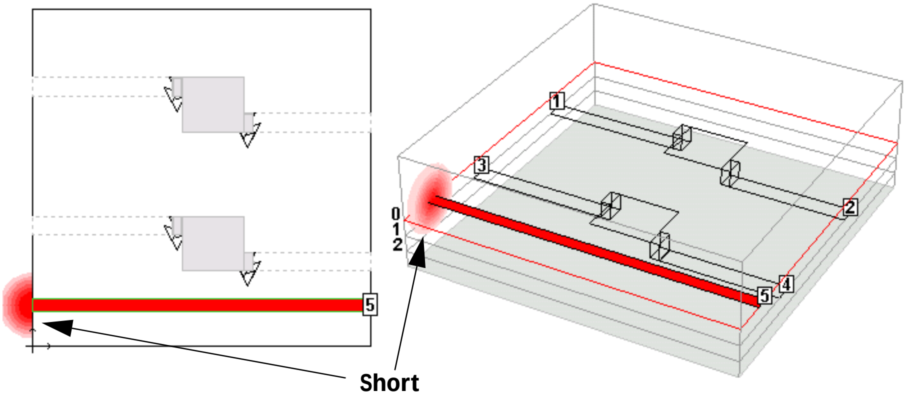
In the next graphic, shown below, polygon B is selected. Notice that the feedline extending from Port 4 is not highlighted. This indicates an open circuit between the via and the feedline.



The gap between the via and the feedline is not discernible at full view, but zooming in on the connection shows that it exists.

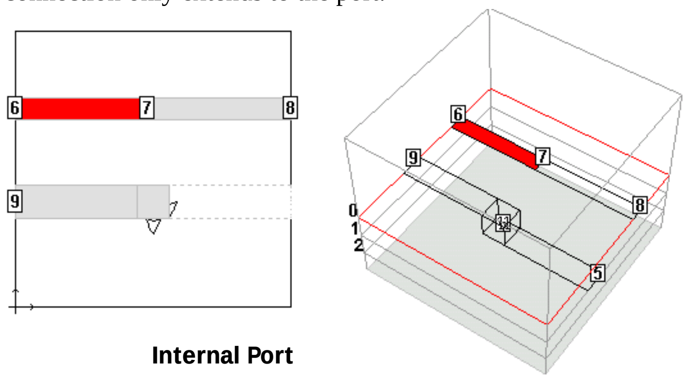


In the next illustration, the trace at the bottom of the circuit is selected. There is a “bloom” shown at one end of the line. Since there is no port on the box wall, the trace is shorted to ground; this is indicated by the “bloom.” The other end of the polygon has a box wall port. Ports are always considered opens which is why there is no bloom on the other end of the polygon.

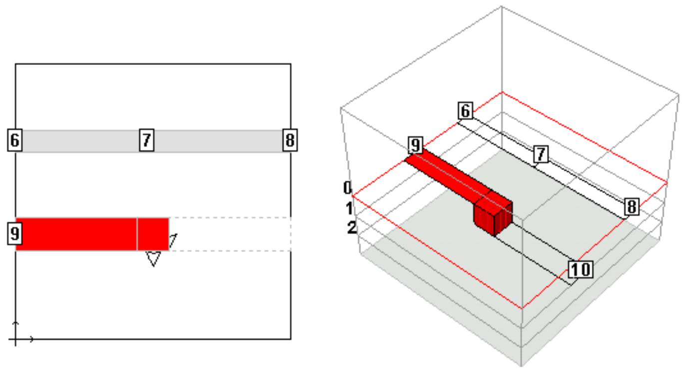


Once you are finished using the Connectivity Checker, press the Escape key to exit this mode and return to normal View mode. To exit the project editor, select *File ⇒ Exit*.

As mentioned earlier, ports are always considered an open circuit. This includes both internal ports and via ports as pictured below. For both the internal port, shown on the top in 2D and 3D views, and the via port pictured on the bottom, the electrical connection only extends to the port.



Internal Port



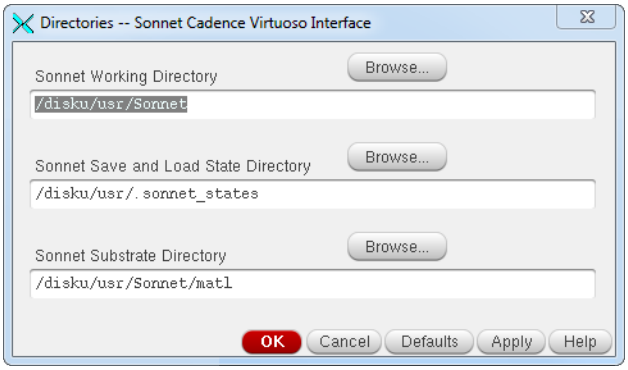
Via Port

A complete list of geometry elements and how they are treated by the connectivity checker is provided below.

