Virtuoso® Layout Editor User Guide

Product Version 4.4.6
June 2000

Cadence Design Systems, Inc., 555 River Oaks Parkway, San Jose, CA 95134, USA

Contents

2
About Menus and Icons 49

3
Using Layout Editor Commands 67

16
Microwave Commands 491

17
Resolving Verification Errors 505

Preface

This document is for developers and designers of integrated circuits. It contains information about how to use the Virtuoso® layout editor and Virtuoso layout accelerator (Virtuoso XL) tools.

The Preface discusses the following:

To print this documentation, see the information about "Printing Documents" in the *Cadence Documentation User Guide*.

Related Documents

The following documents give you more information about related tools and the Cadence® SKILL language.

- For what's new, refer to the *Virtuoso Layout Editor 4.4.6 Product Notes*.
- For outstanding product change requests (PCRs), refer to the *Virtuoso Layout Editor 4.4.6 Known Problems and Solutions*.
- For information about how to install the product, refer to *Cadence Installation Guide*.
- For information about library structure, the `cds.lib` configuration file, or name mapping for data shared across multiple Cadence tools, refer to the *Cadence Application Infrastructure User Guide*.
- For information about how to access the technology file using SKILL, refer to the *Technology File and Display Resource File SKILL Reference Manual*.
- For information about SKILL functions, refer to the *Custom Layout SKILL Functions Reference Manual*.
- For information about how to perform design tasks with the Virtuoso layout accelerator, refer to the *Virtuoso Layout Accelerator User Guide*.
- For information about database SKILL functions, including the data access functions, refer to the *Design Framework II SKILL Functions Reference Manual*.

- For information about creating parameterized cells using the graphic user interface or low-level SKILL functions, refer to the *Virtuoso Parameterized Cell Reference*
- For a tutorial on creating parameterized cells using the graphic user interface, refer to the *Cell Design Tutorial*.
- For information about using relative object design (ROD) functions, refer to the *Virtuoso Relative Object Design User Guide*.
- For examples of pcells, refer to the *Sample Parameterized Cells Installation and Reference*.
- To learn about using the Virtuoso layout synthesizer (LAS), refer to the *Virtuoso Layout Synthesizer User Guide* and the *Virtuoso Layout Synthesizer Tutorial.*
- To learn how to use inherited connections and net expressions with various Cadence tools in the design flow, refer to the *Inherited Connections Flow Guide*.
- To learn the connectivity and naming conventions for inherited connections and how to add and edit net expressions in a schematic or symbol cellview, refer to the *Virtuoso Schematic Composer User Guide*.
- For information about streaming mask data, refer to the *Design Data Translator's Reference*.

Typographic and Syntax Conventions

The syntax conventions used in this documentation are described below.

`literal`	Words in nonitalic monospaced type indicate text you must type exactly as it is presented. These words represent command (function or routine) or option names.
`variables`	Words in italic monospaced type indicate text you must replace with text appropriate to your system. An example is: `cd` *`your_install_dir`*`/tools/dfII/samples/local`
`z_argument...`	Words in italic monospaced type also indicate text you must replace with an appropriate argument. The prefix indicates the data type(s) the argument can accept, for example *`t_`* for text. The three dots indicate that you can repeat the argument. Substitute one or more names or values. Do not type the data type or underscore.

.*italic*	Words in italics indicate names of manuals, commands, and form buttons, form fields, and other features of the user interface (UI).
[ ]	Brackets indicate and enclose optional arguments except when they enclose keyboard bindkeys. Although this document refers to commands by full menu names, if a bindkey is available for a command, it is included in brackets after the command name. For example, *Zoom – In* [z].

1

Virtuoso Layout Editor Overview

This chapter contains these topics:

Using the Virtuoso Layout Editor

You use the Virtuoso® layout tools to prepare custom integrated circuit designs. The layout editor is the base editor in the Virtuoso set of tools.

The layout editor lets you perform the following tasks:

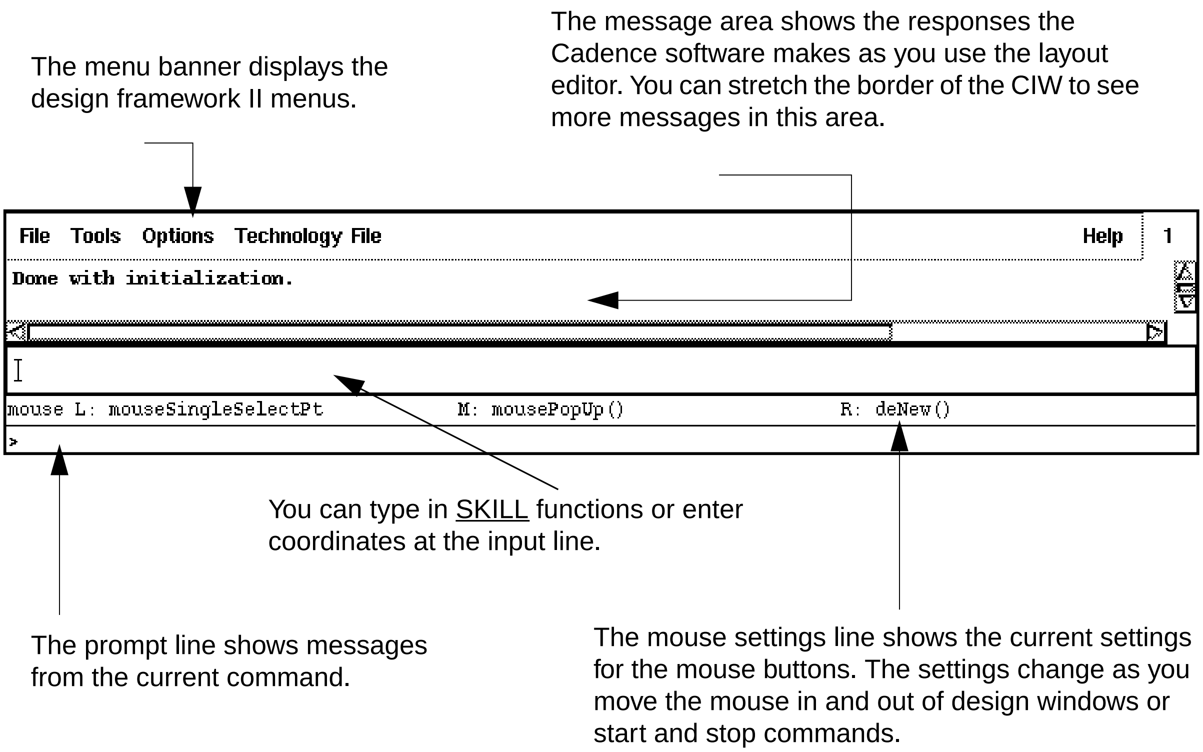
- Create and edit polygons, paths, rectangles, circles, ellipses, donuts, pins, and contacts in layout cellviews
- Place cells into other cells to create hierarchical designs
- Connect a pin or group of pins in a net internally or externally
- Automate each stage of the design task using the Virtuoso layout accelerator (Virtuoso XL)
- Create special parameterized cells (called pcells) containing data that you want to modify quickly or that you want to set with Cadence® SKILL language commands

About the Command Interpreter Window

The Command Interpreter Window (CIW) is the first window that appears when you start a Cadence design framework II (DFII) workbench. The CIW is used in these ways:

- To display menus containing commands and tools for general, non—tool- specific operations
- To enter SKILL commands

- To display messages and warnings



Starting the Layout Editor

To start the Virtuoso layout editor software, you must type the name of an executable in an xterm window. Basic layout editor workbench executable names include

- `layout` – includes the layout editor, Assura™ interactive verification products, plotting, and physical translators
- `layoutPlus` – includes all of the above, plus the Virtuoso compactor and Virtuoso XL

You may have other executables that start the layout editor, depending on what you have installed.

For example,

➤ To start the layout editor and the interactive verification products, type

```
layout &
```

➤ To start the layout editor with the compactor in the *Tools* menu, type

```
layoutPlus &
```

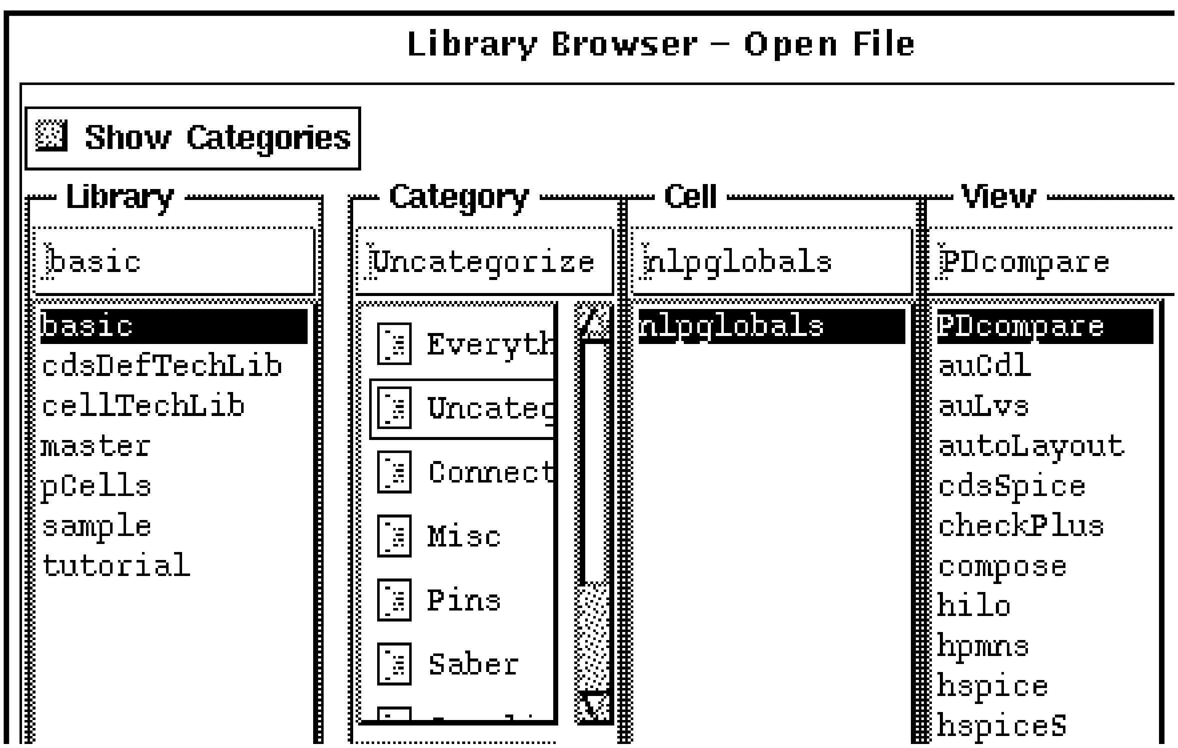
The optional ampersand (&) starts the Cadence software in the background of this window. This means you can use the xterm window for other tasks while the layout editor software is running.

After you type the appropriate command, the window displays a list of the Cadence software products it is checking out. The software is finished loading when the CIW appears.

Using the Library Browser to Open Files

You can open the Library Browser from forms that contain a *Browse* button. When you need to specify a library name, the Library Browser lets you look though the libraries specified in your `cds.lib` file and set the library, cell, and view names for the form that opened it.

➤ To open the Library Browser, click on the *Browse* button in a form.

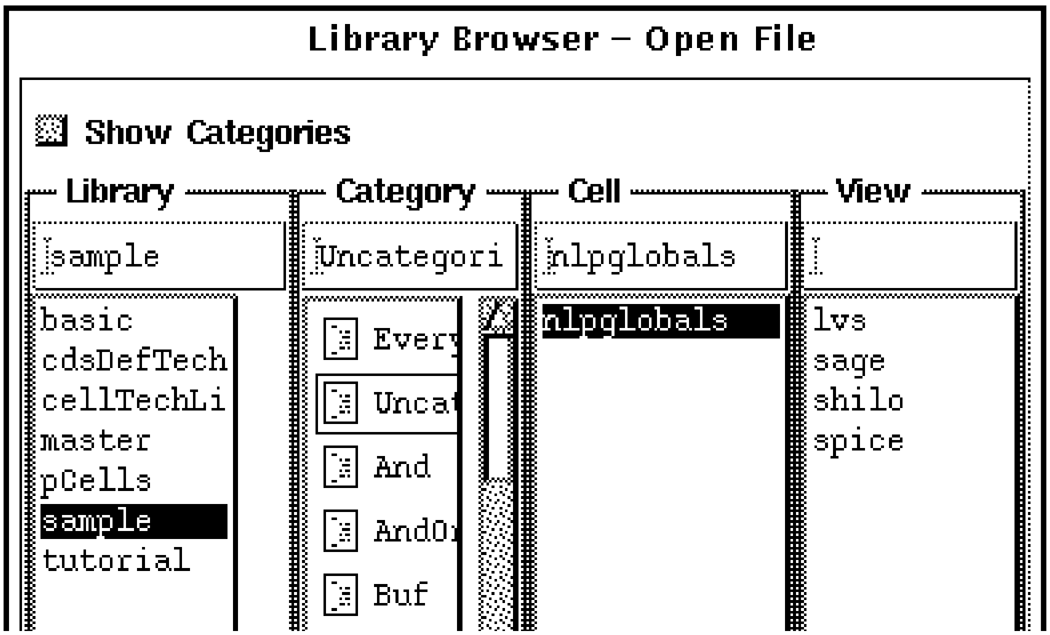


Using the Library Browser

You can use the browser to choose the library, cell, and view for the form that opened the browser.

To choose a library,

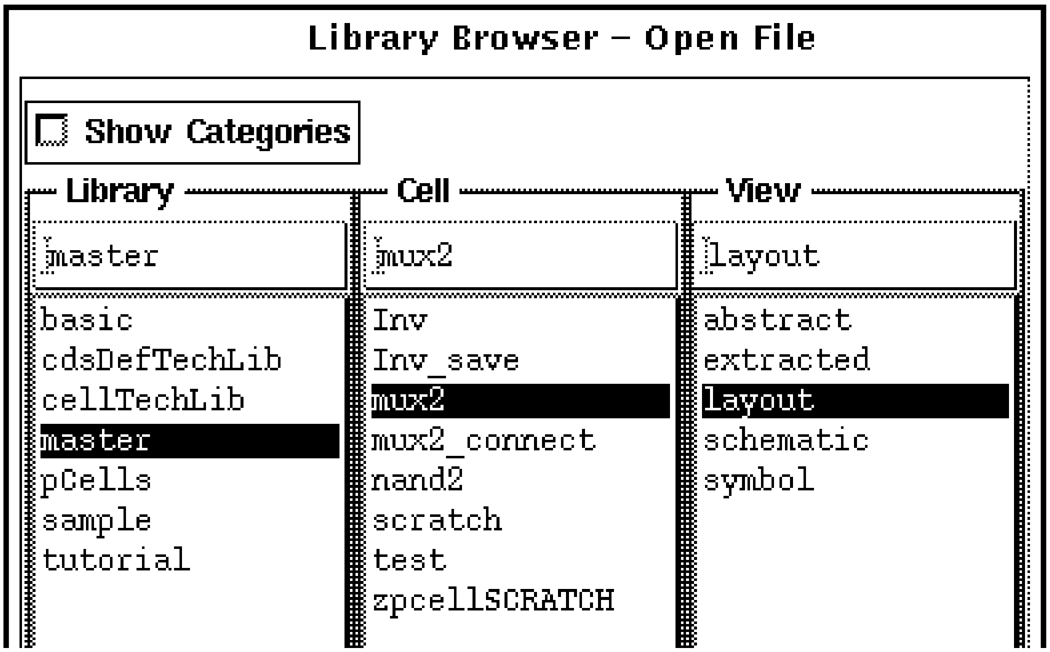
➤ In the *Library* list, click on the library you want.



The library name appears above the list, and the other lists are updated to reflect the categories, cells, and views contained in that library.

To choose a cell or view,

➤ Click on the cell or view name you want.



The cell and view names appear above the lists and in the form that opened the browser.

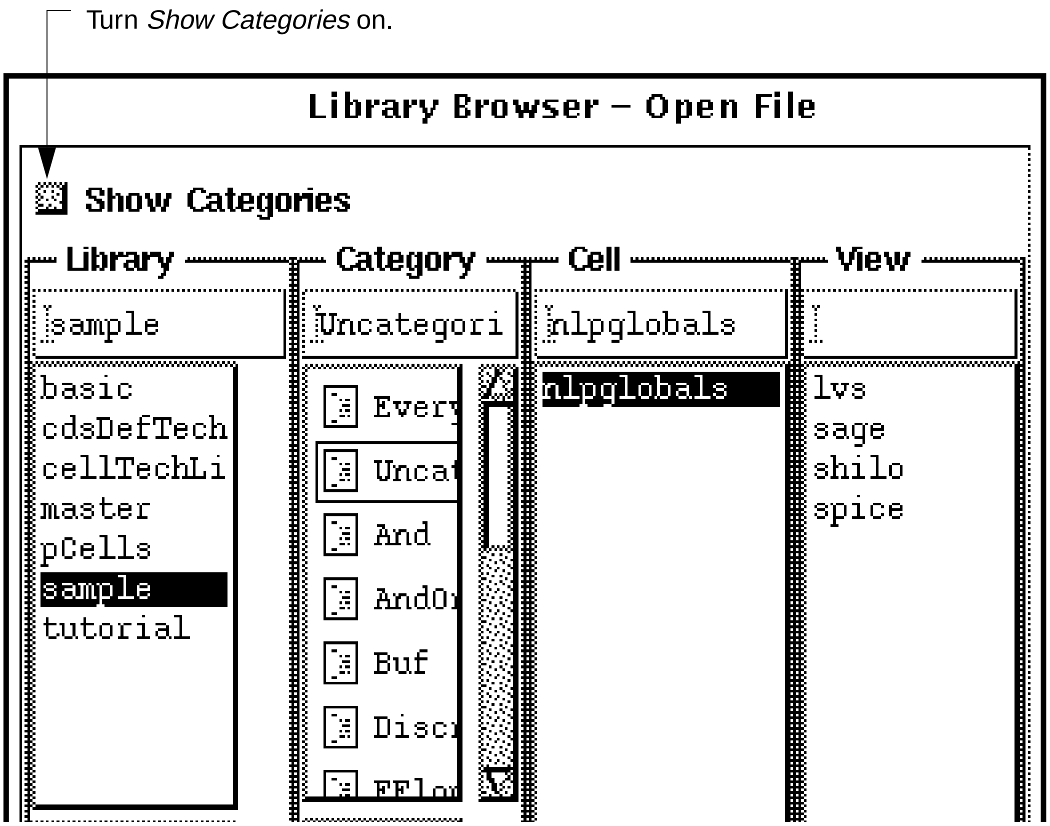
Using Categories

You can categorize cells in a library so that you do not have such a long list in the Library Browser. Categorizing cells is also a useful way to keep track of groups of cells that describe specific sections of your design. You can

- Use the Library Manager to create cell categories
- Show categories in the browser
- Remove categories in the browser

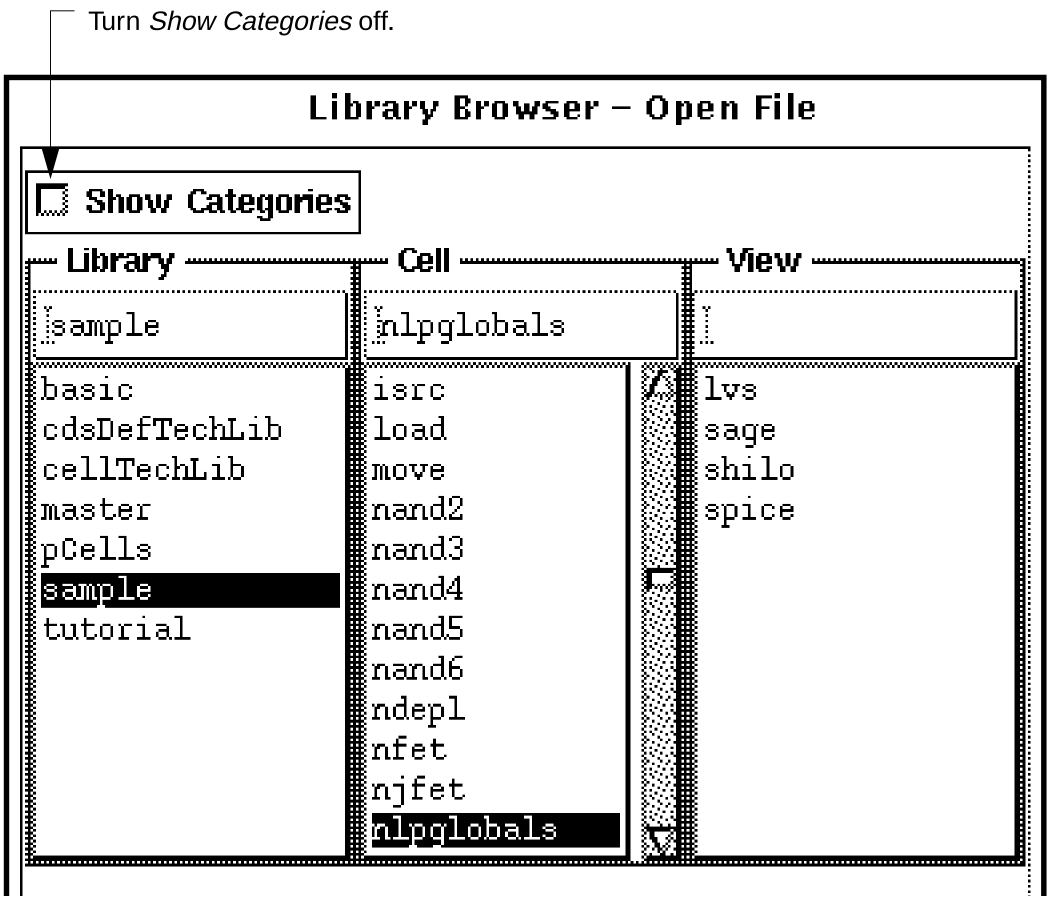
To show the cell categories,

➤ Set the *Show Categories* button on.



To remove the cell categories,

➤ Set the *Show Categories* button off.



Using Filters

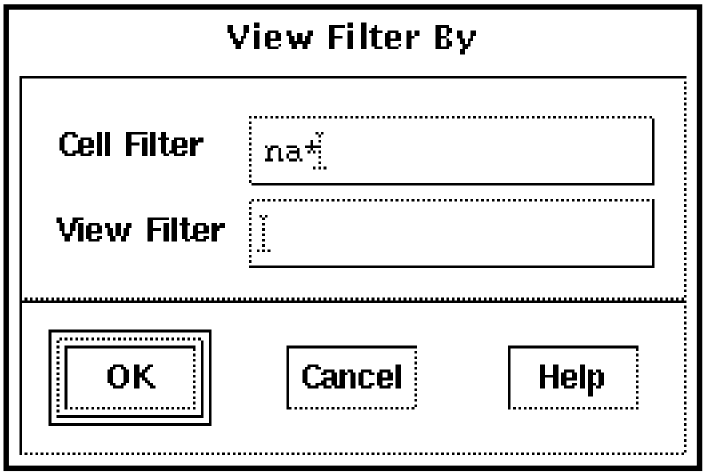
You can use categories or the View Filter By form to specify which cells and views you want to see.

➤ To filter the cell and view lists by a category, click on an entry in the *Category* list.

To filter the cell and view lists using the View Filter By form,

1. Click on the *Filters* button in the browser.

2. In the View Filter By form, type characters and wildcards in the text field and click *OK*.



The form is closed and the browser lists are filtered and updated.

About Libraries

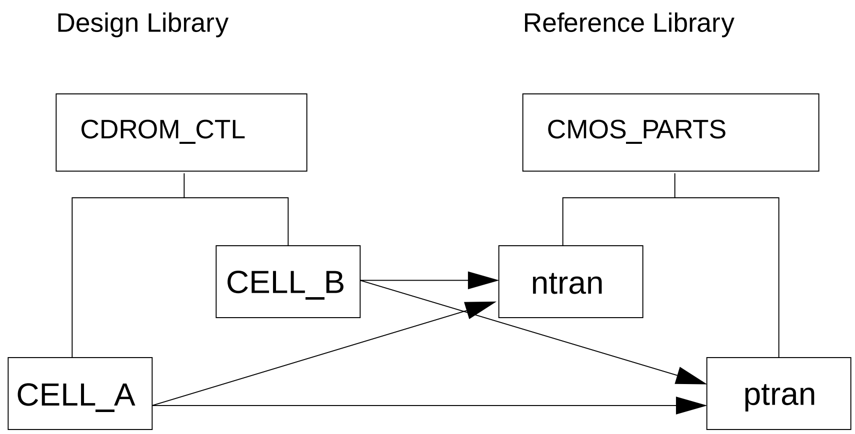
You store cells such as inverters, NAND gates, or NOR gates in a library. Each library keeps a catalog of all cellviews together with the path to the data files. The library is associated with files that define common information such as mask layer names and design rules.

There are two types of libraries: design and reference.

- A design library contains the cells and views you are working on. You and your design team might share the same working design library.
- A reference library contains well-verified cells and views with read-only privileges. Reference libraries are frequently shared within a group or company.

Using Reference Libraries

A reference library contains master cells that other designs use. These master cells are protected from editing so they remain consistent between designs.



Cells in the design library can use cells in the reference library.

To use reference libraries, you

- Create a reference library and set its file access permissions so that other design groups can read its cells
- Add the reference library path to each user's `cds.lib` file

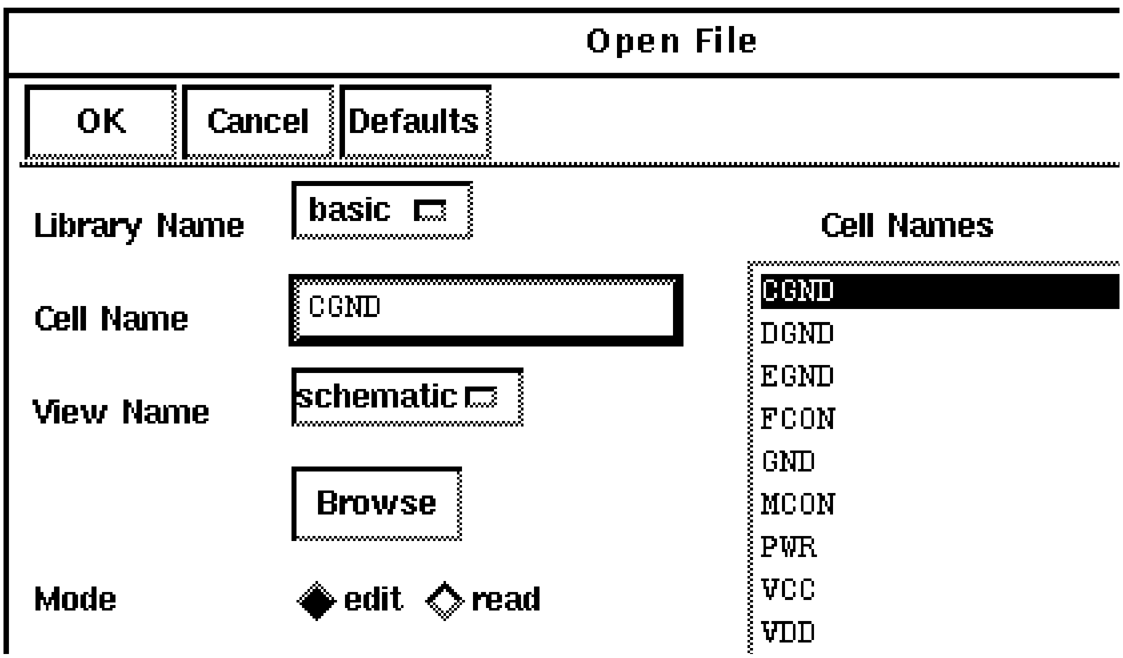
Using the Open File Form to Open Files

The Open File form lets you open a cellview in a design window. You can open a layout cellview in read-only mode without a license, but you do not have any editing capabilities without a license.

There are two ways to open cellviews.

- In the layout editor cellview window, the *Window – Open* command closes the current cellview and replaces it with a new cellview.
- In the CIW, the *File – Open* command opens another window containing the new cellview.

The Open File form is documented in the *Design Framework II Help* manual.



Opening a Cellview

To open a cellview,

1. Choose either *Design – Open* from the cellview menu banner or *File – Open* from the CIW.
2. In the Open File form, choose the library from the *Library Name* field.

   The *Cell Names* list changes to reflect the cell names in the selected library.
3. Choose one of two ways to select a cell name:
   - Click on a *Cell Name* in the list
   - Type a name in the *Cell Name* field
4. In the *View Name* field, choose a view name.
5. Click *OK*.

   If you are opening a managed file, you might be asked if you want to check out the file.

Creating a New Cellview

To create a new layout cellview,

1. Choose *File – New – Cellview* from the CIW.
2. In the Create New File form, choose the library from the *Library Name* cyclic field.
3. Type a cell name in the *Cell Name* field.
4. Type `layout` in the *View Name* field.
5. To set *Tool*, do either of the following:
   - Press `Tab` to automatically choose *Virtuoso*
   - Choose *Virtuoso* from the *Tool* cyclic field
6. Click *OK*.

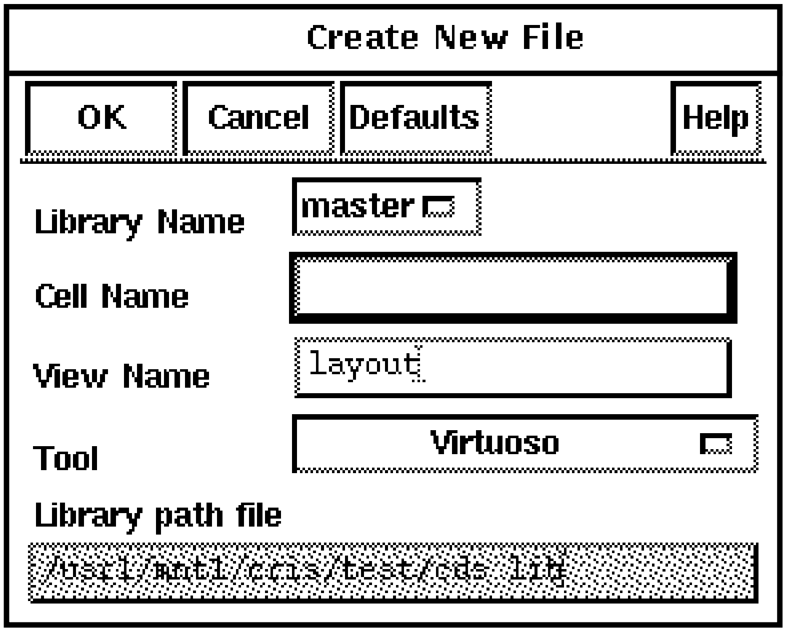
If you want to create a managed file, you need to check in the file after you close it.

You can have the *View Name* and *Tool* fields automatically update one another by setting specific environment variables. For information about these environment variables, see the `deNew` SKILL function documentation in *Design Framework II SKILL Functions Reference*.

About the Create New File Form

The Create New File form lets you create a new cellview.

The Create New File form is documented in the *Design Framework II Help* manual.

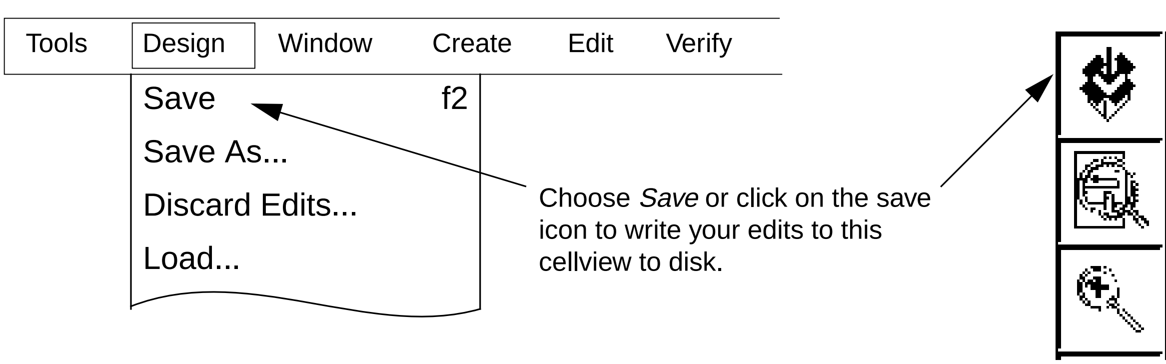


Saving a Cellview

As you edit a cellview, your data exists only in virtual memory. You need to periodically save your data to disk. The *Save* command copies the design from virtual memory to disk.

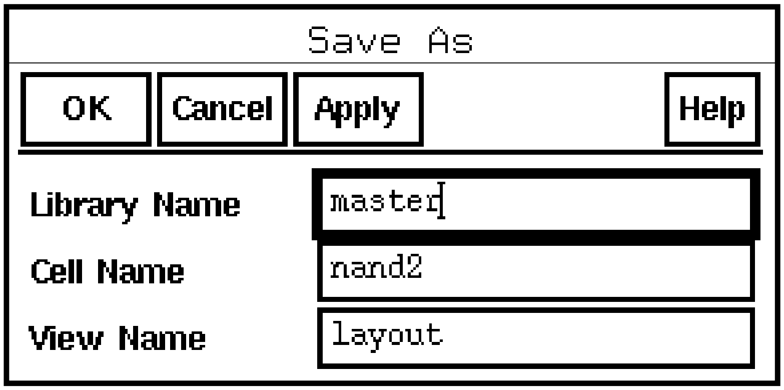
➤ To save your cellview edits to disk, do one of the following:

- Choose *Design – Save.*
- Press F2.
- Click on the save icon in the icon menu.



About the Save As Form

The *Save As* command lets you save the current cellview to disk with a new name.



Library Name sets the name of the library to which you want to save the new cellview.

Cell Name sets the name of the cell for the new cellview.

View Name sets the view name. For the layout editor, this is usually *layout*.

Saving a Cellview Under a Different Name

To save a cellview under a different name,

1. Choose *Design – Save As*.
2. In the Save As form, enter the library, cell, and view names you want to use.
3. Click *OK*.

   A copy of this cellview is created under the library, cell, and view name specified. The layout editor continues to edit in the original cellview rather than switching to the new cellview.

   Depending on how your check-in defaults are set, you might be asked if you want to check in the file. If you are not asked and you want to create a managed file, you need to check the new file in manually.

Closing a Cellview

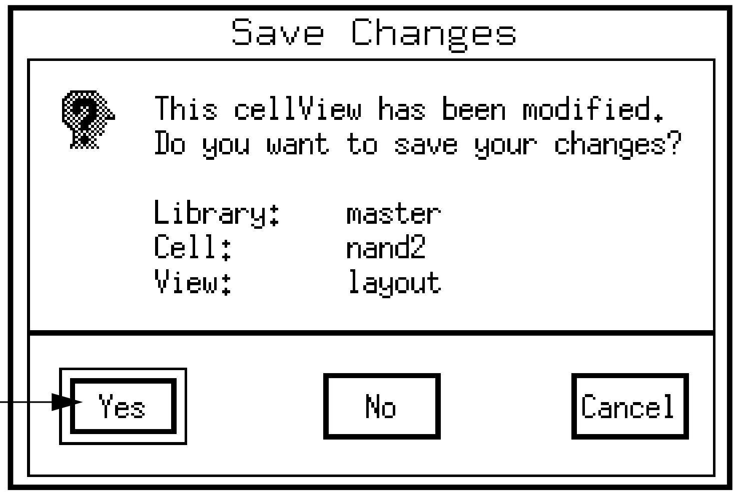
When you are finished editing a cellview, you can close it.

To close a cellview,

1. Choose *Window – Close* [`Control-w`].

   If you edited the cellview and have not yet saved it, a dialog box warns you that you have not saved your edits.

Click *Yes* to save any edits since the last *Save*.



2. To save your edits to disk, click *Yes*.

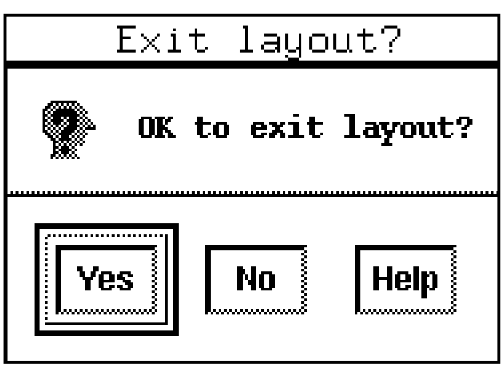
   If this is a managed file, you might be asked if you want to check in the file.

Quitting After Saving All Edits

After you have saved all your edits to cellviews and you want to quit the Cadence software,

1. Choose *File – Exit* from the CIW.

   The Exit dialog box appears.



2. Click *Yes*.

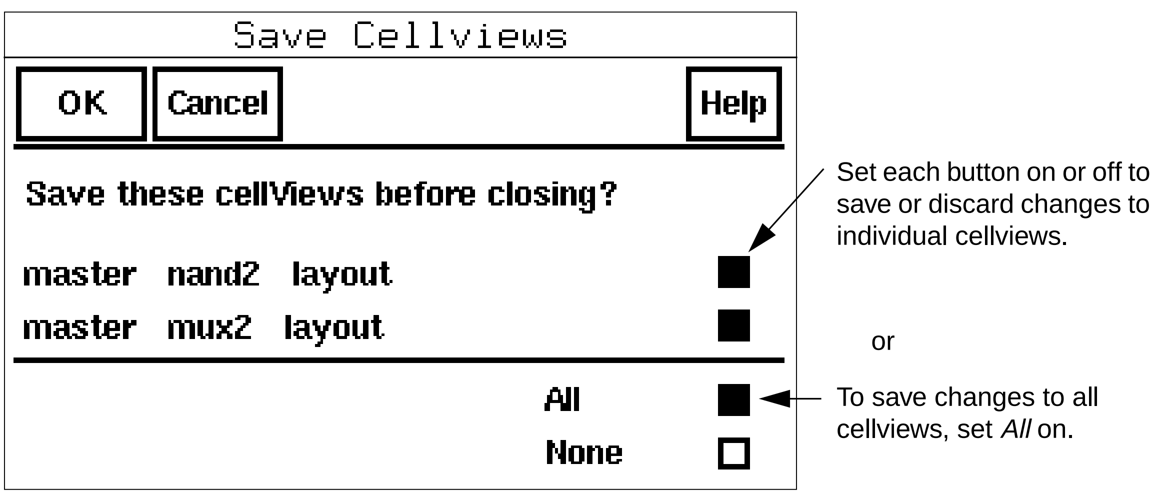
   All the Cadence software windows close.

   If there are any checked-out managed files, you might be asked if you want to check in the file.

Quitting and Saving Edits

If you have not saved all edits to cellviews before you choose the *Exit* command, you are prompted to save data as you quit.

1. Choose *File – Exit* from the CIW.
2. Click *OK* in the Exit dialog box.
3. Fill in the Save Cellviews form as follows.



4. Click *OK*.

   All the Cadence software windows close. If there are any checked-out managed files, you may be asked if you want to check in the file.

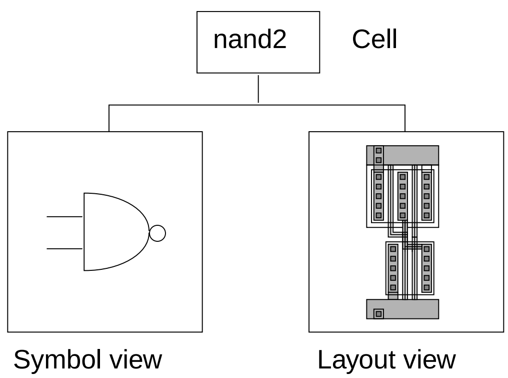
Overview of Cellviews

You create the physical layout of a chip design in a layout cellview.

A cell is the basic design object from which you build a chip or system. Each cell can include layout, schematic, and symbolic data representing a part of your chip design.

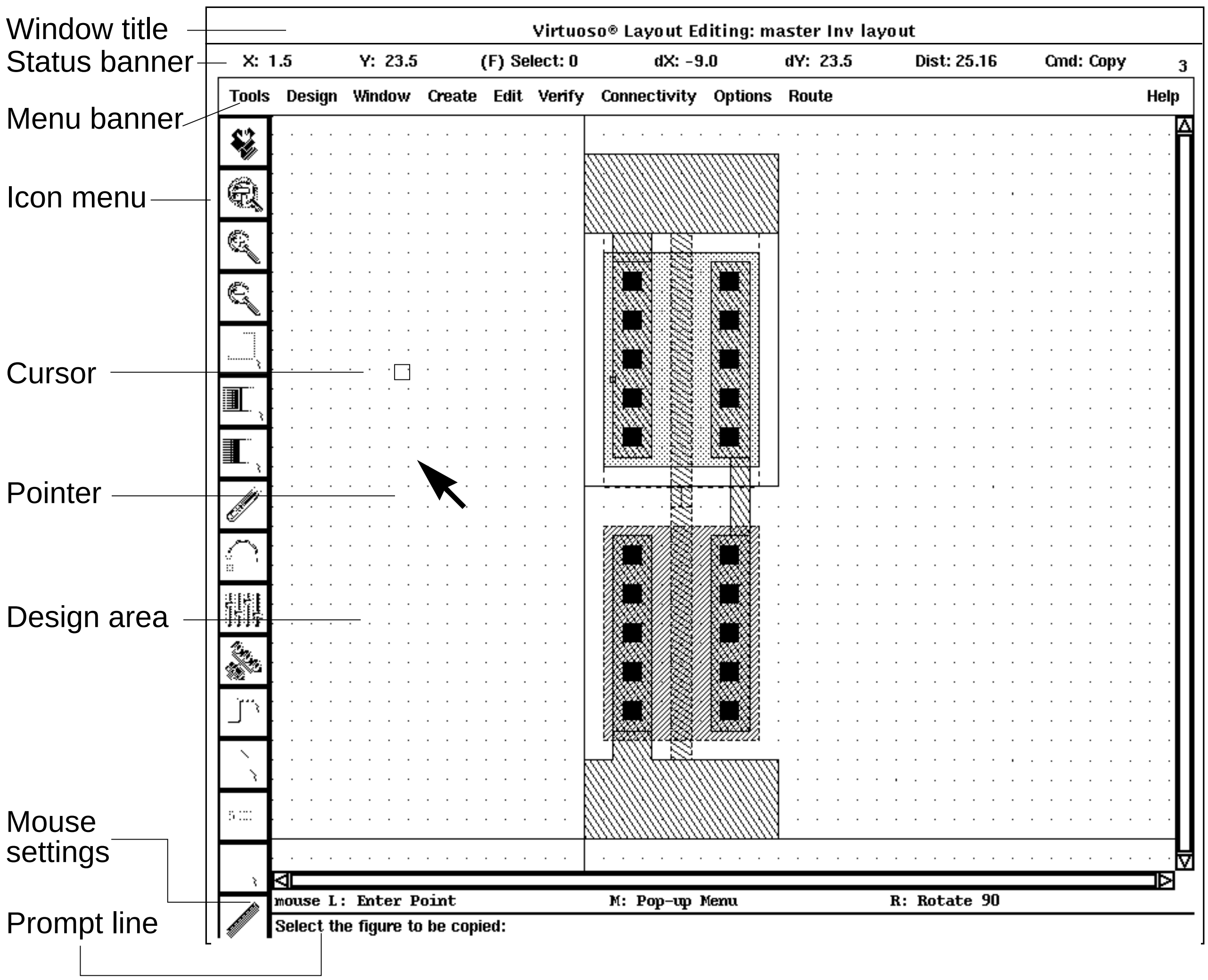
Different types of cell data are displayed in different views. For example, you can create layout, schematic, and symbolic views of an inverter. So a layout cellview is the layout view of a cell.

As shown below, you edit or create a layout of your nand2 cell with the layout editor by opening a layout cellview.



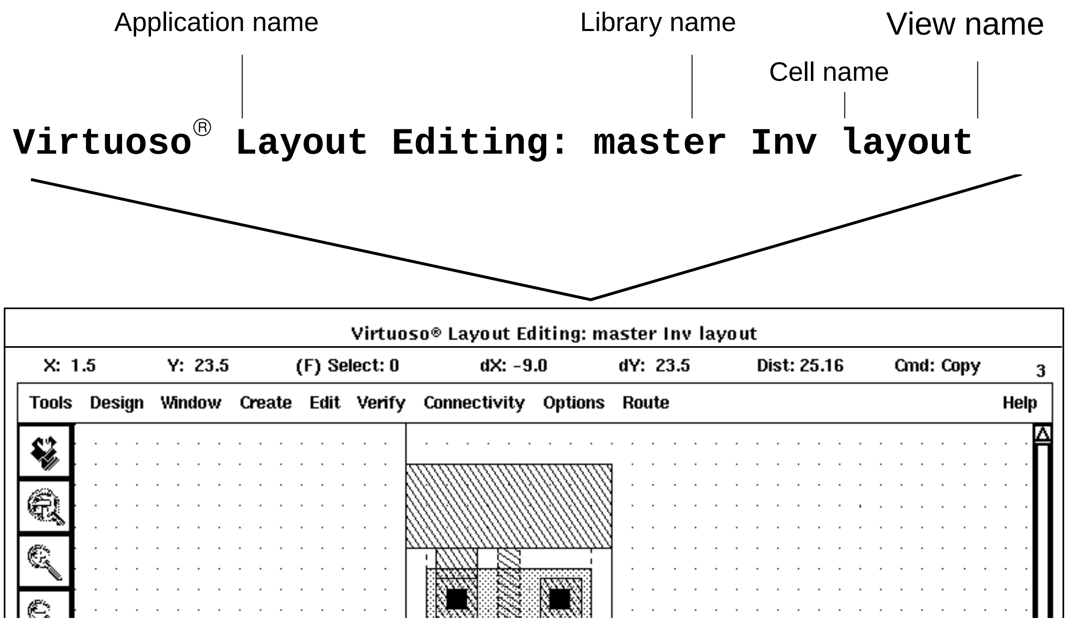
The Parts of a Virtuoso Layout Editor Design Window

The following is an example of a Virtuoso layout editor window.



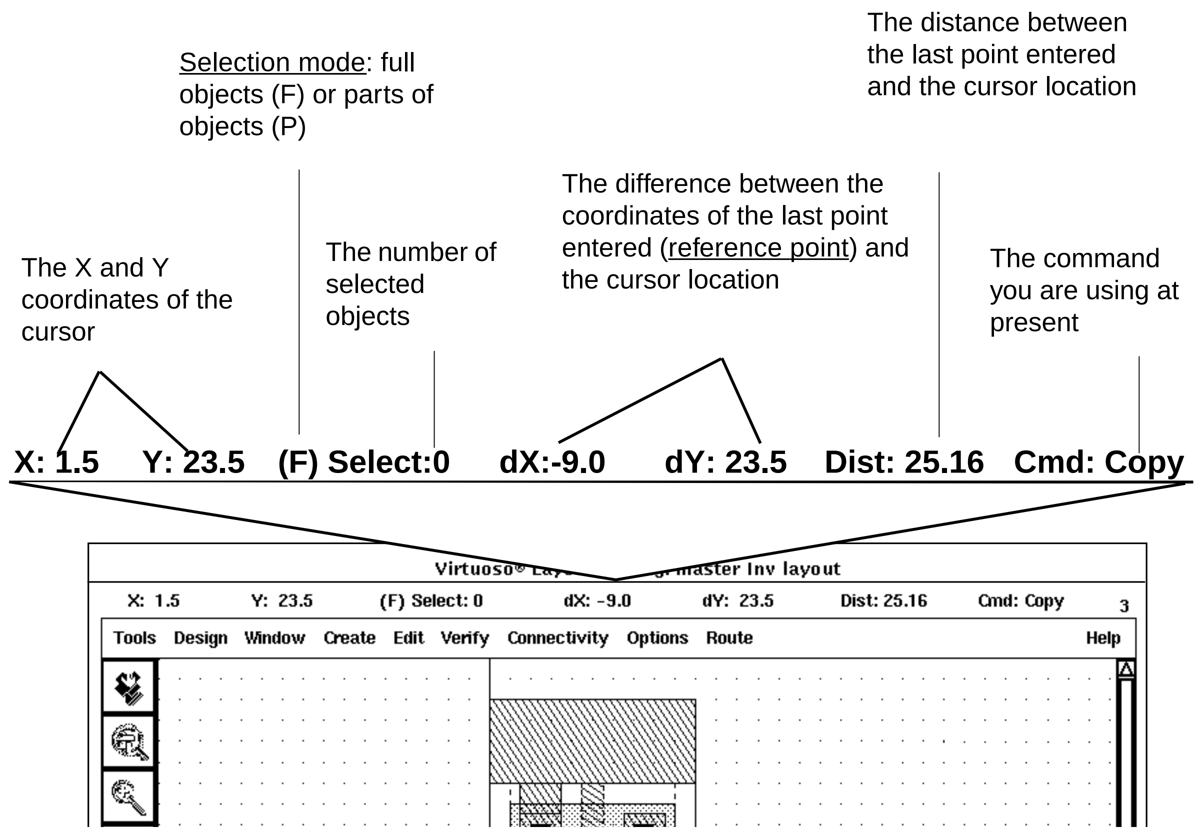
Window Title

The top of the window contains information about the cell you are viewing.



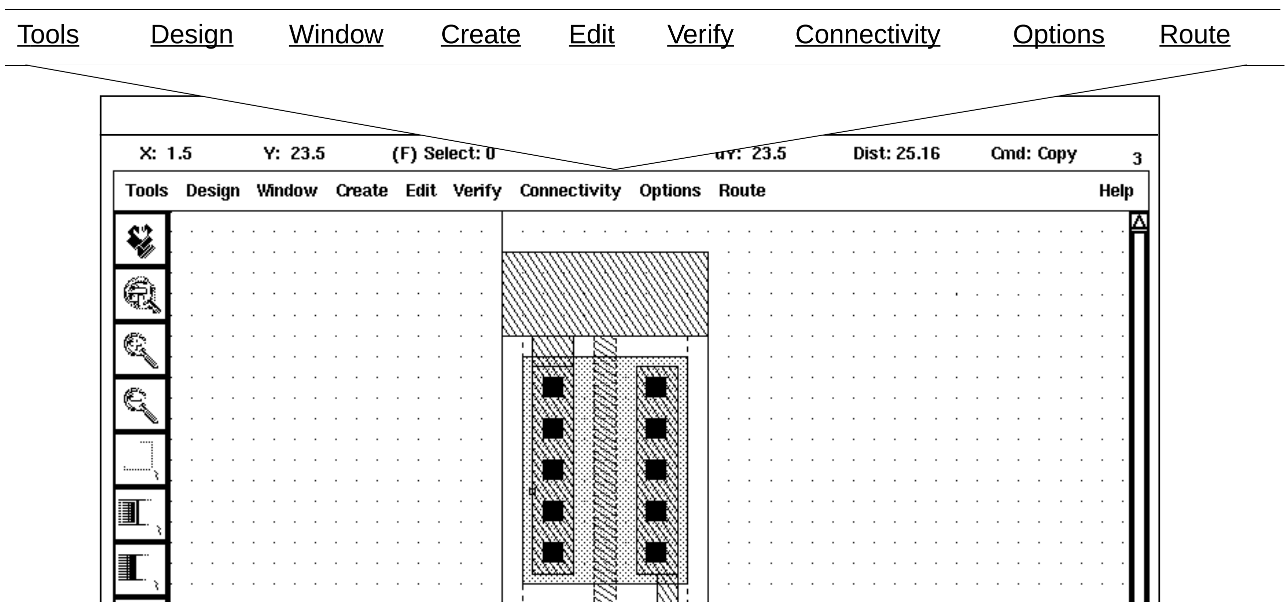
Status Banner

The status banner displays information about the cursor, selection, points, and command.



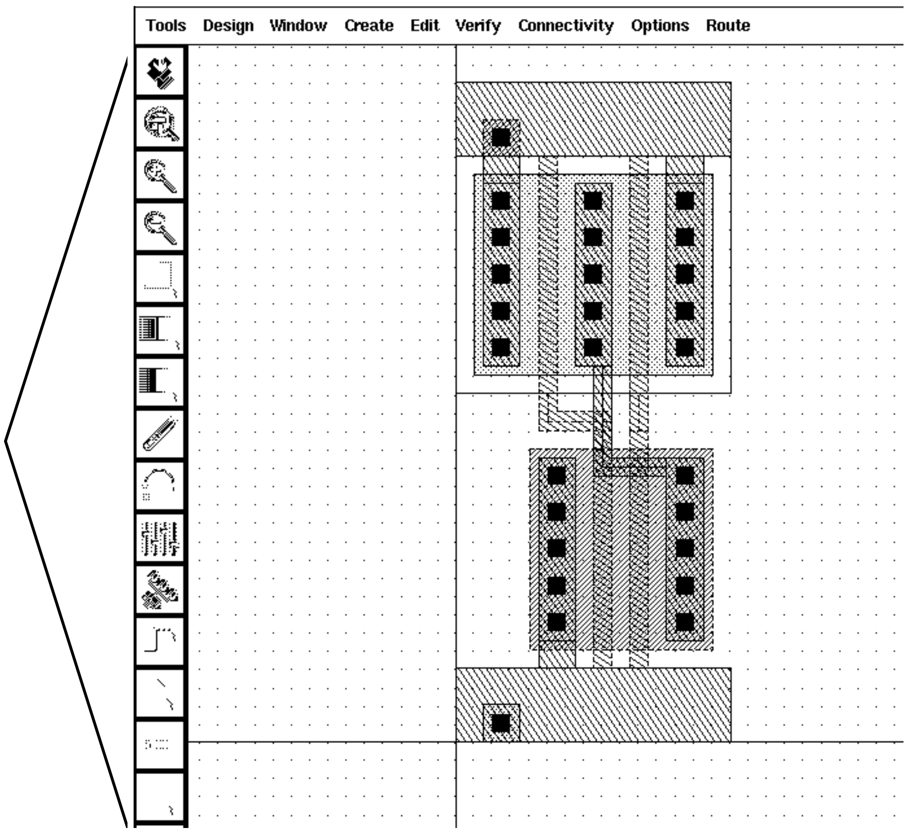
Menu Banner

The menu banner displays the layout editor menus. You can click on an item in the banner to display a menu of commands.



Icon Menu

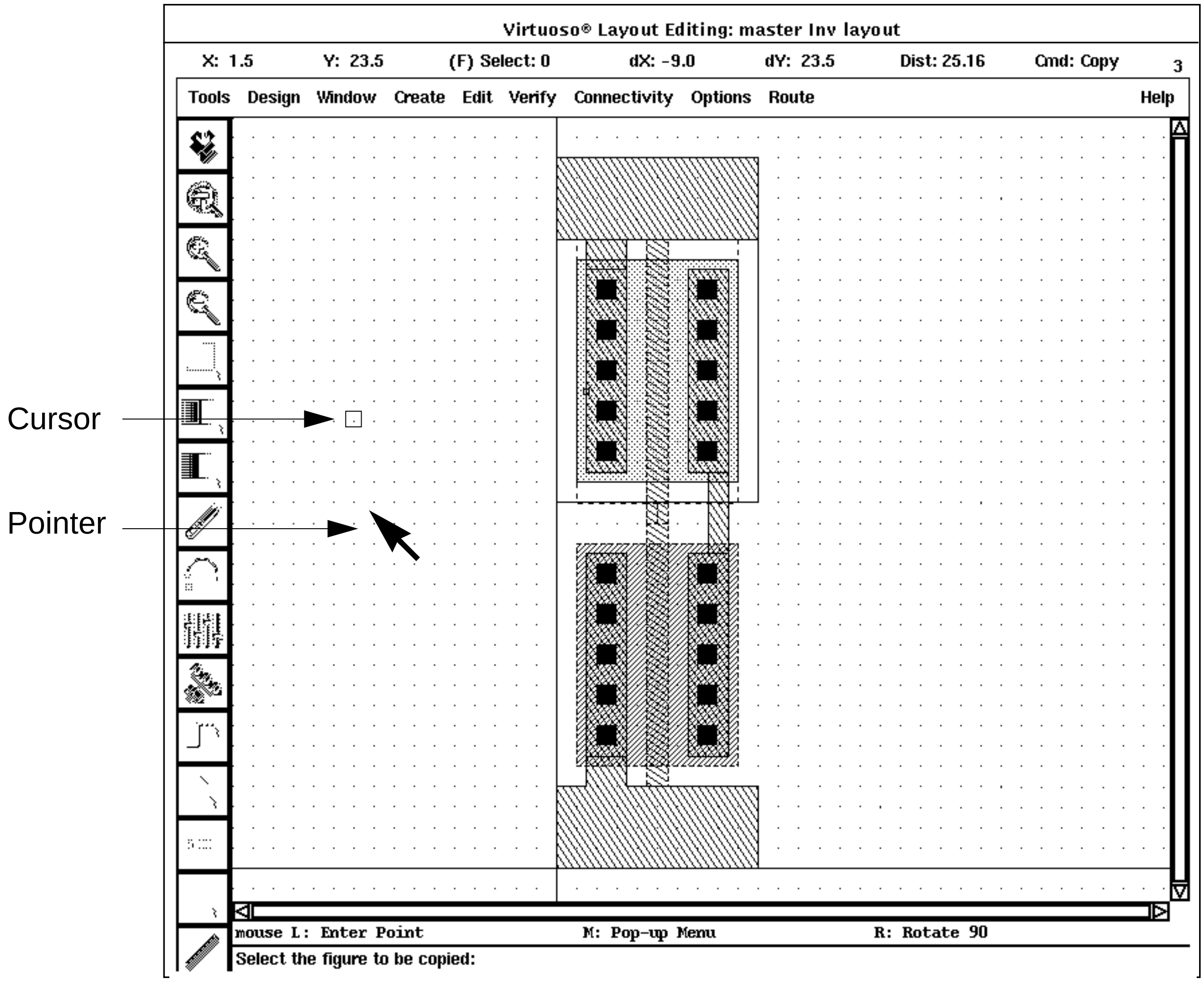
The icon menu.appears on the left side of design windows. You can start common commands quickly by clicking on an icon. To see the command name, move the cursor over the icon. In read-only mode, the editing icons are not displayed.



Icon menu

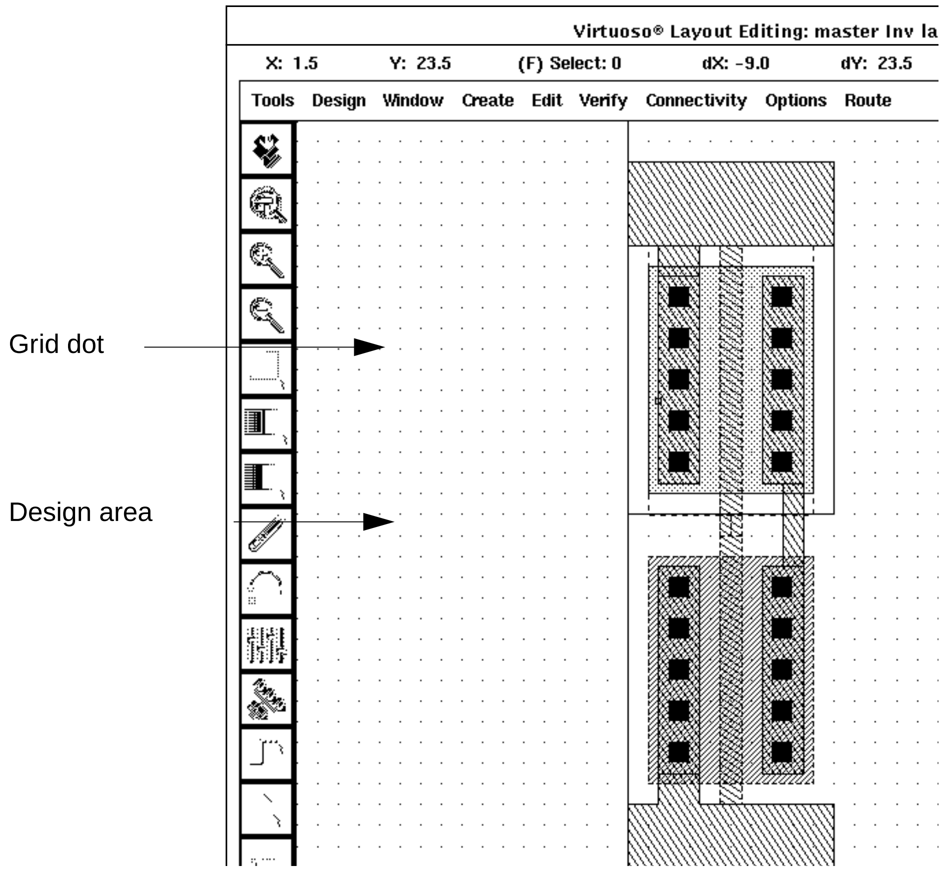
Cursor and Pointer

Use the mouse cursor to enter points or select design objects. Use the pointer to choose menu items or options in command forms.



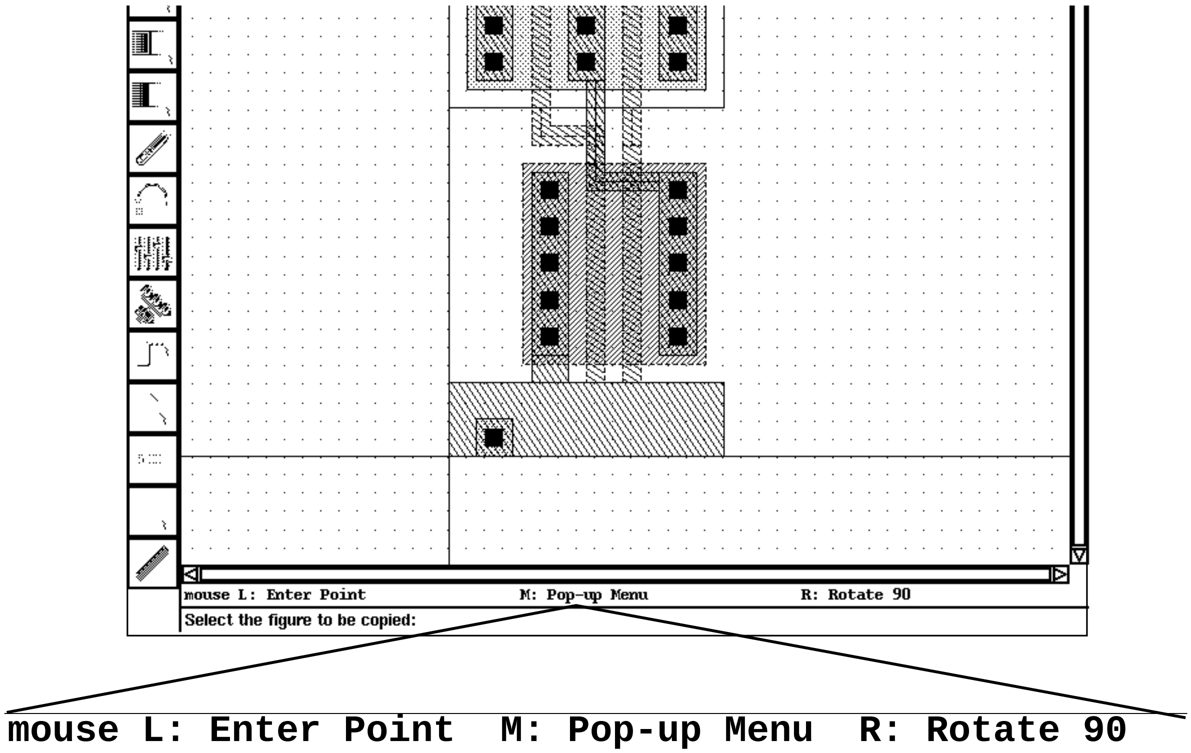
Design Area

In the design area, you create and edit objects: paths, polygons, and other shapes for your physical layout. You can turn the grid on and off in the design area. You use the grid to help create objects.



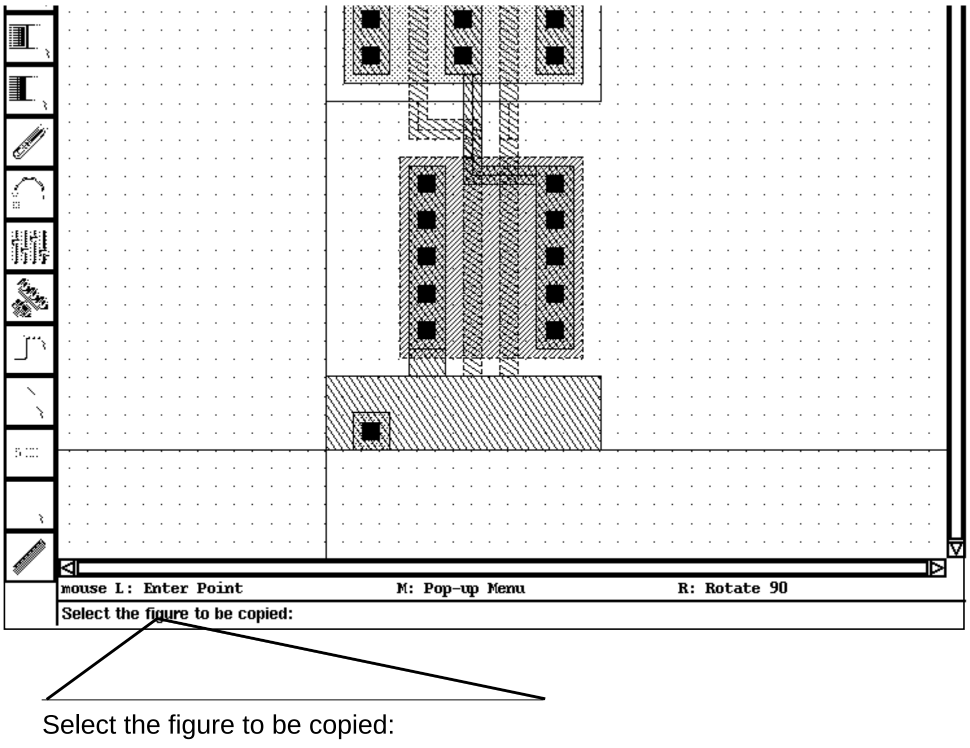
Mouse Settings

The <u>mouse settings</u> line shows what clicking the left, middle, or right mouse button does for the current window and command state.



Prompt Line

The prompt line shows instructions from the current command.



Making Cellviews Editable

To make a cellview editable,

➤ Choose *Design – Make Editable*

You can change a cell so it is read only (you cannot edit it).

To make a cellview read only,

➤ Choose *Design – Make Read Only*

When the cellview is set to read only, this command is replaced by *Design – Make Editable*, which, when chosen, returns the cellview to edit mode. *Design – Make Editable* works only if the UNIX permissions on the file are set to enable write.

2

About Menus and Icons

This chapter contains these topics:

Starting Layout Editor Commands from the Menu Banner

The banner at the top of a layout cellview shows the Virtuoso® layout editor menus.

Tools	Design	Window	Create	Edit	Verify	Connectivity	Options	Place	Route

To start a command,

1. Click on the name of the menu you want to open.
2. Move the pointer to the command you want.
3. Click.

Menu Details

- Commands followed by dots display forms
- Commands followed by a letter indicate bindey options
- Commands followed by an arrow open another menu
- Commands labeled with “Virtuoso XL” can be used only when the Virtuoso layout accelerator (Virtuoso XL) is running.

Create	
Rectangle... r	
Conics ▷	Circle
	Ellipse
	Donut
Pick from Schematic	Virtuoso XL command

When a command name on the menu is shaded instead of solid, you cannot use that command. The color of the menu commands reflects the read/write state of the cellview. For example, when the cellview is opened in read-only mode, the *Save* command is shaded.

Layout Editor Tools Menu

You use the *Tools* menu commands to start applications listed on this menu. These applications edit the type of data in the current cellview window. Some applications may contain a subset of base editor capabilities plus extra capabilities not found in the base editor.

Tools
Abstract
Analog Environment
Compactor
Dracula Interactive
Hierarchy Editor
Layout
Layout Synthesis
Layout XL
Microwave
Pcell
Simulation
Structure Compiler

Returning Layout Editor Commands to the Menu Banner

Selecting a command from the *Tools* menu may add menus to, change menus on, or remove menus from the menu banner at the top of the cellview.

To reset the banner to show just the layout editor commands, choose *Layout*.

Layout Editor Design Menu

You use the *Design* menu commands for management operations on the current cellview.

Design		
Save f2		
Save As...		
Hierarchy ▷	Descend X	
	Return B	
	Return to Level... b	
	Tree... T	
	Edit In Place x	
	Refresh	
Open...		
Discard Edits...		
Make Read Only/Make Editable		
Summary		
Properties... Q		
Plot ▷	Submit...	
	Queue Status...	
Component Types		Virtuoso XL command
Gen from Source...		Virtuoso XL command
Template ▷	Load From File...	Virtuoso XL command
	Save To File...	Virtuoso XL command

Layout Editor Window Menu

You use the *Window* menu commands to manage the current cellview and to determine how the data will be viewed.

Window	
Zoom ▷	In z
	In by 2 ^z
	To Grid ^g
	To Sel Set t
	Out by 2 Z
Pan tab	
Fit All f	
Fit Edit ^x	
Redraw ^r	
Area Display ▷	Set...
	Delete
	Delete All
Utilities ▷	Copy Window
	Previous View w
	Next View W
	Save View...
	Restore View...
Create Ruler k	
Clear All Rulers K	
Show Selected Set	
Close	

Layout Editor Create Menu

You use the *Create* menu commands to insert new objects into your design. When a command name on the menu is shaded instead of black, you cannot use that command. When the cellview is opened in read-only mode, the *Create* commands are disabled.

Create	
Rectangle r	
Polygon P	
Path p	
Label... l	
Instance... i	
Pin... ^p	
Pins From Labels...	
Contact... o	
Device...	
Conics ▷	Circle
	Ellipse
	Donut
Layer Generation...	
Multipart Path...	Virtuoso XL command
Pick from Schematic...	Virtuoso XL command
Clone...	Virtuoso XL command

Layout Editor Edit Menu

You use the *Edit* menu commands to manually change or delete objects in the current cellview. When the cellview is opened in read-only mode, most of the *Edit* commands are disabled.

Edit		
Undo u		
Redo U		
Move m		
Copy c		
Stretch s		
Reshape R		
Delete del		
Properties... q		
Search... S		
Merge M		
Select ▷	Select All ^a	
	Deselect All ^d	
Hierarchy ▷	Make Cell...	
	Flatten...	

Other ▷	Chop C	
	Modify Corner...	
	Size...	
	Split ^s	
	Attach/Detach v	
	Convert To Polygon	
	Move Origin	
	Rotate... O	
	Yank y	
	Paste Y	
	Align...	Virtuoso XL command
	Swap Components	Virtuoso XL command
	Lock/Unlock Selected	Virtuoso XL command
Transistor Chaining		Virtuoso XL command
Transistor Folding		Virtuoso XL command
Place from Schematic		Virtuoso XL command

Layout Editor Verify Menu

You use the *Verify* menu commands to check the accuracy of the physical layout of your design. When the cellview is opened in read-only mode, the *DRC* command is disabled.

Verify	
MSPS Check Pins	
DRC...	
Extract...	
ConclCe...	
ERC...	
LVS...	
Shorts...	
Probe...	
Markers ▷	Explain
	Find...
	Delete
	Delete All

Layout Editor Connectivity Menu

You use the *Connectivity* menu commands to prepare your design for routing and to display connectivity errors in your design.

Connectivity		
Define Pins ▷	Must Connect	
	Strongly Connected	
	Weakly Connected	
	Pseudo Parallel Connect	
Propagate Nets...		
Add Shape to Net...		
Delete Shape from Net...		
Mark Net		
Assign Nets		Virtuoso XL command
Show Incomplete Nets...		Virtuoso XL command
Hide Incomplete Nets...		Virtuoso XL command
XL Probe		Virtuoso XL command
Permute Pins		Virtuoso XL command
Check ▷	Shorts and Opens	Virtuoso XL command
	Against Source	Virtuoso XL command
Update ▷	Components and Nets	Virtuoso XL command
	Layout Parameters	Virtuoso XL command
	Schematic Parameters	Virtuoso XL command
	Device Correspondence	Virtuoso XL command
	Source...	Virtuoso XL command
Change Instance View...		Virtuoso XL command

Layout Editor Options Menu

You use the *Options* menu commands to control the behavior of the current window and the application you are using. The *Display* settings affect only the active window; the *Layout Editor* settings affect all the layout windows.

Options	
Display... e	
Layout Editor... E	
Layout XL...	Virtuoso XL command

Layout Editor Place Menu

You use the *Place* menu commands to automate placement.

Place		
Pin Placement...		Virtuoso XL command
Placement Style...		Virtuoso XL command
Placer...		Virtuoso XL command
Show Congestions		Virtuoso XL command
Rules ▷	Open Rules	Virtuoso XL command
	New Rules	Virtuoso XL command

Layout Editor Route Menu

You use the *Route* menu commands to interact with the router.

Route		
Export to Router...		
Import from Router...		
Rules ▷	Open Rules	
	New Rules	
Connect to Router		Virtuoso XL command
Perform Route		Virtuoso XL command

Layout Editor Microwave Menu

To activate microwave commands,

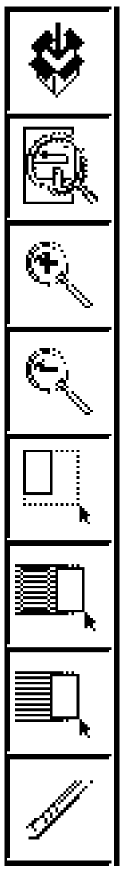
➤ Choose *Tools – Microwave*.

The *Create* menu is updated to include the microwave commands *Trl, Bend,* and *Taper*.

Create
Trl
Bend
Taper

Using the Icon Menu

The icon menu appears on the left side of the design window. You can start frequently used commands by clicking on the appropriate icon.



Save saves this cellview.

Fit Edit fills the display with the cell you are editing.

Zoom In By 2 enlarges the image in the window by 2.

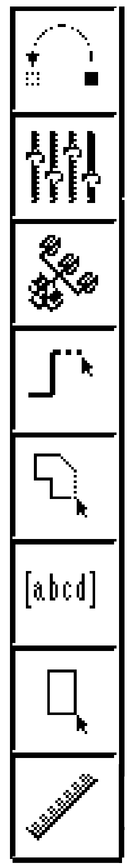
Zoom Out By 2 reduces the image in the window by 2.

Stretch stretches object edges or corners.

Copy duplicates objects in this cellview.

Move moves objects in this cellview.

Delete removes objects from this cellview.



Undo cancels the last edit.

Properties edits features of objects.

Instance places an instance of a cell.

Path creates paths (routing).

Polygon creates polygons.

Label adds text to the cellview.

Rectangle creates rectangles.

Ruler creates rulers.

Starting Icon Commands

To start a command from the icon menu,

➤ Click on the icon.

The command starts immediately.

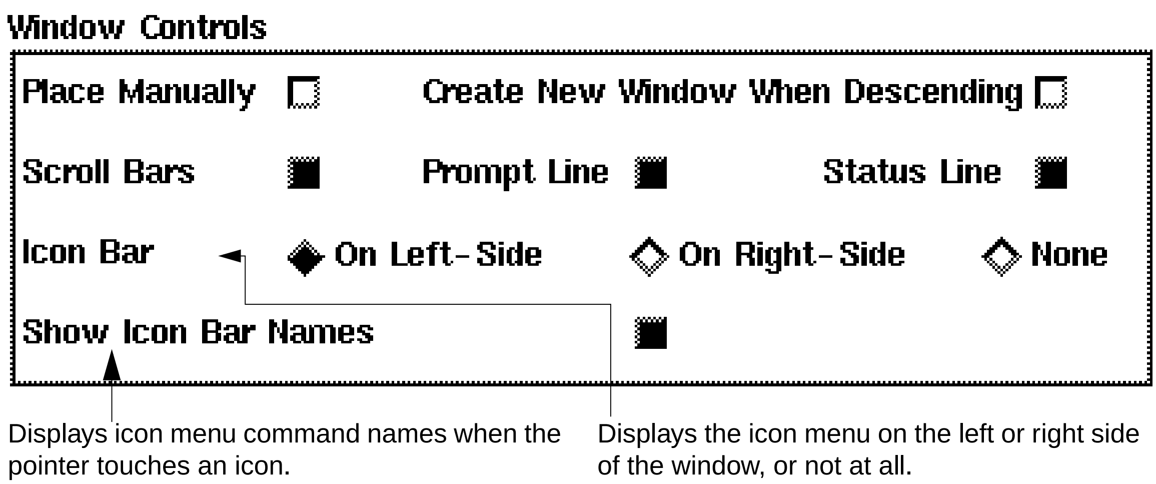
Controlling the Icon Menu's Appearance and Location

You can control

- Where the icon menu appears
- Whether the menu appears at all
- Whether icon names appear

To change the icon menu,

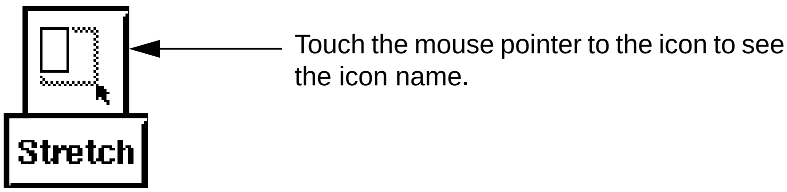
1. Choose *Options – User Preferences* in the Command Interpreter Window (CIW).
2. In the User Preferences form, turn on the window controls shown below.



3. Click *OK*.
4. To see the results of the change, do one of the following,
   - Choose *Window – Utilities – Copy Window* to open a copy of your current window.
   - Close and then reopen the window.

Viewing the Icon Command Names

When the mouse pointer touches the icon, the name of the command appears just below the icon.



3

Using Layout Editor Commands

This chapter contains these topics:

Command Functions

Starting Commands

To start Virtuoso® layout editor commands you can

- Choose a command from a layout editor menu
- Click on an icon
- Move the pointer into the cellview, and press a bindkey
- Choose a command from the layout editor pop-up menu
- Draw a stroke in the design window

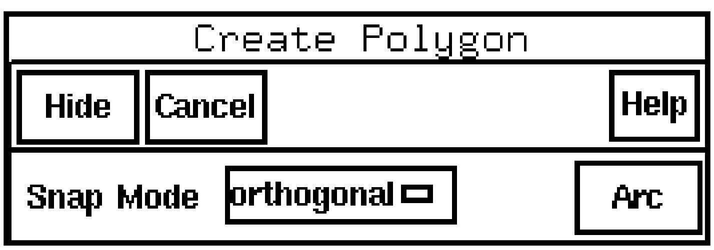
Canceling Commands

To cancel a command without changing your data, or to stop a command that automatically repeats, do one of the following:

- Press `Escape`.

  If the command has a form, the form closes.

- Click *Cancel* in the form.



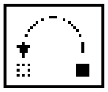
Undoing Commands

You can undo the effects of a command that you just completed.

To undo a command, do one of the following:

- Choose *Edit – Undo*.
- Press u.

- Click on the undo icon in the icon menu.



To reinstate a change you canceled with *Undo*, do one of the following:

- Choose *Edit – Redo*.
- Press `Shift-u`.

You can undo up to 10 previous commands. To set the number of commands you can undo,

1. In the Command Interpreter Window (CIW), choose *Options – User Preferences*.
2. Set the *Undo Limit* field to the number you want.
3. Click *OK*.

Repeating Commands

Many layout editor commands automatically repeat. For example, each time you finish drawing one rectangle, the *Create Rectangle* command prompts you to draw another one.

By default, the following commands are set to repeat:

- On the *Create* menu: all commands
- On the *Edit* menu: *Move*, *Copy*, *Stretch*, *Reshape*, *Delete*, *Chop, Split*, *Merge*, and *Paste*
- On the *Window* menu: *Create Ruler*

To set the automatic repeat (on or off),

1. Choose *Options – Layout Editor* [`Shift-e`].
2. In the Layout Editor Options form, set *Repeat Commands* (on or off).
3. Click *OK*.

To stop a repeating command, do one of the following:

- Press `Escape`.
- If the command has a form, click *Cancel* in the form.

Nesting Commands

You can pause before completing one command to perform a second command. This is called nesting a command. You can nest any command that does *not* automatically repeat inside any other command.

1. Choose *Edit – Move* (to move an object across the cellview).
2. Choose *Zoom – Out by 2* (to zoom out the cellview display).

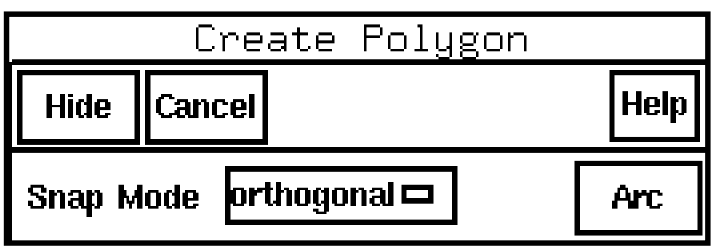
   You are now nesting the *Zoom* command while using *Move*.
3. Finish the *Move* command (click to move the object, or press `Escape` to cancel).

You can nest up to 20 commands. To set the number of commands you can nest,

1. In the CIW, choose *Options – User Preferences*.
2. Set the *Nest Limit* field to the number you want.
3. Click *OK*.

Using Command Forms

A form is a window that appears when you use a command. You use the form to change command settings. For example, in the Create Polygon form, you can change the snap mode to any of the options listed in the *Snap Mode* cyclic field.



There are two types of forms in the Layout Editor:

- Standard forms let you change command settings before you execute commands. They appear automatically when you start a command.
- Options forms let you change command settings while you are running commands. Options forms do not appear automatically if you have *Options Displayed When Commands Start* turned off in the *Options – User Preferences* in the CIW. In this situation, you must double-click middle or press F3 to see the command forms.

Displaying Forms

There are two ways to display forms, depending on whether the command has a standard form or an options form.

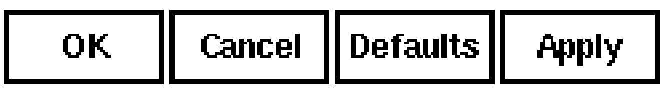
Whenever you choose a menu command that has three dots (...) after it, a standard form appears automatically.

Whenever you double-click middle or press F3 while using a command, an options form appears.

If you are not sure whether a command has a form, double-click middle or press F3 while you use the command. If a form is available, it will appear.

Using Form Buttons

The buttons on a standard form work as follows:



OK: Completes the command and closes the form.

Cancel: Closes the form without executing the command.

Defaults: Resets default values for options on the form.

Apply: Completes the command and keeps the command active and the form on the screen.

The buttons on an options form work as follows:

Hide: Closes the form and lets you go on with the command.

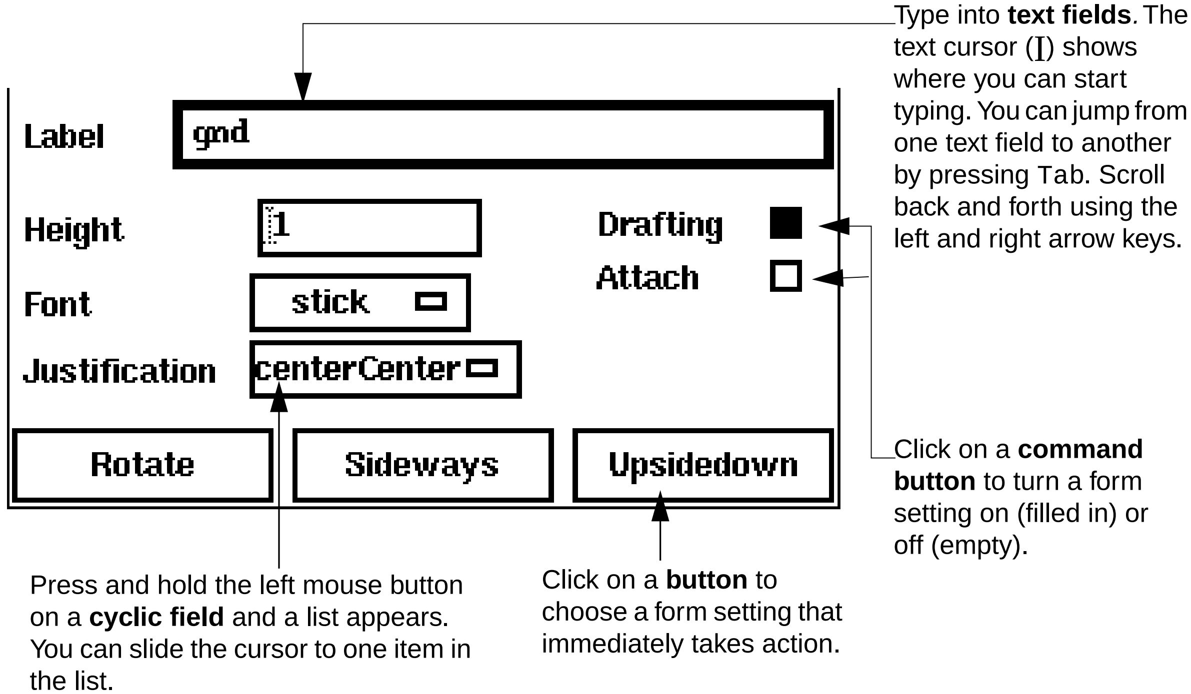
Cancel: Closes the form and stops the command.

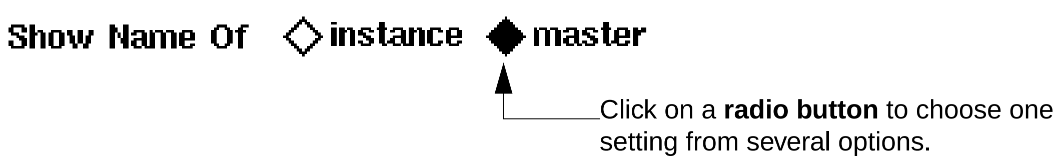
Defaults: Resets default values, if any, for options on the form.

Note: Many forms do not have a *Defaults* button because there are no appropriate default settings for that command. Some commands may have more or fewer buttons than those shown here.

Filling in a Form

In addition to the buttons at the top of a form, there are several types of fields and buttons inside a form, as shown in the following examples.



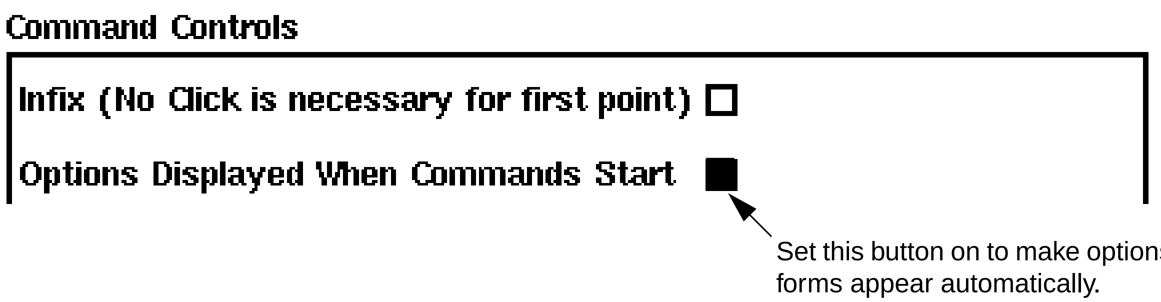


Making Command Forms Appear by Default

To set options forms so they display automatically whenever you choose a command,

1. Choose *Options – User Preferences* in the CIW.

2. In the User Preferences form, turn the *Options Displayed When Commands Start* button on.



3. Click *OK*.

Getting Help for Commands

The *Help* button on forms and in windows displays information about the layout editor.

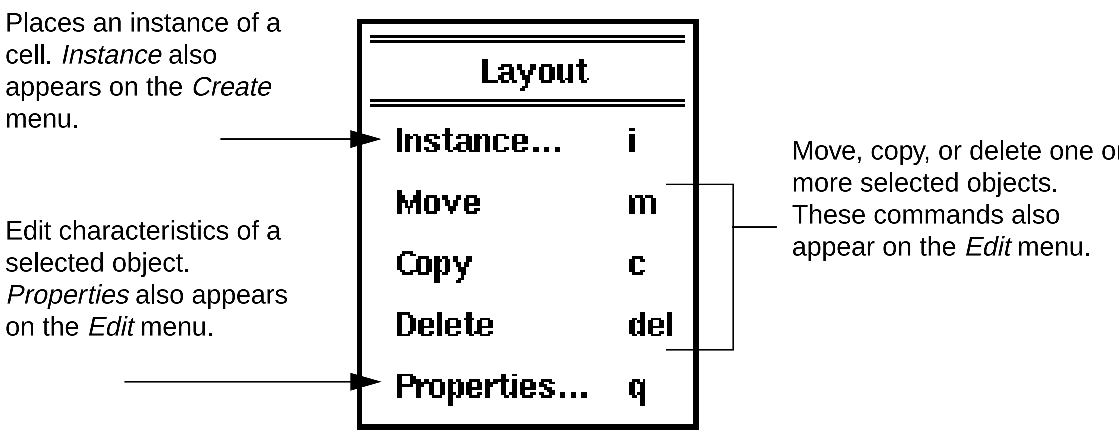
- To display a page of information about the command you are using, click *Help* in the command form or options form.
- To display the *Virtuoso Layout Editor User Guide* table of contents from which you can navigate to the information you want to see, click *Help* in a layout design window and choose *Contents*.
- Another way to display Help is to press the `F1` key at the top of your keyboard. `F1` is the bindkey for Help.
- To display a page of information about a command, press `F1` while your cursor is in the design window and the command is running.
- To see the *Virtuoso Layout Editor User Guide* table of contents, press `F1` while your cursor is in a layout design window.

Using the Pop-Up Menu

The layout editor pop-up menu lets you start a few commonly used layout commands. To start a command from the pop-up menu,

1. Move the mouse pointer into a layout cellview.

2. Press and hold the middle mouse button.
3. Slide the pointer to the command you want and release the middle mouse button.



To close the pop-up menu without using a command,

➤ Move the pointer off of the pop-up menu and release the middle mouse button.

➤ For information about creating your own pop-up menu, see the *User Interface SKILL Functions Reference* manual.

Discarding All Edits

The Discard Edits dialog box ignores all edits you made since the last time you saved.

To cancel your edits,

1. Choose *Design – Discard Edits*.

   A Discard Edits dialog box appears and asks you to confirm that you want to discard your edits.

2. Click *Yes*.

Discard Edits Form

Yes deletes all of the edits you made since the last time you saved.

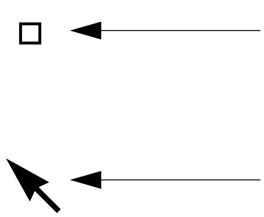
No cancels the command without deleting your edits.

Ways to Use the Mouse

You can use the mouse to perform several functions in the layout editor.

The Mouse Pointer and Cursor

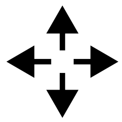
As you move the mouse around in a layout cellview, you see two objects that show where the mouse is pointing.



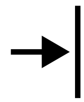
The mouse **cursor** snaps to the grid or to objects in a cellview. Use it to select or create objects.

The mouse **pointer** helps you see the cursor position. Use it to choose commands or options in command forms.

The mouse pointer changes shape after you select an object, to show that you can either move or stretch the object.



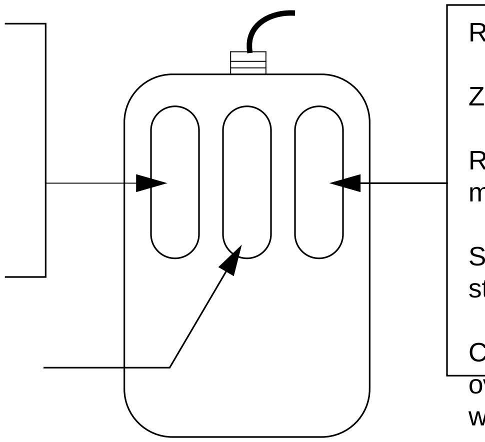
The **Move pointer** shows that you can move an object.



The **Stretch pointer** shows that you can stretch an object.

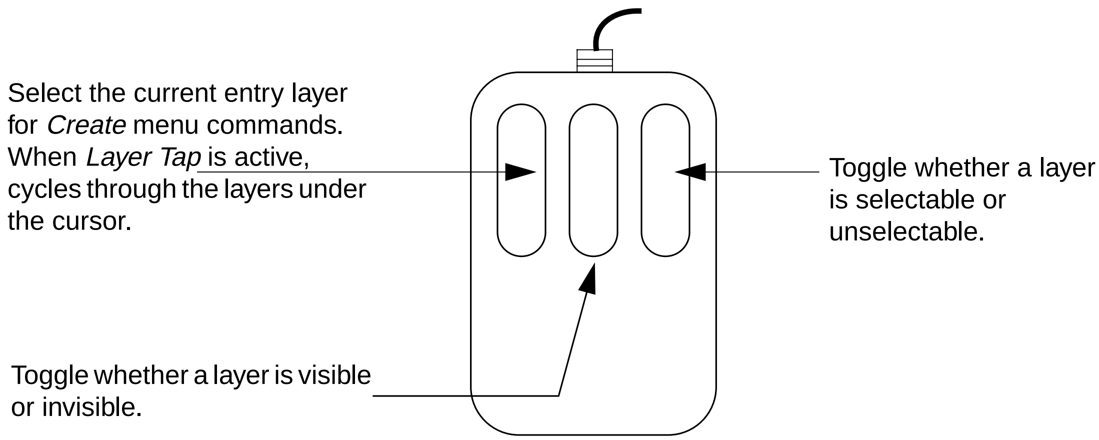
Mouse Buttons in the Layout Cellview

While editing a layout cellview, use the mouse buttons as follows.



Mouse Buttons in the Layer Selection Window

When you click on layer names in the Layer Selection Window (LSW), the mouse buttons work as follows.



Looking at the Mouse Button Settings

The mouse button settings appear at the bottom of a layout cellview. They show you what will happen if you click the left, middle, or right button. For some commands, new mouse settings appear when you press `Control` or `Shift`.

Each time you start a command, the mouse settings line changes to show what the mouse buttons do when you use this command. For example, while using the *Copy* command, the mouse button settings look like this:

```
mouse L: Enter Point M: Pop-up Menu R: Rotate 90
```

When you move the mouse cursor into the LSW, the mouse settings line shows what clicking mouse buttons do in the LSW. For example:

```
mouse L: Set Entry Layer M: Toggle Visibility R: Toggle Visibility
```

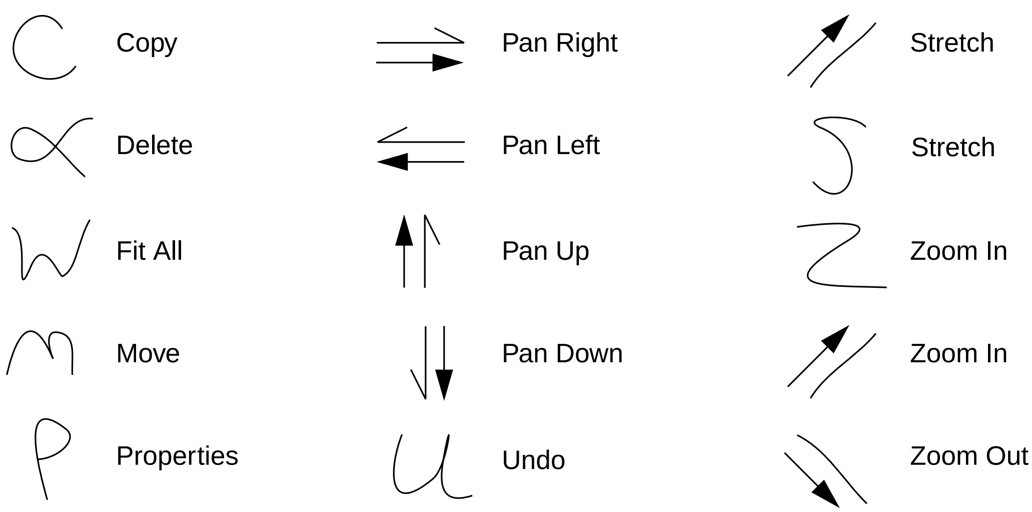
Layout Editor Strokes

A stroke is a unique shape you draw in a cellview by pressing and holding the right mouse button. Cadence ships a set of predefined strokes for the layout editor that you can load.

Note: Before you can use strokes, you must load them from the Cadence installation hierarchy.

To start a command with a stroke,

➤ Press right and draw one of the following strokes.



The arrows show the direction in which you draw the strokes. Note that there are two ways to start *Stretch* and *Zoom In* with strokes.

Loading and Unloading Strokes

To use the predefined layout editor strokes, you must first load them.

To load the strokes,

➤ Type the following in the text line of the CIW:

```
load(prependInstallPath( "etc/sted/stroke.il"))
load(prependInstallPath( "etc/sted/defstrokes.il"))
hiLoadStrokeFile("def.strokes" "Layout")
```

The `stroke.il` file defines mouse key bindings and the applications that recognize strokes. The `defstrokes.il` file lists the Cadence® SKILL language functions used by the predefined strokes. The `def.strokes` file defines the stroke shapes.

You can also add these commands to your `.cdsinit` file, so the strokes are loaded whenever you start the Cadence software.

Note: After you load strokes, you cannot use the right mouse button for any other functions.

➤ To unload the strokes, quit the current session of the Cadence software and restart the software. Also, remove the load strokes commands from your `.cdsinit` file if you have added them to that file.

Starting Commands with Bindkeys

Many layout editor commands can be started using bindkeys. If the bindkeys are loaded, they appear to the right of the command name on the menu.

To start a command using a bindkey,

➤ Move the cursor into the design window and press the bindkey on your keyboard.

For example, to start the *Fit All* command, you press `f`.

To start the *Clear All Rulers* command, you press `Shift-k`.

Window Menu	
Pan tab	Name of key means press that key (`Tab`).
Fit All f	Lowercase f means press `f`.
Fit Edit ^x	Carat (^) with x means press `Control-x`.
Clear All Rulers K	Uppercase K means press `Shift-k`.

This document refers to commands by full menu names. If a bindkey is available for a command, it is included in brackets after the command name. For example, *Zoom – In* [`z`].

The physical representation of the bindkeys on the keyboard is in Appendix A.

Entering Startpoints with Bindkeys

You can set the layout editor to automatically use the current cursor location as the starting point whenever you start commands with bindkeys. This is called infix mode.

To set infix mode on or off,

1. Choose *Options – User Preferences* in the CIW.
2. In the User Preferences form, turn the *Infix* button on or off.

By default, *Infix* is set off.

3. Click *OK*.

After you set *Infix* on, it affects all commands you start with bindkeys.

For example, if you press `Tab` (the bindkey for *Pan*), the layout editor does not prompt you for a panning point but immediately centers the image in the design window around the current cursor location.

Loading Bindkeys

Your system administrator might set your `.cdsinit` file to automatically load the bindkey definitions. If this is not the case, and your `.cdsinit` file does not load the layout editor bindkeys, you can add the line as follows:

1. In a UNIX window, open your `.cdsinit` file.

   ```
   vi ~/.cdsinit
   ```

   Your `.cdsinit` file might be in your home directory, your working directory, or in your site startup directory. The example assumes the file is in your home directory.
2. Add the following line to `.cdsinit`:

   ```
   load(prependInstallPath("samples/local/leBindKeys.il"))
   ```
3. Save and close the `.cdsinit` file.

   The next time you start the Cadence software, the `.cdsinit` file will load the default bindkey settings.

To load the bindkeys while the Cadence software is running,

➤ Type the following command in the CIW:

```
load(prependInstallPath("samples/local/leBindKeys.il"))
```

Defining Bindkeys in SKILL

To set your bindkeys, you can edit the `leBindKeys.il` file.

1. Make sure that your `.cdsinit` file includes a line to load the bindkey file, described in "Loading Bindkeys" on page 81.

Note: This document assumes you use the default bindkeys for the layout editor. If you change your bindkeys, remember that the bindkey instructions in the documentation might not apply.

2. In a UNIX window, copy and then open `leBindKeys.il` for editing:

   cd *your_install_dir*/tools/dfII/samples/local

   ```
   cp leBindKeys.il ~
   ```

   ```
   vi ~/leBindKeys.il
   ```

   The file defines groups of bindkeys. It uses an alias (`bk`) for the `hiSetBindKey()` function that sets bindkeys. Here are some examples from the bindkey file:

   Sets lowercase r to Rectangle.

   ```
   bk("Layout" "<Key>r" "leHiCreateRect()")
   bk("Layout" "Shift<Key>r" "leHiReShape()")
   ```

   Sets uppercase R to Reshape.

   The application name `Layout` in the syntax means that the bindkey applies whenever you run the layout editor.

3. Type `hiSetBindKey()` to create new settings.

   You can set a bindkey to perform any Cadence SKILL language function.

4. When you are finished editing the file, save and close it.

5. Type the following in the CIW to load the new file:

   load "*your_install_dir*/leBindKeys.il"

6. Try your new bindkeys in a Layout window.

7. If you want to load the new bindkeys automatically whenever you start the Cadence software, do one of the following:
   - Add a line to your `.cdsinit` file that loads the new bindkey file. For example:

     `load "`*`path_to_file/`*`leBindKeys.il"`
   - Append the new bindkey file to the end of your `.cdsinit` file.

Note: There is another sample bindkey file in the samples directory called `leSchBindKeys.il`. This sample bindkey file can be used with both the layout and schematic editors. These bindkeys are mapped to similar functions in both editors and are based on the default layout editor bindkeys.

Unexpected Results Commands Might Produce

Canceling or Undoing Edits

You might find that you want to cancel an edit you made. What you do depends on whether you completed the edit and how many edits you want to cancel.

- If you are still running the command, press `Escape` or click *Cancel* to stop the command and cancel the edit you were just doing.
- If you just completed a command and want to cancel it, choose *Edit – Undo* [u].

  Undo cancels the last command you completed.

  To reinstate a change you canceled with *Undo*, press `Shift U` or choose *Edit – Redo*.

If you completed a number of edits and do not want to save any of them, you can discard all edits you made since the last time you saved.

Note: Discarded edits are deleted from memory. You cannot restore them with *Undo*.

The Command Starts Before I Enter Points

If you press a bindkey to start a command and the command seems to start before you click a point, infix mode is probably on. When infix is on, any command you start with a bindkey uses the current cursor location as its first point.

If you do not want bindkey commands to use infix mode, you can turn infix mode off.

Stopping a Command

If you cannot stop a command, one of the following might be happening:

- Each time you finish the command, it restarts. This means you are using a command that automatically repeats.
- A form appears on screen, and you want to close it without executing the command.

In either case, do the following to cancel the command:

➤ Press `Escape`.

It is not always necessary to cancel a repeating command, however.

- If you are running a repeating command and then start a nonrepeating command, the repeating command pauses while the new command runs. After you are finished with the nonrepeating command, the repeating command continues.
- If you start another repeating command, the first repeating command stops.

The Command Does Not Repeat and It Should

By default, editing commands repeat automatically only if you first choose the command, then select the object.

Commands *do not* repeat if you first select the object, then choose the command.

If you chose an editing command before selecting an object, and the command still does not repeat, one of the following might be the cause:

- An object is selected that you cannot see. Look at the number next to *Select* in the status banner to see if one or more objects is selected. For example, if one object is selected the banner displays

  ```
  Select: 1
  ```

  Zoom out the cellview so you can see all objects.
- The *Repeat Commands* setting for the editor has been turned off.

To turn the *Repeat Commands* setting on,

1. Choose *Options – Layout Editor*.
2. In the Layout Editor Options form, set *Repeat Commands* on.
3. Click *OK*.

4

Setting Up Your Environment

This chapter contains these topics:

Setting Layout Editor Defaults

Before you can start working in the Virtuoso® layout editor, several startup files must be initiated. Some of the things these files do include setting up your environment, pointing to libraries, and defining your plotters.

Startup Files

The layout editor uses the following startup files:

File	.cdsinit
Purpose	A Cadence® SKILL language file executed when the Cadence design framework II (DFII) product starts.
User location	`~/.cdsinit`
Sample location	*your_install_dir*/tools/dfII/samples/local/cdsinit
System default location	`./.cdsinit`

File	.cdsenv
Purpose	Holds application defaults for environment variables.
User location	`~/.cdsenv`
Sample location	*your_install_dir/*tools/dfII/samples/.cdsenv
System default location	`./.cdsenv`

File	.cdsplotinit
Purpose	Initialization script for plot operations.
User location	`~/.cdsplotinit`
Sample location	*your_install_dir*/tools/plot/samples/cdsplotinit.sample
System default location	`./.cdsplotinit`

File	display.drf
Purpose	Specifies how you want your layers to appear on your monitor and in your plots.
User location	`~/display.drf`

Sample location	`your_install_dir/share/cdssetup/dfII/default.drf`
System default location	`./display.drf`

File	cds.lib
Purpose	Sets the paths to libraries and other `cds.lib` files. This file is used by the Library Browser, the *File – Open* command, and the *File – New* command.

.cdsinit File

The `.cdsinit` file can be used to initialize your Cadence environment. You can set window and layout editor defaults in the `.cdsinit` file, so your settings are loaded whenever you start Cadence® design framework II (DFII) workbench.

You can store the `.cdsinit` file in any or all of the following locations. DFII looks for the `.cdsinit` file in the following order:

`your_install_dir``/tools/dfII/local/.cdsinit` in your site installation directory

`~/.cdsinit` in your home directory

`./.cdsinit` in your working directory

- To see the sample `.cdsinit` file, type the following in your top Cadence directory:

  `more` *`your_install_dir`*`/tools/dfII/samples/local/cdsinit`

  The sample `.cdsinit` file does not have the "`.`" preceding it.

- To set defaults in the `.cdsinit` file, you enter Cadence SKILL commands that control environment variables.

.cdsenv File

The `.cdsenv` file sets application environment variables. You can use this file to change the layout editor default settings.

To change the layout editor default setting,

1. Copy the `.cdsenv` file to your home directory by typing

```
cp your_install_dir/tools/dfII/samples/.cdsenv ~
```

2. Use a text editor to edit it.

The `.cdsenv` file needs to contain only the defaults you change. If you save the `.cdsenv` file to your home directory, it is loaded when you start the software. You can change your setting during a session by loading a different file.

Note: Do not copy the system default `.cdsenv` file located in the following directory:

your_install_dir`/tools/dfII/etc/tools/layout/.cdsenv`

It contains an extra argument and will not work as your `~/.cdsenv`.

Other information you can store in the `.cdsenv` file is new values for `DBUPerUU`. The default `DBUPerUU` settings are located at

your_install_dir`/tools/dfII/etc/tools/cdba/.cdsenv`

Types of Measurement Units

`userUnits`	Sets the unit of measure for your design and can be microns, millimeters, centimeters, meters, mils, or inches.
`DBUPerUU`	Sets the number of design database units per user unit.

Setting Default Values with Menu Commands

The following layout editor and DFII commands and forms control most of the default values for your design environment:

Display Options form	Controls display options in layout windows.
Layout Editor Options form	Controls the options for layout editor behavior.
Virtuoso XL Options form	Controls the values of the Virtuoso layout accelerator (Virtuoso XL) options.
User Preferences form	Controls your Cadence software environment. These settings affect all Cadence applications, not just the layout editor.

Display Resource Editor (DRE)	Controls the appearance of layers.
Edit Layers form	Controls the data in the Layer Definition class of a technology file.

Setting Up the .Xdefaults File

The `.Xdefaults` file settings control how the Cadence software works in the X Window System environment. If you want to change these settings, copy the sample `.Xdefaults` file from *`your_install_dir`*`/tools/dfII/cdsuser` to your home directory (~) and edit it. The layout editor uses the system default settings if there are no settings in your home `.Xdefaults` file.

For a complete list of `.Xdefaults` settings for Cadence software, read the DFII section of the *Cadence Configuration Guide*.

The following list shows the settings that affect the layout editor and sample options.

Title	**Setting**	**Description**
`Opus.LSWGeometry`	`250x500+100+100`	Specifies size and location of Layer Selection Window (LSW).
`Opus.geometry`	`624x245+515+643`	Location and size of the Command Interpreter Window (CIW) (overrides `opus.x`, `opus.y`, `opus.height`, and `opus.width`).
`Opus.textFont`	`fixed`	Font for text in type-in fields, ShowFile windows, and the CIW input and output areas.
`Opus.attentionTextColor`	`blue`	Color of text in dialog boxes.
`Opus.background`	`LightBlue`	Color of the background for everything except graphics windows.
`Opus.bottomShadowColor`	`MediumBlue`	Color for the bottom shadow of three-dimensional widgets.

`Opus.foreground`	`blue`	Color of menu and listbox text and selected radio and toggle buttons.
`Opus.textColor`	`chocolate4`	Color of text in type-in fields, ShowFile windows, and the CIW input and output areas.
`Opus.topShadowColor`	`azure`	Color of top shadow of three-dimensional widgets.
`Opus.activeBannerColor`	`pink`	Color of the active window banner.
`Opus.borderColor`	`black`	Color of field borders if `hiShowFieldBorders(t)` has been called.
`Opus.buttonColor`	`black`	Color of button and label text.
`Opus.dragColor`	`white`	Color of the selection box you draw around objects; the zoom box you draw to zoom in or out; the outlines of objects you move, copy, or reshape; and the outlines of edges you stretch. `dragColor` controls the highlight that appears *before* you select objects. Objects and edges are highlighted as you move the cursor over them.
`Opus.editorBackground`	`black`	Color of design window background.
`Opus.flashColor`	`pink`	Color of flashing box drawn around error fields in forms.
`Opus.recessColor`	`grey70`	Color of uneditable text fields and depressed buttons.

`Opus.formPlacement`	`left`	Location of form (`top`, `bottom`, `left`, `right`, or `center`).
`Opus.formRelativeTo`	`CIW`	Location of form relative to `screen`, `currentWindow`, or `CIW`.
`Opus.labelFont`	`-adobe-helvetica-bold-r-*-*-12-*`	Font used for form labels, button text, and menu text.
`Opus.optionFormPlacement`	`left`	Placement of option form (`top`, `bottom`, `left`, `right`, or `center`).
`Opus.optionFormRelativeTo`	`CIW`	Placement of option form relative to `screen`, `currentWindow`, or `CIW`.
`Opus.textEditor`	`vi`	Text editor.

Opus.LSWGeometry

When the layout editor starts, `Opus.LSWGeometry` settings determine the size and location of the LSW, excluding the border and window title. The border and window title are determined by the window manager.

`Opus.LSWGeometry` works in the following manner:

- X Window System geometry specifications are used with the exception of negative integers. If negative integers are used, the window is displayed at the default location. For example, if you set `Opus.LSWGeometry` at `200x500+-100+-100`, you would expect the LSW to display 100:100 from the lower right corner of the screen. Because the negative integers are not recognized, the LSW defaults to the upper left corner of the screen, 0:0. The CIW displays a warning that the origin is less than zero and that the origin will default to 0:0.
- The window origin is set at the upper left corner of the screen. For example, if you set `Opus.LSWGeometry` at `200x500+100+100`, the LSW is displayed 100:100 from the upper left side of the screen. The size is 200 pixels wide by 500 pixels high.
- The default size and location is 120 pixels wide by 700 pixels high from the upper left corner of the screen, 0:0.
- The minimum size is 120 pixels wide by 175 pixels high. If a smaller size is set, the CIW displays a warning that the default size will be used.

- The LSW must fit within the screen's width and height. If it is set larger, the size defaults to the screen width and height. The CIW displays a warning that the screen width and height will be used.
- Use `leGetLSWBBox()` to see the bounding box of the LSW.

Note: `Opus.LSWGeometry` might not work with all window manager configurations. For example, in `ctwm`, the option `UsePPosition` must be `on` in order to specify the location of the LSW with `Opus.LSWGeometry`.

Using the Technology File

Each library must be associated with a technology file. The technology file defines

- Design layers and their properties
- Physical design rules used for compaction, symbolic checking, and interactive verification
- Contacts used by the layout editor
- Devices and pins used by the Virtuoso compactor and layout synthesizer tools
- Application-specific rules that control how applications work

You can associate multiple libraries with a single technology file using the *Attach To* command.

For complete details about how to create and edit technology files, refer to any of the following:

- *Technology File and Display Resource File SKILL Reference*
- *Technology File and Display Resource File User Guide*

Technology File Requirements for the Layout Editor

The layout editor requires the technology file to contain complete definitions of the layers used in your design. The technology file can also have

- Physical rules for each design layer
- Definitions of symbolic contacts

  Contact definitions are required only if you wish to place contacts using the *Create Contact* command or perform path stitching, which automatically places contacts using the *Create Path* command. The `mpu.tf` file contains sample contact definitions. It is located in this directory: *your_install_dir*`/tools/dfII/samples/techfile.`

Using Environment Variables

Graphic Environment Variables

Graphic environment variables control the window display. These variables can be stored in the cellview or the `.cdsenv` file in your home directory. There are several ways to change the default settings of graphic environment variables:

- Set the variable in the Display Options form.

  If you want to retain these settings for future sessions, save the settings using the *Save* button in the *Options – Save Defaults* form in the CIW.

- Copy the variables you want to change from the default `.cdsenv` file to the `.cdsenv` file in your home directory.

  It is there that you edit the default value. The default file is

  your_install_dir/`tools/dfII/samples/.cdsenv`

  Note: Do not use the `.cdsenv` system files located in the following directory: they contain system default values for all applications and have extra arguments that do not work in your environment.

  your_install_dir/`tools/dfII/etc/tools/*/.cdsenv`

- Set the variable in the CIW using

  `envSetVal("graphic" "`*name*`" `*type value*`)`

  For example:

  `envSetVal("graphic" "iconsOn" 'boolean t)`

  To determine the current value of a graphic environment variable, type in the CIW

  `envGetVal("graphic" "`*name*`")`

  For example:

  `envGetVal("graphic" "iconsOn")`

Layout Editor Environment Variables

Layout environment variables control how various layout editor commands work. There are several ways to change the default settings of layout environment variables:

- Set the variable in the Layout Editor Options form. If you want to retain these settings for future sessions, save the settings in the Save Defaults form in the *Options* menu in the CIW.
- Include the environment variables in your `.cdsenv` file in your home directory. The default file is

  your_install_dir/`tools/dfII/samples/.cdsenv`
- Include the `envSetVal()` function in your `.cdsinit` file.
- Include the `envSetVal()` function in any other SKILL file you load.
- Type the `envSetVal()` function in the CIW.

For example, to set the `gravityOn` variable, type the following in the CIW or include it in a file:

```
envSetVal("layout" "gravityOn" 't)
```

To determine the current value of any layout editor environment variable, type the `envGetVal()` function in the CIW as follows:

```
envGetVal("layout" "variable_name")
```

You can see the entire list of graphic and layout editor variables in the *Custom Layout SKILL Functions Reference.*

Setting Window and Form Location

Placing a Window Manually

To set the Cadence software so that each time you open any cellview, you define the size and location of the window,

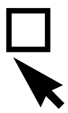
1. Choose *Options – User Preferences* in the CIW.

   The User Preferences form appears.
2. Turn on *Place Manually*.

3. Click *OK*.

4. Choose *Window – Utilities – Copy Window* to see the results of the change in another window.

   A rectangle appears next to the mouse pointer:

5. Click where you want one corner of the design window and drag the pointer to the opposite corner.

   You see an outline showing the size and location of the window.

6. Release the mouse button.

   The window appears inside the outline.

Setting a Default Window Size and Location

To set the default size and location of layout windows,

1. Resize and move a layout window where you want all your layout windows to be displayed.

2. Note the number in the upper right corner of the window.

3. Exit the Cadence software.

4. Open your `~/CDS.log` file and look at the bottom of the file for a `hiResizeWindow` entry for the window number you noted previously.

   The `CDS.log` file is created in your home directory.

   The `hiResizeWindow()` function shows the coordinates of any window you moved or resized. For example:

   ```
   hiResizeWindow(window(2) list(476:303 1103:869))
   ```

5. Add the following to your `.cdsenv` file:

   ```
   layout leWindowBBox string "((coord coord) (coord coord))"
   ```

   For example:

   ```
   layout leWindowBBox string "((476 303) (1103 869))"
   ```

Setting Options for Window Border Utilities

By default, the window display includes these optional utilities:

- The icon menu provides easy access to commonly used commands
- The mouse settings line shows the current mouse button actions
- The prompt line shows the next step for the current command

You can remove any of these objects from the window. You can also add a scroll bar that lets you scroll the design window.

To set what appears in the window border,

1. Choose *Options – User Preferences* in the CIW.
2. The User Preferences form appears. Set any of the window controls on or off.

3. Click *OK*.
4. Choose *Window – Utilities – Copy Window* to see the results of the change in another window.

Setting a Default Form Location

To set the default location for the lower left corner of all forms, do the following:

1. Determine the maximum X and Y coordinates (upper right most coordinate) of your screen. To find this coordinate, type the following in the CIW:

   ```
   hiGetMaxScreenCoords()
   ```

   The CIW displays `(1142 870)` the maximum X and Y coordinates. For this example, the rightmost X coordinate is 1142, and the uppermost Y coordinate is 870.
2. Estimate the coordinates where you want the upper left corner of the forms.
3. To set where forms appear, add the following to your `.cdsinit` file.

```
hiSetFormPosition(coordinate:coordinate)
```

For example:

```
hiSetFormPosition(710:600)
```

With these coordinates, forms open in the lower right side of the screen the next time you start the Cadence software.

Customizing Layout Editor Menus

You can customize (edit, add, remove) the Virtuoso layout editor menu items by editing the `layEdit.menus` file included in your installation hierarchy.

By editing the `layEdit.menus` file, you can define each menu item (specifically, the menu item symbol, its menu text, and the `callBack` function) and organize these definitions into the respective pulldown menus. You can change menu definitions without having to rebuild the SKILL context to see the change. Also, the menu file definitions are easier to maintain from one release to the next because they are independent of the code that implements the menus.

The `layEdit.menus` file contains the layout editor menu items. To edit this ASCII file, you must copy it to `~/menus` and then make your changes. You can leave the edited `layEdit.menus` file in your `~/menus` directory or copy it to a local, work, or project directory.

Location of the layEdit.menus File

The `layEdit.menus` file is read in the following order and at these locations.

your_install_dir`/tools/dfII/etc/tools/menus/layEdit.menus`	System information is read at this location.
your_install_dir`/local/menus/layEdit.menus`	Site-specific information is read at this location to allow for site overrides.
workArea`/menus/layEdit.menus` or `~/menus/layEdit.menus`	Work area, project, or `~/menus` information is read at this location to allow for personal overrides.

To see the changes you made in the `layEdit.menus` file, restart the software and open a cellview window.

Editing Considerations

When editing the `layEdit.menus` file, be aware of the following considerations:

- Never edit the `layEdit.menus` file in the directory

  `your_install_dir``/tools/dfII/etc/tools/menus/`

  Rather, copy it to `~/menus` for editing.

- Do not delete menu items or move pulldown menu items from one pulldown menu to another. Global menu item variables, referenced by other tools, assume menu items are on specific menus.
- Do not reorder the layout editor banner menu items. Other tools might modify the layout editor menus and assume the items are in a specific order.
- Do not add any menu items whose callbacks dynamically update other menu items.
- Functionality to gray out unavailable commands is not included in the `layEdit.menus` file.
- Changing menu items in a pulldown will cause a mismatch with the *Virtuoso Layout Editor User Guide* documentation.

Additional Reference Files

To learn how to install a sample menu file, read the file

`your_install_dir/``tools/dfII/etc/tools/layout/menus.sample`

For more information about customizing layout editor files, read the file

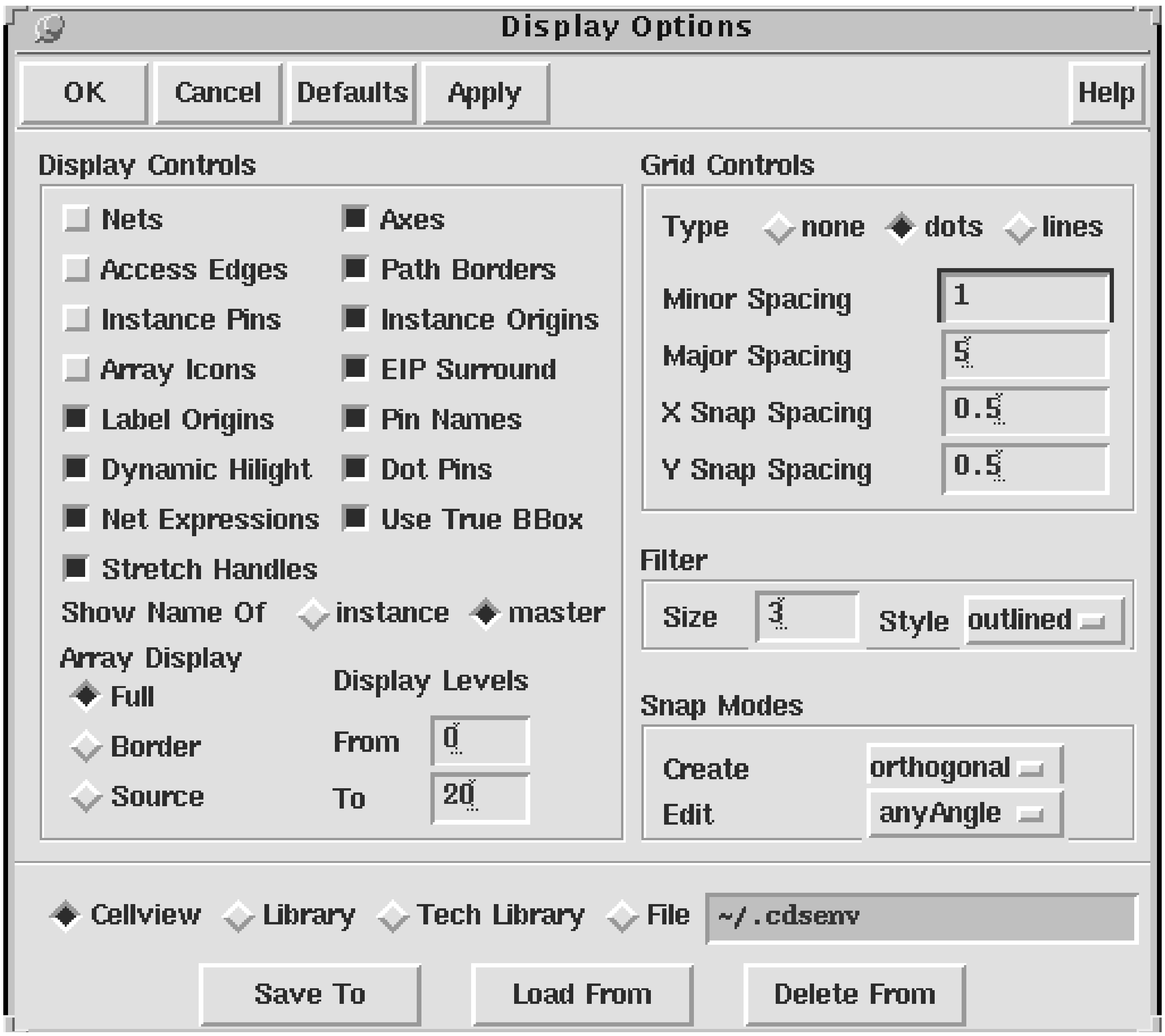
`your_install_dir``/tools/dfII/etc/tools/layout/README_menus.txt`

For general information about customizing menus, read the file

`your_install_dir``/tools/dfII/etc/tools/menus/README`

Using the Display Options Form

The Display Options form controls the appearance of objects and the behavior of commands in this cellview.



Nets shows flight lines between objects on the same net. If your design contains many nets, your screen may turn white, causing the instTerms on top of the instance to not be seen. To see the instTerms, turn on *Instance Pins* and the flight lines will not display, allowing the instTerms to be seen.

Access Edges shows routing edges of pins.

Instance Pins shows pins in instances.

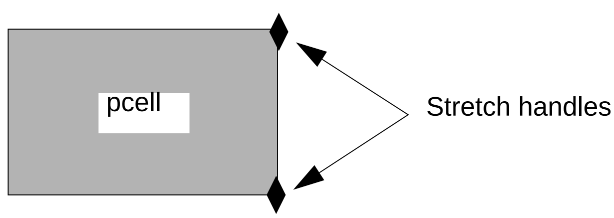
Array Icons shows outlines of array cells when *Display Levels* suppresses cell details.

Label Origins marks the origins of labels with diamond markers.

Dynamic Hilight marks the edge, object, or point that would be selected if a point selection were made. When *Dynamic Hilight* is on and your cellview contains a large number of objects, cursor motion may slow down. You can increase the cursor motion speed by turning this off.

Net Expressions displays the net expression instead of the terminal name of a pin. When there are net expressions in instances, the terminal name is displayed, not the net expression, even when *Net Expressions* is set on.

Stretch Handles displays the handles on a Relative Object Design parameterized cell (pcell) that indicate that the pcell can be stretched. A *stretch handle* is a named set of coordinates assigned to a specific parameter of the pcell. Stretch handles look like small diamonds. For information about stretchable pcells, see “Stretchable Parameterized Cells” in the *Virtuoso Relative Object Design User Guide.*



Axes displays the cellview X and Y axes.

Path Borders shows the border edges of paths. Turn it off to display only path centerlines.

Instance Origins marks cell instance origins with diamonds when you set *Display Levels* to show only instance outlines.

EIP Surround (edit-in-place) displays the surrounding design when you edit a cell in place.

Pin Names shows terminal names of pins that have pin name text displays.

Dot Pins displays the centers of dot pins with diamond markers.

Use True BBox when on, displays the instance master bounding box. When off, displays the cellview bounding box, which can cause large designs to open faster because masters are opened down to the display stop level only.

Show Name Of, when *Display Levels* is set to show only instance outlines, sets whether the instance name (for example, I1) or the master cell name appears on each instance.

Array Display

Full displays all instances in the array.

Border displays only the instances around the outside edge of the array.

Source displays only the instance at the origin of the array.

Display Levels sets the first (*From*) and last (*To*) levels in the design hierarchy that can be seen in detail. The hierarchy levels are numbered 0 to 32. The top cellview is level 0, instances inside of it are level 1, and so forth.

Type controls the grid display.

none turns off the grid display.

dots displays a dot for each grid point.

lines makes a grid of lines, like a graph.

Minor Spacing and **Major Spacing** set the number of user units between the visible grid. Minor grid points are white, major grid points are green by default.

X Snap Spacing and **Y Snap Spacing** set the distance at which the cursor can snap between grid points along the X axis and the Y axis. This is your drawing grid.

Filter determines how much detail of a design is displayed in the cellview. The filter can affect how fast the screen redraws. A smaller filter size allows more objects to display, which can cause the screen to redraw more slowly. A larger filter size allows fewer objects to display, which can cause the screen to redraw faster and can be useful when redrawing large, dense designs.

Size controls the size of the objects that are filtered out. With a smaller filter, more of the design displays. When *Size* is set to 3, the default, objects smaller than 3 pixels are filtered; objects larger than 3 pixels are not filtered.

Style controls how the filtered objects are displayed. Filtered objects appear either filled with their layer color, outlined with their layer color, or empty and nothing is displayed.

Snap Modes locks your cursor to the grid while drawing or editing.

Create controls how line segments snap to the grid as you create objects.

Edit controls how line segments of objects snap to the grid as you move or copy them and how edges or corners move as you stretch them.

Cellview specifies that you want to store, load, or delete the display settings to or from the cellview.

Library specifies that you want to store, load, or delete the display settings to or from the library of the edit cellview.

Tech Library specifies that you want to store, load, or delete the display settings to or from the technology library of the edit cellview.

File specifies the file to which you want to store or from which you want to load the settings.

Save To saves the current settings to either the cellview, library of the cellview, technology library of the cellview, or a specified file. If you are saving to a file, the settings from both the Layout Editor Options and Display Options forms are saved.

Load From sets the current settings to either the cellview, library of the cellview, the technology library of the cellview, or a specified file. If you saved to a file, the settings for both the Layout Editor Options and Display Options forms are loaded.

Delete From deletes the display settings that were saved to either the cellview, library of the cellview, or technology library of the cellview.

Setting Display Controls

In the *Display Controls* area of the form, you can choose what you want displayed.

Display Controls		Results
Nets	Axes	Set which objects appear in the cellview
Access Edges	Path Borders	Set which objects appear in the cellview
Instance Pins	Instance Origins	Set which objects appear in the cellview
Array Icons	EIP Surround	Set which objects appear in the cellview
Label Origins	Pin Names	Set which objects appear in the cellview
Dynamic Hilight	Dot Pins	Set which objects appear in the cellview
Net Expressions		Set which objects appear in the cellview
Stretch Handles		Set which objects appear in the cellview
Show Name of: instance, master		Show the instance or master cell names
Array Display		Display details or outlines of arrays
Display Levels		Displays a range of hierarchy levels

Setting Which Design Objects Appear

You can control the display of nets, edges, pins, array icons, label and instance origins, axes, path borders, pin names, EIP (edit-in-place) Surround, and Dynamic Hilight.

To control which design objects or attributes appear,

1. Choose *Options – Display*.

   The Display Options form appears.

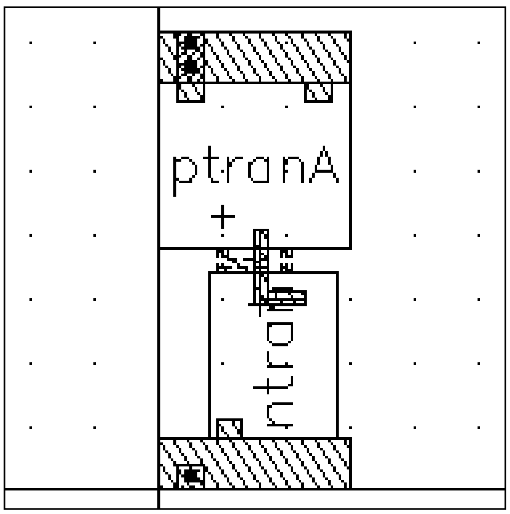
2. Click on any of the display controls.
3. Click *Apply* to see the result of your changes.
4. When you are satisfied with the settings, click *OK*.

Displaying Details or Outlines of Instances

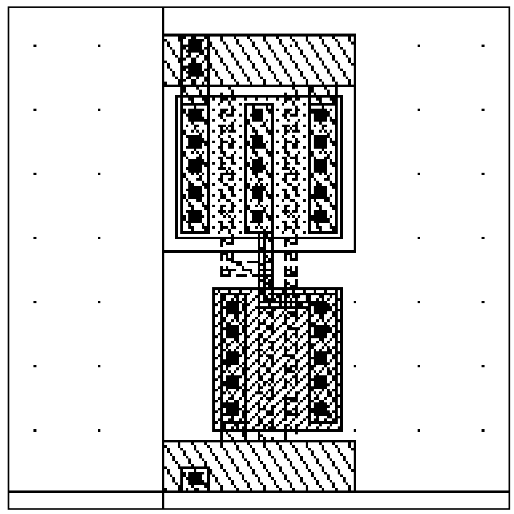
You can control whether details inside of cell instances placed in a cellview appear.

To turn off instance detail, press `Control-f` in the design window. This sets the display stop level to 0.

To turn on instance detail, press `Shift-f` in the design window. This sets the display stop level to 32.



Press `Control-f` and only the outlines of cell instances appear.



Press `Shift-f` and details inside cell instances appear.

Displaying a Particular Hierarchy Level

To choose which level in the design hierarchy is displayed,

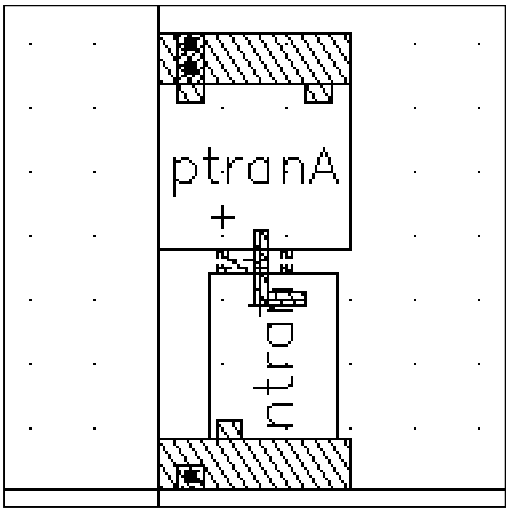
1. Choose *Options – Display*.

   The Display Options form appears.

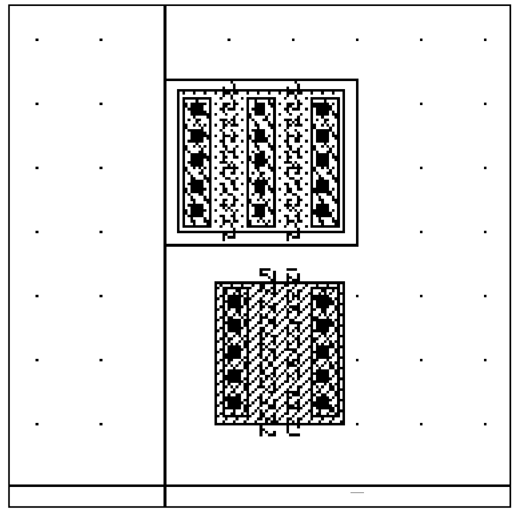
2. Set the display levels.

   From shows the lowest level of hierarchy that appears. By default, the display begins at the current level, 0.

To shows the highest level of hierarchy that appears. Cell instances inside this cellview are level 1, instances inside level 1 are level 2, and so on.



Only level 0 displayed (*From 0 To 0*).



Only level 1 displayed (*From 1 To 1*).

3. Click *OK.*

Using the Set Area View Level Form

The Set Area View Level form lets you turn off detail in most areas of the cellview. This can speed screen redraw time when you are working on a complex design and want to see detail in only one portion of it.

About the Set Area View Level Form

To open the Set Area View Level form,

➤ Choose *Window – Area Display – Set*.

Display Levels specifies the first (*From*) and last (*To*) levels in the design hierarchy that can be seen in detail. The hierarchy levels are numbered 0 to 32. Shapes in the current cellview are at level 0, shapes in the masters of instances inside of it are level 1, and so forth.

Creating Detail Areas

To set the display detail for an area,

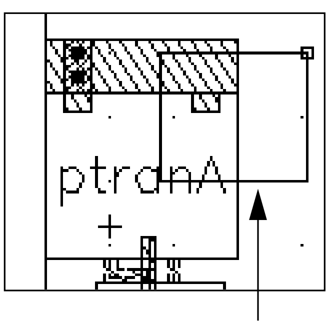
1. Choose *Window – Area Display – Set*.

   The Set Area View Level form appears.

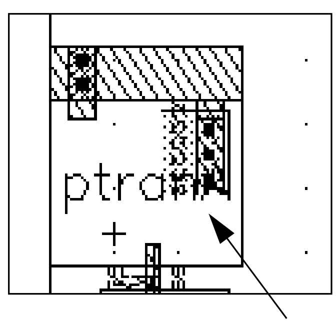
2. Set the *From* and *To* display levels for the area.

   A larger value for *To* shows more detail.

3. Create a box in the cellview around the area.



Create a box around an area.



That area shows the detail levels you specified.

4. You can continue to set detail areas, or click *Cancel* in the form to stop the *Set* command.

Removing Detail Areas

To remove one detail area at a time,

1. Choose *Window – Area Display – Delete*.
2. Click on the detail area you want to remove.

To remove all detail areas,

➤ Choose *Window – Area Display – Delete All.*

Displaying Instance or Master Cell Names

When you suppress cell instance detail (press `Control-f`), the Virtuoso layout editor displays one of the following for each instance:

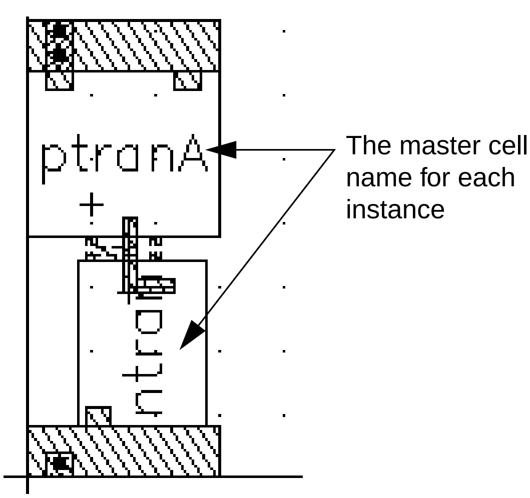
- The master cell name
- The instance name (usually an incremented number)

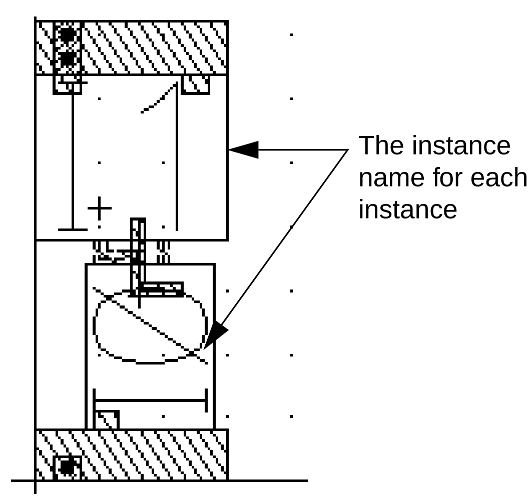
To choose what name you want displayed,

1. Choose *Options – Display*.

   The Display Options form appears.

2. Set *Show Name Of* to *instance* or *master*.
3. Click *OK*.





Displaying Details or Outlines of Arrays

You can adjust the display details of cell arrays.

To set the array detail,

1. Choose *Options – Display*.

   The Display Options form appears.

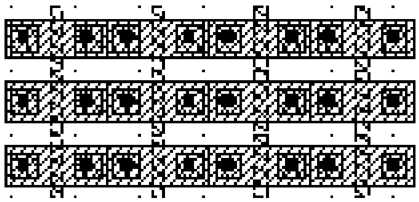
2. Set the *Array Icons*, *Array Display*, and *Display Levels* fields as desired.
3. Click *OK*.

Possible Options and Results

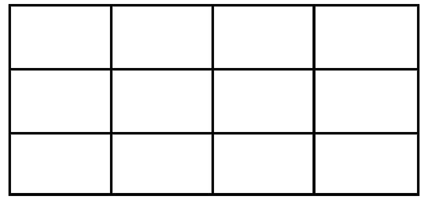
Display Option	Results
Array Icons	When *Array Icons* is on and *Display Levels* is set to *From 0 To 0*, outlines of array elements appear with no detail.
Array Display	*Array Display* settings control the number of elements in the array that appear.
Display Levels	*Display Levels* controls the level of instance detail to display.

Array Samples

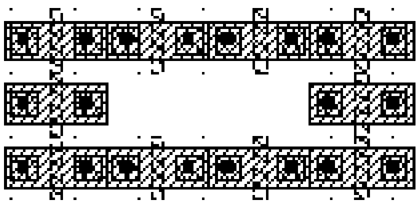
The following figures show sample settings for the display of arrays.



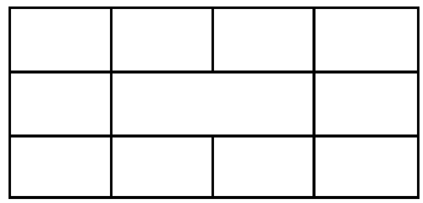
Array Icons: off
Array Display: Full
Display Levels: 0-32



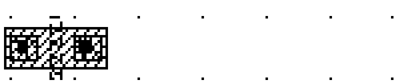
Array Icons: on
Array Display: Full
Display Levels: 0-0



Array Icons: off
Array Display: Border
Display Levels: 0-32



Array Icons: on
Array Display: Border
Display Levels: 0-0



Array Icons: off
Array Display: Source
Display Levels: 0-32

Setting the Visible Grid

By default, the cellview window shows a grid of dots. There are two different grids: the minor (small) grid and the major (large) grid.

- Minor grid points are white and appear at every micron.
- Major grid points are green by default and appear at every 5 microns.

To change the visible grid,

1. Choose *Options – Display*.

   The Display Options form appears.

2. Set the grid controls to the settings you want.
3. Click *OK*.

Setting Filter Size and Style

You can control how much detail of a design is displayed in your cellview by adjusting the *Filter Size* and *Style* settings in the Display Options form. The filter can affect

- How much of a design is drawn when loaded
- How fast the screen redraws

A smaller filter size allows more objects to display, which can cause the screen to redraw more slowly. A larger filter size allows fewer objects to display, which can cause the screen to redraw faster and can be useful when redrawing large, dense designs. Instances smaller than `filterSize` pixels are not opened and read in. As you zoom in, the instances become bigger than `filterSize`, and more cells are read in.

You set *Size* to control the size of objects that are filtered out. *Size* is a floating point number. When *Size* is set to `3.0`, the default, objects smaller than 3 pixels do not display; objects larger than 3 pixels do display. See "Filter Settings Examples" on page 113 for examples. You can toggle through filter sizes of 00, 3.0, 10.0, 25.0, and 50.0 by pressing `F9`. This overrides the filter size setting in the Display Options form.

You set *Style* to determine how the objects are displayed. If an object is less than or equal to the filter size, it appears as one of the following:

- Filled with its layer color
- Outlined with its layer color
- Empty, nothing is displayed

Objects larger than the filter size do not change in appearance.

To change the *Filter* settings,

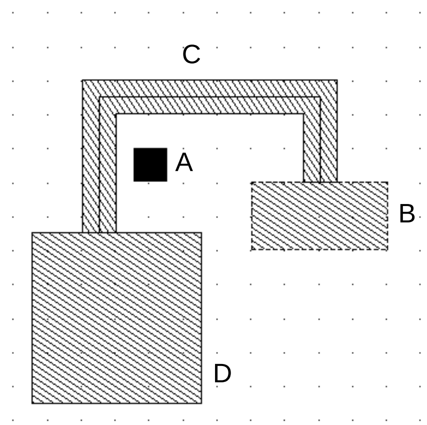
1. Choose *Options – Display*.

   The Display Options form appears.

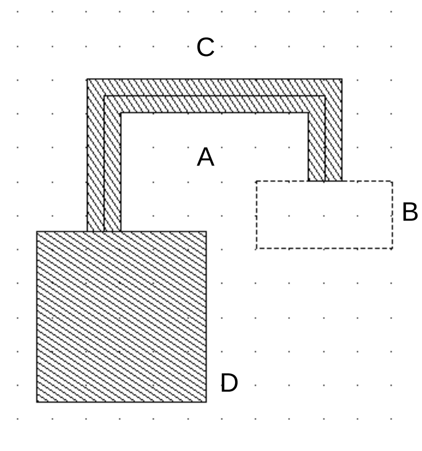
2. Set *Size* and *Style* to the settings you want.

   You have to press `Control-r`, to refresh your screen, to see the results of the new filter settings.

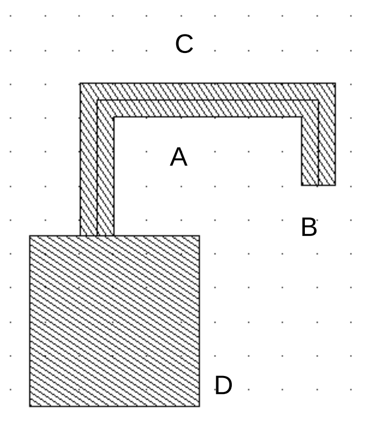
Filter Settings Examples



Filter settings:
Size = 3
Style = filled
All objects are displayed



Filter settings:
Size = 50
Style = outlined
Objects smaller than 50 pixels (A) are not displayed. Objects 50 pixels (B) in size are outlined. Objects larger than 50 pixels (C and D) retain their filled style.



Filter settings:
Size = 50
Style = empty
Objects equal to or smaller than 50 pixels (A and B) are not displayed. Objects larger than 50 pixels (C and D) retain their filled style.

Setting the Snap Grid

The grid defines the points at which the cursor (the small square) snaps to the cellview or to objects. With the snap grid set to 0.5 microns, the cursor can snap to each visible grid point or halfway between each grid point.

For example, if your user units are microns, and you want to draw objects at 0.5 micron intervals, you set the spacing for the snap grid to 0.5 microns.

To set the grid, you set the *X Snap Spacing* (distance the cursor can move along the X axis) and the *Y Snap Spacing* (distance the cursor can move along the Y axis) fields. Setting *X Snap Spacing* and *Y Snap Spacing* to 0 produces the same result as turning the environment variable `snapToGrid` off.

Gravity has precedence over snap grid. Turn *Gravity* off while using snap grid.

Note: The snap grid is not the same as the visible grid in the cellview.

Setting the Snap Grid Spacing

To set the snap grid spacing,

1. Choose *Options – Display*.
2. In the Display Options form, type the smallest distance you want between the points you enter along the X axis (*X Snap Spacing*) and Y axis (*Y Snap Spacing*).

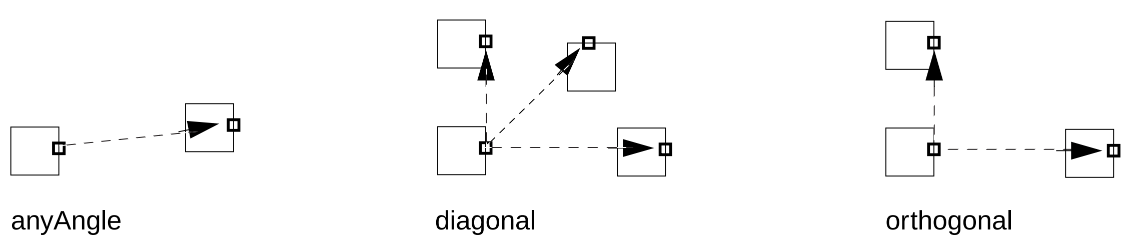
   The value you type is in user units (usually microns).
3. Click *OK*.

Snapping the Cursor as You Edit

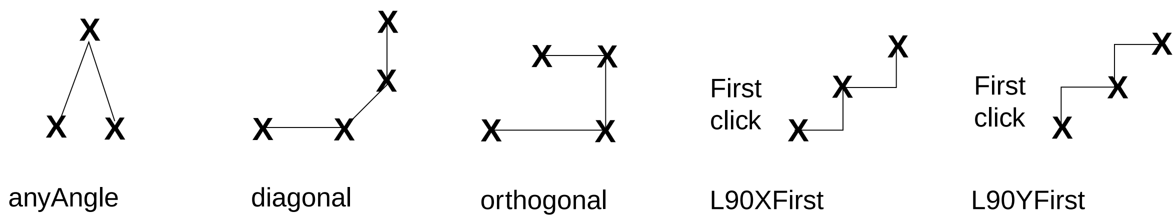
Snap Mode

The snap mode controls how the cursor locks to the grid as you create or edit objects.

- The *Copy*, *Move*, and *Stretch* commands use the snap mode to control where you can move all or part of an object. The settings are shown below.



- The *Chop*, *Create Path*, *Create Pin*, *Create Polygon*, *Reshape*, *Ruler*, *Split*, and *Yank* commands use the snap mode to control the shape of segments as you create or reshape objects. *Snap Mode* settings are illustrated below:



Setting the Default Snap Modes

To set the default snap mode,

1. Choose *Options – Display*.

   The Display Options form appears.

2. Set the snap modes.

   - *Create* controls how segments snap as you create objects. This mode affects the *Chop*, *Create Path*, *Create Pin*, *Create Polygon*, *Reshape*, *Ruler*, *Split*, and *Yank* commands.
   - *Edit* controls how segments snap as you move or stretch objects. This mode affects the *Copy*, *Move*, and *Stretch* commands.

3. Click *OK*.

Saving, Loading, and Deleting Display Settings

You can save, load, and delete the settings in the Display Options form to several locations. Saving the display settings lets people who work on the same design use the same display settings.

The display settings can be saved to the current cellview, library of the cellview, technology library of the cellview, or a specified file. To save the display settings,

➤ Choose *Save To* and either *Cellview*, *Library*, *Tech Library*, or *File*.

To load the display settings, do one of the following,

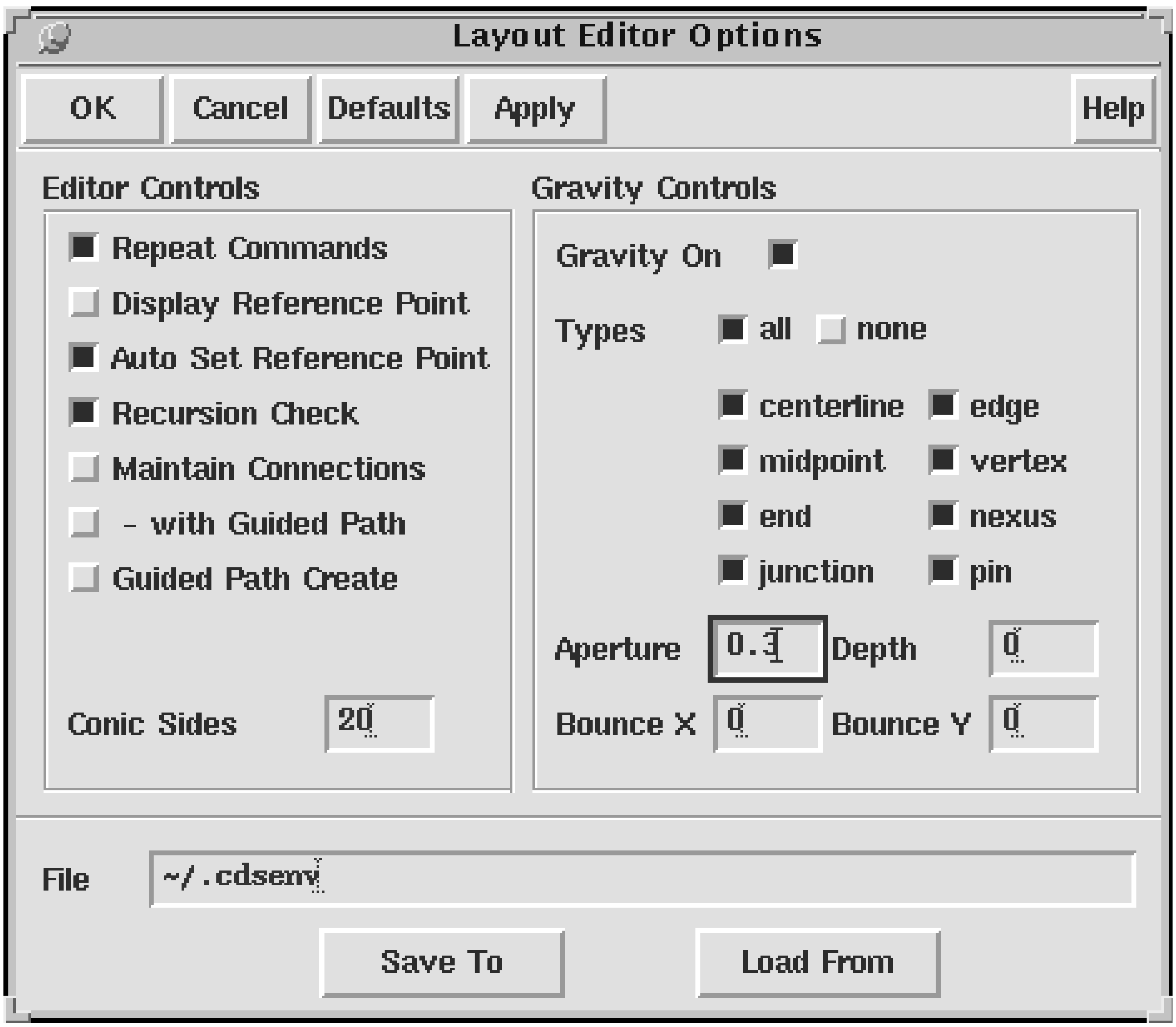
- Start the software. The display settings are loaded in this precedence, from highest to lowest:
    - Cellview
    - Library of the cellview
    - Technology library of the cellview
    - File (`~/.cdsenv`)
- Choose *Load From* and either *Cellview*, *Library*, *Tech Library*, or *File*.

To delete the display settings,

➤ Choose *Delete From* and either *Cellview*, *Library*, or *Tech Library*.

Using the Layout Editor Options Form

The Layout Editor Options form controls command defaults.



Editor Controls

Repeat Commands sets all commands to automatically repeat, if you first select the command and then the object to edit. Does not affect commands if you first select an object, then the command.

Display Reference Point displays an asterisk (*) on the current reference point.

Auto Set Reference Point automatically sets a reference point whenever you enter a new point. You use reference points to measure the distance between points you enter.

Recursion Check turns recursion checking on. Recursion checking prevents the creation of recursive hierarchy (where a cell has an instance of itself at some level) by commands such as *Create Instance*, *Make Cell*, and *Search/Replace*.

Maintain Connections turns on an automatic reconnection mode for pins and paths that have been separated during any editing activity that moves pins.

- with Guided Path works only with *Maintain Connections*. With these two options on, the path reconnects to the pin and follows the minSpacing rules in the technology file for the current layer.

Guided Path Create sets the mode to *Guided* in the Create Path form, causing the path to follow the minSpacing rules defined in the technology file for the current layer.

Conic Sides sets the number of segments to use when you convert conics to polygons. The maximum number of sides is 2,047.

Gravity Controls

Gravity On sets the drawing cursor to automatically snap to objects in the cellview.

Types controls the type of objects or part of an object to which the cursor snaps.

all turns all the *Types* buttons on.

none turns all the *Types* buttons off.

Aperture sets the snap distance between the cursor and an object. When the cursor reaches the distance specified, the cursor snaps to the object.

Depth sets how many levels down in the hierarchy shapes are checked for cursor snapping.

Bounce X and **Bounce Y** set the gravity offset in user units. Use this feature when you want the cursor to snap a fixed distance from objects. You can set a positive or negative bounce value. A positive bounce value makes the cursor snap the set distance outside the selected object. A negative bounce value makes the cursor snap the set distance inside the selected object.

File specifies the file you want to save to or load from.

Save To saves the current setting from the Layout Editor Options and Display Options forms to the file specified in the *File* field.

Load From sets the Layout Editor Options and Display Options forms settings to the values saved in the specified file.

Using Editor Controls

In the *Editor Controls* area of the form, you choose what options you want to set.

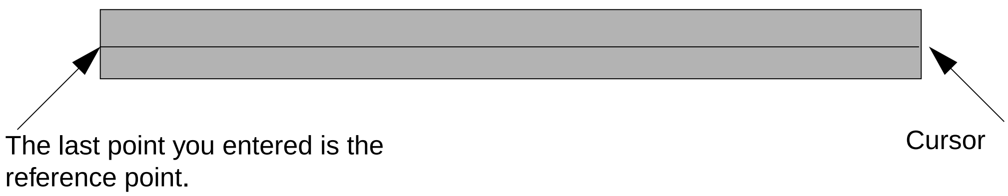
Editor Controls	**Results**
Repeat Commands	Turns the command repeat mode on or off.
Display Reference Point	Sets whether or not to display reference points.
Auto Set Reference Point	Sets whether or not to set reference points automatically.
Recursion Check	Turns recursion checking on or off.
Maintain Connections	Sets whether or not to reconnect paths to pins that are moved.
- with Guided Path	Sets *Maintain Connections* to use minSpacing rules.
Guided Path Create	Sets the *Path* command to use minSpacing rules to lay out paths.
Conic Sides	Sets the number of conic sides.

Using Reference Points

Reference points help you measure the exact distance between objects or line segments as you enter points. Each point you click when creating objects; editing objects; or placing a contact, label, or cell instance becomes the reference point for the next point you enter.

To measure objects from the reference point, look at the coordinate readout at the top of the layout cellview.

dX and dY Display	Results
dX: 11.5 **dY: 0.0**	Shows the difference between the reference point coordinates and cursor location.
Dist: 11.50	The distance between the reference point and the cursor.



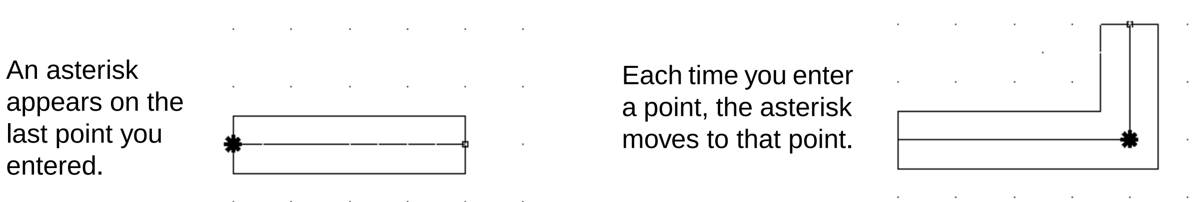
Displaying Reference Points

You can set the Virtuoso layout editor to display the reference point as a bold asterisk in your layout window.

To display a reference point,

1. Choose *Options – Layout Editor*.
2. The Layout Editor Options form appears. Set *Display Reference Point* on.
3. Click *OK*.

   A bold asterisk (⁎) appears on the reference point.

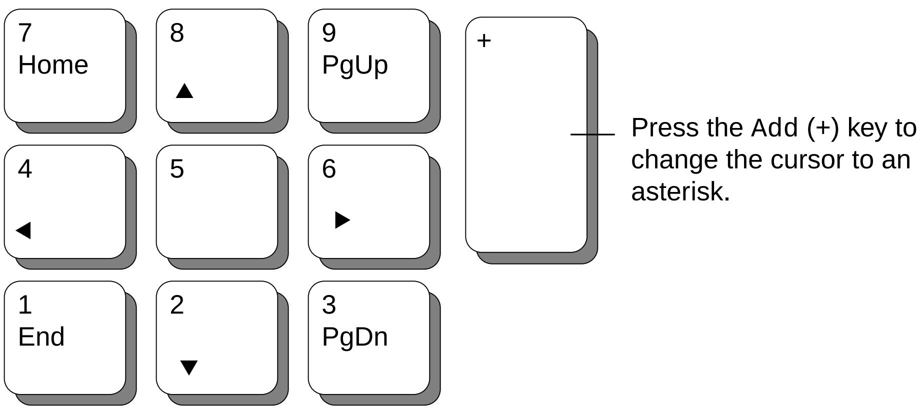


Setting the Reference Point Manually

You can set the reference point to a particular point without using menu commands.

To set a reference point manually,

1. Press the Add (+) key on the keypad.



The cursor changes to an asterisk (*).

2. Click where you want to set the reference point.

   The point you clicked is the new reference point.

Setting a Stationary Reference Point

You can set the layout editor so that the reference point does not change. This way you can create or place several objects using a single reference point.

To set a stationary reference point,

1. Set a reference point by entering it while creating objects, or by using the Add (+) key.
2. Choose *Options – Layout Editor*.
3. In the Layout Editor Options form, set *Auto Set Reference Point* off.
4. Click *OK*.

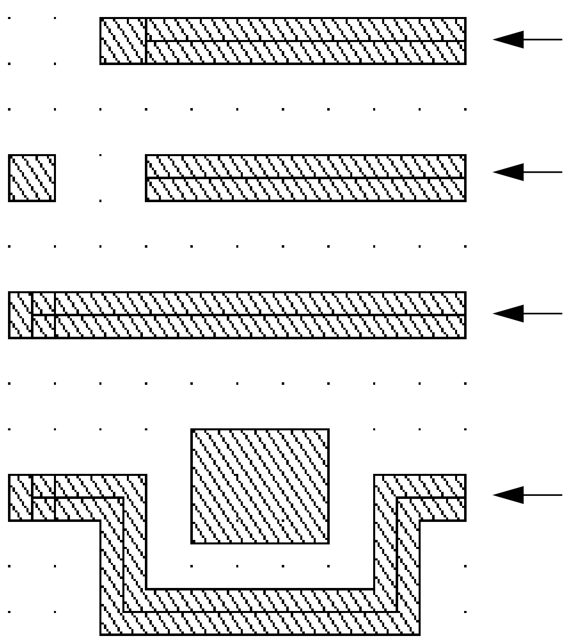
   The reference point you entered in step 1 remains the reference point. It does not change when you enter new points.

Preserving Pin-Path Connections

Maintain Connections

To set the Virtuoso layout editor to automatically reconnect paths to pin connections that are broken during any pin editing activity, use the *Maintain Connections* option in the Layout Editor Options form.

To set the layout editor to follow the minSpacing rules defined in the technology file to automatically reconnect paths to pin connections that are broken during any editing activity, use the *Maintain Connections with Guided Path* option in the Layout Editor Options form.



Pin-path connection.

Move the pin with *Maintain Connections* off. The path remains the same.

Move the pin with *Maintain Connections* on. The path is modified to remain connected.

Move the pin with *Maintain Connections* and *Guided Path* both on. The path is modified to remain connected to the pin and avoid obstacles using the minSpacing rules.

Maintain Connections is a background process that monitors the connectivity of pins that abut or overlap path ends on the same layer. When any editing activity causes the pin to move away from the path end, *Maintain Connections* automatically modifies the path so that it remains connected to the pin.

Note: The modified path might cause DRC (Design Rules Check) violations. To have the new path avoid DRC violations and follow the minSpacing rules, use *Maintain Connections with Guided Path*.

Editing activities that can cause a pin to change location include

- Moving or stretching a pin
- Moving or stretching an instance containing a pin
- Editing pin properties (changing the pin shape or pin location)
- Changing pcell parameters

The *Maintain Connections* option operates on pins and paths with these characteristics:

- The pin and the path are on the same or equivalent layer and have the same layer purpose.
- The path end overlaps or abuts the pin.
- The pin is rectangular, either shape or symbolic pins.
- If the pin is an instance, it must be placed at the same level as the path.
- If the pin and path are ROD objects, they cannot be aligned.

Starting Maintain Connections

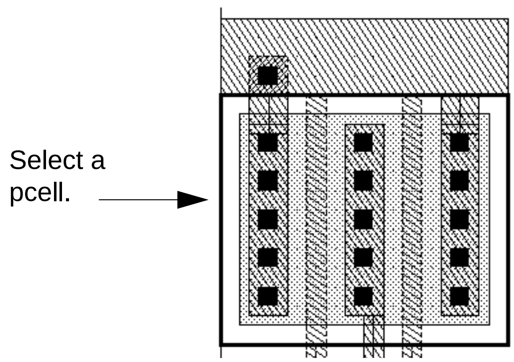
To start *Maintain Connections*, do one of the following:

- Press F6 to toggle *Maintain Connections* on and off. A message in the CIW lets you know the status of *Maintain Connections*.
- In the Layout Editor Options form, set *Maintain Connections* on.

Using Maintain Connections

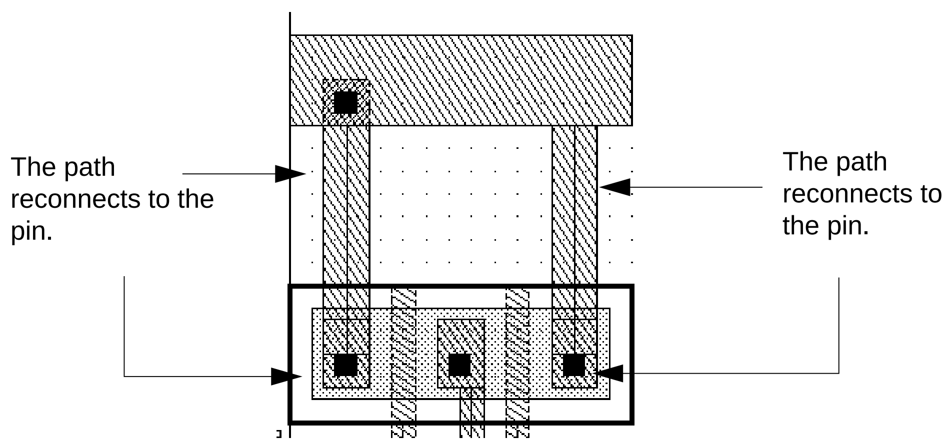
One way to see the effects of *Maintain Connections* is to change the parameters of an instance that contains pin-path connections on the same layer. After you change the parameters, the instance changes shape to the new value, and the path and pin retain their connectivity.

1. Select a pcell containing pin-path connections.



2. Choose *Edit – Properties*.
3. Click on *Parameter* and change the size of the pcell.
4. Click on *Apply*.

   The instance changes shape to the new width value, and the paths reconnect to the pins.



Maintain Connections with Guided Path Overview

To set the layout editor to follow the minSpacing rules defined in the technology file and to automatically reconnect paths to pins that are broken during any editing activity, use *Maintain Connections with Guided Path*. Having the path use the minSpacing rules is helpful because DRC violations are avoided. You activate the *Maintain Connections with Guided Path* option in the Layout Editor Options form.

Editing activities that can cause a pin-path to disconnect and the characteristics of the pin and path are the same for both *Maintain Connections with Guided Path* and for *Maintain Connections*.

Starting Maintain Connections with Guided Path

To start *Maintain Connections with Guided Path*, do one of the following:

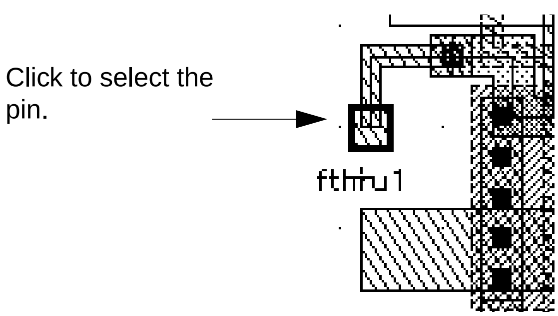
- Press F6 and F7 to toggle these options. Check the CIW for messages indicating whether the options are on or off.
- In the Layout Editor Options form, turn on *Maintain Connections with Guided Path*.

Note: The option, – *with Guided Path*, has no effect unless *Maintain Connections* is on.

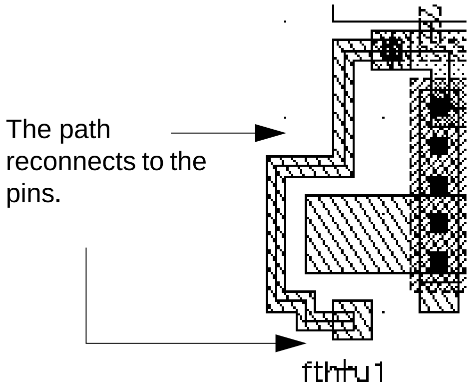
Using Maintain Connections with Guided Path

One way to see the effects of *Maintain Connections with Guided Path* is to use the *Move* command.

1. Choose *Edit – Move*.
2. Select the pin you want to move.



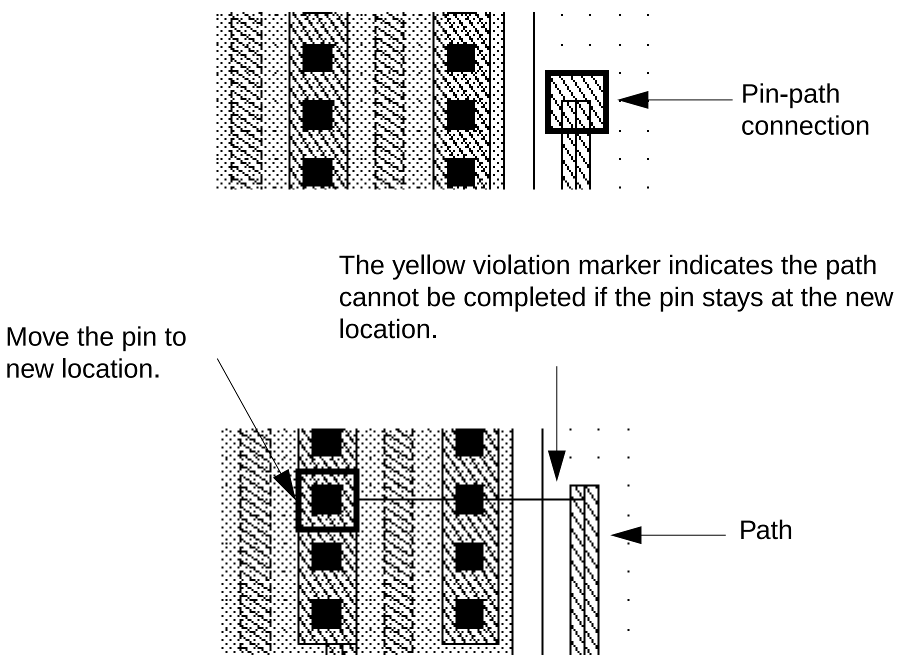
3. Click at the new location for the pin.



Violation Markers

If *Maintain Connections with Guided Path* cannot find a route that follows the minSpacing rules during the pin-path reconnecting process, a yellow violation marker appears in the design and the pin-path connection is not completed.

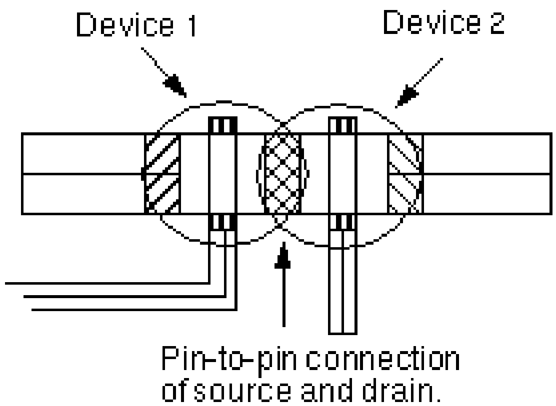
When this happens, you must move either the pin or the obstacle until the pin-path connection can be made without violation markers.



Limitations

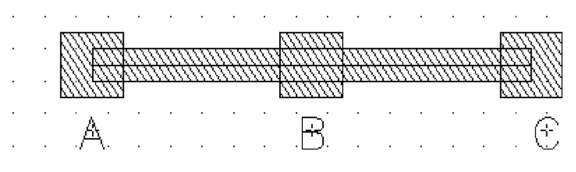
Maintain Connections does not retain connectivity in the following relationships:

- Pin-to-pin connections



- Pins placed in the middle of paths

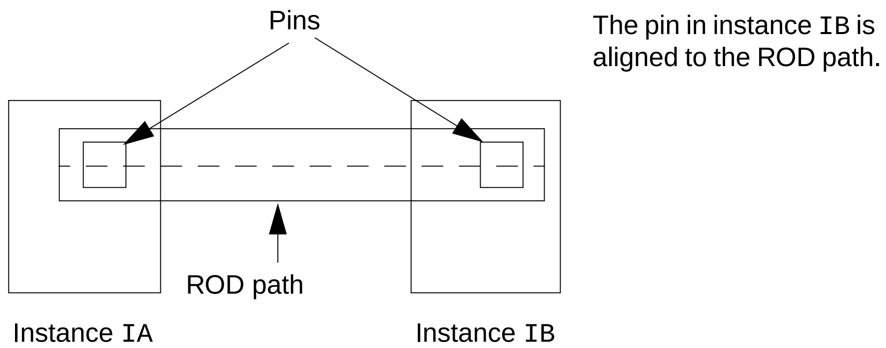
Pins A and C remain connected
to the path, but pin B might not.



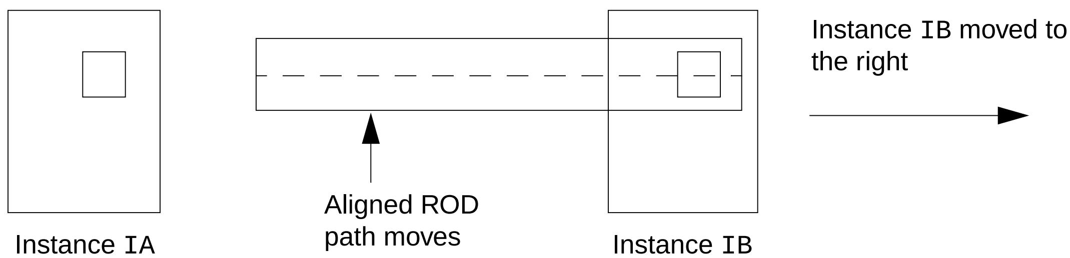
- Aligned ROD objects

  ROD alignments take precedence over maintaining connections.

Before a move with *Maintain Connections*:



After a move with *Maintain Connections*:



Moving instance `IB` causes the whole ROD path to move. The aligned path does not stretch, so moving instance `IB` breaks the electrical connection between the two pins.

Using the Gravity Controls

In the Gravity Controls form, you choose how you want gravity to operate.

Gravity Controls	Results
Gravity On	Turn gravity on or off.
Types	Set what points the cursor snaps to.
Aperture	Set the distance the cursor must be within before gravity snaps the cursor.
Depth	Set how far down in the hierarchy to look for objects to snap to.

Bounce X and Y	Set an offset distance for snapping.

Setting Gravity On and Off

You can use gravity to set the cursor (the small square) to snap to objects as you create them. This helps you create the edge of one object flush with another.

To set gravity on or off, do one of the following:

- Press g to toggle gravity on and off.
- Set *Gravity On* in the Layout Editor Options form and click *OK*.

The cursor does not snap to objects on layers set to be unselectable in the Layer Selection Window (LSW).

Setting the Gravity Distance (Aperture)

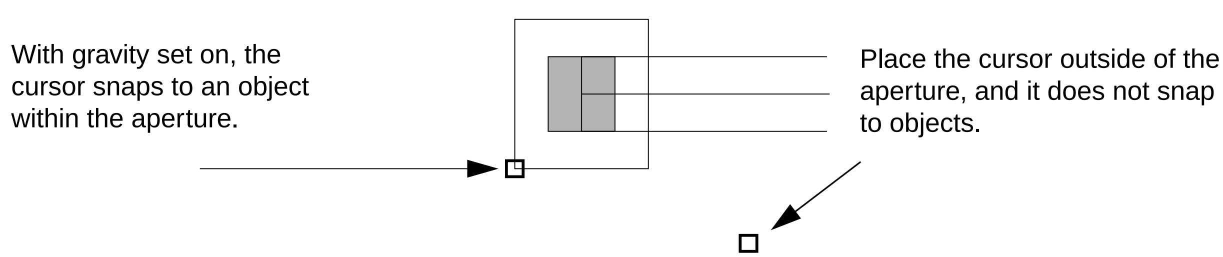
Aperture is a specific area around the cursor. When objects appear within that area, the cursor snaps to them if those objects are assigned as gravity types in the *Types* field of the Layout Editor form. By default, the cursor must be within 0.3 user units of an object before gravity snaps the cursor to the object. You can change this default.

To set the gravity distance,

1. Choose *Options – Layout Editor*.

   The Layout Editor Options form appears.
2. In the *Aperture* field, type a value.

   This is the distance at which you want the cursor to snap to objects. The value is in user units (usually microns).
3. Click *OK*.



Snapping the Cursor to Specific Objects

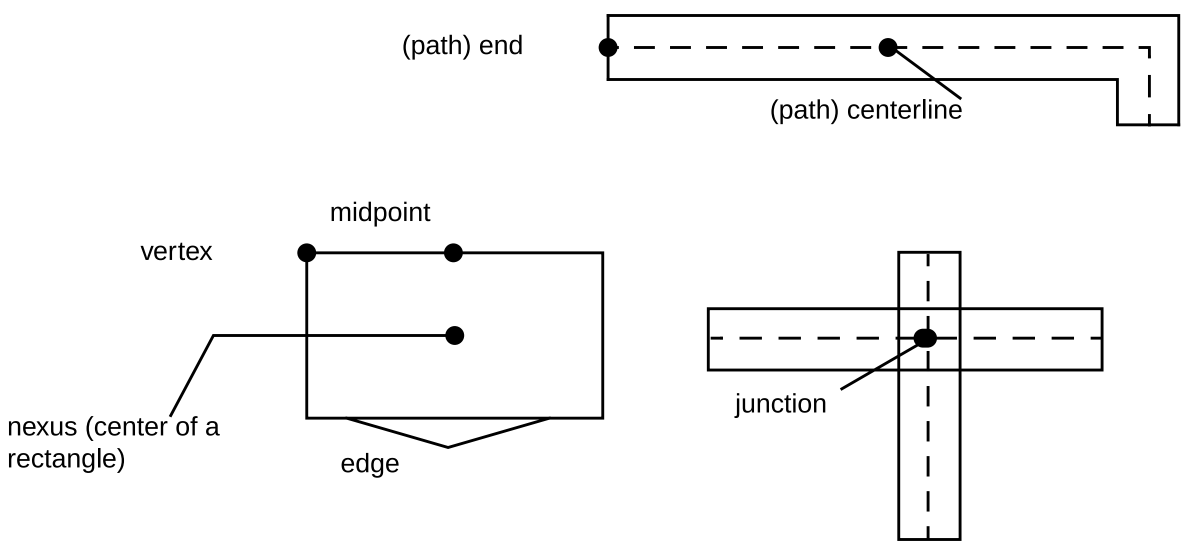
You can set the cursor to snap to specific objects.

To snap the cursor to objects,

1. Choose *Options – Layout Editor*.

   The Layout Editor Options form appears.

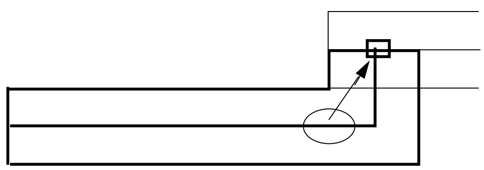
2. In the *Types* section, turn on the types you want the cursor to snap to.



   Pins snap to midpoints, except for rectangle pins. Rectangle pins snap to the center of the edge if *Access Direction* is on in the Create Pin form.

3. Click *OK*.

Here is what happens when you create a path with gravity set to snap to path centerlines:



Move the cursor to a point near a path centerline. Both the cursor and the path you are creating snap to the centerline.

Note: The cursor does not snap to an object if doing so conflicts with the snap mode that controls the shape of the object.

Gravity Types in Order of Precedence

The cursor snaps to objects that appear within the aperture in the following order of precedence, from highest to lowest:

- Pin
- Vertex and end
- Centerline and edge

These conditions affect gravity precedence:

- Instances are examined for the bounding box as well as for pins at depth +1 level down the hierarchy.
- Only edges of rectangular pins with access direction set are considered pins, otherwise the edge is treated just as an edge.
- *End* applies only to shapes with width, like a path. *Vertex* is enhanced for shapes that have width. If the point is closer than a width away, the point snaps to the vertex or end.
- When none of these conditions applies, the cursor snaps to the closest edge; *centerline* is considered a special path edge.

Snapping the Cursor to Objects on Specific Layers

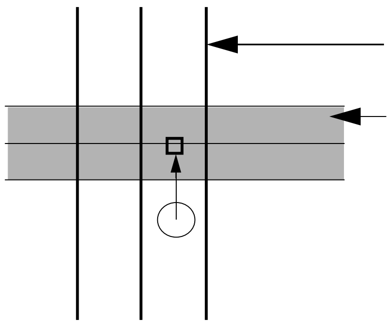
The cursor snaps only to objects on layers set to be selectable.

To toggle whether a layer is selectable or unselectable,

➤ Click right on that layer in the Layer Selection Window (LSW).

The names of any unselectable layers are shaded in the LSW.

Here is what happens when you set gravity to snap to path centerlines and set only the *metal1* layer to be selectable.



Path on *metal2*.

Path on *metal1*.

The cursor snaps to the centerline of a path created on *metal1*, not to the centerline of the path on *metal2*.

Snapping the Cursor to Objects in Instances

You often create objects in the current cellview that align with objects in cell instances. To help you align what you create, you can set the cursor so it snaps to objects inside of instances.

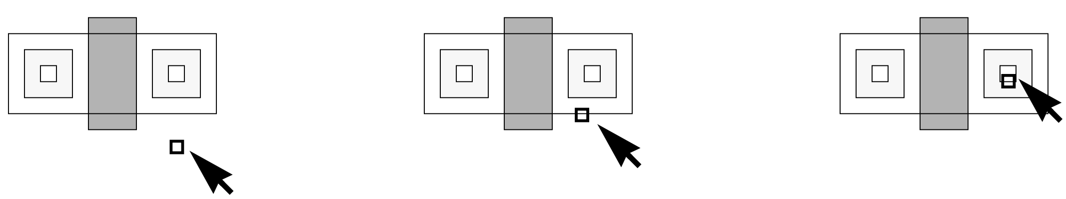
To snap the cursor to objects in instances,

1. Choose *Options – Layout Editor*.

   The Layout Editor Options form appears.

2. Type a value in the *Depth* field.

   Type any integer from 1 to 32 to let the cursor snap to objects from 1 to 32 levels deeper in the design hierarchy.



Depth = 0
Cursor snaps to the transistor cell outline or bounding box (dotted line).

Depth = 1
Cursor snaps to polygons inside the transistor cell.

Depth = 2
Cursor snaps to polygons inside the contact cell, which is inside the transistor cell.

3. Click *OK*.

Setting a Snap Offset (Bounce)

Many times, you want to create objects that must be a certain distance from other objects. You can set an offset distance to make the cursor snap that distance away from other objects. You can set a positive or negative bounce value. A positive bounce value makes the cursor snap the set distance outside the selected object. A negative bounce value makes the cursor snap the set distance inside the selected object.

Note: Bounce works only if the environment variable `applyBounce` is set to `T`. See the Environment Variable Table in the *Custom Layout SKILL Functions Reference* manual for more information.

To set snap offset distance,

1. Choose *Options – Layout Editor*.

The Layout Editor Options form appears.

2. Type an offset value in the *Bounce* fields.

3. Click *OK*.

5

Working with Design Hierarchy

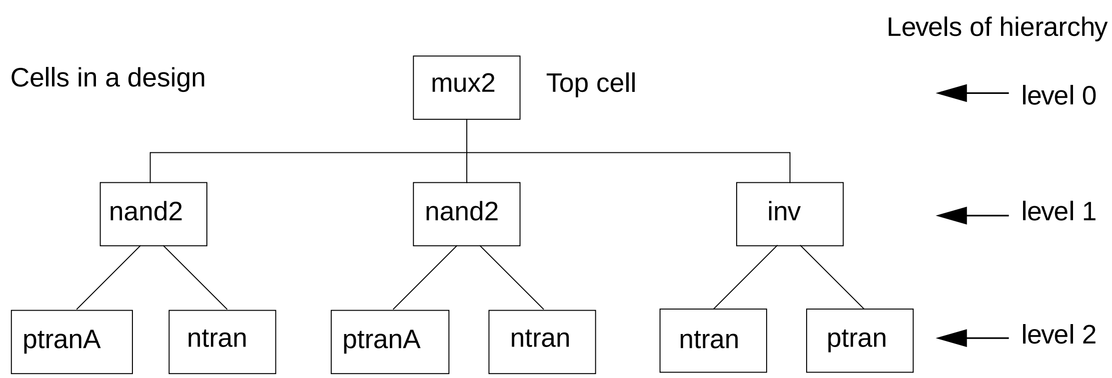
This chapter contains these topics:

About Hierarchical Designs

With the Virtuoso® layout editor, you can create a hierarchical design by placing instances of other references to cells inside other cells. The cell at the top includes the whole design. The cells at the bottom contain the smallest parts of the design.

In the design shown below, for example, instances of the ntran and ptran (N- and P-transistor) cells are placed inside the inv (inverter) cell, which is inside the mux2 cell.

The Cadence® software numbers levels of hierarchy from top to bottom, with the top cell having the lowest number. The top cell is the cellview you are currently viewing in a design window, unless you have descended into the hierarchy or are using *Edit In Place*.

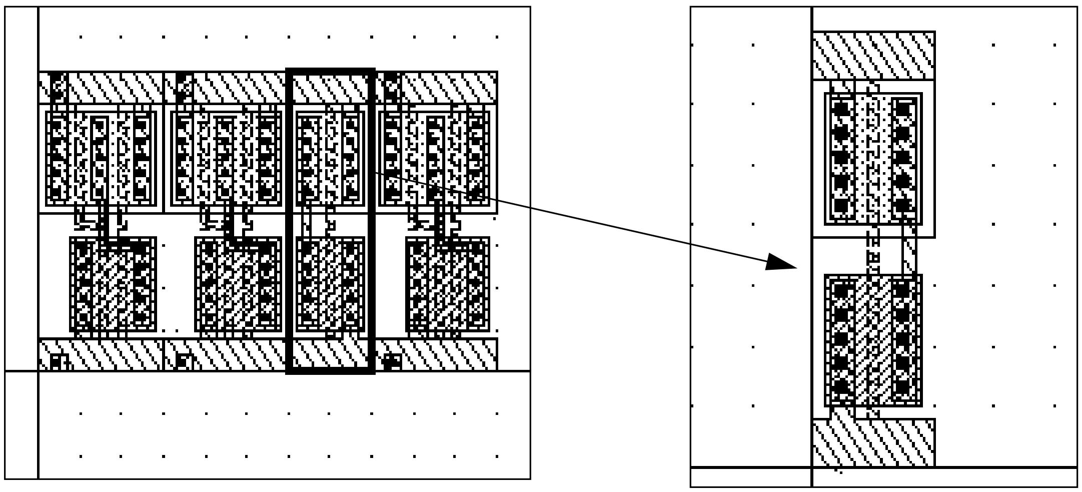


In the above figure, when you view mux2, it is level 0, as shown. When you view the inv cell, it is considered level 1, and the ntran and ptran instances are considered level 2.

Master Cells and Cell Instances

A master cell is any layout cell you have placed in another cell. The placed reference of the cell is called an instance.

For example, the mux2 layout cell contains an instance of the inv layout cell.



The mux2 layout cell contains an instance of the inv layout cell.

The inv layout cell.

If you edit a master cell, all instances of that cell are changed. For example, if you stretch a polygon in the master inv layout cell, the polygon is stretched in all instances of inv.

A Sample Design and Tutorial

The sample design in the *Cell Design Tutorial* is a hierarchical design. Many of the sample cells used in this document are from the tutorial.

The tutorial libraries are stored in your Cadence software directory, under

`your_install_dir/tools/dfII/samples/tutorials/le/cell_design`

Checking for Instance Recursion

You can prevent commands such as *Create Instance*, *Make Cell*, and *Search/Replace* from creating an instance of itself in a cell by turning recursion checking on in the Editor Options form.

To automatically check for recursion when placing instances,

1. Choose *Options – Layout Editor* [`Shift-e`].

   The Layout Editor Options form appears.

2. Click on *Recursion Check*.
3. Click *OK*.

Changing the Master Cell Origin

Place instances by clicking where you want the instance origin. To change how the instances are placed, change the origin of the master cell.

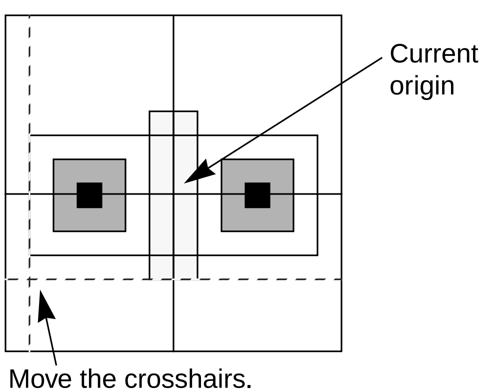
Caution

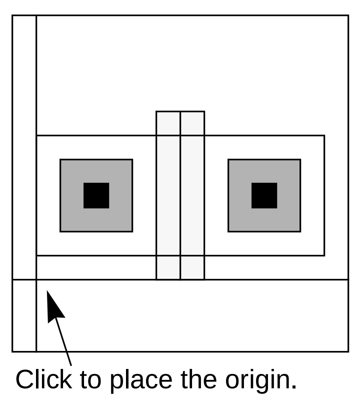
Moving the master cell origin changes the location of all placed instances of that cell.

To change the origin of the master cell,

1. Choose *Edit – Other – Move Origin*.
2. Move the cursor where you want the new origin.

   Two crosshairs showing the new X and Y axes follow the cursor.
3. Click where you want the origin.



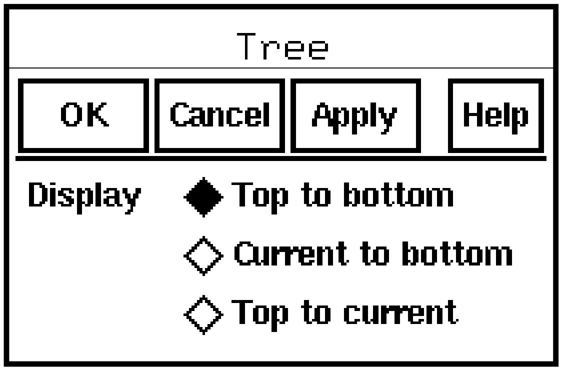


Listing the Hierarchy

Using the Tree Form

The *Tree* command displays a list showing the hierarchy of cell instances in the current cellview.

Note: The *Display Level* settings in the Display Options form limit the number of levels reported. This means if *Display Level* is set to 0 through 3, even though you set *Top to bottom* on the Tree form, the output will be for levels 0 through 3 only.



Display sets how much of the hierarchy is listed.

Top to bottom lists the entire hierarchy of instances in this cellview.

Current to bottom lists the entire hierarchy of instances inside the cell you are editing in place.

Top to current lists the hierarchy above the cell you are editing in place.

Viewing the Results of the Tree Command

To list the hierarchy of all the instances in the current cellview,

1. Choose *Design – Hierarchy – Tree* [`Shift-t`].

   The Tree form appears.

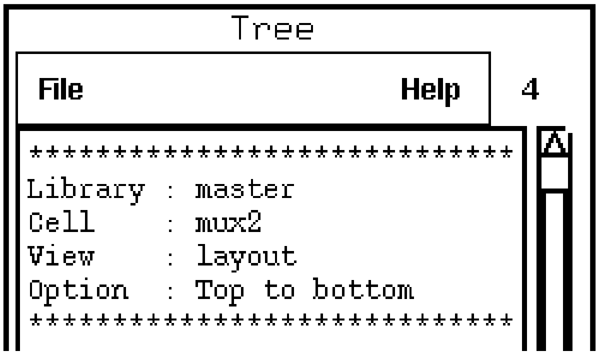
2. Set *Display* to one of the following:

   - To list the entire hierarchy in the cellview, turn *Top to bottom* on
   - To list the hierarchy inside the cell you are editing in place, turn *Current to bottom* on
   - To list the hierarchy above the cell you are editing in place, turn *Top to current* on

   Note: The *Display Level* settings in the Display Options form limit the number of levels reported by this command.

3. Click *OK*.

A text window listing the hierarchy appears.



4. Choose *File – Close Window* to close the window.

Tree Window File Menu

The *File* menu contains commands that let you open, save, search, and close the text window.

Open loads the contents of a text window previously saved with *Save As*.

Save As saves the data in this window to a text file.

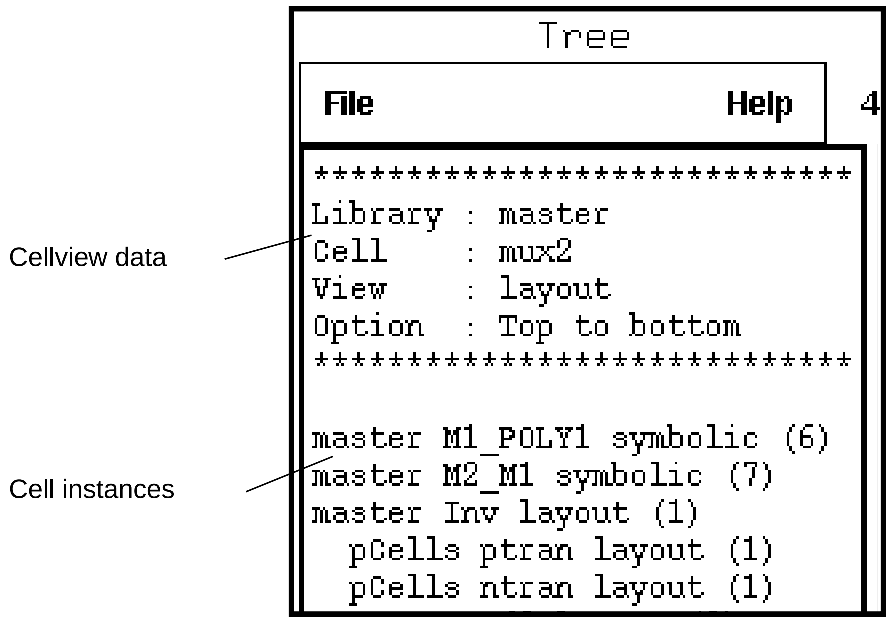
Auto Update On and shaded command names are not used by text windows opened with the *Tree* command.

Search searches through the text in this window for a text string.

Close Window closes this text window.

Tree Window Cellview Data

The cellview data displays information about the cellview.



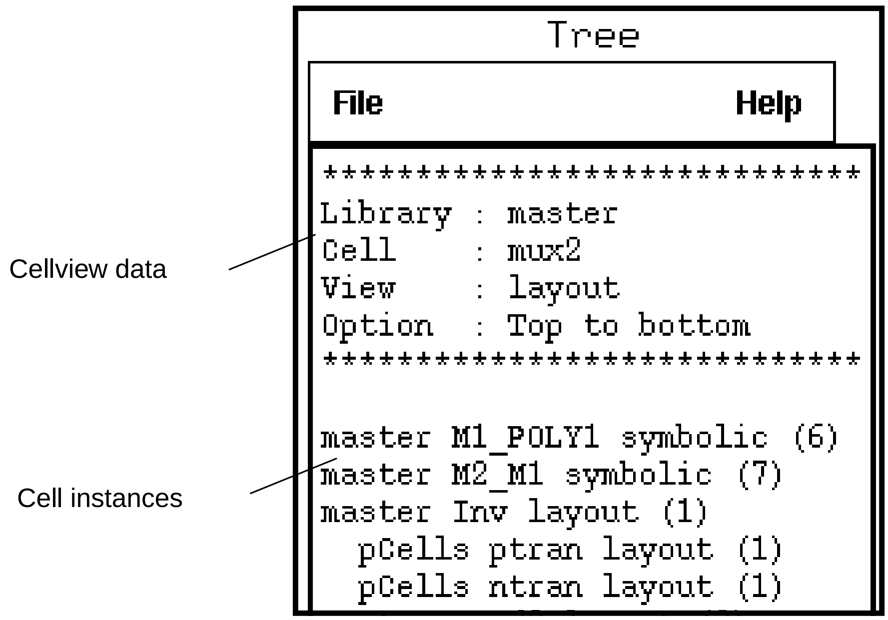
The *Library*, *Cell*, and *View* name fields display the cellview whose contents are listed.

Option shows which *Display* setting you chose in the Tree form.

Tree Window Cell Instances

The cell instances information lists the hierarchy of cell instances in this cellview or the instances above or below the edit-in-place cell. The number in parentheses represents how many times that instance appears in the design.

Indents show that instances of the indented cells are inside the instance listed above. For example, ptran is inside the Inv cell.



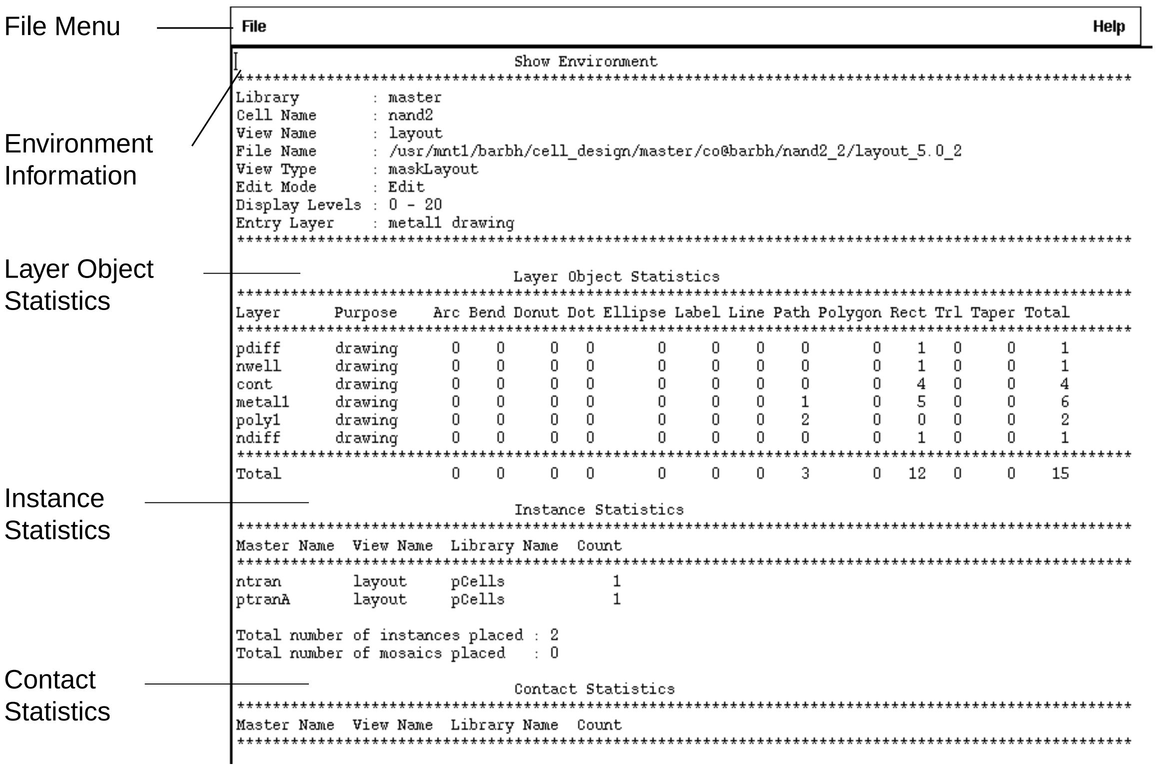
Listing the Data In Cellviews

Listing the Contents of a Cellview

To display a summary of the contents of this cellview,

1. Choose *Design – Summary*.

A window containing the summary appears.




```
                              Show Environment
****************************************************************************************************
Library        : master
Cell Name      : nand2
View Name      : layout
File Name      : /usr/mnt1/barbh/cell_design/master/co@barbh/nand2_2/layout_5.0_2
View Type      : maskLayout
Edit Mode      : Edit
Display Levels : 0 - 20
Entry Layer    : metal1 drawing
****************************************************************************************************

                              Layer Object Statistics
****************************************************************************************************
Layer      Purpose    Arc Bend Donut Dot Ellipse Label Line Path Polygon Rect Trl Taper Total
****************************************************************************************************
pdiff      drawing      0    0     0   0       0     0    0    0       0    1   0     0     1
nwell      drawing      0    0     0   0       0     0    0    0       0    1   0     0     1
cont       drawing      0    0     0   0       0     0    0    0       0    4   0     0     4
metal1     drawing      0    0     0   0       0     0    0    1       0    5   0     0     6
poly1      drawing      0    0     0   0       0     0    0    2       0    0   0     0     2
ndiff      drawing      0    0     0   0       0     0    0    0       0    1   0     0     1
****************************************************************************************************
Total                   0    0     0   0       0     0    0    3       0   12   0     0    15

                              Instance Statistics
****************************************************************************************************
Master Name  View Name  Library Name  Count
****************************************************************************************************
ntran        layout     pCells            1
ptranA       layout     pCells            1

Total number of instances placed : 2
Total number of mosaics placed   : 0

                              Contact Statistics
****************************************************************************************************
Master Name  View Name  Library Name  Count
****************************************************************************************************
```

2. Choose *File – Close Window* to close the window.

Summary File Menu

The *File* menu lets you manipulate this text window and its contents.

Open loads the contents of a text window previously saved with *Save As*.

Save As saves the data in this window to a text file.

Auto Update On and shaded command names are not used by text windows opened with the *Summary* command.

Search searches through the text in this window for a text string.

Close Window closes this text window.

Show Environment

The *Show Environment* section shows information about the cellview and its edit mode, display levels, and current entry layer (entry layer).

```
                               Show Environment
****************************************************************
Library        : master
Cell Name      : nand2
View Name      : layout
File Name      : /usr/mnt1/barbh/cell_design/master/co@barbh/nar
View Type      : maskLayout
Edit Mode      : Edit
Display Levels : 0 - 20
Entry Layer    : metal1 drawing
****************************************************************
```

Layer Object Statistics

The *Layer Object Statistics* section lists the shapes created in the cellview and the layer on which each is present.

```
                               Layer Object Statistics
****************************************************************
Layer      Purpose    Arc Bend Donut Dot Ellipse Label Line Patl
****************************************************************
poly1      drawing      0    0     0   0       0     0    0    2
cont       drawing      0    0     0   0       0     0    0    0
metal1     drawing      0    0     0   0       0     0    0    1
ndiff      drawing      0    0     0   0       0     0    0    0
pdiff      drawing      0    0     0   0       0     0    0    0
nwell      drawing      0    0     0   0       0     0    0    0
****************************************************************
Total                   0    0     0   0       0     0    0    3
```

Instance Statistics

The *Instance Statistics* section shows the names of the cell instances and arrays of instances (mosaics) that have been placed in this cellview.

```
                              Instance Statistics
***************************************************************
Master Name  View Name  Library Name  Count
***************************************************************
ntran        layout     pCells            1
ptranA       layout     pCells            1

Total number of instances placed : 2
Total number of mosaics placed   : 0
```

Contact Statistics

The *Contact Statistics* section shows the names of the contact cells that have been placed in this cellview.

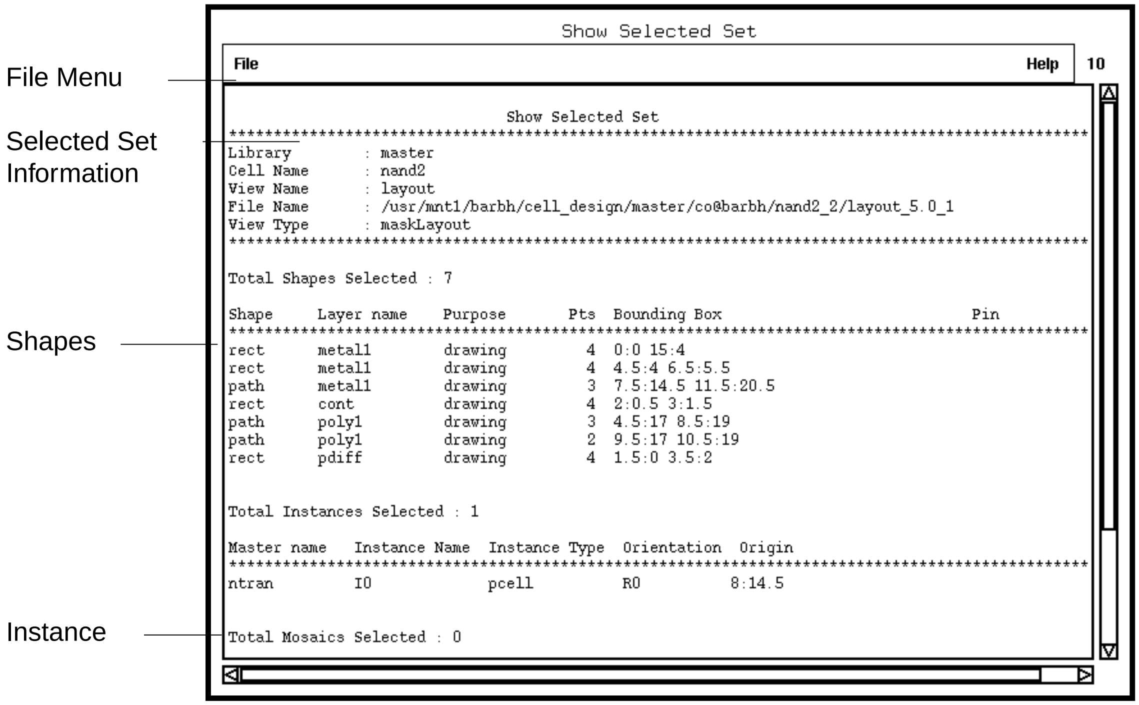
```
                              Contact Statistics
***************************************************************
Master Name  View Name  Library Name  Count
***************************************************************

Total number of contacts placed  : 0
```

Listing All Selected Objects

To display information about objects,

1. Select the objects.

2. Choose *Window – Show Selected Set.*



Show Selected Set File Menu

The *File* menu lets you manipulate this text window and its contents.

Open loads the contents of a text window you previously saved with *Save As*.

Save As saves the data in this window to a text file.

Auto Update On and shaded command names are not used by text windows opened with the *Show Selected Set* command.

Search searches through the text in this window for a text string.

Close Window closes this text window.

Show Selected Set

The *Show Selected Set* section shows information about the cellview.

```
                                Show Selected Set
*****************************************************************
Library         : master
Cell Name       : nand2
View Name       : layout
File Name       : /usr/mnt1/barbh/cell_design/master/co@barbh/na
View Type       : maskLayout
*****************************************************************

Total Shapes Selected : 7
```

Shapes

The *Total Shapes Selected* section lists the selected objects, their layers, the number of points used to create them, and the coordinates of the bounding box that surrounds each object.

```
Total Shapes Selected : 7

Shape     Layer name    Purpose       Pts  Bounding Box
*****************************************************************
rect      metal1        drawing         4  0:0 15:4
rect      metal1        drawing         4  4.5:4 6.5:5.5
path      metal1        drawing         3  7.5:14.5 11.5:20.5
rect      cont          drawing         4  2:0.5 3:1.5
path      poly1         drawing         3  4.5:17 8.5:19
path      poly1         drawing         2  9.5:17 10.5:19
rect      pdiff         drawing         4  1.5:0 3.5:2
```

Instances

The *Total Instances Selected* section lists all selected cell instances, including each instance name, placement orientation, and the coordinates of its origin.

If any arrays of cell instances are selected, they are listed under *Total Mosaics Selected*.

```
Total Instances Selected : 1

Master name   Instance Name  Instance Type  Orientation  Origin
***************************************************************
ntran         I0             pcell          R0           8:14.5


Total Mosaics Selected : 0
```

Editing Instances

Editing a Cell in Place

You can edit the master cell of an instance while viewing the instance in the current cellview. This is called editing in place.

Note: You cannot edit parameterized cells (pcells) in place because the layout of a pcell instance does not necessarily match the layout of the master pcell.

To edit a master cell in place,

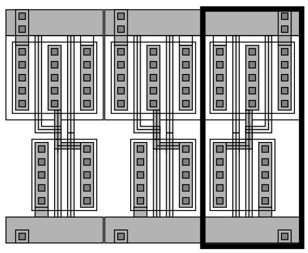
1. Choose *Design – Hierarchy – Edit In Place* [x].

   If you are editing a managed file, you might be asked if you want to check out the file.

2. Click on the instance whose cell you want to edit.

   The banner at the top of the window changes to show that you are now editing the master cell for that instance.

3. Choose *Window – Redraw* to see a border outlining the edit-in-place cell.



Redraw the cellview to see a border outlining the instance you are editing in place.

4. Choose *Design – Hierarchy – Return* to stop editing in place.

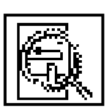
   If you are returning from a managed file, you might be asked if you want to check in the file.

Note: When you are done editing with *Edit In Place*, and as you return up the hierarchy, each level you pass is accessible in edit mode if you started in a cell that is in edit mode. If a lower-level cell was originally in read-only mode, it will return to read only once you return up the hierarchy. If you have changed anything in a lower cell, you will be prompted to save that master cell.