- **Ports:** All port types are treated as an open.
- **Components:** Components are treated as an open.
- **Dielectric Bricks:** Dielectric Bricks are treated as an open.
- **Lossy Metal:** Lossy Metal is treated as a closed circuit. The loss of the metal is not taken into account by the Connectivity Checker.

Changing Working Directories

When you select *Setup* ⇒ *Directories* from the main menu of the Cadence Virtuoso Interface the Directories dialog box appears. This dialog box allows you to select your working directories for the Cadence Virtuoso Interface. All of these directories should have read and write permissions set. Click on the Help button in the dialog box to open help which explains the various fields.



Check License

You may check the status of your Sonnet license for the interface by selecting the *Help* ⇒ *Support* ⇒ *Check License* command from the main menu of the Sonnet Cadence Virtuoso Interface window. When you select this command, the license status is output to the CIW window. You may also use the command *Sonnet* ⇒ *Support* ⇒ *Check License* in the CIW window.

Customizing the Virtuoso Environment

Sonnet supplies a list of environmental variables for the Cadence Virtuoso Interface with your installation which you may use to customize your Virtuoso environment. The file “sonnet.cdsenv” is included in your release at the following location:

< Sonnet Directory>/sonnet_virtuoso_dk/cdsenv/sonnet.cdsenv

where < Sonnet Directory > is the directory in which your Sonnet software was installed. You edit the contents of this file; for details on the variables and settings please refer to the readme file.

< Sonnet Directory>/sonnet_virtuoso_dk/cdsenv/sonnet_cdsenv_readme.txt

Local Variables

If you wish to use the environmental variables only for yourself, you will copy the file “sonnet.cdsenv” into your $HOME directory.

Global Variables

If you wish to have the variables affect the settings of everyone using the Cadence Virtuoso Interface, do the following:

1 **At the command prompt, enter:**

cd < Cadence Directory>/tools/dfII/etc/Tools

where < Cadence Directory > is the directory in which the Cadence Virtuoso software is installed.

2 **If the directory “sonnet” does not yet exist, create it by entering:**

mkdir sonnet

3 **Copy the edited version of “sonnet.cdsenv” into the “sonnet” directory as .cdsenv.**

Alternately, you can edit the setup.loc file in < Cadence Directory >/share/cdssetup to include < Sonnet Directory /data where < Sonnet Directory > is the directory in which Sonnet is installed. An example of this is included in < Sonnet Directory >/sonnet_virtuoso_dk/cdsenv/setup.loc.

Using the Cadence Setup Search File Mechanism

You may also use the Cadence Setup Search File mechanism to load your environmental variable file. Cadence software searches for environmental variables files (.cdsenv) in a particular search order; the search order is provided in the setup.loc file. The values from each file that is read overwrite those from the files read previously. You can control the search order with the CDS_LOAD_ENV shell environment variable. Setting this value to CSF uses the Cadence Setup Search File mechanism to find the .cdsenv files to load. For more details on the CDS_LOAD_ENV variable or the Cadence Setup Search File mechanism, please refer to your Cadence documentation.

To use your Sonnet file, do the following:

1 **Type "**CDS_LOAD_ENV=CSF; export CDS_LOAD_ENV**" at the command prompt.**

Setting this variable to CSF will cause the Cadence software to use the Cadence Setup Search File mechanism to find the .cdsenv files to load.

2 **Copy the sonnet.cdsenv file from the Sonnet Directory to your $HOME directory under the file name sonnet.cdsenv.**

This uses the Sonnet environmental variables no matter which Cadence version you invoke. You may also edit the setup.loc file.

Unlocking Sonnet Cadence Virtuoso Interface

When you are running a simulation using the *Analysis* ⇒ *Simulate* command, the Sonnet Cadence Virtuoso Interface is locked until the analysis is complete. You may not make any changes or execute any commands in the interface until it is unlocked when Sonnet is closed.