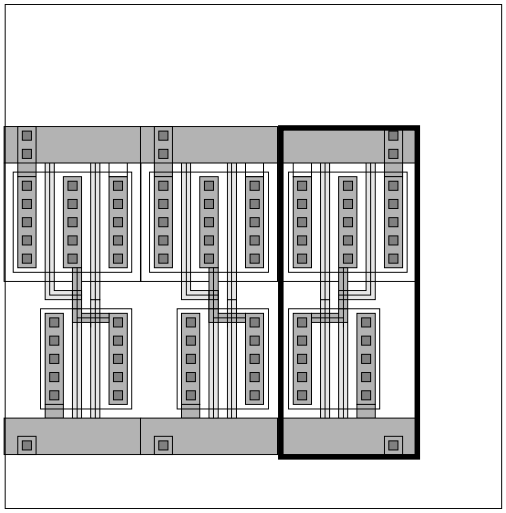
Fitting the Edit-in-Place Cell in the Window

To fit the contents of the edit-in-place cell into the window,

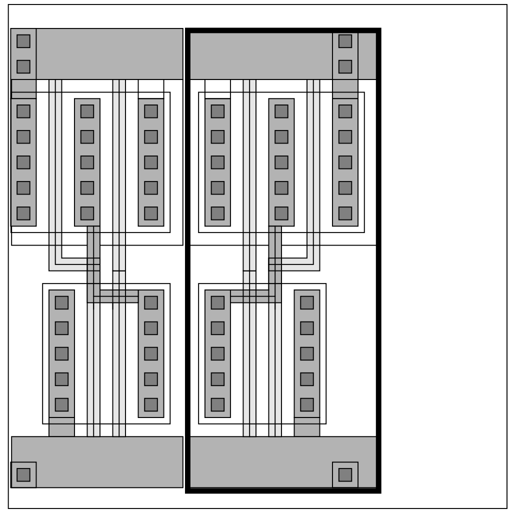
➤ Do one of the following:

- Choose *Window – Fit Edit*.
- Press `Control-x`.
-  Click on the *Fit Edit* icon in the icon menu.

The window is redrawn so the edit-in-place cell fills it. You can see a portion of the surrounding design.



Before *Fit Edit*



After *Fit Edit*

Displaying Only the Edit-in-Place Cell

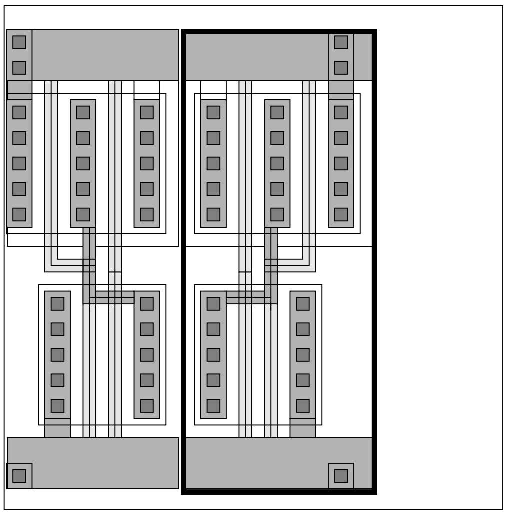
To set the cellview so that only the cell you are editing in place appears,

1. Choose *Options – Display* [e].

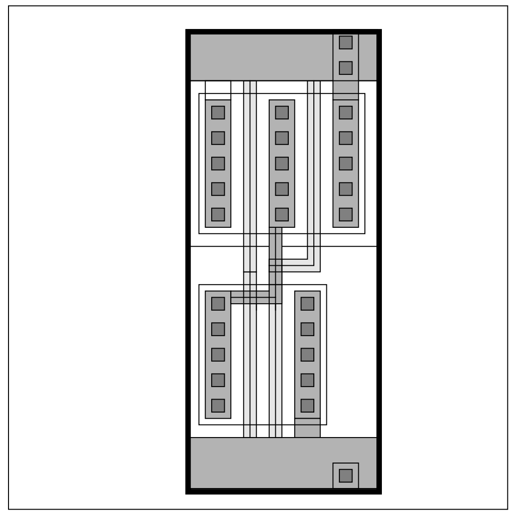
   The Display Options form appears.

2. Set *EIP Surround* off.
3. Click *OK*.

The window is redrawn. Only the instance whose master you are editing in place appears.



With *EIP Surround* set on



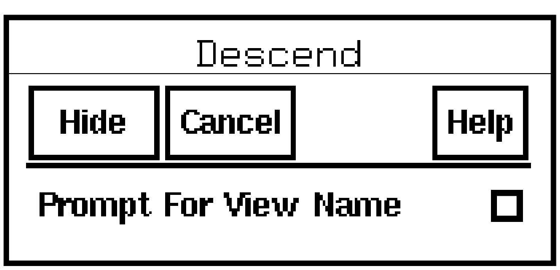
With *EIP Surround* set off

Using the Descend Form

The *Descend* command lets you descend into an instance, and the Descend form lets you descend into a nonlayout view.

Note: The layout editor does not use the *Create New Window When Descending* option in the User Preferences form. Selecting or deselecting this option will not affect your work in the layout editor.

About the Descend Form



Prompt For View Name By default, *Descend* descends into the same view name as the current cellview. If this field is on, the (view name) Descend form appears when you click on an instance. You can choose a different view to open. If there is only one view of the master cell, *Descend* does not prompt you for a view name.

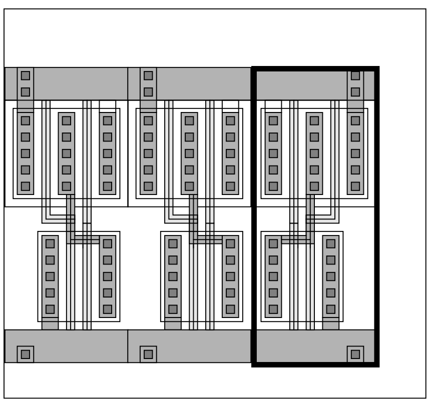
Descending into an Instance

To edit the master cell for any instance in the current window,

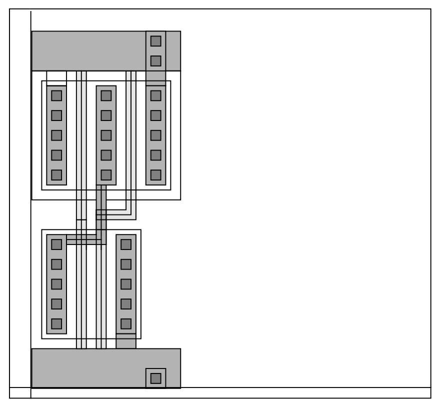
1. Click on the instance whose cell you want to edit.
2. Choose *Design – Hierarchy – Descend* [`Shift-x`].

   If you are descending into a managed file, you might be asked if you want to check out the file.

   The window changes to show the layout view of the instance.



Select the instance you want to open.



Descend opens the layout view of the cell.

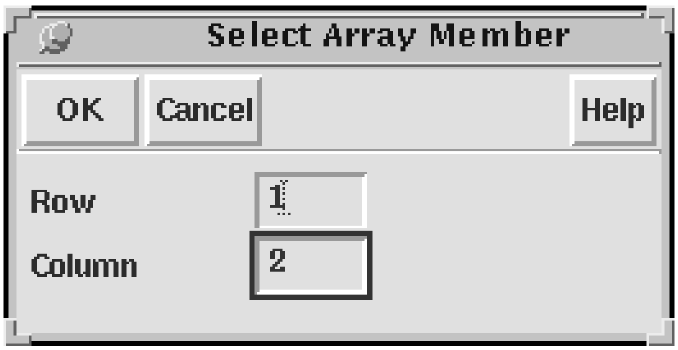
3. If you selected a parameterized cell (pcell) instance, a dialog box appears. Click *OK* to close the box and descend into the pcell.
4. Choose *Design – Hierarchy – Return* to return to the cell you were editing before you opened the instance.

   If you are returning from a managed file, you might be asked if you want to check in the file.

Descending into a Member of an Instance Array

When you want to descend into an instance array, the *Descend* command lets you choose which member of the array you want to descend into.

About the Select Array Member Form



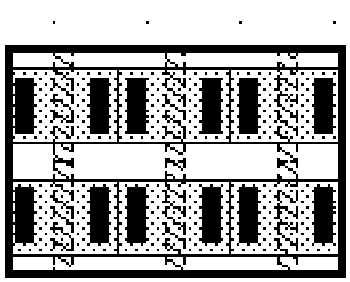
Row indicates the row the array member is in.

Column indicates the column the array member is in.

Using the Select Array Member Form

To edit a member of an instance array,

1. Click on the instance array you want to edit.



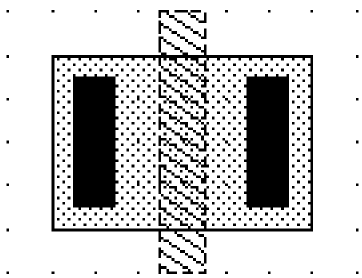
2. Choose *Design – Hierarchy – Descend.*

   The Select Array Member form appears.

3. Enter the row and column information for the array member you want to edit.

4. Click *OK*.

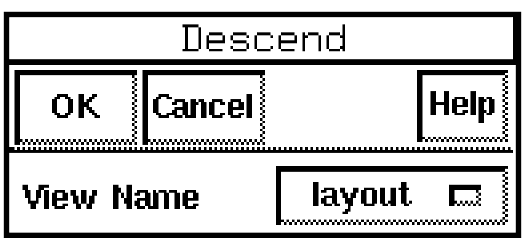
The array member is displayed for editing.



Using the (view name) Descend Form

The (view name) Descend form lists all the available master views for the instance you chose.

About the (view name) Descend Form



View Name lets you choose a different view of the master cell to open with the *Descend* command. For example, you can open a schematic view of an instance rather than its layout view.

Descending into a Nonlayout View of an Instance

To descend into a nonlayout view of an instance,

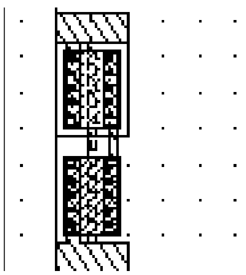
1. Choose *Design – Hierarchy – Descend* `[Shift-x]`.

   The Descend form appears.

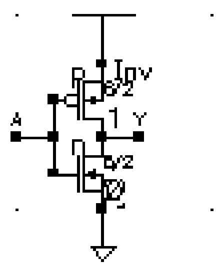
   If you are descending into a managed file, you might be asked if you want to check out the file.

2. Set *Prompt For View Name* on.
3. Click on the instance whose master you want to view.

   The Descend View Name form appears listing all the available master views for the instance you chose.
4. Click to choose the name of the view you want to edit.
5. Click *OK*.



Select an instance of an inverter to descend into.



With *Prompt For View Name* on, you can descend into the schematic view of the inverter.

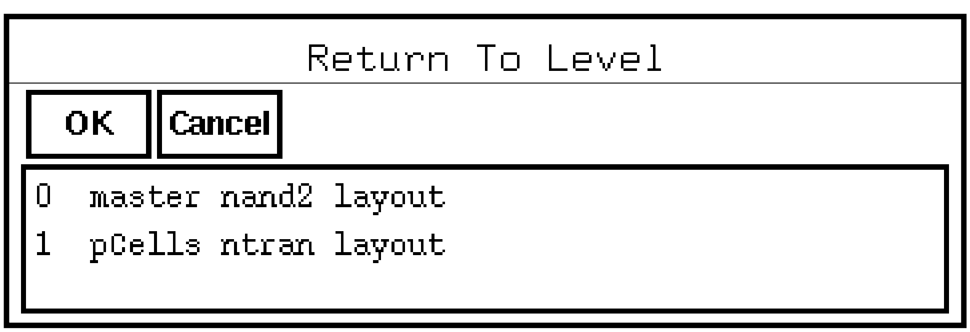
6. Choose *Design – Hierarchy – Return* to return to the cell you were editing before you opened the instance.

   If you are returning from a managed file, you might be asked if you want to check in the file.

Using Return To Level

The *Return To Level* command lets you return to a cellview that is higher in the design hierarchy than the cell you are currently viewing.

About the Return To Level Form



The text box lists the levels above the current one in the hierarchy and the cell associated with each level.

Returning to a Different Hierarchy Level

If you have use *Edit in Place* or *Descend*, you can return to a higher level in the hierarchy.

To return to a specific level,

1. Choose *Design – Hierarchy Return To Level* [b].

   If you are returning from a managed file, you might be asked if you want to check in the file.

2. Click on a cellview name in the Return To Level form.
3. Click *OK*.

   If you are using *Edit In Place*, the window border changes to show the cell you chose.

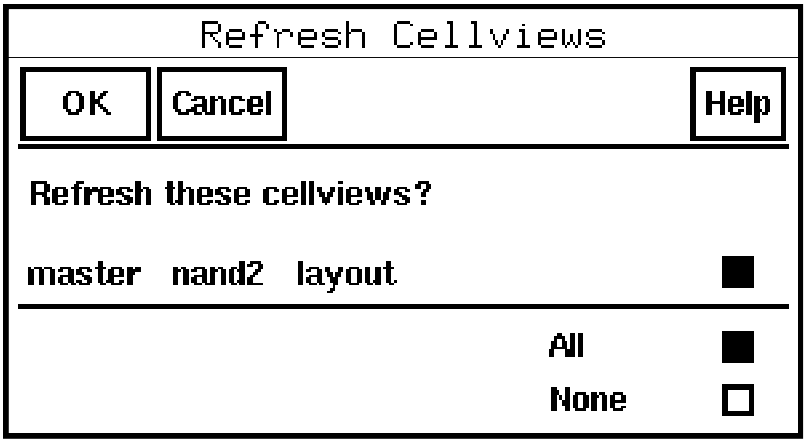
   If you previously used *Descend,* the window contents change to show the cell you chose.

Refreshing Memory after Editing

Using the Refresh Cellviews Form

Another user might edit a master cell whose instances appear in the current cellview. You can update the memory of your computer to reflect the results of such edits with the *Refresh* command.

About the Refresh Cellviews Form



Refresh these cellviews? shows the library, cell, and view name of cells that have been edited since you opened this cellview. You can choose to refresh, in memory, your copies of any of the listed master cells.

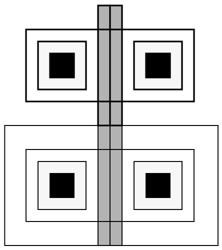
All or **None** lets you choose to update memory for all or none of the listed cellviews.

Refreshing Memory from Other Users' Edits

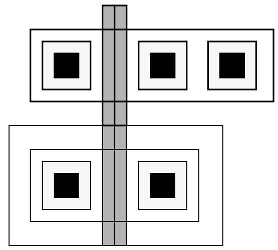
To view other users' edits,

1. Choose *Design – Hierarchy – Refresh*.
   - If no other users have made edits that affect this cellview, a dialog box tells you that no cellviews were refreshed.
   - If the master cells for any instances in this cellview have changed, a message asks if you want to update these instances.
2. In the Refresh Cellviews form, set any instances you want to update.
3. Click *OK*.

The cellview is redrawn to show changes made to the master cells.



Two cell instances as they appear before using *Refresh.*



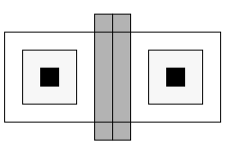
After using *Refresh*, you can see that the top cell has been edited.

Overview of Parameterized Cells (Pcells)

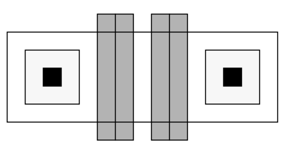
You can use the layout editor to create parameterized cells, or pcells. You use pcells to describe basic components of your design. You create the master for the pcell in its simplest form. Then you add parameters—settings that let you change the size, shape, or contents of each cell instance—without changing the master cell.

For example, you can create a master layout that describes a simple N-transistor. You can create parameters that control the length or width of the transistor, the number of gates, or the number of contact cuts.

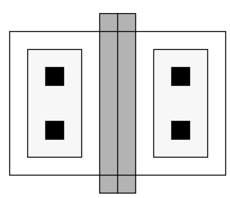
When you place instances of the transistor, you can change values for the parameters in the Create Instance form or the Create Device form. The original master cell is unchanged.



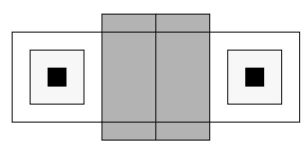
Master pcell



Instance with multiple gates



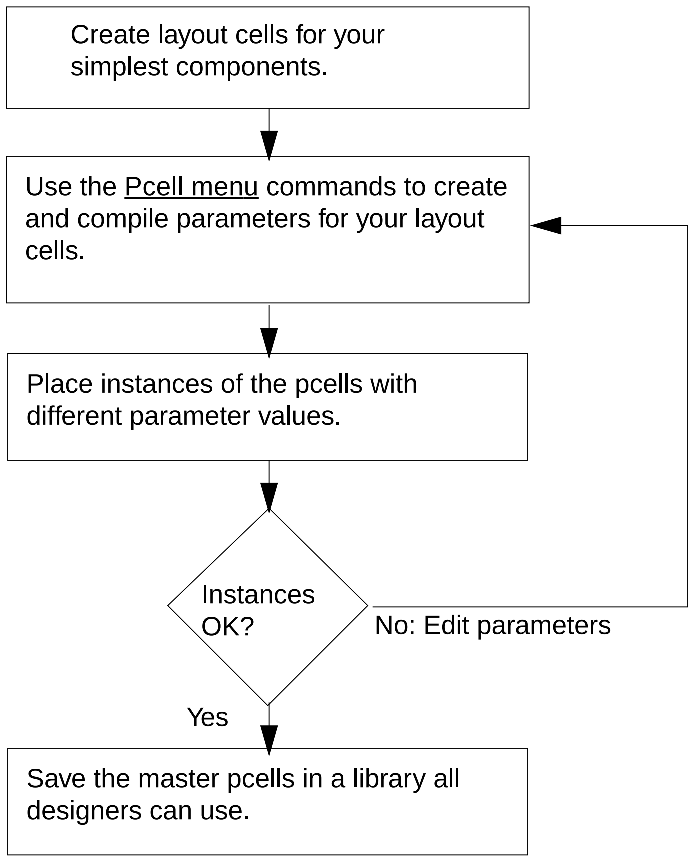
Instance stretched along its width



Instance stretched along its length

Pcell Design Flow

The following steps outline how you create pcells.



Setting Parameters of Pcell Instances

To set the parameters of a pcell instance when you place the instance,

1. Choose *Create – Instance* [`i`].

   The Create Instance form appears.

2. Type the name of the master pcell.
3. Press `Tab` to display the pcell parameters.

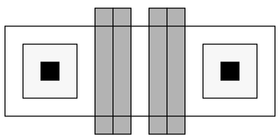
Note: If the Instance form is at the bottom of your screen, you may need to move the form up to see the parameter fields.

4. Type new values for any of the parameters.

5. Move the cursor into the design window.

   An outline of an instance of the pcell appears, using the parameters you entered. If the instance does not appear the way you want, keep editing the form until the instance is correct.

6. Click in the design window.



Instance with gates parameter set to 2

Changing Instance Parameters

To change the parameter values after you have placed an instance of a parameterized cell,

1. Choose *Edit – Properties* [q].

   The Edit Properties form appears.

2. Select the instances whose parameters you want to change.
3. Click on the *Parameter* button.

   The form shows the parameters for the first selected instance.

4. If you want to change the common parameters for a group of instances, set *Common* on.
5. Type new values for any of the parameters.
6. Click *Apply*.

   The instance changes according to the new parameter values.

7. Click *Cancel*.

Searching for and Replacing Parameters

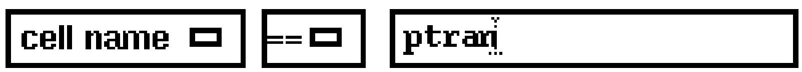
To search for and replace a parameter value,

1. Choose *Edit – Search* [`Shift-s`].

   The Search form appears.

2. In the *Search for* field, choose *instance.*

3. Click *Add Criteria* to open a criterion line.
4. Choose *cell name* and type the name of the master pcell in the fields that appear after you click *Add Criteria*.



5. Click *Add Criteria*.
6. Set the first field to *property* and type the property name and value you want to replace in the fields that appear after you click *Add Criteria*.

7. In the *Replace* field, set the first field to *property* and set the name, type, and value fields.

   Pcell properties are usually either *string* (for text values) or *float* (for numeric values).

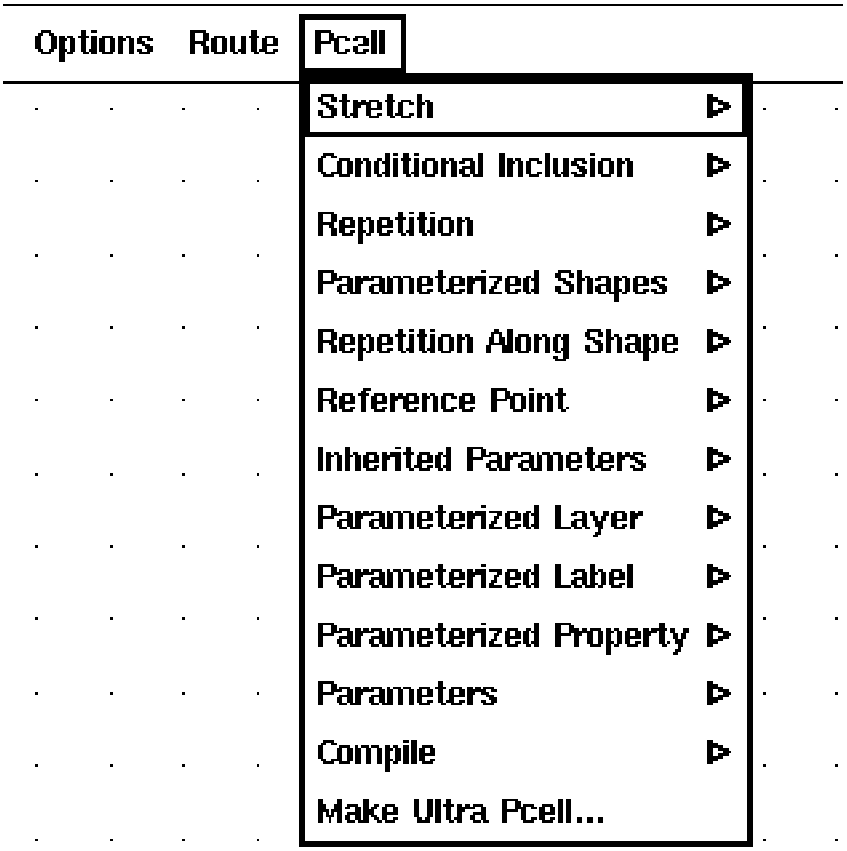
8. Click *Apply* to search for instances.
9. Click *Replace* to replace the top instance in the search stack or *Replace All* to replace all highlighted instances.

Adding the Pcell Menu to the Layout Editor

To add the *Pcell* menu to the layout editor,

➤ Choose *Tools – Pcell*.

The *Pcell* menu contains these commands:



See the *Virtuoso Parameterized Cell Reference Manual* for complete instructions about how to use pcells.

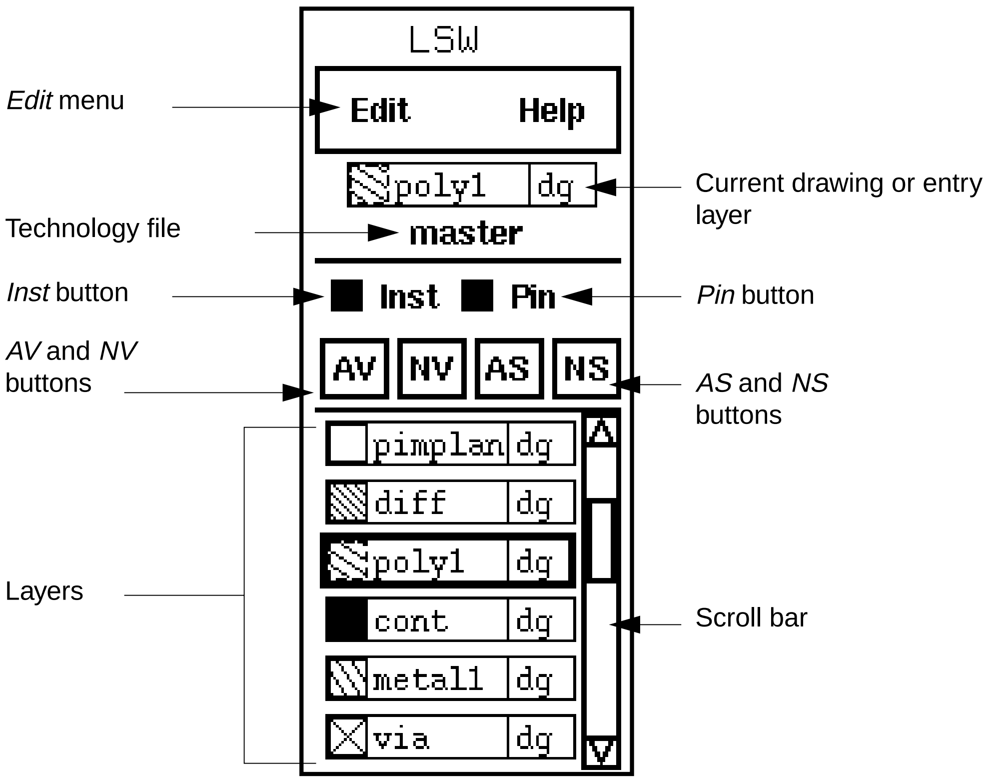
6

The Layer Selection Window (LSW)

This chapter contains these topics:

About the Layer Selection Window

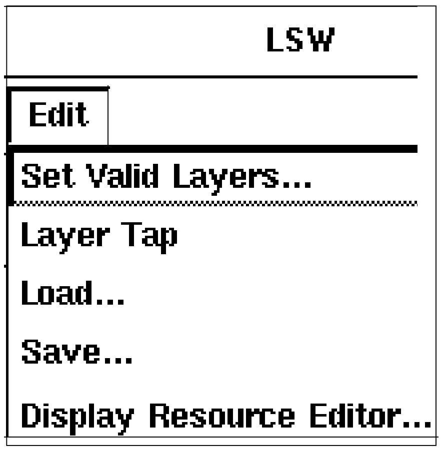
The Layer Selection Window (LSW) lets you choose the design layer for each shape you create, make design layers visible or invisible, or make instances and pins selectable or unselectable.



The default location of the LSW is the upper left side of the screen. You can change the size and location of the LSW by using `leSetLSWBBox()`.

Edit Menu

The *Edit* menu lets you set, change, save, or load layer settings.



The *Set Valid Layers* command lets you choose which layers appear in the LSW.

Note: If you are using the `leLswLayers` section of the `leRules` class of the technology file, only the layers listed there appear in the Set Valid Layer form.

The *Layer Tap* command lets you click on an object to set the current layer. Click repeatedly to cycle through all the layers under the cursor.

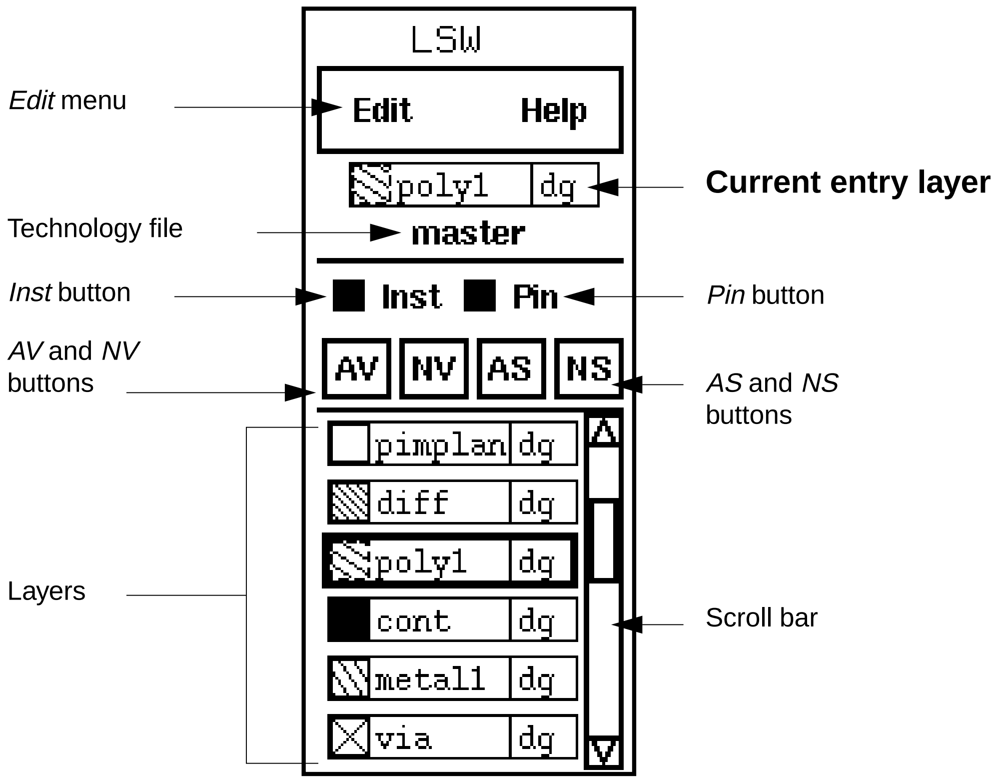
The *Load* command adds previously saved layers to the LSW.

The *Save* command writes your current LSW layer attributes to an ASCII file or to a technology file in virtual memory. If you want to make these attributes permanent, you must use the Save Technology File form in the Command Interpreter Window (CIW).

The *Display Resource Editor* command lets you change how layers appear.

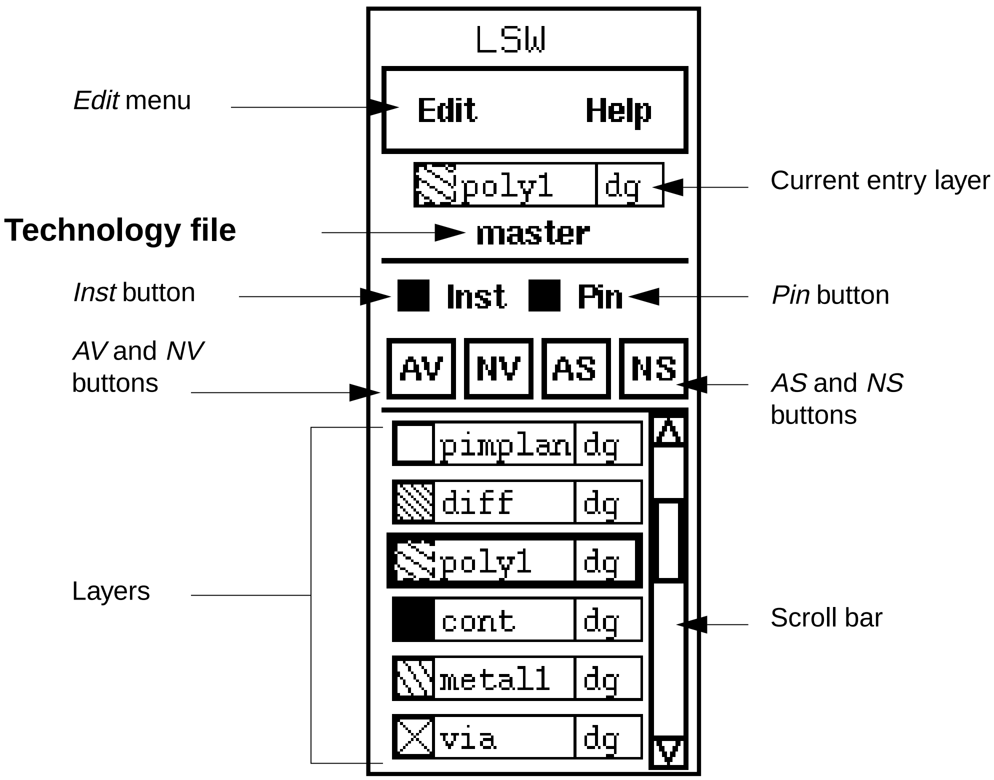
Current Entry Layer

The current entry field shows the current layer for all shape creation commands.



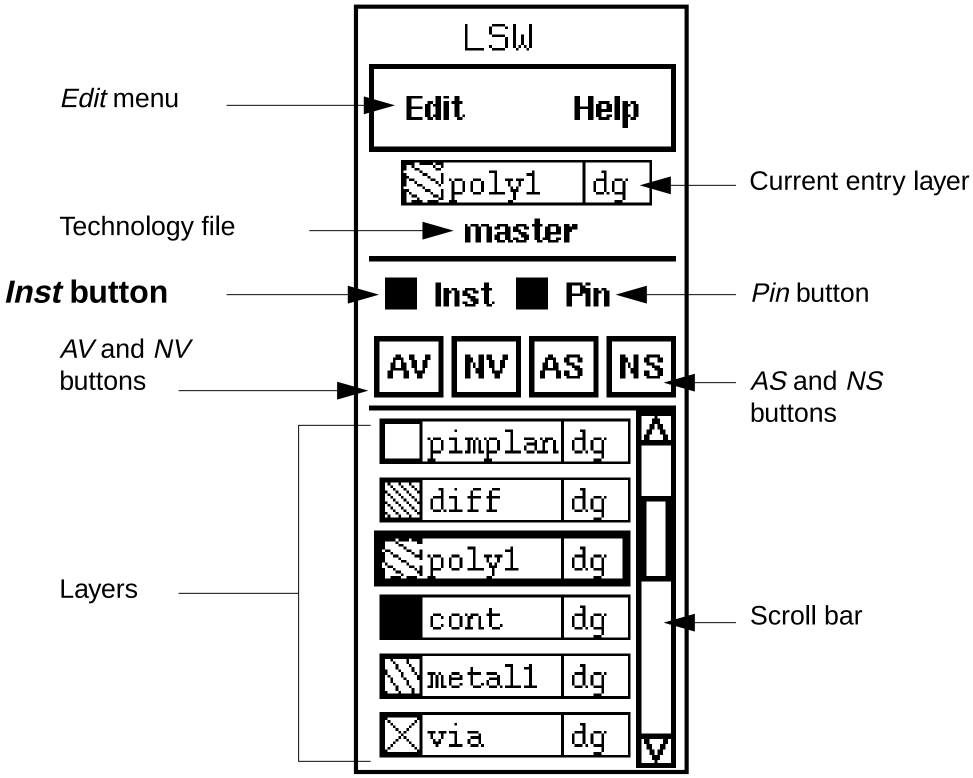
Technology File

The technology file field shows the name of the technology file that you are using.



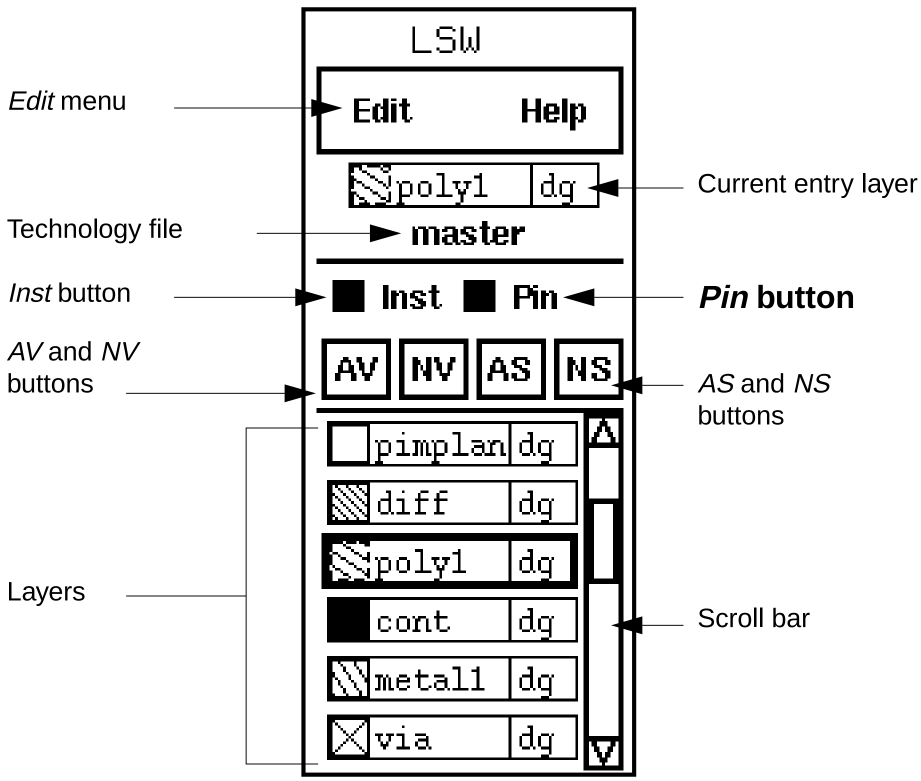
Inst Button

The *Inst* toggle button turns cell instances selectable (on) or unselectable (off).



Pin Button

The *Pin* toggle button turns pin instances used by Cadence® routing tools selectable (on) or unselectable (off). To select pins on a current layer, see Searching for Shapes and Pins on One Layer.

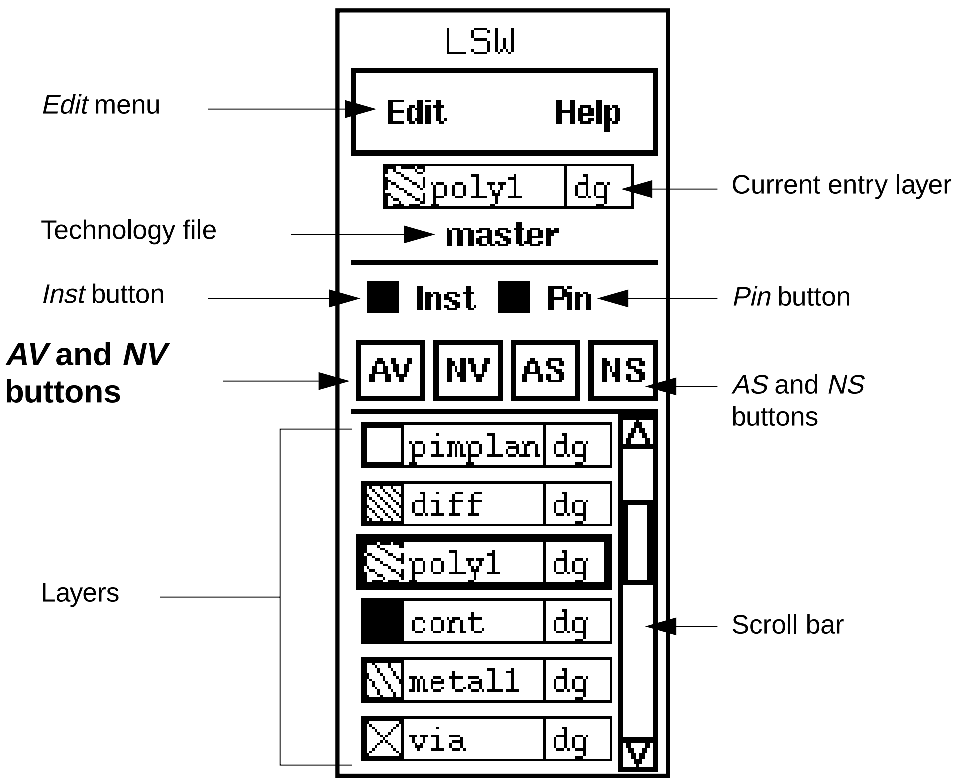


AV and NV Buttons

The *AV* button turns all layers visible.

The *NV* button turns all layers invisible in the layout window. This can cause the design to redraw faster. Invisible layers are not selectable in the LSW.

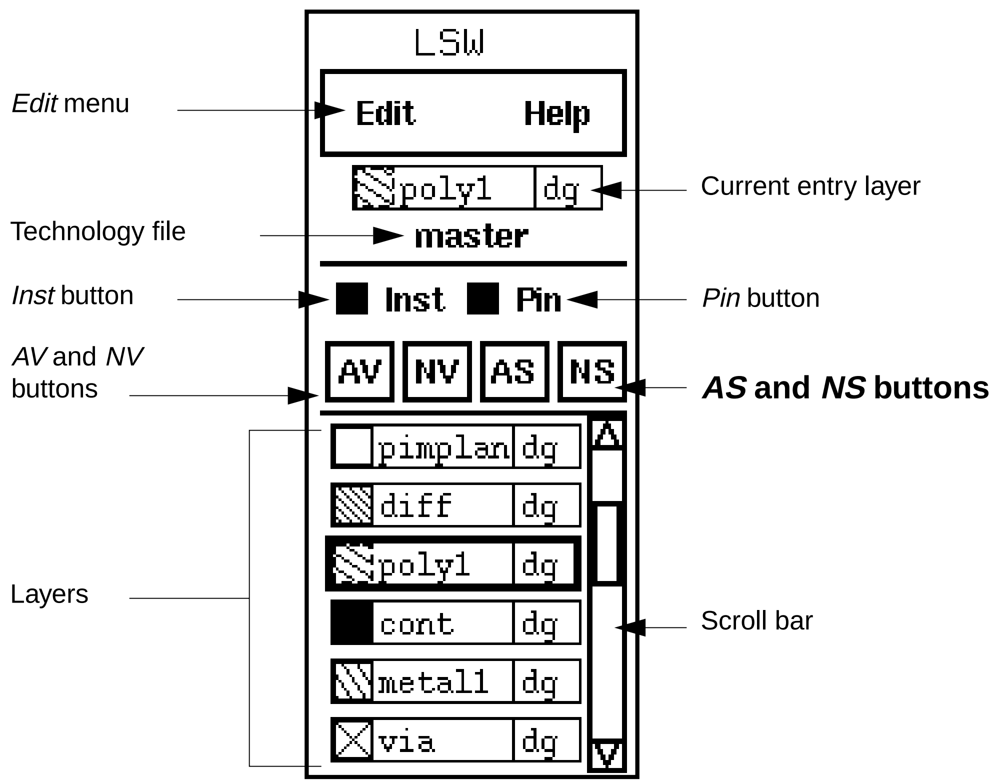
To see the result, choose *Window – Redraw*.



AS and NS Buttons

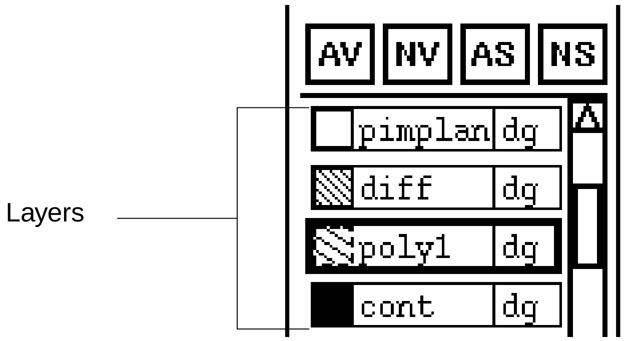
The *AS* button turns all layers selectable.

The *NS* button turns all layers unselectable. When a layer is unselectable, it cannot be selected in the layout window. You can make a complex, dense design easier to edit by making fewer layers selectable. This helps you avoid editing objects on the wrong layer.



Layers and Mouse Functions

The LSW lists all the valid layer-purpose pairs defined in the technology file. The layer-purpose pairs are sorted by display priority. If the `leLswLayers` rule is defined in the technology file, then only the valid `leLswLayers` layers are listed. In that case, the layer-purpose pairs are listed in the same order as they are defined in the `leLswLayers` rule.



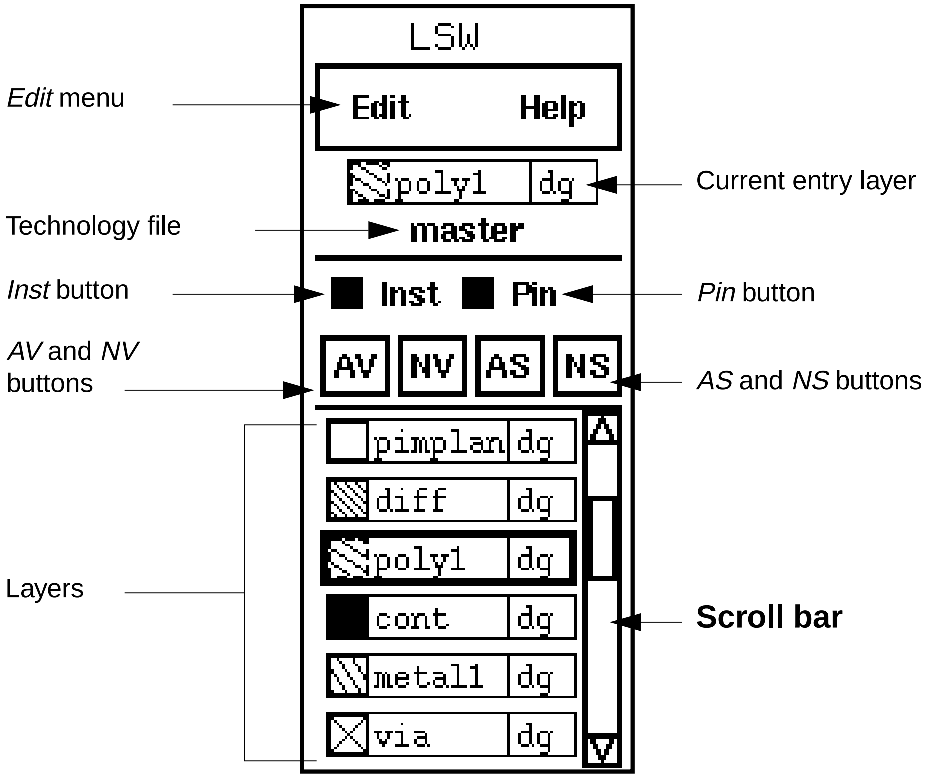
Click with one of the mouse buttons on a layer name to do any of the following:

- To set the current (entry) layer, click.
- To change layer appearance, press `Shift` and click.
- To toggle a layer between visible and invisible, click middle.
- To make all layers but one invisible, press `Shift` and click middle.
- To toggle a layer between selectable and unselectable, click right.
- To make all layers but one unselectable, press `Shift` and click right.

You can specify the order of the layers in the LSW by editing the `leLswLayers` class of the technology file. You can specify the layer abbreviation name by editing the `techLayers` class of the technology file.

Scroll Bar

Use the scroll bar to move up and down in the list of layers.



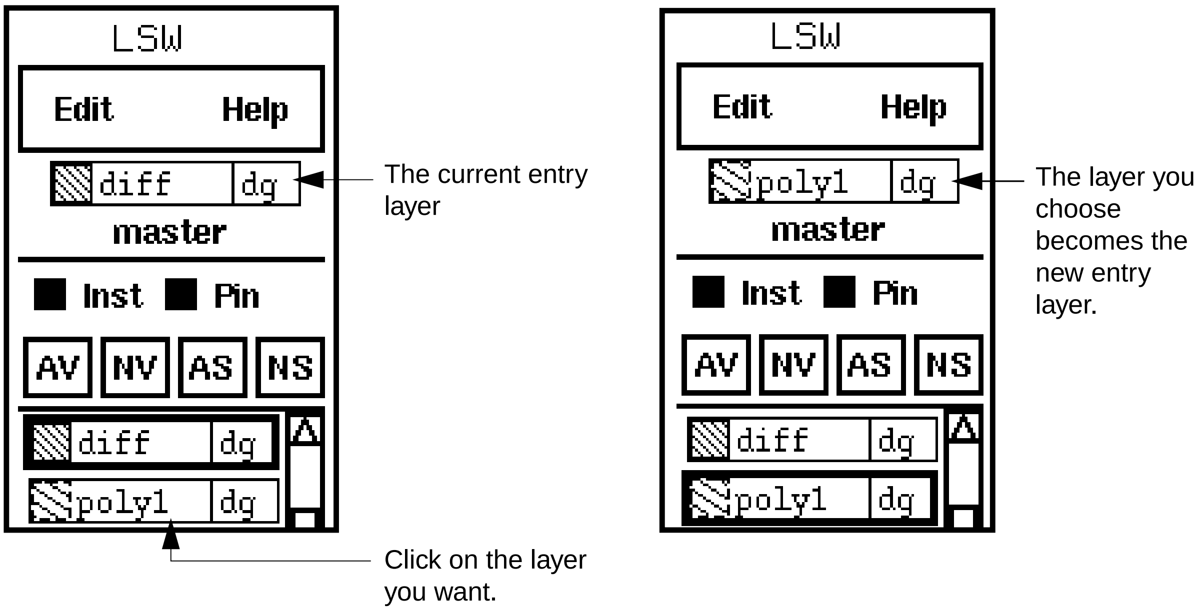
- To move down in the list of layers, click on the bottom arrow.
- To move up in the list of layers, click on the top arrow.

Choosing the Current Entry Layer

The layer shown at the top of the LSW is the current entry layer. You create objects on layers such as metal or via, which correspond to a layer for your physical design. You choose the entry layer with the LSW, which shows all the layers in your library that are valid.

➤ To choose an entry layer, click on the layer in the LSW.

The name of the layer appears at the top of the LSW.



Choosing the Entry Layer from an Object

The *Layer Tap* command lets you click on an object to set the entry layer. This is useful if you want to create objects on the same layer as some other object in your design. .

You can have the layer tap command behave in one of two ways by setting the environment variable `layerTapCycle` to either `t` or `nil`.

- When set to nil, the default, the system selects the layer of the object you tap on whose edge is closest to or coincident with the cursor.
- When `t`, you can cycle through the layers of overlapping objects by tapping repeatedly without moving the cursor until the layer you want is displayed.

To see which layer is currently the entry layer, look at the Layer Selection Window.

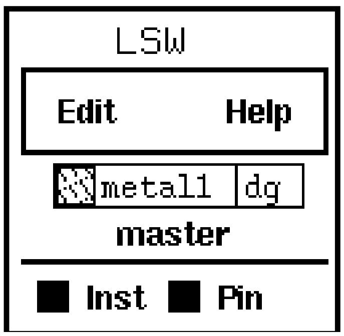
Using Layer Tap

To change entry layers using the *Layer Tap* command,

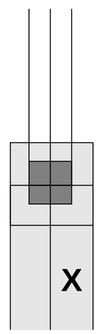
1. In the LSW, choose *Edit – Layer Tap* [`t`].

2. Click left on an object. If objects are overlapped, continue to click until you select the layer you want.

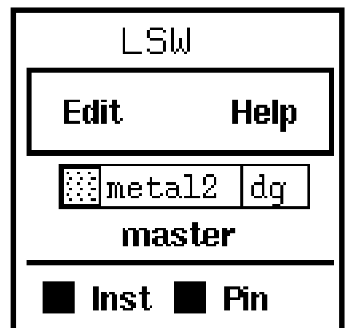
   The layer of the object becomes the current layer.



Before *Layer Tap*, the drawing layer is *metal1*.



Using *Layer Tap*, click on an object on *metal2*.



The drawing layer changes to *metal2*.

Note: If the object is drawn on a layer that is not in the LSW, the current entry layer is not changed.

You can use the bindkey `t` to start the *Layer Tap* command in a layout window even though the bindkey is not listed in the LSW *Edit* menu. It is not listed in the LSW *Edit* menu because it affects only layout windows, not the LSW.

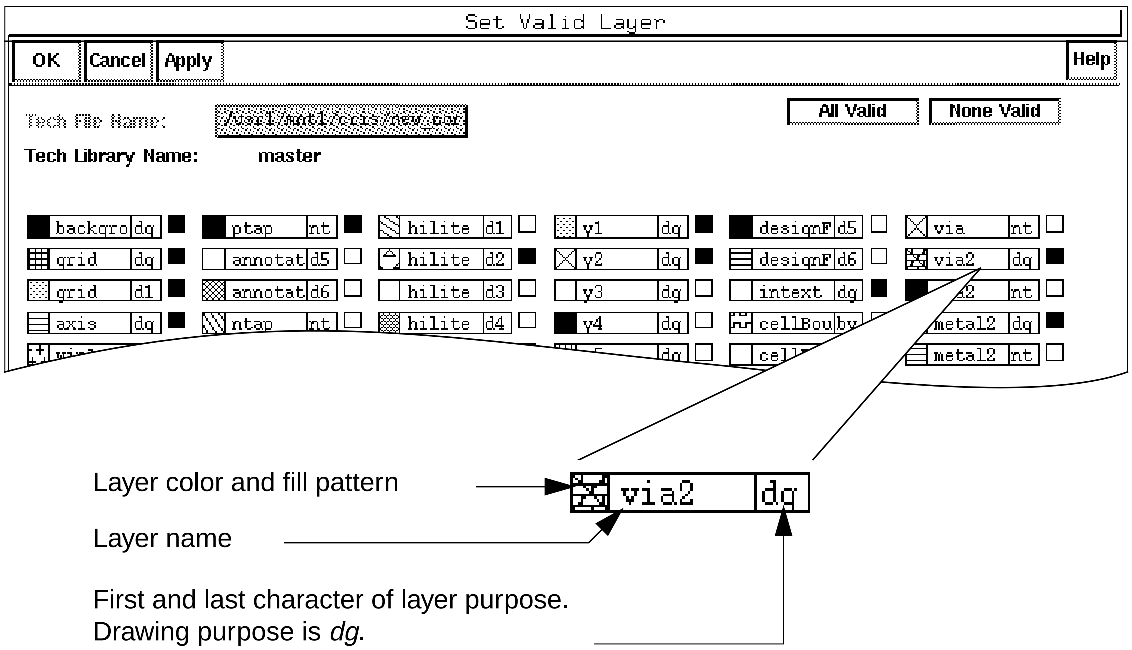
Changing the Available Entry Layers

The LSW lists all the valid layer-purpose pairs defined in the technology file. The layer-purpose pairs are sorted by display priority. If the `leLswLayers` rules are defined in the technology file, only the valid `leLswLayers` layers are listed. In that case, the layer-purpose pairs are listed in the same order as they are defined in the `leLswLayers` rule.

Note: If you are using the `leLswLayers` section of the `leRules` class of the technology file, only the layers listed there appear in the Set Valid Layer form.

The Set Valid Layer form is used to specify which layers are valid and therefore displayed in the LSW. The Set Valid Layer form displays all layer-purpose pairs in the technology file. However, if the `leLswLayers` rule is defined in the technology file, only the `leLswLayers` layers will be displayed in the Set Valid Layer form.

About the Set Valid Layer Form



Tech File Name displays the technology file being used for the current cellview.

Tech Library Name is the name of the library that contains the technology file.

All Valid makes all the layer-purpose pairs valid, making them appear in the LSW.

None Valid makes all layer-purpose pairs not valid, removing them from the LSW, except the current entry layer.

Layers lists all the layer-purpose pairs defined in the technology file. If the `leLswLayers` rule is defined in the technology file, only the `leLswLayers` layers are listed. The box to the right of the layer-purpose pair icon indicates whether or not the layer is valid. If the box is filled, the layer is valid and is displayed in the LSW. If the box is empty, the layer is marked invalid and is not displayed in the LSW.

Setting Valid Layers

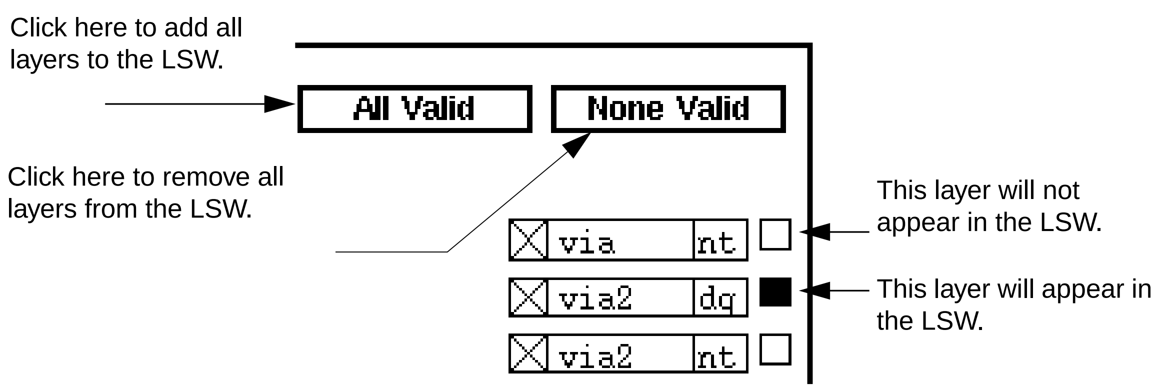
To add or remove layers from the LSW (making them valid or not valid),

1. Choose *Edit – Set Valid Layers* in the LSW.

The Set Valid Layer form appears.

2. Do any of the following:
   - To add a layer to the LSW, click on the square to the right of the layer name so it is filled in.
   - To remove a layer from the LSW, click on the square to the right of the layer name so it is empty.
   - To add all layers to the LSW, click *All Valid*.
   - To remove all layers except one layer from the LSW, click *None Valid*.

Note: Even though you click *None Valid*, one layer will always be valid in the LSW.



3. Click *OK*.

   Your changes are stored in the LSW database.

   To save your changes to a file, use the Save form. To save your changes to the technology file, first use the Save form to save the changes to memory, then use the Save Technology File form to write the technology file to disk.

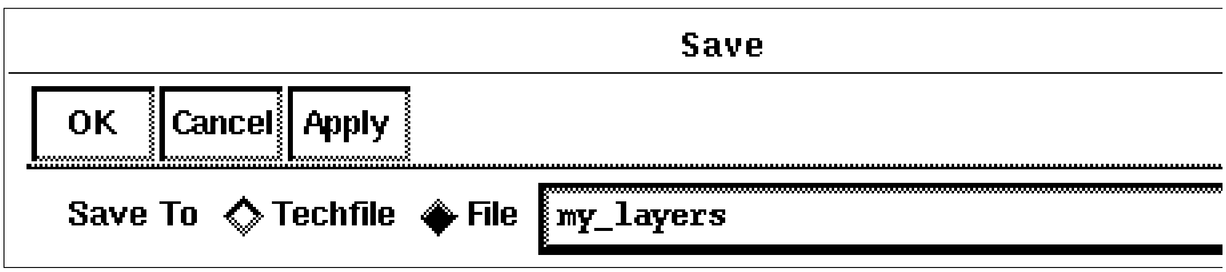
You might want to use different sets of drawing layers to edit different types of cells in a library. The *Save* command saves the current set of drawing layers shown in the LSW to a file or to the technology file. Later you can load layers from the file or technology file.

Note: If you are using the `leLswLayers` section of the `leRules` class of the technology file, only the layers listed there appear in the Set Valid Layer form.

Using the Save Form

The Save form writes your current LSW layer attributes to an ASCII file or to a technology file in virtual memory. If you want to make these attributes permanent, you must choose *Technology File – Save* from the CIW menu banner.

About the Save (Valid Layers) Form



Save To lets you save all layers' validity, visibility, and selectability attributes to either a technology file or an ASCII file.

Techfile saves all layers listed in the LSW as valid layers in the technology file used for this library.

File saves all layers in the LSW to a file that you can later load with the *Edit – Load* command in the LSW.

Saving a Set of Entry Layers

To save layers,

1. In the LSW, choose *Edit – Save*.

   The Save form appears.

2. Do one of the following:

   - To save to a file, choose *File* and type the name of the file. Include a path if you want to save the file outside your current directory.

   - To save to the technology file (shown at the top of the LSW) in memory, choose *Techfile*.

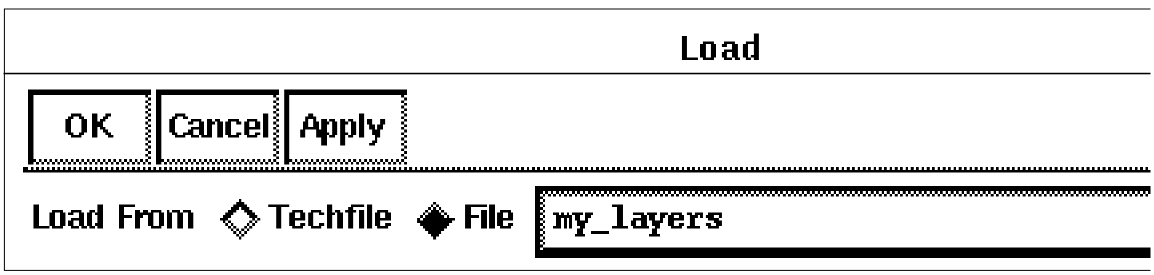
3. Click *OK*.

The LSW's current validity, visibility, and selectability attributes are written to the specified file or technology file in virtual memory. If you want to make these attributes permanent, you must choose *Technology File – Save* f rom the CIW menu banner.

Using the Load Form

The Load form lets you change the LSW validity, visibility, and selectability attributes from a saved file or the technology file in virtual memory. Because the LSW displays only valid layers, the *Load* command can change which layer-purpose pairs are displayed in the LSW. Loading from a file can be used to restore a predefined set of valid, visible, and selectable layer-purpose pairs. Loading from the technology file can be used to restore the default validity, visibility, and selectability attributes.

About the Load (Valid Layers) Form



Load From loads layers defined in either a technology file or an ASCII file.

Techfile loads all layers defined as valid drawing layers in the technology file used for this library.

File loads layers you previously saved to a file using the *Edit – Save* command in the LSW.

Loading Layers from a File

To use a set of layers you saved,

1. In the LSW choose *Edit – Load*.

   The Load form appears.

2. Choose *File* and type the name of the file you want to load.

Include a path if you want to load a file stored outside your current directory.

3. Click *OK*.

   The list of layers in the LSW changes.

Setting How Layers Appear

Changing Layer Appearance

The layers and how they appear on the screen and in your plots are defined by the technology file and the `display.drf` file.

The technology file defines the layer-purpose pairs. This includes the layer name, purpose name, display priority, display packet, and layer rules.

➤ To change a layer definition in the technology file, use the <u>Edit Layers form</u>.

Note: Changing the technology file affects all libraries that use that technology file.

The `display.drf` file defines the display resources including colors, stipples, line styles, and packets. Each packet is a named combination of stipple, line style, fill color, and outline color. Packets are associated with one or more layer-purpose pairs in the technology file.

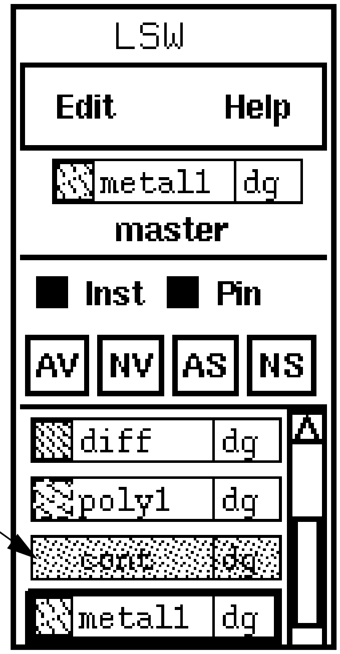
➤ To change a layer definition in the `display.drf` file, use the <u>Display Resource Editor</u>.

Making One Layer Visible or Invisible

To reduce the amount of detail in the cellview, you can make some layers invisible.

To make a layer invisible,

1. In the LSW, click middle on the layer.



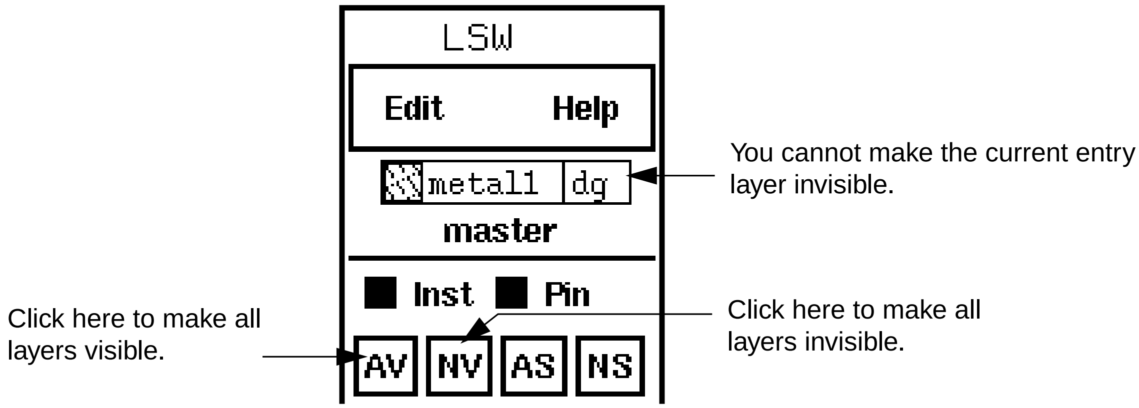
The layer color disappears. The layer name is shaded, to show that the layer is also unselectable.

2. Choose *Window – Redraw* to see the result.
3. Repeat steps 1 and 2 if you want to make the layer visible again.

Making All Layers Visible or Invisible

To make all layers except the current entry layer invisible,

1. Click *NV* (not visible) at the top of the LSW.



2. Choose *Window – Redraw* to see the result.
3. Click *AV* (all visible) at the top of the LSW to make all layers visible again.
4. Choose *Window – Redraw* to see the result.

   You cannot make the current drawing layer, shown at the top of the LSW, invisible.

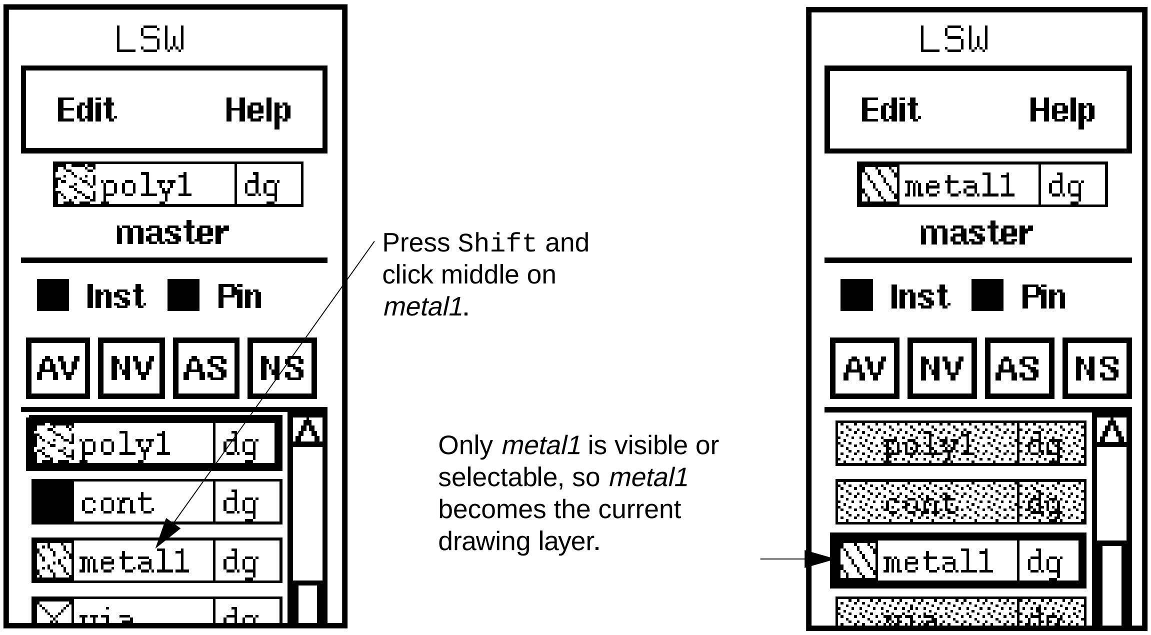
   Note: If you are using the `leLswLayers` section of the `leRules` class of the technology file, only the layers listed there appear in the Set Valid Layer form.

Making All but One Layer Invisible

To make only one layer visible,

1. In the LSW, press `Shift` and click middle on the layer you want to be the only visible layer.

All other layers are shaded to show that they are invisible and unselectable.



2. Choose *Window – Redraw* to see the result of your changes.

7

Moving Around in the Window

This chapter contains these topics:

Zooming or Panning the Display

Zooming In or Out

The zoom commands control the magnification of the cellview.

The *Zoom – In* command prompts you to create a box and then magnifies the image defined by the box to fit in the cellview window.

To zoom in on a specific area,

➤ Choose *Window – Zoom – In* and then create a box enclosing the area.

The *Zoom – To Sel Set* command increases the image to the largest magnification at which the selected objects can be viewed in the cellview window.

To zoom in on selected objects,

➤ Select the objects and then choose *Window – Zoom – To Sel Set*.

The *Zoom – To Grid* command reduces the image to the smallest magnification at which the grid is visible.

To zoom to grid,

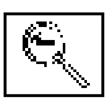
➤ Choose *Window – Zoom – To Grid*.

The *Zoom – In by 2* and *Zoom – Out by 2* commands let you magnify or reduce the image in the cellview window by a factor of 2.

To zoom out by 2,

➤ Do one of the following:

- ❑ Choose *Window – Zoom – Out by 2.*



- ❑ Click on the zoom-out icon in the icon menu.

To zoom in by 2,

➤ Do one of the following:

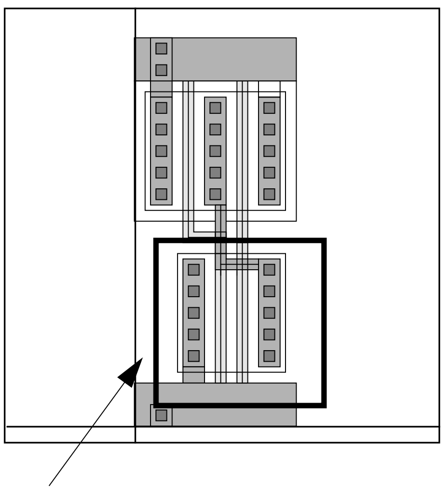
- ❑ Choose *Window – Zoom – In by 2.*
- ❑ Click on the zoom-out icon in the icon menu.

Zooming with the Mouse

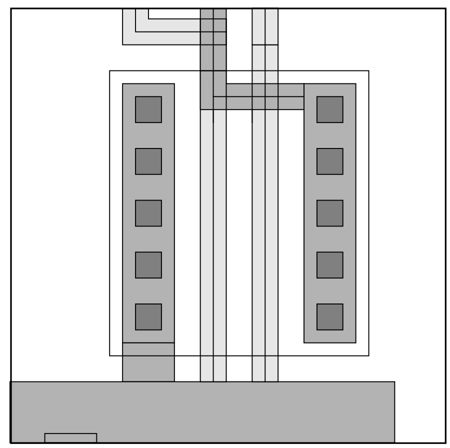
You can create a box to zoom in on a specific area or zoom out to a specific size.

To zoom in,

➤ Click right, and drag the cursor to create a box around the area you want to enlarge.



Press right to create a box around the area you want to enlarge.



That area enlarges to fill the window.

To zoom out,

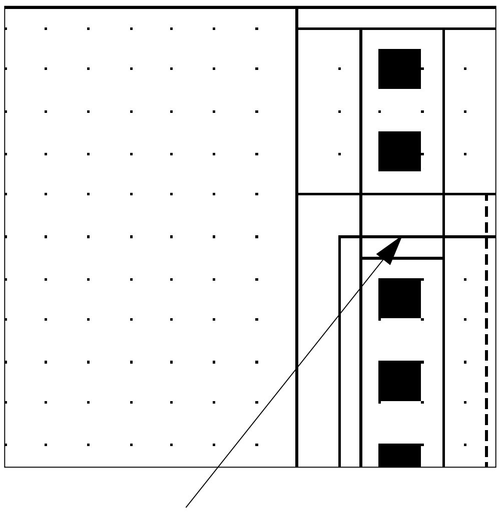
➤ Press `Shift`, click right, and drag the cursor to create a box into which you want the reduced image to fit.

Panning to a Point

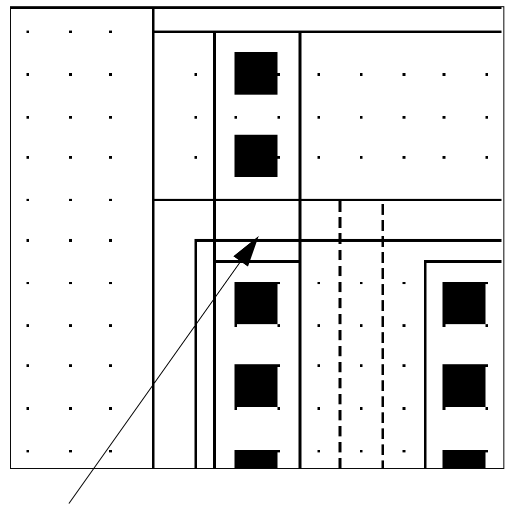
To center the image in the cellview window around a point you enter,

1. Choose *Window – Pan* [`Tab`].
2. Click on the point you want to center.

The image moves so the point you chose is in the center of the window.



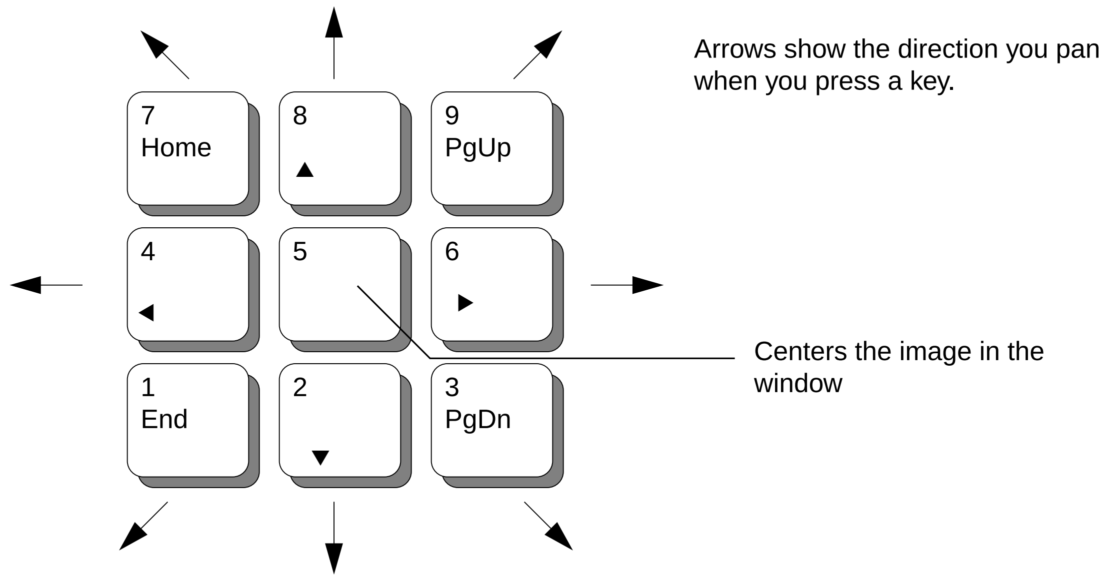
Click on the point you want to center.



The cellview is centered around that point.

Panning Across the Cellview

The arrow keys on the keypad let you pan across the cellview in any direction.



For example, to pan to the left,

➤ Press 4 on the keypad.

The image moves to the right to display new data on the left.

Returning to a Previous Zoom or Pan Image

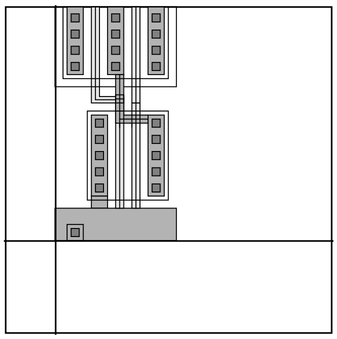
The Virtuoso® layout editor keeps the last three images you viewed in memory. You can move forward and back through these previously zoomed or panned images when you need to.

To return to the last image,

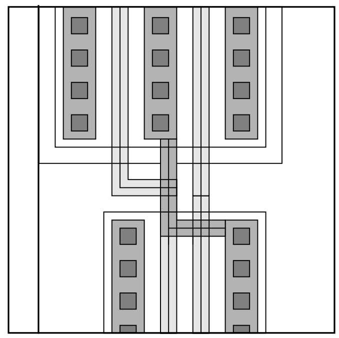
➤ Choose *Window – Utilities – Previous View* [w].

To go forward to the next image,

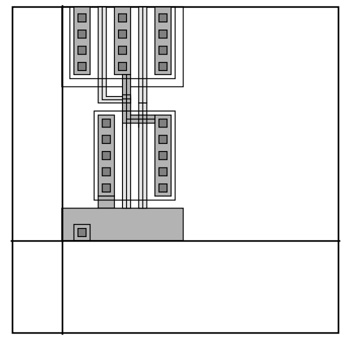
➤ Choose *Window – Utilities – Next View* [`Shift-w`].



A panned image.



Use *Zoom – In* to enlarge the image.



Use *Utilities – Previous View* to return to the first image.

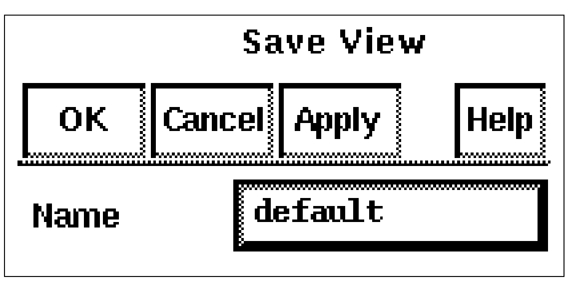
Saving a Zoom or Pan Image

The Save View form saves the image shown in the window to a file that lasts for the current editing session.

About the Save View Form

To open the Save View form,

➤ Choose *Window– Utilities – Save View*.



Name sets the name you want to assign this view. This name will appear in the Restore View form.

To save a zoom or pan image,

1. Choose *Window – Utilities – Save View*.

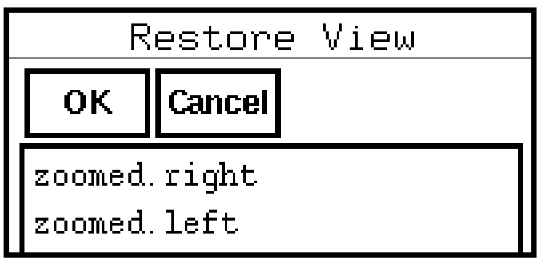
2. In the Save View form, type the name you want to assign the file.
3. Click *OK*.

Restoring a Zoom or Pan Image

After you have saved the image, you can later restore it with the *Restore View* command.

1. To restore a saved image, choose *Window – Utilities – Restore View*.

   The Restore View window opens.



2. Click on the name of the file you want to restore.
3. Click *OK*.

   The image in the window is replaced by the image you saved.

Note: Images are saved only during the current editing session. They are deleted when you exit the Cadence® software.

Working with a Reference Window

Creating a Reference Window

You can display a small copy of the whole cellview. This is often called a reference window because you use it to see your whole layout while you edit a zoomed-in portion.

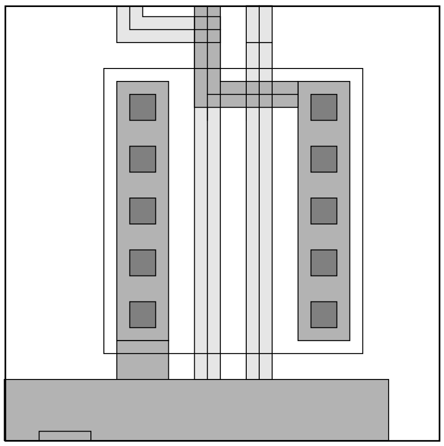
To create a reference window,

1. Choose *Window – Utilities – Copy Window*.

   A second window appears, displaying the same cellview.

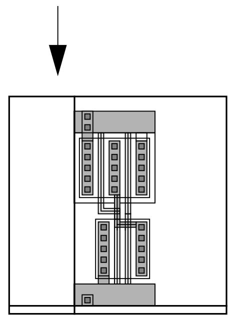
2. To shrink the new window, click on a corner of the new window and drag it towards the opposite corner.

3. In the new window, choose *Window – Fit All* [`f`].



The original cellview window.

Fit All displays the entire design in the reference window.



The new reference window.

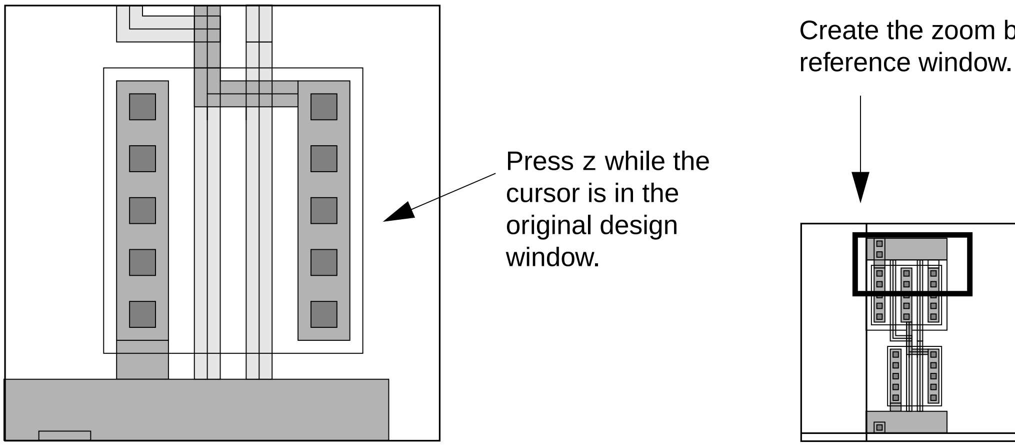
Using a Reference Window to Zoom or Pan

When you have two windows displaying the same cellview, you can start a *Zoom* or *Pan* command in your original window and then enter the points to zoom or pan in the reference window. This way, you can pinpoint the area you want to display using the reference window.

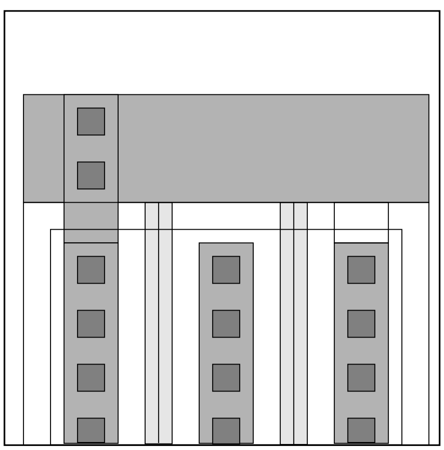
To zoom in a reference window,

1. With the cursor in the original window, choose *Zoom – In*.
2. Move the cursor into the reference window.

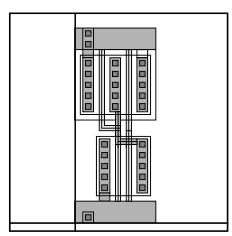
3. Click to create the zoom box in the reference window.



The original window zooms in to the area you designated in the reference window, but the reference window remains unchanged.



Original window is zoomed in.



Reference window is unchanged.

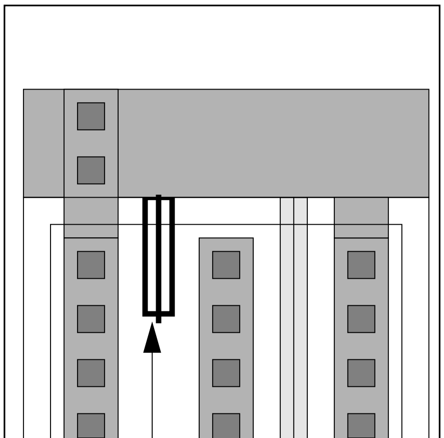
Using a Reference Window to Create or Edit Objects

When you have two windows displaying the same cellview, you can start an editing command in one and finish it in the other window.

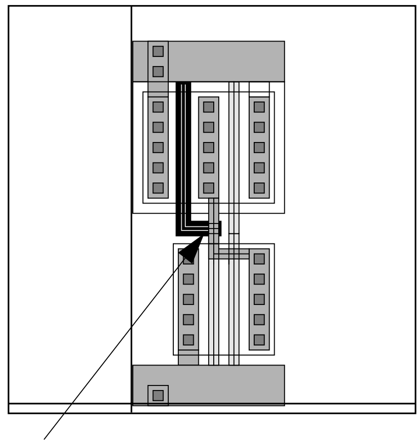
To create a path in a reference window,

1. In the original window, choose *Create – Path*.

2. Click to begin creating the path.
3. Move the cursor into the reference window, and double-click to finish the path.



Start creating the path in the original window.



Finish the path in the reference window.

Redrawing the Display

After you have made many edits to a cellview, the design area can sometimes show portions of lines or objects that you have moved or deleted.

To redraw the cellview, do one of the following:

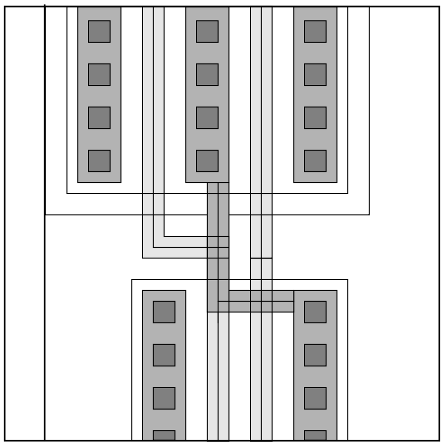
- Move the pointer into the cellview and press `Control-r`.
- Choose *Window – Redraw*.

Fitting the Entire Cellview in a Window

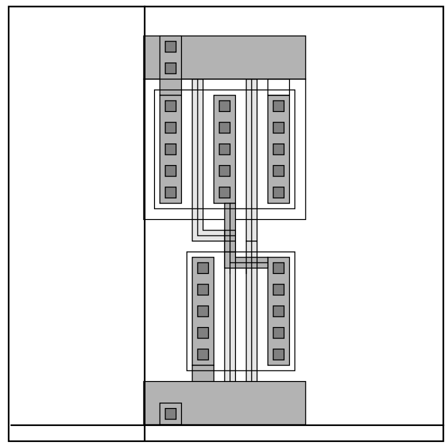
The *Fit All* command redraws the window so that all objects in the cellview appear in the window. This saves time after you zoom or pan and want to see the entire cellview again.

To fit the cellview in the window,

➤ Choose *Window – Fit All* [f].



Before *Fit All*.



After *Fit All*. All data in the cellview is centered in the window.

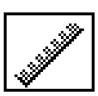
Note: There is also a *Fit Edit* command in the *Window* menu. You use *Fit Edit* during edit-in-place to center the cell you are editing. If you are not editing in place, *Fit Edit* has the same result as *Fit All.*

Using Rulers

The *Create Ruler* command lets you use rulers to measure objects in the cellview.

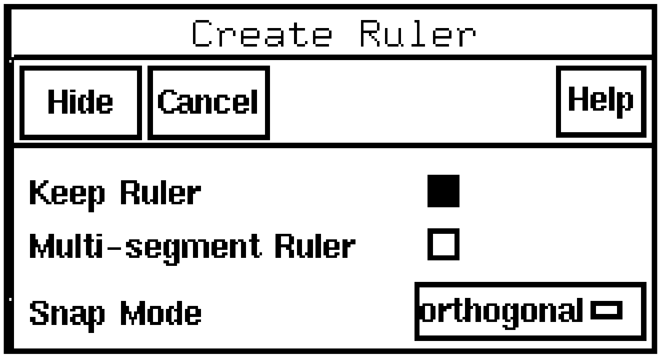
To open the Create Ruler form, do one of the following:

- ❑ Choose *Window – Create Ruler* and either double-click middle or press F3.
- ❑ Press k and either double-click middle or press F3.



- ❑ Click on the ruler icon in the icon menu and either double-click middle or press F3.

About the Create Ruler Form



Keep Ruler creates a ruler that remains until you delete it with the *Clear All Rulers* command or close the window.

Multi-segment Ruler lets you create a ruler with several line segments.

Snap Mode controls the shape of the ruler segments.

Creating Rulers to Measure Objects

You use rulers to measure objects or the distance between objects.

Note: The ruler is not a permanent part of the database. If you close a cellview, any rulers you drew will not appear when you reopen it.

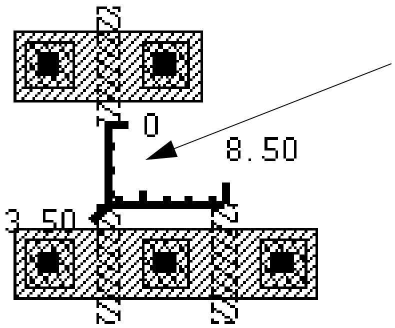
To create a ruler,

1. Choose *Window – Create Ruler*.

   To create a ruler with more than one line segment, set *Multi-segment Ruler* on in the Create Ruler form.

2. Click to enter each point of the ruler.

3. Double-click on the last point.



A multisegment ruler (bold lines) shows the total distance, in user units, between each point of the ruler.

4. Choose *Window – Clear All Rulers* to clear any rulers from the screen.

8

Creating Objects

This chapter contains these topics:

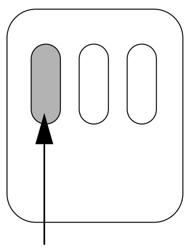
Entering and Deleting Points

When you create objects in the Virtuoso® layout editor, you enter and delete points to create the shape you want.

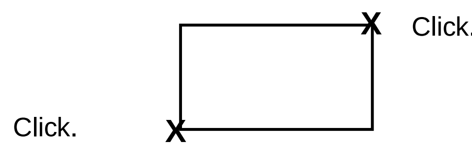
Entering Points

To enter points,

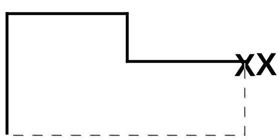
1. Click and release the left mouse button.
2. Double-click to enter the last point for a polygon or path.



Click to enter each point.



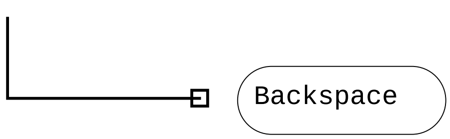
Example: Creating a rectangle



Double-click to close the polygon.

Deleting Points

To delete a point, press `Backspace`.

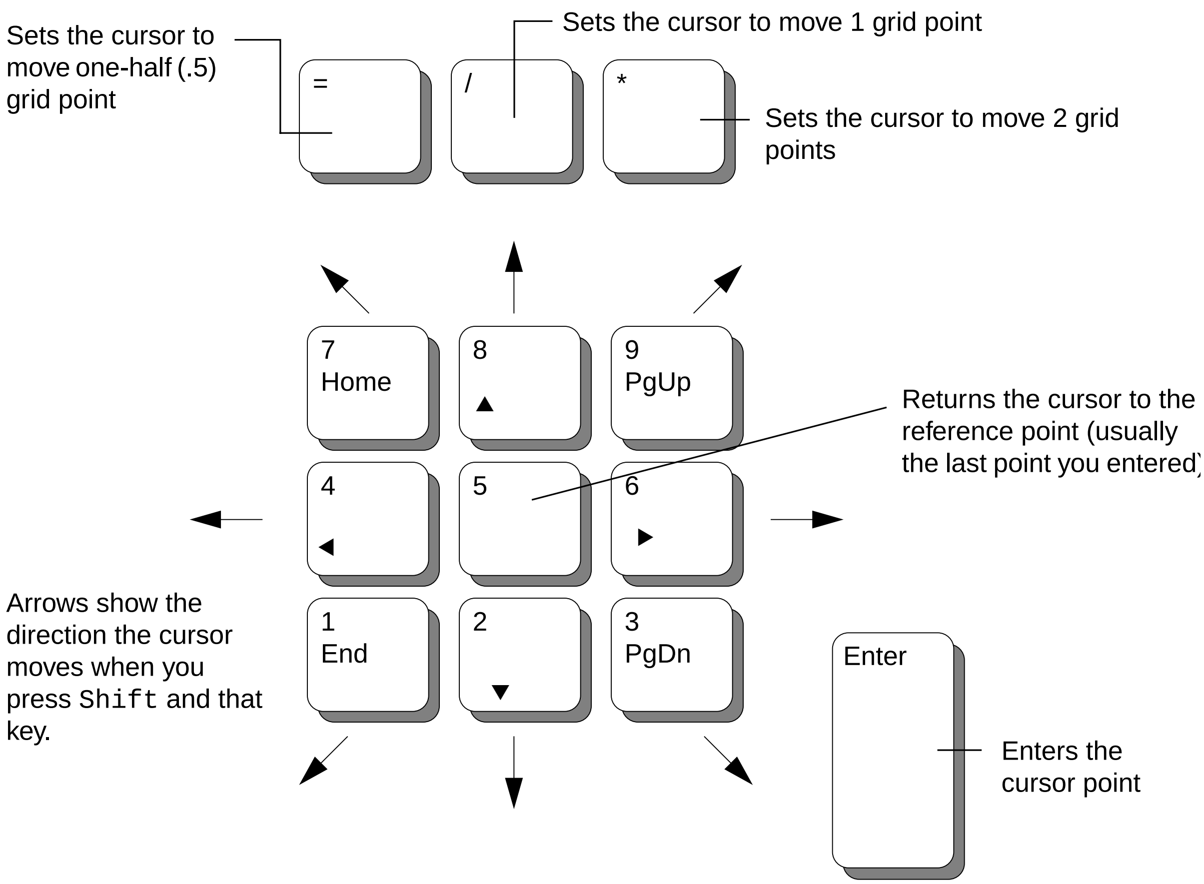


Press `Backspace` to cancel a point.

The cursor returns to the previous point.

Entering Points with the Keypad

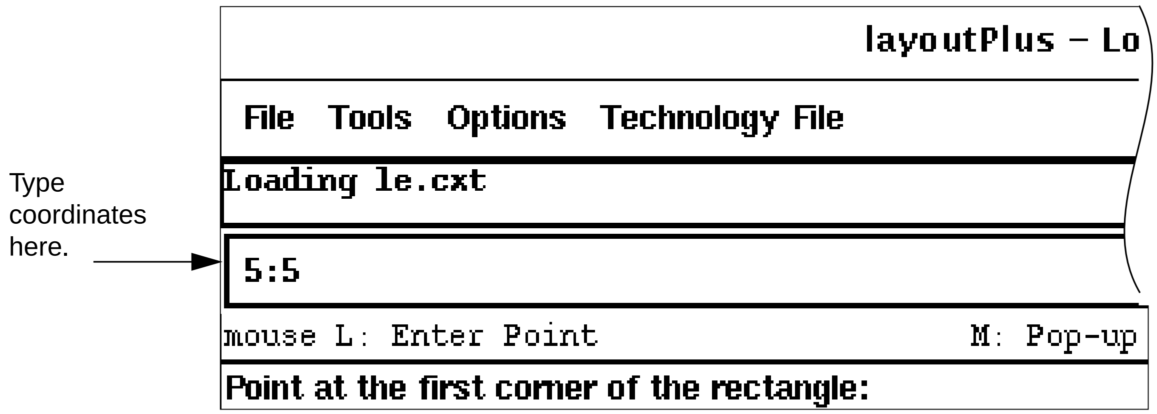
You can use the keypad to move the cursor and to enter points. This can be helpful if, for example, you want to enter a point just a few units away from the last point.



Typing Points in the CIW

You can enter points by typing coordinates into the Command Interpreter Window (CIW.)

When you are prompted for a point, type a coordinate pair into the CIW, use a colon to separate the coordinates, as shown, and press `Return`.



You can continue typing coordinates each time you are prompted for a point.

If you are creating a polygon or path, type in the same coordinates twice to indicate the last point.

How Creation Commands Work with ROD Objects

The following tables summarize the level of support for how creation commands work on relative object design (ROD) objects in the current release.

Using commands that are not fully supported for ROD objects could cause the objects to lose the ROD information associated with them, changing the objects into ordinary shapes.

Note: ROD functionality is not implemented for mosaics.

Create Command	Degree of ROD Support
Rectangle... r	Create and name new rectangles as ROD objects using the *Create Rectangle* form.
Polygon... P	Create and name new polygons as ROD objects using the *Create Polygon* form.
Path... p	Create and name paths as ROD objects using the *Create Path* form.

Multipart Path...	Using Virtuoso® layout accelerator, create and name new multipart paths as ROD objects using the *Create Multipart Path* form. This form lets you save form values as a template in your technology file or choose a template from your technology file.
Label... l	You cannot create a label as a ROD object. However, you can make an existing label a ROD object by assigning it a name with the `rodNameShape` function.
Instance... i	An instance is automatically a ROD object because it has a unique name. The ROD object name is the same as the instance name.
Pin... ^p	Create and name new pins as ROD objects using the *Create Pin Shape* form.
Pins From Labels...	You cannot create a pin from a label as a ROD object. However, you can make an existing pin a ROD object by assigning it a name with the `rodNameShape` function.
Contact... o	A symbolic contact is automatically a ROD object because it is an instance and has a unique name. The ROD object name is the same as the instance name.
Device...	A symbolic device is automatically a ROD object because it is an instance and has a unique name. The ROD object name is the same as the instance name.
Conics ▷ Circle	Circles are not supported as ROD objects.
Ellipse	Ellipses are not supported as ROD objects.
Donut	Donuts are not supported as ROD objects.
Layer Generation...	You cannot create a shape using a layer generation operation as a ROD object. However, you can make a generated shape a ROD object by assigning it a name with the `rodNameShape` function.

Creating Rectangles

The *Create Rectangle* command lets you create rectangular shapes. When you create a rectangle, you have the option of designating it as a relative object design (ROD) object. The ROD object contains information about the rectangle, including its name and database ID.

Creating a ROD object using the *Create Rectangle* command lets you

- Specify connectivity to associate this rectangle with other shapes on the same net by typing the net name in the *Net Name* field
- Give the rectangle a ROD name by either using the default database name or typing another name in the *ROD Name* field

  ROD names can have spaces in them. For example, `rectangle 1a` is a valid ROD name.

Only one ROD object can be created and assigned to a net at a time. You cannot enter a series of names in the *Net Name* and *ROD Name* fields and expect the names to peel off at the spaces separating the words.

You can use the environment variable `rodAutoName` to set the Create Rectangle form to automatically

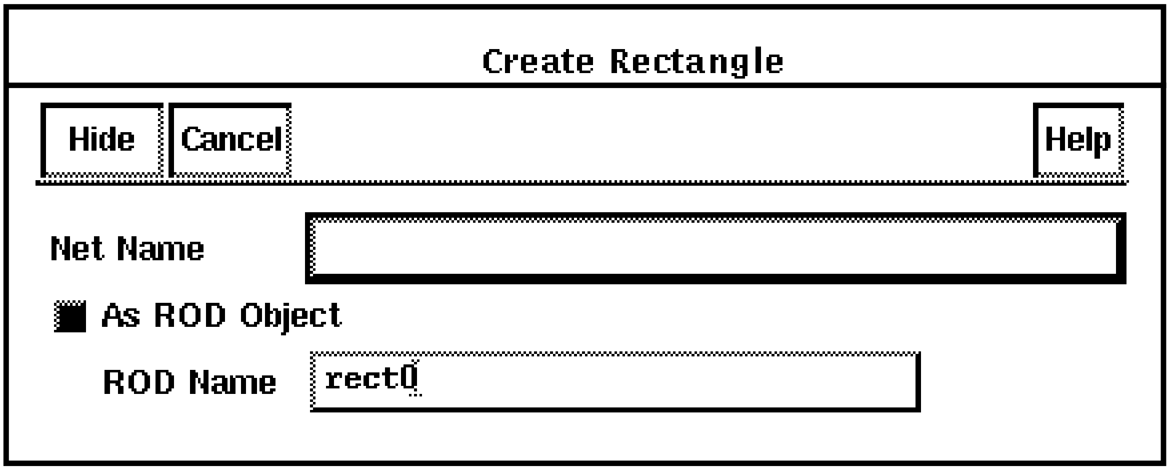
- Create ROD objects
- Make the *ROD Name* field editable

Once the rectangles are created, you can edit them by typing ROD function commands in the CIW. These commands let you

- Access ROD objects by name through all levels of hierarchy
- Access ROD objects' handle values through all levels of hierarchy
- Align ROD objects to each other or to specific coordinates
- Assign names to unnamed rectangles, polygons, lines, and paths
- Create complex parameterized cells more easily

See the *Virtuoso Relative Object Design User Guide* for information about ROD objects.

About the Create Rectangle Form



Net Name assigns the rectangle to a specific net.

As ROD Object creates the rectangle as a ROD object.

ROD Name assigns a name to the new rectangle in the current layout cellview if *As ROD Object* is on. The name must be unique in the cellview. If you do not edit this field, the system assigns a unique name consisting of the prefix `rect`, followed by a number. For example, the first ROD rectangle would be named `rect0`, the second, `rect1`, and so on.

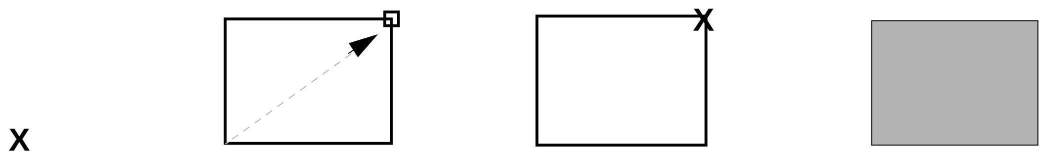
Creating a Rectangle

To create a rectangle,

1. In the Layer Selection Window, click on the layer you want.
2. Choose *Create – Rectangle* [`r`].
3. In the *Net Name* field, type the net name you want the rectangle to be associated with.
4. If you want to create a ROD object rectangle, turn on *As ROD Object*.

   The *ROD Name* field becomes editable, allowing you to change the name from the default name that is displayed. If you do not change the name, the default name is assigned to the rectangle.
5. Click to enter the first corner of the rectangle.

6. Click to enter the opposite corner of the rectangle.



Click left. | Move the mouse. | Click left to finish the rectangle. | The finished rectangle

Creating Polygons

The *Create Polygon* command lets you create polygonal shapes. When you create a polygon, you have the option of designating it as a <u>ROD</u> object. The ROD object contains information about the polygon, including its name and database ID.

Creating a ROD object using the *Create Polygon* command lets you

- Specify connectivity to associate this polygon with other shapes on the same net by typing the net name in the *Net Name* field
- Give the polygon a ROD name by either using the default database name or typing another name in the *ROD Name* field

  ROD names can have spaces in them. For example, `polygon 1a` is a valid ROD name.

Only one ROD object can be created and assigned to a net at a time. You cannot enter a series of names in the *Net Name* and *ROD Name* fields and expect the names to peel off at the spaces separating the words.

You can use the environment variable <u>`rodAutoName`</u> to set the Create Polygon form to automatically

- Create ROD objects
- Make the *ROD Name* field editable

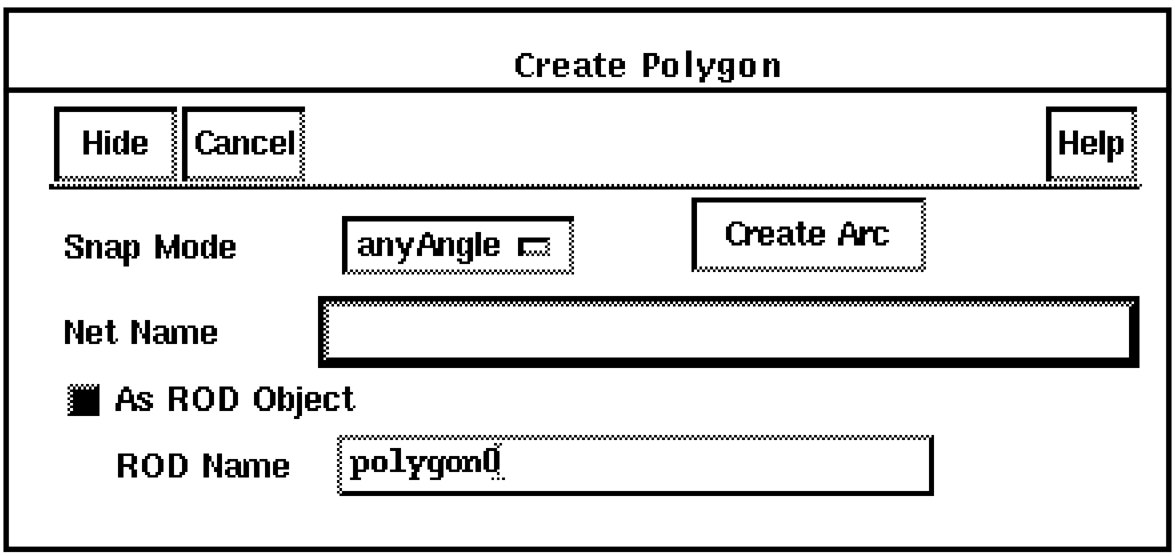
Once the polygons are created, you can edit them by typing ROD function commands in the CIW. These commands let you

- Access ROD objects by name through all levels of hierarchy
- Access ROD objects’ handle values through all levels of hierarchy

- Align ROD objects to each other or to specific coordinates
- Assign names to unnamed rectangles, polygons, lines, and paths
- Create complex parameterized cells more easily

See the *Virtuoso Relative Object Design User Guide* for information about ROD objects.

About the Create Polygon Form



Snap Mode limits how the cursor snaps when you create a polygon.

anyAngle creates lines at any angle.

diagonal creates lines parallel to the X or Y axis or at a 45-degree angle to the axes.

orthogonal creates lines parallel to the X or Y axis.

L90XFirst creates orthogonal two-segment lines and creates the first line in the X direction.

L90YFirst creates orthogonal two-segment lines and creates the first line in the Y direction.

Create Arc lets you create an arc in the polygon.

Net Name assigns the polygon to a specific net.

As ROD Object creates the polygon as a ROD object.

ROD Name assigns a name to a new polygon in the current layout cellview if *As ROD Object* is on. The name must be unique in the cellview. If you do not edit this field, the system assigns a unique name consisting of the prefix `polygon`, followed by a number. For example, the first ROD polygon would be named `polygon0`, the second, `polygon1`, and so on.

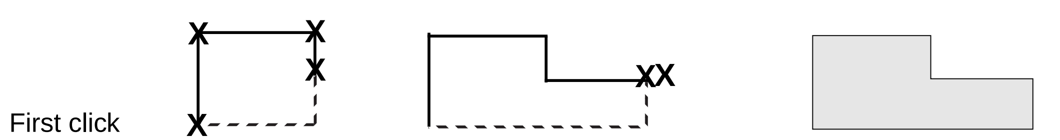
Creating a Polygon

To create a polygon,

1. In the LSW, click on the layer you want.
2. Choose *Create – Polygon* [`Shift-p`].
3. In the *Net Name* field, type the net name you want the polygon to be associated with.
4. If you want to create a ROD object polygon, turn on *As ROD Object*.

   The *ROD Name* field becomes editable, allowing you to change the name from the default name that is displayed. If you do not change the name, the default name is assigned to the polygon.
5. Click to enter the first point.
6. Move the cursor and click to enter the next point.
7. Continue to move the cursor and click to enter points.

   As you enter each point, a dotted line shows how the layout editor will close the polygon if you stop at that point.
8. Double-click to finish the polygon.



Each click creates another segment.

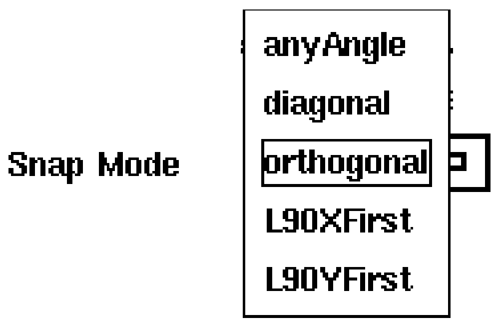
Double-click to close the polygon.

The finished polygon.

Creating Polygons Using Different Snap Modes

To create polygons using the snap mode to control the angle of the edges,

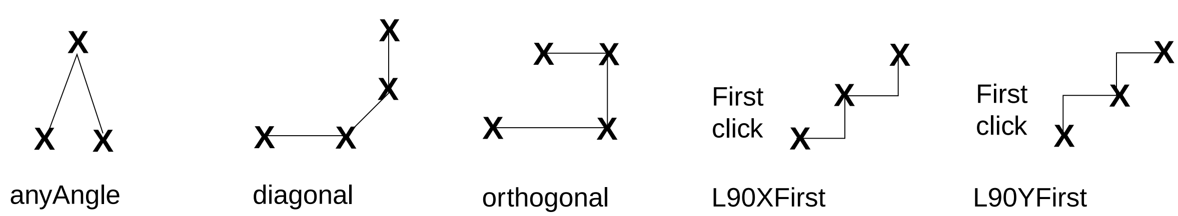
1. Open the Create Polygon form.
2. In the *Snap Mode* cyclic field, choose a value.



Snap Mode settings are illustrated here.

If you chose an L90 mode, you can switch to the other L90 mode by clicking right.

Keep clicking right to toggle between *L90XFirst* and *L90YFirst.*

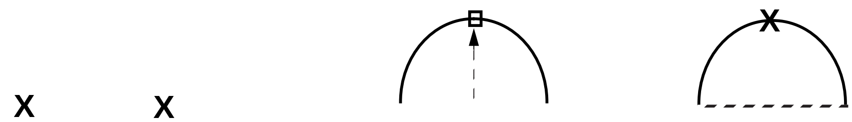


Creating an Arc in a Polygon

To create an arc in a polygon,

1. Open the Create Polygon form.
2. Click on *Create Arc* in the form.

3. Click to indicate the startpoint, the endpoint, and a radius point on the arc.



Click to enter the startpoint. | Click to enter the endpoint. | Move the cursor to see the arc. | Click to create the arc.

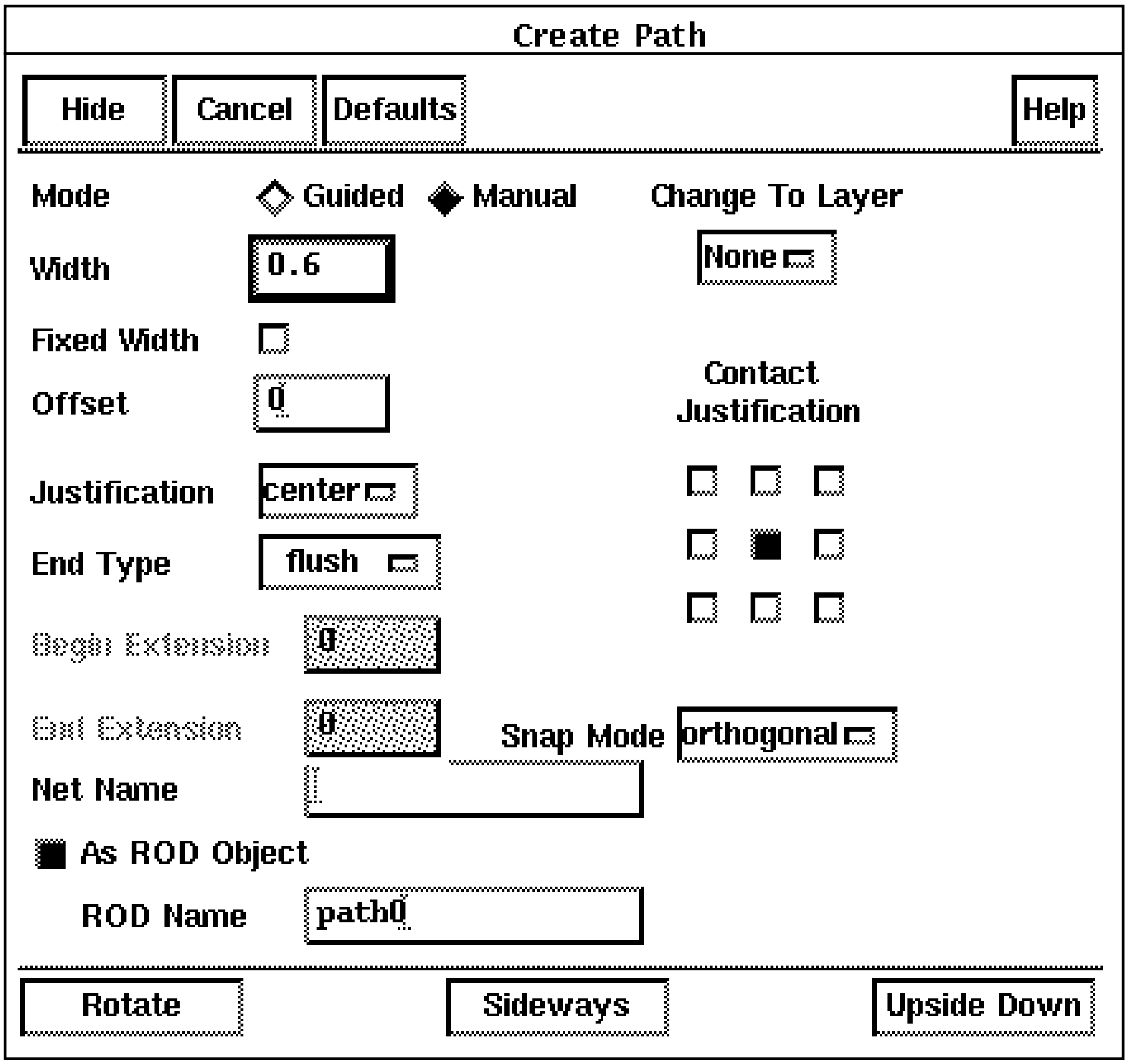
If you have already entered some points of a polygon, the last point you clicked is the startpoint of the arc.

Note: You can also use the *Modify Corner* command to curve the corner of a polygon.

Creating Paths

The *Create Path* command lets you create paths. A path is the course over which electrical currents flow in circuits. This is also known as routing.

About the Create Path Form



Mode

Guided uses the `minSpacing` rules in the technology file to automatically avoid obstacles while creating paths. In *Guided Mode*, the Create Path form changes. Unnecessary fields are removed and new options become available.

Manual does not follow the `minSpacing` rules in the technology file. You create the paths manually.

Width specifies the path width in user units.

Fixed Width, when on, uses the width you specify in the *Width* field for all segments of the path. When off, the path width remains set at the last setting until you click the *Defaults* button. The *Defaults* setting uses the path width defined in the technology file for the current layer.

Offset offsets the line you use to create the path from the final path. To control which edge of the path is offset, set the *Justification* cyclic field.

Justification controls which edge of the path you create: left, center, or right. If you set a path offset, *Justification* controls which edge of the path is offset from the line you create.

End Type controls how the path ends are created.

> **flush:** the path ends and path points end at the same points.
>
> **offset:** the path ends extend from the path points by one half the path width.
>
> **octagon:** the path ends extend from the path points by one half the path width, creating an octagonal shape.
>
> **variable:** the path ends extend from the path points by whatever value you enter in the *Begin Extension* and *End Extension* fields.

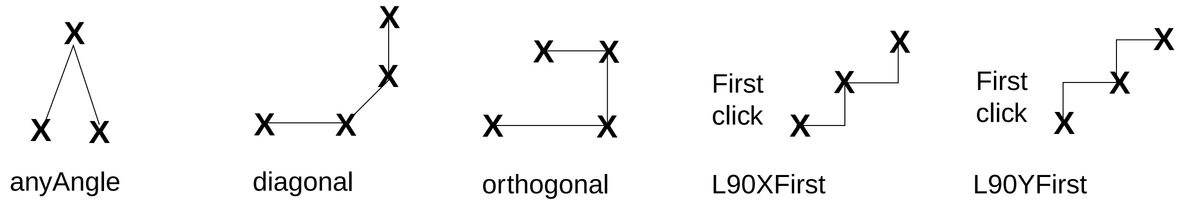
Begin Extension and **End Extension,** when *End Type* is set to *variable*, let you enter a beginning and an ending extension in user units.

Change To Layer lets you automatically place a contact and change layers while creating a path. While creating the path, you can select the new layer by selecting the layer from the cyclic menu. If the Create Path form is not open, you can change layers by clicking `Control` right in your design to cycle through the layers in the menu.

Contact Justification is the field where you click on one of the *Contact Justification* radio buttons to control which edge or corner of the contact connects to the path end. Your technology file must include definitions of all contacts.

Snap Mode controls how path segments snap to the grid.

> *Snap Mode* examples:



> To toggle between *L90XFirst* and *L90YFirst* while you are creating a path, click right.

Net Name assigns the path to a specific net.

As ROD Object creates the path as a ROD object.

> **ROD Name** assigns a name to the new path in the current layout cellview if *As ROD Object* is on. The name must be unique in the cellview. If you do not edit this field, the system assigns a unique name consisting of the prefix `path`, followed by a number. For example, the first ROD path would be named `path0`, the second, `path1`, and so on.

Rotate turns the contact 90 degrees counterclockwise. You can also click right to rotate the contact.

Sideways mirrors the contact along the X axis.

Upside Down mirrors the contact along the Y axis.

Guided Mode creates paths that follow the `minSpacing` rules defined in the technology file. You must create the path within the specified routing boundary or the `minSpacing` rules might not be maintained. When on, *Routing boundaries* appear.

Routing boundaries

> **auto boundary** (default) optimizes *Guided Path* within the area immediately surrounding the cursor as you move the cursor to create the path. This is the best option for large designs.
>
> **window** optimizes *Guided Path* in the area displayed within the cellview window.
>
> **prBoundary or cellBoundary** optimizes *Guided Path* in the *prBoundary* or *cellBoundary* layers. These layers are defined in the technology file.

The following fields are removed from the Create Path form when *Guided Mode* is on: *Offset*, *Justification*, *End Type*, *Begin Extension*, *End Extension*, and *Snap Mode*.

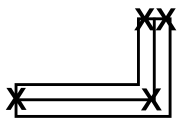
Creating a Path

To create a path,

1. In the LSW, click on the layer you want to use.
2. Choose *Create – Path*.
3. Enter the first point by clicking in the cellview.
4. Move the cursor to the next point and click.
5. Continue to move the cursor and click to enter points.

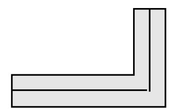
6. Double-click to finish the path.

First click



Double-click on the last point.

Each click creates another segment.

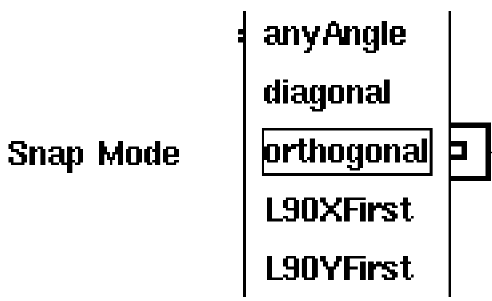


The finished path.

Changing Path Shape

To change the path shape as you create a path,

1. Open the Create Path form.
2. In the *Snap Mode* cyclic field, choose a value.



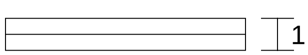
Press left to choose the *Snap Mode.*

Changing Path Width

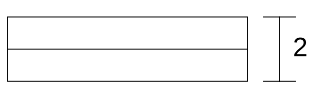
To change the width while creating a path,

1. Open the Create Path form.
2. In the *Width* field, type in the new width.

   The width setting is calculated in user units, which are usually microns. The default path width is the default width for the entry layer as defined in the technology file.



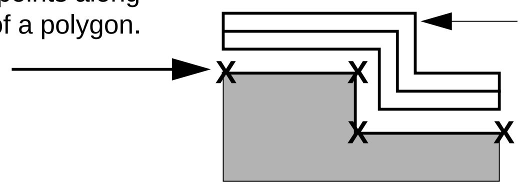
Width set to 1 user unit



Width changed to 2 user units

Creating Offset Paths

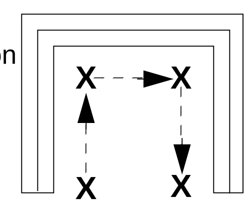
You can create a path that is offset from the points you enter.



1. Open the Create Path form.
2. In the *Offset* field, type a value.

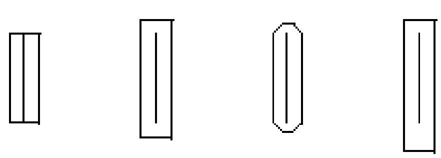
   The edge of the path is offset by the amount you type in the *Offset* field. For example if you type `0.5`, the path is created 0.5 microns from the points you enter.
3. In the *Justification* cyclic field, select the edge of the path you want to offset.

   The layout editor offsets the path to the left of the points you enter from the bottom of the cellview to the top.



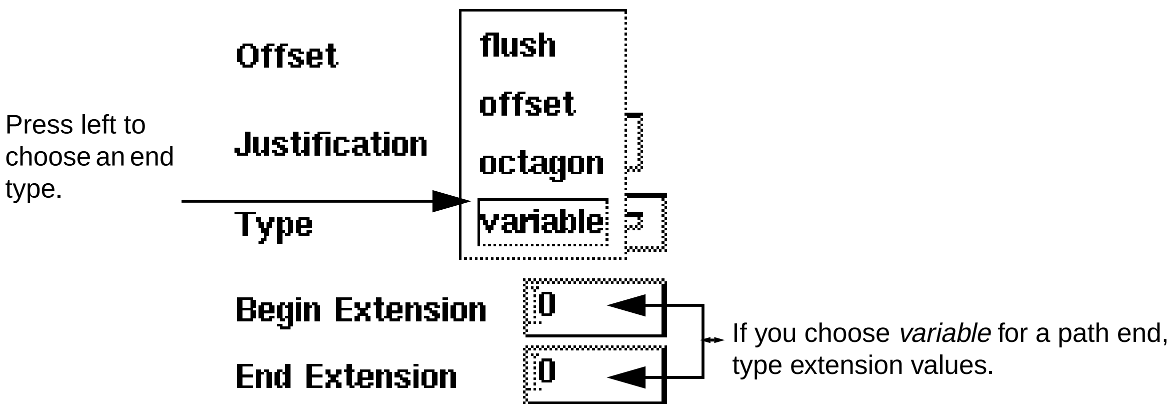
Changing Path Ends

The path *End Type* settings change the shape and spacing of path ends.



1. Open the Create Path form.

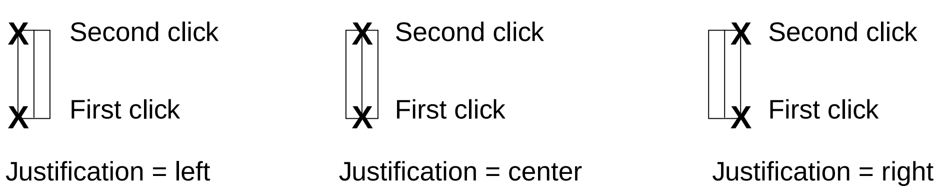
2. In the *End Type* cyclic field, choose an end type.



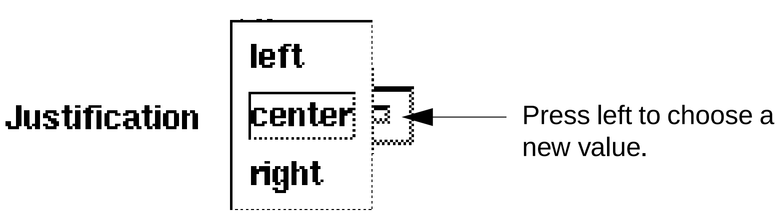
3. If you chose *variable*, type values in the *Begin Extension* and *End Extension* fields.

Setting Path Justification

The path justification settings control whether you enter points along the centerline or the edge of a path.



1. Open the Create Path form.
2. In the *Justification* cyclic field, choose a new value.



Displaying Path Centerlines or Borders

You can control how paths appear in the cellview.

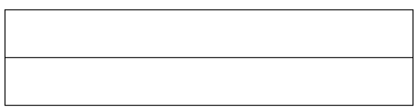
To display path centerlines or borders,

1. Choose *Options – Display* [e].

   The Display Options form appears.

2. Do one of the following:
   - Turn *Path Borders* on, which displays both the path centerline and its outer edges.
   - Turn *Path Borders* off, which displays only the path centerline.

3. Click *OK*.

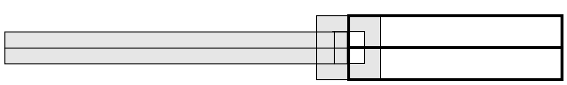


A path displayed with *Path Borders* set on.

The same path with *Path Borders* set off.

Path Stitching

You can create a path that switches from one layer to another, automatically placing an appropriate contact at the point where the layer changes. This function is often called *path stitching*.



While path stitching, you can automatically place a contact and switch to another layer.

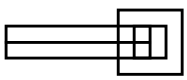
1. Open the Create Path form.
2. Click the first point of the path.

3. In the *Change To Layer* cyclic field, select the layer you want. If the Create Path form is not open, you can change layers by pressing `Control` and the right mouse button in your design to cycle through the layers listed in the cyclic field.

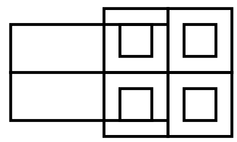
   If no layers appear in the *Change To Layer* cyclic field, no contacts are defined in the technology file for the current layer and you cannot stitch the path. For more information about defining contacts, see “Defining Contacts” on page 219.

4. Move the pointer back into the cellview.

   The cursor drags the outline of a contact or a contact array.

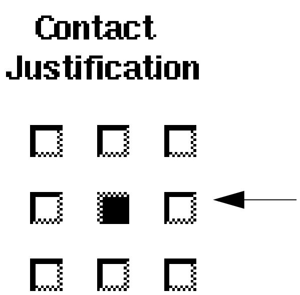


A single contact



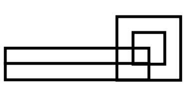
A contact array

5. If you want to change the contact alignment, click on the *Contact Justification* button in the form that represents where you want the contact to align.

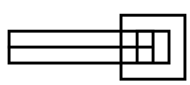


Click where you want the contact to align. For example, click here to set the origin to right center.

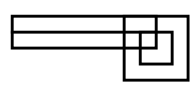
6. If you want to change how the contact snaps to the path, change the path justification.



Justification = left

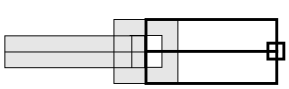


Justification = center



Justification = right

7. Click to place the contact.



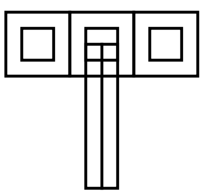
The previous path segment and the contact are saved. A new path segment follows the cursor.

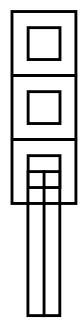
Rotating Contacts During Path Stitching

You can rotate the contacts you place while you are path stitching.

1. Open the Create Path form.
2. Click the first point of the path.
3. In the *Change To Layer* cyclic field, select the layer you want.
4. Move the pointer back into the cellview.

   The cursor drags the outline of a contact or a contact array.
5. Click *Rotate* or press `Shift` and click right to rotate the contact.
6. Move the pointer.


Before rotation


After rotation

7. Click to place the contact.

User-Defined Contacts

You can create your own contacts and have them appear in the *Change To Layer* cyclic field. To do this, you must edit either the `syEnhContact` or `syContact` device class when defining and declaring your contacts. You can use the syntax format in the `syEnhContact.tf` sample file to define your custom contacts. The class and formal parameters must be the same name as that used for creating a `symContactDevice` device class. You can add parameters, but do not change the names of the existing ones.

Contacts created in either the `syEnhContact` or `syContact` device class also appear in the *Device Class* and *Device Type* fields in the Create Device form and in the *Contact Type* field in the Create Contact form.

For information about how to edit the technology file and add customized contacts, read the “Devices” chapter in the *Technology File and Display Resource File User Guide*.

Defining Contacts

To define contacts, do the following:

- Use the Cadence® predefined system contacts described in the sample file

  your_install_dir/dfII/samples/techfile/devices.tf

- Create your own contacts described in the sample file

  your_install_dir/dfII/samples/techfile/syEnhContact.tf

Creating Paths Using Guided Path

To create a path that follows the minSpacing rules in the technology file, use the *Guided Path* option. With this option turned on, the path avoids obstacles and takes the shortest route while it is being created from one point to the next.

Guided Path plots a ghost path as you move your cursor. The ghost path is displayed on a highlighting layer and shows you what the path will look like if you click the mouse where you have moved your cursor.

With the *Guided Path* option,

- The path follows the minSpacing rules for the current layer
- Path stitching can be performed
- Pins attached to the path must be rectangular, either shape or symbolic pins
- The path is not completed if *Guided Path* cannot find a route that can follow the minSpacing rules

Starting Guided Path

To start *Guided Path,*

1. Open the Create Path form.
2. Set *Mode* to *Guided*.

The Create Path form changes in the following ways:

- *Routing boundary* options become visible
- Unnecessary fields are removed.

If you do not use command options forms while running commands such as the *Create Path* command, you can start *Guided Path* by doing one of the following:

- Press F8 to toggle *Guided Path* on and off. A message in the CIW lets you know the status of *Guided Path Create*.
- Start *Guided Path* in the Layout Editor Options form by turning *Guided Path Create* on.

Using Guided Path

Guided Path examines objects on the current entry layer as you create a path. As you create your path, *Guided Path* reads the `minSpacing` rules for the objects on that layer and plots the path to avoid objects. You can define the area containing shapes for *Guided Path* to examine by using the *Routing Boundary* options:

Auto Boundary (default) optimizes *Guided Path* within the area immediately surrounding the ghost path as you move your cursor to create the path. This option might be the best one to use for large designs.

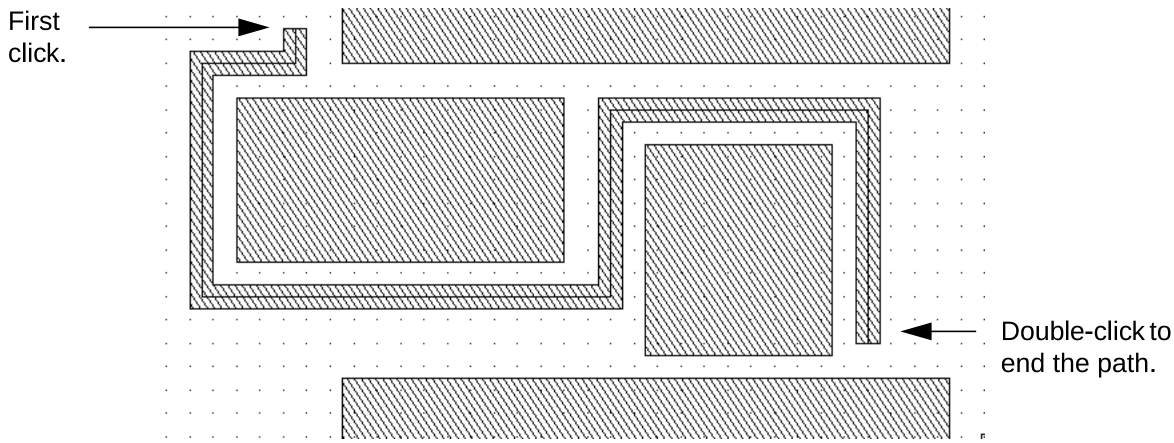
Window optimizes *Guided Path* in the area displayed within the cellview window.

prBoundary or cellBoundary optimizes *Guided Path* inside the *prBoundary* or *cellBoundary* layers.

To create a path in the guided mode,

1. In the LSW, click on the layer you want.
2. Choose *Create – Path*.
3. Set *Mode* to *Guided*.
4. Choose the *routing boundary*.
5. Click in the cellview where you want to start your path.

6. Move the cursor to where you want to end the path and double-click. The path is automatically created between your first and second clicks and avoids obstacles.



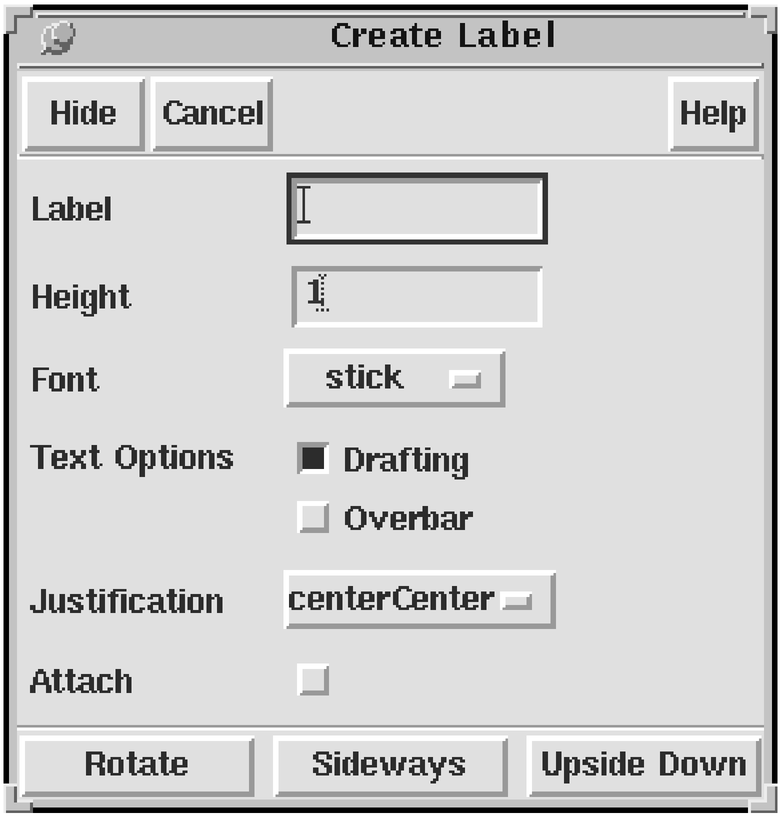
Using Guided Path with Maintain Connections

If the path is attached to a pin, you can move the pin and have the path reattach to it automatically using the *Maintain Connections with Guided Path* option. This option is set in the Layout Editor Options form.

Creating Labels

The *Create Label* command lets you enter text in the cellview.

About the Create Label Form



Label sets the text you want to appear.

Height sets the height of the label in user units (usually microns).

Font sets the text style of the label.

Text Options

> **Drafting** prevents the label from being rotated more than 90 degrees.
>
> **Overbar** determines how text strings containing underscore characters are displayed.

Justification sets the location of the label origin. The origin appears as a small square on the label when you place or select it.

Attach lets you attach the label to an object in the cellview.

Rotate turns the label 90 degrees counterclockwise.

Sideways mirrors the label along the X axis. Drafting must be off for this button to work.

Upside Down mirrors the label along the Y axis. Drafting must be off for this button to work.

Placing and Attaching a Label

You can identify objects or portions of your design by adding labels. You typically attach labels to objects on a net, so the LVS (Layout Versus Schematic) program can verify the nets in your layout against those in your schematic.

To create and attach a label,

1. Choose *Create – Label*.

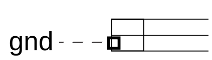
   The Create Label form appears.

2. In the *Label* field, type the text of the label.

   You can include spaces and special characters.

3. Choose a font.
4. Set *Attach* on.
5. Click where you want to place the label.
6. Click on the object to which you want to attach the label.

   A dashed line extends from the label to the object and disappears when you click on the object.



A dashed line shows the object to which you attach the label.

Rotating Labels

To rotate or mirror a label as you place it,

1. Set *Drafting* off in the Create Label form.

   When *Drafting* is on, you cannot rotate text more than 90 degrees.

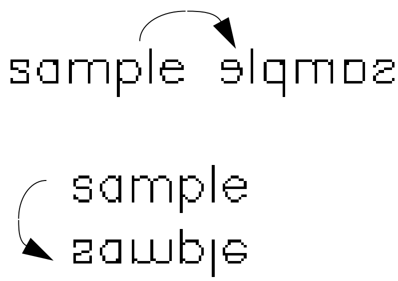
2. Do any of the following:
   - ❑ Click right to rotate the label 90 degrees.
   - ❑ Press `Shift` and click right to mirror the label.

- In the Label form, click on *Rotate*, *Sideways*, or *Upside Down*.

You can click on the rotate and mirror buttons, or click right, as many times as you want. Each time you click a button or click right, the label is further rotated or mirrored.

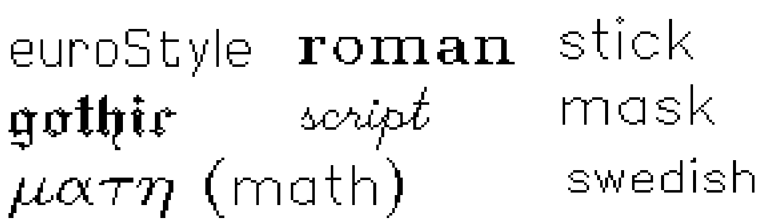
Label rotated sideways

Label rotated upside down



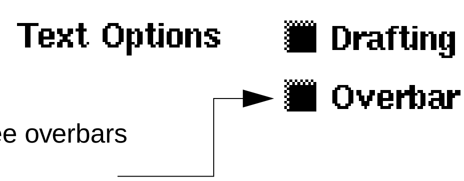
Choosing a Font

To choose your label font, select one of the fonts in the Create Label form. The font styles are shown below.



Using the Overbar Option

The overbar display option determines how text strings containing underscore characters are displayed in a layout window.



Set *Overbar* on to see overbars in text strings.

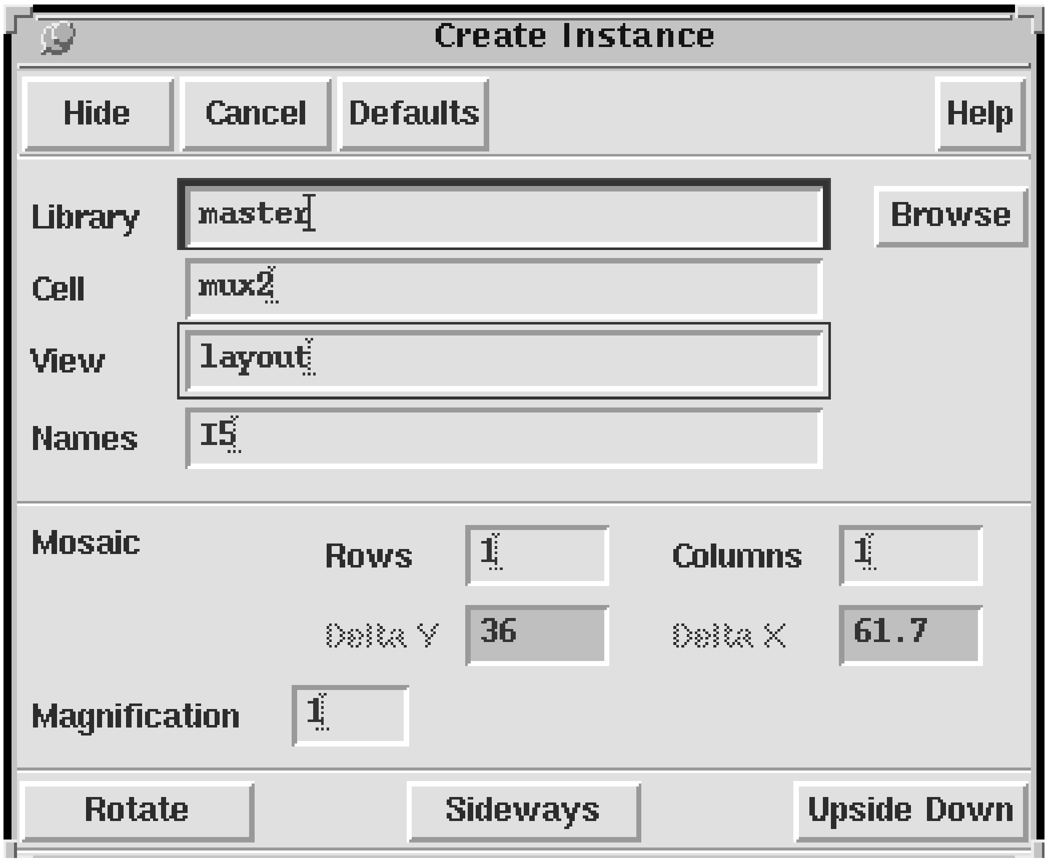
When overbar is off (default), the software displays underscore characters (_) as part of the text string. When overbar is on, the software interprets underscore characters in the text string name as toggle switches that control where overbars begin and end. Overbars appear above the text string, as shown in the examples.

Text String	**Appears in Layout Window As**
_abcde	$\overline{\text{abcde}}$
ab_cde	$\text{ab}\overline{\text{cde}}$
_abc_de	$\overline{\text{abc}}\text{de}$

Placing Instances and Arrays

An instance is a database object that represents a master cellview. You can have several instances of the same cellview in a design. The *Create Instance* command lets you place an instance of a cell into the current cellview.

About the Create Instance Form



Library, **Cell**, and **View** set the library, cell, and view names of the master cell you want to place as an instance in this cellview.

Browse lets you select the library, cell, and view names by clicking on them in the browser.

Names sets the name assigned to this instance. You can type any name unique to the cellview here or let the layout editor automatically assign instance names that begin with the letter `I`, followed by a number. You can enter multiple names (separated by a space) to place several instances of the same cell.

Mosaic

> **Rows** and **Columns** set the number of rows and columns in an array of instances.
>
> **Delta X** and **Delta Y** set the spacing between rows and columns in an array of instances.

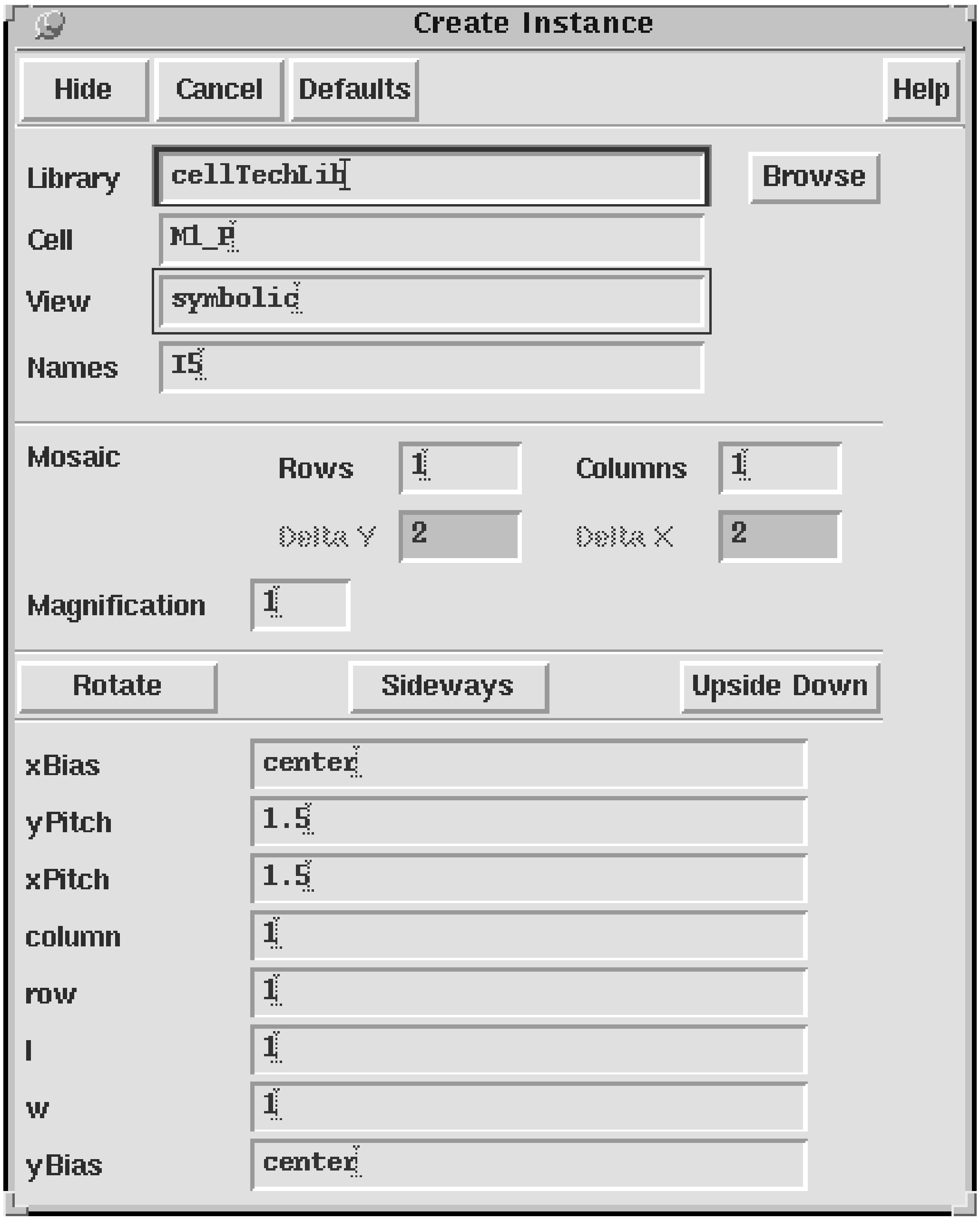
Magnification enlarges or reduces the size of the cell instance.

Rotate turns the instance 90 degrees counterclockwise.

Sideways mirrors the instance along the X axis.

Upside Down mirrors the instance along the Y axis.

If the master cell is a parameterized cell, fields for one or more parameters appear at the bottom of the form after you type the cell name and press `Tab`. You can change the layout of the cell instance by changing values for the parameters.



Placing a Cell Instance

To place a cell inside the current cellview,

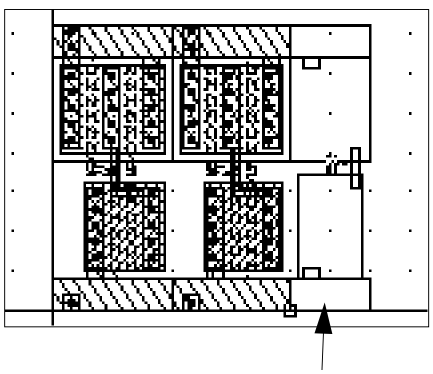
1. Choose *Create – Instance*.

   The Create Instance form appears.

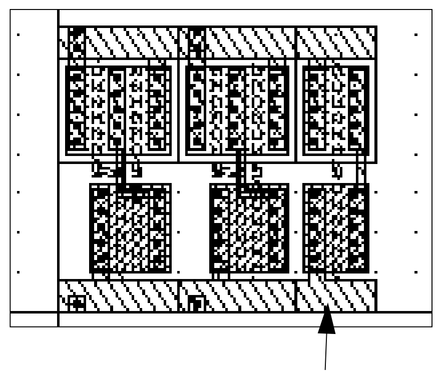
2. Fill in the *Library*, *Cell*, and *View* fields.
3. Move the cursor into the cellview.

   An outline of the cell you want to place follows the cursor.

4. Click where you want to place the instance.



Click to place the outline of the instance.



The placed cell instance

Naming Instances

The layout editor automatically assigns names to instances, starting with `I1` (Instance 1) and continuing with `I2`, `I3`, and so forth. If the instances are mosaics, the default names are `M1`, `M2`, and so forth.

Overriding Instance Names

To override the automatic naming of instances,

1. Open the Create Instance form.
2. Type one or more instance names into the *Names* field.

   Leave a space between each name.

3. In the cellview, click to place each instance.

   The first name in the form is assigned to the first instance you place. The next name is assigned to the next instance you place, and so forth.

   Each time you place an instance, its name disappears from the *Names* list.

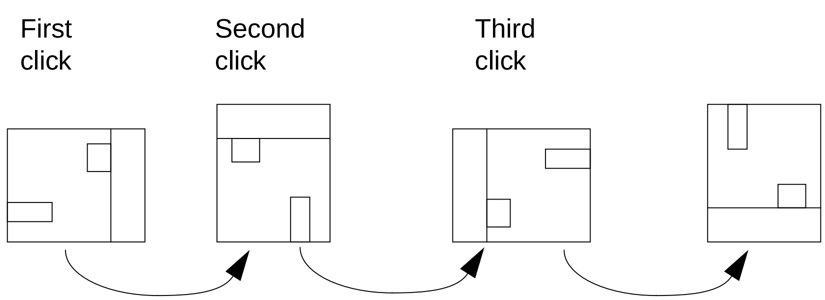
You can use the Display Options form to display instance names in the cellview.

Rotating and Mirroring Instances

To rotate instances as you place them,

➤ Click right.

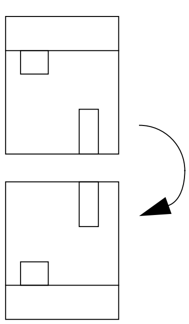
   Each time you click right, the instance or array rotates 90 degrees counterclockwise.



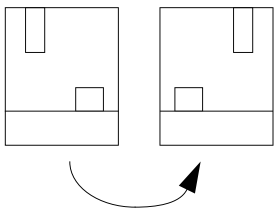
To mirror the instance,

➤ Press `Shift` and click right.

   At the first click, the instance mirrors along the X axis.



At the second click, the instance mirrors along the Y axis.



To use the Create Instance form to rotate or mirror instances,

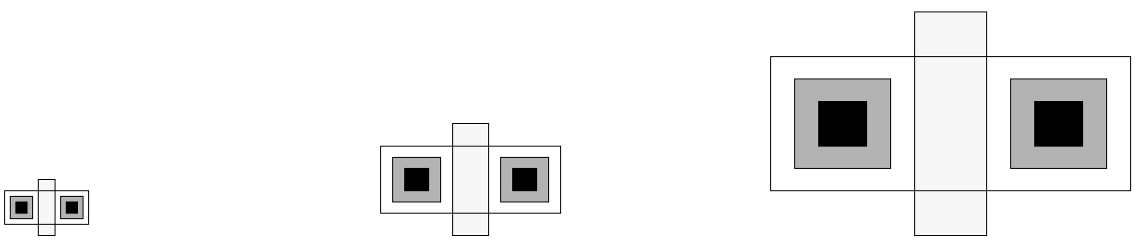
➤ Click *Rotate*, *Sideways*, or *Upside Down*.

Setting Magnification of Instances

To enlarge or reduce an instance as you place it,

➤ In the Create Instance form, type a value in the *Magnification* field.

The value is the factor by which the instance is enlarged or reduced. For example, 2 doubles the size of the instance, while 0.5 shrinks it to half its original size.



Magnification set to 0.5

Original (Magnification set to 1)

Magnification set to 2

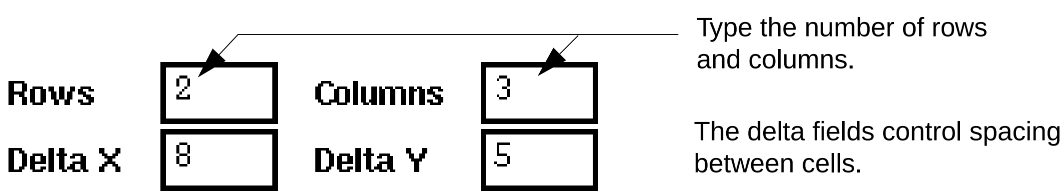
Placing an Array of Instances

You can use the *Create Instance* command to place many instances in an array (sometimes called a mosaic). The array is considered a single object in the database. It is assigned a single instance name, which is, by default, `M1`, `M2`, and so forth.

To place an array,

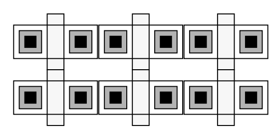
1. Open the Create Instance form.
2. In the *Rows* and *Columns* fields, type the number of rows and columns you want in the array.
3. To change the spacing between the rows or columns, type values for *Delta X* or *Delta Y.*

   By default, the fields are set to the width and length of the original cell, so that the edges of each instance touch but do not overlap.



4. Click in the cellview to place the array.

Click to place the array outline.



The placed array

Creating Pins

The *Create Pin* command lets you create pins manually or place symbolic pins to connect this cellview to another cell in your design hierarchy.

There are two kinds of pins:

- Shape pins, created by the Create Shape Pin form. A shape pin is a shape you create to represent a pin.
- Symbolic pins, created by the Create Symbolic Pin form. A symbolic pin is an instance of a predefined parameterized cell that represents a pin.

You can place a pin automatically or create a pin manually.

To create a pin manually,

1. Choose *Create – Pin*.

2. The Create Shape Pin form or the Create Symbolic Pin form opens. If the form that opens is not the one you want, click on *shape pin* or *sym pin*.

   You can specify which form will open by setting the environment variable `pinsAreSymbolic`. When set to `t`, the Create Symbolic Pin form opens; when `nil`, the Create Shape Pin form opens.

3. In the *Terminal Names* field, type the terminal name (the name of the net to which this pin connects).

4. Click on an *I/O Type* button to indicate the direction of the signal into or out of the pin.

5. Do one of the following:

- If you are creating a shape pin,
  - Set the *Mode* to *rectangle*, *dot*, or *polygon*.

    The system prompts you to create a rectangle or polygon.

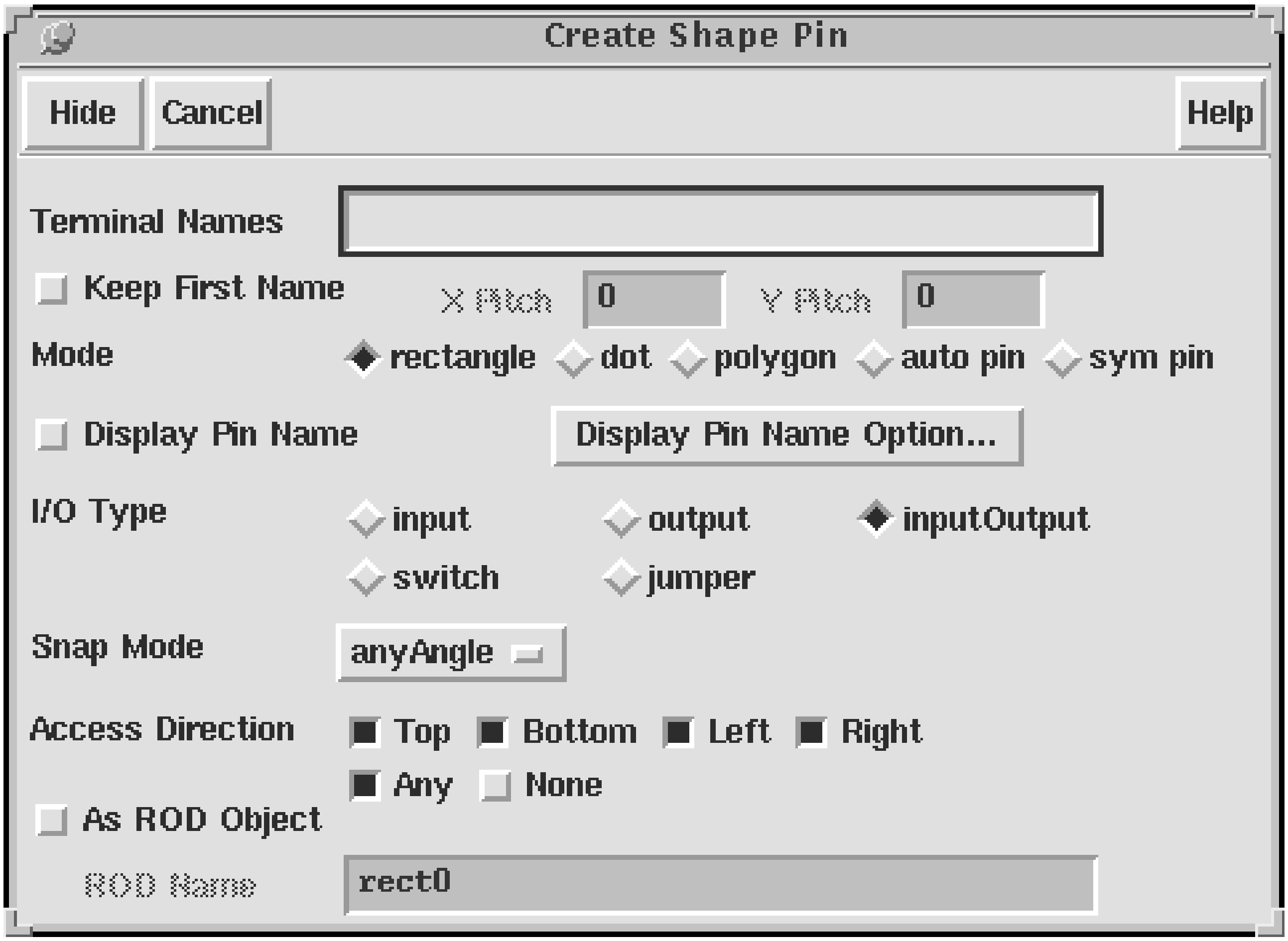
    Note: To create a zero-size dot pin, click twice on the same point.
  - If you are creating a rectangle pin, click on the *Access Direction* buttons to specify the direction in which the routing tools can connect routing to the pin.
  - Set the *Snap Mode* cyclic field.
  - Create the pin the same way you create a rectangle or polygon.
- If you are placing a symbolic pin manually,
  - Set *Mode* to *manual pin*.
  - Set the *Pin Type* cyclic field.
  - Type the width for the pin.
  - Click to place the pin.

About the Create Shape Pin Form

To open the Create Shape Pin form,

➤ Do one of the following:
  - Choose *Create – Pin.*
  - Press `Control-p`.

If the Create Symbolic Pin form appears, click on *Mode – shape pin*.



Create Shape Pin Form

Terminal Names assigns a name to this pin. Terminal names are used to identify the net to which this pin connects. You can create multiple pins by typing the terminal name for each pin and separating the names with spaces. If *Keep First Name* is off, one name is removed every time a pin is placed.

Keep First Name, when on, does not remove the first name in the *Terminal Names* field.

X Pitch sets the horizontal distance, in user units, between the center points of the pins. This field is editable when the terminal name represents a bus. For example, `data <0:7>`.

Y Pitch sets the vertical distance, in user units, between the center points of the pins. This field is editable when the terminal name represents a bus.

Mode specifies the shape of a manually created pin, turns on the automatic pin placement mode, or opens the Create Symbolic Pin form.

rectangle lets you create a rectangular pin.

dot lets you create a dot pin. You drag a rectangle to define the extent of the dot. Click twice on the same point for a zero-area dot pin.

polygon lets you create a polygon pin.

auto pin automatically places a rectangular pin at the ends of a path.

sym pin closes this form and opens the Create Symbolic Pin form, which lets you create symbolic pins.

Display Pin Name attaches a label, showing the terminal name, to the pin.

Display Pin Name Option opens the Pin Name Display form, which lets you set the font, height, justification, and orientation of the pin name.

I/O Type assigns a property used by routers to identify the direction of the signal into or out of this cellview. The signal can be input, output, inputOutput (bidirectional), switch (carries data either in or out, but not simultaneously), or jumper (passes data through this cellview).

Snap Mode limits how the cursor snaps when you create a polygon pin.

anyAngle creates lines at any angle.

diagonal creates lines parallel to the X or Y axis or at a 45-degree angle to the axes.

orthogonal creates lines parallel to the X or Y axis.

L90XFirst creates orthogonal two-segment lines and creates the first line in the X direction.

L90YFirst creates orthogonal two-segment lines and creates the first line in the Y direction.

Access Direction assigns a property used to identify the part of the pin to which routers can connect routing.

As ROD Object creates the pin as a ROD object.

ROD Name lets you assign a name to the new shape pin in the current layout cellview if *As ROD Object* is on. The name must be unique in the cellview. If you do not edit this field, the system assigns a unique name consisting of the prefix `rect`, `dot`, or `polygon`, depending on the type of shape pin you are creating. The prefix is followed by a number.

For example, the first ROD rectangle pin would be named `rect0`, the second, `rect1`, and so on.

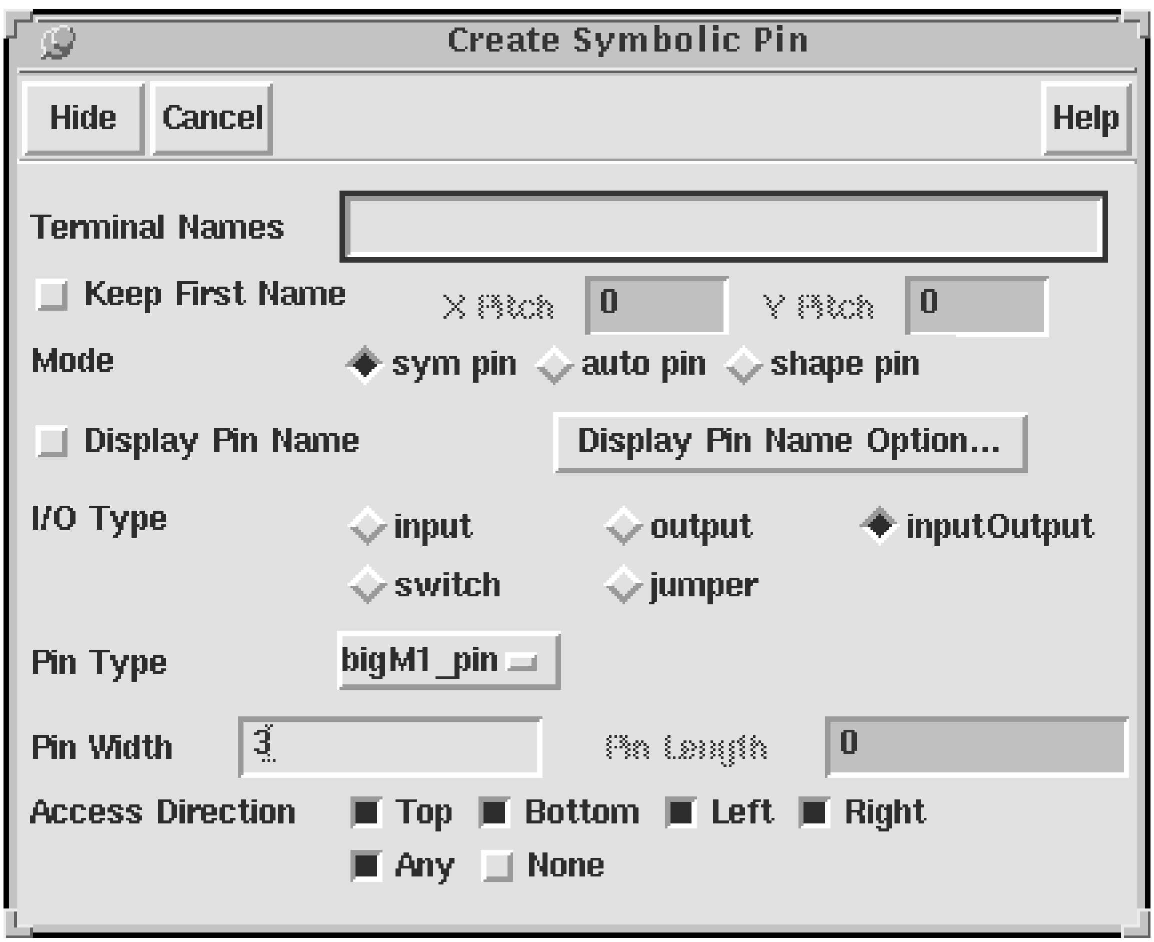
About the Create Symbolic Pin Form

To open the Create Symbolic Pin form,

➤ Do one of the following:

- Choose *Create – Pin*.
- Press `Control-p`.

If the Create Shape Pin form appears, click on *Mode – sym pin*.



Terminal Names assigns a name to this pin. Used by the layout editor, Layout Versus Schematic (LVS) program, and routers to identify the net to which this pin connects. You can create multiple pins by typing the terminal name for each pin and separating the names with spaces. If *Keep First Name* is not on, one name is removed every time a pin is placed.

Keep First Name does not remove the first name in the *Terminal Names* field.

X Pitch sets the horizontal distance, in user units, between the center points of the pins. This field is editable when the terminal name represents a bus. For example, `data <0:7>`.

Y Pitch sets the vertical distance, in user units, between the center points of the pins. This field is editable when the terminal name represents a bus.

Mode specifies whether you want manual or automatic pin placement mode or lets you open the Create Shape Pin form.

> **manual pin** lets you change *Pin Type, Pin Width*, and *Access Direction* and manually place the pin.
>
> **auto pin** lets you automatically place the pin on a path.
>
> **shape pin** removes this form and opens the Create Shape Pin form.

Display Pin Name attaches a label to the pin, showing the terminal name.

Display Pin Name Option opens the Pin Name Display form, which lets you set the font, height, justification, and orientation of the pin name.

I/O Type assigns a property used by routers to identify the direction of the signal into or out of this cellview. The signal can be input, output, inputOutput (bidirectional), switch (carries data either in or out, but not simultaneously), or jumper (passes data through this cellview).

Pin Type sets the type for the pin.

Pin Width sets the width of the pin.

Pin Length sets the length of the pin.

Access Direction assigns a property used to identify the part of the pin to which routers can connect routing.

Creating Pins as ROD Objects

When you create a shape pin, you have the option of designating it as a relative object design (ROD) object. The ROD object contains information about the pin, including its name and database ID.

To create a ROD object, turn on *As ROD Object* in the Create Shape Pin form. The *ROD Name* field becomes editable, and you can either use the default database name or type another name. ROD names can have spaces in them. For example, `rect 1a` is a valid ROD name.

Only one ROD object can be created at a time. Even though you can type several pin names in the *Terminal Names* field, only one name can be typed in the *ROD Name* field. This means that after each pin is created, you must type in the next name.

Once the pins are created, you can edit them by typing ROD function commands in the CIW. These commands let you

- Access ROD objects by name through all levels of hierarchy
- Access ROD objects' handle values through all levels of hierarchy
- Align ROD objects to each other or to specific coordinates
- Assign names to unnamed rectangles, polygons, lines, and paths
- Create complex parameterized cells more easily

See the *Virtuoso Relative Object Design User Guide* for information about ROD objects.

Using the rodAutoName Environment Variable

You can use the environment variable `rodAutoName` to set system default values in the Create Pin, Create Rectangle, and Create Polygon forms to

- Turn on *As ROD Object* so all shapes are created as ROD objects
- Make the *ROD Name* field editable, allowing you to either use the default name or type another name

The following example sets the Create Rectangle, Create Pin, and Create Polygon forms to create shapes as ROD objects. In the Command Interpreter Window (CIW), type

```
envSetVal("layout" "rodAutoName" 'string "rectangle pin polygon")
```

To turn off *As ROD Object* in one or all of the forms listed in the `rodAutoName` variable, do one of the following:

- Turn off *As ROD Object* in any of the forms
- Retype the `envSetVal` command, excluding the type of object you no longer want to create as ROD objects

For example, if you turn off *As ROD Object* in the Create Polygon form, the following is true for `rodAutoName`:

```
envGetVal("layout" "rodAutoName")
```

```
"rectangle pin"
```

You can also set the `rodAutoName` variable in your `.cdsenv` file, so the layout editor defaults to the options you want at startup. The syntax to use in your `.cdsenv` file is

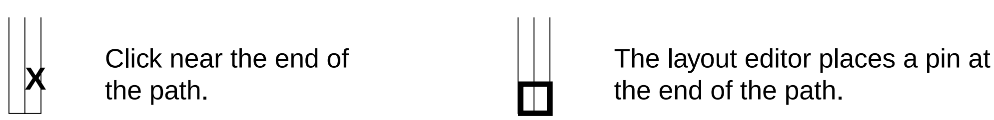
```
layout rodAutoName string "rectangle pin polygon"
```

Placing a Pin on a Path Automatically

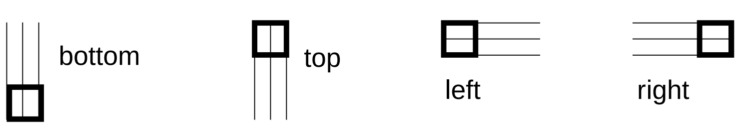
The layout editor can create a pin at the end of a path automatically. This feature works only on path ends parallel to the X or Y axis.

1. In the Create Shape Pin form or the Create Symbolic Pin form, set *Mode* to *auto pin*.
2. Click near the path end.

   The layout editor places a square pin at the end of the path nearest to the point you clicked. The pin width matches the width of the path and is placed on the same layer as the path.



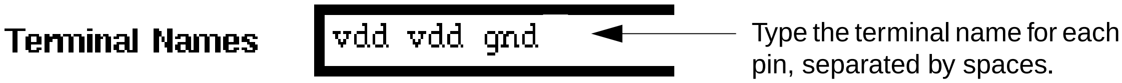
   When you place an auto pin, the layout editor automatically sets the access direction, no matter what setting appears in the form.



Creating Multiple Pins

Each pin you create needs a terminal name (the name of the net to which the pin connects). You can create several pins quickly by typing several terminal names.

1. In the Create Shape Pin form or the Create Symbolic Pin form, type the terminal name for each pin you want to create, leaving a space between each name.



   The terminal name does not have to be unique. In the above example, `vdd` is used twice, to create two pins that connect to the vdd net.

2. If you want to create a label of the terminal name, set *Display Pin Name* on.
3. In the layout window, click to create or place the first pin.

   The first name in the *Terminal Names* field is assigned to this pin.

4. If *Display Pin Name* is on, click to place the pin name.
5. Continue creating or placing pins.

   Each pin is assigned the next name in the form.

You can apply net expression properties to pins.

Placing Pin Arrays

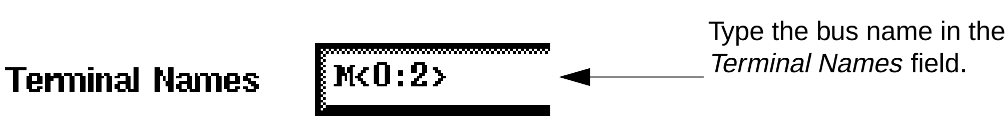
A pin array is one line of pins with a single origin. You might create such an array if you have many nets bundled together into a bus. The pins in the array are placed for every bit of the bus and are spaced according to the *XPitch* and *YPitch* coordinates.

To place a pin array,

1. In either the Create Shape Pin form or the Create Symbolic Pin form, type the following in the *Terminal Name* field:

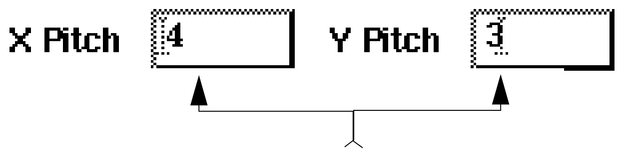
   baseName<*x*:*x*>

   where *baseName* = `M` and `x:x` = `0:2`



For bus naming information, see the naming conventions section in *Virtuoso Schematic Composer User Guide.*

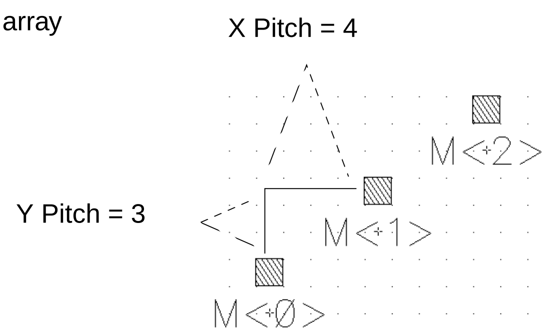
2. In the *XPitch* and *YPitch* fields, type the pin location coordinates.



Type the location for the pin array.

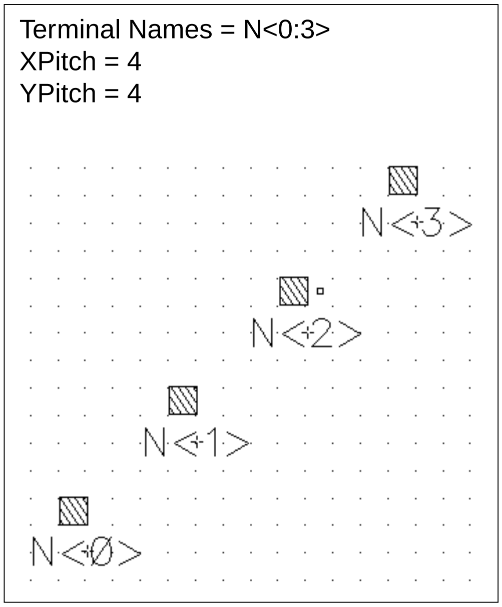
3. Move the cursor into the cellview window and place the first pin where you want to start the array. The rest of the pins are automatically placed at the designated X and Y locations for every bit of the bus. They are measured from the center point of the previously placed pin.

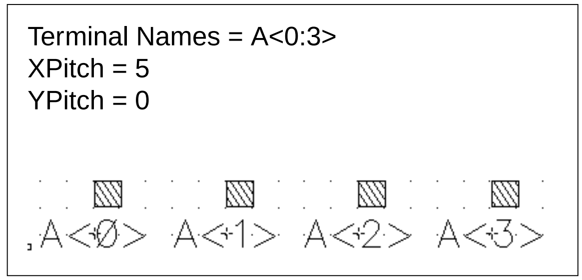
The resulting pin array



You can apply net expression properties to pins in a pin array.

Examples of Pin Arrays with Different X and Y Pitch





Displaying Pin Names

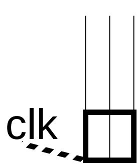
To display the pin name,

1. In the Create Shape Pin form or the Create Symbolic Pin form, set *Display Pin Name* on.

   Note: Creating a pin with a label turns on the display of all pin labels.

2. In the cellview window, click to create the pin.

   After you create the pin, you see an outline of the pin name.



The dotted line shows that the name will be attached to the pin.

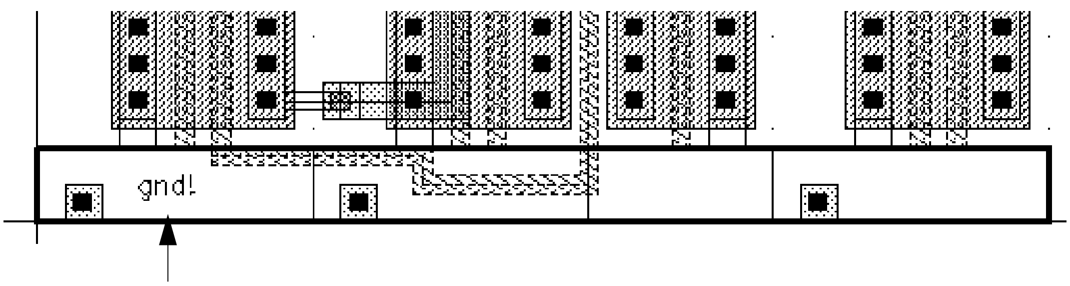
3. Double-click to place the pin name.
4. Click on *Display Pin Name Option* and use the Pin Name Display form if you want to change the font characteristics.

Note: If *Pin Names* in the Display Options form is not set on, none of the pin labels in the cellview are visible. To display all pin names at any time, set *Pin Names* on in the Display Options form.

Connecting Hierarchy with Pins

To indicate where and how the layout in a cell connects to another cell within a design hierarchy, you create a pin to do the following:

- Show where an instance of a cellview can connect to routing or to other instances
- Show which net a pin is on
- Control the direction in which the Cadence routing tools can connect routing to the cellview instance

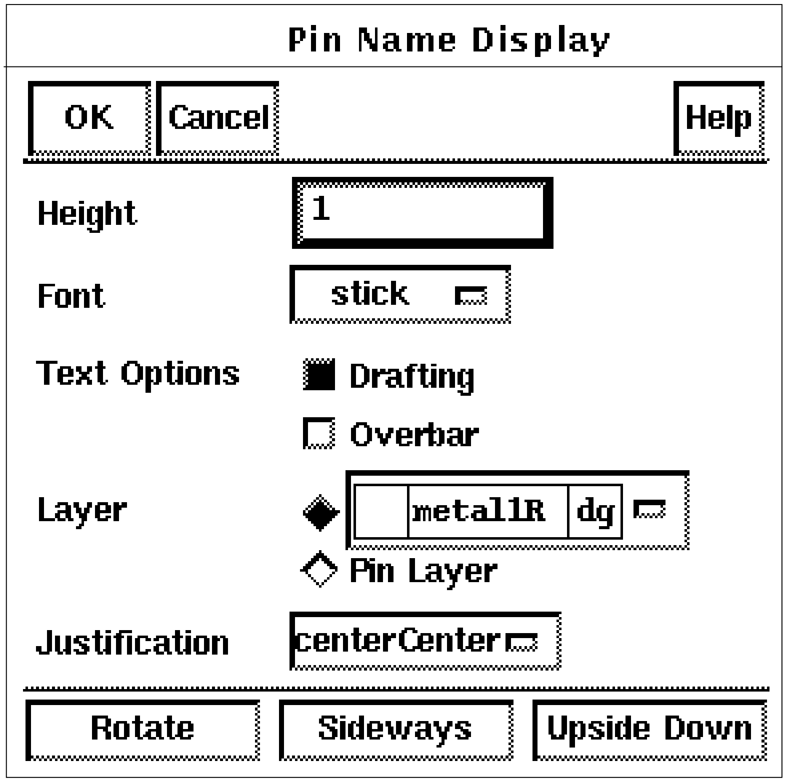


The gnd! pin (bold outline) shows where routing for the gnd! net can connect to any instance of this cellview.

Pin Name Characteristics

About the Pin Name Display Form

The Pin Name Display form appears after you click *Display Pin Name Options* in the Create Shape Pin form or the Create Symbolic Pin form. The *Display Pin Name Options* command lets you set how pin names appear. The Layout Versus Schematic (LVS) program uses this name to identify objects on a net when comparing nets in a schematic against a layout.



Height sets the height of the name in user units (usually microns).

Font sets the text style of the name.

Text Options

> **Drafting** prevents the label from being rotated more than 90 degrees.
>
> **Overbar** determines how text strings containing underscore characters are displayed. See "Using the Overbar Option" on page 224.

Layer sets the layer for the pin name.

Pin Layer puts the pin name on the same layer as the pin being created.

Justification sets the location of the name origin. The origin appears as a small square on the name when you place or select it.

Rotate turns the name 90 degrees counterclockwise.

Sideways mirrors the name along the X axis.

Note: Drafting must be off for this button to work.

Upside Down mirrors the name along the Y axis.

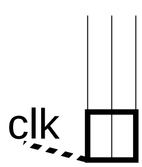
Note: Drafting must be off for this button to work.

Changing Pin Name Characteristics

To specify pin name characteristics,

1. In the Create Shape Pin form or the Create Symbolic Pin form, set *Display Pin Name* on.
2. Click *Display Pin Name Options*.
3. Fill in the Pin Name Display form.
4. In the cellview window, click to create the pin.

   After you create the pin, you see an outline of the pin name.



The dotted line shows that the name will be attached to the pin.

5. Double-click to place the pin name.

Setting the Pin Name Layer

To set the pin name layer,

1. In the Create Shape Pin form or the Create Symbolic Pin form, set *Display Pin Name* on.
2. Click *Display Pin Name Options*.
3. In the Pin Name Display form, do one of the following:
   - To choose a layer for the pin name, turn on the *Layer* button and choose a layer from the cyclic field.
   - To place the pin name on the same layer as the pin being created, set *Pin Layer* on.

4. Click *OK*.

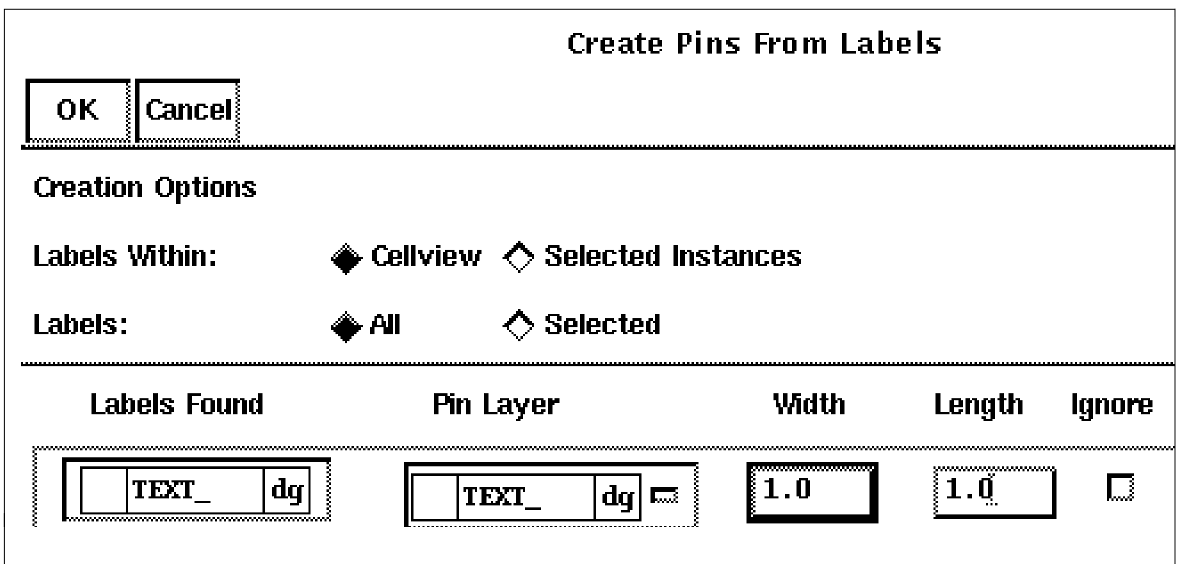
Creating Pins from Labels

The *Create Pins From Labels* command lets you create pins from text labels in your layout cellview. This command creates pins with terminal names matching labels on a specified text layer with pin dimensions that you specify, centered on the origin of your text label.

Pin names will differ from text labels if the text label contains a colon (:). When this occurs, the colon and trailing text are not part of the pin name. For example, given the text label VDD:P,28, the resulting pin name is simply VDD.

If a pin already exists for a text label you have selected, a new label will not be created for that text label. The pin and label are not attached to each other. If you move the pin, the label will not follow. To attach the label to the pin, use the *Edit – Other – Attach/Detach* command.

About the Create Pins From Labels Form



Creation Options control the scope of pin generation and which labels the system uses to create pins.

Labels Within:

Cellview creates pins only at the top-level cellview.

Selected Instances creates pins within the currently selected instances.

Labels:

All creates a pin for every label found in the current cellview or the selected instance.

Selected creates a pin for each label in the selected set.

Labels Found shows the layers of the selected text labels. One layer appears for each layer of text labels selected. This field is not editable.

Pin Layer displays the available layers for pins to be created on. Choose the pin layer from the list of layers.

Width sets the width of the pins.

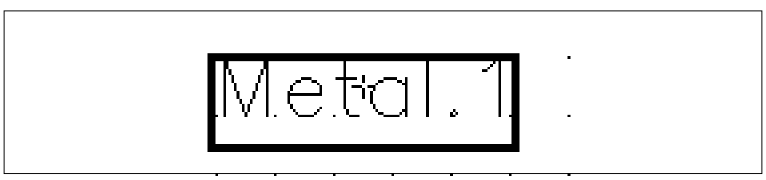
Length sets the length of the pins.

Ignore does not create pins for that layer.

Using the Create Pins From Labels Command

To create a pin from a label,

1. Select the text labels you want the pins created for.

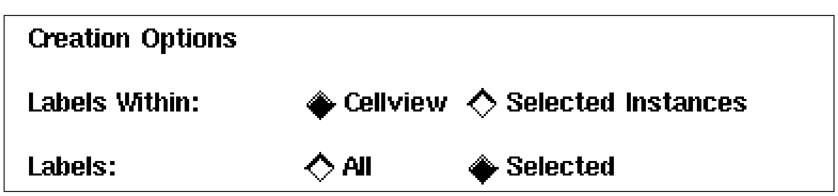


2. Choose *Create – Pins From Labels.*

   The Create Pins From Labels form appears.

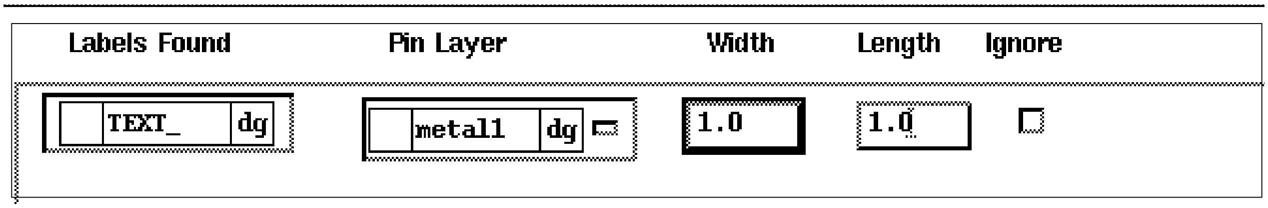
   One row of fields appears for each layer of labels you have selected.

3. In the Create Pins From Labels form, turn on *Labels Within: Cellview* and *Labels: Selected.*



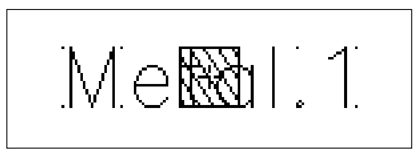
4. In the *Pin Layer cyclic* field, choose the entry layers you want your pins created on. If you selected labels on more than one layer, choose the appropriate pin layer for each label layer from the list of layers.

5. Type the width and length of your pin in the *Width* and *Length* fields.



6. Click *OK*.

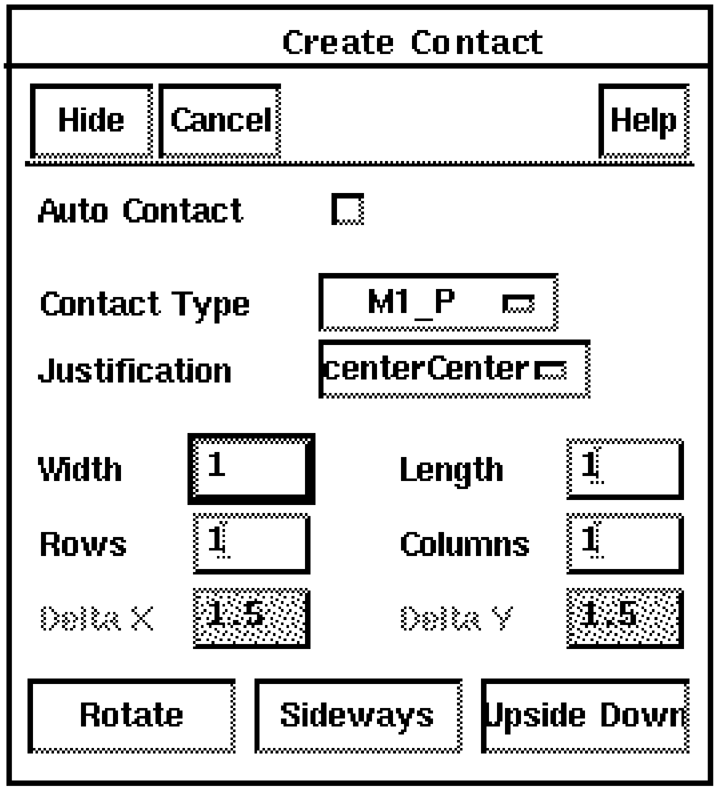
   The pins are placed in the center of the labels.



Creating Contacts

The *Create Contact* command lets you connect objects on two different layers within this cellview.

About the Create Contact Form



Auto Contact automatically places a contact on the intersection of two paths.

Contact Type selects the contact from the technology file and controls the layers on which the contact is entered.

Justification sets the origin of a contact array.

Width and **Length** set the width and length of the contact or via cut, in user units (typically microns).

Rows and **Columns** set the number of rows or the number of columns of contact cuts in a contact array. The default values for *Rows* and *Columns* are set in the technology file.

Delta X sets the horizontal distance, in user units, between the center points of the contacts when *Rows* is set to greater than 1.

Delta Y sets the vertical distance, in user units, between the center points of the contacts when *Columns* is set to greater than 1.

Rotate turns the contact 90 degrees counterclockwise. You can also click right to rotate the contact.

Sideways mirrors the contact along the X axis.

Upside Down mirrors the contact along the Y axis. You can also press `Shift` and click right to mirror the contact.

Placing a Contact

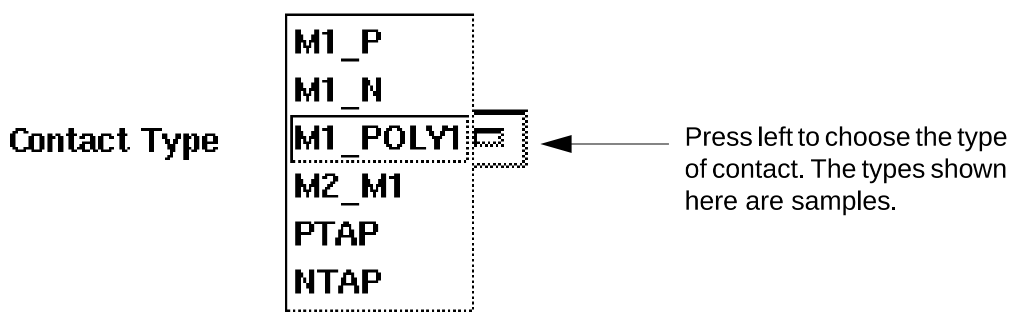
Contacts are special cells identified in your technology file that connect two layers inside a cellview.

To place a contact,

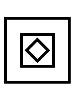
1. Choose *Create – Contact*.

   The Create Contact form appears.

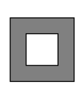
2. In the *Contact Type* cyclic field, choose the contact you want.



3. Click to place the contact.



The outline of the contact before you click. The diamond shows the contact origin.



The placed contact

You can create your own contacts and have them appear in the *Contact Type* cyclic field by adding them to the technology file. You must use either the `syEnhContact` or `syContact` device class when defining and declaring your custom contacts. The class and formal parameters must be the same name as that used for creating a `symContactDevice` device class. Parameters can be added, but the names of existing parameters must not be changed. Use the following syntax to define your custom contact:

`*your_install_dir*/tools/dfII/samples/techfile/syEnhContact.tf`

Contacts created in either `syEnhContact` or `syContact` device classes also appear in the *Device Class* and *Device Type* fields in the Create Device form and in the *Change To Layer* field of the Create Path form.

For information about how to edit the technology file and add your customized contacts, read about defining devices in the *Technology File and Display Resource File User Guide*.

Placing an Array of Contacts

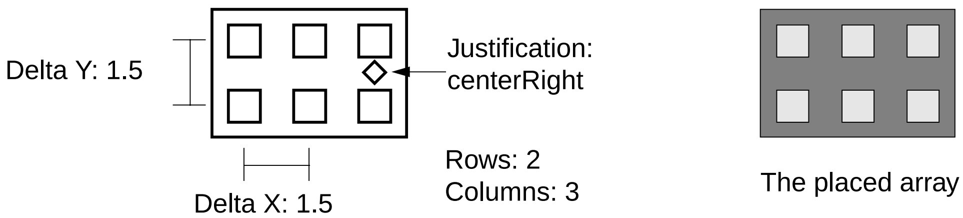
An array of contacts is a group of contacts with a single origin. You might create such an array to connect two wide paths or to create part of a large transistor.

To place an array of contacts,

1. Type the number of rows and columns in the Create Contact form.
2. Type in new values for *Delta X* and *Delta Y* if you want to change the spacing between the center of each contact cut.

3. Choose a new *Justification* setting to change the array origin.

4. Click where you want to place the contact array.

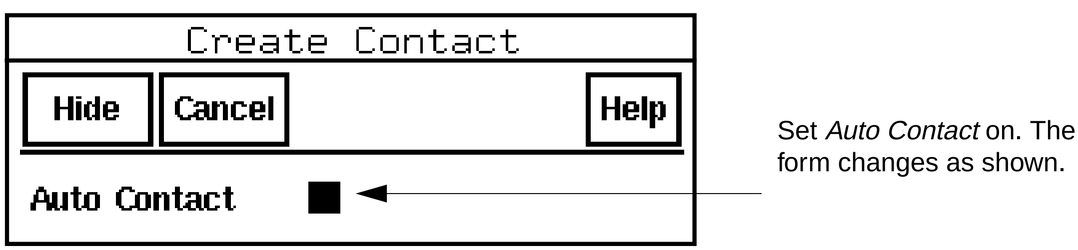


Automatically Placing a Contact on Two Paths

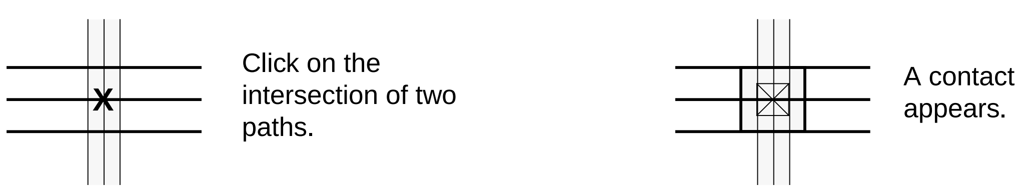
You can automatically place a contact at the intersection of two paths. The path segments must be orthogonal (parallel to the X or Y axis). When you place contacts with *Auto Contact* on, the contact origin snaps to the center of the overlap of the paths no matter what the *Snap Spacing* settings are in the Display Options form.

1. In the Create Contact form, set *Auto Contact* on.

   The form changes so that only the *Auto Contact* field appears.



2. Click on two orthogonal, intersecting path segments.



   If the contact you place is defined in your technology file for the two intersecting path layers, an appropriate contact is placed on the intersection. If the contact is not defined for the two intersecting path layers, the system gives a warning beep, which indicates the contact was not placed, and you must do either of the following:

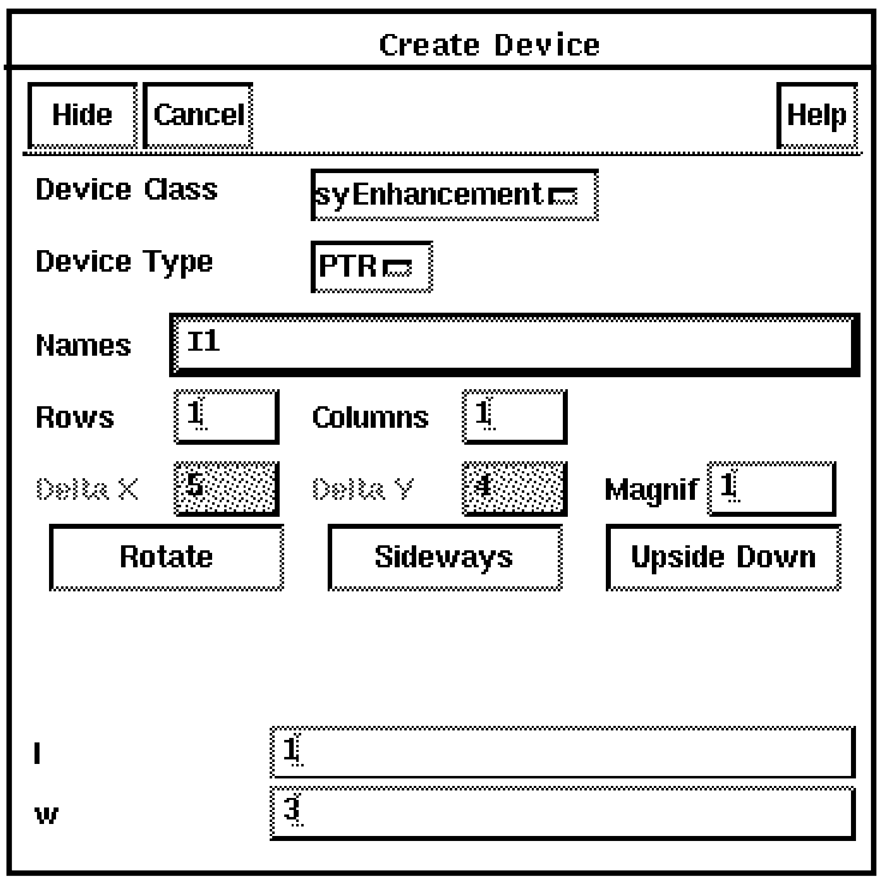
- ❑ Edit the technology file to include the definition for the contact connecting the two layers. The contacts can be either system defined or user defined.
- ❑ Change the path layers to match a contact defined in your technology file.

Note: You can also use the *Create Path* command to place contacts between path segments on different layers by path stitching.

Creating Devices

The *Create Device* command lets you place a symbolic device. A symbolic device is an instance of a parameterized cell that has been defined in the technology file.

About the Create Device Form



Device Class sets the overall device group. The classes that appear in this field are defined in the technology file.

Device Type sets the specific device type. The types that appear in this field are defined for the selected device class in the technology file. These are usually the contact or pin names.

Names sets the instance name assigned to this device. You can enter any name here or let the layout editor automatically assign instance names that begin with the letter `I`, followed by a number. You can enter multiple names (separated by a space) to place several devices of the same cell.

Rows and **Columns** set the number of rows and columns in an array of devices.

Delta X and **Delta Y** set the spacing between rows and columns in an array of devices.

Magnif(ication) enlarges or reduces the size of the device.

Rotate turns the device 90 degrees counterclockwise.

Sideways mirrors the device along the X axis.

Upside Down mirrors the device along the Y axis.

Parameters If the master cell is a parameterized cell, displays fields for one or more parameters at the bottom of the form after you select the device class. You can change the layout of the device by changing values for the parameters.

Placing a Device

To place a device inside the current cellview,

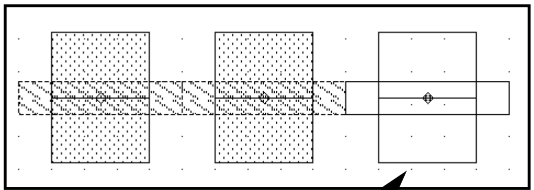
1. Choose *Create – Device*.

   The Create Device form appears.

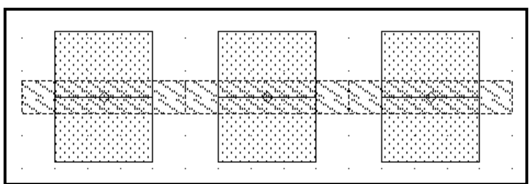
2. In the *Device Class* cyclic field, choose the device you want.
3. In the *Device Type* cyclic field, choose the type you want.
4. Move the cursor into the cellview.

   An outline of the device you want to place follows the cursor.

5. Click where you want to place the device.



Click to place the outline of the device.



The placed device

Naming Device Instances

The layout editor automatically assigns names to device instances, starting with `I1` (Instance 1) and continuing with `I2`, `I3`, and so forth.

Overriding Device Names

To override the automatic naming of devices,

1. Open the Create Device form.
2. Type one or more instance names into the *Names* field.

   Leave a space between each name.
3. In the cellview, click to place each device.

The first name in the form is assigned to the first instance you place. The next name is assigned to the next instance you place, and so forth.

Each time you place an instance, its name disappears from the *Names* list.

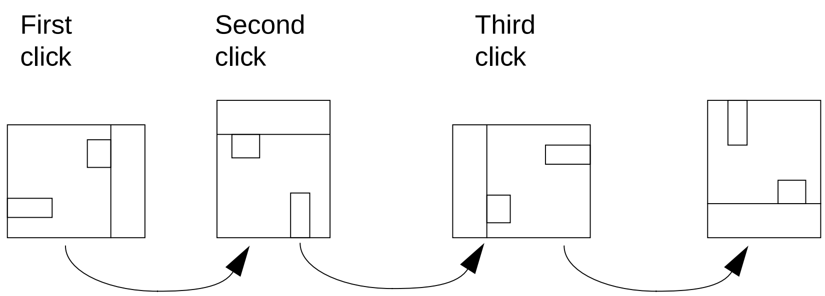
You can use the Display Options form to display instance names in the cellview.

Rotating or Mirroring Devices

To rotate or mirror devices,

➤ Click right.

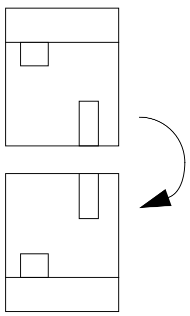
Each time you click right, the device or array rotates 90 degrees.



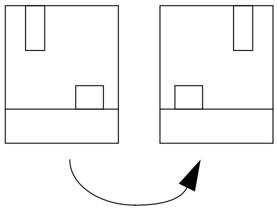
To mirror the instance,

➤ Press `Shift` and click right.

At the first click, the instance mirrors along the X axis.



At the second click, the instance mirrors along the Y axis.



To use the Create Device form to rotate or mirror instances,

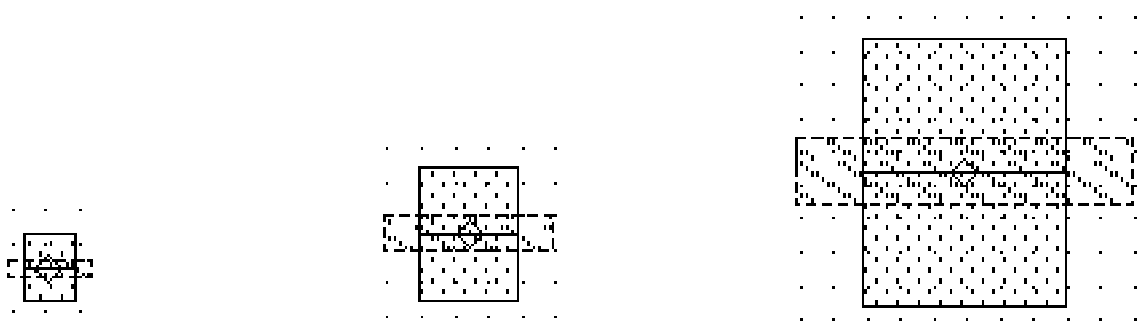
➤ Click *Rotate*, *Sideways*, or *Upside Down*.

Setting Magnification of Devices

To enlarge or reduce a device as you place it,

1. Open the Create Device form.
2. Type a value in the *Magnification* field.

The value is the factor by which the device is enlarged or reduced. For example, 2 doubles the size of the device, while 0.5 shrinks it to half its original size.



Magnification set to 0.5

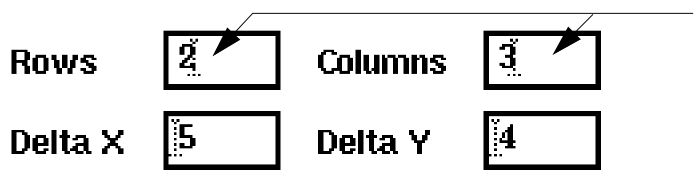
Original
(Magnification set to 1)

Magnification set to 2

Placing an Array of Devices

You can use the *Create Device* command to place many devices in an array (sometimes called a mosaic). The array is considered a single object in the database. It is assigned a single device name.

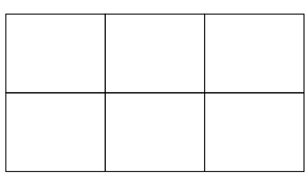
1. Open the Create Device form.
2. In the *Rows* and *Columns* fields, type the number of rows and columns you want in the array.
3. Type values for *Delta X* or *Delta Y* to change the spacing between the rows or columns.
4. By default, the fields are set to the width and length of the original device, so that the edges of each device touch but do not overlap.



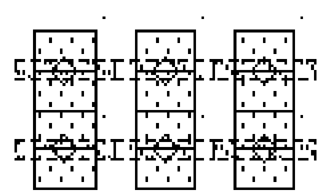
Type the number of rows and columns.

The *Delta* fields control spacing between devices.

5. Click in the cellview to place the array.



Click to place the array outline.



The placed array

Creating Conics

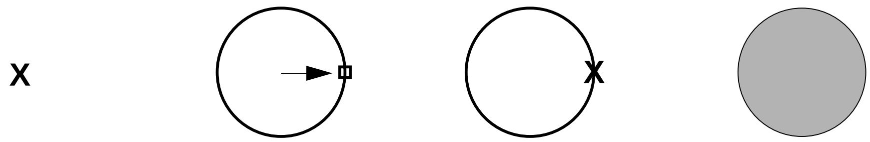
You use the *Create Conics* command to create different types of round objects.

Creating Circles

To create a circle,

1. Click on the entry layer you want in the LSW.
2. Choose *Create – Conics – Circle*.

3. Click where you want the center of the circle.
4. Click where you want the edge of the circle.

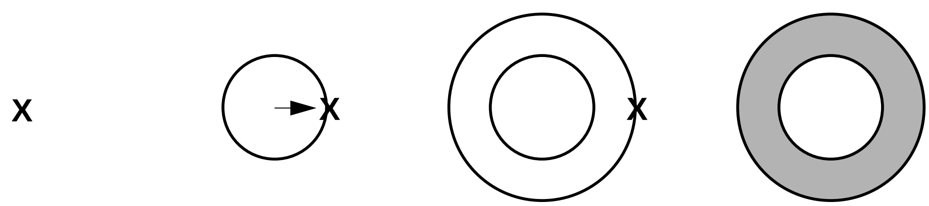


Click on the center.	Move the mouse.	Click to finish the circle.	The finished circle.

Creating Donuts

To create a donut,

1. Click on the entry layer you want in the LSW.
2. Choose *Create – Conics – Donut*.
3. Click where you want the center of the donut.
4. Click where you want the inner edge of the donut.
5. Click where you want the outer edge of the donut.



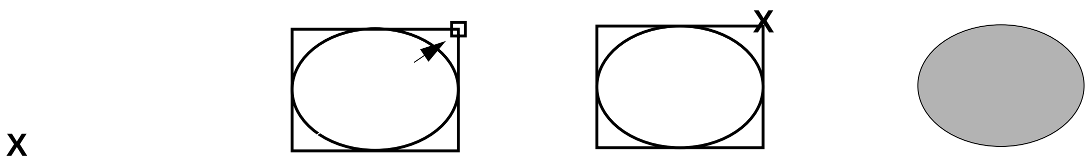
Click on the center.	Click on the inner edge.	Click on the outer edge.	The finished donut.

Creating Ellipses

You create an ellipse by creating a rectangle that surrounds the ellipse.

To create an ellipse,

1. In the LSW, click on the entry layer you want.
2. Choose *Create – Conics – Ellipse*.
3. Click where you want the first corner of the rectangle.
4. Click where you want the opposite corner of the rectangle.



Click on the first corner. | Move the mouse. | Click on the opposite corner. | The finished ellipse.

Converting Conics to Polygons

Circles, donuts, and ellipses are stored in the database as true conics with centerpoints and radii. You might need to convert conics to polygons before streaming out your database for chip production.

To convert a conic to a polygon,

1. Choose *Edit – Other – Convert To Polygon*.
2. Click on the conic you want to convert.

   The conic is redrawn as a polygon, using many straight-line segments to create each curve.
3. Continue clicking on each conic you want to convert.
4. Press `Escape` to stop the command.

Note: You can change the number of polygon segments used to create the converted conics by setting the *Conic Sides* field in the Layout Editor Options form. The default number of segments is 20.

Setting Conic Sides

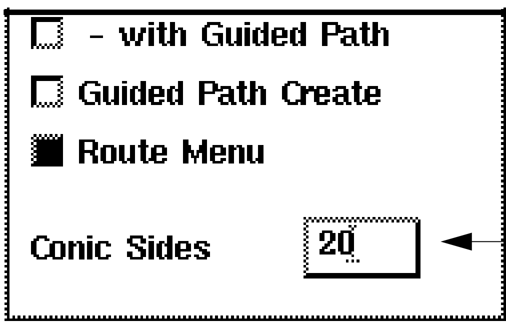
You can set the number of segments used when recreating conics when you convert them to polygons. The maximum number of conic sides is 2,047.

To set the number of conic sides,

1. Choose *Options – Layout Editor* [`Shift-e`].

   The Layout Editor Options form appears.

2. In the *Conic Sides* field, type a value.



Type the number of segments you want the converted conics to have.

3. Click *OK*.

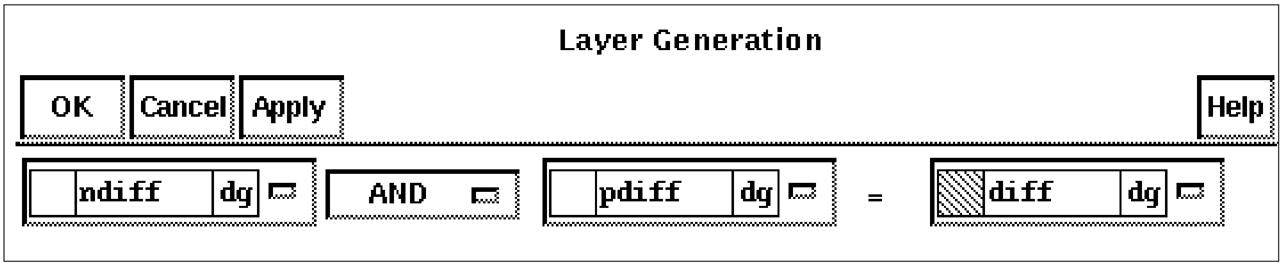
Changing Objects on a Layer

The *Layer Generation* command lets you perform Boolean edits on layers, including resizing and merging objects on a layer. *Layer Generation* edits copies of the objects; the original objects are unchanged.