If for some reason the solver is interrupted in such a way as to prevent a graceful shutdown, such as a hung program or a program crash, then the Sonnet Cadence Virtuoso interface is not unlocked. In this case, you may manually unlock it by selecting *Analysis* ⇒ *Unlock Project* from the main menu of the Sonnet Cadence Virtuoso Interface window.

Appendix I Via Simplification

Introduction

Several manufacturing processes used to produce RF circuits utilize via arrays or bar via groups to provide the trace metal layer to layer connections. Both of these types of vias present an analysis challenge which drives the Sonnet model memory and analysis time requirements beyond what is practical to analyze. Via simplification provides an approach to via arrays and bar vias that reduce the time and memory requirements without sacrificing accuracy. These two processes are discussed in this appendix.

Via Array Simplification

For via arrays, the small size of the individual vias and the large number in the array usually drive Sonnet model memory and analysis time requirements beyond what is practical to analyze. This often requires that you simplify the via geometry detail before performing your EM simulation. In the past, via array simplification would need to be done manually by deleting vias and replacing with a single, larger via polygon.

The Simplify Via Array feature automatically performs this simplification during the translation process. It can be invoked inside the Cadence Virtuoso Interface. It may also be invoked in Sonnet's project editor when performing an import using the GDSII Translator or DXF Translator.

NOTE: **Via array simplification is only used when running an EM simulation using the Accurate Solver.**

Via Array Criteria

There are six criteria, all of which must be met, before a group of vias in the original geometry is considered an array and therefore simplified by the software.

Number of Vias: There must be a minimum number of vias in order for them to be considered an array.

Via Size: The vias must be the same size or nearly the same size.

Via Spacing: The vias must be within a certain distance of one another.

Layer Pass Through: The vias must pass through the same layer(s).

Metal Polygons Pads: The vias must be contained within the same metal polygon at either the top or the bottom of the vias.

Material: The vias must be set to the same material, whose conductivity is the same.

Additional Simplify Via Array Options

This text entry box should only be used at the direction of a Sonnet representative.

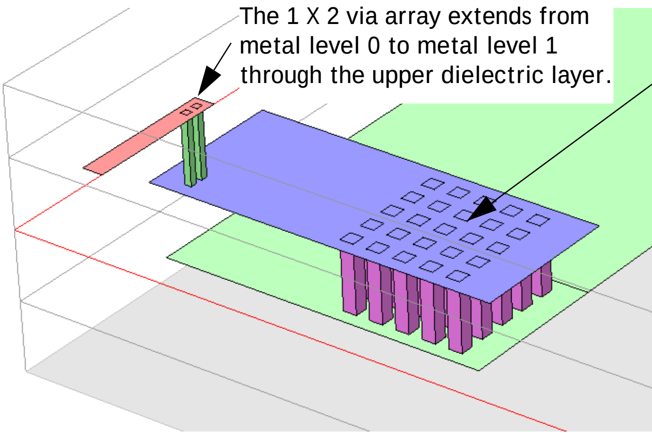
Simplify Via Array Options

There are six control options for the simplify via array feature which are discussed in detail in the following sections. We will use a simple example circuit to illustrate how the options affect the via simplification.

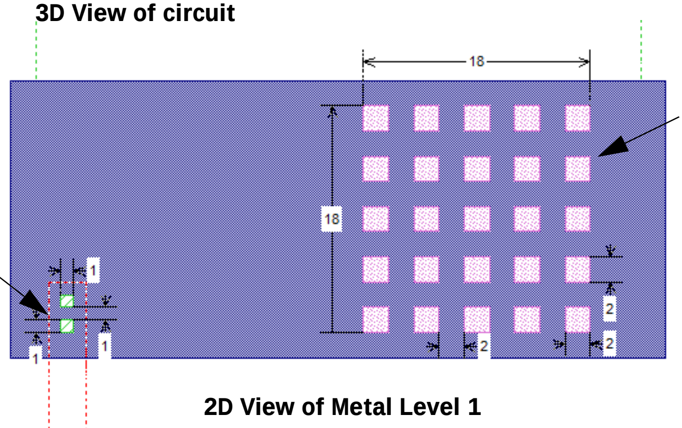
TIP

The default values were determined based on extensive testing, and in the majority of cases will provide reasonable via simplification behavior. However, Sonnet handles a wide range of layouts and processes, so these controls were provided so that a user can customize the translation if it proves necessary.

The example structure consists of three metal layers and two interconnecting via arrays, as shown below. Since the vias pass through different dielectric layers and use different metal types, the 1 X 2 array and the 5 X 5 array would never be grouped together. Either reason would be sufficient on its own to prevent these two arrays being grouped together.