About the Layer Generation Form

To open the Layer Generation form,

➤ Choose *Create – Layer Generation*.



Input Layers (the first and third fields) sets the layer or layers you want to modify with *Layer Generation*.

Boolean Operators (second field) combines the selected objects on the input layers.

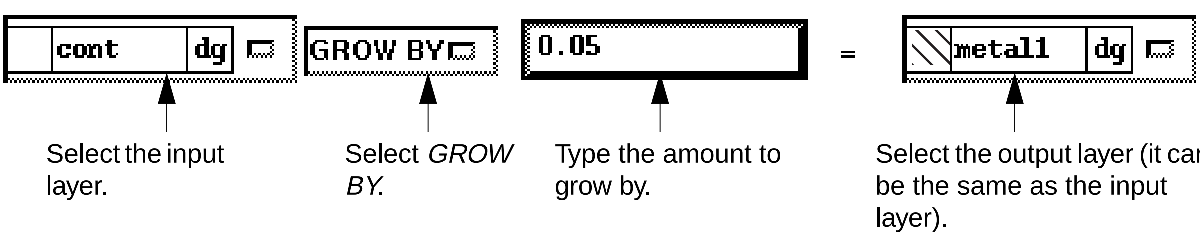
> **GROW BY** redraws the Layer Generation form so you can enter a resizing value. Enter the number of user units (resizing value) by which to enlarge selected objects on the input layer in the field to the right.

Output Layer (fourth field) sets the layer on which the new shapes will appear.

Making Changes to Objects on Layers

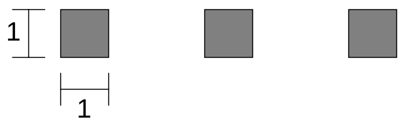
To enlarge copies of selected objects,

1. Select the objects you want to resize.
2. Choose *Create – Layer Generation*.
3. Fill out the Layer Generation form as shown below.

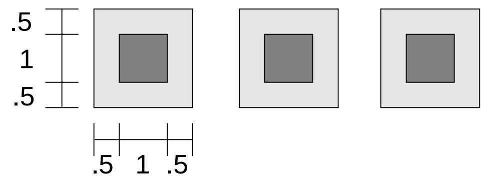


4. Click *OK*.

Copies of the selected objects are enlarged.



Original objects on contact layer

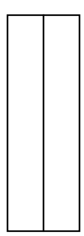


After *GROW BY*, enlarged copies of originals appear on the *metal1* layer.

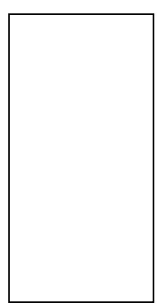
How Sizing Converts Objects to Polygons

When you resize any object using *Size* or *Layer Generation*, the object is converted to a polygon.

For example, if you enlarge a path with *Size*, the resulting object is a polygon.



A path before resizing



After resizing, the path has been converted to a polygon.

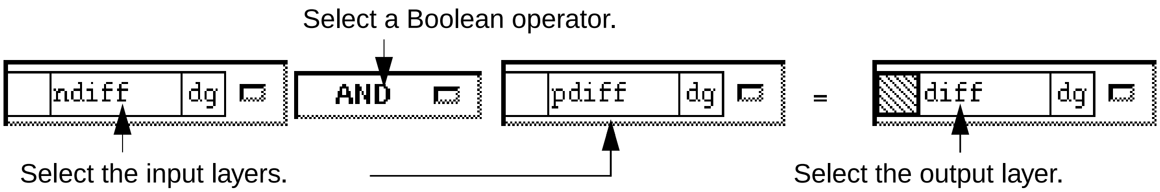
Making Boolean Edits by Layer

You can perform Boolean logical operations on objects on one or two layers. You edit copies of the objects, so the originals are unchanged.

1. Select the objects you want to edit.
2. Choose *Create – Layer Generation*.

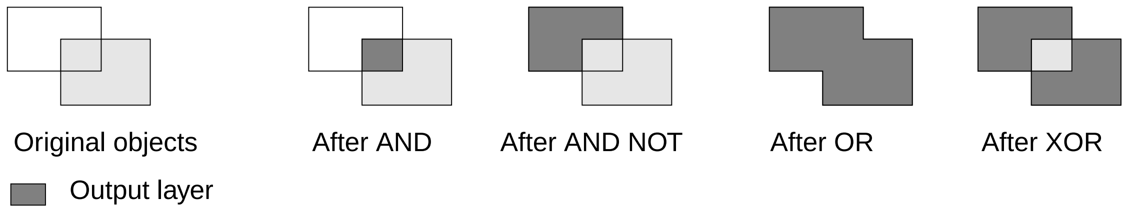
   The Layer Generation form appears.

3. Choose the layers you want to edit and the Boolean operator you want to use.



4. Click *OK*.

The following figure shows the results of different Boolean operations.



9

Editing Objects

This chapter contains these topics about the Virtuoso® layout editor:

How Editing Commands Work with ROD Objects

The following tables summarize the level of support for how editing commands work on relative object design (ROD) objects in the current release.

Using commands that are not fully supported for ROD objects could cause the objects to lose the ROD information associated with them, changing the objects into ordinary shapes.

Edit Command	**Degree of ROD Support**
Undo u	The *Undo* command fully supports ROD objects.
Redo U	The *Redo* command fully supports ROD objects.
Move m	Moving ROD objects is supported as follows: ■ You can move a ROD object within the same cellview or to another cellview. ■ Within the same cellview, moving a ROD object that has other objects aligned to it causes the aligned objects to move as well. ■ When you move a ROD object between cellviews, and the ROD object is aligned to another ROD object(s), the system preserves alignment only when the aligned ROD object(s) is also in the selected set; otherwise the alignment is broken. ■ Avoid rotating aligned ROD objects during a move because the aligned handle names are not updated after the move, so the results might not be what you want.

Copy c	Copying ROD objects is supported as follows: ■ You can copy a ROD object within the same cellview or to another cellview. The system automatically assigns unique names to the copies. ■ Alignments between ROD objects in the selected set result in alignments between the corresponding copy objects. ■ Alignments to objects not in the selected set are ignored. ■ Avoid rotating aligned ROD objects during a copy because aligned handle names are not updated after the copy, so the results might not be what you want.
Stretch s	The *Stretch* command fully supports ROD objects including stretchable parameterized cell (pcells).
Reshape R	The *Reshape* command fully supports ROD objects.
Delete del	The *Delete* command fully supports ROD objects.
Properties ...q	You can use the *Edit Properties* command for ROD objects to ■ View system-defined and user-defined handle names and handle values ■ View alignments for the selected ROD object ■ Modify the X and Y separation between the selected ROD object and other ROD objects
Search ...S	The *Search* command is not supported for ROD objects.
Merge...M	The *Merge* command is not supported for ROD objects. When ROD objects are merged, the resulting shape is not a ROD object.
Select – Select All ^a	The *Select All* command fully supports ROD objects.
Select – Deselect All ^d	The *Deselect All* command fully supports ROD objects.
Hierarchy – Make Cell...	The *Make Cell* command fully supports ROD objects.
Hierarchy – Flatten	The *Flatten* command fully supports ROD objects. The system assigns the flattened object a name based on the hierarchical name of the ROD object by replacing slashes with dashes. For example, when you flatten the ROD object `I1/I4/rect3`, the resulting object is named `I1-I4-rect3`.

Other – Chop C	The *Chop* command fully supports ROD multipart paths, but not ROD rectangles or ROD polygons. When you chop a ROD rectangle or ROD polygon, the resulting shape is not a ROD object.
Other – Modify Corner...	The *Modify Corner* command fully supports ROD rectangles.
Other – Size...	The *Size* command fully supports ROD objects.
Other – Split ^s	The *Split* command fully supports ROD objects.
Other – Attach/Detach v	The *Attach/Detach* commands fully support ROD objects.
Other – Convert to Polygon	The *Convert To Polygon* command fully supports ROD objects. When you convert a multipart path, the polygon resulting from the master path is a ROD object and takes the name of the multipart path; subparts, if any, become unnamed, non-ROD shapes.
Other – Move Origin	The *Move Origin* command fully supports ROD objects.
Other – Rotate... O	The *Rotate* command supports the rotation of ROD objects, unless there are alignments. Avoid rotating aligned ROD objects because aligned handle names are not updated after the rotation, so the results might not be what you want.
Other – Yank y	The *Yank* command does not support ROD objects. The *Yank* command copies only the shapes, but not the names of the shapes or the alignments. When you paste, the result is unnamed, unaligned, non-ROD shapes.
Other – Paste Y	The *Paste* command does not support ROD objects. If you yank and paste ROD objects, the result is unnamed, unaligned, non-ROD shapes.

Using the Move Command

The *Move* command lets you move an object to another location in this or another cellview.

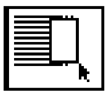
When you move ROD objects and multipart paths into a new cellview, all of the data remains intact. If the multipart path is chopped, that data also moves with it. The ROD objects and multipart paths are renamed if there are objects with similar names in the new cellview. To retain alignments, each of the aligned objects must be part of the selected set to be moved. Avoid rotating aligned ROD objects during a move.

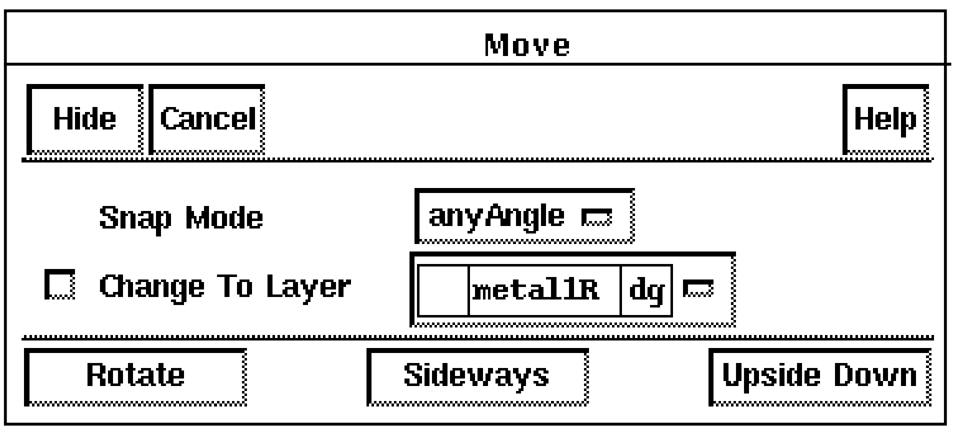
For more information about moving ROD objects, see “How Editing Commands Work with ROD Objects” on page 265

About the Move Form

To open the Move form,

➤ Do one of the following:

- Choose *Edit – Move*.
- Press `m`.
-  Click on the move icon in the icon menu.



Snap Mode controls the direction in which you can move the object.

Change To Layer lets you move an object to another layer.

Rotate turns the object 90 degrees counterclockwise.

Sideways mirrors the object along the X axis.

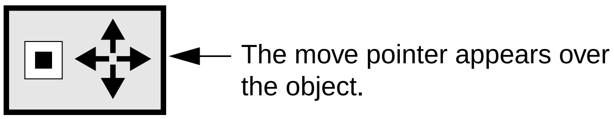
Upside Down mirrors the object along the Y axis.

Moving Objects by Direct Manipulation

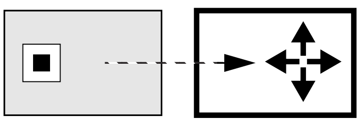
You can move objects by selecting and dragging them.

To move an object,

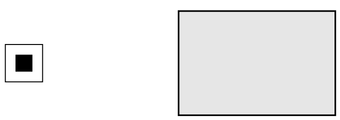
1. Select an object.
2. Move the pointer over the object to see the move pointer.



3. Press the left mouse button at the reference point for the move (the point from which the move starts) and drag the pointer to the destination point.



4. Release the left mouse button.



The object is moved.

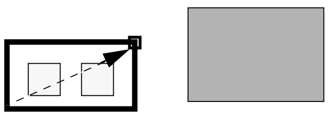
Moving Objects with the Move Command

To move objects using the *Move* command,

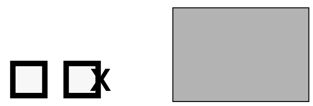
1. Choose *Edit – Move* [`m`].
2. Select one or more objects.

   Click on the reference point for the move (the point from which the move starts), then click the pointer on the destination point.

If you create a selection box or press `Shift` while selecting all objects, the *Move* command prompts you for a reference point.

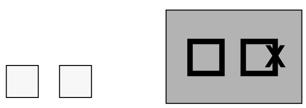


Press left to create a selection box.

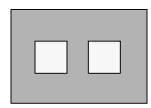


Click to enter a reference point.

3. Click where you want to move the objects.



Click to move the objects.



The objects move.

Moving or Copying Objects to Another Layer

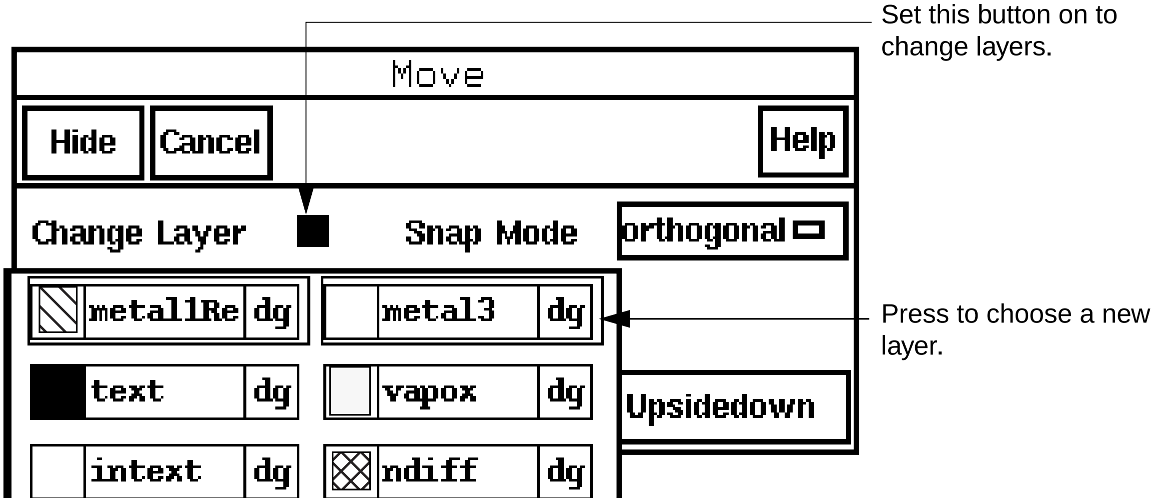
To move or copy objects to another layer,

1. Choose *Edit – Move* [`m`] or *Edit – Copy* [`c`].

   The Move form or Copy form appears.

2. Set *Change Layer* on.

3. Click on the layer field and choose the layer you want.



4. Click on the object whose layer you want to change.
5. Click where you want to move or copy the object.

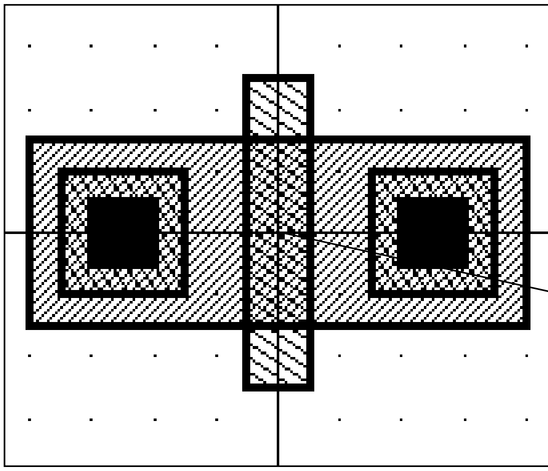
   The object is moved or copied and is now on the new layer.

Moving Objects to Another Cellview

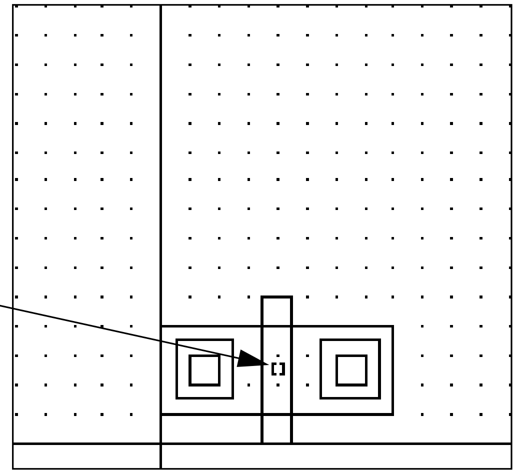
To move objects to another open cellview,

1. Select the objects you want to move.
2. Choose *Edit – Move* [`m`].

3. Place the pointer in another open cellview.



Select the objects in the first cellview.



Move the objects into the other cellview.

4. Click to place the objects.

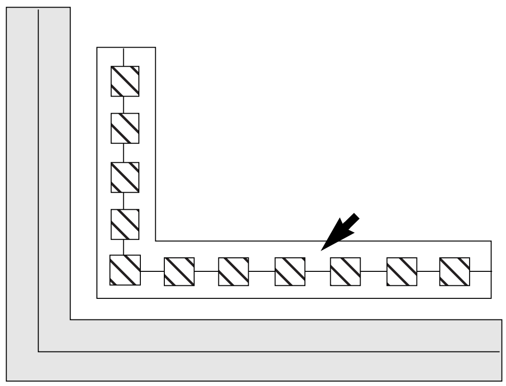
Moving Multipart Paths

When you move a multipart path within a cellview, its master path and subparts move together. You cannot move the master path separately from its subparts, nor can you move subparts separately from the master path.

When you move a multipart path that has objects aligned to it, the objects move to stay in alignment with the multipart path.

1. Choose *Edit – Move* [`m`].

2. In full selection mode, choose the multipart path by clicking on the master path or any subpart.



Click on any part of the multipart path.

3. Click to place the multipart path; the aligned objects follow.

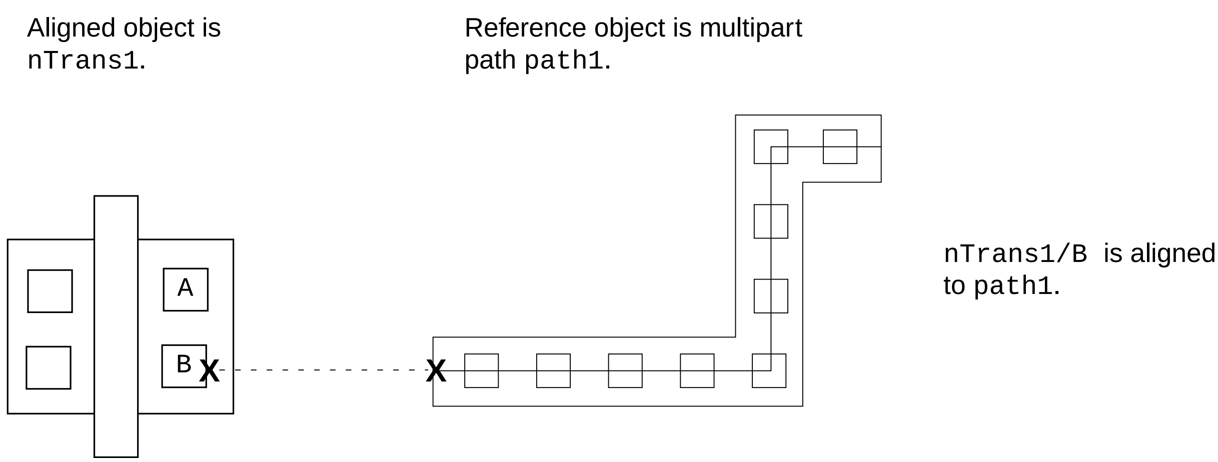
The multipart path and all objects aligned to it appear in the new location.

How Moving Affects Multipart Paths with Aligned Objects

When you move a multipart path, any objects aligned to the multipart path move with it. To retain alignments when you move a multipart path to a new cellview, all objects aligned with the multipart path must be part of the selected set. If aligned objects are not selected, they will not be moved and the alignments will be broken.

In the example below, the multipart path `path1` is the reference object. Contact `B` in instance `nTrans1` is aligned to `path1`, with a separation of -10 units in the direction of the `X` axis.

Therefore, `nTrans1` behaves as if it is aligned to `path1`. When you move `path1`, `nTrans1` moves with it.



The system automatically recalculates the alignment of objects when you open a layout cellview in edit mode or edit an aligned object in any way.

For an overview of relative object design alignment, see "Aligning Named Objects" in the *Virtuoso Relative Object Design User Guide*.

For multipart paths, see "Multipart Paths" and the `rodCreatePath` function in the *Virtuoso Relative Object Design User Guide*.

Using the Copy Command

The *Copy* command lets you place a copy of an object in a cellview.

You can copy ROD objects within the same cellview or to another cellview. The system automatically assigns unique names to the copies. Avoid rotating aligned objects during a copy. For further information about copying ROD objects see, "How Editing Commands Work with ROD Objects" on page 265.

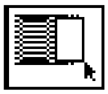
To copy parts of objects or groups of objects see "Yanking and Pasting" on page 348.

About the Copy Form

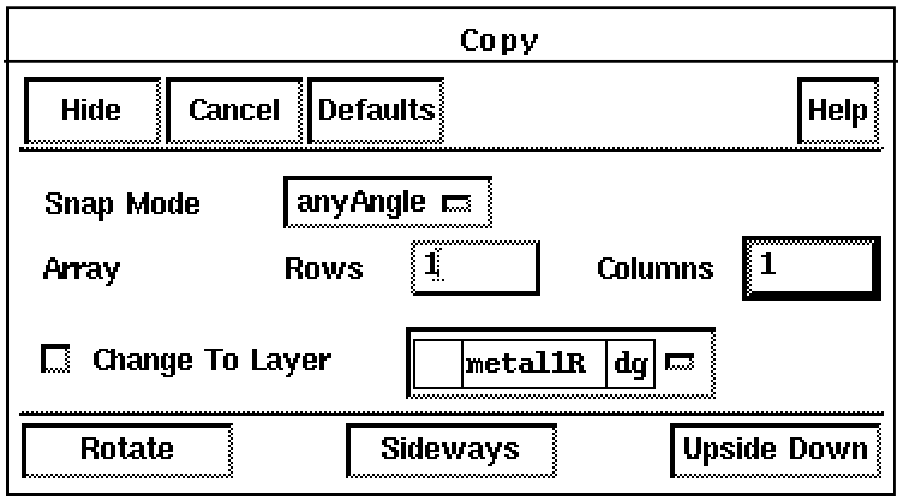
To open the Copy form,

➤ Do one of the following:

- Choose *Edit – Copy*.
- Press c.



- Click on the copy icon in the icon menu.



Snap Mode controls the direction in which you can move the copied object.

Array

> **Rows** and **Columns** let you create an array of copied objects. You enter the number of rows and columns in the array. The array of copied objects can be placed only in the original cellview; it cannot be copied to another cellview.

Change To Layer lets you copy an object to another layer.

Rotate turns the object 90 degrees counterclockwise each time it is pressed.

Sideways mirrors the object along the X axis.

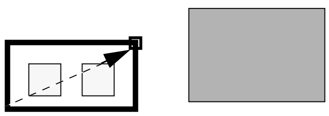
Upside Down mirrors the object along the Y axis.

Copying Objects

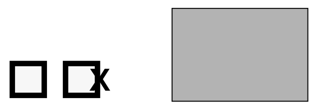
To copy objects,

1. Choose *Edit – Copy* [c].
2. Select one or more objects.

- ❑ Click to select the first object, which point serves as a reference point for moving the copies.
- ❑ Create a selection box, or press `Shift` while selecting all objects, and the *Copy* command prompts you for a reference point.

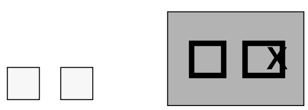


Click and drag to create a selection box.

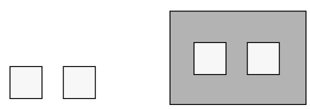


Click to enter a reference point.

3. Click where you want to move the copies.



Click to move the copies.



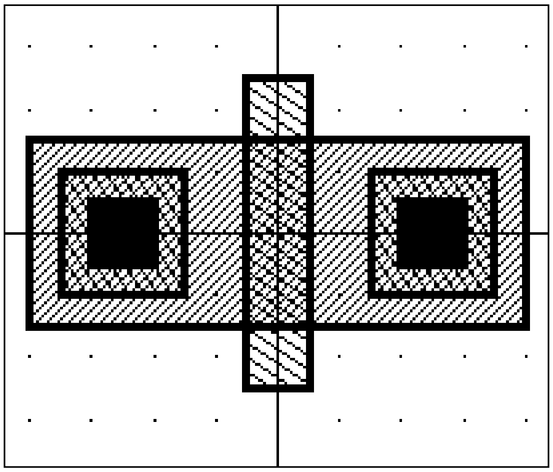
The copies appear.

Copying Objects to Another Cellview

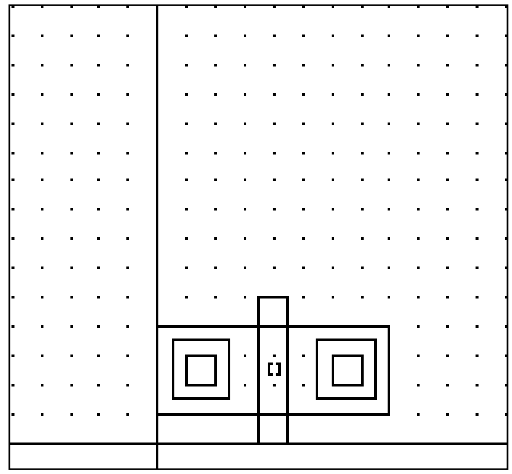
To copy objects to another open cellview,

1. Select the objects you want to copy.
2. Choose *Edit – Copy* [c].

3. Move the pointer into another open cellview.



Select the objects in the original cellview.



Move the copies into the other cellview.

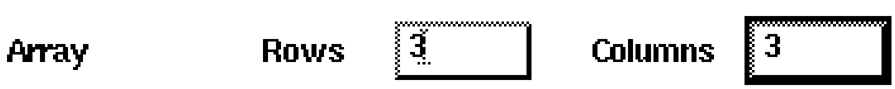
4. Click to place the copies.

Note: You cannot use the *Array* fields in the Copy form to create and copy an array of copied objects from one cellview to another. Only the original objects you selected are copied to the next cellview even if you have typed numbers in the *Rows* and *Columns* fields.

Creating an Array of Copies

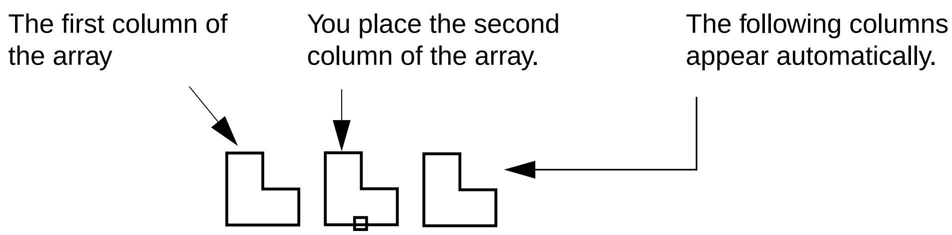
You can create an array of copied objects. This array is flat, meaning all objects in it can be selected individually. The array of copied objects can be placed only in the original cellview; it cannot be copied to another cellview.

1. Open the Copy form.
2. In the *Rows* and *Columns* fields, type the number of rows and columns you want in the array.



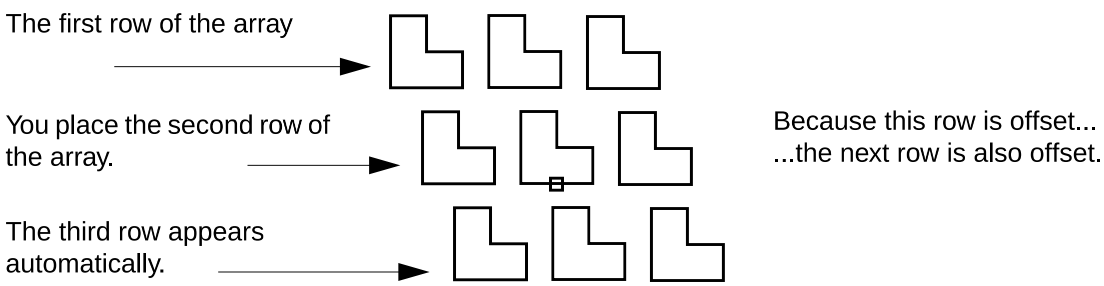
3. Click to place the first object of the array.
4. Click to place the columns of the array.

The distance between the first object and the second one controls the spacing and orientation between all the columns.



5. Click to place the rows of the array and complete the array.

   The distance between the first and second rows controls the spacing and orientation between all the rows.



Note: You cannot use the *Copy* command to copy arrays from one cellview to another.

Stretching Objects

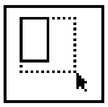
The *Stretch* command lets you stretch an edge or corner of an object.

About the Stretch Form

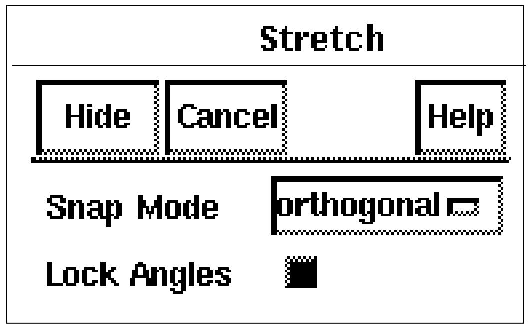
To open the Stretch form,

➤ Do one of the following:

- ❑ Choose *Edit – Stretch*.
- ❑ Press s.



❑ Click on the stretch icon in the icon menu.



Snap Mode controls the direction in which you can stretch an edge.

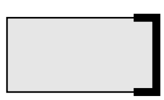
Lock Angles prevents you from changing the angle of a corner or edge as you stretch it.

Stretching Objects by Direct Manipulation

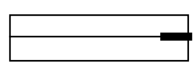
You can stretch objects by selecting and dragging their edges or corners.

To stretch an object,

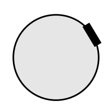
1. Select one or more edges or corners as follows:



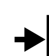
Press F4 and click on an edge or corner of a polygon.

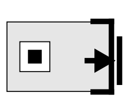


Press F4 and click on an endpoint on a path centerline.

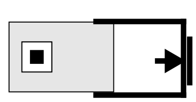


Press F4 and click on a point on a circle or donut.

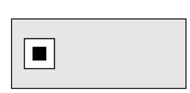
2. Move the pointer over the edge until you see the stretch pointer: 
3. Click and drag the pointer to stretch the object.
4. Release the mouse button.



Press on a reference point.



Drag the pointer.



Release the mouse button. The object is stretched.

Note: You cannot use *Stretch* to delete objects. If you try to stretch an object to a zero area, a warning appears in the Command Interpreter Window (CIW) and a dialog box opens stating the shape was not modified because it would be illegal.

Stretching with the Stretch Command

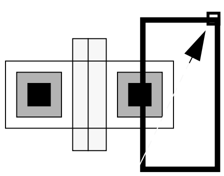
You can use the *Stretch* command to stretch objects.

To stretch an object,

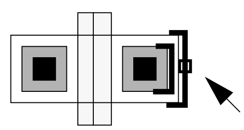
1. Choose *Edit – Stretch* [s].
2. Select the edges or corners you want to stretch.

   In partial selection mode, if you press `Shift` and click on an edge or corner, that edge or corner is the reference point for the stretch.

   If you create a selection box around the edges, *Stretch* prompts you for a reference point. The reference point does not have to be on the selected shapes.

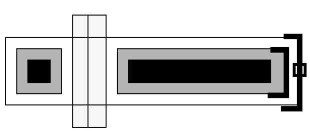


Select a group of edges.

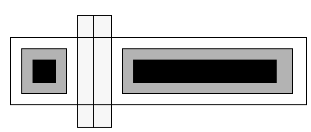


Click on a reference point.

3. Move the cursor and click to stretch the edges.



Move the cursor to stretch the edges.



The stretched objects

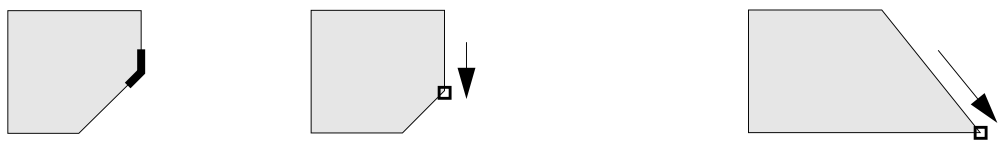
Changing the Angle of Corners

To change the angle of a corner by stretching,

1. Choose *Edit – Stretch* [s].

   The Stretch form appears.

2. Turn *Lock Angles* off.
3. Stretch the corner where you want and click.



The selected corner

With *Lock Angles* on, you cannot change the angle of the corner.

With *Lock Angles* off, you can stretch the corner in any direction.

Note: If you create a nonorthogonal segment while stretching the corner of a multipart path, subrectangles cannot regenerate in that particular segment. (Subpaths can and will regenerate.)

Stretching and Moving Objects

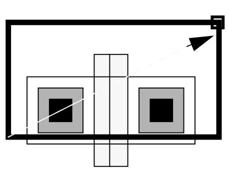
You can simultaneously stretch and move objects. This is often called a window stretch.

To stretch and move objects,

1. Choose *Edit – Stretch* [s].

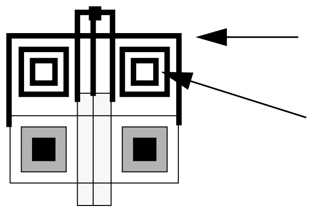
   The Stretch form appears.

2. Press `Shift` and create a selection box around the objects you want to move and the edges you want to stretch.



3. Select a reference point for the stretch.

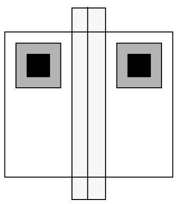
4. Move the cursor to stretch the edges where you want.



This selected edge stretches.

This selected object moves.

5. Click.



The stretched and moved objects

Stretching Paths

You can stretch the ends, segments, and/or corners of a path. You stretch ends and corners in the same way you stretch any other object. To stretch segments, you need to select the path centerline in the segment(s) you want to stretch.

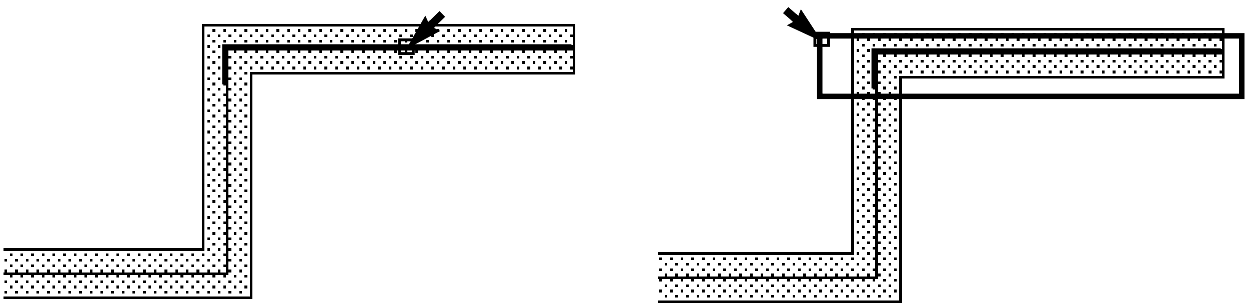
To stretch path segments,

1. Choose *Edit – Stretch* [s].

Stretch automatically changes to partial selection mode and prompts you to select the object you want to stretch.

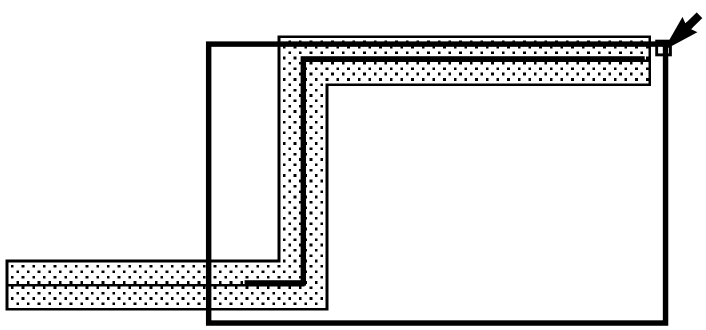
2. Select one or more segments by doing one of the following:

- For a single segment, click on the middle of the centerline of the segment or create a selection box around the centerline of the segment, making sure to include the ends of the segment centerline.



Highlighting shows the path segment is selected.

- For multiple segments, to add another segment, press `Shift` and click or create a selection box around the centerline of the segments, making sure to include the ends of the segment centerlines.

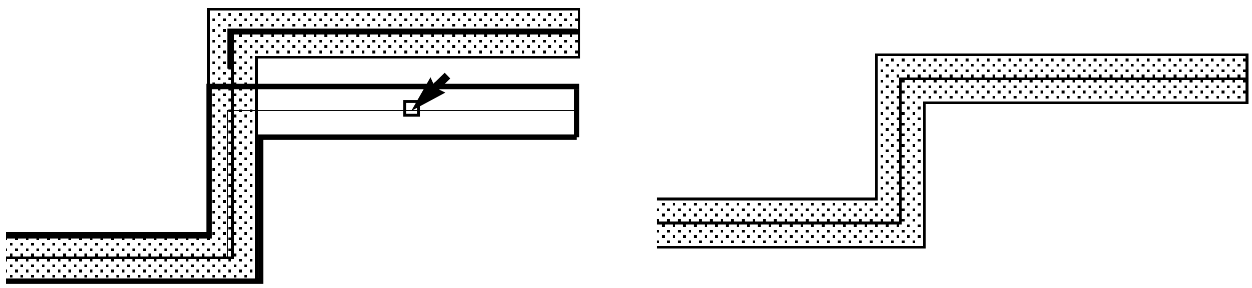


Highlighting shows the path segments are selected.

When you use a selection box, *Stretch* prompts you for a reference point. The reference point does not have to be on the selected shapes.

3. If you are prompted for a reference point, click where you want the stretch to begin.

4. Point to the new location for the stretch and click.



5. To end the *Stretch* command, press `Escape`.

Stretching Multipart Paths

You can stretch the ends, segments, and/or corners of a multipart path in the same way you stretch single-part paths; the master path and its subparts stretch together. You cannot stretch the master path separately from its subparts, nor can you stretch subparts (except chopped subpaths) separately from the master path.

If you attempt to stretch a subpart, the system applies the stretch to the whole multipart path. The only exception is that you can stretch the chopped ends of subpaths. When you stretch a chopped subpath, all other chopped subparts are stretched also.

Note: You cannot directly select a set of subrectangles, but selecting and stretching the chopped end of a subpath causes all chopped parts in the multipart path to be regenerated, including chopped sets of subrectangles. The system regenerates subrectangles along orthogonal segments only.

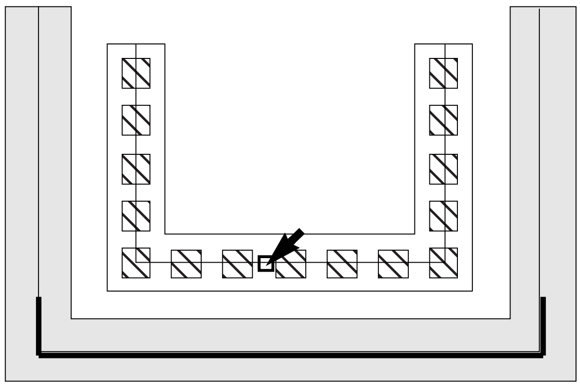
To stretch one segment in a multipart path,

1. Choose *Edit – Stretch* [s].

   Stretch automatically changes to partial selection mode and prompts you to select the object you want to stretch.

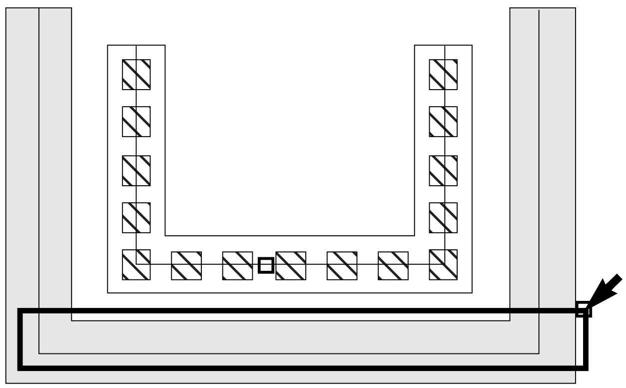
2. Select the segment you want to stretch by doing one of the following:

- Click on the middle of the centerline of the master path segment or on the comparable subpath segment.



The nearest master path segment is selected.

- Create a selection box around the centerline of a segment.

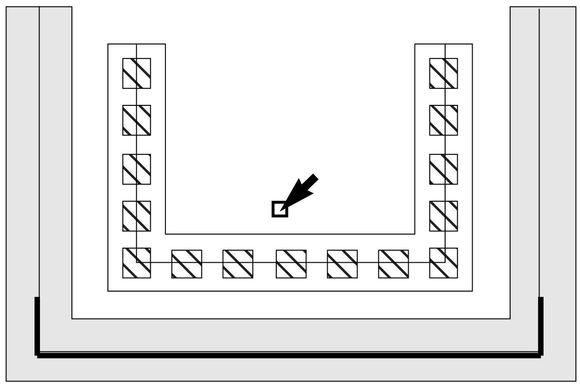


Be sure to include the ends of the segment centerline.

When you use a selection box, *Stretch* prompts you for a reference point.

3. If you are prompted for a reference point, click where you want the stretch to begin.

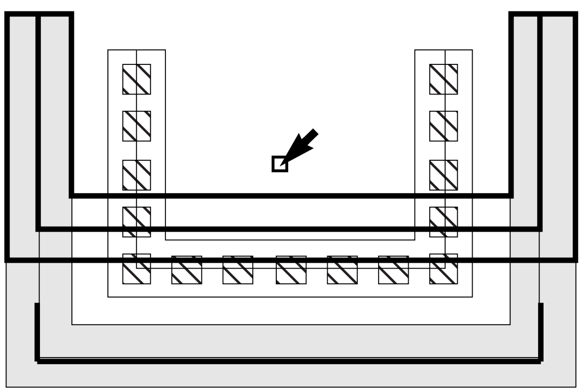
The reference point does not have to be directly on the multipart path.



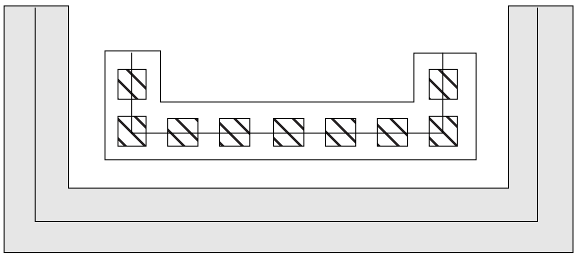
Stretch prompts you to point to the new location.

4. Point to a new location for the stretch.

As you point, an outline of the master path segment shows you the new location.



5. Click to select the new location for the selected multipart path segment.



Stretching Chopped Subpath Ends in Multipart Paths

You can select and stretch the ends of chopped subpaths in a multipart path. You might want to do this to reconnect the ends. When you stretch the chopped end(s) of one or more subpaths, all chopped subparts stretch with the selected subpath(s).

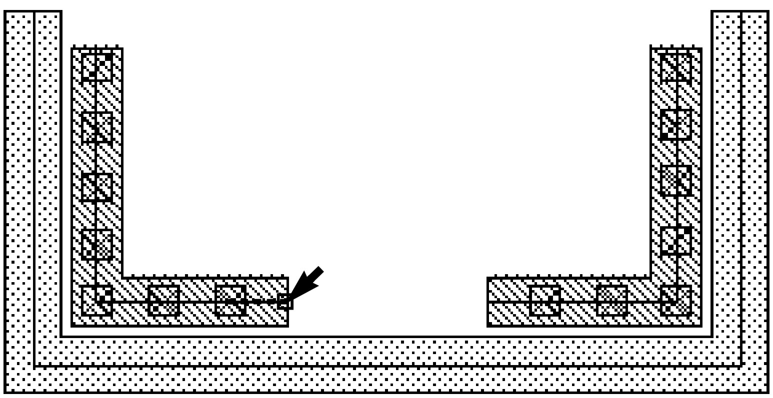
Note: You cannot directly select a set of subrectangles, but selecting and stretching the chopped end of a subpath causes all chopped parts in the multipart path to be regenerated, including chopped sets of subrectangles. The system regenerates subrectangles along orthogonal segments only.

To stretch the chopped end of a subpath,

1. Choose *Edit – Stretch* [s].

   Stretch automatically changes to partial selection mode and prompts you to select the object you want to stretch.

2. Select a chopped subpath end with a selection box or by clicking on it.



When you use a selection box, *Stretch* prompts you for a reference point.

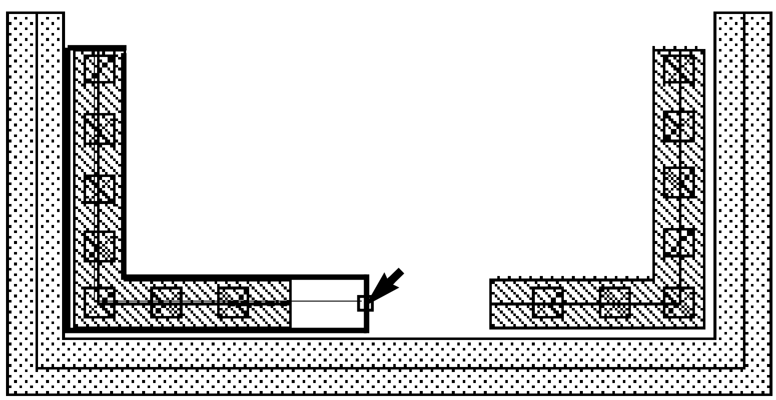
3. If you are prompted for a reference point, click where you want the stretch to begin.

The reference point does not have to be directly on the path end.

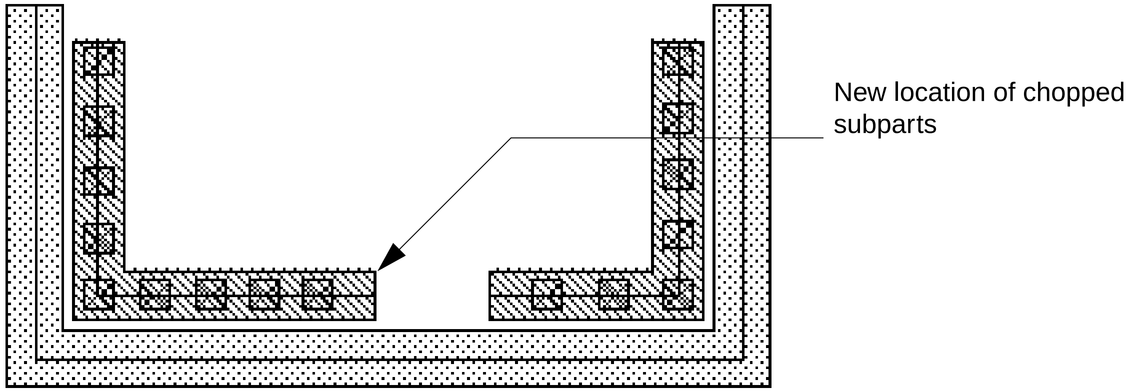
Stretch prompts you to point to the new location.

4. Point to a new location for the stretch.

As you point, an outline shows you the new location of the subpath end.



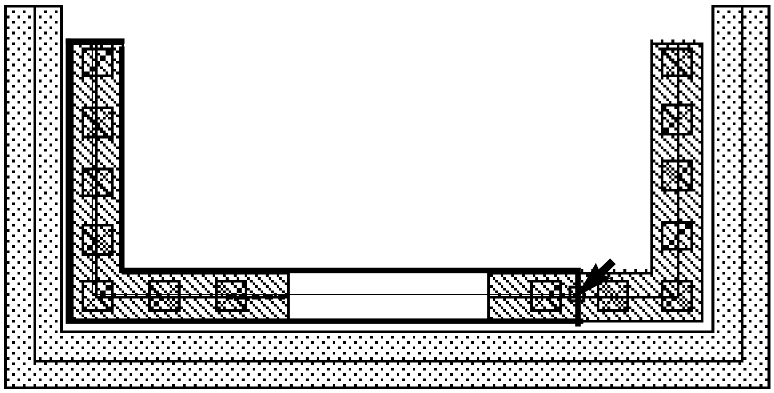
5. Click to select the new location for the chopped subpath end.



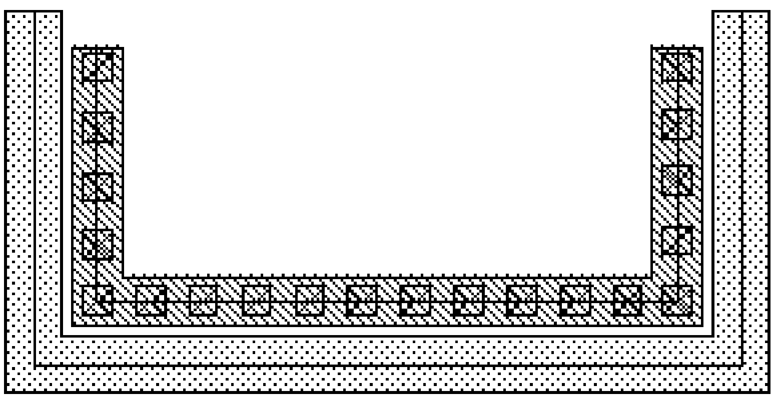
You can reconnect chopped subparts by stretching one subpath end over to the other subpath end. Even if you overlap the ends, the system regenerates the subrectangles correctly.

6. Select the same chopped subpath end, and point to select the new location so that it overlaps the other chopped subpath end.

As you point, an outline shows you the new location of the subpath end.



The system reconnects the subpaths and regenerates subrectangles to fill the reconnected segment.



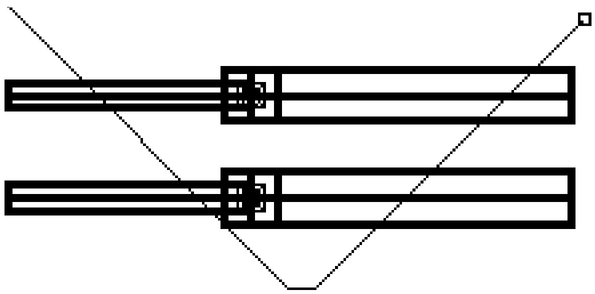
Adding a Jog to Several Paths (a Bus)

You can add a jog to several paths (a bus) so that it wraps around an object by splitting a section of the paths and stretching the split section.

To stretch several paths,

1. Select all paths and connected objects you want to stretch.
2. Choose *Edit – Other – Split* [`Control-s`].
3. Click to create a line through the segments.

To stretch a segment of each path, create a split line that crosses each path twice, once at each end of the segment you want to stretch.

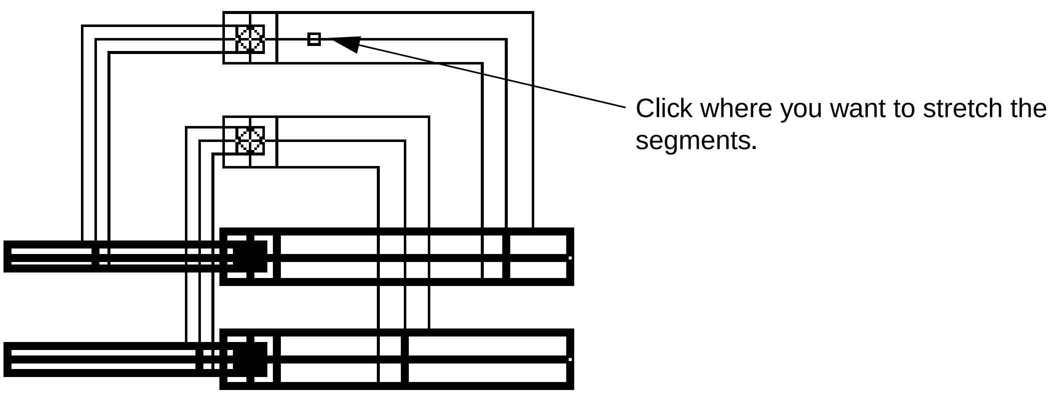


Create the split line at a 45-degree angle through the segments, so that the edges of the segments are offset from each other. This prevents the path segments from overlapping when you stretch them.

4. Double-click on the last point of the split line.

5. Click on a starting point for the stretch.

6. Click where you want to stretch the segments.



The path segments are stretched. Also, these other objects move:

- Any objects attached to the segments (such as contacts) that have been selected
- Any objects not recognized by the *Split* command (instances, labels, mosaics, bends, and tapers) but attached to the path, if they have been selected
- Objects aligned to a multipart path you stretch stay in alignment, even if you did not select them

Subrectangles and subpaths in multipart paths regenerate after the stretch.

Adding a Jog to Multipart Paths

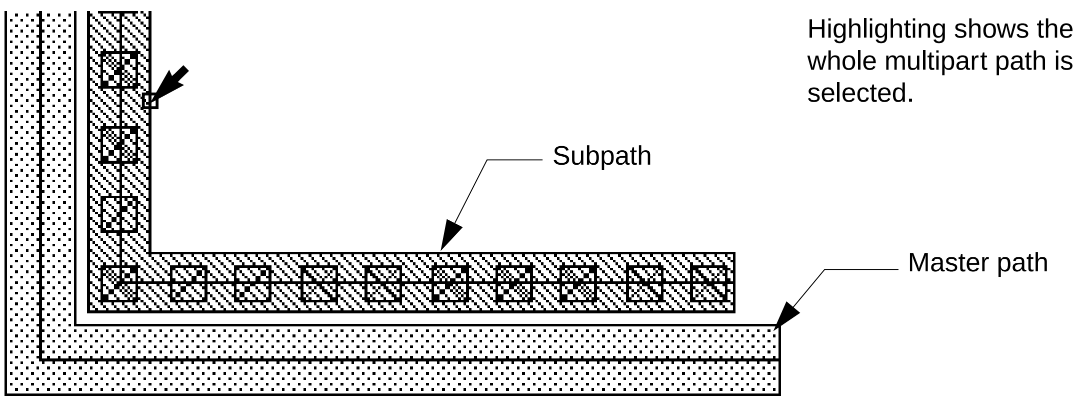
You can add a jog to a section of a multipart path in the same way you add a jog to single-part paths; the jog affects the master path and all of its subparts. You might want to split and stretch a multipart path to direct it around an object or to add new connecting segments.

Any objects aligned with or attached to the part of the multipart path that moves, move with the stretch. You can split paths only in full selection mode.

Note: You cannot split or stretch the master path separately from its subparts, nor can you split or stretch subparts separately from the master path. If you attempt to split a subpart, the system applies the split to the whole multipart path.

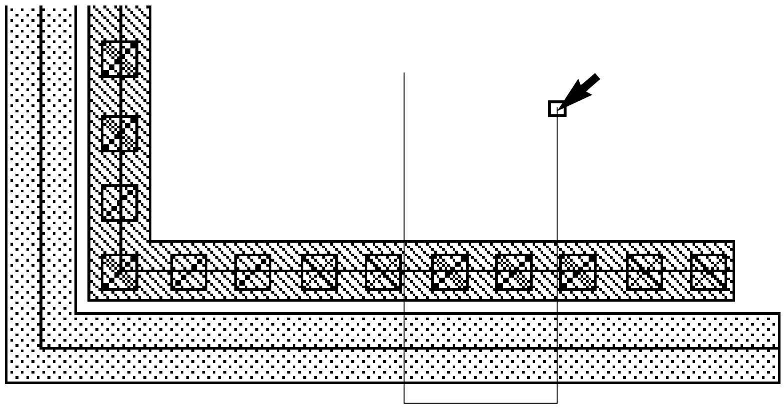
To stretch a section of a multipart path segment,

1. Choose *Edit – Other – Split* [`Control-s`].
2. Select the object to be split by clicking anywhere on the multipart path.



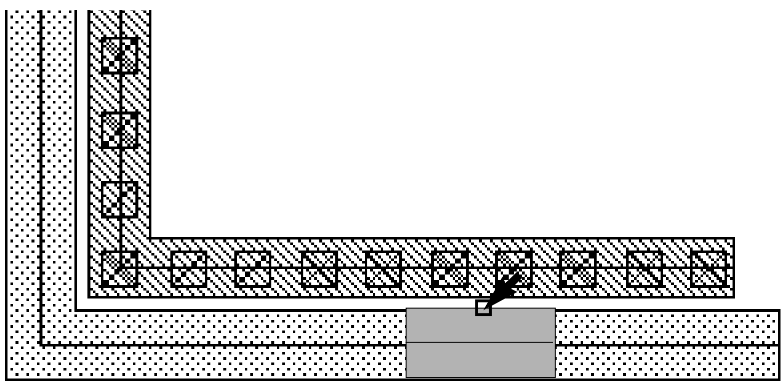
Split prompts you to draw a split line by clicking on points. To define the section you want to stretch, your split line must cross the centerline of the master path in two places.

3. To create a split line, click on four points as shown below, then press `Return` to end the split line.



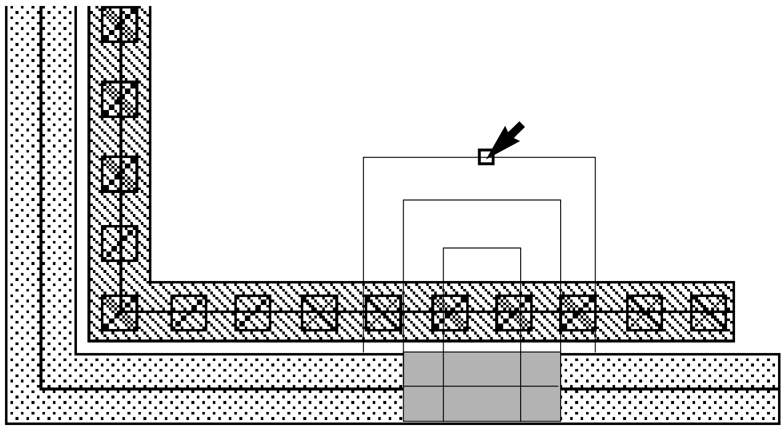
Split prompts you for a reference point.

4. Click on a starting point for the stretch.



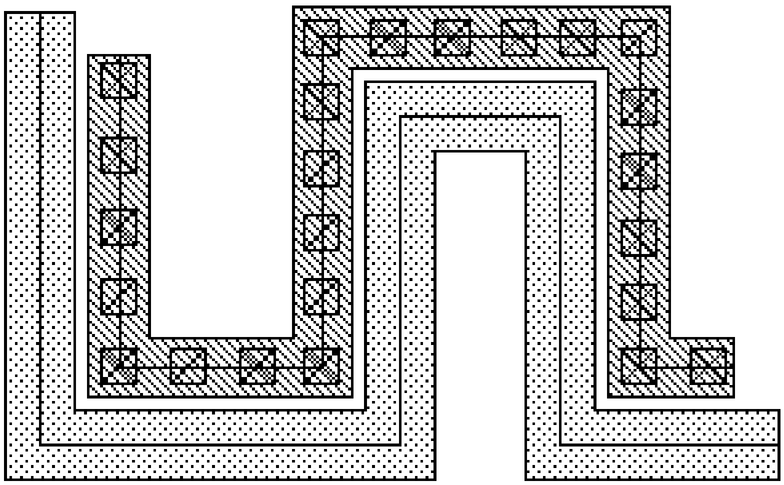
Split prompts you for the new location for the stretch.

5. Point to a new location for the stretch.



An outline of the segment shows you the position of the stretch.

6. Click to select the new location for the section being stretched.



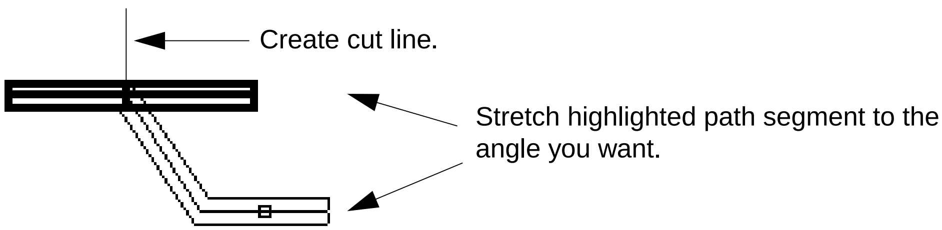
Using Split to Add Any-Angle Segments to Paths

As you stretch path segments, you can change the angle of the segments.

To stretch path segments to any angle,

1. Open the Split form.

2. Turn *Lock Angles* off.
3. Create a cut line.
4. Stretch the highlighted path segment to the angle you want.



For multipart paths, see "Multipart Paths" and the `rodCreatePath` function in the *Virtuoso Relative Object Design User Guide.*

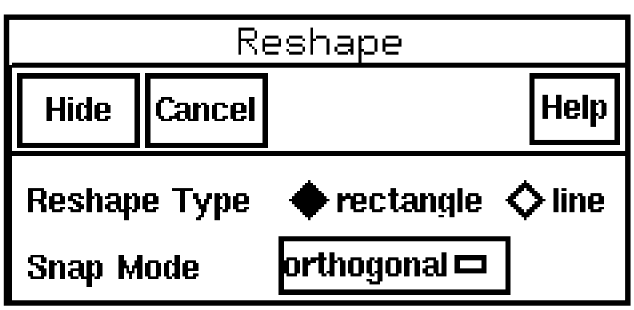
Reshaping Objects

The *Reshape* command lets you change the shape of a selected object.

About the Reshape Form

To open the Reshape form,

➤ Do one of the following:
- ❑ Choose *Edit – Reshape*.
- ❑ Press `Shift-r`.



Reshape Type sets the geometry to use for reshaping a selected object.

rectangle lets you add or remove a rectangle shape.

line lets you add a polygon to a shape or reshape a path. Only *line* can be used to reshape a path.

Snap Mode controls the shape of line segments; applies only when *Reshape Type* is set to *line*.

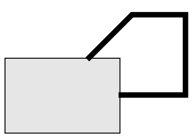
Reshaping Polygons

To reshape a polygon,

1. Choose *Edit – Reshape* [`Shift-r`].

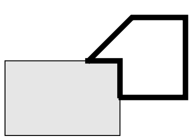
   The Reshape form appears.

2. Select a polygon.
3. Turn on *Reshape Type: line*.
4. Create the new section of the polygon.

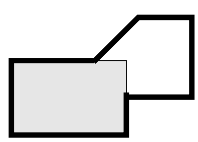


The first and last points must touch the original shape.

5. Double-click when you are finished entering points.
6. Click right to toggle between highlighting the new shape and highlighting both the old and new shape.



Highlight the new shape to replace the original shape.



Highlight both shapes to reshape the original shape.

7. When the shape you want is highlighted in yellow, click.

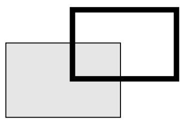
Adding a Rectangle to a Polygon

To add a rectangle to a polygon,

1. Choose *Edit – Reshape* [`Shift-r`].

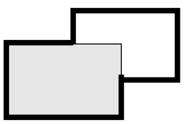
   The Reshape form appears.

2. Select a polygon.
3. Turn on *Reshape Type: rectangle*.
4. Create a rectangle that intersects the polygon.

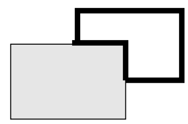


One corner of the rectangle must intersect the original shape.

5. Click right to toggle between highlighting the new shape and highlighting both the old and new shape.



Highlight the new shape to replace the original shape.



Highlight both shapes to reshape the original shape.

6. When the shape you want is highlighted in yellow, click.

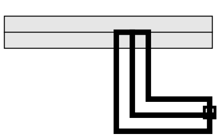
Reshaping a Path

To reshape a path,

1. Choose *Edit – Reshape* [`Shift-r`].

   The Reshape form appears.

2. Select a path.
3. Turn on *Reshape Type: line*.
4. Starting from the centerline, create the new section of the path.



Start the new segment from the path centerline.

5. Double-click when you are finished entering points.

6. Click right to toggle between highlighting the reshaped path options.

Reshape options

7. When the shape you want is highlighted in yellow, click.

Reshaping a Multipart Path

You can reshape a segment of a multipart path in the same way you reshape a segment of a single-part path; however, you must start the reshaped segment on the master path centerline.

The master path and its subparts reshape together. You cannot reshape the master path separately from its subparts, nor can you reshape subparts separately from the master path.

To reshape one segment in a multipart path,

1. Choose *Edit – Reshape* [`Shift-r`].

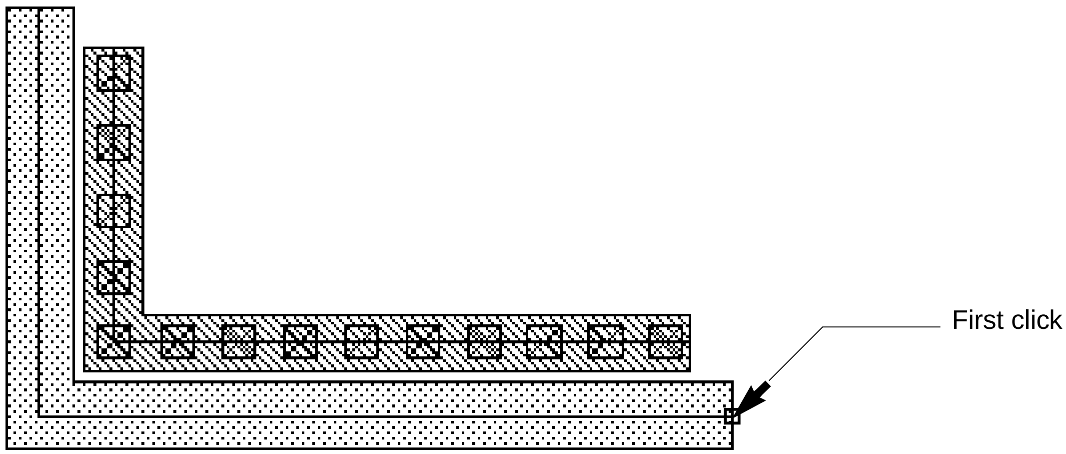
   The Reshape form appears.

2. Select a multipart path.

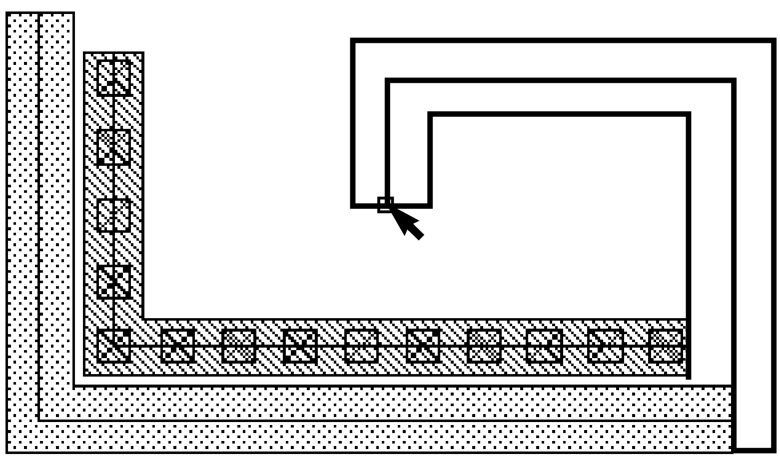
   You are prompted to enter the first point.

3. Turn on *Reshape Type: line*.

4. Click on the centerline of the master path to enter the first point.



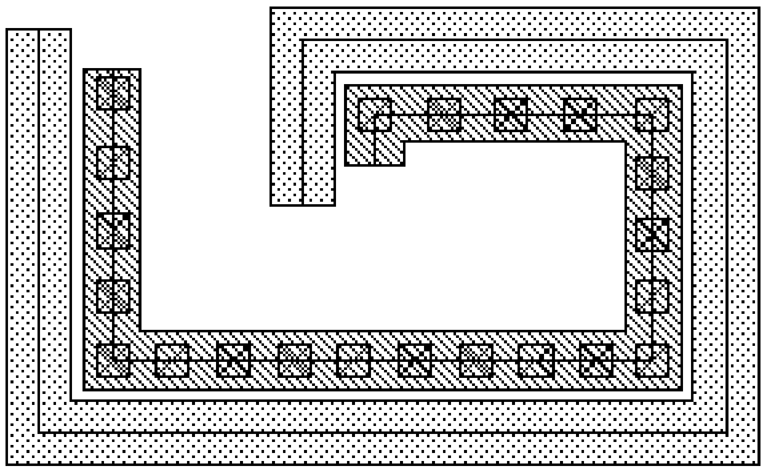
As you move the cursor, an outline of the master path shows the new shape.



5. Double-click when you are finished entering points.

6. Click right to toggle between highlighting the new path and highlighting both the old and new path.

7. When the shape you want is highlighted, click.



For multipart paths, see “Multipart Paths” and the `rodCreatePath` function in the *Virtuoso Relative Object Design User Guide*.

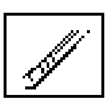
Deleting Objects, Edges, or Corners

To delete an object,

1. Choose the objects, edges, or corners you want to delete.

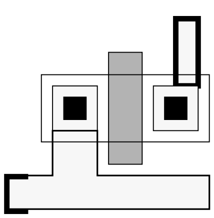
2. Do one of the following:

- Choose *Edit – Delete.*
- Press `Delete`.

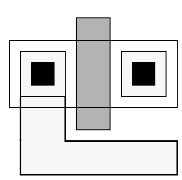


- Click on the delete icon in the icon menu.

All selected objects are deleted.



Select the objects or edges.



The selected objects and edges are deleted.

If you selected an edge or corner, the edge or corner is deleted and the object is redrawn.

Note: You cannot delete a point if it changes the shape so that the shape becomes invalid. For example, a path must have at least two points; if you try to delete one point of a two-point path, you see a warning dialog box. You must close the dialog box and choose a different point to delete.

Canceling Delete

If you discover you deleted the wrong object,

➤ Choose *Edit – Undo* [u] to restore the object.

Delete automatically repeats if you select the *Delete* command first and then choose objects.

➤ If delete automatically repeats, press `Escape` to cancel the command.

Merging Objects on a Layer

You can merge one or more objects created on the same layer into one object. This is sometimes referred to as performing a logical AND on the objects.

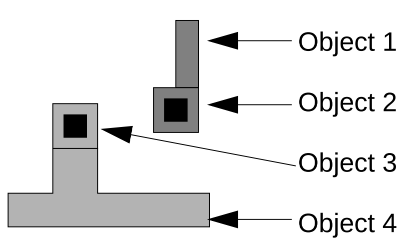
Note: When ROD objects are merged, the resulting shape is not a ROD object.

To merge objects,

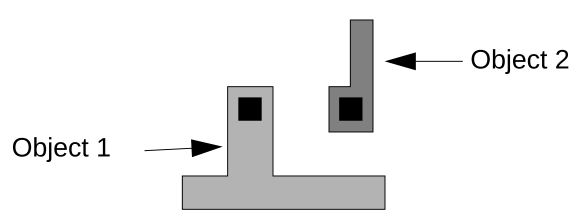
1. Choose *Edit – Merge* [`Shift-m`].
2. Select one or more objects on the same layer. The objects must touch or overlap each other.

   The objects are merged.

   You can click and drag to create a selection box and merge several objects at once. Only those objects that are on the same layer and that touch will be merged.



Select a group of objects while using *Merge*.



Objects on the same layer are merged.

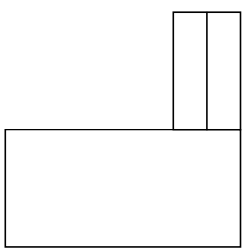
How Merging Converts Objects

When you merge objects using *Merge* or *Layer Generation*, all merged objects are converted to polygons.

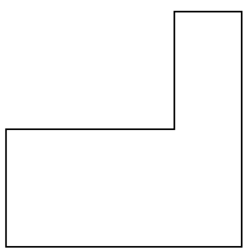
For example, if you merge a group of polygons and paths with *Merge*, the resulting object is a polygon. However, when you merge paths of the same width abutting each other, they will merge into a single path. When you merge overlapping paths, they will merge into a polygon or rectangle.

When you merge objects, the resulting polygon should not have more than 2,047 points (vertexes).

Note: When ROD objects are merged, the resulting shape is not a ROD object



A polygon and a path before merging.

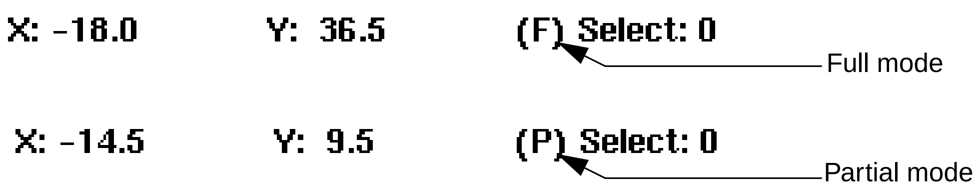


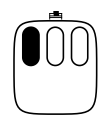
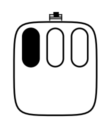
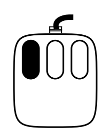
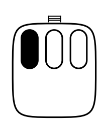
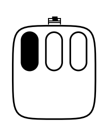
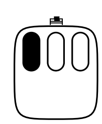
After merging, the new object is a polygon.

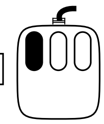
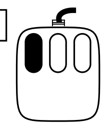
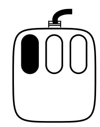
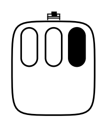
Selecting and Deselecting Objects

Selecting Objects

There are two selection modes: full and partial. In full mode, you can select an entire object. In partial mode, you can select entire objects, edges, or corners of objects. Press `F4` to toggle the selection mode. The selection mode is displayed in the status banner.



To Select...	Do This	
One object	Click on the object.	
One edge or corner	Press `F4` to turn on partial selection. Then click on the edge or corner.	F4 
A group of objects	Click and drag to create a selection box around the group.	
An additional object	Press `Shift` and click inside the object.	Shift 
	Press the a key and click on the object.	A 
An additional edge or corner	Press `F4` to turn on partial selection. Then press `Shift` and click on the edge or corner.	F4 Shift 

To Select...	Do This	
An additional group	Press `Shift` and click and drag to create a selection box around the objects.	Shift 
	Press `Shift-a`, then create a selection box around the objects.	Shift A 
A group of edges or corners	Press `F4` to turn on partial selection. Then create a selection box around the edges or corners.	F4 
All objects	Press `Control-a`, or choose *Edit – Select – Select All.*	Control A
An object located under other objects or cycle through overlapping objects	Select the top object, press `Control-y`. Continue to press `Control-y`, or click right, until you select the object you want.	Control Y 

Selecting an Object or Edge

To select an object or edge,

1. Do one of the following:
   - To select a single object, move the pointer so the object you want is highlighted with a dotted line and click.
   - To select an edge or corner, press `F4`, then move the pointer so the object you want is highlighted with a dotted line and click.

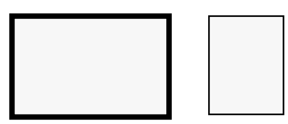
Point inside an object to highlight it.

Point to an edge to highlight it.

Point to a corner to highlight it.

2. When the object or edge you want is highlighted, click.

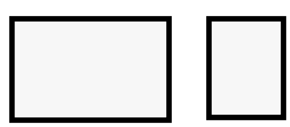
Click on the object.



The object is selected.

3. Press `Shift` and click to select an additional object or edge.

Press `Shift` as you click.



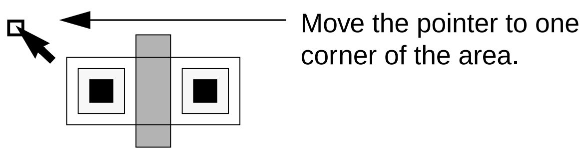
A second object is selected.

Selecting a Group of Objects

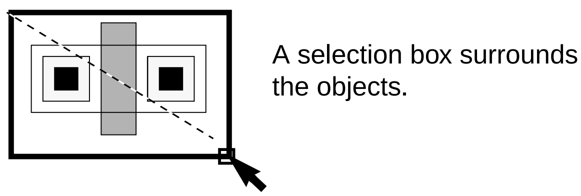
You can select a group of objects by creating a selection box around them. This is sometimes referred to as window selection.

To select a group of objects,

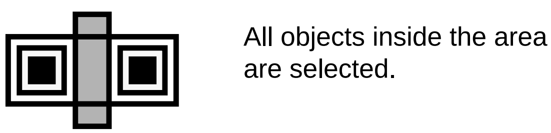
1. Move the pointer to one corner of the area you want to select.



2. Click and drag the pointer to the opposite corner of the area.



3. Release the mouse button.



4. Press `Shift` and create another selection box to select an additional group.

Selecting Objects Under One Selection Point

You can select individual objects that are stacked under one selection point by pressing `Control-y`. The objects must be overlapping under the selection point and as you press `Control-y,` each object is highlighted as the selection process cycles through the stack of objects.

Selecting a Group of Edges

By default, when you select objects by creating a selection box, you select only whole objects. You can set the editor to select a group of edges and corners by turning on partial selection.

To turn partial selection on,

1. Press `F4`.

The status banner shows you are in partial selection mode.

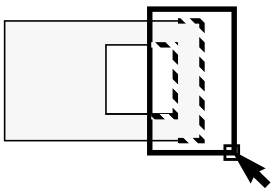
Before pressing F4, full selection is on.

(F) Select: 0

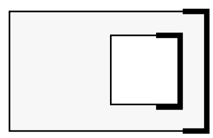
After pressing F4, partial selection is on.

(P) Select: 0

2. Create a selection box around those edges you want to select.



Click and drag to create a selection box.



All edges inside the box are selected. Objects entirely inside the box are selected.

3. To turn partial selection off, press F4 again.

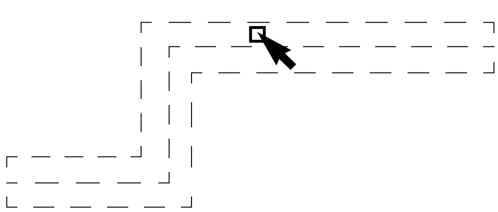
Selecting a Whole Path

To select a whole path,

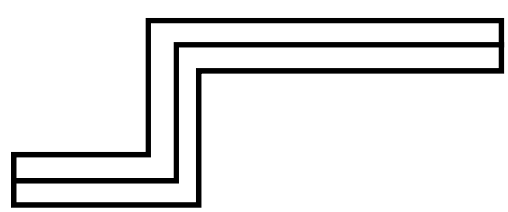
➤ Do one of the following:

- In full selection mode, point anywhere on the path and click.

  As you point, the whole path is highlighted with dotted lines.



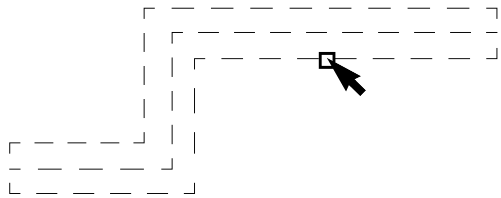
Point anywhere on the path and click.



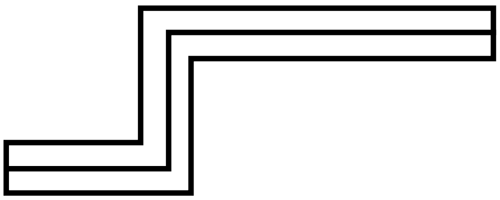
Highlighting shows the whole path is selected.

- In partial selection mode, point to the outer edge of any path segment and click.

  As you point, the whole path is highlighted with dotted lines.

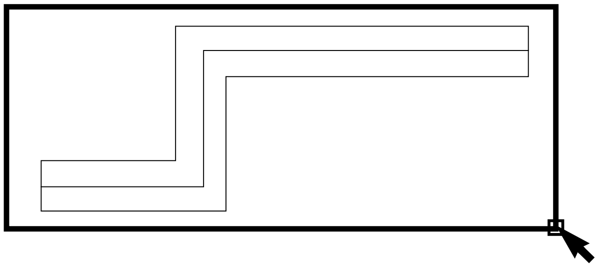


Point to the outer edge of a path segment and click.

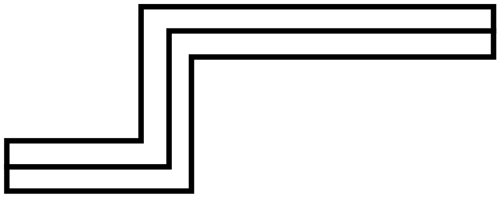


Highlighting shows the whole path is selected.

- In full or partial selection mode, create a selection box around the whole path.



Create a selection box around the whole path.



Highlighting shows the whole path is selected.

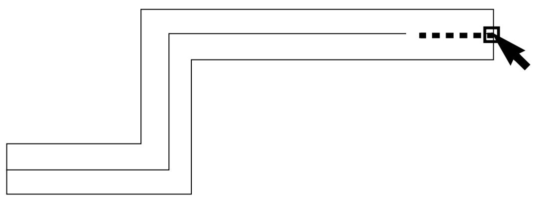
Selecting Path Ends

To select a path end,

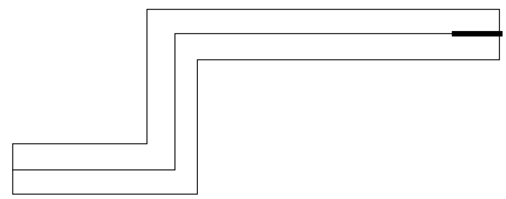
➤ Do one of the following:

- In partial selection mode, point to an end of the path centerline and click.

As you point, the end of the centerline is highlighted with dotted lines.

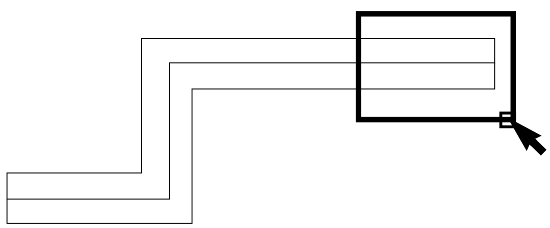


Point to an end of the path centerline and click.

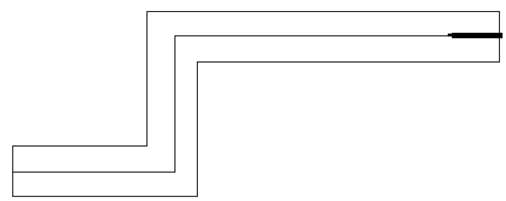


Highlighting shows the path end is selected.

- ❑ In partial selection mode, create a selection box around an end of the path.



Create a selection box around an end of the path.



Highlighting shows the path end is selected.

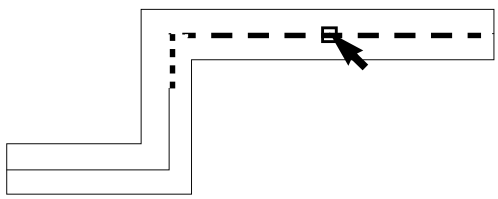
Selecting Path Segments

To select a single path segment,

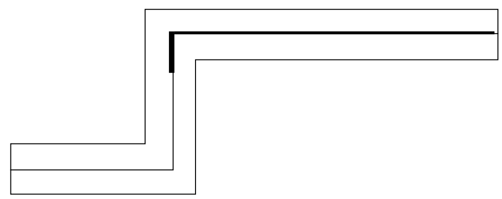
➤ Do one of the following:

- ❑ In partial selection mode, point to the centerline in the middle of a segment and click.

As you point, the centerline of the segment is highlighted with dotted lines.

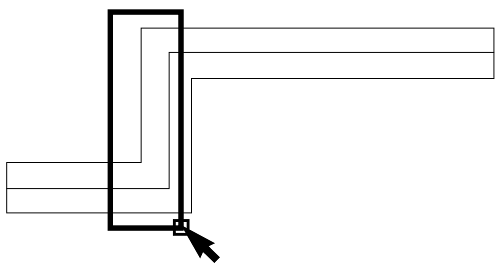


Point to the path centerline in a segment and click.

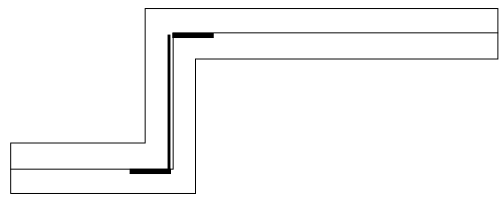


Highlighting shows the path segment is selected.

- ❑ In partial selection mode, create a selection box around a segment. Make sure both end points of the segment centerline are inside the box.



Create a selection box around the centerline of a segment.

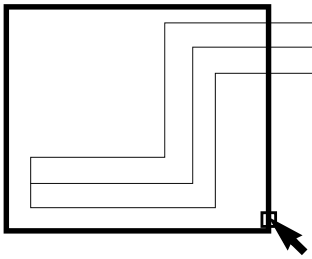


Highlighting shows the path segment is selected.

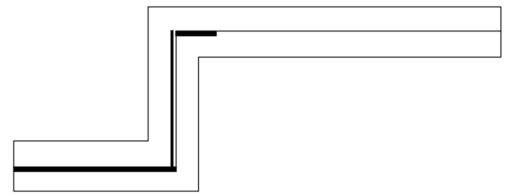
To select more than one path segment,

➤ Do one of the following:

- In partial selection mode, create a selection box around multiple segments. Make sure the end points of each segment centerline are inside the box.



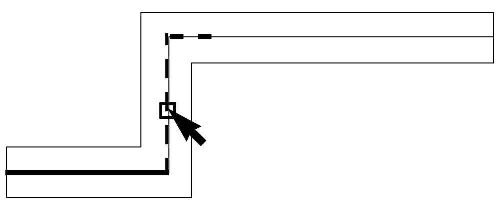
Create a selection box around the centerlines of multiple segments.



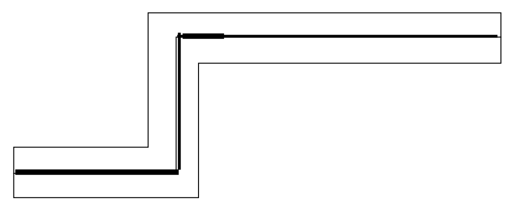
Highlighting shows the path segments are selected.

- In partial selection mode, click on the centerline near the middle of a segment, then press `Shift` and click to select additional segments.

  As you point, each segment centerline is highlighted with dotted lines.



Point to a segment centerline and click, then press `Shift` and click to select another segment.



Highlighting shows the path segments are selected.

Selecting a Path Vertex

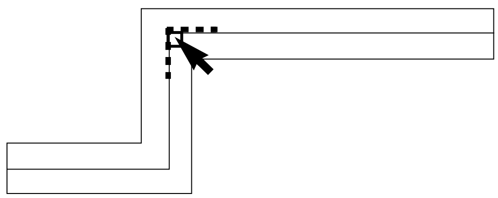
A vertex is a point on the path centerline where two segments join.

To select a vertex,

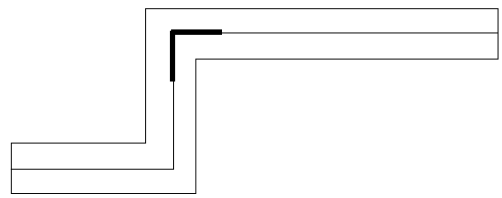
➤ Do one of the following:

- In partial selection mode, point to the vertex and click.

As you point, the vertex is highlighted with dotted lines.



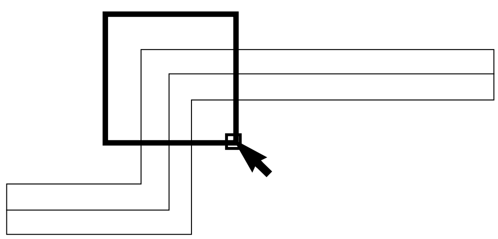
Point to the vertex and click.



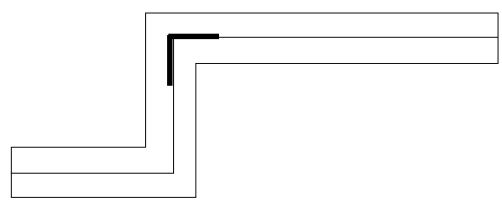
Highlighting shows the vertex is selected.

If you are not able to click on a vertex, the current setting of the *Gravity* option is too restrictive. In the Layout Editor Options form, turn off *Gravity*.

- In partial selection mode, create a selection box around the vertex.



Create a selection box around the vertex.

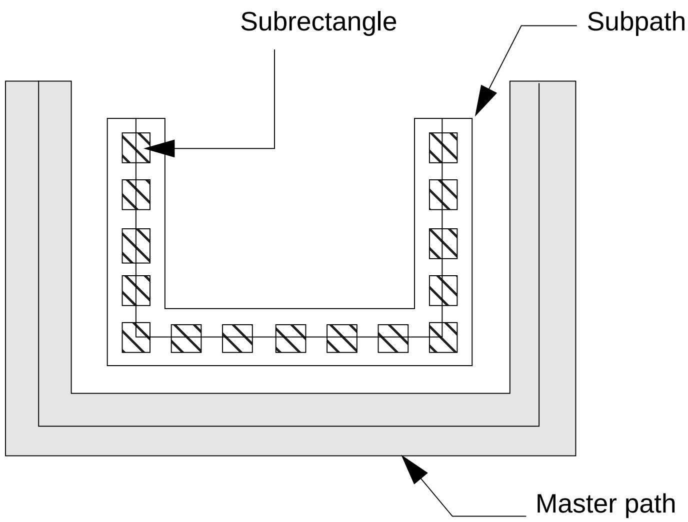


Highlighting shows the vertex is selected.

Selecting Multipart Paths

A multipart path is a single object composed of multiple parts on the same or different layers. The parts are a single master path and one or more subparts. The master path is the primary part; subparts exist in relation to the master path. A subpart can be an offset subpath, an enclosure subpath, or a set of subrectangles. Individual subrectangles are not individual shapes but are part of a specific set of subrectangles.

The multipart path below has one subpath and one set of subrectangles. Both the subpath and the set of subrectangles are offset from the master path.



Selecting multipart paths is similar to selecting regular paths: you point and click or create a selection box in a similar manner. In general, you cannot select subparts separately from the master path; however, you can select the chopped end of subpaths to stretch them.

- In full selection mode, when you select any part of a multipart path, the whole multipart path is selected. The master path is highlighted on the current selection layer, while subparts are highlighted on a different layer. This lets you see which part is the master path and which parts are subparts.
- In partial selection mode, you can select an end, segment, or vertex of the master path.
- In partial selection mode, you can select the chopped end(s) of one or more subpath(s) and stretch them. When you stretch a chopped subpath, all other choppable subparts are stretched also.

Note: You cannot directly select a set of subrectangles, but selecting and stretching the chopped end of a subpath causes all chopped parts in the multipart path to be regenerated, including chopped sets of subrectangles. The system regenerates subrectangles along orthogonal segments only.

Selecting a Whole Multipart Path

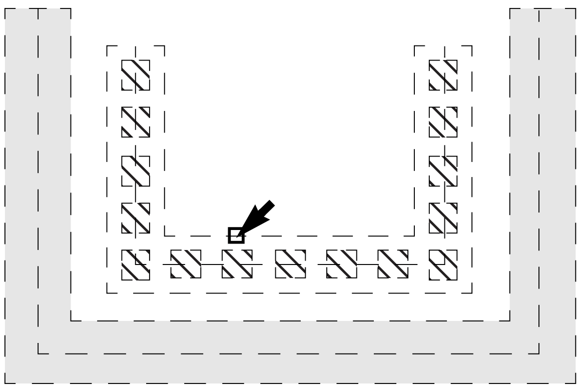
In full selection mode, selecting multipart paths is similar to selecting regular paths. When you select any part of a multipart path, the whole multipart path is selected. The master path is highlighted on the current selection layer, while subparts are highlighted on a different layer. This lets you see which part is the master path and which parts are subparts.

To select a whole multipart path,

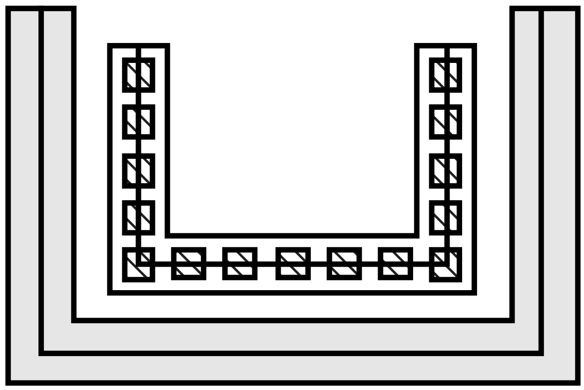
➤ Do one of the following:

- In full selection mode, point to any part of the multipart path and click.

  As you point, the whole path is highlighted with dotted lines.

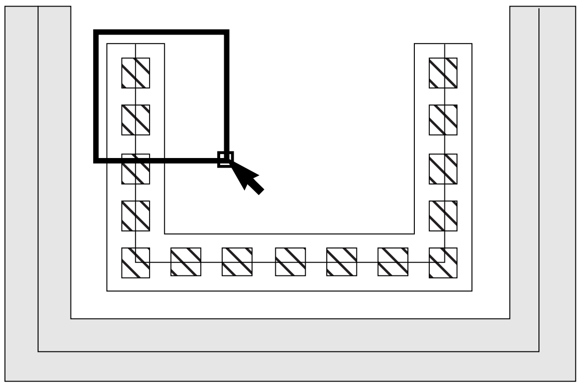


Point to any part of the multipart path and click.

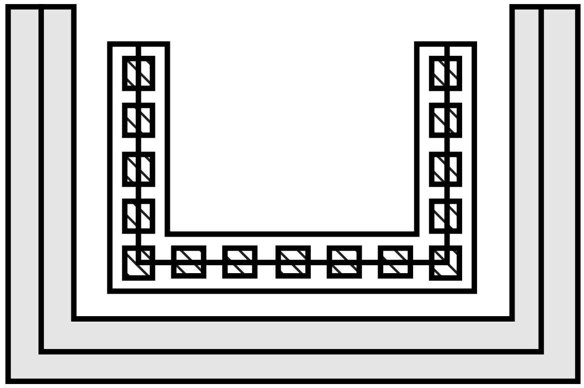


Highlighting shows the whole multipart path is selected.

- In full selection mode, create a selection box around any part of the multipart path.



Create a selection box around any part of the multipart path.



Highlighting shows the whole multipart path is selected.

Selecting Multipart Path Ends

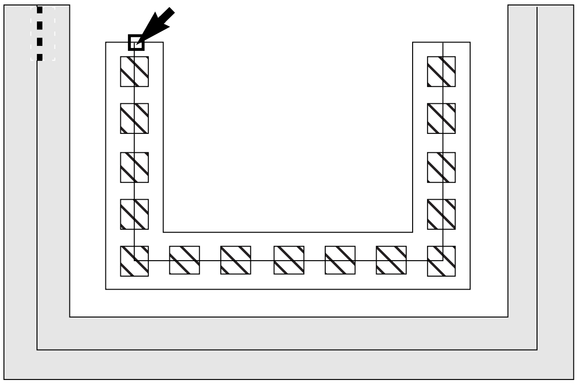
In partial selection mode, you can select an end of a master path in a multipart path. You cannot select the end of a subpart, except for chopped subpath ends. If you attempt to select the non-chopped end of a subpart, the nearest end of the master path is selected.

To select an end of a master path,

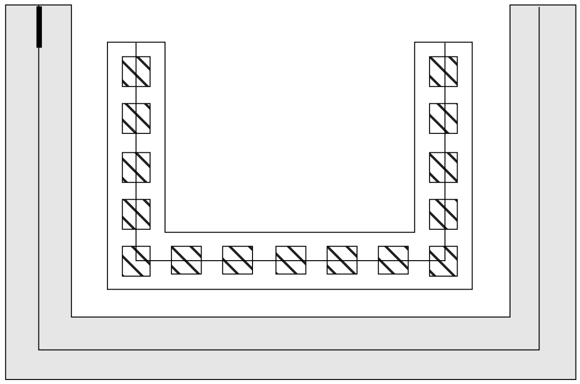
➤ Do one of the following:

- In partial selection mode, point to an end of the master path or an end of a subpath and click.

As you point, the nearest end of the master path is highlighted with dotted lines.

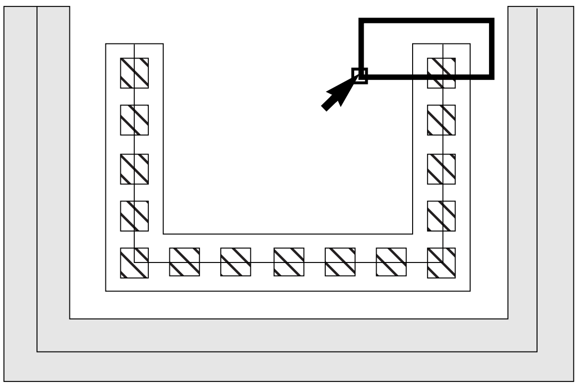


Point to an end and click.

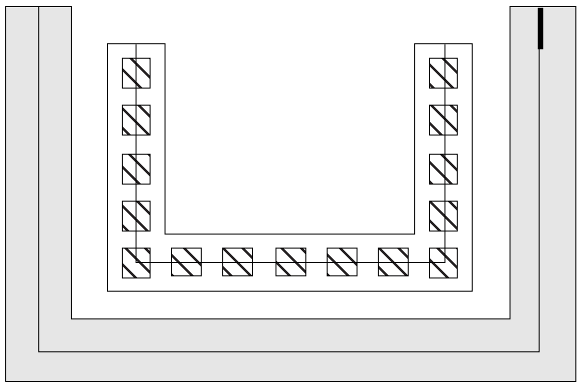


Highlighting shows the (nearest) master path end is selected.

- In partial selection mode, create a selection box around an end of the master path or around an end of a subpath.



Create a selection box around an end of the master path or subpath.



Highlighting shows the (nearest) master path end is selected.

Note: Even in partial selection mode, if you select one or more whole subrectangles, the whole MPP is selected.

Selecting Chopped Subpath Ends in Multipart Paths

In partial selection mode, you can select the end(s) of one or more chopped subpaths in a multipart path and stretch them.

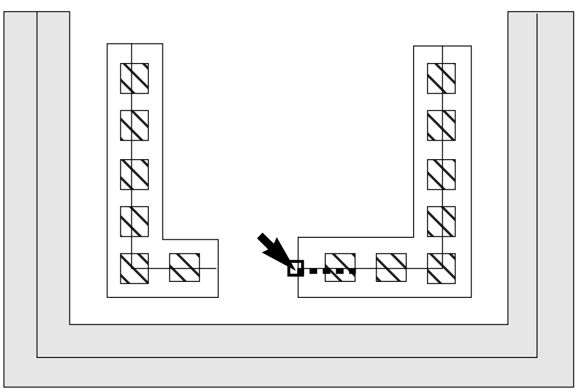
Note: You cannot directly select a set of subrectangles, but selecting and stretching the chopped end of a subpath causes all chopped parts in the multipart path to be regenerated, including chopped sets of subrectangles. The system regenerates subrectangles along orthogonal segments only.

For example, to select the end of one chopped subpath,

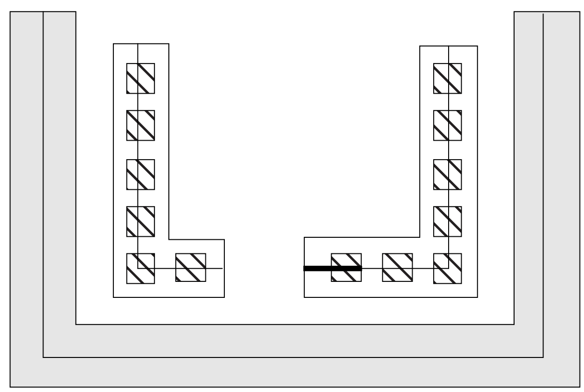
➤ Do one of the following:

- In partial selection mode, point to a chopped end and click.

  As you point, the chopped end is highlighted with dotted lines.

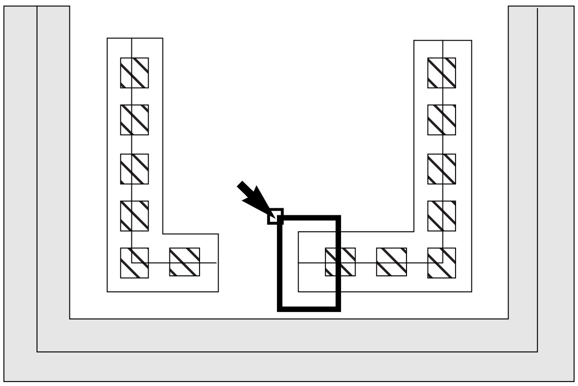


Point to a chopped subpath end and click.

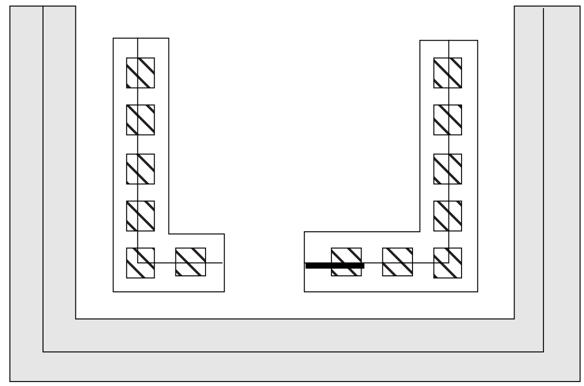


Highlighting shows the chopped subpath end is selected.

- In partial selection mode, create a selection box around the chopped end of a subpath.



Create a selection box around a chopped end of a subpath.



Highlighting shows the chopped subpath end is selected.

Selecting Multipart Path Segments

In partial selection mode, you can select one or more segments of the master path in a multipart path. You cannot select segments of a subpart. If you attempt to select a subpart segment, the nearest segment of the master path is selected.

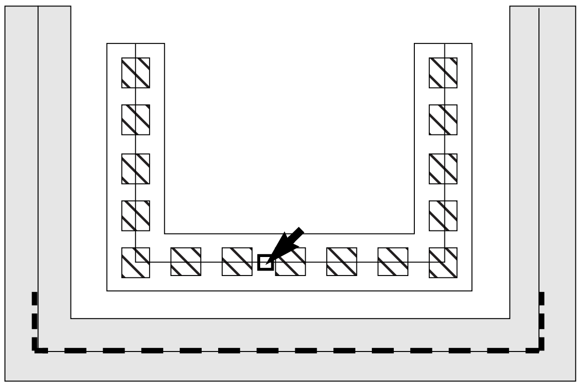
Note: In partial selection mode, if you click on the edge of a subrectangle, the nearest master path segment is selected.

To select a single master path segment,

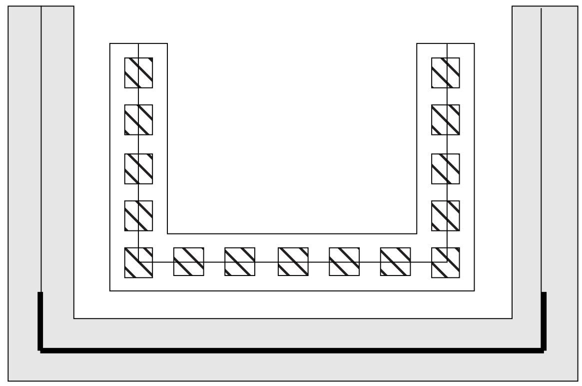
➤ Do one of the following:

- In partial selection mode, point to the middle of the centerline in a master path segment or subpath segment and click.

As you point, the nearest master path segment is highlighted with dotted lines.