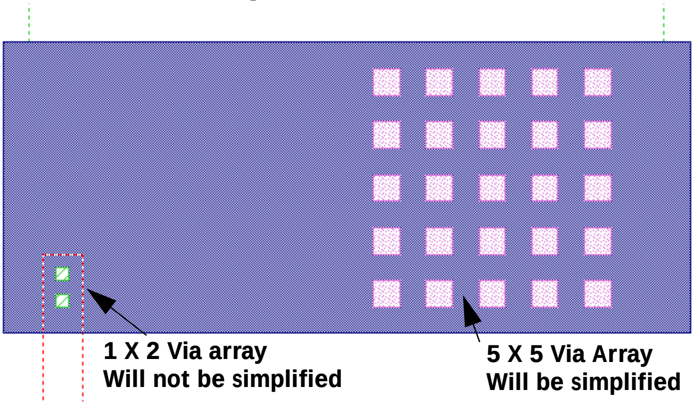
3D View of circuit



2D View of Metal Level 1

Minimum Vias in Array

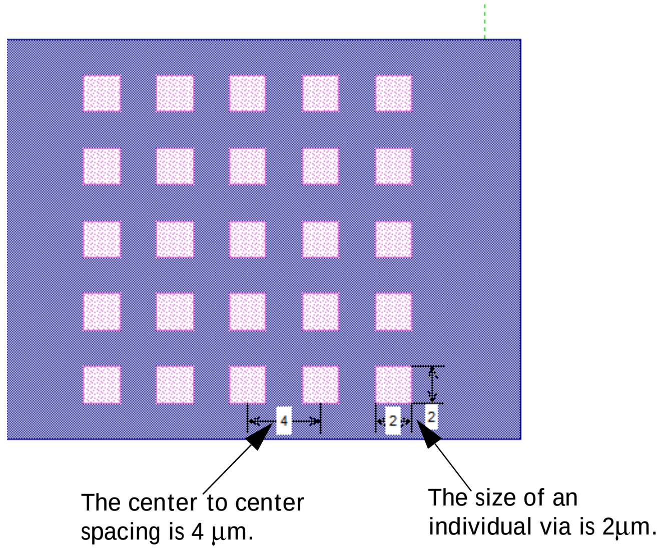
This control defines the minimum number of vias which can be considered part of the same array. The default value for this setting is 5, so that only arrays with 5 or more vias will be considered for simplification. Therefore, in our example the 1 x 2 via array would not be simplified, but the 5 X 5 would be analyzed to see if this array meets the rest of the simplification criteria. You may enter any integer value to set the minimum number required.



Max Distance to Size Ratio

This control defines the maximum spacing between vias which can be considered part of the same array. While the distance between vias is measured from the center lines of the individual vias, the control is a ratio of distance to via size. This distance cannot exceed the value of this ratio multiplied by the via size. The default setting is 4.0 and the larger the value, the more widespread the vias can be and still be grouped in the same array. Since individual via cross-sections can be of any shape, the square root of the via area is used as the via size.

For our example the size of the vias is 2.0 µm and the center to center spacing is 4.0 µm, as pictured below. This results in a Distance to Size ratio of 4.0/2.0 = 2.0 so that this array meets the via spacing criteria.



Maximum Size to Size Ratio

This control defines the maximum allowable difference in via size to be considered part of the same array. The default value of this ratio is 1.5 and the larger the value, the greater the difference of via size is allowed within an array. Since individual via cross-sections can be of any shape, the square root of the via area is used as the via size. For this example, the via cross-section is a 2.0 X 2.0 µm square. The area is 4 μm^2 and taking the square root, yields a via size value of 2 µm.

Since all of the vias are the same size in this array (2.0 µm) the Size to Size ratio is 1.0, so that this array meets the via size criteria.

TIP

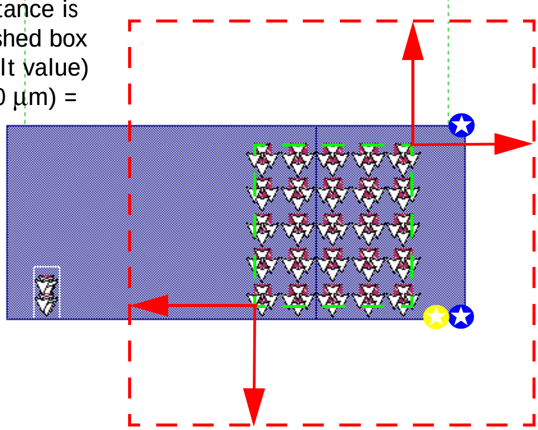
If you wish to limit your arrays to vias of the same size, set this control to 1.0.