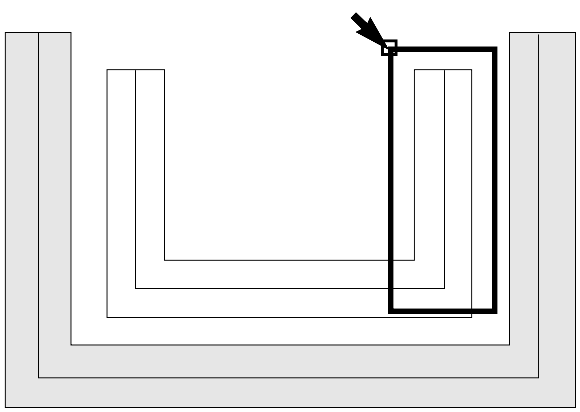


Point to the centerline in a path segment and click.

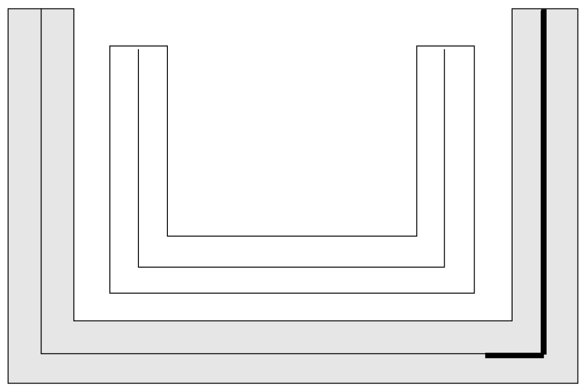


Highlighting shows the (nearest) master path segment is selected.

- ❑ In partial selection mode, create a selection box around a master path or subpath segment. Make sure both end points of the segment centerline are inside the box.



Create a selection box around the centerline of a path segment.

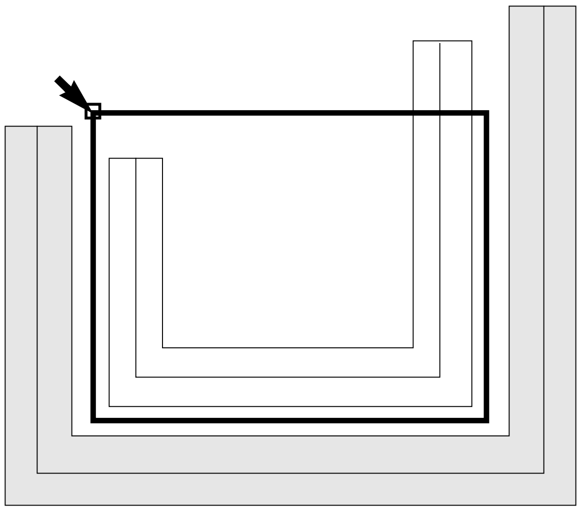


Highlighting shows the (nearest) master path segment is selected.

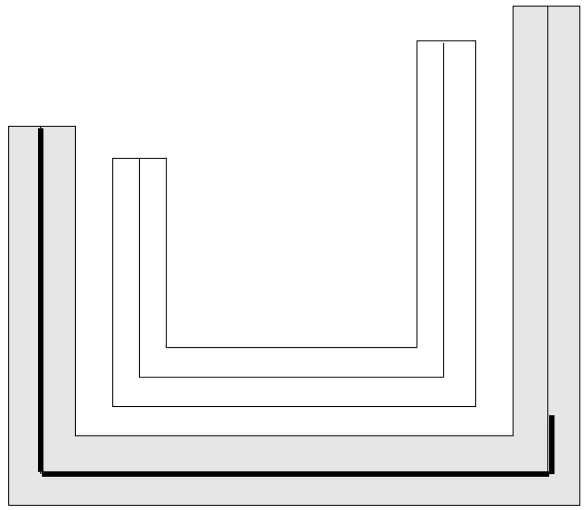
To select more than one master path segment,

➤ Do one of the following:

- In partial selection mode, create a selection box around multiple master path or subpath segments. Make sure the end points of each segment centerline are inside the box.



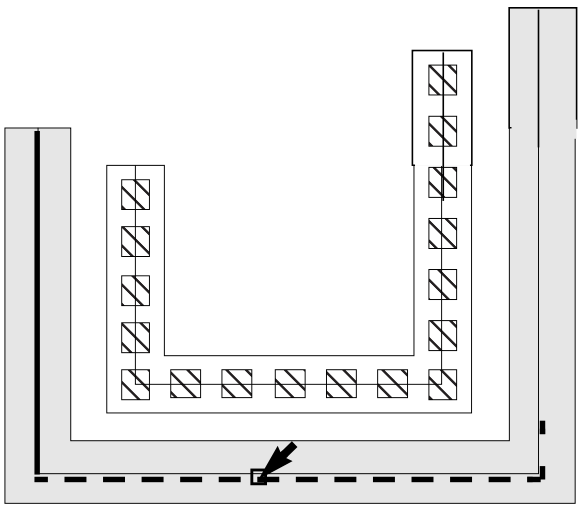
Create a selection box around the centerline of multiple segments.



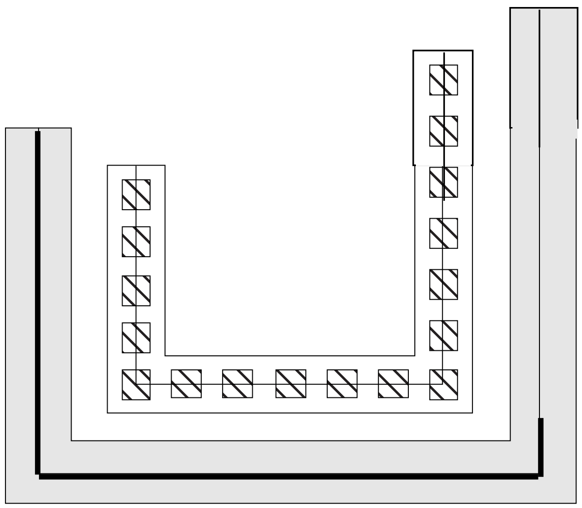
Highlighting shows the (nearest) master path segments are selected.

- In partial selection mode, point to the middle of the centerline in a master path or subpath segment and click. Press `Shift` and click to select additional segments.

As you point, the centerline of the nearest master path segment is highlighted with dotted lines.



Point to a segment centerline and click, then press `Shift` and click to select another segment.



Highlighting shows the path segments are selected.

Selecting a Multipart Path Vertex

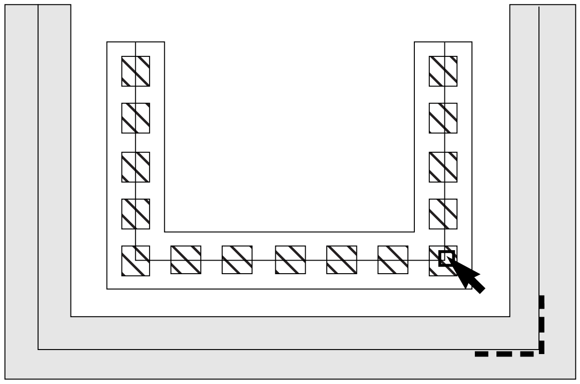
For multipart paths, a vertex is a point on the master path centerline where two segments join. You can select a vertex of the master path in a multipart path. To do so, you either select the master path vertex directly or select the nearest vertex of a subpath.

To select a vertex of the master path,

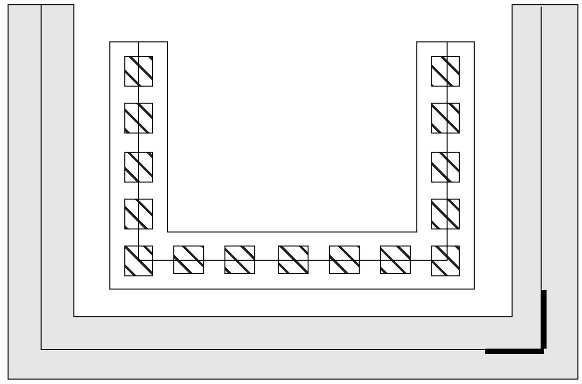
➤ Do one of the following:

- In partial selection mode, point to a vertex on the centerline of the master path or centerline of a subpath and click.

As you point, the nearest master path vertex is highlighted with dotted lines.



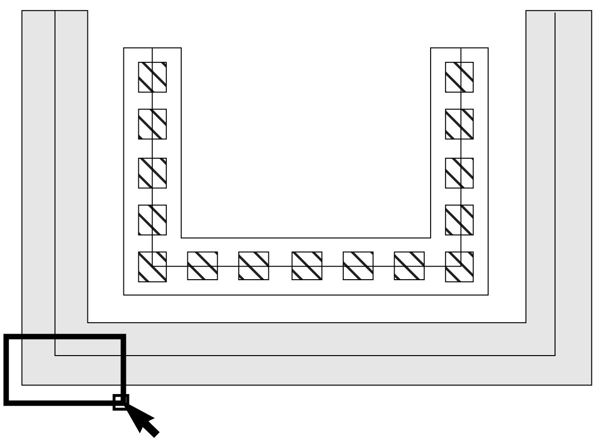
Point to a vertex on the master path or subpath and click.



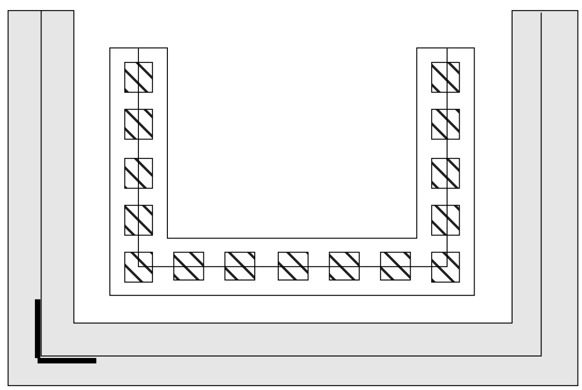
Highlighting shows the (nearest) master path vertex is selected.

If you are not able to click on a centerline vertex, the current setting of the *Gravity* option is too restrictive. In the Layout Editor Options form, turn off *Gravity*.

- In partial selection mode, create a selection box around a vertex on the centerline of a master path or subpath.



Create a selection box around a vertex of the master path or a subpath.



Highlighting shows the (nearest) master path vertex is selected.

Note: If you click on the corner of a subrectangle, the nearest master path vertex is selected.

For more information about multipart paths, see "Multipart Paths" and the `rodCreatePath` function in the *Virtuoso Relative Object Design User Guide*.

Deselecting Objects

You deselect objects almost the same way you select objects, except you press `Control` as you click on an object, corner, or edge.

Here is a quick reference to all of the ways you can deselect objects.

To Deselect...	Do This
All objects	Click on an empty portion of the design, or press `Control-d`, or choose *Edit – Select – Deselect All*.
One object	Press `Control` and click on the object.
A group	Press `Control` and click and drag to create a deselection box around the objects.

Pre- and Postselection of Objects

You can select objects either before or after you start a command.

If you select the object before starting the command (preselect),

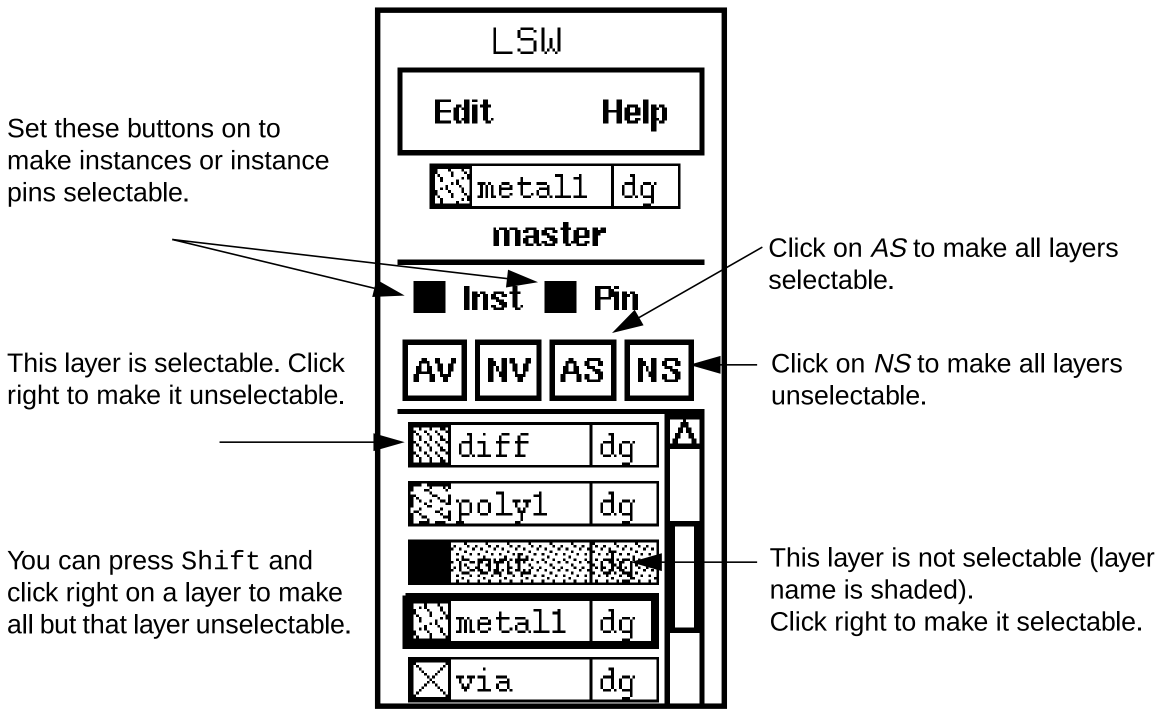
- Editing commands do not automatically repeat, even if repeat mode is set on
- Some editing commands prompt you for a starting point, called the reference point, for the edit

If you select the command and then the object (postselect),

- Editing commands automatically repeat if repeat mode is on (the default)
- Editing commands do not necessarily prompt you for a reference point. If you click to select the first object, the editor uses that point as the reference point.

Making Objects Selectable or Unselectable

You can use the Layer Selection Window to set whether instances, pins, or objects created on specific layers are selectable or unselectable.



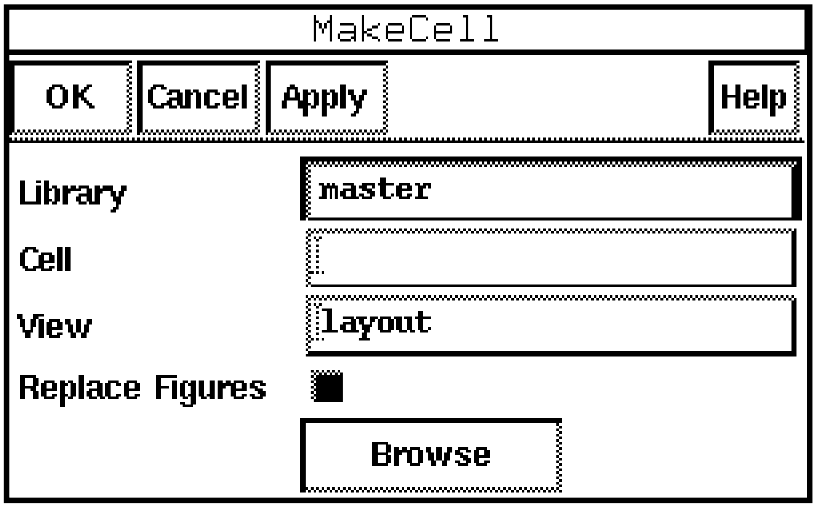
Adding or Removing Levels of Hierarchy

You can copy selected objects in the current cellview into a new cell with the Virtuoso® layout editor.

About the Make Cell Form

To open the Make Cell form,

➤ Choose *Edit – Hierarchy – Make Cell*.



Library sets the library in which to create the new cell.

Cell sets the name of the new cell.

View sets the view name of the new cell.

Replace Figures replaces the selected objects in this cellview with an instance of the new cell.

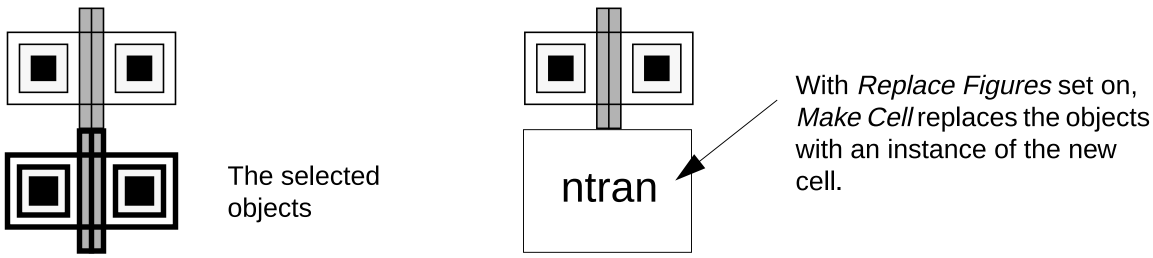
Browse lets you select the library, cell, and view names by clicking on them in the browser.

Creating a Cell from the Selected Set

To create a cell from objects in your layout,

1. Select the objects you want to place into a new cell.
2. Choose *Edit – Hierarchy – Make Cell*.
3. Type the library, cell, and view names for the new cell.
4. In the Make Cell form, do one of the following:
    - Turn *Replace Figures* on to replace the selected objects with an instance of the new cell.
    - Turn *Replace Figures* off to leave the objects unchanged.
5. Click *OK*.

Note: If you type the name of an existing cell, a dialog box lets you choose to overwrite the existing cell with a new one.



Flattening Instances

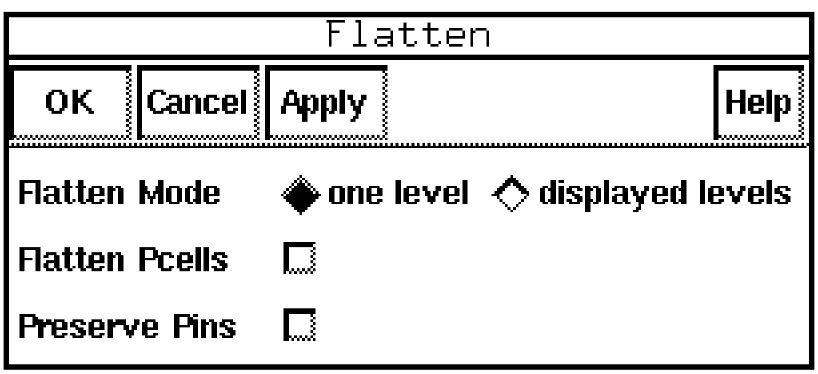
The *Flatten* command moves the contents of a cell or array up one or more levels in the hierarchy.

When a ROD object is flattened, the system assigns the flattened object a name based on the hierarchical name of the ROD object by replacing slashes with dashes. For example, when you flatten ROD object `I1/I4/rect3`, the resulting object is named `I1-I4-rect3`.

About the Flatten Form

To open the Flatten form,

➤ Choose *Edit – Hierarchy – Flatten*.



Flatten Mode controls the number of design hierarchy levels that are flattened. You can move an instance up one level or move the contents of all displayed instances up to the current cellview level.

Flatten Pcells flattens any selected parameterized cells.

Preserve Pins preserves the connectivity information of flattened pins.

Removing Hierarchy (Flattening Instances)

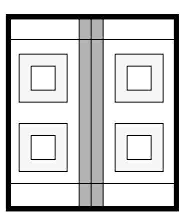
You can move the contents of an instance up into the current cellview. This is often called flattening an instance.

To display the detail in all instances you want to flatten,

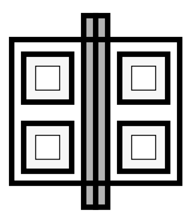
1. Choose *Options – Display* [e].

   The Display Options form appears.

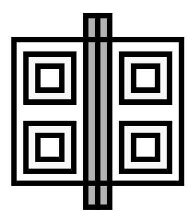
2. Set *Display Controls* to display the detail in all instances you want to flatten.
3. Click *OK*.
4. Choose *Edit – Hierarchy – Flatten*.
5. Set the Flatten Mode to flatten one or all levels of hierarchy.
6. Click *OK*.



The original instance (bold outline) contains four contact instances.



One level flattened—geometries in the contact instances are not flattened.



All levels flattened

Copying and Cutting Through the Hierarchy

Yank and *Paste* can cut and copy all or part of instances and shapes in a designated area.

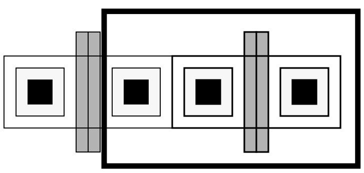
To copy parts of cell instances,

1. Choose *Options – Display* [e].

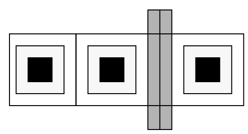
The Display Options form appears.

2. Set *Display Controls* to display the detail in all instances you want to copy.
3. Click *OK*.
4. Choose *Edit – Other – Yank* [y].

   The Yank form appears.
5. Set the *Yank Levels*.
6. Click and drag to create a box around the area you want to yank.
7. Choose *Edit – Other – Paste* [`Shift-y`].
8. Click to place the copies.



The yank box (bold) cuts through the transistor instance on the left and encloses an instance on the right.



Paste places flattened copies of objects from the left instance plus the whole instance on the right.

Cutting Objects

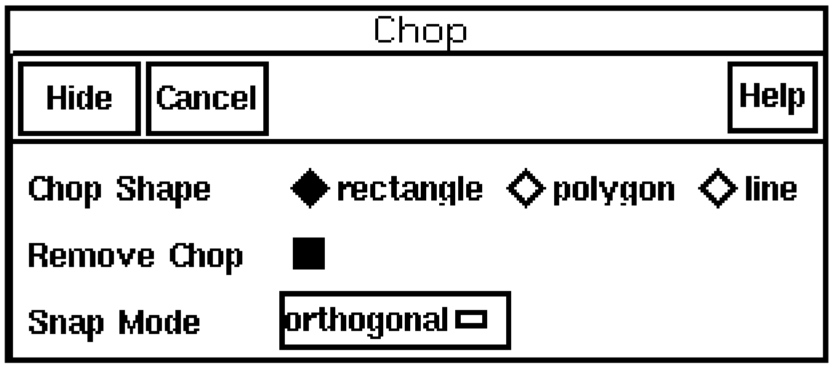
The *Chop* command lets you cut away part of an object or cut an object into pieces.

About the Chop Form

To open the Chop form,

➤ Do one of the following:

- ❑ Choose *Edit – Other – Chop*.
- ❑ Press `Shift-c`.



Chop Shape controls the shape you want to use as the cutter. If *Chop Shape* is set to *line*, *Remove Chop* has no effect.

Remove Chop removes the part of the object enclosed by *Chop Shape*.

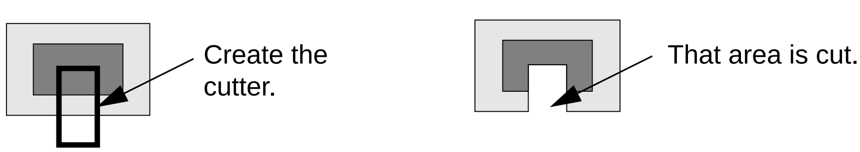
Snap Mode controls the shape of polygon or line segments. *Snap Mode* applies only when *Chop Shape* is set to *polygon* or *line*.

Chopping an Object

To cut away part of an object,

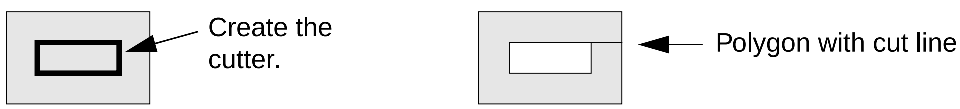
1. Choose *Edit – Other – Chop* [`Shift-c`].
2. Select one or more objects.
3. In the Chop form, set *Chop Shape* to *rectangle*.
4. Click to enter the first corner of the rectangle cutter.
5. Click to enter the opposite corner of the rectangle cutter.

   The second click completes the chop.



   If you create a polygon cutter, you must double-click to complete the polygon cutter and the chop.

If you cut a hole in an object, it is redrawn as a polygon with a cut line.



Splitting an Object into Pieces

To split an object into multiple pieces,

1. Choose *Edit – Other – Chop* [`Shift-c`].
2. Select one or more objects.
3. In the Chop form, set *Chop Shape* to *line*.
4. Click to create the cutter line.
5. Double-click to complete the line and split the object.

Create a line that intersects a rectangle.

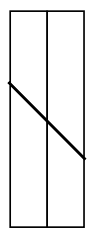
The rectangle is split into three rectangles.

Note: You can also split an object by turning off the *Remove Chop* option in the Chop form and using a rectangle or polygon chop shape. This saves both the area you cut and the original objects.

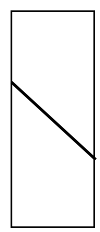
How Chopping Converts Objects to Polygons

When you cut or split any object using *Chop*, the object might be converted to a polygon depending on the *Chop Shape* and *Snap Mode* chosen in the Chop form.

For example, if you cut a path with a diagonal line, the resulting object is a polygon.



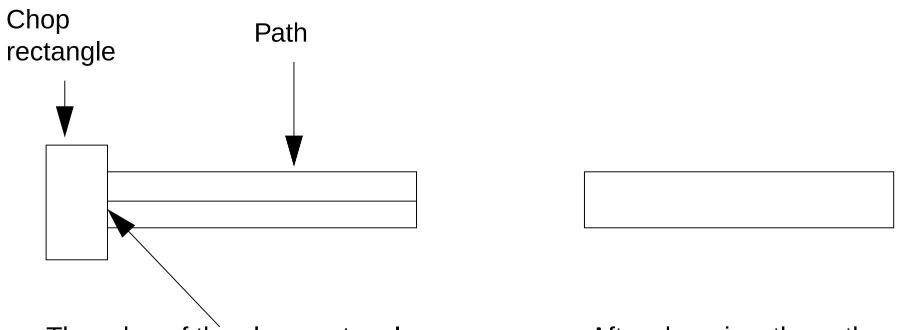
Cutting the path with a diagonal line



After cutting, the path has been converted to a polygon.

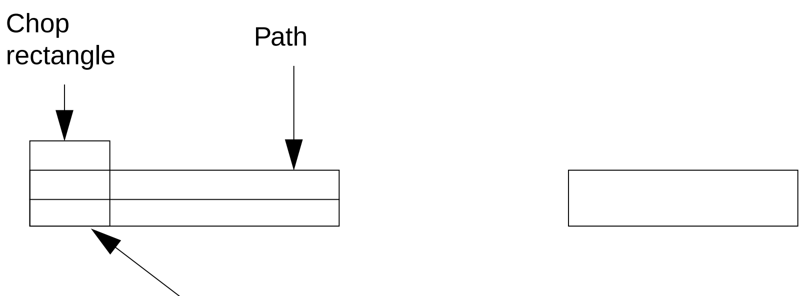
Paths convert to polygons when *Chop Shape* is set to *polygon* or *line* and *Snap Mode* is set to *anyAngle* or *diagonal*.

Paths convert to polygons when *Chop Shape* is set to *rectangle* and the edge of the chop rectangle is coincident with the vertex of a path end and path edge, as illustrated below.



The edge of the chop rectangle touches the path end.

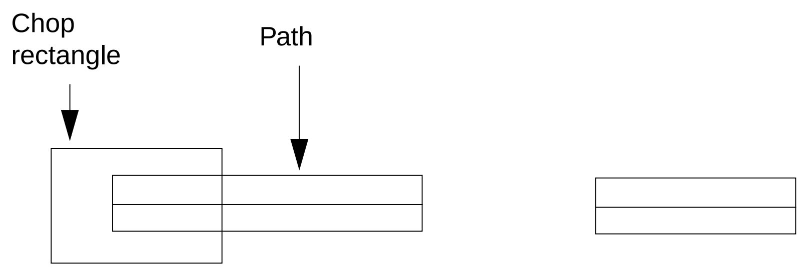
After chopping, the path converts to a polygon.



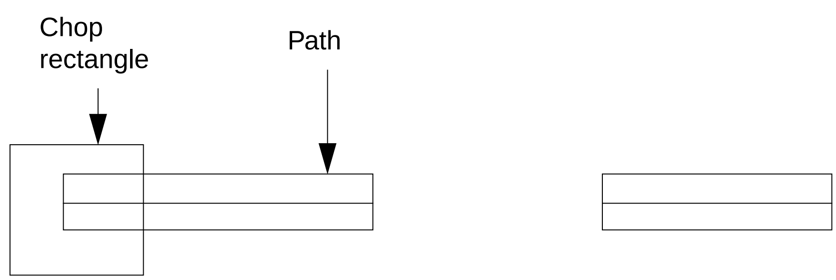
The edges of the chop rectangle touch the path end and path edge.

After chopping, the path converts to a polygon.

To prevent paths from converting to polygons while *Chop Shape* is set to *rectangle*, keep the edges of the chop rectangle away from the vertex of the path end and path edge, as illustrated below.



After chopping, the path remains a path.



After chopping, the path remains a path.

How Chopping Affects Multipart Paths

When you use *Chop* on a multipart path, the result depends on what parts of the multipart path are choppable. You specify whether a part is choppable when you create the multipart path.

- If you specify the master path as choppable, all of its subparts must be choppable also. When you specify the master path as not choppable, you can specify each of its subparts as choppable or not.
- If the master path is choppable, you can chop the whole multipart path into two or more separate multipart paths by chopping all the way through the master path at 90 degrees.

- If the master path is choppable and you chop at an angle other than 90 degrees and/or chop only part of the way through the master path, the chop changes all objects in the multipart path into ordinary paths, polygons, and rectangles. If this is not what you wanted, you can undo the chop with the *Edit – Undo* command.
- If the master path is not choppable, you can chop all subparts that are specified as choppable.
- If the master path is not choppable, the system will not allow you to chop at an angle other than 90 degrees or to chop only part of the way through the multipart path.
- When you chop through sets of subrectangles that are choppable, the system regenerates subrectangles along orthogonal segments only.

After you chop a multipart path, you can select and stretch the chopped ends of subpaths. Although you cannot directly select, stretch, or chop a set of subrectangles, all choppable sets of subrectangles are also chopped when you chop a subpath.

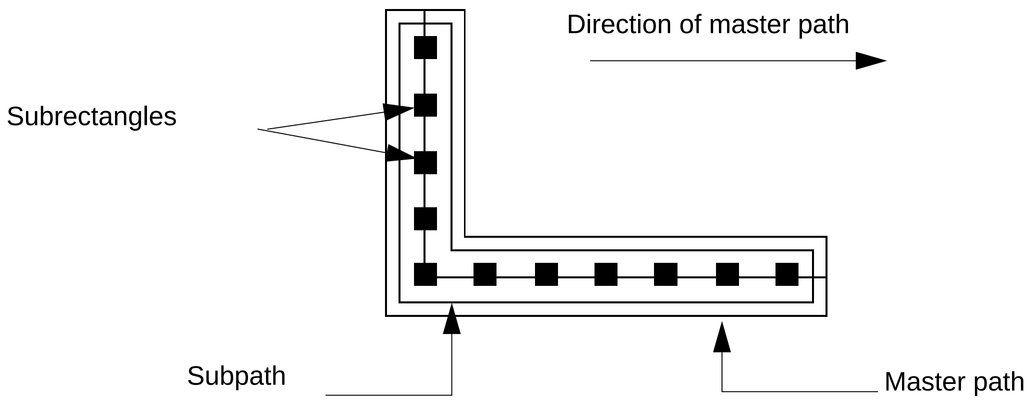
When you chop a multipart path into one or more separate multipart paths, the system assigns the name of the original multipart path to the first new multipart path. The system assigns unique names to the other new multipart paths, starting with `path0`, `path1`, etc.

Chopping Multipart Paths with Aligned Objects

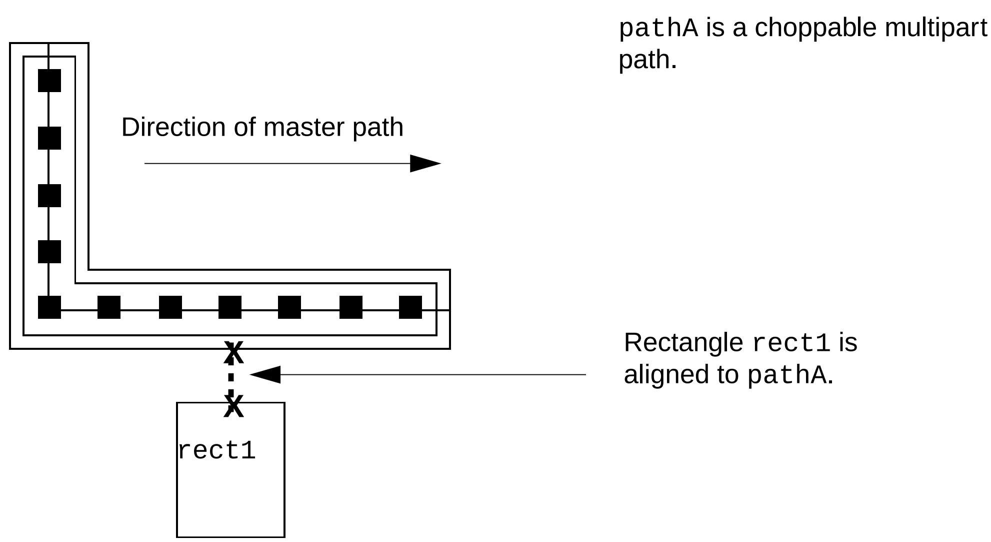
When you use *Chop* with a multipart path,

- If the master path is not choppable, the system preserves alignments of other objects to the multipart path
- If the master path is choppable, the system keeps alignment constraints to other objects with the first new multipart path resulting from the chop, where “first” is relative to the direction of the master path; no objects are aligned to the other new multipart paths

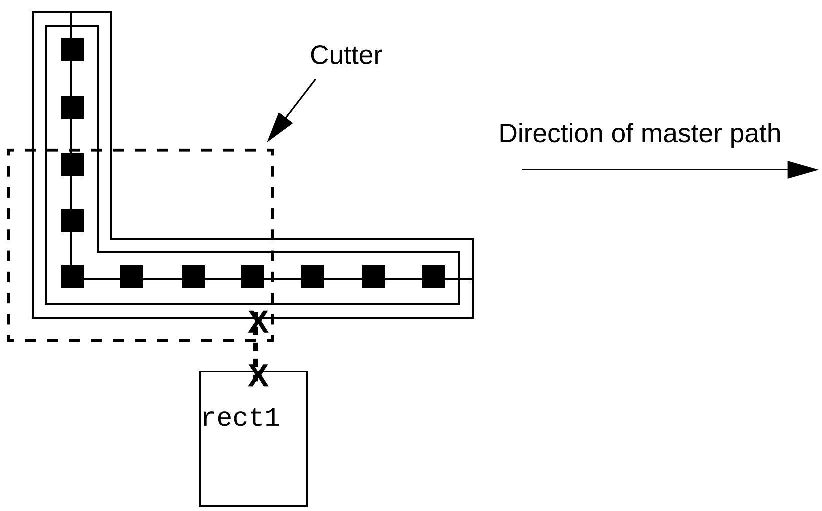
The following example shows what happens to an aligned object after cutting away a section of the multipart path below.



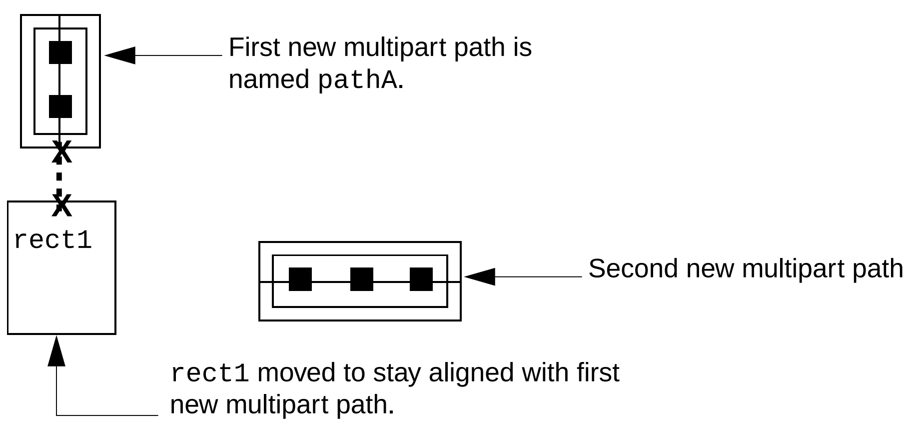
In the example, multipart path `pathA` has a choppable master path. Rectangle `rect1` is aligned to `pathA`, with a separation of -4 in the direction of the Y axis.



When you use the cutter to chop out part of the multipart path like this,



the result is two new, shorter multipart paths.



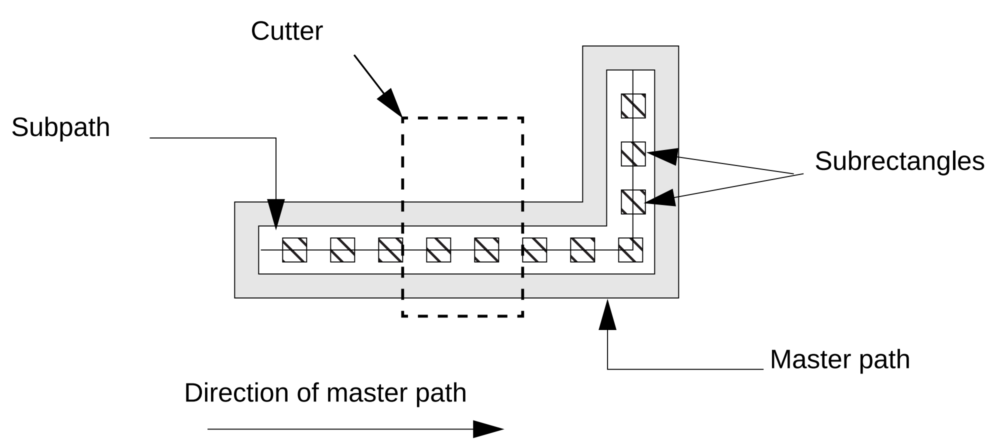
The system keeps the name `pathA` with the first new multipart path and moves the rectangle `rect1` to keep it aligned with `pathA`. The system assigns a unique name in the format of `path`*n* to the other new multipart path.

Chopping a Multipart Path

The multipart path shown below has one subpath and one set of subrectangles.

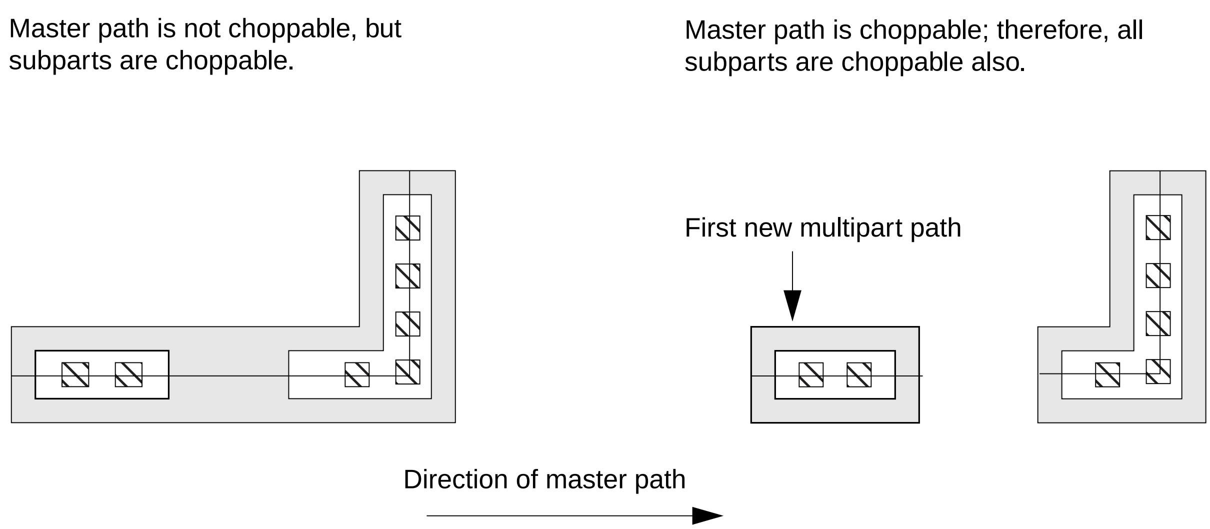
To cut away part of a multipart path,

1. Choose *Edit – Other – Chop* [`Shift-c`].
2. Select the multipart path.
3. Create a chop rectangle around the parts you want to cut away.



To chop a multipart path, you must chop all the way through the master path. It is not necessary to chop through subparts.

The results of the chop depend on whether the master path is choppable or not, as shown below.



For multipart paths, see “Multipart Paths” and the `rodCreatePath` function in the *Virtuoso Relative Object Design User Guide*.

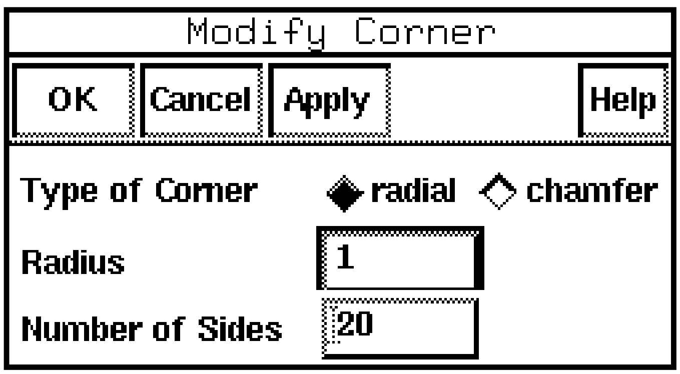
Using the Modify Corner Command

The *Modify Corner* command lets you reshape the corner of a polygon to make a rounded or chamfered (45-degree) edge.

About the Modify Corner Form

To open the Modify Corner form,

➤ Choose *Edit – Other – Modify Corner*.



Type of Corner controls whether you create a rounded corner (radial) or a 45-degree edge (chamfer).

Radius sets the radius of the rounded corner, in user units.

Note: If you choose to create a chamfer corner, the *Radius* field changes to *Distance*.

Distance sets the distance in user units from the vertex to begin beveling.

Number of Sides sets the number of straight segments used to create a rounded corner. This setting is used only if *Type of Corner* is set to *radial*.

Modifying the Corners of a Polygon

To reshape a corner,

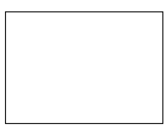
1. Choose *Edit – Other – Modify Corner*.

The Modify Corner form appears.

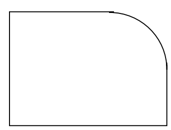
2. Do one of the following:
   - Select *radial* and set *Radius* (the radius of the curve) and *Number of Sides* (the number of segments to create the curve).
   - Select *chamfer* and set *Distance*.

Note: The maximum value you can set is half the length of the shortest adjacent line segment. The layout editor applies this value or the maximum allowable value for each corner, whichever is shorter.

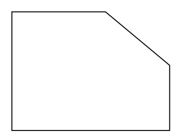
3. Click on the corner you want to change. To change more than one corner, `Shift`-click on each additional corner.
4. Click *OK*.



Original object



Object with a radial corner



Object with a chamfer corner

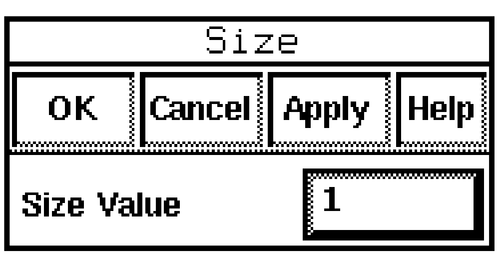
Enlarging or Reducing Objects

The *Size* command enlarges or reduces a shape.

About the Size Form

To open the Size form,

➤ Choose *Edit – Other – Size*.



Size Value controls the amount by which you enlarge or reduce the object. Positive numbers enlarge, negative numbers reduce.

Sizing Objects

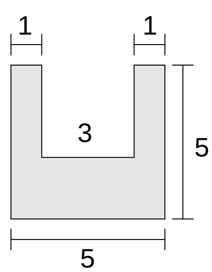
The *Size* command reduces or enlarges objects by stretching each edge in or out by the given number of user units.

To size objects,

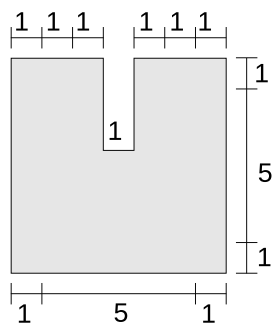
1. Select the objects you want to resize.
2. Choose *Edit – Other – Size*.

   The Size form appears.
3. In the *Size Value* field, type the number of units you want to stretch each side.

   A positive number enlarges the object, a negative number reduces it.
4. Click *OK*.



Object before resizing.



Object after using *Size* with a value of 1 to grow each edge by 1 unit.

Splitting and Stretching Objects

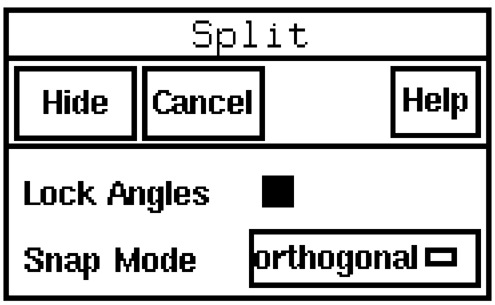
The *Split* command lets you split and stretch a portion of an object. You usually use it to add a jog by stretching a section of a path or group of paths (a bus). You can also use *Split* to add a jog by stretching a section of a multipart path.

About the Split Form

To open the Split form,

➤ Do one of the following:

- Choose *Edit – Other – Split*.
- Press `Control-s`.



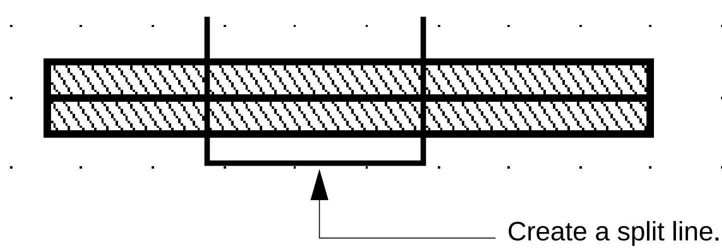
Lock Angles prevents you from changing the angles of objects as you stretch them.

Snap Mode controls the direction in which you can draw line segments when you draw a line to split the object. As you stretch the split object, *Snap Mode* controls the direction in which you can stretch an edge.

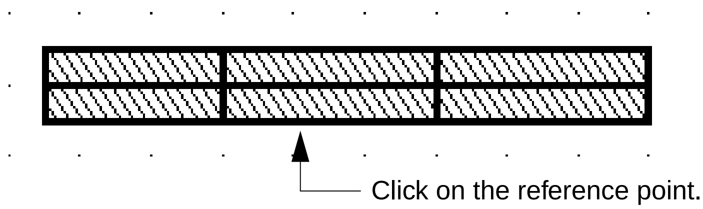
Splitting an Object

To split and stretch an object,

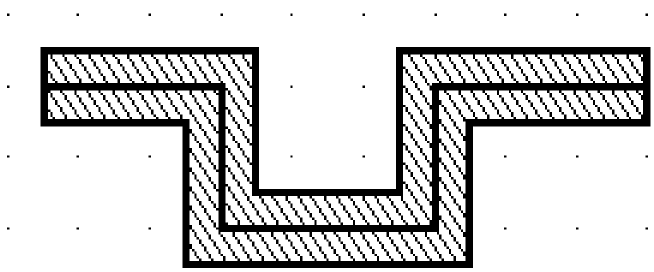
1. Choose *Edit – Other – Split* [`Control-s`].
2. Select the object you want to split.
3. Click on the points of the split line.



4. Click on the reference point for the stretch.



5. Click on the new location for the stretch.



The completed split and resulting stretched object

Attaching and Detaching Objects

Attaching Objects

When you attach one object to another, you create a parent-child relationship between the two objects. The attached object is the child of the object to which it is attached. Parent-child relationships behave in the following manner:

- When the parent moves, all child objects move with it.
- When a child moves, the parent does not move.
- When the parent is deleted, all child objects are deleted.
- When a child is deleted, the parent is not deleted.

To attach objects,

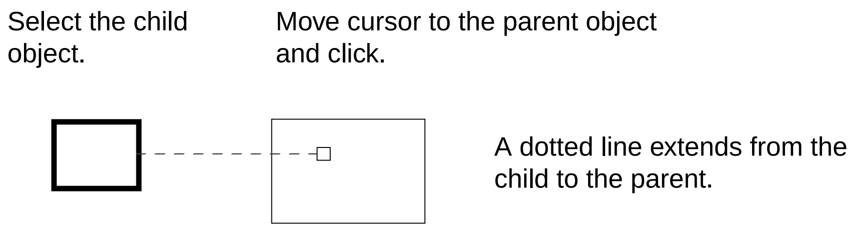
1. Choose *Edit – Other – Attach/Detach* [v].

   The following message appears in the Command Interpreter Window:

   ```
   Select child object to be attached or detached.
   ```

2. Click on the object you want to be the child object. A dotted line extends from the child to the cursor.

3. Move the cursor to the object you want to be the parent object and click.



The child is attached to the parent.

Detaching Objects

To detach objects that have been joined using the *Attach* command,

1. Select all of the child objects you want to detach.

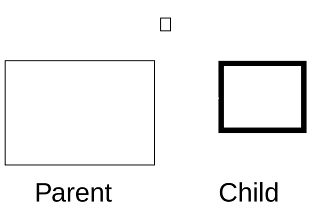
2. Choose *Edit – Other – Attach/Detach* [v].

   The following message appears in the CIW:

   ```
   To attach child object, click on the parent object;
   to detach child object, click in empty area.
   ```

   A dotted line extends from the child to the cursor.

3. Move the cursor to an empty area of your cellview and click.



   The child is detached from the parent.

4. If you selected more than one child object, click again in an empty area.

Using the Rotate Command

The *Rotate* command lets you change the orientation of most geometric objects.

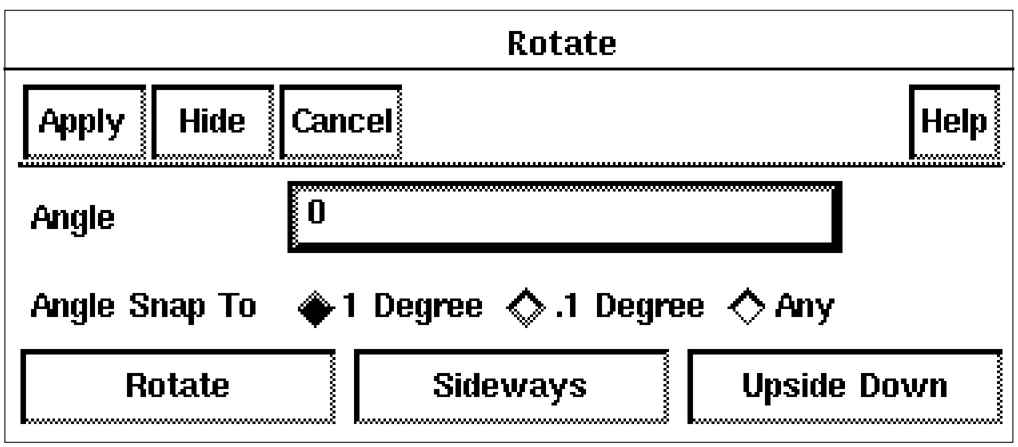
Note: You can rotate any object except donuts and dots with the *Rotate* command. To rotate donuts and dots, use the *Move*, *Copy*, or *Paste* command.

About the Rotate Form

To open the Rotate form,

➤ Do one of the following:

- Choose *Edit – Other – Rotate*.
- Press `Shift-o`.



Angle sets the angle to rotate the object you select. As you move your cursor, the current rotation angle appears here. You can also type the specific angle you want and click *Apply*.

Angle Snap To controls the precision of the angle of the snap.

1 Degree lets you use whole numbers; for example, 10.

.1 Degree lets you use one decimal place; for example, 12.5.

Any lets you use three decimal places; for example, 15.125, assuming 1,000 data base units/user units.

Rotate turns the object, instance, or label 90 degrees counterclockwise.

Sideways mirrors the object about the Y axis or turns an instance or label 180 degrees counterclockwise.

Upside Down mirrors the object about the X axis or turns an instance or label 270 degrees counterclockwise.

Rotating Geometric Objects to Any Angle

To rotate a geometric object to any angle, use the *Rotate* command. To rotate donuts or dots, use the *Move*, *Copy*, or *Paste* command.

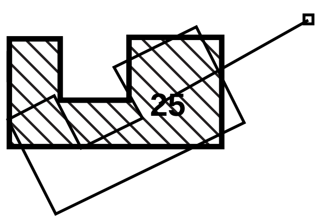
To rotate an object using the cursor,

1. Choose *Edit – Other – Rotate* [`Shift-o`].

   The Rotate form appears.

2. Select the object you want to rotate.
3. Click on a reference point and move the cursor.

   The object rotates and the current rotation angle appears:



4. Click to stop the rotation.

To rotate the object using the Rotate form,

1. Choose *Edit – Other – Rotate* [`Shift-o`].

   The Rotate form appears.

2. Select the object you want to rotate.
3. Click on a reference point.
4. Type a degree in the *Angle* field, or click one of the rotate buttons.
5. Click *Apply*.

Rotating or Mirroring Objects

To rotate any object in 90-degree increments or to mirror an object, use the *Move*, *Copy*, or *Paste* command. To rotate most geometric objects to any angle, use the *Rotate* command. Instances and labels cannot be rotated to other than 90-degree increments. To rotate donuts or dots, use the *Move*, *Copy*, or *Paste* commands.

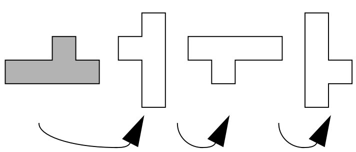
To rotate an object using the Move command,

1. Choose *Edit – Move* [`m`].

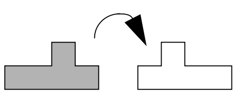
   The Move form appears.

2. Select the objects you want to rotate.
3. Click on the *Rotate*, *Sideways*, or *Upside Down* button.

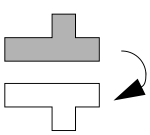
   Continue clicking the buttons to further rotate or mirror the object.



Each time you click *Rotate,* the object rotates 90 degrees counterclockwise.



Sideways mirrors the object along the X axis.



Upside Down mirrors the object along the Y axis.

4. Click where you want to place the rotated object.

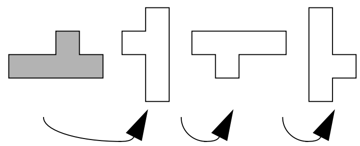
Using the Mouse to Rotate or Mirror Objects

You can use the right mouse button to rotate or mirror objects.

While using *Move*, *Copy*, or *Paste*,

➤ Do one of the following:

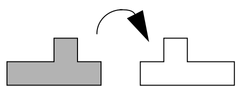
- To rotate the object 90 degrees, click right.



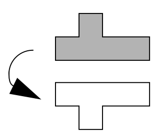
Each time you click right, the object rotates 90 degrees counterclockwise.

- To mirror the object, press `Shift` and click right.

  The object mirrors first along the X axis, then along the Y axis.



The object first mirrors along the X axis.



Next, the object mirrors along the Y axis.

Rotating Instances and Labels

1. Choose *Edit – Other – Rotate* [`Shift-o`].

   The Rotate form appears.

2. Select the instance or label you want to rotate.

   The *Angle* and *Angle Snap To* fields are grayed out because instances and labels can be rotated only at 90-degree increments.

3. Click on the reference point.

4. To rotate the object,

   - 90 degrees, click on Rotate.
   - 180 degrees, click on Sideways.
   - 270 degrees, click on Upside Down.

To rotate instances, labels, donuts, or dots, you can also use the *Move*, *Copy*, or *Paste* commands.

Yanking and Pasting Objects

The *Yank* command performs a “cookie cutter” copy. *Yank* copies objects, and parts of objects, into a temporary buffer. The *Paste* command places copies of objects from a temporary buffer.

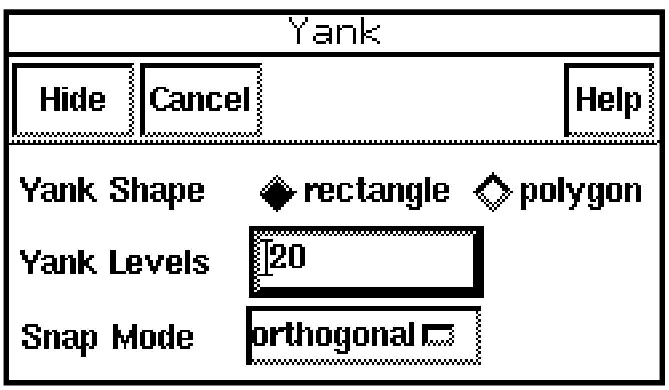
Note: The *Yank* and *Paste* commands do not support ROD objects.

About the Yank Form

To open the Yank form,

➤ Do one of the following:

- Choose *Edit – Other – Yank*.
- Press y.



Yank Shape sets the type of shape you draw around, or through, the objects you want to copy.

Yank Levels sets the number of levels of design hierarchy through which *Yank* can copy shapes. The current cellview is level 0, instances inside it are level 1, and so forth.

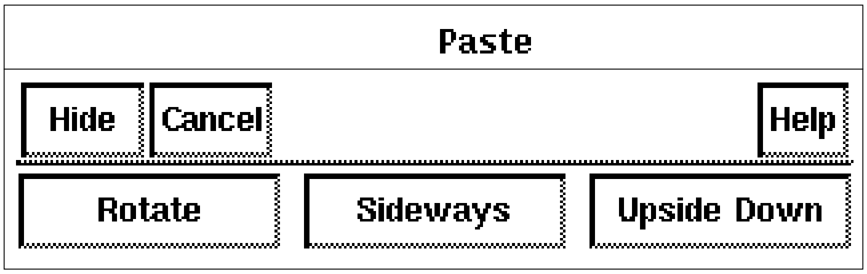
Snap Mode controls the shape of line segments when *Yank Shape* is set to *polygon*.

About the Paste Form

To open the Paste form,

➤ Do one of the following:

- ❑ Choose *Edit – Other – Paste*.
- ❑ Press `Shift-y`.



Rotate turns the object 90 degrees counterclockwise. You can also click the right mouse button to rotate or mirror objects you paste.

Sideways mirrors the object along the X axis.

Upside Down mirrors the object along the Y axis.

Yanking and Pasting

Yank and *Paste* are similar to *Copy* except that objects are chopped by the *Yank* shape. *Yank/Paste* is a “cookie cutter” copy.

To copy part of an object, group of objects, or an instance,

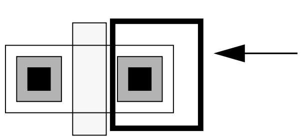
1. Choose *Edit – Other – Yank* [y].

   The Yank form appears. In the Yank form the Yank Shape is set by default to create a rectangle.

2. If you are in full selection mode, press F4 to go to partial selection mode.

3. Create the yank shape around the objects you want to copy.

   All objects and parts of objects inside the box are copied into a special yank buffer. If an instance is inserted by the *Yank* shape, the objects in it are brought to the top level (flattened).

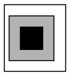


All the objects inside the box you created are copied.

4. Choose *Edit – Other – Paste* [`Shift-y`].

   Outlines of the objects you yanked follow the pointer.

5. Click to place the copied objects.



The copies appear where you clicked.

10

Editing and Defining Properties

This chapter contains these topics:

Understanding Properties

The *Properties* command lets you edit the information that defines selected objects. Every object in your database has information associated with it. The Edit Properties form displays this information, dividing it into the following categories:

- Attributes define the object. Each type of object has specific, built-in attributes. For example, a polygon always has a layer and a set of coordinates that define vertexes. You can view and change attributes of an object.
- The Connectivity category shows information on how this object relates to a net and/or a terminal and associated net expressions.
- Parameters control values associated with a cell. Only parameterized cells (pcells) can have parameters.

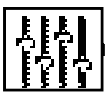
- Properties are user-created information that further define the object. However, some applications add properties.
- ROD displays the relative object design (ROD) name, handle, and alignment information of the object or instance.

About the Edit Properties Form

To open the Edit Properties form,

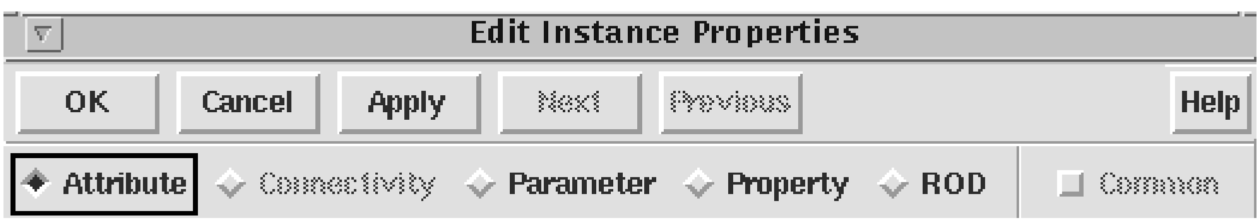
➤ Select an object before or after doing one of the following:

- Choose *Edit – Properties*.
- Type q.



Click on the properties icon in the icon menu.

The title and contents of the form vary with the object you select.



Next highlights the next object in the group and resets the form to show that object's properties.

Previous highlights and resets the form for the previous object in the group.

Attribute represents the characteristics of the object. The available attributes vary, depending on the type of object. If the object is a ROD multipart path (MPP), the Subpart option appears. See "Viewing the Attributes of a ROD Multipart Path" on page 373.

Connectivity displays routing and net information about selected pins and shapes on a net. Only pins display *I/O Type* or *Access Direction*.

Net Name displays the name of the net to which the pin is connected. You cannot edit this field.

Terminal Name sets the name of the terminal which is associated with this pin. The terminal name should always be the same as the net name.

Net Expression assigns a net expression of the terminal listed in the *Terminal Name* field.

Property sets the override property name to the net expression.

Default defines the net to be used if no override property is defined in the hierarchy above this point (in the schematic view). Unless a different signal name is entered, the terminal name is used.

Net Status shows whether a hierarchical pin is connected (used by the Cadence® place-and-route tools).

Net Criticality sets a weighting factor that determines the routing priority for this net for the Cadence place-and-route tools.

I/O Type assigns a property used by routers to identify the direction of the signal into or out of this cellview. The signal can be input, output, inputOutput (bidirectional), switch (carries data either in or out, but not simultaneously), or jumper (passes data through this cellview).

Access Direction assigns a property used to identify the part of the pin to which the routers can connect routing. Applies only to rectangle pins.

Parameter displays any parameters defined for a parameterized cell.

Property sets the properties of the object and lets you add your own properties to an object.

Add opens the Add Property form, which lets you add a new cellview property.

Delete deletes a selected property created with *Add*.

Modify opens the Modify Property form, which lets you change the definition for any property created with *Add*.

ROD displays the ROD properties of the object.

Common lets you edit properties common to a group of selected objects. The Common functionality does not support ROD objects.

Layer displays the layer of a common property if it is the same for all objects. If the layer is *not* the same for all objects, *AS IS* appears in the field.

Left, Right, Bottom, and **Top** display the value of a common property if it is the same for all objects. If the value of a common property is not the same for all objects, the words *AS IS* appears in the fields.

Displaying Properties

To display the properties of an object,

1. Choose *Edit – Properties* [q].
2. Select one or more objects.

   The Edit Properties form for the first selected object appears.

3. Click on the appropriate radio button at the top of the form to see attributes, connectivity, parameters, or user-defined properties.
4. Set *Common* to see the common properties of all selected objects.
5. Click *Next* to display the data for another selected object.
6. Click *Previous* to display the data for the previous object.
7. When you are finished, click *Cancel*.

Editing an Object’s Properties

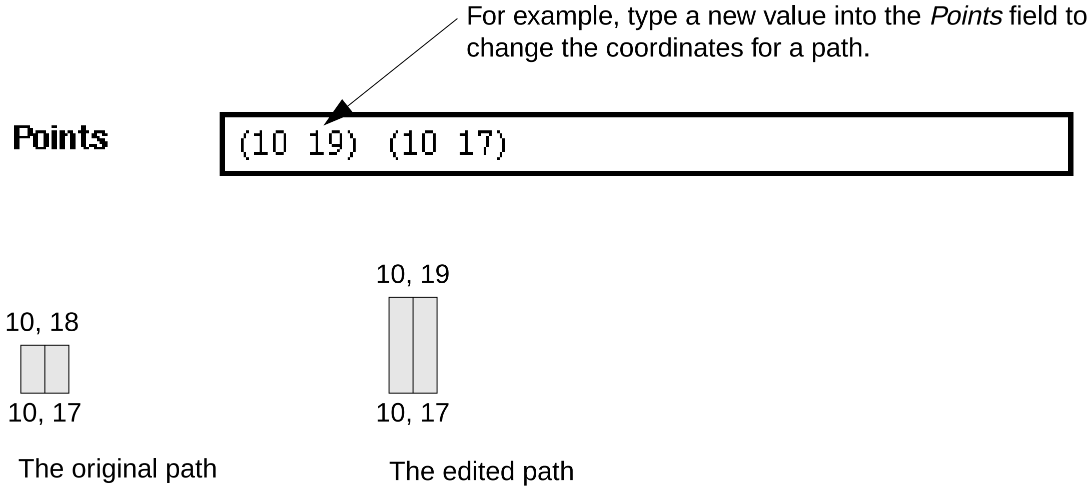
To edit the properties for one object,

1. Select the object.
2. Choose *Edit – Properties* [q].

   The Edit Properties form appears.

3. Type or select new values for the object.
4. Click *OK*.

The object is changed to show the new values.



Editing Properties for a Group of Objects

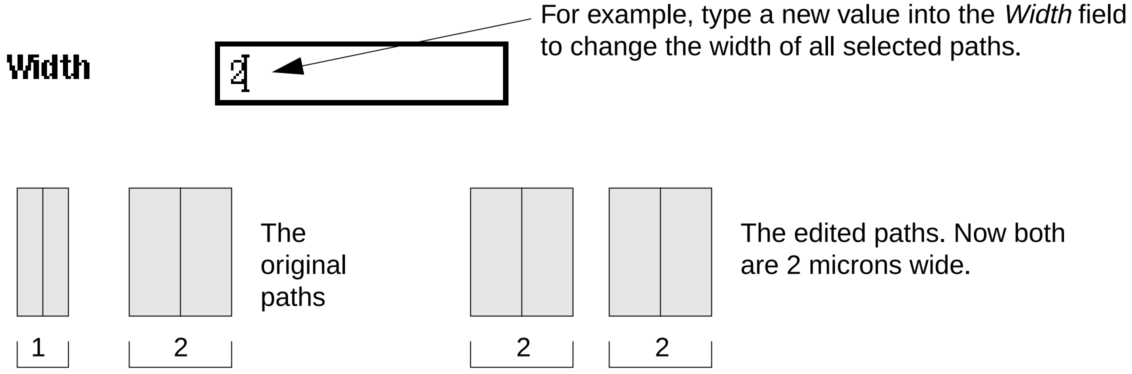
To edit properties for a group of objects,

1. Select all the objects you want to edit.
2. Choose *Edit – Properties* [q].

   The Edit Properties form appears.
3. Set *Common* on.

   The values for all properties common to the selected objects appear. If a property does not apply to all or is different for some of the selected objects, its value appears as *AS IS*, meaning this property cannot be changed on any of the selected objects.
4. Type or select new values for the objects.

5. Click *OK*.

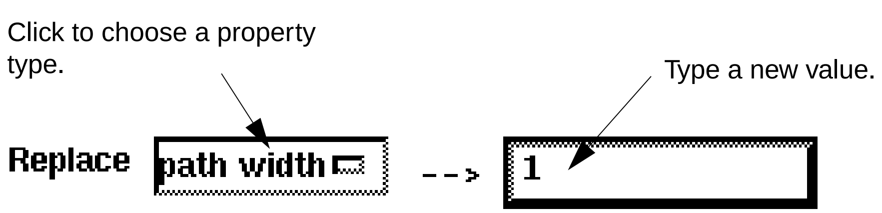


Searching for and Replacing Properties

You can search for one or more objects with identical properties and replace those properties with a new value. This way you can quickly change a large number of objects.

To search for objects and replace their properties,

1. Choose *Edit – Search* [`Shift-s`].
2. Use the Search form to choose the object you want to search for and to set any search criteria.
3. In the *Replace* cyclic field, choose the type of property you want to replace.
4. Type or select the new value.



5. Click *Apply*.

   The layout editor highlights all of the objects it finds.

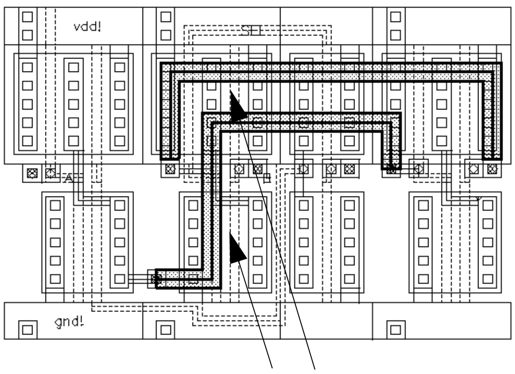
   All the objects are now in a search group. The first object in the group is highlighted in a different color.

6. Click *Previous* or *Next* to select different objects in the search group.

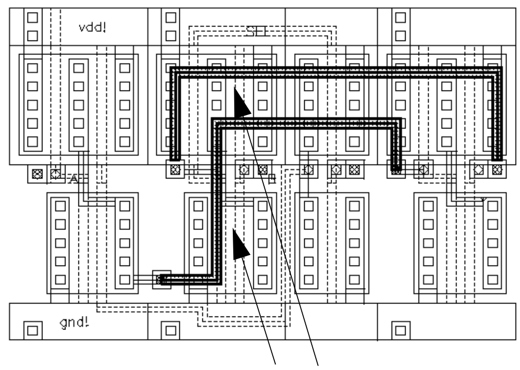
7. To change the properties, do one of the following:

- ❑ Click *Replace* to change the property for the first object in the search stack.
- ❑ Click *Replace All* to change properties for all of the highlighted objects.

8. When you are finished changing objects, click *Cancel*.



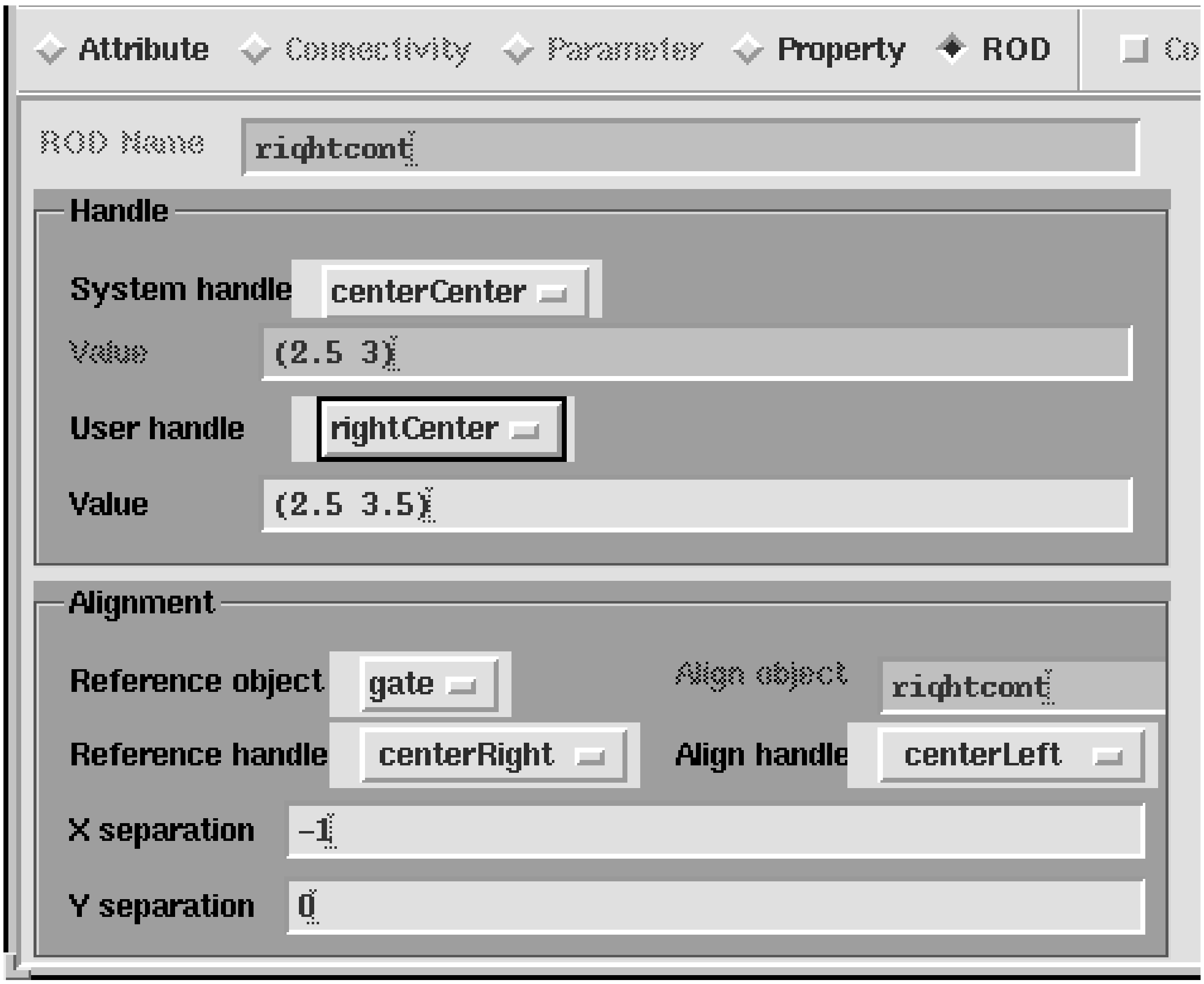
Search highlights the paths it finds.



Replace All changes the width of the paths.

About the ROD Property Form

The ROD property form displays the ROD name, handle, and alignment information of the object or instance.



ROD Name displays the name of the ROD object.

Handle displays the system handle and user handle information. A handle is an attribute of, or item of information about, a ROD object, such as the coordinates of a point on the bounding box around an object, the width of the bounding box of an object, or the resistance of an object.

System handle displays handle names assigned to the object. When you choose a different handle from the cyclic field, the *Value* field updates to reflect the current handle.

Value is a non-editable field displaying the location of the system handle displayed in the *System handle* field. These values are predefined reference points.

User handle displays handle names assigned by the user to the object. When you choose a different handle from the cyclic field, the *Value* field updates to reflect the current handle. If a user handle is not assigned, the fields are empty.

Value displays the current location of the system handle displayed in the *User handle* field. You can change these values by typing new coordinates in this field.

Alignment displays the alignment information about two ROD objects. You can edit these fields if the objects have alignment values assigned to them, otherwise the fields are empty. When looking at the alignment information for two objects, the selected object is *Align object*. The separation is the distance from *Reference object* (unselected) to *Align object* (selected).

Reference object displays the name of the object assigned to be the reference object of *Align object*. If there is more than one object assigned to be *Reference object*, you can choose that object from the cyclic field.

Align object displays the name of the object aligned to *Reference object*. This field is not editable.

Reference handle displays the name of the handle *Align object* is aligned to.

Align handle displays the name of the handle *Reference object* is aligned to.

X separation displays the distance in the X direction from either the reference point handle or the reference point on *Reference object* to the alignment point handle on *Align object*. The value can be positive or negative or a Cadence SKILL language expression that evaluates to a positive or negative number.

Y separation displays the distance in the Y direction from either the reference point handle or the reference point on *Reference object* to the alignment point handle on *Align object*. The value can be positive or negative or a SKILL expression that evaluates to a positive or negative number.

For complete information on setting ROD handles and alignment, see the *Virtuoso Relative Object Design User Guide*.

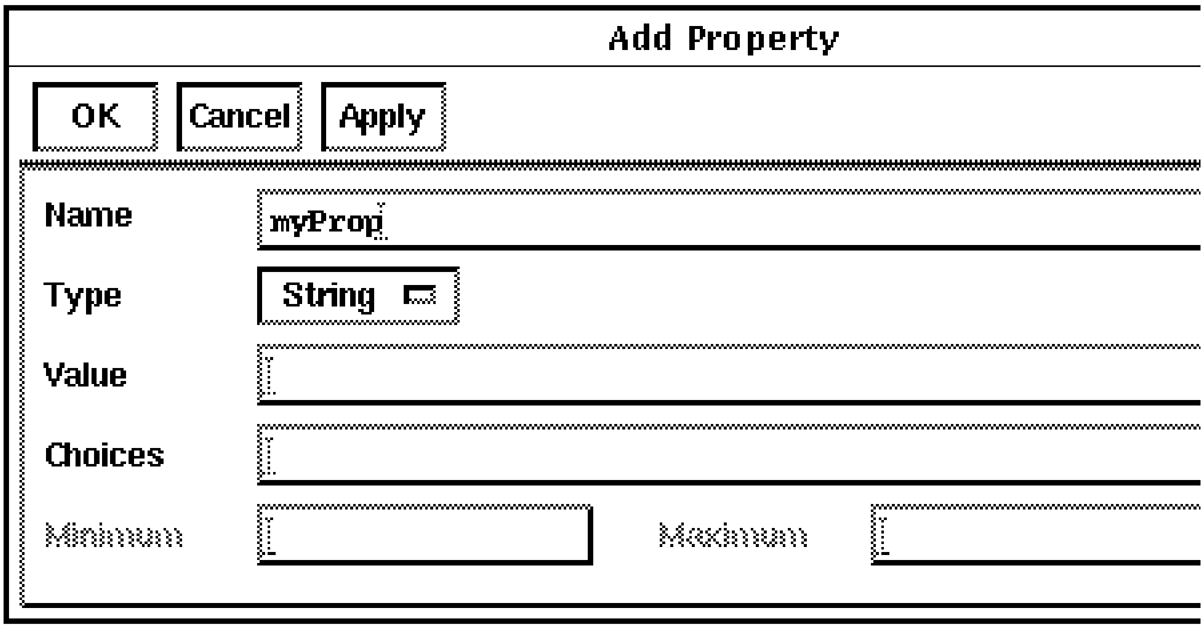
For examples of how to edit aligned ROD objects using the Edit Properties form, see the *Cell Design Tutorial*.

About the Add Property Form

You use the Add Property form to define a new property.

To open the Add Property form,

➤ Click *Add* in the Edit Cellview Properties form or the Edit Properties form.



Name specifies the name you want to assign to this property. This name will appear on the Edit Properties or the Edit Cellview Properties form.

Type controls the type of value for the property. This can be *Int* (integer), *Float* (floating-point number), *String* (any text, displayed in a cyclic field), *Time* (date and time), *Boolean* (on or off), or *ILList* (list of IL values).

Value shows either a text entry field, a Boolean button, or a cyclic field, depending on the *Type* setting.

Choices sets the possible values that appear in the cyclic field for a string property. One of the values listed here must match the default shown in *Value*.

Minimum and **Maximum** display the minimum and maximum values for integer, floating-point, and time properties. The values appear in parentheses after the property name in the Edit Properties or the Edit Cellview Properties form.

Defining a New Cellview Property

A cellview property is information that you or a Cadence application assigns to the cellview. This information can be used to override default information or to add information for further processing by other applications. For example, when postprocessing cells, you could have a SKILL routine use a property to identify which cells to process.

To define a new cellview property,

1. Choose *Design – Properties* [`Shift-q`].

   The Edit Cellview Properties form appears.

2. Click *Property*.

   The properties of the cellview appear. Most of these properties correspond to settings in the Display Options form.

3. Click *Add*.

   The Add Property form appears.

4. Fill out the Add Property form.

5. Click *OK*.

   Your new property appears at the bottom of the Edit Cellview Properties form.

Defining a New Object Property

An object property is information that you assign to a contact, circle, donut, ellipse, instance, label, path, pin, polygon, or rectangle. This information can be used to identify an object or to add information for further processing by other applications.

To define a new object property,

1. Choose *Edit – Properties* [q].

2. Select one or more objects.

   The Edit Properties form for the first selected object appears.

3. Click *Properties*.

   The properties of the object appear.

4. Click *Add*.

   The Add Property form appears.

5. Fill out the Add Property form.

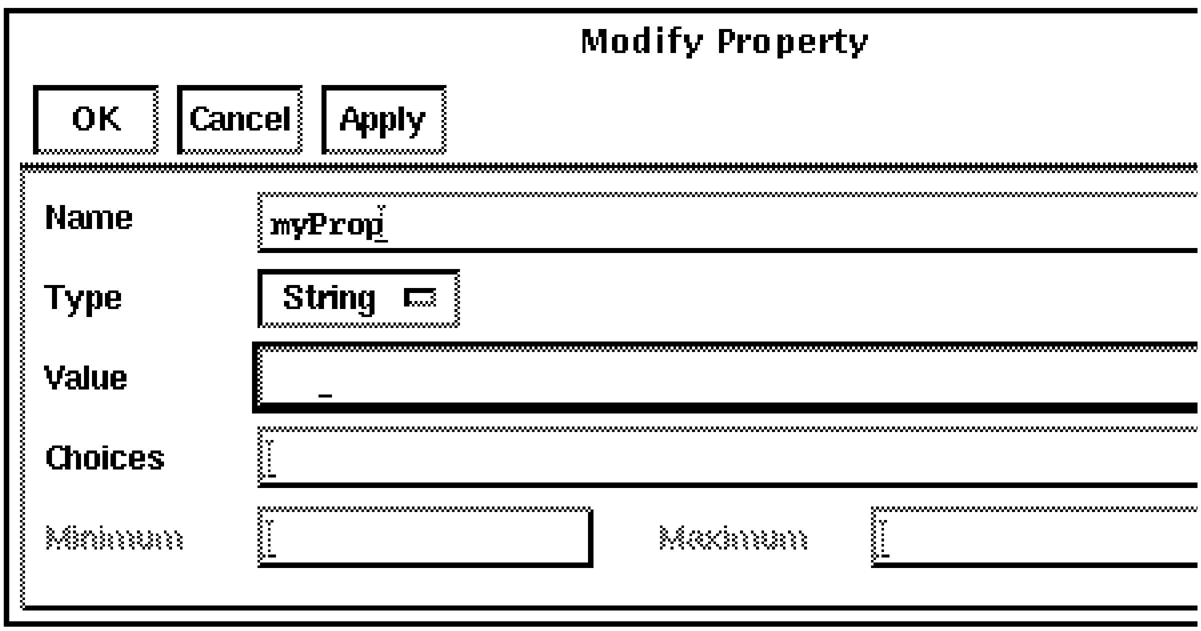
6. Click *OK*.

Your new property appears at the bottom of the Edit Properties form.

About the Modify Property Form

The Modify Property form lets you change a user-created property.

Note: You can modify and delete only those properties you added using the Add Property form.



Name displays the name of the property.

Type controls the type of value for the property. This can be *Int* (integer), *Float* (floating-point number), *String* (any text, displayed in a cyclic field), *Time* (date and time), *Boolean* (on or off), or *ILList* (list of IL values).

Value sets the default value for this property. This line can show a text entry field, a Boolean button, or a cyclic field, depending on the *Type* setting.

Choices sets the possible values that appear in the cyclic field for a string property. One of the values listed here must match the default shown in *Value*.

Minimum and **Maximum** set the minimum and maximum values for integer, floating-point, and time properties. The values appear in parentheses after the property name in the Edit Cellview Properties or the Edit Properties form.

Changing a Property Using the Modify Property Form

To change a user-created property,

1. Click *Modify* in the Edit Cellview Properties form or the Edit Properties form.

   The Modify Property form appears.

2. Do one of the following:

   - If you want to change a cellview property, choose *Design – Properties* [`Shift-q`].
   - If you want to change an object property, select the object and then choose *Edit – Properties* [q].

     The Edit Properties form or the Edit Cellview Properties form appears.

3. Click *Properties*.

   The properties of the object or cellview appear.

4. Click on the property label you want to change.
5. Click *Modify*.

   The Modify Property form appears.

6. In the Modify Property form, make your changes.
7. Click *OK*.

Deleting a Property

To delete a property,

1. Do one of the following:

   - If you want to delete a cellview property, choose *Design – Properties* [`Shift-q`].
   - If you want to delete an object property, select the object and then choose *Edit – Properties* [q].

     The Edit Properties form or the Edit Cellview Properties form appears.

2. Click *Properties*.

   The properties of the object or cellview appear.

3. Click on the property label you want to delete.

4. Click *Delete*.

   The property is deleted from the form.

Editing Object Attributes

The characteristics that define an object are known as the object's attributes. All design objects have default attributes, such as color.

To change the attributes of an object,

1. Choose *Edit – Properties* [q].
2. Select the object.

   The Edit Properties form for that object appears.
3. In the Edit Properties form, set the *Attributes* button on.
4. Change any of the attributes.
5. Click *Apply*.

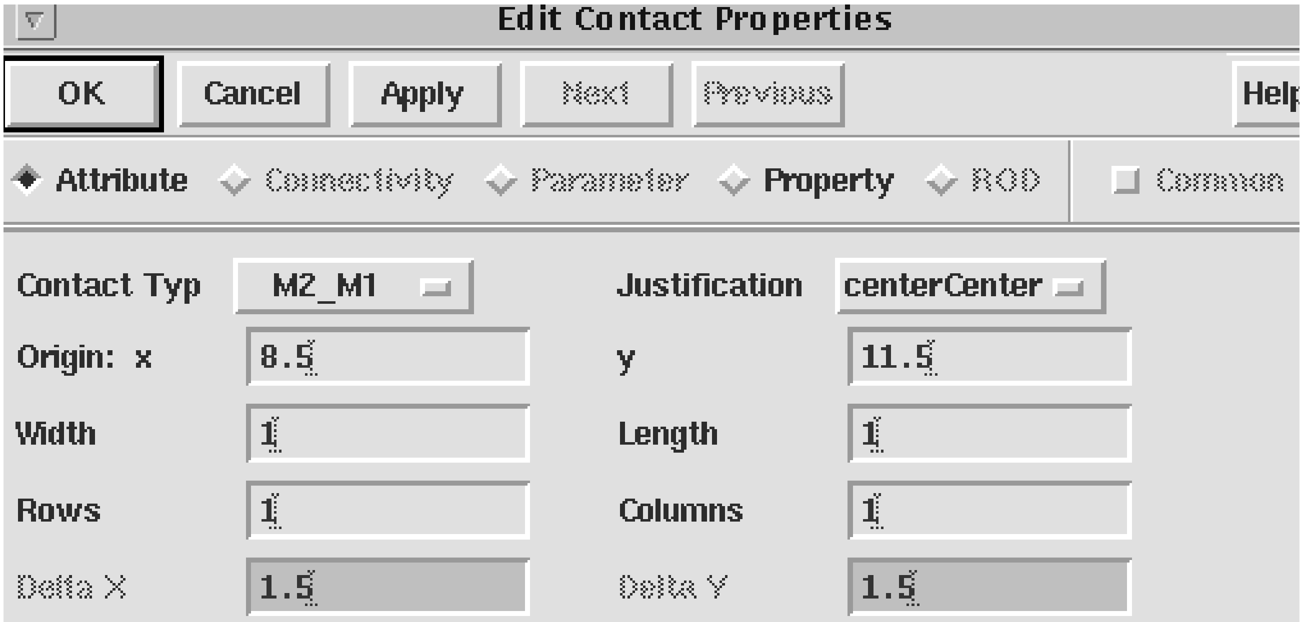
Viewing the Attributes of a Contact

The Edit Contact Properties form lets you view and change the attributes and add, modify, or delete properties of a contact.

To view the attributes of a contact,

1. Select a contact.
2. Choose *Edit – Properties* [q].

The Edit Contact Properties form appears.



Contact Type selects the contact from the technology file and controls the layers on which the contact is drawn.

Justification sets the origin of a single contact or of a contact array.

Origin: x and **y** set the X and Y coordinates of the contact origin.

Width and **Length** set the width and length of the contact or via cut, in user units (typically microns).

Rows and **Columns** set the number of rows or the number of columns of contact cuts in a contact array.

Delta X sets the horizontal distance between the center points of the contacts when *Rows* is set to greater than 1.

Delta Y sets the vertical distance between the center points of the contacts when *Columns* is set to greater than 1.

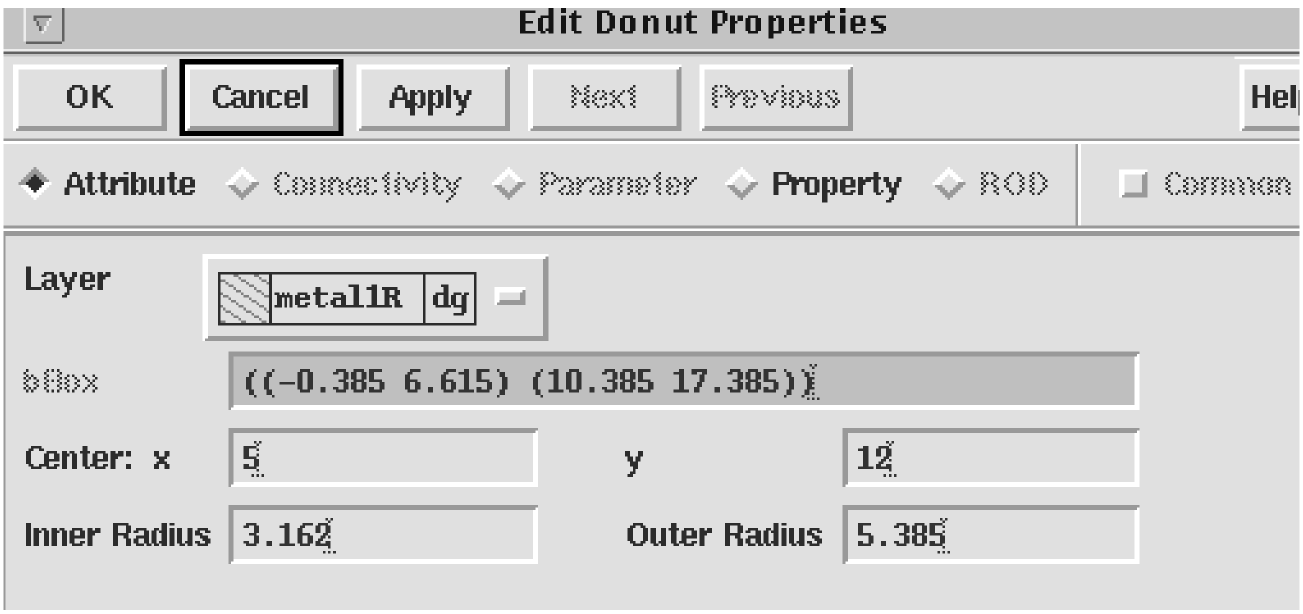
Viewing the Attributes of a Donut

The Edit Donut Properties form lets you view and change the attributes and add, modify, or delete properties of a donut.

To view the attributes of a donut,

1. Select a donut.
2. Choose *Edit – Properties* [q].

   The Edit Donut Properties form appears.

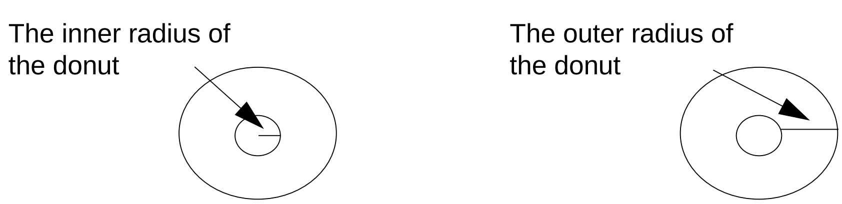


Layer sets the layer on which the donut is drawn.

bBox displays the coordinates of the box surrounding the donut.

Center: x and **y** set the X and Y coordinates of the center of the donut.

Inner Radius and **Outer Radius** set the inner and outer radius of the donut.



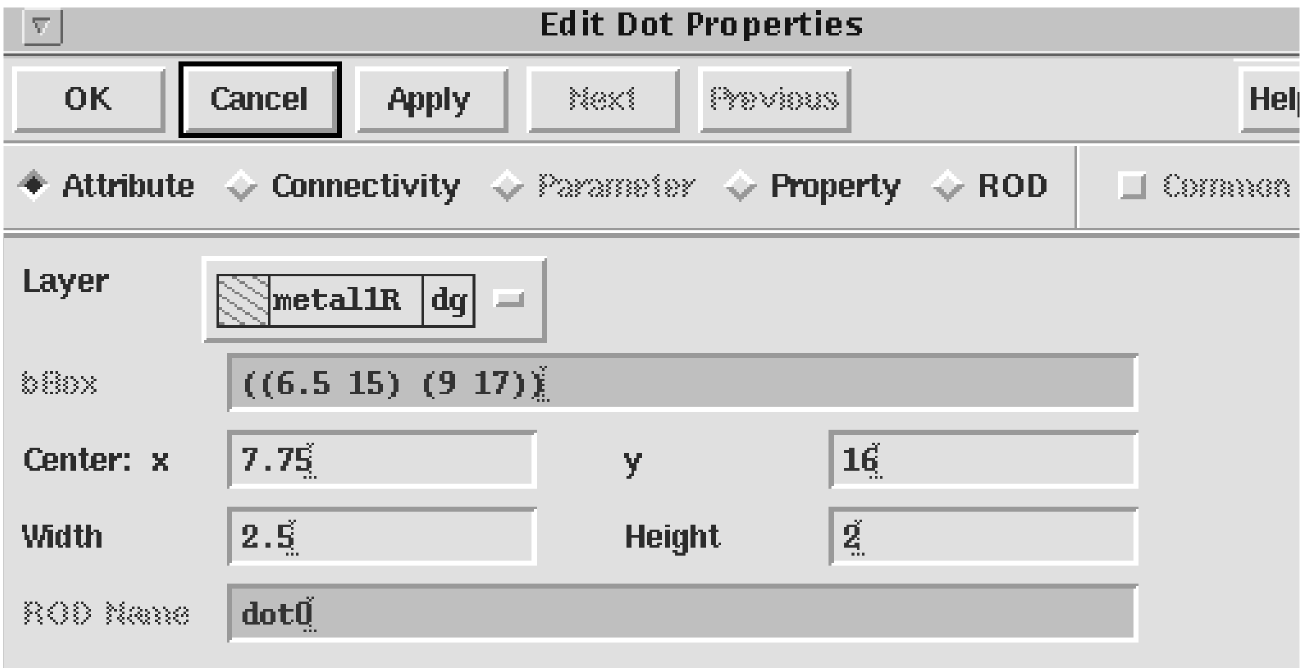
Viewing the Attributes of a Dot

The Edit Dot Properties form lets you view and change the attributes and add, modify, or delete properties of a dot. If the dot is a ROD object, you can edit the ROD properties by choosing the ROD category.

To view the attributes of a dot,

1. Select a dot pin.
2. Choose *Edit – Properties* [q].

   The Edit Dot Properties form appears.



Layer sets the layer of the dot.

bBox displays the coordinates of the bounding box of the dot.

Center: x and **y** set the X and Y coordinates of the dot origin.

Width sets the width of the dot in user units (usually microns).

Height sets the height of the dot in user units (usually microns).

ROD Name displays the name of the dot if it is a ROD object.

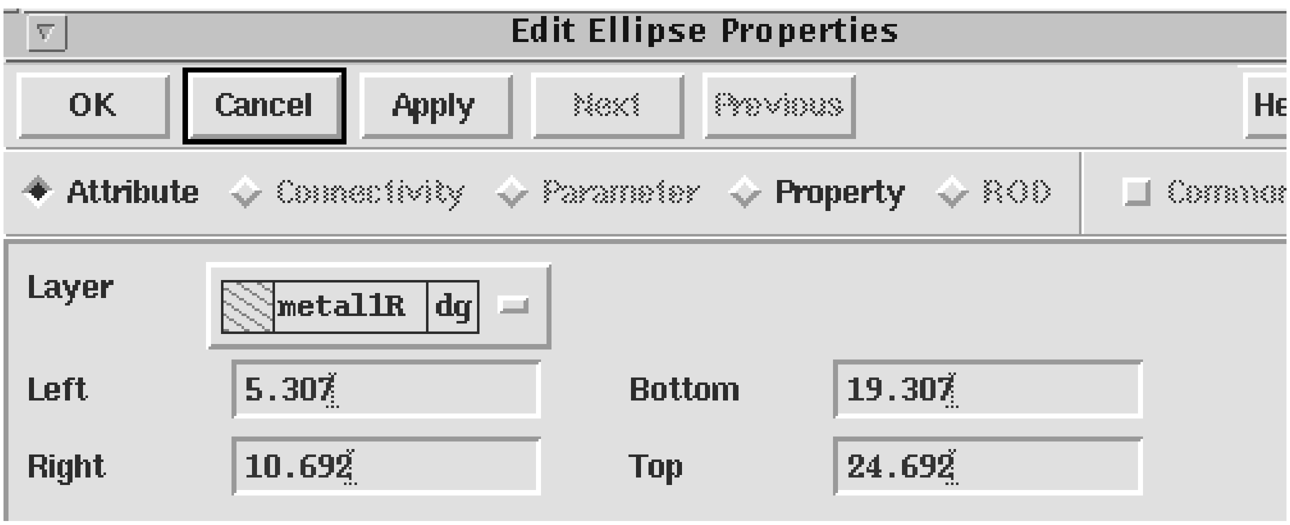
Viewing the Attributes of an Ellipse or Circle

The Edit Ellipse Properties form lets you view and change the attributes and add, modify, or delete properties of an ellipse or circle.

To view the attributes of an ellipse or circle,

1. Select an ellipse or circle.
2. Choose *Edit – Properties* [q].

   The Edit Ellipse Properties form appears.



Layer sets the layer of the ellipse or circle.

Left, **Right**, **Bottom**, and **Top** set the coordinates of the box that defines the ellipse or circle.

Viewing the Attributes of an Instance

The Edit Instance Properties form lets you view and change the attributes and add, modify, or delete properties of a instance. If the instance contains ROD objects, you can edit the ROD properties. by choosing the ROD category.

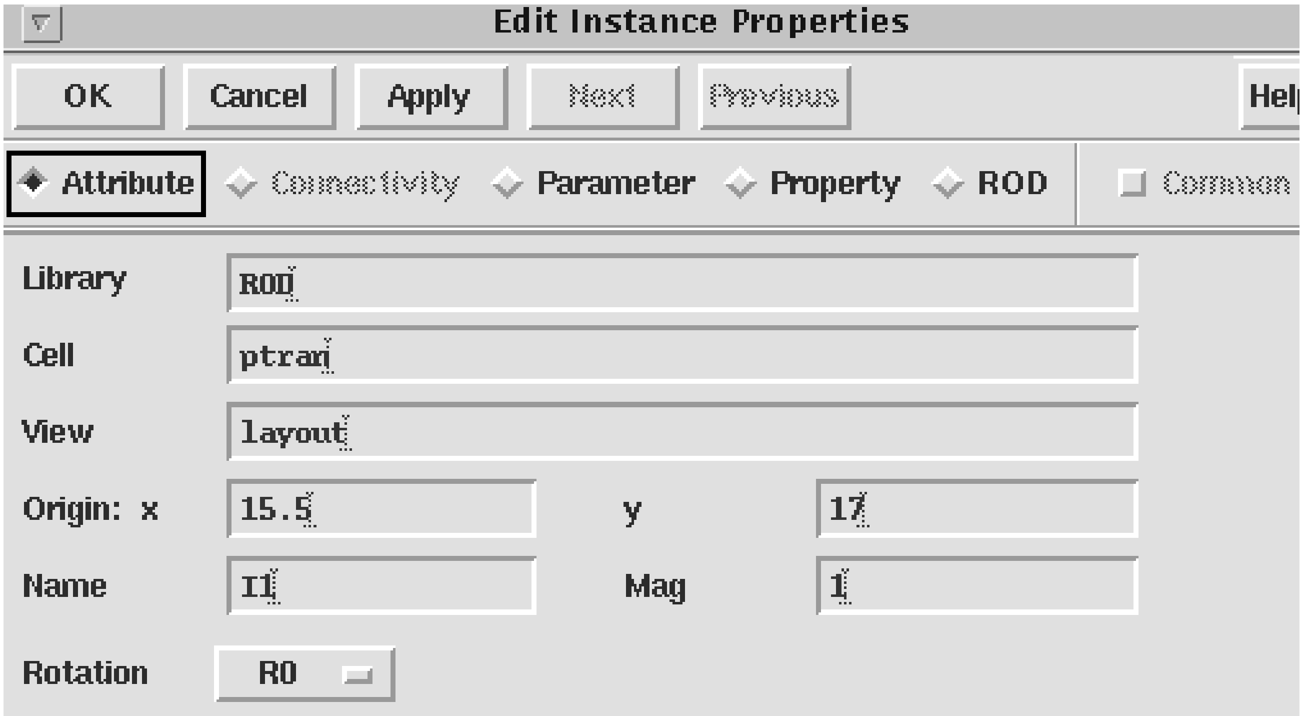
When you route your design to the Virtuoso® custom router, the changes you make in the Edit Instance Properties form are reflected in your custom router design.

To view the attributes of an instance,

1. Select an instance.

2. Choose *Edit – Properties* [q].

The Edit Instance Properties form appears.



Library, **Cell**, and **View** set the library, cell, and view names of the master cell for this instance.

Origin: x and **y** set the X and Y coordinates of the origin of the instance.

Name sets the name assigned to this instance. The layout editor automatically assigns instance names that begin with the letter `I`, followed by a number.

Mag(nification) enlarges or reduces the size of the cell instance.

Rotation sets whether the instance is rotated or mirrored.

R0 = no rotation
R90 = rotated 90 degrees
R180 = rotated 180 degrees
R270 = rotated 270 degrees
MY = mirrored over the Y axis
MYR90 = mirrored over Y, rotated 90 degrees
MX = mirrored over the X axis
MXR90 = mirrored over X, rotated 90 degrees

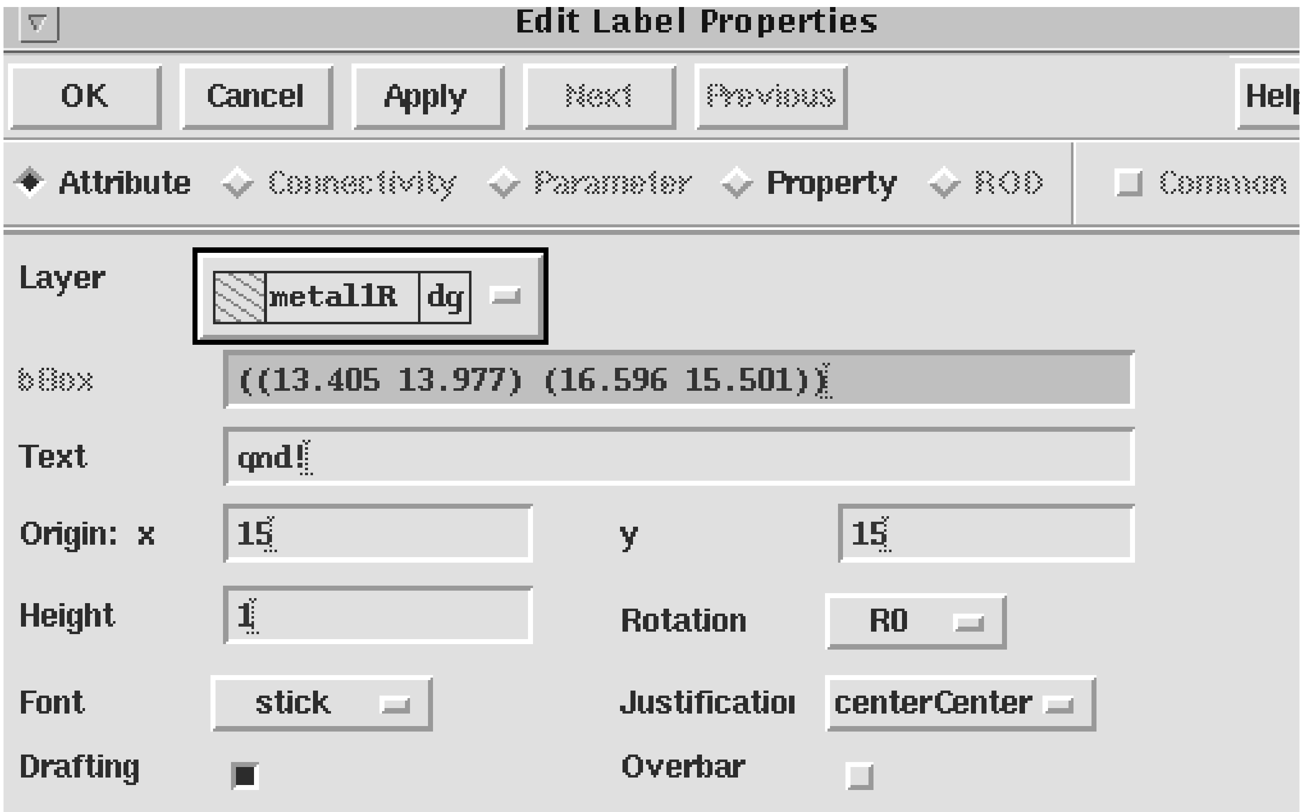
Viewing the Attributes of a Label

The Edit Label Properties form lets you view and change the attributes and add, modify, or delete properties of a label.

To view the attributes of a label,

1. Select a label.
2. Choose *Edit – Properties* [q].

   The Edit Label Properties form appears.



Layer sets the layer on which the label is drawn.

bBox displays the coordinates of the box surrounding the label.

Text sets the text that appears in the label.

Origin: x and **y** set the X and Y coordinates of the label origin.

Height sets the height of the label, in user units (usually microns).

Rotation sets whether the label is rotated or mirrored.

R0 = no rotation
R90 = rotated 90 degrees
R180= rotated 180 degrees
R270 = rotated 270 degrees
MY = mirrored over the Y axis
MYR90 = mirrored over Y, rotated 90 degrees
MX = mirrored over the X axis
MXR90 = mirrored over X, rotated 90 degrees

Font sets the text style of the label.

Justification sets the location of the label origin. The origin appears as a small square on the label when you place or select it.

Drafting prevents the label from being rotated more than 90 degrees.

Overbar is a display option that determines how text strings containing underscore characters are displayed in a layout window.

When the overbar is disabled (default), the software displays underscore characters (_) as part of the text string. When the overbar is enabled, the software interprets underscore characters (_) in the text string name as toggle switches that control where overbars begin and end. Overbars appear above the text string, as shown in the examples.

Text String	Appears in Layout Window As
_abcde	$\overline{\text{abcde}}$
ab_cde	$\text{ab}\overline{\text{cde}}$
_abc_de	$\overline{\text{abc}}\text{de}$

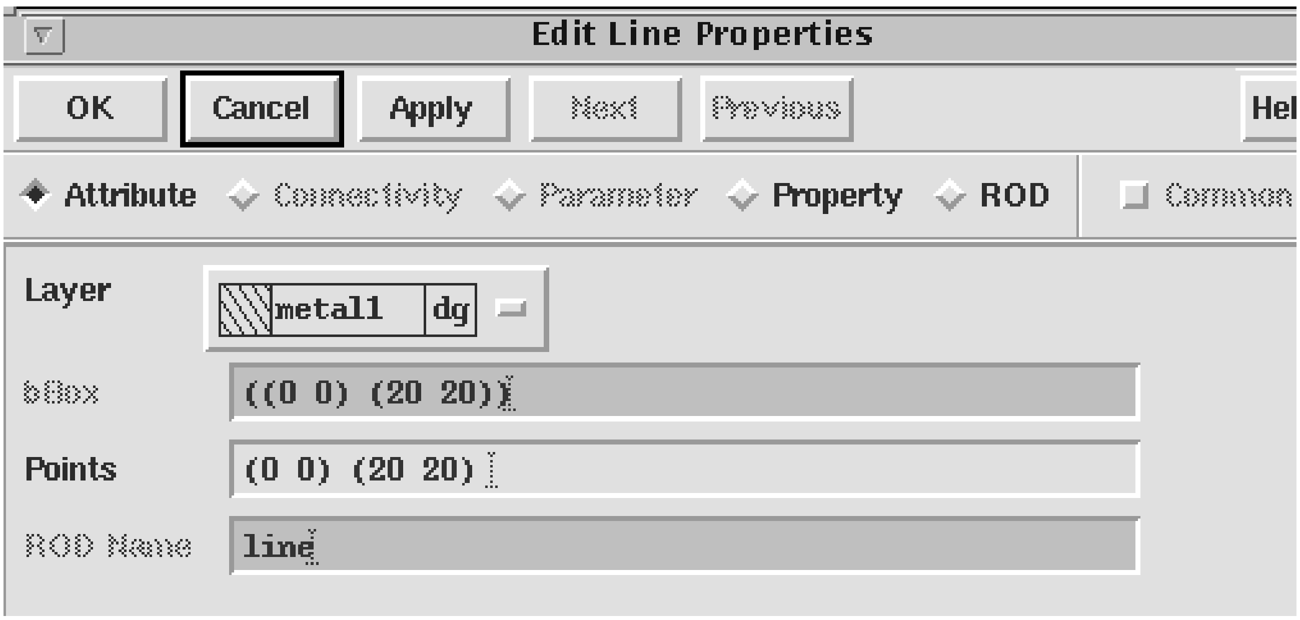
Viewing the Attributes of a Line

The Edit Line Properties form lets you view and change the attributes and add, modify, or delete properties of a line. If the line is a ROD object, you can edit the ROD properties by choosing the ROD category.

To view the attributes of a label,

1. Select a line.
2. Choose *Edit – Properties* [q].

   The Edit Line Properties form appears.



Layer sets the layer of the line.

bBox sets the coordinates of the box surrounding the line.

Points sets the coordinates of each point of the line.

ROD Name displays the name of the line if it is a ROD object.

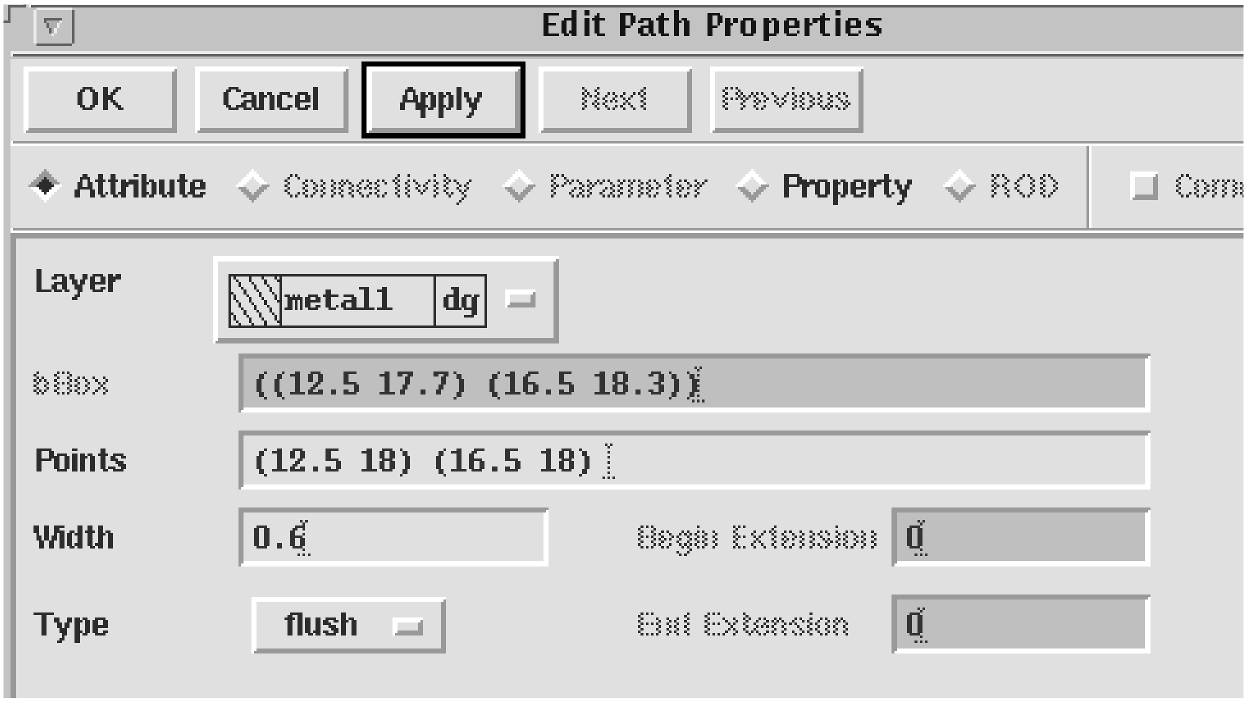
Viewing the Attributes of a Path

The Edit Path Properties form lets you view and change the attributes and add, modify, or delete properties of a path. If the path is a ROD object, you can edit the ROD properties by choosing the ROD category.

To view the attributes of a path,

1. Select a path.
2. Choose *Edit – Properties* [q].

The Edit Path Properties form appears.



Layer sets the layer of the path.

bBox sets the coordinates of the box surrounding the path.

Points sets the coordinates of each point of the path.

Width specifies the path width in user units (usually microns).

Type controls how the path ends are drawn.

Begin Extension and **End Extension** set the length of the beginning and ending extension in user units if *Type* is set to *variable*.

ROD Name appears if the path is a ROD object.

Subpart appears if the path is a multipart path and has subparts.

Viewing the Attributes of a ROD Multipart Path

A multipart path (MPP) is a single ROD object consisting of one or more parts at level zero in the hierarchy on the same or on different layers. A multipart path consists of a single *master* *path* and one or more *subparts*. The master path is an ordinary path; however, it is the defining part of a multipart path; all subparts are based on the master path.

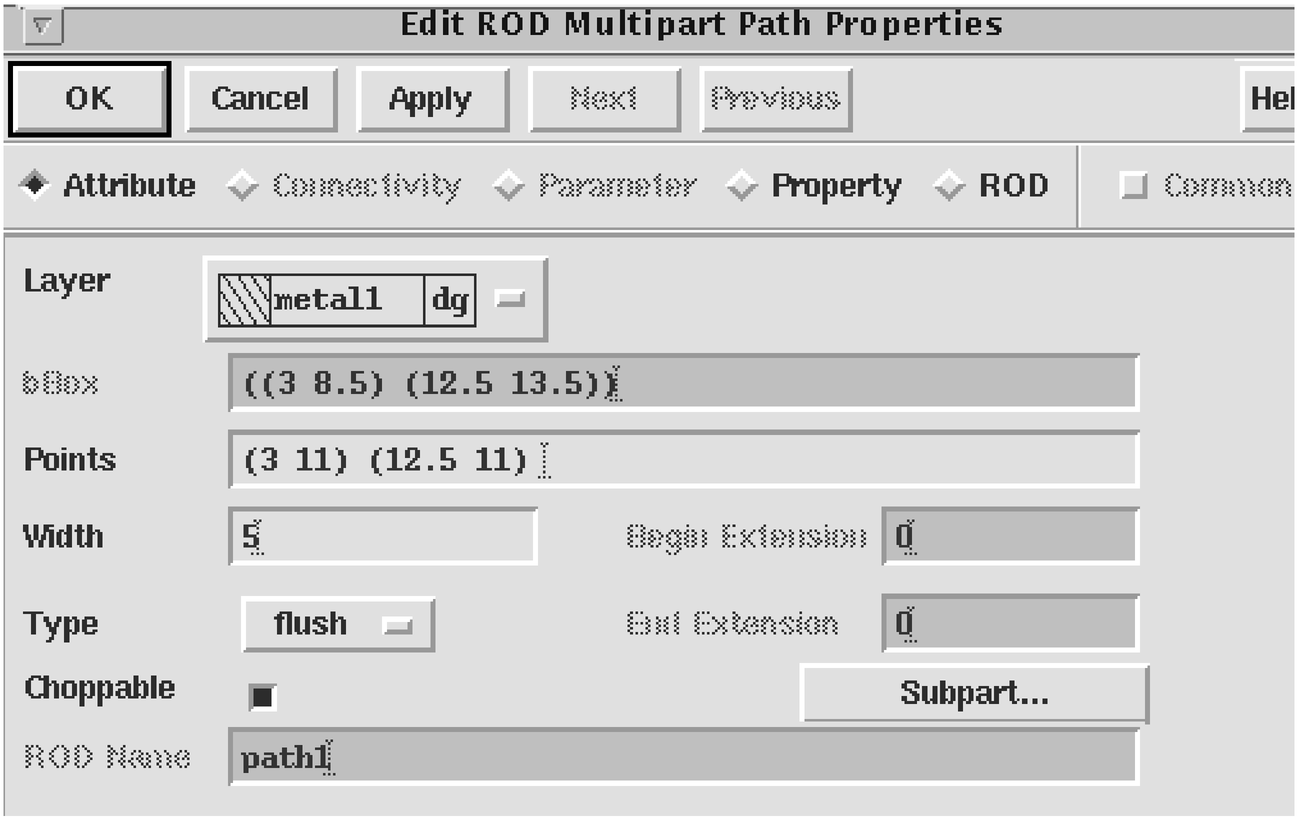
The Edit ROD Multipart Path Properties form lets you

- View and change the attributes of the master path
- View and change the attributes of the subparts, except for the layer
- Change the ROD information for the entire multipart path including
  - System handle
  - User handle
  - Alignment
- View and change the ROD properties

To view the attributes of an MPP,

1. Select an MPP.
2. Choose *Edit – Properties* [q].

The Edit ROD Multipart Path Properties form appears.



Master Path Fields

Layer sets the layer of the master path.

bBox displays the coordinates of the box surrounding the path.

Points sets the coordinates of each point of the master path.

Width specifies the path width in user units (usually microns).

Type controls how the path ends are drawn.

Begin Extension and **End Extension** set the length of the beginning and ending extension in user units if *Type* is set to *variable*.

Choppable indicates whether or not a ROD path can be chopped. The value must be `t` or `nil`. When a path has subparts and the master path is choppable, all subpaths and sets of subrectangles must be choppable also. When a path has subparts and the master path is not choppable, each subpath and/or set of subrectangles can be choppable or not. The default is choppable.

ROD Name displays the name of the MPP.

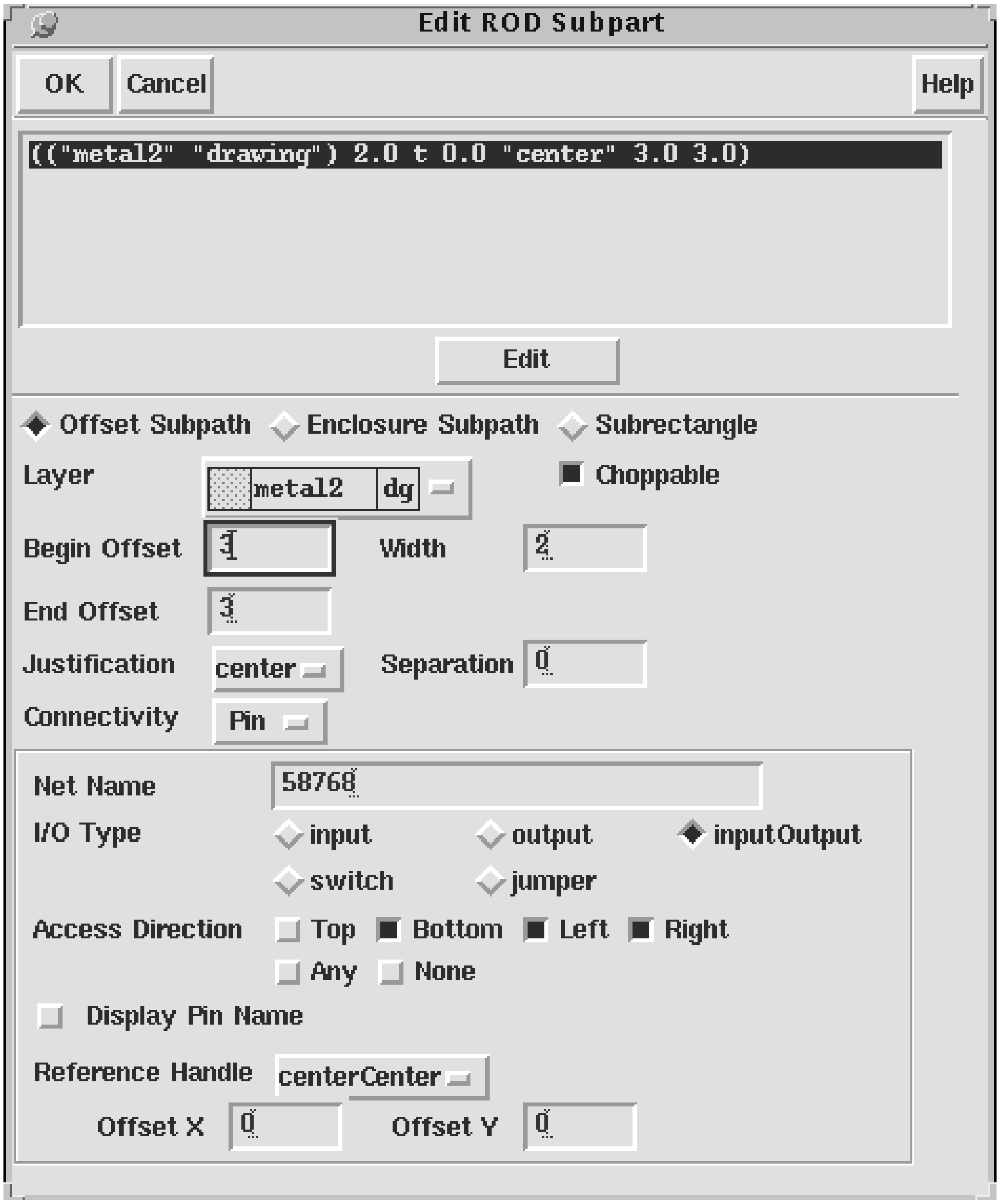
Subpart opens one of three subpart forms:

- Offset Subpath
- Enclosure Subpath
- Subrectangle

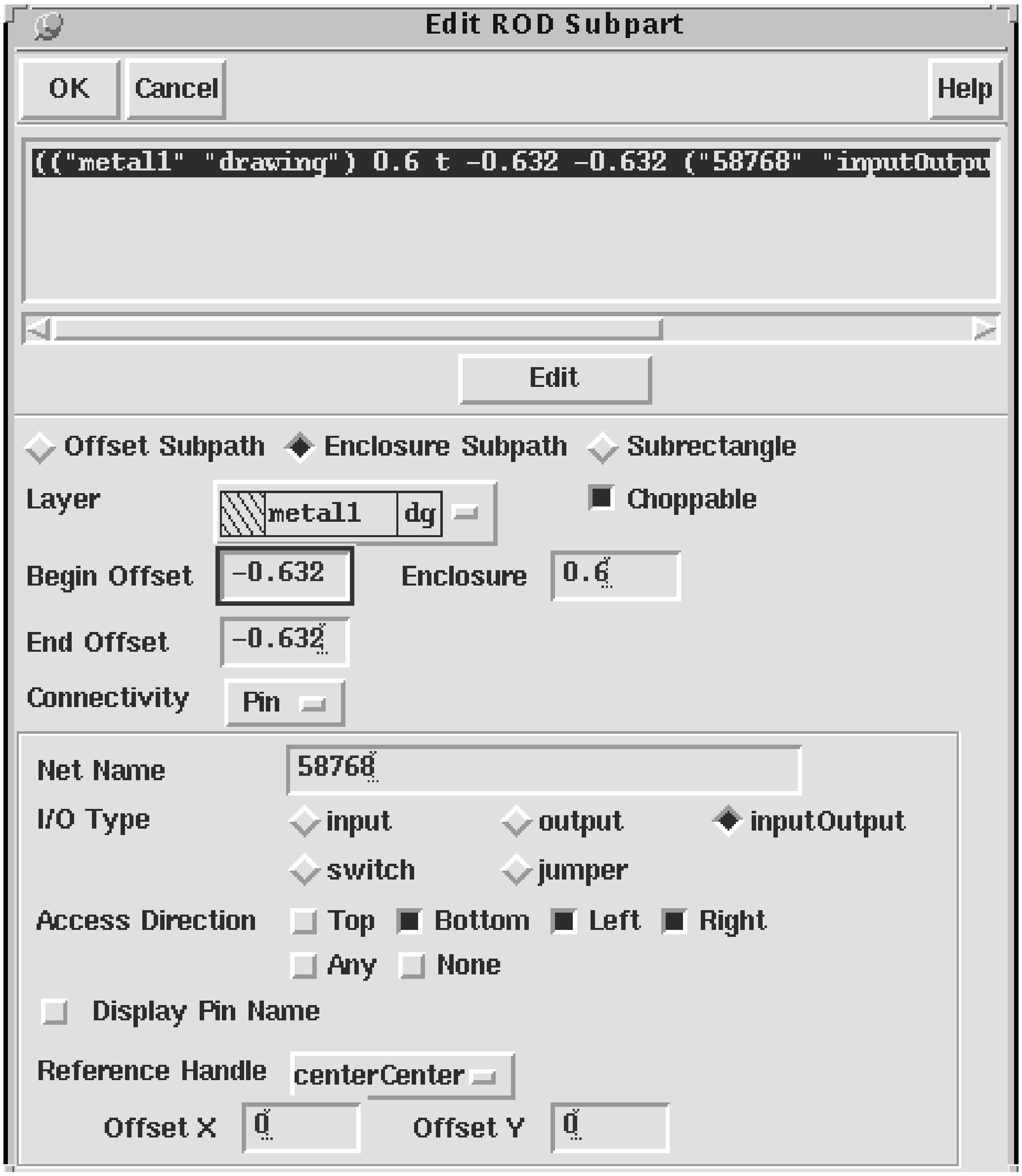
For complete form field descriptions, see the ROD Subpart Form section of the *Virtuoso Layout Accelerator User Guide*.

For an example of how to edit an MPP, see "Editing Multipart Paths" on page 385.

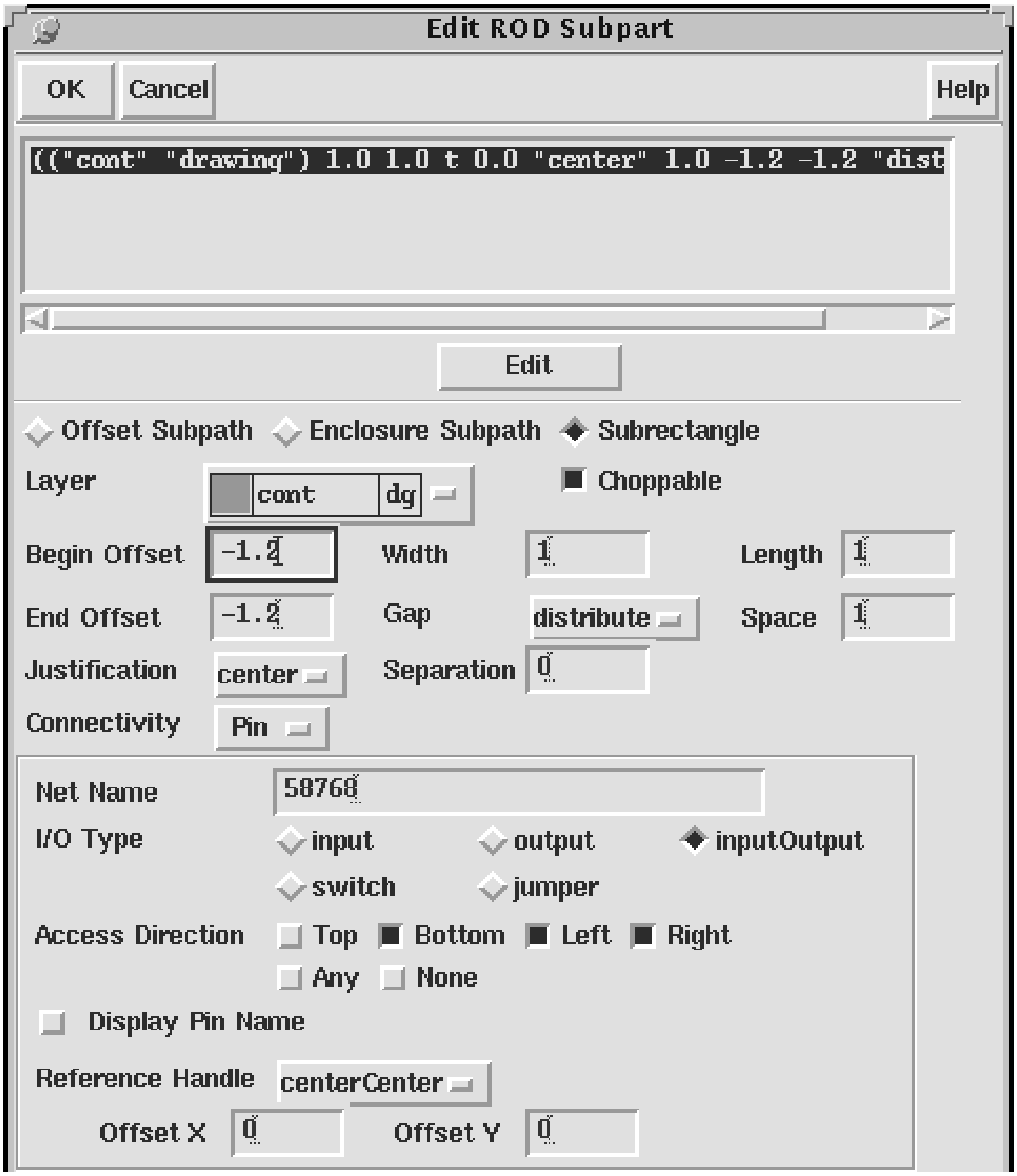
Offset Subpath Fields



Enclosure Subpath Fields



Subrectangle Fields



Limitations to Editing Multipart Paths

There are a few attributes that cannot be edited using the Edit ROD Multipart Path Properties form.

- Adding or deleting subparts of an existing MPP

To add or delete subparts of an existing MPP, you must use the *Create – Multipart Path* command. For example, if you want to delete all the subrectangles from an existing MPP, you would have to create a new MPP minus the subrectangles, rather than deleting the rectangles in the Edit ROD Multipart Path Properties form.

However, you can create a new MPP quickly by loading a template containing all the data about the particular MPP. After loading the template, you can delete the subrectangle information. For complete information about ROD MPPs and templates, see the *Creating Multipart Paths* section of the *Virtuoso Layout Accelerator User Guide*.

- Changing subpart layers

  The *Layer* field displays the current layer for the selected subpart, but it is not editable. To change a subpart layer, you must create a new multipart path containing the subpart layer you want.

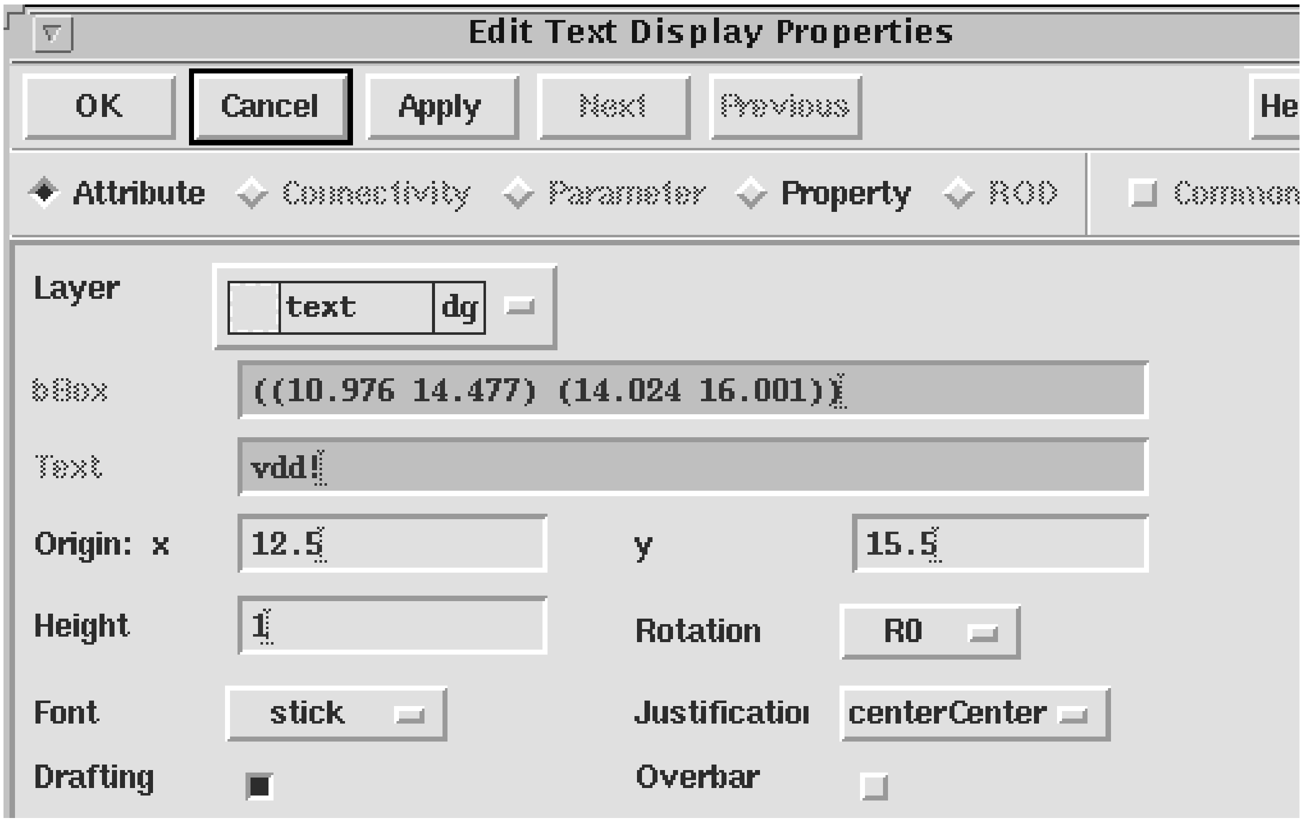
Viewing the Attributes of a Pin Name or Other Text Display

The Edit Text Display Properties form lets you view and change the attributes and add, modify, or delete properties of a text display, such as a pin name display.

To view the attributes of text display,

1. Select a text display.
2. Choose *Edit – Properties* [q].

The Edit Text Display Properties form appears.



Layer sets the layer of the text display.

bBox displays the coordinates of the box surrounding the text display.

Text displays the text of the text display for the associated pin and is not editable.

Origin: x and **y** set the X and Y coordinates of the text display origin.

Height sets the height of the text display in user units (usually microns).

Rotation sets whether the text display is rotated or mirrored.

R0 = no rotation
R90 = rotated 90 degrees
R180 = rotated 180 degrees
R270 = rotated 270 degrees
MY = mirrored over the Y axis
MYR90 = mirrored over Y, rotated 90 degrees
MX = mirrored over the X axis
MXR90 = mirrored over X, rotated 90 degrees

Font sets the text style of the text display.

Justification sets the location of the origin of the text display origin. The origin appears as a small square on the pin name when you place or select it.

Drafting prevents the text from being rotated more than 90 degrees.

Overbar is a display option that determines how text strings containing underscore characters are displayed in a layout window.

When the overbar is disabled (default), the software displays underscore characters (_) as part of the text string. When the overbar is enabled, the software interprets underscore characters (_) in the text string name as toggle switches that control where overbars begin and end. Overbars appear above the text string, as shown in the examples.

Text String	Appears in Layout Window As
_abcde	$\overline{\text{abcde}}$
ab_cde	ab$\overline{\text{cde}}$
_abc_de	$\overline{\text{abc}}$de

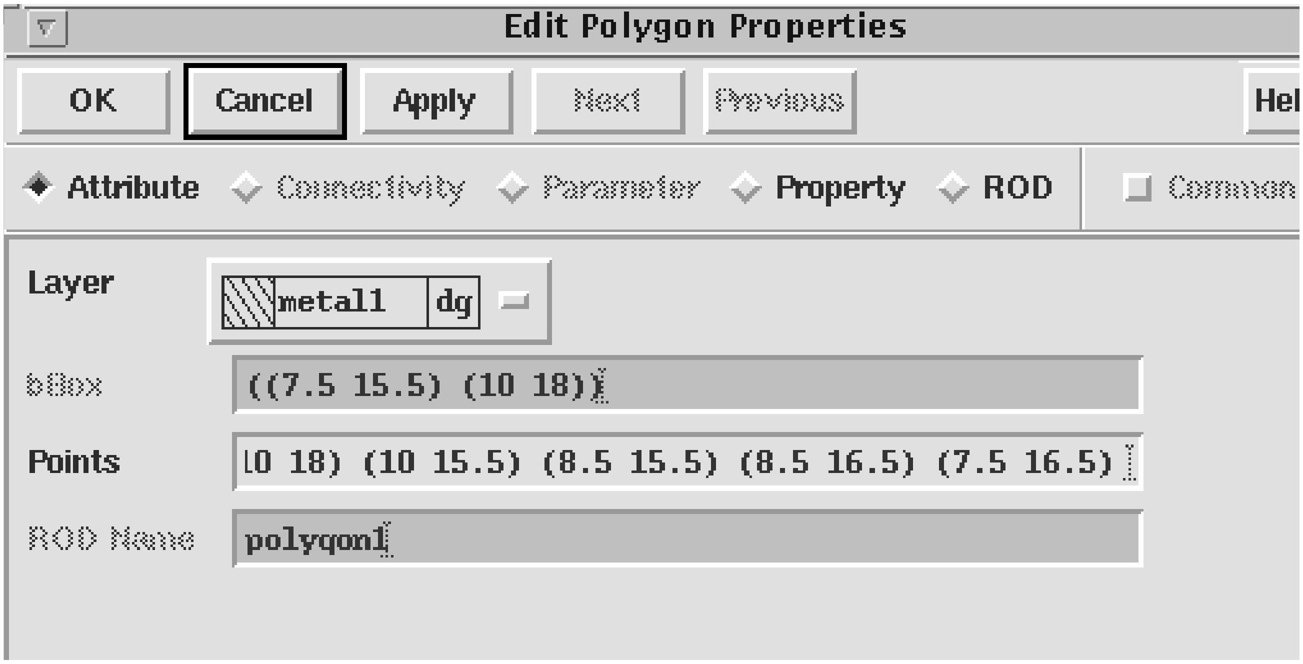
Viewing the Attributes of a Polygon or Polygon Pin

The Edit Polygon Properties form lets you view and change the attributes and add, modify, or delete properties of a polygon or polygon pin. If the polygon or polygon pin is a ROD object, you can edit the ROD properties by choosing the ROD category.

To view the attributes of a polygon,

1. Select a polygon or polygon pin.
2. Choose *Edit – Properties* [q].

The Edit Polygon Properties form appears.



Layer sets the layer of the polygon.

bBox sets the coordinates of the box surrounding the polygon.

Points sets the coordinates of each point of the polygon.

ROD Name displays the name of the polygon if it is a ROD object.

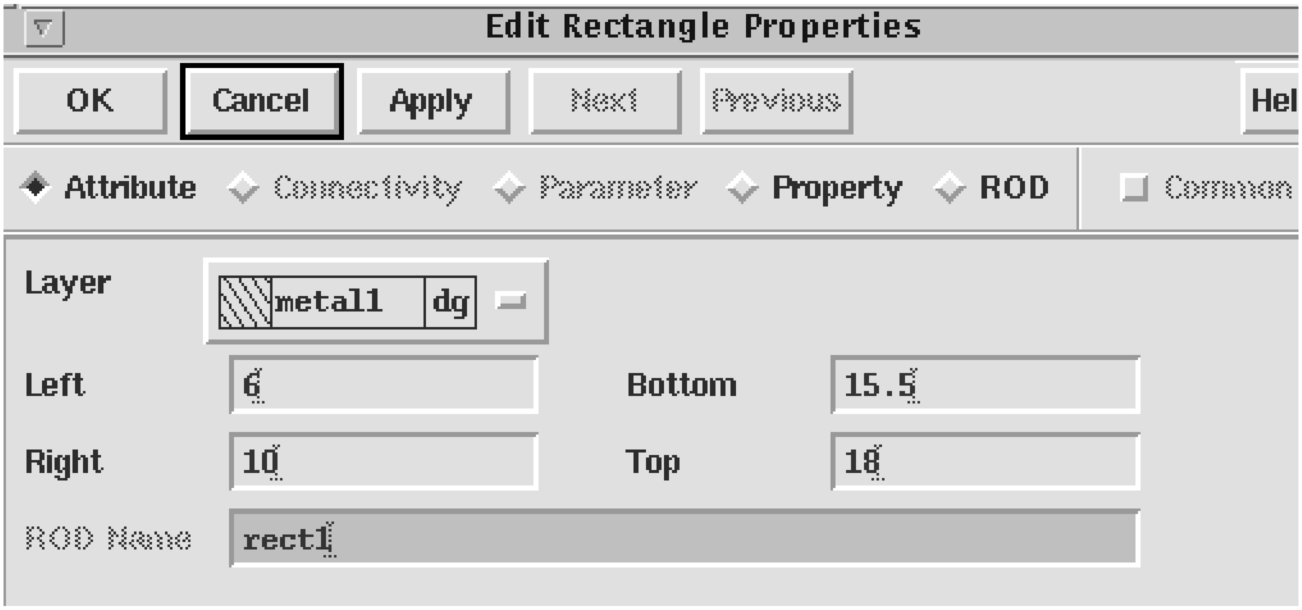
Viewing the Attributes of a Rectangle or a Rectangle Pin

The Edit Rectangle Properties form lets you view and change the attributes and add, modify, or delete properties of a rectangle or rectangle pin. If the rectangle or rectangle pin is a ROD object, you can edit the ROD properties by choosing the ROD category.

To view the attributes of a rectangle,

1. Select a rectangle or rectangle pin.
2. Choose *Edit – Properties* [q].

The Edit Donut Properties form appears.



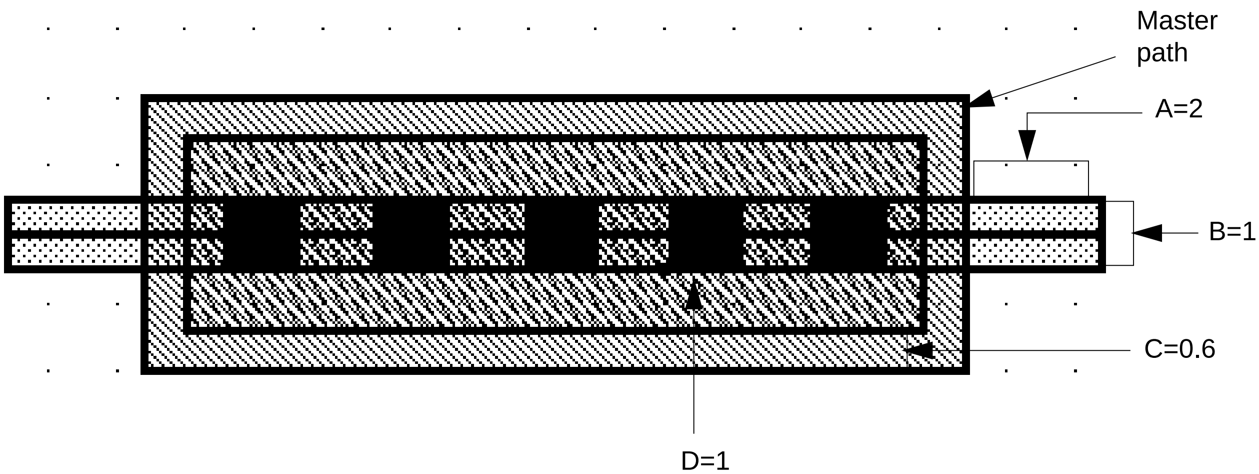
Layer sets the layer of the rectangle.

Left, **Right**, **Bottom**, and **Top** set the coordinates of the rectangle.

ROD Name displays the name of the rectangle if it is a ROD object.

Editing Multipart Paths

You can change the attributes of the MPPs master path and subparts through the Edit ROD Multipart Path Properties form. In this procedure, you change this data in the sample MPP shown here.



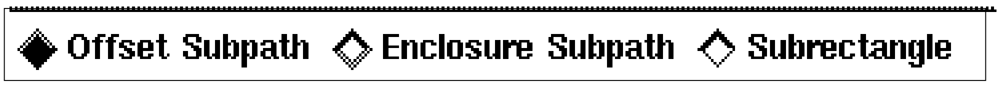
Label	Description	Original data	Edited data
A	Offset subpath begin and end offset	2	1
B	Offset subpath width	1	3
C	Enclosure subpath	0.6	-1
D	Subrectangle length and width	1	2

Edit the subparts of the sample by doing the following:

1. Select the MPP.
2. Choose *Edit – Properties*.

   The Edit ROD Multipart Path Properties form appears.
3. Click *Subpart*.

The Edit ROD Subpart form appears. The three subpart choices appear in the middle of the form.



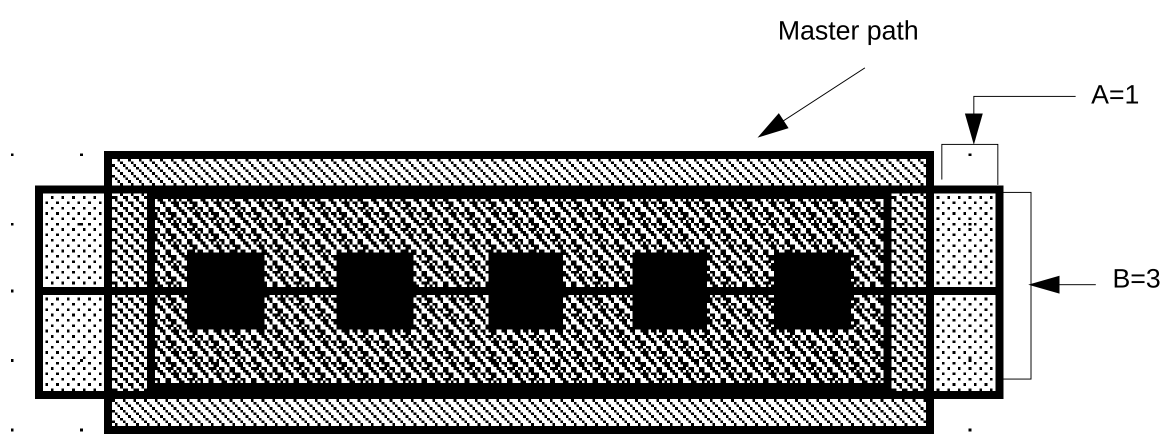
4. Click *Offset Subpath*.

   The form changes and the offset subpath fields appear.

5. Select the subpart in the scroll window at the top of the form.

6. Change *Begin Offset* and *End Offset* from 2 to 1.
7. Change *Width* from 1 to 3.
8. Click *Edit* to apply the change.
9. Click *OK* to view the change.

   The MPP contains the edited offset subpath.



10. Click *Subpart* in the Edit ROD Multipart Path Properties form.

    The Edit ROD Subpart form appears.

11. Click *Enclosure Subpath.*

The form changes and the enclosure subpath fields appear.

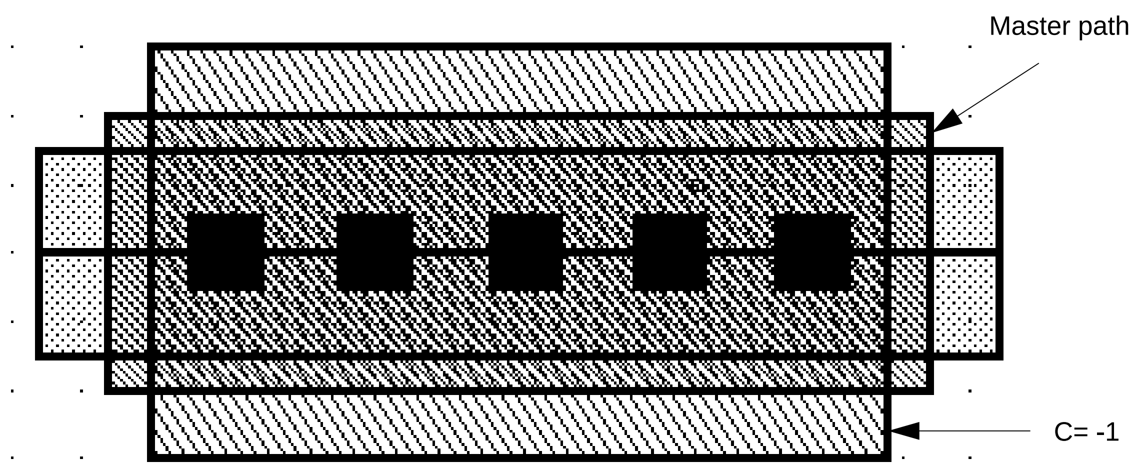
12. Select the subpart in the scroll window at the top of the form.

13. Change *Enclosure* from `0.6` to `-1`.

14. Click *Edit* to apply the change.

15. Click *OK* to view the change.

The MPP contains the edited enclosure subpath.



16. Click *Subpart* in the Edit ROD Multipart Path Properties form.

The Edit ROD Subpart form appears.

17. Click *Subrectangle*.

The form changes and the subrectangle fields appear.

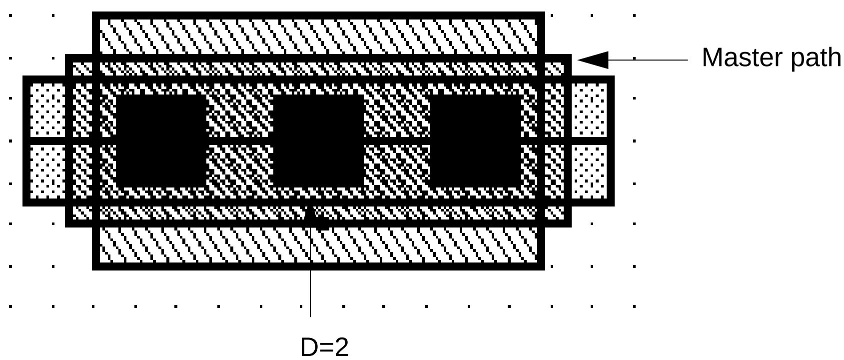
18. Select the subpart in the scroll window at the top of the form.

19. Change the *Width* and *Length* from `1` to `2`.

20. Click *Edit* to apply the change.

21. Click *OK* to view the change.

The MPP contains the edited subrectangles.



22. Click *OK* to close the Edit ROD Multipart Path Properties form. The three subparts of the MPP have been edited.

For complete information about ROD MPPs, see “Creating Multipart Paths” in the *Virtuoso Layout Accelerator User Guide*.

Using Net Expressions and Inherited Connections

Inherited Connections

Inherited connections is an extension to the connectivity model that allows you to create global signals and override their names for selected branches of the design hierarchy. This flexibility allows you to use

- Multiple power supplies in a design
- Overridable substrate connections
- Parameterized power and ground symbols

To learn about using inherited connections and net expressions with various Cadence tools in the design flow, refer to the *Inherited Connections Flow Guide*.

To learn about connectivity and naming conventions for inherited connections and how to add and edit net expressions in a schematic or symbol cellview, refer to the *Virtuoso Schematic Composer User Guide*.

Net Expressions in the Virtuoso Layout Editor Environment

A net expression is a special property placed on a net or terminal to define its connectivity. A net expression consists of a property name and a default net name. You use a net expression to override the connection made by a signal or terminal. This allows you to use, for example, multiple power supplies in your design. The signal is redefined due to the value assigned to it. Redefining the signal eliminates the problem of global nets being merged into a single, electrically-equivalent signal, which occurs when the signal traverses the design hierarchy.

You can create, edit, and delete net expressions for pins by changing the net expression information in the *Connectivity* category of the Edit Properties form.

Once the net expression information is applied to a terminal, the net expression is displayed in the text labels of pins. The following must be true for the net expression to display:

- The pin is created with *Create Label* set on in the Create Pin form
- *Net Expressions* is set on in the Display Options form

  If *Net Expressions* is set off, the terminal name is displayed.

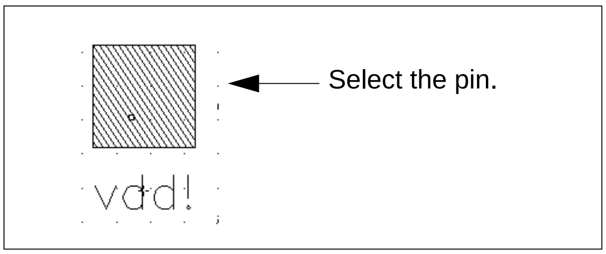
Note: Net expressions are not interpreted by the display code in the Virtuoso layout editor. The net expressions are displayed as is, but other tools, such as the Layout Versus Schematic

(LVS) program, use and interpret net expressions. The net expressions on layout terminals must match those found in schematic views.

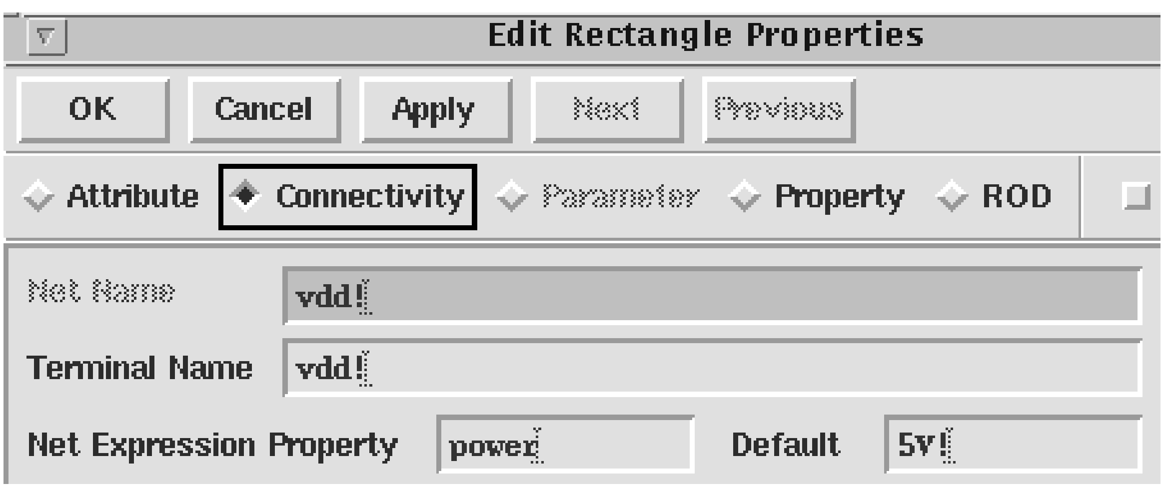
Creating a Net Expression

To create a net expression,

1. Select the pin whose terminal you want to assign a net expression to.

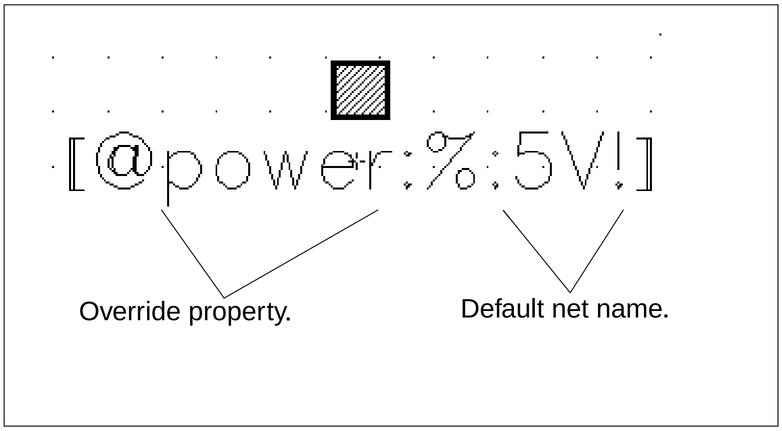


2. Open the Edit Properties form.
3. Choose the *Connectivity* radio button.
4. Type the net expression override property name in the *Net Expression Property* field.
5. Type the default net name in the *Default* field. This is used when no override property is defined in the hierarchy above this point in the schematic view. If you do not enter a default net name, the terminal name is used.



6. Click *Apply*.

The pin label display changes to show the net expression property preceded by the @ symbol and the default net name preceded by the % symbol.



Note: The *Net Expressions* field in the Display Options form must be set on for the net expressions to display in your cellview.

Editing Net Expressions

To edit net expression information for pins,

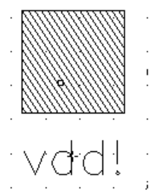
1. Select the pin whose terminal's net expression you want to change.
2. Open the Edit Properties form.
3. Choose the *Connectivity* radio button.
4. Change the information in the *Net Expression Property* and *Default* fields.

(Clearing the *Net Expression Property* field in the Edit Properties form deletes the net expression from your design.)

If several pins in your design have the same terminal name, you need update only one pin and the rest will update automatically. You see the edited text labels after you use *Window – Redraw*.

Viewing Instances Containing Net Expressions

When you view an instance containing a pin whose terminal has a net expression, the label display is the terminal name, not the net expression. To see the net expression in a hierarchical design, you must descend into the cellview containing the net expression to see the full text display of the net expression.



Pin name display when placed in an instance. Net expression does not display.

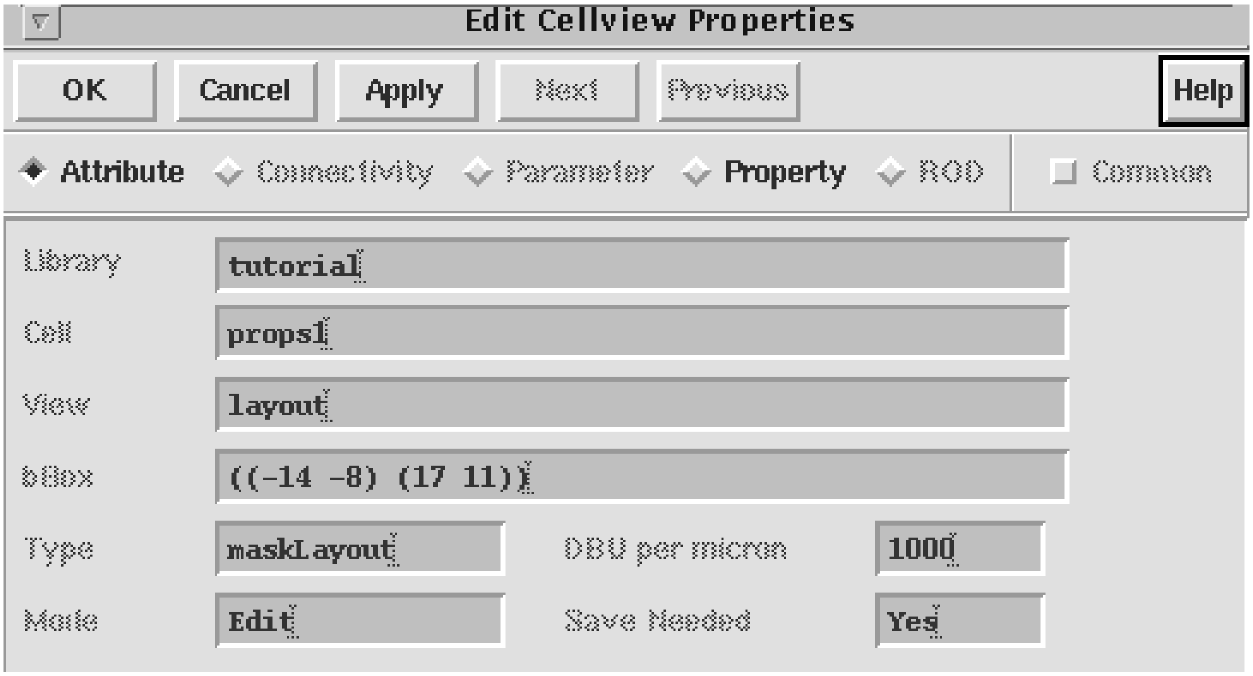
Using the Edit Cellview Properties Form

The *Design Properties* command lets you edit the attributes or properties (defaults) of this cellview. The form can display either attributes or properties.

About the Edit Cellview Properties Form

To open the Edit Cellview Properties form,

➤ Choose *Design – Properties* [`Shift-q`].



Attribute represents the characteristics of the object. None of these fields can be changed.

Library displays the name of the library containing this cellview.

Cell displays the name of this cell.

View displays the name of this view.

bBox displays the coordinates of the invisible box surrounding the design in the cellview.

Type displays the view type that corresponds to the view name for this cellview. View types are defined in your technology file.

DBU per micron displays the number of database units (DBU) per user unit (usually microns). Database units and user units are defined in the `.cdsenv` file or by the library manager.

Mode shows whether this cellview is opened in edit mode (`Edit`) or read-only mode (`Read`).

Save Needed shows whether you made changes to this cellview that need to be saved to disk.

Property shows the properties for this cellview. Most properties correspond to settings in the Display Options form. Other Cadence applications such as DRC or Extract can also add properties.

Add opens the Open Property form, which lets you add a new cellview property.

Delete deletes a selected property created with *Add*.

Modify opens the Modify Property form, which lets you change the definition for any property created with *Add*.

Common is not used when you edit cellview properties. This button is used when you edit properties of objects inside this cellview with the *Edit – Properties* command.

Next and **Previous** (at the top of the form) are not used when you edit cellview properties but when you edit properties of objects inside this cellview with the *Edit – Properties* command.

Connectivity is not used when you edit cellview properties. This button is used when you edit properties of objects inside this cellview with the *Edit – Properties* command.

Parameter is not used when you edit cellview properties. This button is used when you edit properties of objects inside this cellview with the *Edit – Properties* command.

ROD is not used when you edit cellview properties. This button is used when you edit properties of ROD objects inside this cellview with the *Edit – Properties* command.

Viewing and Editing Cellview Properties

As you set and change the display options, your changes can be saved as properties of the current cellview.

To view and change these properties for any cellview,

1. Choose *Design – Properties* [`Shift-q`].

   The Edit Cellview Properties form appears.

2. Set the *Property* button on.

   The settings you saved to this cellview using the Display Options form appear. Any properties attached to the cellview by other applications, such as the last time you ran DRC on the cellview, appear.

3. To edit any of the properties, type or select a new value.
4. Click *OK*.

11

Searching for Objects

This chapter contains these topics:

Using Search

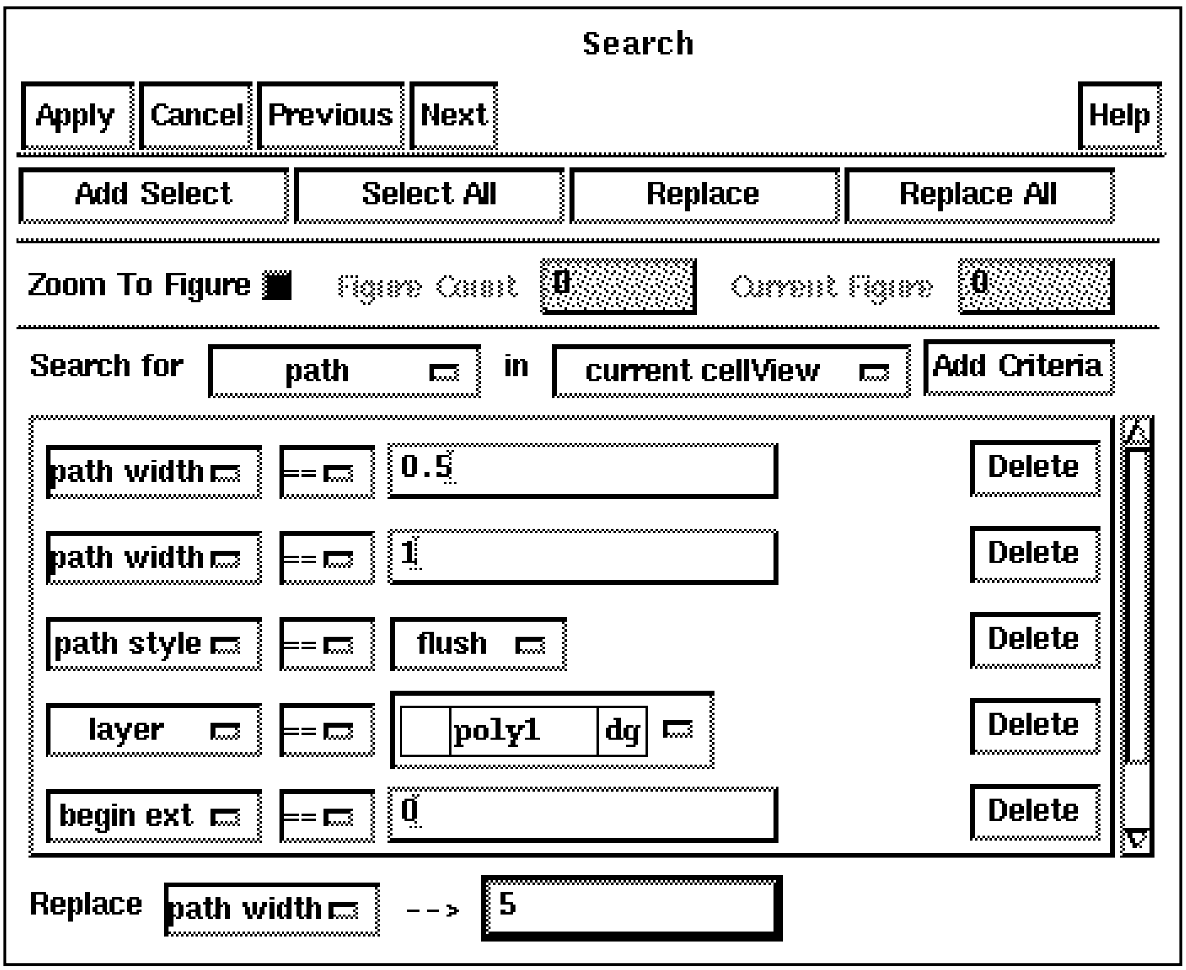
The *Search* command lets you search for objects with specific attributes or property values.

Note: Search is not supported for ROD objects.

About the Search Form

To open the Search form,

➤ Choose *Edit – Search* [`Shift-s`].



Apply searches for objects matching the settings in the Search form.

Cancel closes the Search form.

Previous highlights the previous object in the search group.

Next highlights the next object in the search group.

Add Select selects the current object in the search group (highlighted in beige).

Select All selects all objects found by *Search*.

Replace edits or replaces the current object (highlighted in beige) with the settings shown in the *Replace* field.

Replace All replaces all objects found by *Search* with the settings shown in the *Replace* field.

Zoom To Figure centers the cellview around the current object in the search group (highlighted in beige) when you click *Apply*, *Previous*, or *Next*.

Figure Count, after you click *Apply*, shows the number of objects found by *Search*.

Current Figure shows which object in the search group is highlighted as you click *Previous* or *Next*.

Search for sets the type of object to search for.

In sets where *Search* looks for objects.

> **area** searches an area of this cellview. You are prompted to create the area.
>
> **current cellView** searches anywhere in this cellview.
>
> **current to bottom** searches anywhere in this cellview and in any cell instances inside this cellview.
>
> **current to stop level** searches anywhere in this cellview and in any cell instances whose details are displayed.

Add Criteria adds a line of fields to the form that let you make your search more specific.

> **cell name** lets you search for specific attributes, instance, library, or view name, or properties for this type of object.
>
> **==** or **!=** lets you search for objects equal to (==) or not equal to (!=) the criteria.
>
> **value field** sets the value of the attribute or property to use in the search.

Delete deletes this line of criteria from the form.

Replace lets you replace the object's *Search* finds with a new object, attribute value, or property value. To replace parameter values for a parameterized cell, set *Replace* to *property* and type the parameter name.

Searching for and Selecting Objects

You can use the *Search* command to search for and select specific objects. You might want to do this if

- You have a large design

  For example, you want to find a particular polygon that is buried in a dense design.

- You want to use a Boolean expression to let you select a group of objects

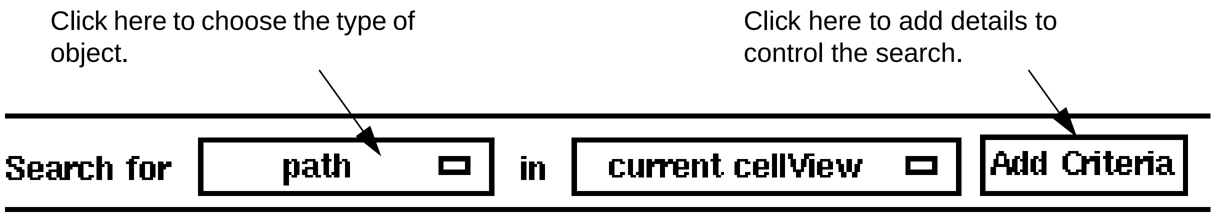
  For example, if you want to select all paths on your *metal2* layer that have a width of 1 micron.

To search for objects,

1. Choose *Edit – Search* [`Shift-s`].

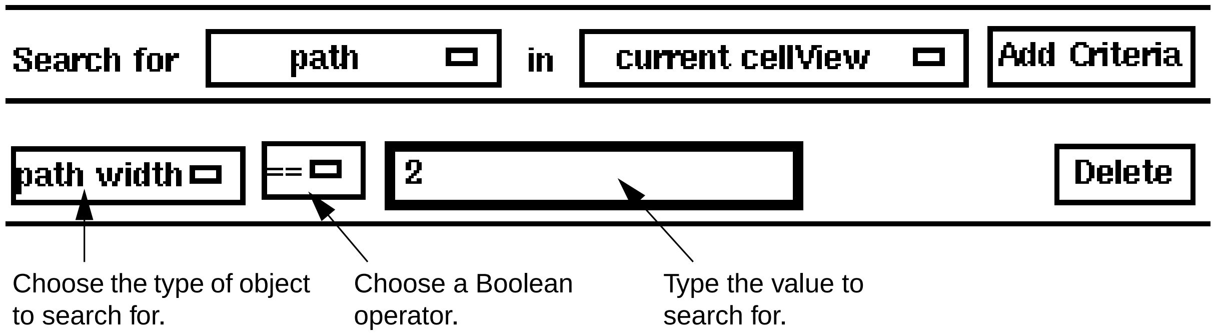
   The Search form appears.

2. In the *Search for* cyclic field choose the type of object you want to find.
3. Click *Add Criteria* to add Boolean expressions, which control the search (search criteria).



4. Set the search criteria by doing the following:
   - Click on the first field to choose the type of value to search for.
   - Click on the Boolean operator field to choose the operator you want: equal (==) or not equal (!=).
   - Type or choose the value you want.

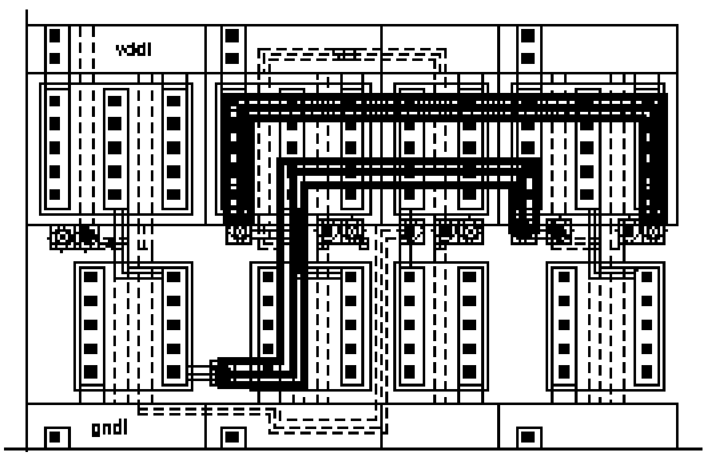
The value field can be a text entry or a cyclic field, depending on the type of value.



- Click on *Delete* at the end of the row to delete a row of search criteria.
- Click *Add Criteria* again to add more search criteria.

5. Click *Apply*.

The Virtuoso® layout editor highlights all of the objects it finds and puts them in a search group. The first or current object in the group is highlighted in a different color.



Objects that meet the search criteria are highlighted.

6. Click *Previous* or *Next* to search through the group.

7. Click *Add Select* to select the current object in the search group.

8. Click *Select All* to select all of the highlighted objects.

9. When you are finished selecting objects, click *Cancel*.

Searching for Shapes and Pins on One Layer

To select all shapes and pins on one layer, without selecting pins on other layers,

1. Enable the *Pin* button on the Layer Selection Window (LSW).
2. Choose *Edit – Search* [`Shift-s`].

   The Search form appears.
3. In the *Search for* cyclic field choose *any shape* and *current cellView*.
4. Click *Add Criteria* to add search criteria.
5. Set the search criteria by doing the following:
   - Click on the first field and choose *layer*.
   - Click on the Boolean operator and choose equal (==).
   - Choose the layer you want to select.
6. Click *Apply*.

   The Virtuoso layout editor highlights all shapes on the layer.
7. Click *Select All* to select all shapes on the layer, including pins.
8. Click *Cancel* to end the search.

   All shapes on the selected layer, including pins, will be selected.

Searching for and Replacing Instances

Searching for Instances

To search for instances in the current cellview,

1. Choose *Edit – Search* [`Shift-s`].

   The Search form appears.
2. In the *Search for* cyclic field choose *instance*.
3. Click *Add Criteria* to add search criteria.
4. Click on the first field in the criterion line to search for the instance by one of the following:

- *cell name*, the name of the master cell
- *instance name*, the name assigned to the instance as you placed it

5. Type the cell or instance name.

   You can use the wildcard characters * and . when typing a cell or instance name. The character * means any number of characters starting in this position, and the character . means any single character in this position.

6. Click *Add Criteria* to further define the search.
7. Click *Apply*.

   The Virtuoso layout editor highlights all the instances in the cellview that match the criteria you entered.

Searching for Objects in Instances

To search for objects or instances inside other instances,

1. Choose *Edit – Search* [`Shift-s`].

   The Search form appears.

2. In the *Search for* cyclic field choose the type of object you want to find.
3. Turn on *in* and choose either
   - *current to bottom*, to search the cell you are editing and all cells in it
   - *current to stop level*, to search from the cell you are editing to the bottom hierarchy level displayed
4. Click *Add Criteria* to further define the search.
5. Click *Apply*.

   During the search, instances are highlighted as they are found.

Note: If you use Edit In Place to edit a cell while the Search form is open, you must cancel *Search* and then restart it. Otherwise, *Search* does not know that you have changed your current edit level.

Replacing Instances

To replace instances of one master cell with instances of a different master cell,

1. Choose *Edit – Search* [`Shift-s`].

   The Search form appears.

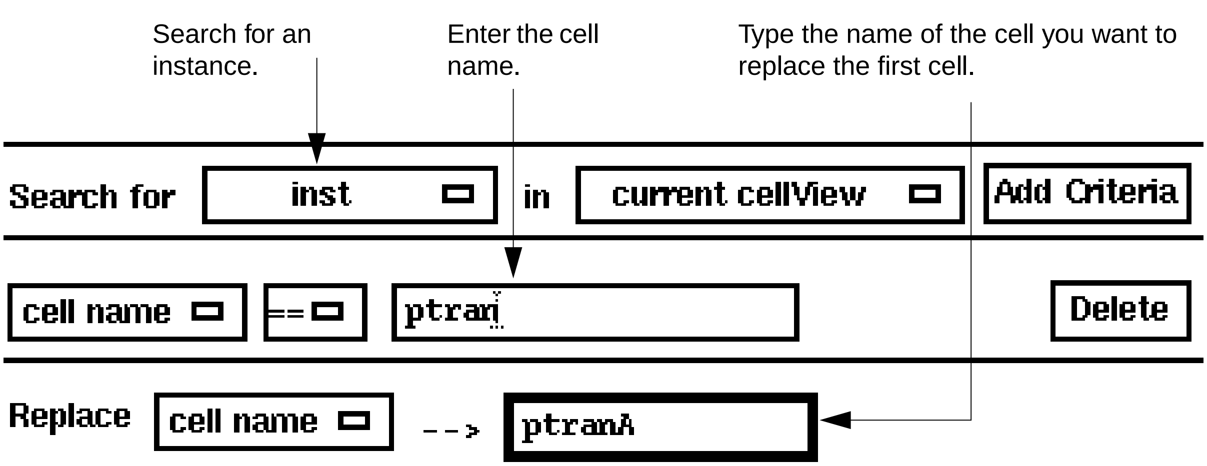
2. In the *Search for* cyclic field choose *instance*.
3. Click *Add Criteria* to add search criteria.
4. Click on the first field in the criterion line to search for the instance by one of the following:
   - *cell name*, the name of the master cell
   - *instance name*, the name assigned to the instance as you placed it
5. Type the cell or instance name.

   You can use the wildcard characters * and . when typing a cell or instance name. The character * means any number of characters starting in this position, and the character . means any single character in this position.

6. Click *Add Criteria* to further define the search.
7. Click *Apply*.

   The Virtuoso layout editor highlights all the instances in the cellview that match the criteria you entered.

8. In the *Replace* cyclic field choose *cell name*.
9. Type the name of the new master cell.

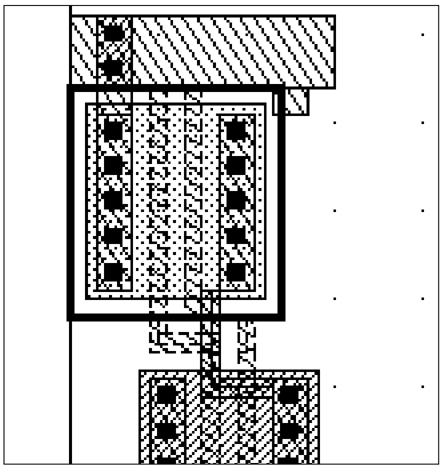


10. Click *Apply*.

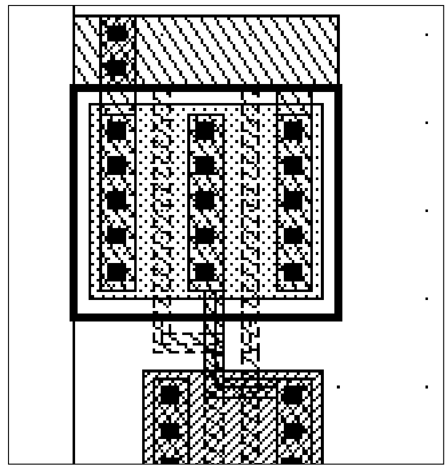
    The Virtuoso layout editor highlights all the instances of the master cell.

11. Click *Previous* or *Next* until the instance you want is highlighted. Then replace only one instance of the cell.

12. Click the *Replace* button to replace the top cell in the search stack, or click the *Replace All* button to replace all instances of the highlighted cell.



A selected instance of ptran



After using *Search* to replace ptran with ptranA

12

Using Connectivity

This chapter contains these topics:

See also *Implementing External Connections (Must Connects) in Parameterized Cells*. This is an application note on SourceLink `http://sourcelink.cadence.com`, the online customer support system, containing information and examples about how to create Cadence® SKILL language-based parameterized cells (pcells) that use external connections.

Overview of Pin Connectivity

The Layout window Connectivity menu *Define Pins* commands let you tell the router at what level to connect a pin or group of pins in a net, internally or externally.

The *Define Pins – Must Connect* command lets you tell the router to connect a selected pin or group of pins in a net externally (at a higher level). When you connect pins externally, you do not need to route between the pins inside the cell.

The *Define Pins – Strongly Connected* command lets you tell the router to connect the selected pin or group of pins in a net internally (within the device). When you connect pins internally (strong connect), you do not need an external connection.

The *Define Pins – Weakly Connected* command lets you tell the router to connect the selected pin or group of pins in a net externally. Weakly connected pins have limited external connection to avoid specific internal connections (typically ones with high-resistance paths).

The *Define Pins – Pseudo Parallel Connect* command lets you tell the router that the selected instance terminals (terminals within an instance) do not need to be connected, even though they are on the same net.

A device is any object with pins, which includes transistors, resistors, or capacitors. By default, pins are connected internally.

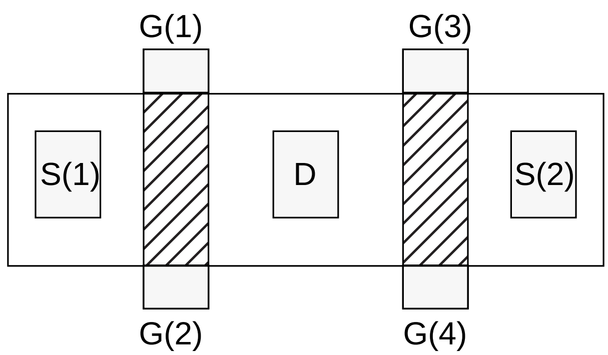
The ability to define connection types for pins can be useful if

- You are a library developer designing a library device for use in a block or circuit design and need to provide layout designers with maximum flexibility by defining some connections internally and leaving other connections to be completed at the next level of design
- You are a layout designer using a device designed by others in a layout and need to specify, later in the layout cycle, which pins to connect externally

The following examples show how the ability to define pins as must connects, strongly connected, weakly connected, or pseudo parallel connects can make the design process more efficient for both these job functions.

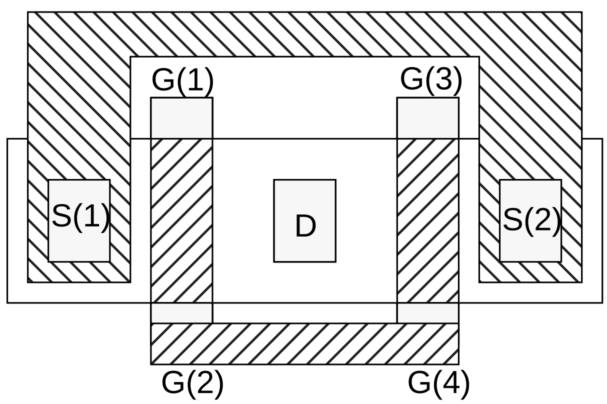
For Library Developers

If you are working as a library developer, a typical design might be a multifingered FET like this.

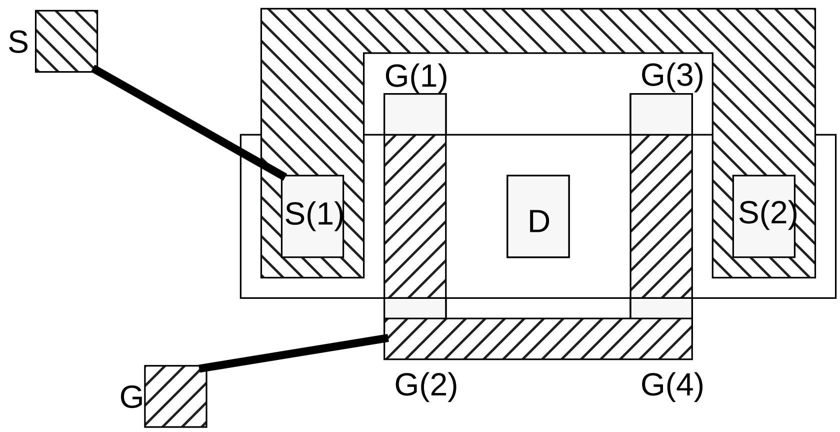


- G(1) and G(2) are weakly connected (inside the device)
- G(3) and G(4) are weakly connected (inside the device)
- The pair G(1)/G(2) must be externally connected to the pair G(3)/G(4)
- S(1) must be externally connected to S(2)

To use this cell in a design, if no external (must-connect) connections are defined, you must make internal (strongly connected) connections so the cell looks like this:

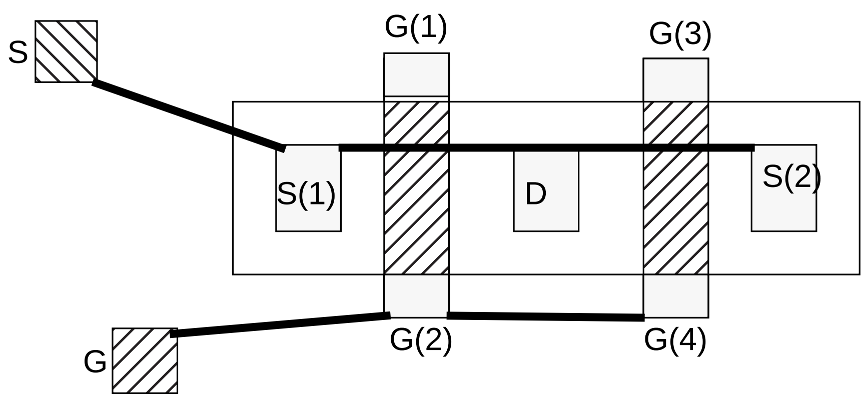


Then, when this cell is used in a design, the flight lines look like this:



If the layout designer makes connections between G and G(1) or G(2) or G(3) or G(4), the Virtuoso® layout editor accepts the connection as complete. If the layout designer makes connections between S and S(1) or S(2), the layout editor accepts the connection as complete.

If you, as a library developer, do not want to complete the connections to the source or gate inside the device, but want the connections made at the next level of hierarchy, you do not wire the internal connections. Then, when the cell is used in a design, the flight lines look like this:



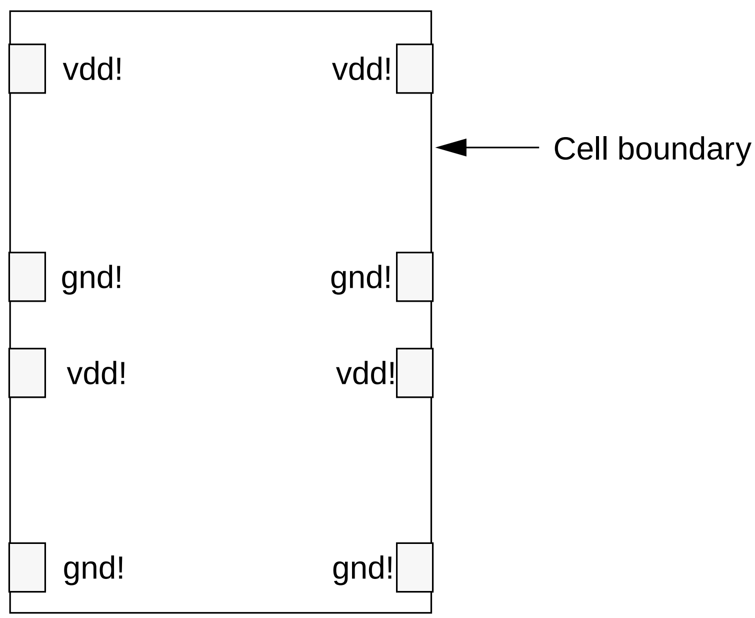
If the layout designer makes a connection between S and S(1), the software does not accept the connection as complete and displays a flight line to S(2). If the layout designer makes a connection between G and G(2), the software does not accept the connection as complete and displays a flight line to G(4).

If you make a connection between G and G(2) and G(4), the software accepts the connection as complete and displays no more flight lines on this net, because G(1) and G(3) are weakly connected to G(2) and G(4) respectively.

If you make a connection between G and G(2), and G between G(1) and G(3), this is considered an error and is indicated by a marker. The software would not accept the internal weak connection between (1) and G(2) to complete the net.

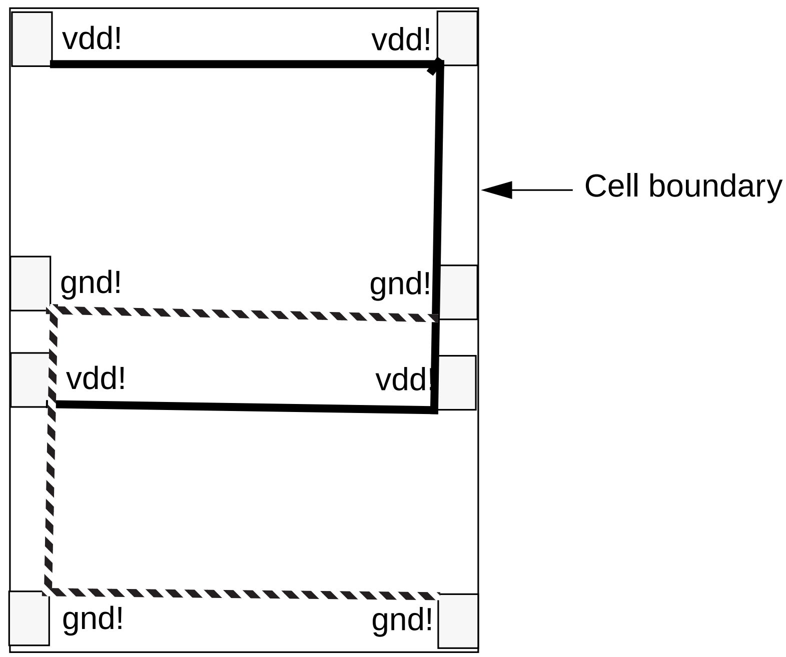
For Layout Designers

If you are working as a layout designer, a typical design might be a two-row-high standard cell layout like this.

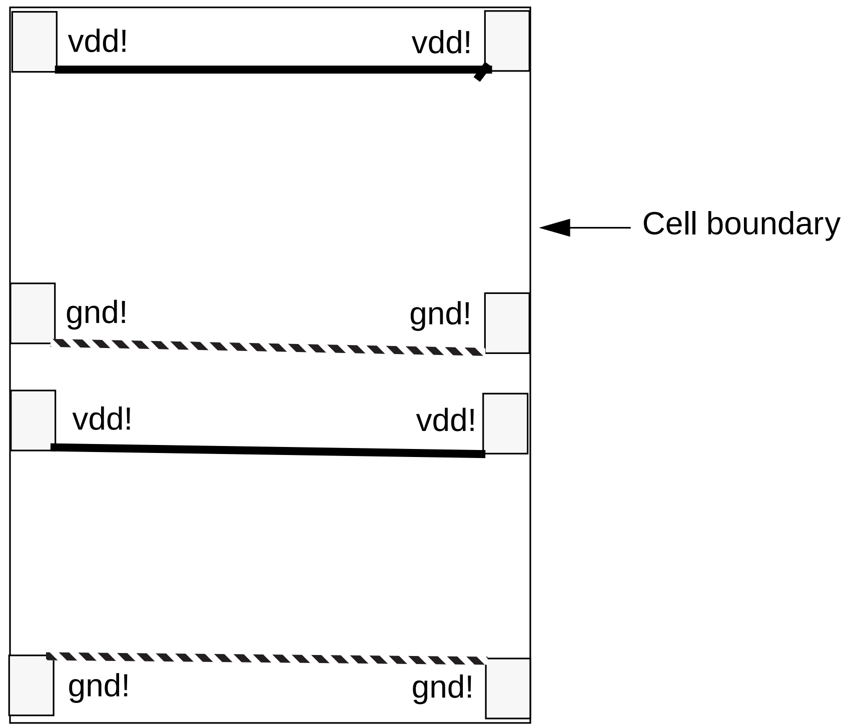


Floorplanning information regarding external connections is provided on the boundary of the design by an outside source, such as a circuit designer or library. Given that information early, you can identify which pins must be must connects, strong connects, or weak connects, and can use that information for device placement and wiring.

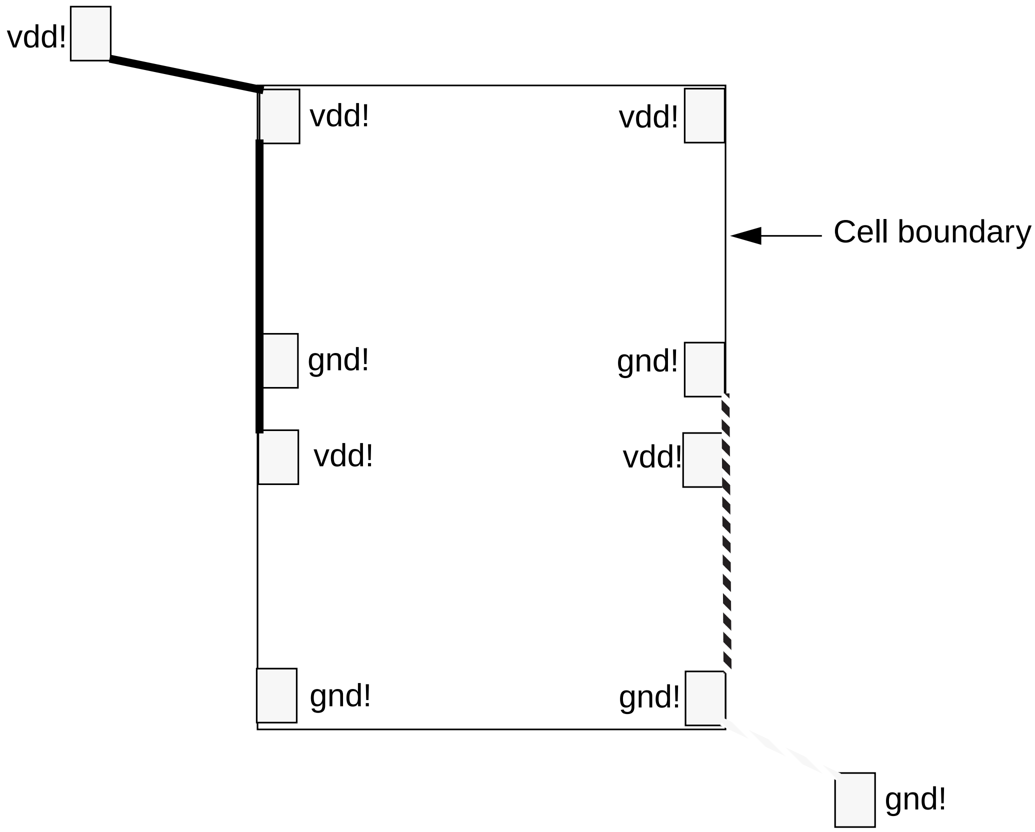
If you define no external connections, the flight lines look like this:



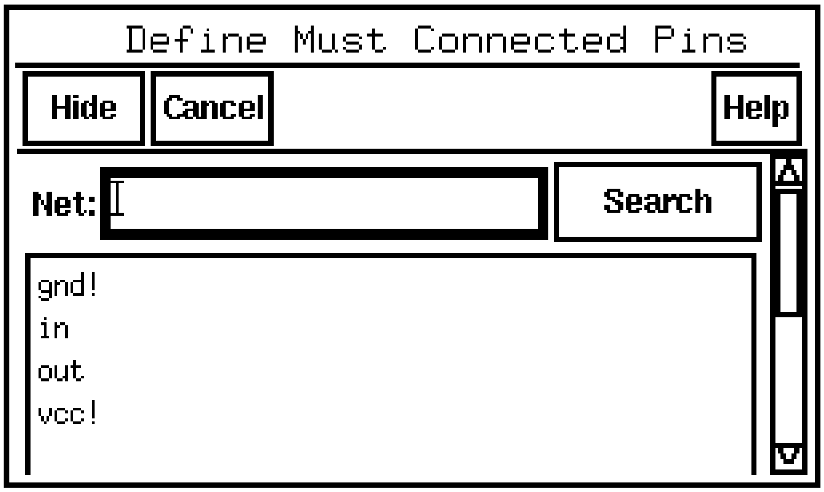
The library developer usually does not connect the top two vdd! pins to the bottom two vdd! pins but does connect the top two vdd! pins together. The same is true for the gnd! pins. So the library developer defines the top two vdd! pins as being externally connected to the bottom two vdd! pins and the same for the gnd! pins. Then, when you are working inside the cell, the flight lines look like this:



When you use the cell in the next level of hierarchy, the flight lines look like this. You must make the connections as shown by the flight lines. You are provided information regarding pin connections by an outside source, such as a circuit designer or library.



About the Define Must Connected Pins Form



Net lets you enter the name of a net you want to select. You can type the following in the *Net* field:

- A net name

  The list box highlights the net and scrolls the list to make the net name visible.

- A part of a net name (*net*, for example)

  The list box highlights all the nets with *net* as an element in the name (*net10, net12, net23, net30, net31*, for example) and scrolls the list to make the first example visible.

- A regular SKILL expression using part of a net name (**bias*, for example)

  The list box highlights all the nets with the string *bias* (*Nbias, NMbias, Pbias*, for example) and scrolls the list to make the first example visible.

Search finds and highlights in the list box the names of nets you type in the *Net* field.

The list box displays the names of all the nets in the design. To select a net from which to define externally connected pins, you can click on the net you want to select or click and hold down the `Control` key to select more than one net.

When you select a net in the list box, the software selects the net in the layout and displays the flightlines for the net.

Defining Must Connect Pins

To define a set of pins on a net to be connected externally,

1. From the Layout window, choose *Connectivity – Define Pins – Must Connect*.

   The Layout window prompts you to click on a pin to select a net.

2. To select the net that includes the pins you want to connect externally, click on a shape in that net.

Note: You can select a net by selecting a shape in the layout before you choose the *Must Connect* command.

   Flight lines appear, connecting only the pins of the net you choose. The flight lines are drawn between pins that still need to be internally (strongly) connected. If incomplete nets are being displayed, the other net flight lines disappear.

   Selecting a net does not make a connection change.

   The Layout window prompts you to select one or more pins to be connected externally.

3. To select the pins you want to connect externally, click on one or more pins in the selected net.

   You can click on a pin, `shift`-click on subsequent pins, or select pins by area (click and drag with the right mouse button).

   Clicking on the pins of the selected net makes the connection change.

   - If you select a single pin, all other pins on the net are deselected and that pin is connected externally to all other pins on the net.
   - If you select multiple pins, those pins are connected strongly (internally) as a set and are connected as must connects (externally) to all other pins on the net.

   When you define the connection of each pin, the flight line display of the net immediately changes to show the connection.

   You can also use the Define Must Connected Pins options form to select nets.

4. To open the Define Must Connected Pins form, press `F3` while the *Define External Pins* command is active. The Define Must Connected Pins form appears.

5. In the form, type the name of a net in the *Net* field or click on the net names in the list box to select them.

   To select more than one net at a time to work on, you can

   - Choose multiple nets in the option form

- Use drag by area when the command line prompts you to select a net

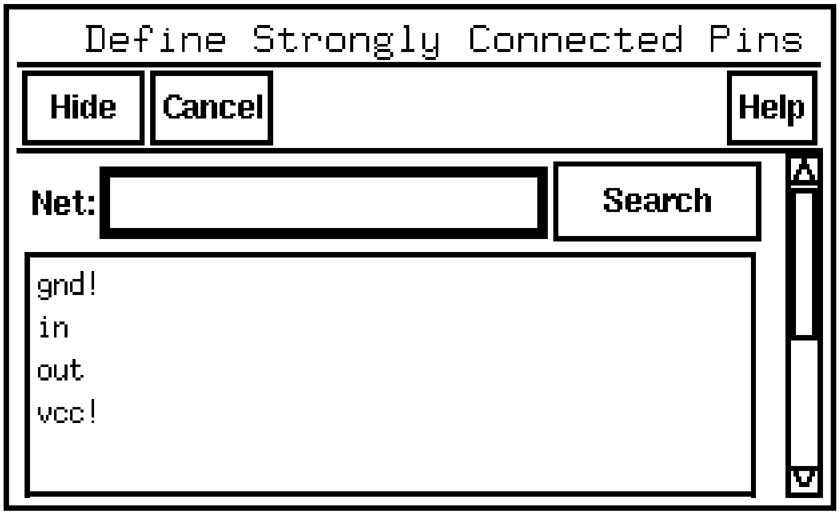
Note: You must define external pin connectivity within a net. You cannot define external pin connectivity between nets.

6. When you are finished, press `Escape` to cancel the command.

   When you define pins as must-connects, the router routes them at the next level of hierarchy.

Note: You can change from the *Must Connect* command to the *Strongly Connected* command to the *Weakly Connected* command to the *Pseudo Parallel Connect* command with the right mouse button. Must Connect pins must be changed to Strongly Connected before being changed to Weakly Connected.

About the Define Strongly Connected Pins Form



Net lets you enter the name of a net you want to select. You can type the following in the *Net* field:

- A net name

  The list box highlights the net and scrolls the list to make the net name visible.

- A part of a net name (*net*, for example)

  The net box highlights all the nets with *net* as an element in the name (*net10, net12, net23, net30, net31,* for example) and scrolls the list to make the first example visible.

- A regular SKILL expression using part of a net name (**bias*, for example)

  The list box highlights all the nets with the string *bias* (*Nbias, NMbias, Pbias*, for example) and scrolls the list to make the first example visible.

Search finds and highlights in the list box the names of nets you type in the *Net* field.

The list box displays the names of all the nets in the design. To select a net from which to define externally connected pins, you can click on the net you want to select or click and hold down the `Control` key to select more than one net.

When you select a net in the list box, the software selects the net in the layout and displays the flightlines for the net.

Defining Strongly Connected Pins

To define a set of pins on a net to be connected strongly (internally),

1. From the Layout window choose *Connectivity – Define Pins – Strongly Connected*.

   The Layout window prompts you to click on a pin to select a net.

2. To select the net that includes the pins you want to connect, click on a shape in that net.

Note: You can select a net by selecting a shape in the layout before you choose the *Strongly Connected* command.

   Flight lines appear, connecting only the pins of the net you chose. The flight lines are drawn between pins that still need to be strongly connected. If incomplete nets are being displayed, the other net flight lines disappear.

   Selecting a net does not make a connection change.

   The Layout window prompts you to select two or more pins to be connected strongly.

3. To select the pins you want to connect strongly, click on one or more pins in the selected net.

   You can click on a pin, `shift-click` on subsequent pins, or select pins by area (click and drag with the right mouse button).

   Clicking on the pins of the selected net makes the connection change.

   - If you select a single pin, the other pins are not deselected, and that pin is connected strongly to the current selection of strongly connected pins.
   - If you select multiple pins, those pins are connected strongly to all other strongly connected pins on the net.

   When you click on pins to be strongly connected, the net flight line display immediately changes to show the new information.

   You can also use the Define Strongly Connected Pins form to select a net.

4. To open the Define Strongly Connected Pins form, press F3 while the *Strongly Connected Pins* command is active. The Define Strongly Connected Pins form appears.

5. In the form, type the name of a net in the *Net* field or click on net names in the list box to select them.

6. In the layout, click on the pins you want to define.

   When you add or remove pins to be strongly connected, the net flight line display immediately changes to show the new information.

To select more than one net at a time to work on, you can

- ❑ Choose multiple nets in the option form
- ❑ Use drag by area when the command line prompts you to select a net

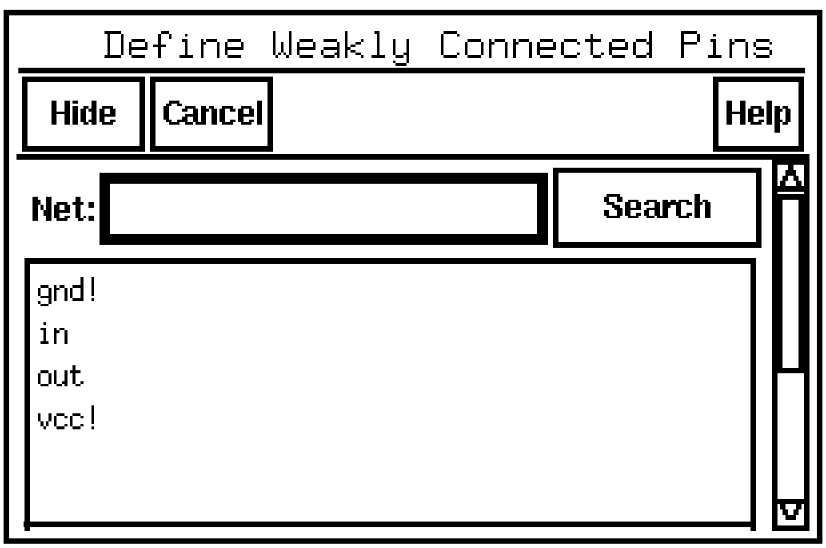
Note: You must define strong pin connectivity within a net. You cannot define strong pin connectivity between nets.

7. When you have finished, press `Escape` to cancel the command.

   When you define pins as strongly connected, the router routes them within the design.

Note: You can change from the *Must Connect* command to the *Strongly Connected* command to the *Weakly Connected* command to the *Pseudo Parallel Connect* command with the right mouse button. Must Connect pins must be changed to Strongly Connected before being changed to Weakly Connected.

About the Define Weakly Connected Pins Form



Net field lets you enter the name of a net you want to select. You can type the following in the *Net* field:

- A net name

  The list box highlights the net and scrolls the list to make the net name visible.

- A part of a net name (net, for example)

The list box highlights all the nets with net as an element in the name (*net10, net12, net23, net30, net31,* for example) and scrolls the list to make the first example visible.

- A regular SKILL expression using part of a net name (*bias, for example)

  The list box highlights all the nets with the string bias (*Nbias, NMbias, Pbias*, for example) and scrolls the list to make the first example visible.

Search finds and highlights in the list box the names of nets you type in the *Net* field.

The list box displays the names of all the nets in the design. To select a net from which to define externally connected pins, you can click on the net you want to select or click and hold down the `Control` key to select more than one net.

When you select a net in the list box, the software selects the net in the layout and displays the flightlines for the net.

Defining Weakly Connected Pins

To define a set of pins on a net to be connected externally because the internal physical connection has a significant resistance,

1. From the Layout window choose *Connectivity – Define Pins – Weakly Connected*.

   The Layout window prompts you to click on a pin to select a net.

2. To select the net that includes the pins you want to connect externally for this reason, click on a shape in that net.

Note: You can select a net by selecting a shape in the layout before you choose the *Define Pins – Weakly Connected* command.

   Flight lines appear, connecting only the pins of the net you chose. Dotted flight lines are used to show the internal high-resistance path between the pins. If incomplete nets are being displayed, the other net flight lines disappear.

   Selecting a net does not make a connection change.

   The Layout window prompts you to select two or more pins to be connected.

3. To select the pins you want to connect, click on one or more pins in the selected net.

   You can click on a pin, `shift-click` on subsequent pins, or select pins by area (click and drag with the right mouse button).

   Clicking on the pins of the selected net makes the connection change.

- ❑ If you select a single pin, the other pins are not deselected, and that pin is connected externally to the current selection of externally connected pins.
- ❑ If you select multiple pins, those pins are connected externally to all other externally connected pins on the net.

When you click on pins to be externally connected as weak-connects, the net flight line display immediately changes to show the new information.

You can also use the Define Weakly Connected Pins form to select a net.

4. To open the Define Weakly Connected Pins form, press F3 while the *Weakly Connected Pins* command is active. The Define Weakly Connected Pins form appears.
5. In the form, type the name of a net in the *Net* field or click on net names in the list box to select them.
6. In the layout, click on the pins you want to define.

   When you add or remove pins to be weakly connected, the net flight line display immediately changes to show the new information.

   To select more than one net at a time to work on, you can

   - ❑ Choose multiple nets in the option form
   - ❑ Use drag by area when the command line prompts you to select a net

Note: You must define weak pin connectivity within a net. You cannot define strong pin connectivity between nets.

7. When you have finished, press `Escape` to cancel the command.

   When you define pins as weakly connected, the router routes them within the design.