Max Expansion Coefficient

This control helps define the size of the resulting simplified via by allowing it to be larger than the original via array perimeter (also referred to as the bounding box). The default value is 7.0, which allows the simplified via to expand outward by a factor of 7 times the largest via size in the array. The advantage in expanding the simplified via is that it can often be sized to match the polygons to which the via attaches. Having the via polygon edge and pad polygon edge in alignment can significantly reduce the subsection density in the region and thereby reduce the memory requirement of the model.

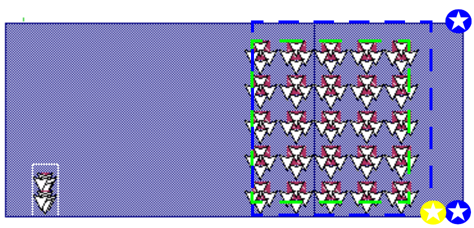
An example is shown below. The algorithm looks outward from an imaginary rectangle (bounding box) drawn around the perimeter of the array (green rectangle). The distance checked out from the perimeter is the Max Expansion Distance (shown in red arrows). It is equal to the Max Expansion Coefficient times the largest sized existing via in the array. If a vertex from a pad polygon is encountered within this window (red rectangle), the expansion stops and this sets the simplified polygon edge. If no vertices are found in a particular direction, the edge of the simplified via rolls back to the existing via array perimeter. Please note that all metal levels are examined when looking for vertices within the maximum expansion distance.

The Max Expansion Distance is denoted by the red dashed box and is 7.0 times (default value) the largest via size (2.0 µm) = 14.0 µm.



The blue stars are vertices within the max expansion distance on this metal level. The yellow star indicates a vertex found on another metal level.

The via array perimeter is indicated by the dashed green box. If no vertices are found in that direction, then the perimeter defines the edge of the simplified via in that direction.



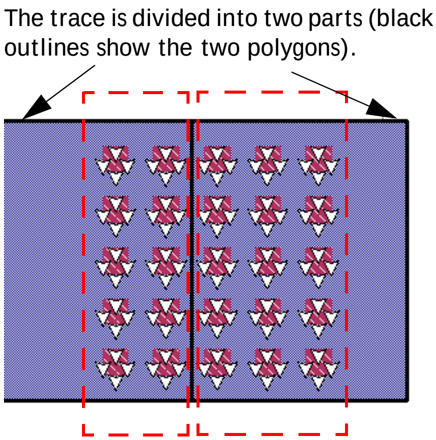
The dashed blue box above shows the final size of the simplified via. On the top, bottom and right hand side the via extends to the vertices and on the left side it conforms to the array bounding box (dashed green box) since no vertices were found within the maximum expansion distance in that direction.

TIP

The default setting for Max Expansion Coefficient is 7.0. If you wish the simplified via to be the bounding box around the array, set the Max Expansion Coefficient to 0, which allows no expansion.

Merge Planar Polygons During Simplification

In order to be considered an array, a group of vias must connect to a single polygon on the top and bottom of the group. This control allows the user to merge the polygon pads or traces prior to simplifying the vias. This results in larger arrays being recognized leading to the least number of simplified vias, thereby producing the most efficient model. This option is enabled by default. The option is illustrated below using the example circuit.



If the Merge Planar Polygon option is not selected, then the array is treated as two via arrays (indicated by the dashed red boxes) and two simplified vias are created. If the option is on, however, the two parts of the trace are merged into one polygon and the whole array is simplified into one via.

NOTE: The polygons are only temporarily merged for via simplification and will not be merged in the resulting Sonnet project.

Number of New Via Metals Created

Via metal types using the Array loss model are created to model your simplified vias in Sonnet. This setting controls how many via metal types are created in your translated project in order to model your simplified array. The fill factor is used to determine how many metal types are created. For more information on the fill factor and the Array loss model, please see "Array Loss Model" in the **Native Environment User's Guide**.

There are three choices:

Minimum: Creates the least number of via metal types but may be less accurate as a wider range of fill factors will be grouped together.

Automatic: Creates the via metal types based on an algorithm that balances the trade off between accuracy and the number of via metal types produced. This is the default setting.

Maximum: Creates the highest number of via metal types providing the most accurate answer since a via metal type is created for each unique fill factor.

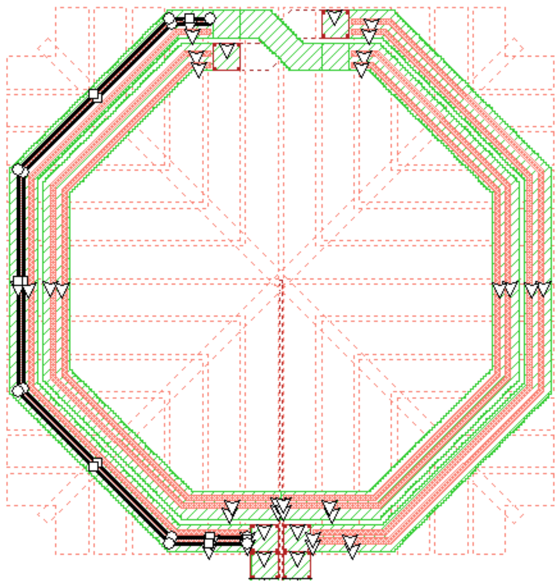
Simplified Via Array Loss

When an array is converted to a simplified via, a via metal type using the Array Loss model is created, if it does not yet exist. For example, you are translating an array which uses "ViaMetal1" for an array. During the translation a via metal type "ViaMetal1" is created that uses the Array Loss model. During translation, one source metal may be translated into two metal types if there is a significant difference in individual via size in different arrays using the same metal. For more information about via metal types and defining their loss, please see "Array Loss Model," in the **Native Environment User's Guide**.

The simplified via is modeled using the same meshing fill as the vias in the original via array. For a detailed discussion of meshing fill for via polygons, please see "Meshing Fill for Vias" in the **Native Environment User's Guide**.

Bar Via Group Simplification

Bar vias are vias whose length is significantly longer than their width. They are typically used in stacked multi-level conductors where vias carry horizontal currents. An example is shown below, with one of the bar vias highlighted in black. For vias whose aspect ratio is smaller, such as 1:1, the Sonnet model assumes that there is little to no horizontal current in the via. The assumption is that current flows in the vertical or z-direction. For bar vias, whose aspect ratio is larger, the current flow in the horizontal direction is more significant. Bar via groups placed on metal traces also drive a very fine resolution in the meshing that can require an inordinate amount of processing resources.



The Identify Bar Vias feature identifies bar vias in the translated circuit based on length to width ratio entered by the user. These vias are assigned the Bar via meshing fill, then during the analysis multiple adjacent bar vias are identified as a bar via group and merged into one wider via during the subsectioning by the analysis engine, ***em***.

Since via arrays are simplified during translation, their appearance in the project editor is that of the simplified via polygon into which the via array was converted. However, in the case of bar vias, the actual via polygon input by the user is displayed in the project editor, because no simplification has yet been done. If you wish to see the actual metal for the simplified bar vias that are used in the simulation, you should view the subsections for the circuit. For a detailed explanation of how bar vias are simplified and modeled by the analysis engine, please see "Simplifying Bar Vias in the Analysis Engine," page 286 in the **Native Environment User's Guide**.

NOTE: **The Simplify Via Arrays feature must be on in order to use the Identify Bar Vias feature.**

This feature can be invoked inside the Cadence Virtuoso Interface. It may also be invoked in Sonnet's project editor when performing an import using the GDSII Translator or DXF Translator.

NOTE: **Identify Bar Vias is only used when running an EM simulation using the Accurate Solver.**

New Via Metals and Bar Via Loss

Metal used for bar vias should always use the Volume loss model since this model supports horizontal current. Any vias that use the Array loss model or the Surface loss model will not be identfied as bar vias during the simplification process.

Changes in meshing fill do not change the loss associated with a via polygon, only the way in which the via is subsectioned. Loss is based on the loss model used for the metal type. For more information about via metal types and the Volume loss model, please see "Volume Loss Model" in the **Native Environment User's Guide**. Note that this description details the loss in the vertical direction. For the horizontal loss, bar vias are modeled in the same way as Thick metal. For more information on Thick metal modeling, see the "Thick Metal" chapter in the **Native Environment User's Guide.** For a detailed discussion of meshing fill for via polygons, please see "Meshing Fill for Vias," in the **Native Environment User's Guide**.

Index

V

W