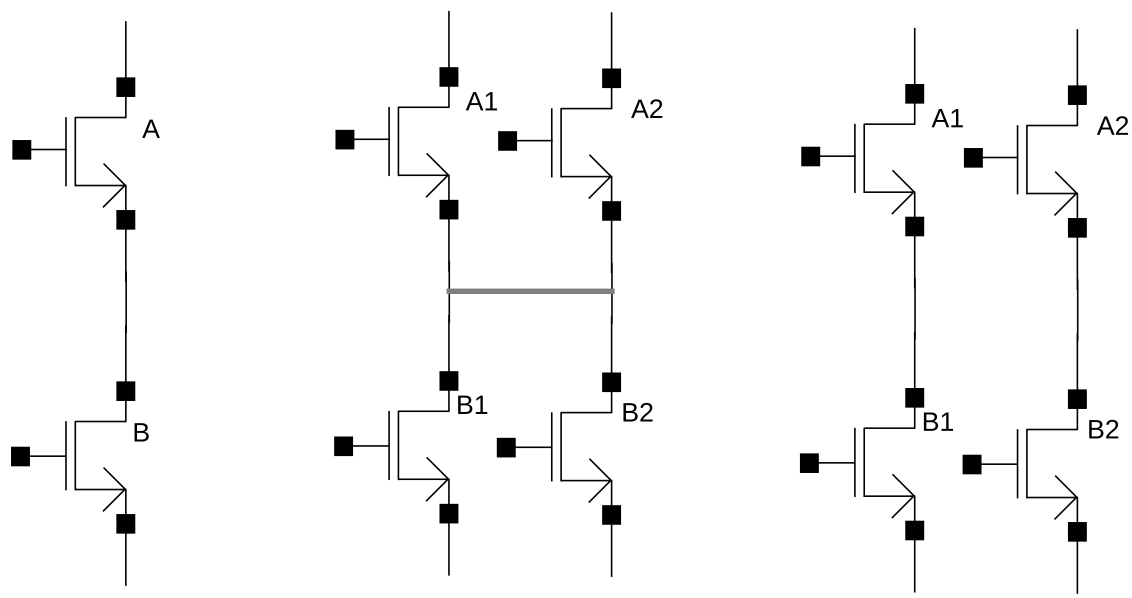
Note: You can change from the *Must Connect* command to the *Strongly Connected* command to the *Weakly Connected* command to the *Pseudo Parallel Connect* command with the right mouse button. Must Connect pins must be changed to Strongly Connected before being changed to Weakly Connected.

About Pseudo Parallel Connections

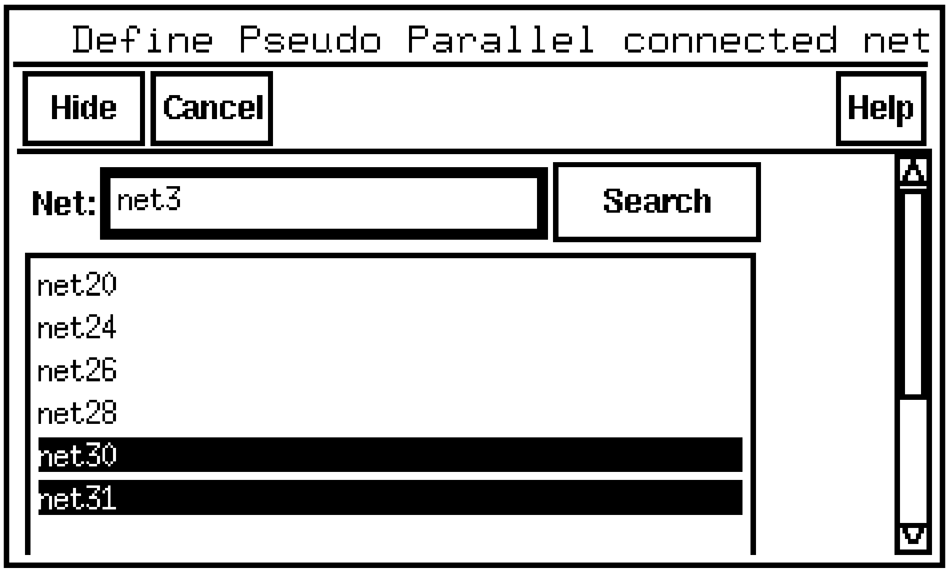
Pseudo parallel connections are made when you use the *Gen From Layout* command with the Folding and Chaining option active and are made automatically as part of chaining transistor and folding.

You might use pseudo parallel connect manually if you had a symmetric series of N-transistors A and B tied in a series pulldown chain to ground, as shown below. Both are folded into two legs, with A1 and B1 connected in series to ground, and likewise A2 and B2 connected in series to ground—both pairs in parallel. The nodes between A1 and B1, and A2 and B2, are electrically equivalent. However, no current passes through that connection and it need not actually be made.

The Assura™ Diva® verification tool understands pseudo parallel connections and does not report unconnected nets in such situations.



About the Define Pseudo Parallel Connected Net Form



Net lets you enter the name of a net you want to select. You can type the following in the *Net* field:

- A net name

  The list box highlights the net and scrolls the list to make the net name visible.

- A part of a net name (*net*, for example)

  The list box highlights all the nets with *net* as an element in the name (*net10, net12, net23, net30, net31*, for example) and scrolls the list to make the first example visible.

- A regular SKILL expression using part of a net name (**bias*, for example)

  The list box highlights all the nets with the string *bias* (*Nbias, NMbias, Pbias*, for example) and scrolls the list to make the first example visible.

Search finds and highlights in the list box the names of nets you type in the *Net* field.

The list box displays the names of all the nets in the design. To select a net from which to define parallel connections, you can click on the net you want to select or click and hold down the `Control` key to select more than one net.

When you select a net in the list box, the software selects the net in the layout and displays the flightlines for the net.

Defining Pseudo Parallel Connected Nets

Pseudo parallel connections are like must connects within an instance and represent cases where instance terminals do not need to be connected because current does not pass between them.

To define a set of instance terminals on a net for pseudo parallel connection,

1. From the Layout window, choose *Connectivity – Define Pins – Pseudo Parallel Connect*.

   The Layout window prompts you to click to select a net.

   Note: You can select a net by selecting a shape in the layout before you choose the *Pseudo Parallel Connect* command.

2. To select the net that includes the instance terminals you want to connect, click on a shape in that net.

   Flight lines appear, connecting only the pins of the net you choose. The flight lines are drawn between pins that can be pseudo parallel connected. If incomplete nets are being displayed, the other net flight lines disappear.

   Selecting a net does not make a connection change.

   If you did not select a net or instance terminal before you chose the command, the Layout window prompts you to select an instance terminal (instTerm).

3. To select the instance terminals you want to connect in pseudo parallel connection, click on one or more pins in the selected net.

   You can click on an instance terminal, `shift-click` on subsequent pins, or select terminals by area (click and drag with the right mouse button).

   Clicking on the instance terminals of the selected net makes the connection change.

   When you define the connection of each terminal, the flight line display of the net immediately changes to show the connection.

   You can also use the Define Pseudo Parallel Connected Pins options form to select nets.

4. To open the Define Pseudo Parallel Connected Pins form, press F3 while the *Pseudo Parallel Connect* command is active.

   The Define Pseudo Parallel Connected Net form appears. The form lists all nets without I/O pins (except le_ex_# nets).

5. In the form, type the name of a net in the *Net* field or click on the net names in the list box to select them.

To select more than one net at a time to work on, you can

- Choose multiple nets in the option form
- Use drag-by-area when the command line prompts you to select a net

Note: You must define pseudo parallel connectivity within a net. You cannot define pseudo parallel connectivity between nets.

6. When you are finished, press the `Escape` key to cancel the command.

   When you define pins as pseudo parallel connects, the router does not route them.

Note: You can change from the *Must Connect* command to the *Strongly Connected* command to the *Weakly Connected* command to the *Pseudo Parallel Connect* command with the right mouse button.

Checking Connectivity of Pins

To see whether pins on a net are defined for external connection,

1. From the Layout window *Connectivity* menu, choose *Define Pins – Must Connect*.
2. Press F3 to see the *Define Must Connected Pins* form.

   The form appears, showing all nets that include terminals (possible connections to devices external to the device).

3. Click on one of the nets shown in the form.

   Flight lines for the net appear on the layout, showing whether the terminals are defined as must-connect, strong-connect, or weak-connect.

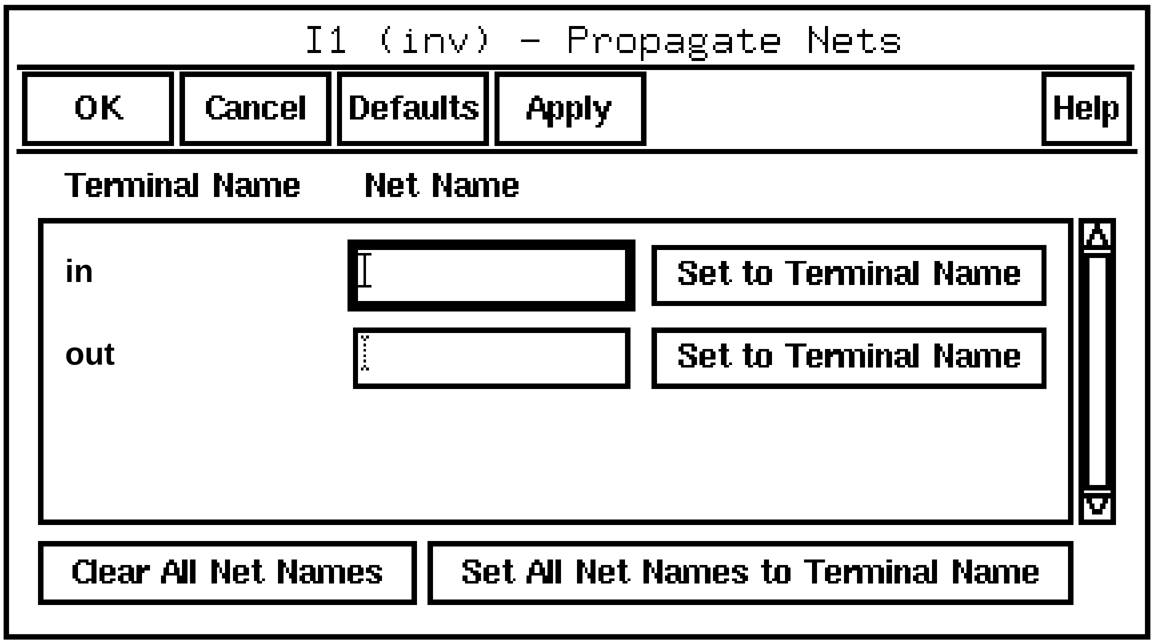
Propagating Nets

The *Propagate Nets* command maps terminals on an instance you select to nets in the cellview; that is, it defines how the instance is logically connected to the cellview. Using this command, you can assign or reassign a top-level net name for each terminal in the selected instances. For each such terminal, the command creates a top-level net, if necessary, and creates an instTerm on that net for the instance terminal.

Routers use this logical connectivity information to physically connect the pins of the instance to the rest of the layout.

You use this command when you do not have a schematic for your layout cellview.

About the Propagate Nets Form



Terminal Name is the terminal name. Several pins can be defined on one terminal.

Net Name is the top-level net name. If a net name field is blank, the corresponding terminal is not currently mapped to a top-level net. Clicking *Defaults* displays the current top-level net name of the terminal.

Set to Terminal Name sets the net name to the terminal name.

Clear All Net Names clears the text fields.

Set All Net Names to Terminal Name sets all net names to the terminal names.

Promoting Net Information

Note: In the following steps, “top-level” refers to the cellview in which the instance is placed.

To promote net information from an instance in your layout,

1. Select one or more instances in the layout.
2. From the Layout window, choose *Connectivity – Propagate Nets*.

The Propagate Nets form appears. If you select more than one instance, the *Previous* and *Next* buttons are displayed at the top of the form.

The instance displayed in the form is highlighted in the layout. The form title shows the name and master cell of the instance currently highlighted.

The form shows the terminals of the instance and the top-level nets they are mapped to, if any.

3. For each terminal, type in the top-level net to which you want to map the terminal.

If a *Net Name* field is left blank, the corresponding terminal is not mapped to a top-level net. *Net Name* fields can be left blank, but the pins on the corresponding terminals will not be routed.

If you want the net to have the same name as the terminal name, type in the terminal name or click *Set to Terminal Name*.

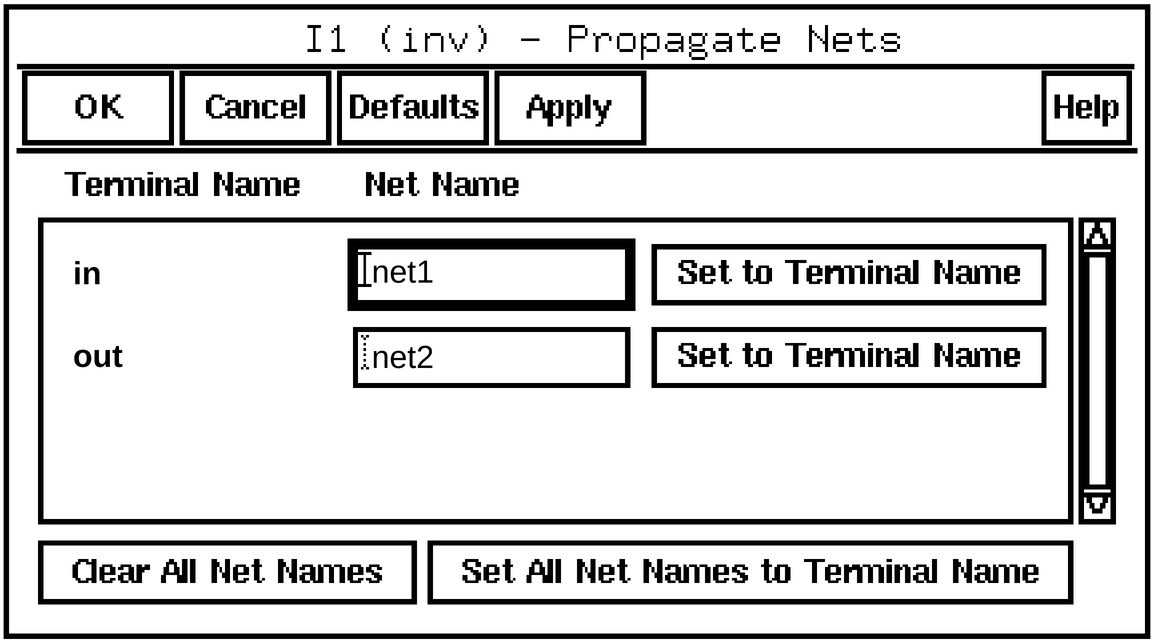
If you want all the top-level nets to have the same name as the terminal name, click *Set All Net Names to Terminal Name*.

If you want to undo all the changes to that instance, click *Defaults*. The net names revert to their current value.

If you want to clear all the net names, click the *Clear All Net Names* button.

4. Click *Previous* and *Next* to edit other instances.

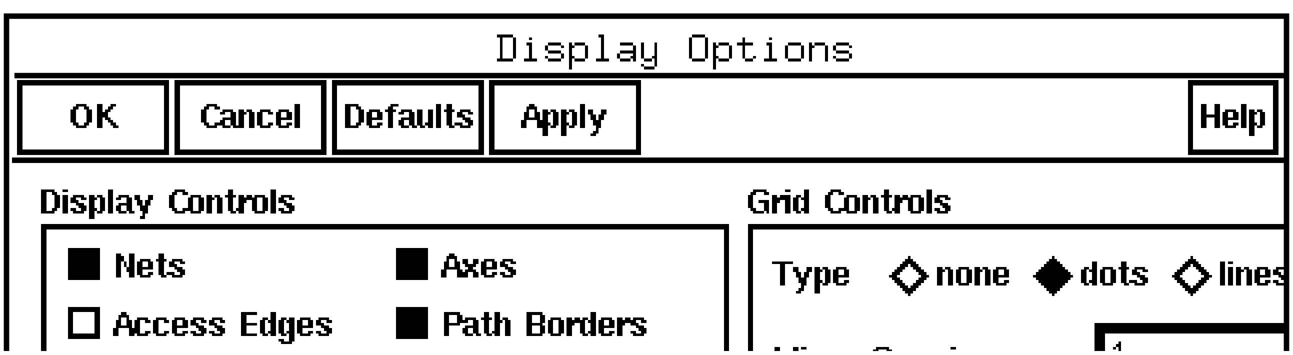
The form keeps track of all the changes you have made. You do not need to click *Apply* after each change before editing other instances.



5. Click *OK* to apply all the changes.

   All the changes you have made are saved to the database.

6. Choose *Options – Display*.
7. In the *Display Controls* area, turn on *Nets*.

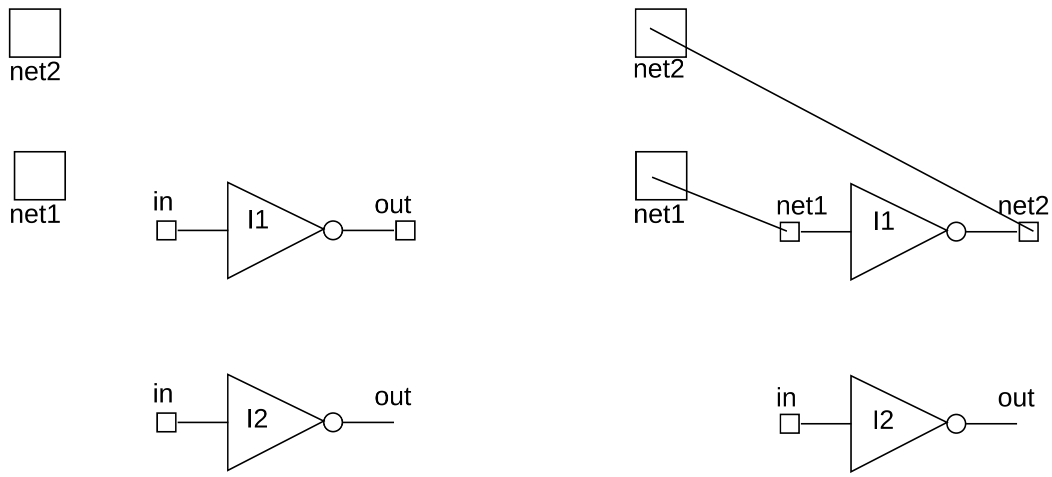


8. Click *OK*.
9. Choose *Window – Redraw*.

The new net assignments are displayed in the Layout window.

Before assigning new nets with. the *Propagate Nets* command.

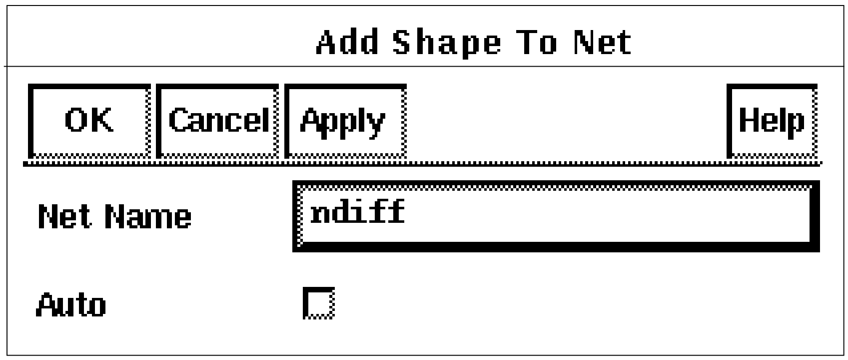
After assigning new nets with the *Propagate Nets* command.



Adding and Deleting Shapes on Nets

The *Add Shape To Net* command lets you add selected shapes to nets that are attached to pins. A shape cannot be added to more than one net. The *Delete Shape From Net* command lets you delete selected shapes from a net.

About the Add Shape to Net Form



Net Name lets you enter a net name when the *Auto* button is off.

Auto determines whether you enter a net name or get the net name automatically:

- *On* selects the net based on the pins overlapping the selected shapes. The overlapping pin and shape must have the same layer-purpose pair. Each selected shape is given the net of a pin that overlaps it, if the shape does not already have a net.
- *Off* lets you specify the net name. The selected shape does not have to be overlapping the specified pin.

Adding Shapes to a Net

To add a shape to a net,

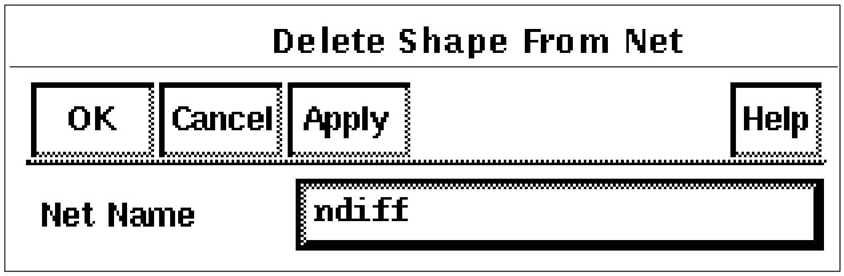
1. Select one or more shapes.
2. Choose *Connectivity – Add Shape To Net*.

   The Add Shape To Net form appears.
3. Do one of the following:
   - To specify the net based on the selection, turn *Auto* on. For each shape, the software looks for a pin that overlaps the shape and adds the shapes to the net.
   - To specify the net manually, turn *Auto* off and type the net name.
4. Click *OK* to close the form and add the shapes to the nets.

Important

When there are zero-area dot pins on the edge of the shape to be added, the pin is used only if one of its access directions points toward the center of the shape. Symbolic pins must have a dot on the correct layer, overlapping the shape to be added.

About the Delete Shape From Net Form



Net Name shows the net of the selected shape. If more than one shape is selected, *Net Name* shows the net of the first shape. If the net name shown is not the net you want the shapes deleted from, type the correct net name.

Deleting Shapes from a Net

To delete a shape from a net,

1. Select one or more shapes to be deleted from a net.
2. Choose *Connectivity – Delete Shape From Net*.

   The Delete Shape From Net form appears.
3. If the net name shown is not the net you want the shapes deleted from, type the correct net name.
4. Click *OK* to close the form and delete the shapes from the specified net.

Tracing Nets Using Mark Net

The *Mark Net* command lets you visually trace a net in a layout design without having to use a schematic. *Mark Net* extracts the metal and via layer information from the technology file and highlights the metal and via layers in the layout as the net passes from one layer to the next through the hierarchy. The trace is not a selectable object, it is just a highlight of the net.

Your technology file must define metal and via layers. If it does not, *Mark Net* does not work. The *Technology File and Display Resource File User Guide* describes how to create and edit a technology file. For information on defining metal layers, see Layer Definitions. For information on specifying via layers, see Layer Rules.

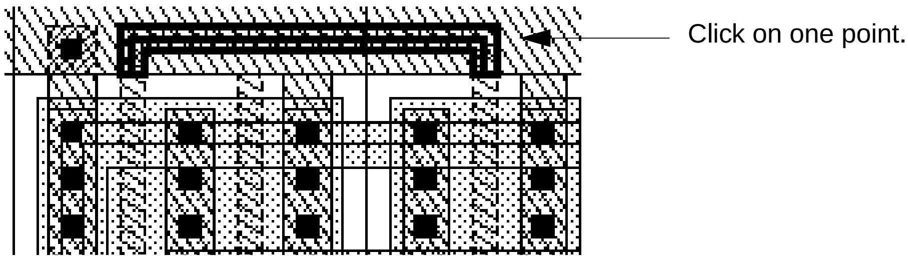
To trace a net using *Mark Net*,

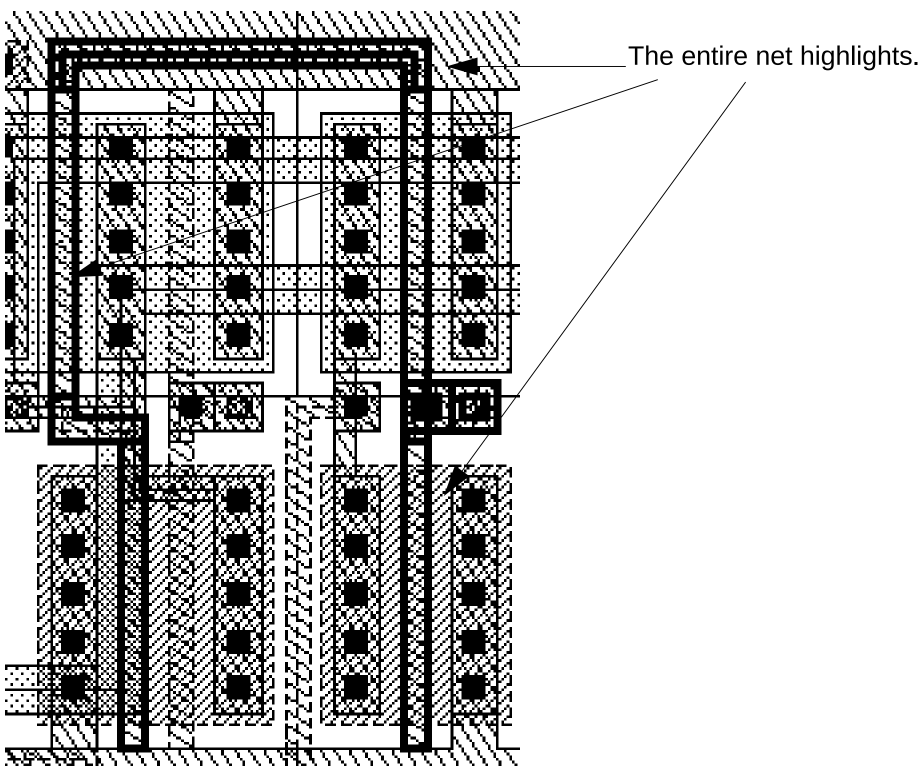
1. Choose *Connectivity – Mark Net*.

   This message appears in the Command Interpreter Window (CIW):

   `Point at a net.`

2. Click on the net you want to highlight.





3. Press `Escape` to end *Mark Net* and to remove the highlight from the layout.

13

Plotting Your Design

This chapter contains these topics:

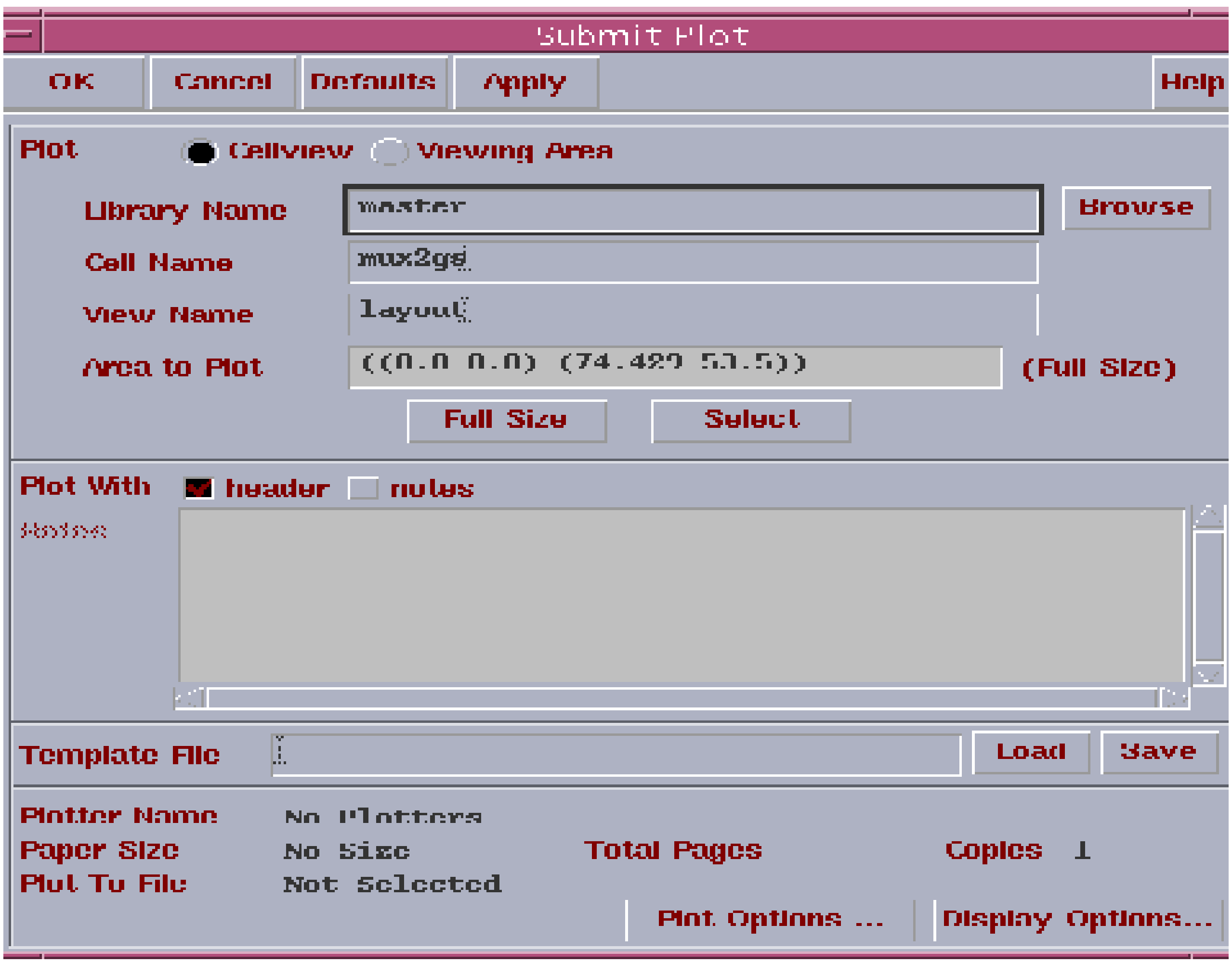
Using the Submit Plot Form

The Submit Plot form plots a part of or an entire cellview.

About the Submit Plot Form

To open the Submit Plot form,

➤ Choose *Design – Plot – Submit.*



Plot sets how much of the cellview to plot.

Cellview plots the entire cellview.

Viewing Area plots the area shown in your window.

Library Name, **Cell Name**, and **View Name** set the library, cell, and view names of the cellview you want to plot.

Browse lets you select the library, cell, and view names by clicking on them in the browser.

Area to Plot lets you set what area to plot.

Full Size plots the entire cellview, and the cellview bounding box coordinates are displayed.

Select lets you select an area to plot.

Plot With specifies what to include in the output.

Header prints a separate header page listing

Your name
Today’s date
The total plot size
The magnification used
Number of pages printed
The library, cell, and view names and the version number of the cellview

Notes lets you type notes that appear as part of the plot header.

Template File specifies the name of the ASCII file.

Load updates the Submit Plot form with the settings from the file.

Save saves the settings to the file.

Plotter Name, **Paper Size**, **Total Pages**, **Copies**, and **Plot to File** display the setting specified in the Plot Options form. To change any of these fields, click on *Plot Options*.

Plot Options opens the Plot Options form, to let you specify the plotter, page size, and plot job settings.

Display Options opens the plotter Display Options form, to let you control the appearance of the objects you plot.

Plotting a Cellview

To plot all data in a cellview,

1. Choose *Design – Plot – Submit.*

   The Submit Plot form appears.

2. Set *Plot* to *Cellview*.

3. Click *Plot Options*.

   The Plot Options form appears.

4. Set the plotter, paper, and plot job settings to your specifications.

5. Click *OK* in the Submit Plot form.

   The entire design in this cellview is plotted to the printer you chose.

Plotting the Window Contents

To plot the portion of the cellview shown in the cellview window,

1. Choose *Design – Plot – Submit*.

   The Submit Plot form appears.

2. Set *Plot* to *Viewing Area*.

3. Click *Plot Options*.

   The Plot Options form appears.

4. Set the fields to your specifications.

5. Click *OK* in the Submit Plot form.

   The portion of the cellview that you can see in the window is plotted to the printer you chose.

Plotting a Selected Area of a Cellview

To plot a selected area of the cellview,

1. Choose *Design – Plot – Submit*.

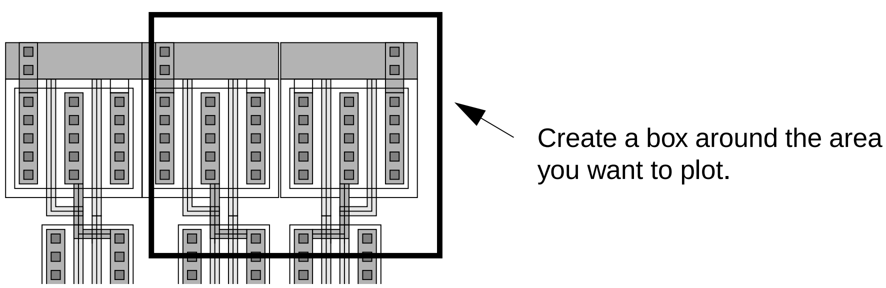
   The Submit Plot form appears.

2. Set *Plot* to *Cellview*.

3. Choose *select* from the *Plot Area* cyclic field.

   You are prompted to select the area you want to plot.

4. Click to create a box around the area you want to plot.



5. Click *Plot Options*.

   The Plot Options form appears.

6. Set the fields to your specifications.

7. Click *OK* in the Submit Plot form.

   The area you chose is plotted.

Plotting Hierarchy

To control whether a plot shows details about cell instances or arrays,

1. Choose *Design – Plot – Submit*.

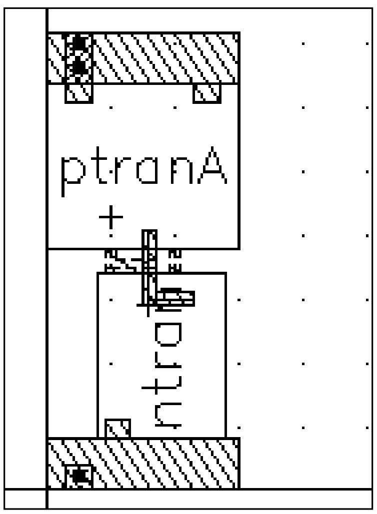
   The Submit Plot form appears.

2. Click *Display Options.*

   The Display Options form appears.

3. Set the options you want.
4. Click *OK*.
5. Click *OK* in the Submit Plot form.

   The cellview is plotted using the display options you set.



A plot with *Display Levels* set *From 0 To 0*, so instance details do not appear, and grid points set on.

Enlarging or Reducing the Plot Size

To control the plot size by setting a scale factor or specifying the plot height or width,

1. Choose *Design – Plot – Submit*.

   The Submit Plot form appears.

2. Click *Plot Options*.

   The Plot Options form appears.

3. Do one of the following:
   - To scale the plot by any factor, type the factor into the *Scale* field.
   - To set the dimensions of the plot, type the value into the *Plot Size* fields and choose the measurement from the cyclic field.

The other fields are automatically updated to reflect the value you typed. For example, if you type in a scale factor, you can see the resulting X and Y plot size in the *Plot Size* and *Total Plot Size* fields.

4. Click *OK* in the Plot Options form.
5. Click *OK* in the Submit Plot form.

   The design is plotted at the scale or size you chose.

Plotting at a Later Time

If you have a large design or the plotter is currently in use, you might want to plot your cellview at a later time.

To schedule your plot for a later time,

1. Choose *Design – Plot – Submit*.

   The Submit Plot form appears.

2. Click *Plot Options*.

   The Plot Options form appears.

3. Do one of the following:
   - ❑ Choose a time in the *Queue Plot Data At* cyclic fields.
   - ❑ To save the plot to a file formatted for your chosen plotter, type a filename in the *Send Plot Only To File* field.
4. Click *OK* in the Plot Options form.
5. Click *OK* in the Submit Plot form.

   If you set a specific plot time, your job is automatically sent to the printer or plotter at that time.

If you saved a plot data file formatted for your plotter, you can later use the appropriate UNIX command to plot the file. For example, if you can plot files using the `lpr` command, you can type

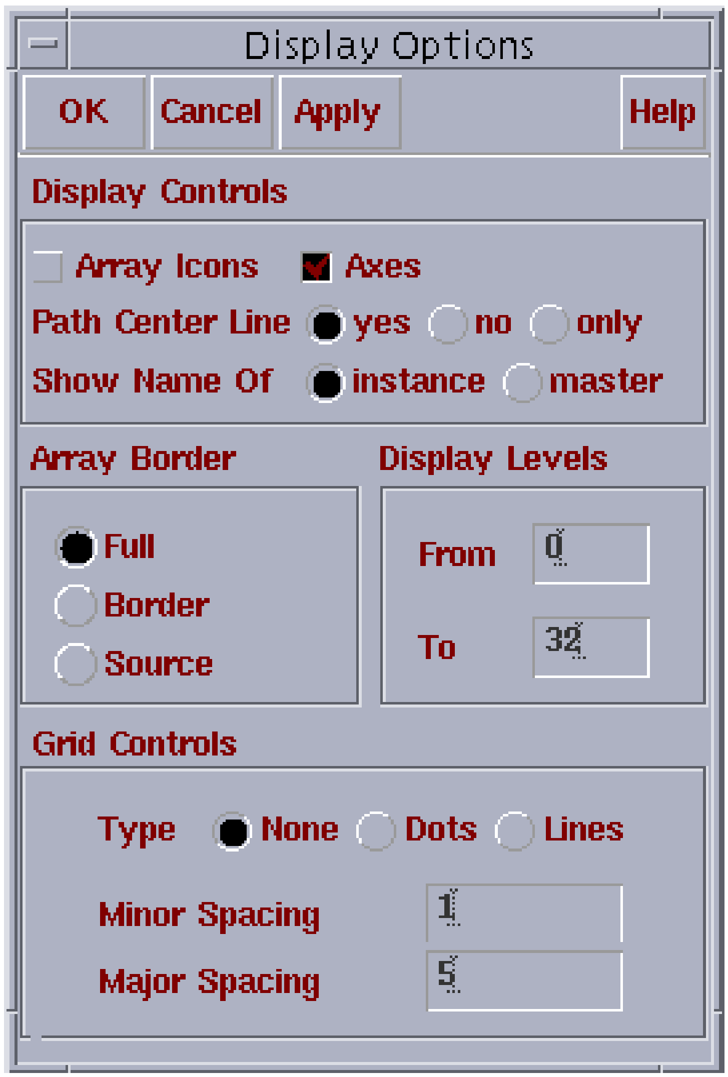
```
lpr -Pyour_plotter plot_filename
```

where *`your_plotter`* is the name of the plotter defined in your `/etc/printers.conf` file and *`plot_filename`* is the name you typed in the Plot Options form.

Using the Display Options Form

The Display Options form for the plotter sets plot controls but does not affect the current display of the cellview. To open the Display Options form for the plotter, click *Display Options* in the Submit Plot form.

About the Display Options Form



Array Icons shows only outlines of the instances in arrays, when *Array Border* is set to show only instance outlines.

Axes includes the cellview axes in the plot.

Path Center Line sets how you want paths to appear in the plot.

yes plots the path center line.

no does not plot the path center line.

only plots only the path center line.

Show Name Of when *Display Levels* is set to show only instance outlines, sets whether the instance name (for example, `I1`) or the master cell name appears on each instance.

Array Border sets which instances in the array you want plotted.

Full prints all instances in the array.

Border prints only the instances around the outside edge of the array.

Source prints only the instance in the lower left corner of the array.

Display Levels sets the first (*From*) and last (*To*) levels in the design hierarchy that are plotted in detail. The hierarchy levels are numbered 0 to 32. The current cellview is level 0, instances inside of it are level 1, and so forth.

Grid Controls controls whether you plot the cellview grid and how it appears.

Type sets whether you want no grid, dots, or lines plotted.

none does not print the grid.

dots prints a dot for each grid point.

lines prints a grid of lines.

Minor Spacing and **Major Spacing** control the spacing, in user units (typically microns), between the dots or lines of the grid.

Using the Plot Options Form

The Plot Options form controls the plotter, paper, and plot job setting.

About the Plot Options Form

To open the Plot Options form,

➤ Click *Plot Options* in the Submit Plot form.

Display Type lists all the plotter types defined in your technology file. The default, *display*, plots the same stipple patterns, colors, and line styles used to display the cellview on your monitor screen.

Plotter Name sets the type of plotter as defined in your Cadence® plotter support file. This field includes all plotters from your `.cdsplotinit` file.

Paper Size sets the paper size. This field includes all paper sizes supported by the selected plotter.

Orientation sets what edge of the paper to use as the top.

Portrait plots the cellview as it appears in the window.

Landscape rotates the plot 90 degrees counterclockwise.

Automatic prints whichever way fits best.

Scale scales the plot by the entered factor. Entering a scale updates the *Plot Size* and *Total Plot Size* fields.

Center Plot automatically adjusts the offset, centering the plot on the plotted page. If the plot spans multiple pages, the plot is centered across all pages.

Fit to Page scales the plot to fit on one page. The *Scale* and *Plot Size* fields are updated to reflect the scaled plot.

Plot Size is the width and height of the cellview or viewing area after it is plotted. You can specify what size you want the image to be.

The cyclic field to the right of the *Plot Size* fields specifies the display units for all the fields on the Plot Options form.

Offset specifies the X and Y origin of the cellview or viewing area on the plotted page. If the plot spans more than one page, the offset is from the bottom left corner.

Total Plot Size is the sum of the plot size and the offset. You cannot edit this field.

Image Position assists in setting desired plot options. It is a graphical representation displayed in the form using orientation, scale, fit, plot size, and offset chosen in the plot options form and shown on an outline of the selected paper size.

Total Pages displays the number of pages that will be printed. You cannot edit this field.

Number Of Copies indicates the number of copies that will print.

Local Tmp Directory is the temporary directory used by the *Plot* command.

Queue Plot Data At sets the time and day to run the plot job.

Send Plot Only To File saves the plot to the specified file formatted for your chosen plotter. You can then use the appropriate UNIX commands for your plotter to plot this file.

Mail Log To sends e-mail to the specified address when the plot finishes.

Using a Plot Template File

If you want to use the same plot options for other schematics, you can store the options in a plot template file. The plot template file stores plot options in property list format. You can use a template file to plot schematics in batch mode.

Creating a Plot Template File

To create a template file,

1. Choose *Design – Plot – Submit*.

   The Submit Plot form appears.

2. In the *Template File* field, type the path to the directory in which you want to store the file.
3. Complete the rest of the form to indicate all settings you want to save for the plotting.
4. Click *Save*.
5. Click *OK*.

Loading a Plot Template File

When you load a template file, the system imports plot setup information that you have previously saved.

To load a template file,

1. Choose *Design – Plot – Submit*.

   The Submit Plot form appears.

2. In the *Template File* field, type the path to the directory in which you want to store the file.

   You can also type the path to one of the sample plot template files:

   `your_install_dir/tools/dfII/samples/plot/schPlot.il`

   `your_install_dir/tools/dfII/samples/plot/schMetPlot.il`

3. Click *Load*.
4. Click *OK*.

Creating a Default Plot Template File

To create a default plot template file,

➤ Add the following line to your `.cdsinit` file:

`schPlotTemplate = "`*`/usr/myPath/myTemplate`*`"`

The system loads the default plot template file when you use the *Design – Plot – Submit* command for the first time. Once you change the path in your `.cdsinit` file, use the *Load* button to load other template files.

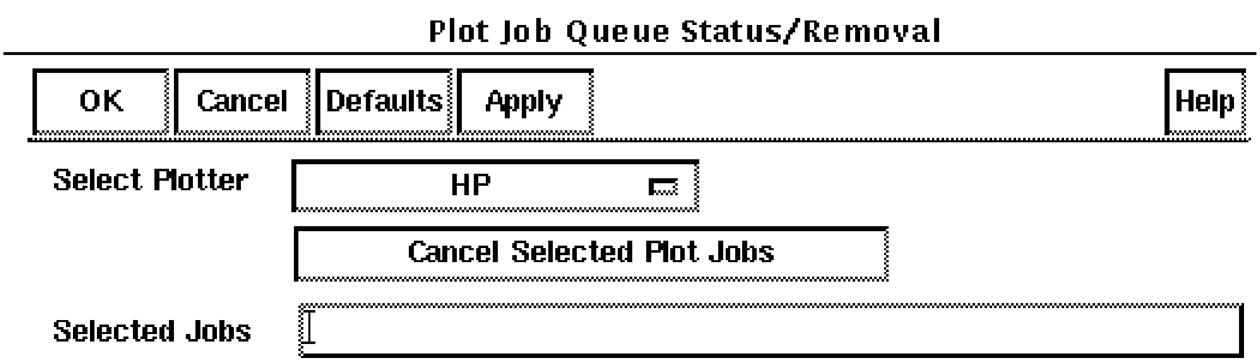
Using the Queue Status Form

You can monitor your plot with the *Queue Status* form. This form lets you view a list of and cancel any plotting jobs waiting at a selected plotter or printer.

About the Queue Status Form

To open the Queue Status form,

➤ Choose *Design – Plot – Queue Status*.



Select Plotter lists the available plotters as defined in your `.cdsplotinit` file.

Cancel Selected Plot Jobs removes the selected plot jobs from the plot queue.

Selected Jobs lets you type the number of each print job you want to select.

Job list lists print and plot jobs waiting to be printed. You can select any job by clicking on that job.

active	cris	1	standard input	40326
1st	cris	2	standard input	40447
2nd	cris	3	standard input	40505
3rd	cris	4	standard input	40500

Using the Queue Status Form

To view or cancel any of the print jobs you sent to a plotter,

1. Choose *Design – Plot – Queue Status*.

   The Queue Status form appears.

2. Choose a plotter from the *Select Plotter* cyclic field.
3. Do either of the following:
   - Type the number of the job you want under *Selected Jobs* field.
   - Click on a job number in the list of queried jobs.
4. Click *Cancel Selected Plot Jobs* to cancel a job.
5. Click *OK*.

Required Files for Plotting

Before you can plot, your system must contain certain files and directories, depending on the type of UNIX operating system you use.

For complete information on how to load the plotting utility and how to set up your system to plot, see the Cadence *Plotter Configuration User Guide*.

Plot Support File

The Cadence software uses the `.cdsplotinit` file to identify each printer or plotter on your network.

The `.cdsplotinit` file includes printer and plotter names for your system, along with information such as the UNIX commands used to manage plotting, the maximum pages to send, the print resolution (dots per inch), a map of pen numbers to colors (for plotters), and the available paper sizes.

You can store the `.cdsplotinit` file in any or all of the following directories. The Cadence software looks for plotter definitions in each of these files, in the following order:

your_install_dir/`tools/plot/.cdsplotinit`	Cadence design framework II directory
`./.cdsplotinit`	Your working directory
`~/.cdsplotinit`	Your home directory

If the same plotter is defined in more than one `.cdsplotinit` file, the software uses the last definition it finds for that plotter.

To see an annotated sample of the `.cdsplotinit` file, type the following in your top design framework II directory:

```
more tools/plot/samples/cdsplotinit.sample
```

Plotter Names in the Submit Plot Form

The `.cdsplotinit` file controls the following information in the Submit Plot form:

- Names of printers and plotters
- Paper sizes defined for those printers and plotters

For example, the following portion of a `.cdsplotinit` file lists entries for an Imagen PostScript printer. The underlined information in the example appears in the Plot form.

Printer name that appears in the form

```
UP Imagen|Imagen Postscript: \
:spool=lpr -Pup -Lultrascript: \
:query=upq: \
:remove=uprm $3: \
:manufacturer=QMS: \
:type=postscript1: \
:maximumPages#30: \
:resolution#300: \
:paperSize="A" 2400 3150 75 75: \
:paperSize="B" 3150 4950 75 75:
```

Paper sizes that appear in the form

14

Handling Unexpected Results While Using the Virtuoso Layout Editor

This chapter contains these topics:

Design Access Problems

You might encounter the following situations while working in the Virtuoso® layout editor.

I Can't Find a Library

If you do not see the name of the library you want in the *Library Name* field of the Open File form, one of the following might be true:

- The library path in the `cds.lib` file is incorrect
- The library is not in the `cds.lib` file

To fix either of these problems,

➤ Edit your `cds.lib` file.

I Can't Open a Cellview

If you cannot open a cellview in a library, you might not have read access to the cellview files or the library might be empty. In these situations, the Open File form displays `<None>` in the *Cell Name* and *View Name* fields.

To gain read access to the cellviews in a library,

➤ Do one of the following:

  - ❑ Change the access permissions using the *Library Manager – Edit – Access Permissions form*.
  - ❑ Use the UNIX command `chmod` to change the access permissions in the UNIX directory containing the library.

I Can't Write to a Cellview

If you have write access to a library but cannot open a cellview to edit, one of the following might be true:

- You do not have write access for the cellview file
- Another user is editing the cellview and locked it

If the owner of the cellview set privileges to read only, you see a message like this:

`The cellView could not be opened for edit. Do you want to open it for read?`

To gain write access to the cellviews in a library,

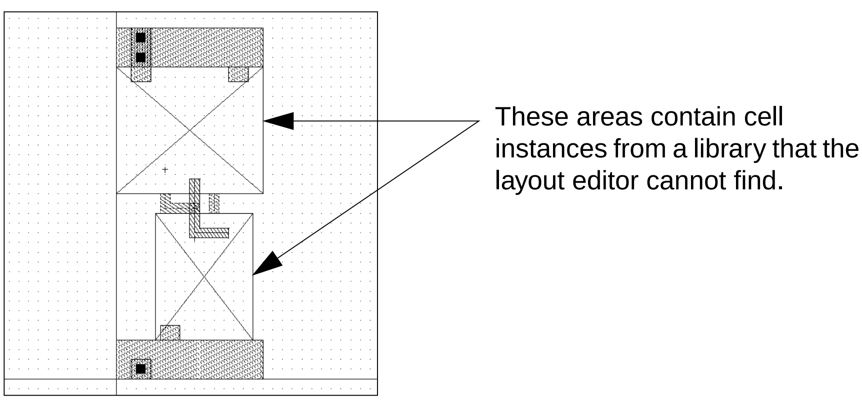
➤ Do one of the following:

- Change the access permissions using the *Library Manager – Edit – Access Permissions form*.
- Use the UNIX command `chmod` to change the access permissions in the UNIX directory containing the library.

Cell Instances Are Missing

A cellview often contains instances of cells from other design libraries. If you open a cellview that contains instances of cells from a library that the layout editor cannot find, the following happens:

- When you try to open the cellview, you see a warning dialog box listing cells that the layout editor cannot find
- When you close the dialog box, the cellview opens, but each area containing a missing cell displays a flashing box with an X



To include the missing cells,

➤ Add the path to the library containing the cell masters to the `cds.lib` file.

Problems with Editing and Entering Points

I Made a Mistake Entering Points

If you enter a point incorrectly, you can delete it.

To delete the last point entered,

1. Press `Backspace`.

   The last point you entered is deleted.

2. Click to enter a new point.

Click to enter a point. | Press `Backspace`. | The point is deleted. | Click to enter a new point.

If you finish creating an object and then find it is incorrect, remove the object by undoing the last command (choose *Edit – Undo*).

I Entered Points on the Wrong Layer

You create objects on layers that represent the layers of your physical design. The Layer Selection Window (LSW) controls which layer the object will be created on.

➤ If you notice you are creating objects on the wrong layer, click on the layer you want in the LSW.

  The current entry layer changes to the one you chose.

If you finish creating an object and notice it is on the wrong layer, change the layer for that object by doing either of the following:

- To move the object to a new layer, use the *Move* command.
- To change the object's layer property, use the *Properties* command.

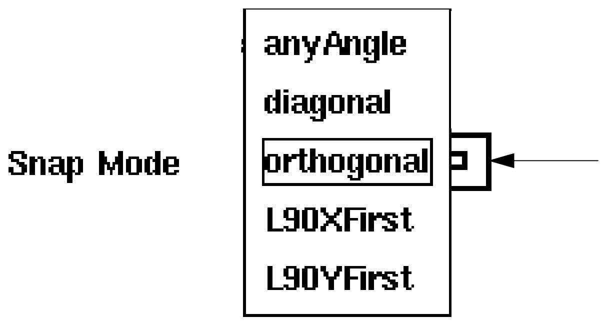
I Can’t Move Objects or Enter Points Where I Want

You might need to change the snap mode if you see the following behavior:

- You cannot enter points to create or recreate objects
- You cannot move or stretch an object

To change this behavior, change the snap mode by doing the following:

1. Open the pop-up options form by double-clicking middle, or press F3.
2. Choose the snap mode you want.



The *Snap Mode* field appears on forms for commands that let you create or recreate shapes.

To change the create and edit snap mode defaults,

1. Choose *Options – Display*.

   The Display Options form appears.
2. Choose a new value for the *Create* or *Edit Snap Modes*.
3. Click *OK*.

Object Moves Instead of Stretches

If an object moves when you expected it to stretch, you probably selected the whole object rather than its edge or corner. To stretch an object, you must select only the edge or corner you want to stretch.

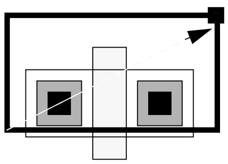
If you want to select edges or corners (vertexes), you must first set partial selection on. Partial selection lets you select edges or corners with the selection box.

To toggle partial selection on or off,

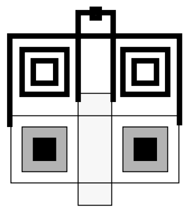
➤ Press F4.

When partial selection is on, the window banner shows a P in front of the number of selected objects.

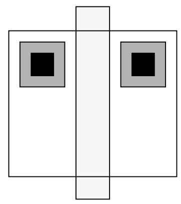
```
(P) Select: 1
```



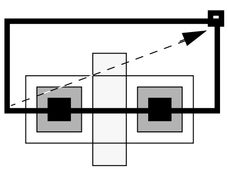
If you select whole objects along with edges...



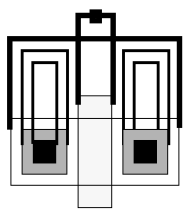
...the whole objects move.



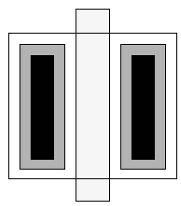
The stretched and moved objects



If you select only partial objects (edges or corners)...



...all objects stretch.

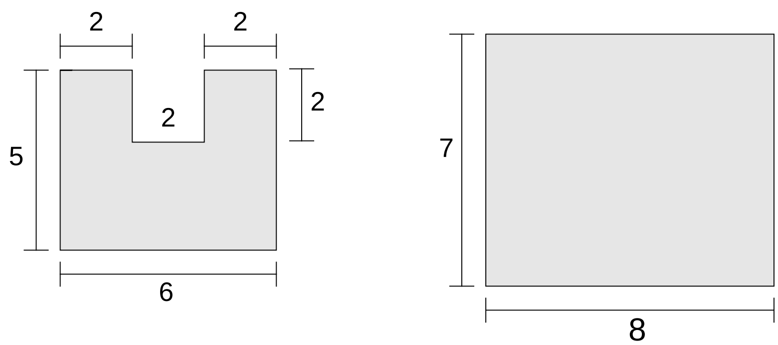


The stretched objects

Object Changes Shape When Resizing

The *Size* command and the *Grow By* option for the *Layer Generation* command each let you resize an object by stretching its corners. Sometimes enlarging or reducing the object causes it to change shape.

For example, if you use *Size* to enlarge the following polygon by 1 unit, each vertex stretches in both the X and Y direction by 1 unit, and the notch in the polygon disappears. The resulting shape is stored as a rectangle, rather than a polygon.



Original shape

After using *Size* to enlarge the shape by 1 unit

To cancel the changes you made by resizing a shape,

➤ Immediately choose *Edit – Undo*.

I Can’t Paste an Object

If you use *Yank* and *Paste* to cut through the hierarchy and place parts of an instance into a different library, the new library must define all the layers you yanked. If it does not, you see the message

```
Shapes on invalid or invisible layers not pasted.
```

The *Yank* command lets you copy an area of the cellview, cutting through layers of hierarchy. You can select just part of an instance to copy with *Yank*, then use *Paste* to place the parts of the instance you copied into any cellview.

Pcell Parameters Do Not Appear

The parameters for parameterized cells (pcells) appear in the Create Instance form as you place an instance of the pcell.

When you type the name of a cell into the Create Instance form, you must tell the system you are done before the pcell parameters will appear.

➤ If pcell parameters do not appear in the Create Instance form, press `Tab` to tell the system you are done typing the cell name.

If the form is at the bottom of your screen, the parameter section can scroll off the screen.

➤ If the Create Instance form is at the bottom of your screen, move the form up to see the parameters.

If you use Component Description Format (CDF) C-level function parameters for your library or for any cells in the library, and your pcell was compiled in version 4.2.1 or earlier, the CDF parameters suppress the pcell parameter display.

➤ To recompile pcells created in version 4.2.1 or earlier, choose *Pcell – Compile – To Pcell.*

Mouse and Cursor Behavior

Right Mouse Button Doesn't Work

By default, the right mouse button works as follows:

- To repeat the last command, click right once
- To zoom in, press and hold right and create a box
- To zoom out, press the `Shift` key and hold right and create a box
- To change options while using some editing commands, press and hold right

If the right mouse button will not do any of these tasks, it is probably set to create strokes. A stroke is a preprogrammed figure you can create to start a command.

➤ To cancel the stroke-creation capability, you must exit and restart the Cadence® software.

If strokes are still on when you restart, then the commands to load strokes are included in your `.cdsinit` file.

To remove the stroke commands from your `.cdsinit` file,

1. Use a text editor such as `vi` to open your `.cdsinit` file (usually located in your current or working directory).
2. Look for these lines in your `.cdsinit` file:

```
load(prependInstallPath( "etc/sted/stroke.il"))
load(prependInstallPath( "etc/sted/defstrokes.il"))
hiLoadStrokeFile("def.strokes" "Layout")
```

3. Type a semicolon (`;`) in front of each line to comment it out.

4. Save the edited `.cdsinit` file.
5. Exit and restart the Cadence software.

Pointer Changes Shape

The mouse pointer changes shape to show you how you can use it to select, move, or stretch objects. If you see the pointer change shape, this is not an error.

Cursor Sticks to Objects

If the mouse cursor seems to jump to or stick to objects in your cellview, the gravity setting is most likely on.

Gravity causes the mouse cursor to automatically snap to objects and grid points in the cellview. You typically use gravity when you want to snap the cursor to specific types of objects as you create them.

To turn gravity off,

➤ Move the mouse into a layout window and press `g`.

  If this does not work, your system administrator changed the default key bindings. Use the Layout Editor Options form to set gravity.

Cursor Snaps to the Wrong Object

If you want to use gravity but the cursor snaps to the wrong object, you might need to change the gravity settings. Use the Layout Editor Options form to change the gravity settings.

Cursor Doesn’t Snap to a Point

If gravity is set off and the cursor does not snap to the grid point you want, the snap spacing might be set too wide. The X and Y snap spacing settings control the number of units at which the cursor snaps to the grid. Use the Display Options form to change the snap spacing settings.

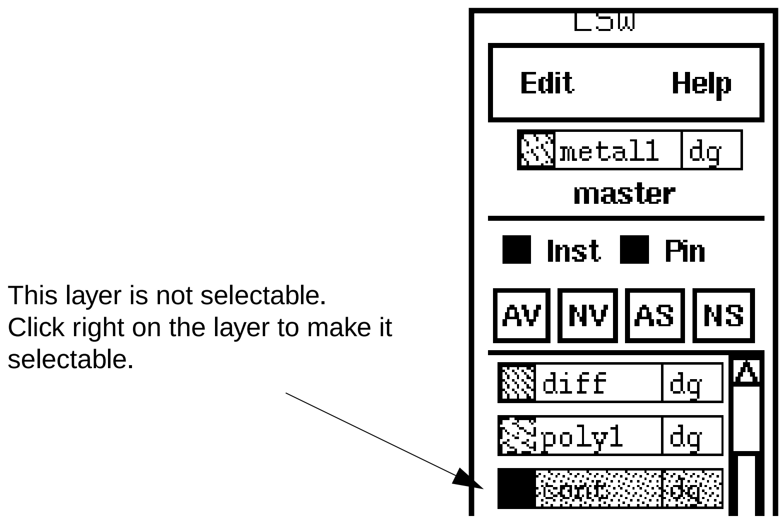
Selection Tips

How to Make a Layer Selectable

In the Layer Selection Window (LSW), you can turn off selection for one or more layers. When you do, a layer that is not selectable appears with its name shaded in the LSW.

To make a layer selectable,

➤ Click right on the layer name in the LSW.

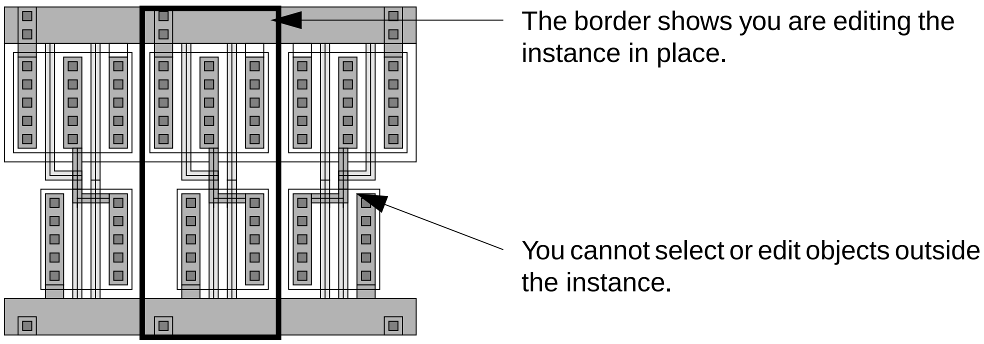


The only objects you can select in a cellview are on selectable layers in the LSW.

Note: To add a layer to the LSW, choose *Edit – Set Valid Layers* in the LSW. If you are using the leLswLayers section of the Technology File, only the layers listed in that file appear in the Set Valid Layers form.

Selection During Edit-In-Place

If you are editing a cell instance in place, you can select and edit only those objects within the instance. Even though you can see objects around the instance, you cannot edit them.



- To check to see if you are editing in place, look at the cellview banner to see what instance you are editing.

  If the name in the banner does not match the name of the current cellview, then you are editing a cell in place.

- To return to editing the current cellview, choose *Design – Hierarchy – Return*.

How to Select Objects in a Dense Design

If you click on objects in a dense design and the layout editor does not select the object you want, try any of the following:

- If objects share the edge you chose, click again in the same place.

  The layout editor selects another of the objects. (The layout editor toggles between only two possible objects.)

- If possible, move the cursor closer to another edge of the object and click.
- If possible, zoom in on the edge you want to select.

  Use the *Window – Utilities – Previous View* command to zoom back out again after selecting the object.

- Use the Layer Selection Window (LSW) to turn off selectability for
  - Layers that overlap the object

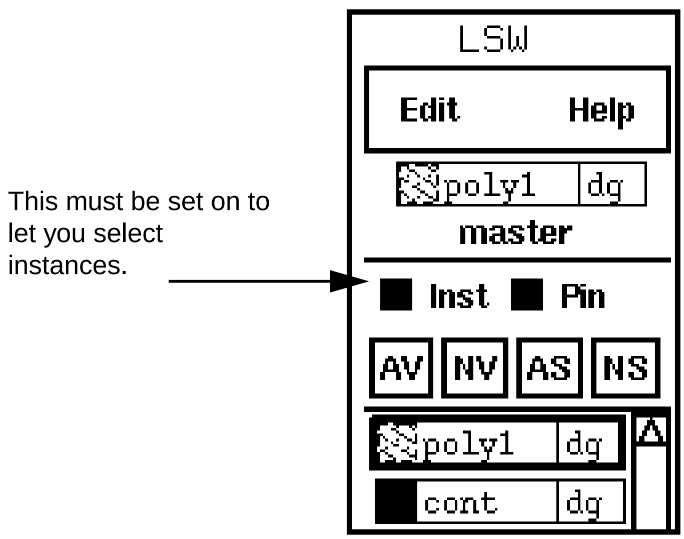
- ❑ Pins or instances (if a pin or instance overlaps the object)

Instance Selectability

If you cannot select an instance, check to see if *Inst* in the Layer Selection Window (LSW) is turned off.

If *Inst* is turned off,

➤ Click to turn it on.



Problems Selecting Copied Objects

If you move or copy an object into a cellview in a different library and then cannot select the object,

1. Click in the new cellview.
2. Look for the name of the layer for the object in the Layer Selection Window (LSW).
   - ❑ If the layer name is unselectable (shaded), click right to make it selectable.
   - ❑ If the layer name does not appear in the LSW, the layer either is not defined for this library or has been removed from the LSW.

If the layer is not defined for the library, cancel the copy by doing one of the following:

➤ Choose *Edit – Undo* in the original cellview.

➤ Choose *Design – Discard Edits* in the new cellview to delete all edits you made since you last saved the cellview.

If the layer has been removed from the LSW, replace it by choosing
Edit – Set Valid Layers.

Mouse Buttons for Zooming and Selecting

If you create a selection rectangle and the image in the cellview is enlarged or reduced, you drew the rectangle using the right mouse button instead of the left mouse button. Dragging right starts *Zoom In* and *Zoom Out* commands.

- Creating a rectangle with the right mouse button enlarges the cellview image.
- Creating a rectangle by pressing `Shift` with the right mouse button reduces the cellview image.

If you inadvertently zoom in or out instead of selecting objects,

1. Choose *Previous* from the *Window – Utilities – Previous View* menu to redisplay the cellview as it appeared before you zoomed in or out.
2. Click and drag to create a rectangle to select objects.

How to Select Invisible Objects

If a selection highlight outlines an area of your cellview that appears blank, you probably turned off the visibility of all layers that appear in an instance and selected that invisible instance.

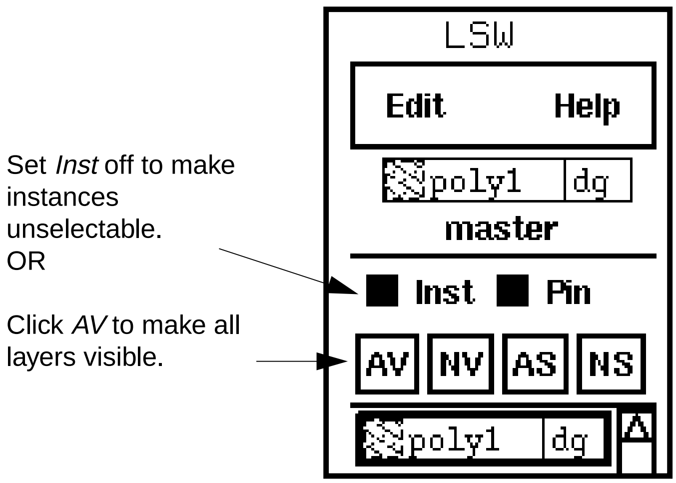
For example, if you clicked on *NV* in the Layer Selection Window (LSW) to set all layers invisible, but *Inst* in the LSW is still on, you can select the invisible instances.

To select invisible objects,

➤ Do one of the following:

  ❑ Set *Inst* off in the LSW so that instances are not selectable.

- Click *AV* in the LSW to make all layers visible.



Window Display Tips

How to Control What Appears in a Window

You can control any of the following:

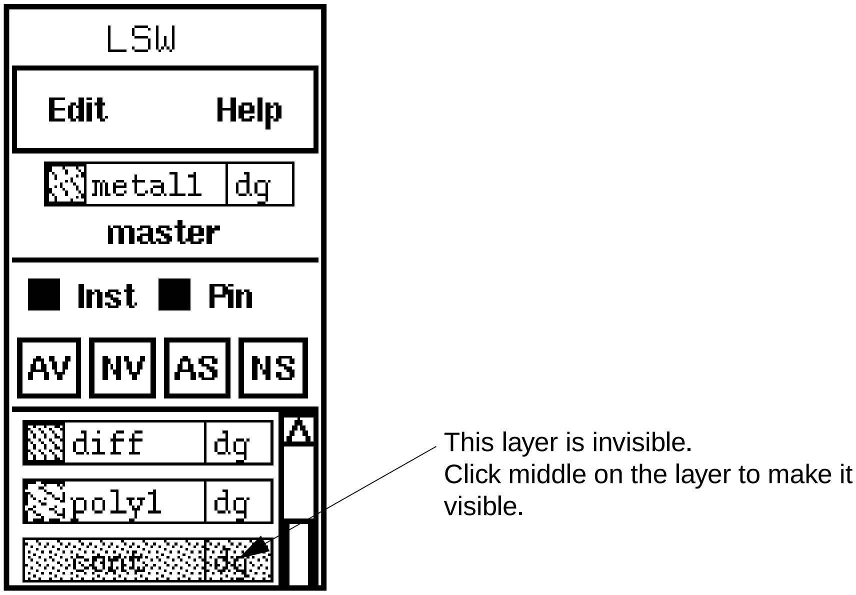
- Where the layout design window appears
- Where forms appear on the screen
- The appearance of objects in a window
- Whether the cellview grid appears
- Whether cell or array instance details appear
- The appearance of scroll bars, prompts, or icons

How to Make a Design Layer Visible

If you cannot see one of your design layers, it is probably set to be invisible. A layer that is invisible appears with its layer color shaded in the Layer Selection Window (LSW).

To make a layer visible,

1. Click middle on a shaded layer.



2. Choose *Window – Redraw* to see the results of your changes.

How to See Instance Contents

If you see only an outline of each instance instead of the details of objects inside the instance, change the display level settings.

To turn on instance detail,

➤ Press `Shift-f`.

   `Shift-f` is the bindkey equivalent of using the Display Options form to set the display levels to 0 through 32.

Displaying Instances You Are Editing

If you are editing a cell in place, you can set the window to display only the instance you are editing.

To see the rest of the instances,

1. Choose *Options – Display.*

   The Display Options form appears.

2. Set *EIP Surround* on.
3. Click *OK*.

How to See What the Search Command Finds

If you use the *Search* command to search for or select objects and your cellview is zoomed in, *Search* might find an object that is not visible in the current window.

To zoom the cellview so the objects *Search* finds appear in the window,

➤ In the Search form, set *Zoom To Figure* on.

Now each time you click *Previous* or *Next*, the cellview is redrawn so that the object is centered in the window.

How to See Selection Highlights

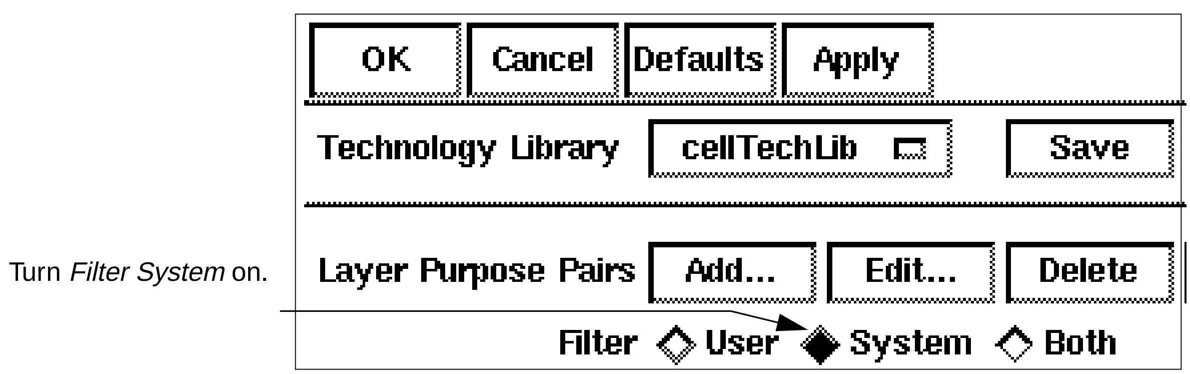
The layout editor highlights objects when you select them. Selection highlights are created using a special system layer. If you cannot see the highlight, the highlight layer visibility might be turned off.

To turn on the highlight layer visibility,

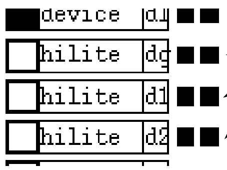
1. Choose *Technology File – Edit Layers* in the Command Interpreter Window (CIW).

   The Edit Layers form appears.

2. Set *Filter System* on.



3. Click to fill in the visibility field for the layers *hilite dg*, *hilite d1*, and *hilite d2*.



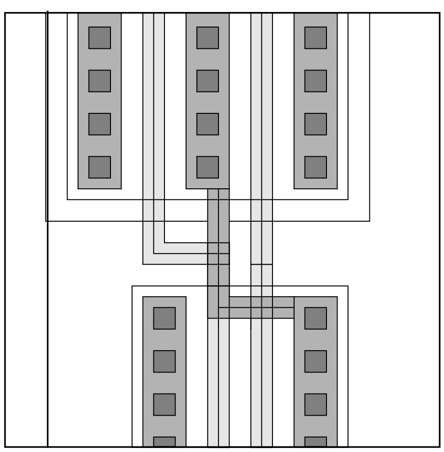
Make sure the visibility (v) buttons for these layers are set on.

4. Click *OK*.

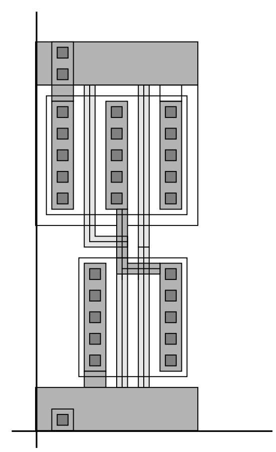
Plotting Problems

Plot Shows More than the Window

In the Submit Plot form, if you set *Plot* to *Cellview*, all data in your cellview is plotted. Even if your cellview window is zoomed in so you see only a portion of the cellview, all data is plotted.



The cellview shows only one area of the design.



Plot Cellview plots the entire design.

To plot only what is displayed in the window,

➤ Set *Plot* to *Viewing Area* in the Submit Plot form.

To plot a selected portion of the design,

➤ Set *Plot Area* to *Select* in the Submit Plot form and select the area to plot.

No Plot Form Appears

If you try to plot and the Submit Plot form does not appear, one of the following might be the cause:

- If there is no data in your cellview, the *Plot* command does not work. The *Plot* command cannot plot an empty cellview.
- If you have set *Plot Area* to *Select*, you must follow the instructions in the Command Interpreter Window (CIW) to create a rectangle around the area to plot.
- If you do not have a `.cdsplotinit` file, or your `.cdsplotinit` file does not list any plotters, the *Plot* command does not work. In this case, you also see a message in the CIW similar to one of the following when you try to plot:

```
*WARNING* Plotter configuration not loaded
*WARNING* No Plotters is not a valid plotter
*WARNING* There were no system .cdsplotinit files found.
```

To configure your system so the plot utility can run correctly, you must follow the instructions in the *Plotter Configuration User Guide.*

Plotter I Want Is Not in the Plot Form

The `.cdsplotinit` file controls which plotters appear in the Submit Plot form.

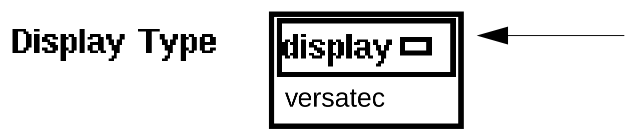
To add a plotter,

➤ Edit the `.cdsplotinit` file to include your plotter.

Scroll through the sample file in *`your_install_dir/`*`tools/plot/samples/cdsplotinit.sample` for more details about what to put in the `.cdsplotinit` file.

No Plotter Display Devices Appear in the Display Options Form

In addition to choosing a plotter from the Plot Options form, you also choose a display type. Each display type corresponds to definitions in your technology file for the colors, stipple patterns, and line styles available for plotters and for your monitor screen.



Display Type lists output devices: display monitor and plotters.

If your technology file defines only a monitor display device (*display*), only that device appears in the Submit Plot form.

If you want your plot to use the same stipples and colors that are displayed on screen, choose *display* in the *Display Type* field.

If you want your plot to use different stipples or colors, you must add a device description for your plotter in the technology file.

The sample technology file, `your_install_dir/tools/plot/samples/techfile/mpu.tf`, lists sample entries for a number of printers or plotters.

An Error Message Appears in the Plot Queue Status Form

The Queue Status form lets you look at the jobs you sent to different plotters. The message area of this form displays messages from whatever UNIX commands you use to send plots to the queue.

You see error messages if the `.cdsplotinit` file does not

- List the correct UNIX spooling commands for each plotter
- List plotter names in the UNIX spooling commands that match the plotters in your UNIX plot support files (such as `/etc/printers.conf`)
- List correct information about your plotter, such as pen numbers

For example, a `.cdsplotinit` file includes a line setting the BSD UNIX `spool` command for a Versatec plotter:

```
lpr -Pvt2
```

If there is no vt2 printer identified in your `/etc/printers.conf` file, the following error message appears in the Queue Status form when you choose a Versatec plotter:

```
lpq: vt2: unknown printer
```

If you see error messages in the Queue Status form,

1. Use the form to remove the job with the error from the queue.
2. Do any of the following:
   - Check your `.cdsplotinit` file to make sure the entries are correct.
   - Check your UNIX plot support files, such as `/etc/printers.conf`, to make sure they list the printers in the `.cdsplotinit` file.
   - Check your system executables to make sure your system has the UNIX spooling commands listed in the `.cdsplotinit` file.

See the *Plotter Configuration User Guide* for details about other UNIX support files required for plotting.

15

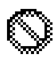
Dialog Box Messages

This chapter contains these topics:

About Messages

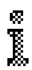
The Virtuoso® layout editor displays error and other messages in dialog boxes. A dialog box is a Cadence® window that displays a warning or other information about the current command.

There are four types of messages:

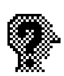


Error messages display a international *No* symbol and indicate the layout editor cannot complete a command you have started.

Warning messages display an exclamation point and warn you of a result you might not have anticipated.



Informational messages display an *i* symbol and provide information about a command's progress.



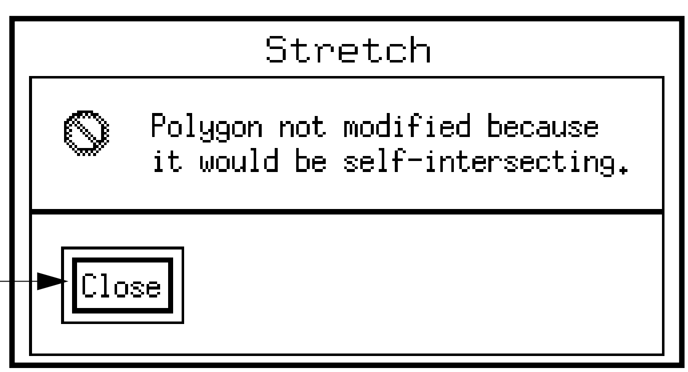
Questions display a profile of a head and ask you to make a choice.

Closing Dialog Boxes

Most information, error, and warning boxes have one button, *Close*.

You must click *Close* to close the dialog box before you can continue editing.

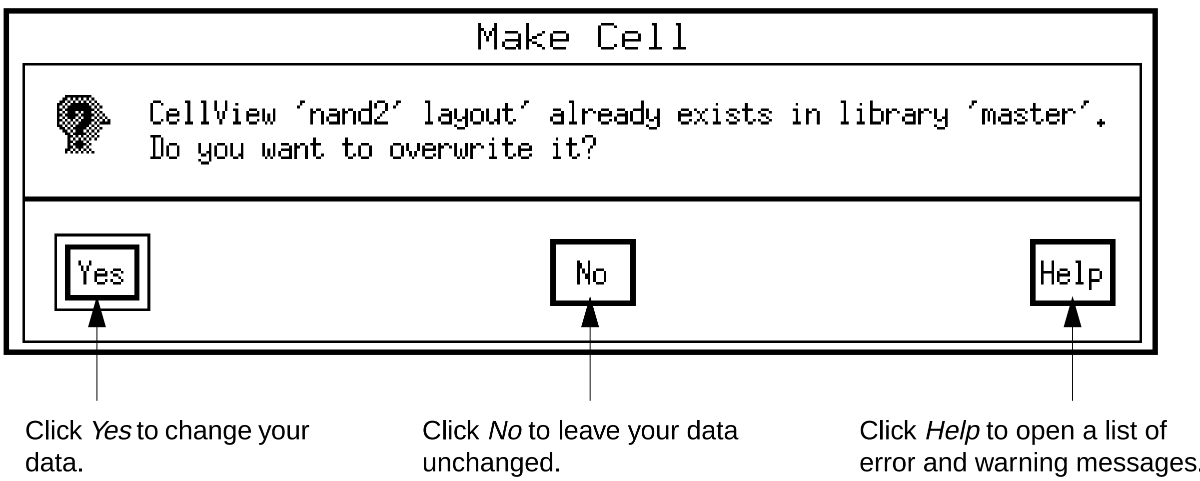
Click here to close a dialog box.



Using Question Dialog Boxes

Question dialog boxes let you choose, for example, between changing data or leaving it unchanged.

- To change data, click *Yes*.
- To leave data unchanged, click *No*.
- To see the list of error and warning messages, click *Help*.



List of Error Messages

The following list shows the error and warning messages you might see while using the layout editor.

The messages are listed alphabetically. Several messages begin with variables (a cellview name you typed, or the number of shapes you were editing). Those messages are listed first.

If you can not find the error message you are looking for, you can contact the Cadence hotline at 1-877-CDS-4911. The hotline is available for customers on maintenance.

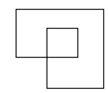
Error Message	Explanation and Solution
"cellname viewname" already exists in library *"name"*. Do you want to overwrite it?	You tried to save this cellview with a name that exists. ■ Click *Yes* to write over the existing file. ■ Click *No* to cancel the *Save As* command.
"cellname viewname" is being edited in a window. Do you want to overwrite it?	You tried to create a cellview with the same library, cell, and view name as one that is open for editing. ■ Click *Yes* to write over the existing file. ■ Click *No* to cancel the *Create File* command.
"cellname" is not a valid cell name.	The cell name field in the form is blank. Click *Browse* in the form to use the Library Browser and select the correct cell name. To create a new cell, choose *File – New – Cellview* in the Command Interpreter Window (CIW).
"libraryname cellname viewname filename" already exists. Do you want to overwrite it?	You tried to create a cellview with a name that exists. ■ Click *Yes* to write over the existing file. ■ Click *No* to cancel the *Create File* command.
"*Number*" shapes not modified because they would become malformed.	When editing a group of shapes, you tried to delete points or edges that would cause the shapes to become invalid. For example, a path must have at least two points, and a polygon must have at least three points.
"*Number*" shapes not modified because they would be illegal.	You stretched a shape to a zero area, which would delete it. You cannot use *Stretch* to delete shapes.

Error Message	Explanation and Solution
"Shapenames" shapes not modified because they would be self-intersecting.	You stretched the shape so it overlaps itself.
"Shapenames" shapes were not modified because they would become malformed.	You tried to delete part of an object. To delete an object, select the entire object.
Apply the command first to find figures that meet the current search criteria.	You clicked *Select*, *Select All*, *Replace*, or *Replace All* in the Search form before clicking *Apply*. You must click *Apply* to search for objects first.
Cannot open library *"name"* for writing.	You do not have permission to write to the library name you typed in the Save As form. ■ Type a different library name. ■ Have the owner of the library change the permissions.
Cannot create a pin because the path for the pin is non-orthogonal.	The *autopin* option for *Pin* only works on path ends parallel to the X or Y axis.
Cannot create a recursive instance placement.	You tried to use *Create – Instance* to place an instance of the master cell you are editing in this cellview into this cellview, or you tried to use *Make Cell* to create and place a new cell with the same name as the current cellview.
Cannot create view file "*name*."	The *Show Selected Set*, *Summary*, and *Tree* commands create and open a temporary file in `/tmp` that contains the data displayed by these commands. You must have write and read access to `/tmp` for these commands to work. See your system administrator or become root and use `chmod` to set the `/tmp` directory permissions.
Cannot descend into a techfile device.	You tried to descend into a device that is defined in the technology file. To edit this device, you must edit its definition in the technology file.

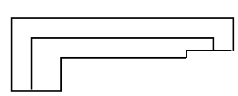
Error Message	Explanation and Solution
Cannot descend into instance "*name*" because it is unbound.	The library for the instance you chose with *Descend* ■ Has been deleted ■ Is not in your library path; add it to your `cds.lib` file
Cannot edit-in-place a parameterized cell.	You chose a parameterized cell (pcell) to edit with *Edit In Place*. You cannot edit pcell instances in place because a pcell instance is not identical to the pcell master. You can edit the pcell parameters for the instance using the *Edit – Properties* command.
Cannot modify figure, because cellView not open with write permission.	You tried to use the *Replace* or *Replace All* buttons in the Search form to edit objects you found with *Search*, but the objects are in cell instances in the current cellview. To open an instance to edit, use *Edit In Place*.
Cannot open cellView "*cellname viewname*" in library "*libName*"	You do not have write permission to either the cellview or the library you typed in the Make Cell form. You need to change the permissions for the cells or libraries.
Cannot open file "*name*" for read. Cannot open file "*name*" for write.	The *Summary* and *Show Selected Set* commands create and open a temporary file in `/tmp` that contains the data displayed by these commands. You must have write and read access to `/tmp` for these commands to work. See your system administrator or become root and use `chmod` to set the `/tmp` directory permissions.

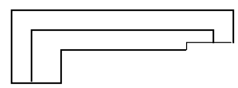
Error Message	**Explanation and Solution**
Cannot open file “*name*” for reading.	You tried to save the settings in the Display Options, Layout Editor Options, or Submit Plot form to an ASCII file, but you do not have write access to the directory or file. ■ Type a new filename or include a path to a directory to which you have write access. By default, the layout editor tries to save the file to your current directory.
Cannot select figure that is not in current cellView.	You set the *Search* command to search for objects inside instances in the current cellview, then tried to use *Search* to select an object found in an instance. Search cannot select objects inside instances. To open an instance and select an object inside it, use *Edit In Place*.
Cannot set the entry layer to be invisible.	You clicked middle to try to set the current entry layer in the LSW to be invisible. The current entry layer must be visible. If you want to set the current entry layer to be invisible, first make a different layer the current entry layer.
Cannot set an invisible layer to be the entry layer.	You clicked on a layer in the LSW that is set to be invisible. If you want this layer to be the entry layer, first click middle on the layer to make it visible and then click.
Cannot set view file “*name*.”	The *Summary*, *Show Selected Set*, and *Tree* commands create and open a temporary file in `/tmp` that contains the data displayed by these commands. You must have write and read access to `/tmp` for these commands to work. See your system administrator or become root and use `chmod` to set the `/tmp` directory permissions.

Error Message	Explanation and Solution
Cannot write to file *"name"*.	The filename you typed in the Submit Plot form is incorrect for one of the following reasons: ■ The directory does not exist. ■ You do not have permission to write to the directory. ■ You do not have permission to write to the file.
Cell "*name*" does not exist in library.	The cell name you typed in the Create Instance form is incorrect. Click *Browse* in the form and use the Library Browser to select a different library or cell name.
Cellview "*cellname viewname"* already exists in library *"library name*." Do you want to overwrite it?	While using *Make Cell*, you typed the name of a cell and view that already exists in the library you chose. ■ Click *Yes* to overwrite the original cell with the new objects you have selected with *Make Cell*. You can choose this only if you have write access to the original cell. ■ Click *No* to avoid overwriting the cell. Type a new cell name in the Make Cell form.
Cellview "*cellname viewname*" does not exist in library.	The cell name you typed in the Create Instance form is incorrect. Click *Browse* in the form and use the Library Browser to choose a different library or cell name.
CellView is already read only.	You tried to set the mode to read only when the mode was already read only.
CellView is already editable.	You tried to set the mode to edit when the mode was already edit.
Chopper polygon area cannot be zero.	You clicked and then double-clicked on the same point when creating the polygon to cut an object with *Chop*.

Error Message	Explanation and Solution
Chopper polygon is self-intersecting.	You drew a self-intersecting polygon to chop an object. This shape is invalid. It is a self-intersecting polygon. 
Chopper rectangle area cannot be zero.	You clicked the same point twice or drew a vertical or horizontal line when creating the rectangle to cut an object with *Chop*.
Circle not created because it would have zero area.	You clicked the same point twice when creating a circle with *Circle*.
Contact "*contact name*" does not exist in library "*library name*" using "*contact name*."	You used `envSetVal("`*`contactName`*`"` *`contactType`*`)` to set the default type of contact to place with *Contact*, but *`contactType`* (the name of a contact to place) is not defined for this library.
Could not save cellView *"libraryname cellname viewname reason"*.	You tried to save a cellview. The software could not for the reason stated.
Delete figure?	You tried to delete a selected object. ■ Click *Yes* to delete the figure. ■ Click *No* to cancel the *Delete* command.
Deleting the layer purpose pair "*layer* (*purpose*)" (or layer purpose pairs) might cause errors in other technology file classes that reference it (or them). Delete "*layer* (*purpose*)" (or the layer purpose pairs) anyway?	You selected one or more layer-purpose pairs in the Layer Purpose Pair Editor form and clicked on the *Delete* button. ■ Click *OK* to delete the selected layer-purpose pair or pairs. ■ Click *Cancel* to cancel the *Delete* command.
Delta X is less than width of master's bBox.	The value you typed for the *Delta X* field in the Create Instance form would cause the instances to overlap.

Error Message	Explanation and Solution
Delta Y is less than height of master's bBox.	The value you typed for the *Delta Y* field in the Create Instance form would cause the instances to overlap.
Descending into a parameterized cell. Is this ok?	You chose an instance of a parameterized cell (pcell) to edit with *Descend*. Because a pcell master does not always resemble the instance, this message appears to remind you that you are about to open a pcell master. ■ Click *OK* to descend into the master view of the pcell. ■ Click *Cancel* to stop *Descend*.
Do you want to save your changes?	You tried to change the mode to read only without writing your edits to disk. ■ Click *No* to discard your edits. ■ Click *Yes* to save your edits.
Donut not created because inner radius equals outer radius.	You clicked the same point twice when entering the last two points of a donut.
Donut not created because of zero inner radius.	You clicked the same point twice when entering the first two points of a donut.
Ellipse not created because it would have zero area.	You clicked the same point twice when creating the bounding box of an ellipse.
Error found while reading "*file line*."	There is a syntax error on the line in the file you tried to load using the *Load From* button in the Display Options, Layout Editor Options, or the *Load* button in the Submit Plot form. See the *Custom Layout SKILL Functions Reference* for details about Cadence SKILL language syntax.

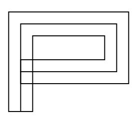
Error Message	Explanation and Solution
Figures cannot be modified, because you do not have write permission for this cellView.	You tried to use the *Replace* or *Replace All* buttons in the Search form to edit objects you found with Search, but the current cellview is open in read-only mode. Use *Design – Make Editable* to open the current cellview in edit mode.
Figures not in current cellView will not be selected.	You set the *Search* command to search for objects inside instances in this cellview, then tried to use *Search* to select objects it found in the instances. *Search* cannot select objects inside instances. To open the instances and select objects inside them, use *Edit In Place*.
File *"name"* already exists. Do you want to overwrite it?	You tried to save this cellview with a name that exists. ■ Click *Yes* to write over the existing file. ■ Click *No* to cancel the *Save* command.
Find failed. *"reason"*	The *Markers – Find* command could not find any markers for the reason stated.
First or last segment of created path has length less than or equal to half the path width.	This usually happens when you complete a path and one of the path segments is smaller than half of the path width. You also can see this message when you reshape a path and the result is one of the path segments becomes smaller than half of the path width. 

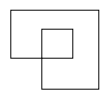
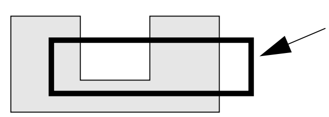
Error Message	Explanation and Solution
First or last segment of modified path has length less than or equal to half the path width. Use “Cancel” to interrupt the command.	Path modification was incorrect. This usually happens when you stretch one or more paths and the stretch causes one or more of the path segments to be smaller than half of the path width.  ■ Click *OK* to close the dialog box. If you have more than one incorrect path segment, a dialog box opens for each one. After you close one dialog box, another one opens for the next incorrect path segment. You can click OK until all of the dialog boxes are closed. ■ Click *Cancel* to close the dialog box and interrupt the command. Nothing more will be stretched, and the warning process stops.
hiToggleEnterForm(), normally bound to the F3 key, will show or hide the form associated with an active enterfunction if that enterfunction has an associated form. Do you wish to see this dialog again when no enterfunction is active or no form exists to toggle?	You pressed the F3 key expecting to see the option form for the command running, but an option form does not exist for the command. ■ Click *Yes* and the box appears every time you press F3 for commands that do not have option forms. ■ Click *No* and the box does not appear again during the editing session.
Instance name *“name*” contains a syntax error.	The instance name you typed in the Create Instance form is not allowed. The instance name syntax, a form of BNF (Backus Naur Format) notation, is `instName ::= baseName [ < number [ : number ] > ]`

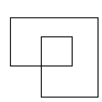
Error Message	Explanation and Solution
Instance base name "*base name*" is not unique in this block. Instance member name "*member name*" is not unique in this block. Instance name "*instance name*" is not unique in this block.	The base, member, or instance name you typed in the Create Instance form is already used in this design cellview. See *"Instance name 'name' contains a syntax error"* for the instance name syntax.
Invalid window: *"windowID"*.	The window ID is incorrect.
Layout editing capability is not enabled.	You do not have the Cadence license for reading and viewing or changing and saving layout views (the `leEdit` or `le` license, 300 or 11400).
Layout must be opened with write permission.	You cannot use the commands on the *Create* or *Edit* menus if the layout cellview is opened in read-only mode. Use *Design – Make Editable* to open the cellview in edit mode.
Library "*library name*" does not exist.	The library name you typed in the Create Instance or Make Cell form is incorrect. ■ Click *Browse* in the form and use the Library Browser to select the correct library. ■ Correct your library path.
Modifying "object" throughout hierarchy. Is this okay?	You tried to replace a value for an object you found with *Search*, and the object is inside an instance in the current cellview. Any edits you make to the instance in this cellview also affect the master cell for the instance. ■ Click *Yes* to replace the object through the hierarchy. ■ Click *No* to cancel replacing the object.
Mosaicl/Donut/Dot is not supported in the Rotate command. Please de-select it.	You tried to use the *Rotate* command on either a mosaic, donut, or dot pin. To rotate these objects, use the *Move* command.

Error Message	Explanation and Solution
No cellviews were refreshed.	No masters of cell instances in the current cellview have been edited and saved since you opened the current cellview, so the *Refresh* command did not update any of the instances.
No contacts defined in library "*name*."	You tried to place contacts with *Contact* or search for contacts with *Search*, but no contacts are defined in the technology file for this library.
No current window exists.	You tried to use a command that needs a graphic editor window.
No figures found that meet the current search criteria.	*Search* could not find any objects in this cellview that match the criteria you set in the Search form.
No figures selected.	You tried to use the *Make Cell* command without selecting the objects you want to make into a new cell.
No instances selected to be flattened.	The objects you chose to flatten are not cell instances.
No markers match the search criteria.	No markers were found as specified. Check your search criteria.
No modifications have been made.	You chose *Discard Edits* to delete any edits you made since the last time you saved the current cellview, but you have not edited this cellview since the last *Save*.
No more figures.	While using *Search*, you clicked *Previous* when there is no previous object in the group of objects *Search* found or you clicked *Next* when there is no succeeding object in the group of objects *Search* found.
No objects selected.	You chose the *Show Selected Set* command without first selecting a group of objects.

Error Message	Explanation and Solution
No shapes merged.	You did not select two or more overlapping shapes on the same layer to merge with *Merge*. Continue clicking on shapes to select the objects you want to merge.
No shapes selected.	You tried to use the *Layer Generation* command without selecting any objects on the layers you want to edit.
No shapes selected to be sized.	You chose the *Size* command without first selecting the objects you want to resize.
No shapes selected were modified.	You tried to modify a corner, but there were no polygon or rectangle corners selected.
No views saved for this window.	You tried to restore a saved image for this window, but there are no saved images. Use the *Window – Utilities – Save View* command to save an image.
Not a layout window.	You typed the SKILL version of a layout editor command in the CIW, but your current design window is not a layout view. Try clicking in the layout window first. If there is no layout view created for this cell, choose *File – New – Cellview* in the CIW to create a layout view (view name layout).
Nothing to paste.	You tried to use *Paste* without first using *Yank* to copy the objects you want to paste.
Only shapes can be sized.	You chose an instance or mosaic to resize with *Size*, but *Size* can only edit shapes. You can change the size of a selected instance or mosaic by using *Properties* and editing the *Magnification* field.

Error Message	Explanation and Solution
Open read-only cells for edit to replace *"cellname"* ?	The *Search* and *Replace* command found something in a cell that is opened in read-only mode. ■ Click *Yes* to open the cell for editing. ■ Click *No* to skip this replacement.
Shape is already on net "*name*."	The shape you tried to add to a certain net is already on a net. Shapes can be added to only one net.
Path not created because it would be self-intersecting.	You tried to create a self-intersecting path with *Path* or *Stretch*. Paths are self-intersecting if any edge of the path overlaps another edge of the path.  A self-intersecting path
Path width is greater than maximum width.	You typed a width value in the Path form that is greater than the `maxWidth` property defined for the layer in the technology file.
Path width is less than minimum width.	You typed a width value in the Path form that is less than the `minWidth` property defined for the layer in the technology file.
Pin not created because it would be self-intersecting.	Polygon pins follow the same rules as polygons. They cannot be created as self-intersecting shapes.
Please enter file name to save the display information change.	You said you wanted to save your changes before exiting. ■ Type a filename.
Please point at the reference point for rotation.	Click on the point you want to use as a pivot point for the rotation.

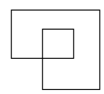
Error Message	Explanation and Solution
Polygon not created because it would be self-intersecting.	You tried to create a self-intersecting polygon with *Create Polygon* or *Stretch*. When possible, the *Create Polygon* command beeps and recreates the polygon so it is not self-intersecting; otherwise, the command returns the dialog box message.  This shape is invalid. It is a self-intersecting polygon.
Really discard edits?	You chose *Discard Edits* to discard all the edits you made since the last time you saved this cellview. ■ Click *Yes* to discard all edits you made since the last *Save*. ■ Click *No* to cancel the *Discard Edits* command and keep your edits.
Rectangle must intersect shape.	The rectangle you drew with *Reshape* does not intersect the object you selected to edit.
Rectangle must intersect shape at no more than two places.	The rectangle you drew with *Reshape* intersects the object you selected to edit in more than two places. ■ Close the dialog box, and create a new rectangle.  The reshaped rectangle intersects at more than two sides.
Rectangle not created because it would have zero area.	You clicked on the same point twice or drew a vertical or horizontal line when creating a rectangle.

Error Message	Explanation and Solution
Reference point is not active.	You pressed the – (minus) key on the keypad to make the reference point inactive and then tried to press `Shift-R11` (the 5 key on the keypad) to move the cursor back to the reference point. Check the status banner; if the *dX* and *dY* fields on the status banner show no value, the reference point is inactive. These fields normally show the delta distance along the X and Y axes between the current point and the reference point. ■ Press the + (plus) key on the keypad. The system prompts you to enter a reference point.
The Selector Parameter name “*name*” matches a parameter of the child Pcells.	You chose a selector parameter name that matches one of the child pcell parameters. Change the selector parameter name to one that does not match any of the child pcells.
Send Plot Only to File option must be set for \%s\ plot.	The plotter specified in the Plot Options form requires that the plot be sent to a file. ■ Select a different plotter or type a filename for the plot.
Send Plot to File requires an output file name.	You tried to plot to a file but did not specify a filename. ■ Type the filename in the Plot Options form.
Shape not created because it would be self-intersecting.	You tried to create a self-intersecting shape with *Reshape* or *Split.*  This shape is invalid. It is a self-intersecting polygon.

Error Message	Explanation and Solution
Shape not modified because it would become malformed.	You tried to delete a point or edge that would cause the shape to be invalid. For example, a path must have at least two points, and a polygon must have at least three points.
Shape(s) on invalid or invisible layer(s) not pasted.	You yanked objects on layers that ■ Are not in the technology file for this library ■ You have since set to be invisible in the Layer Selection Window (LSW) (click *AV* in the LSW to make all layers visible)
Shape width must be greater than zero.	You typed zero or a negative number for the *Width* value in the Path form. ■ Type a positive number in the *Width* field.
Some read-only cells could not be opened to replace *“cellname”*. Continue?	The software could not open read-only cells for editing.
Specify the library name.	The *library* name field in the form is blank. ■ Type a library name.
Terminal name “*name*” already exists with I/O type “*old I/O type*.” Do you want to change the I/O type to “*new I/O type*?”	You are trying to place a pin with a terminal name that matches the terminal name for one or more pins already placed in this cellview. The I/O (Input/Output) type for the other pin does not match the I/O type for the pin you are now placing. ■ Click *Yes* to change the I/O type for this pin and all other pins with the same terminal name. ■ Click *No* to keep the I/O type for this pin the same as the I/O type for all other pins with the same terminal name.

Error Message	Explanation and Solution
Terminal name "*name*" contains a syntax error.	The terminal name you typed in the Create (Symbolic or Shape) Pin form is not allowed. The terminal name syntax, a form of BNF (Backus Naur Format) notation, is `termName ::= baseName [ < number [ : number ] > ]`
The cellview for the command is no longer available.	The command received a bad cellview ID.
The cellView could not be opened for edit. Do you want to open it for read?	You tried to open a read-only cellview using the *File – Open* command. ■ Click *Yes* to open the cellview in read-only mode. ■ Click *No* to cancel the *File – Open* command. To open the cellview in edit mode, you must change the permissions in the directory.
The current cellView is open with read-only permission.	You tried to save the current settings in the Display Options form to this cellview, but you do not have write permission to the cellview. ■ Use the *Design – Make Editable* command to reopen the cellview in edit mode.
The design could not be reopened because the data on disk has been updated by others. Read in new data?	You tried to change the edit/read-only mode and the software found that some of the cellviews in the window do not have the current data. ■ Click *Yes* to update the cellviews. ■ Click *No* to continue with current data.
The following lists inconsistencies among techfiles used in this design: Top level library: *"techfilename"* Lower lever library: *"techfilename"* *"problem"* *"list"*	The cellview you are opening has conflicts between the top-level technology file and the lower-level technology file. ■ Edit the technology files listed to correct the listed problems.

Error Message	Explanation and Solution
The following masters are used in this cellview, but cannot be found in your search path: *“path”*	The masters could not be found in the path in your `cds.lib` file. ■ Edit the `cds.lib` file to include the missing master cellviews.
The range of the rotate angle value is [-360..360].	The value you typed in the Rotate form is not within the range. ■ Type a rotation angle between -360 and 360.
The rectangle area cannot be zero.	You clicked the same point twice or drew a vertical or horizontal line when creating a rectangle with *Reshape* or *Yank* or when creating a parameterized rectangle for a pcell instance.
This command is only available for layout views.	You typed the SKILL version of a layout editor command in the CIW, but your current design window is not a layout view. ■ If there is no layout view created for this cell, choose *File – New – Cellview* in the CIW to create a layout view (view name layout).
Unable to convert ellipse to polygon. The number of point would exceed the 2047 limit. Try a numPoints value less than 2048.	The value of *Conic Sides* in the Layout Editor Options form is more than 2047. Change the number to 2047 or less.
View “*name*” does not exist in library “*library name*.”	The view name you typed in the Make Cell form is incorrect. ■ Click *Browse* in the form to use the Library Browser and select the correct view name. Then choose *File – New – Cellview* in the CIW.
This is the first marker.	When viewing the first marker, you clicked the *Previous* button in the Find Markers form.
This is the last marker.	When viewing the last marker, you clicked the *Next* button in the Find Markers form.

Error Message	Explanation and Solution
Yank polygon area cannot be zero.	You clicked and then double-clicked on the same point when creating a polygon around objects you want to yank.
Yank polygon is self-intersecting.	You drew a self-intersecting polygon around the objects you want to yank.  This shape is invalid. It is a self-intersecting polygon.
Yank rectangle area cannot be zero.	You clicked the same point twice or drew a vertical or horizontal line when creating a rectangle around objects you want to yank.
You cannot save *“cellname viewname”* in library *“libraryname”*, because it is opened in “read only” mode.	You do not have permission to write to the library name you typed in the Save As form. ■ Type a different library name. ■ Have the owner of the library change the permissions.
You don’t have permission to write file *“filename”*.	You do not have permission to write to the filename you typed in the Save As form. ■ Type a different filename. ■ Have the owner of the file change the permissions.
You must specify a valid cell name.	You left the *Cell Name* field in the Create New File form blank. ■ Type a cell name.
You must specify a valid view name.	You left the *View Name* field in the Create New File form blank. ■ Type a view name.

16

Microwave Commands

This chapter contains these topics:

About Microwave Layouts

Microwave layouts consist of placed component instances such as resistors, capacitors, inductors, and transistors, and the interconnects drawn to connect the components.

You can connect components in a microwave layout with regular layout geometries such as polygons, rectangles, and paths, as well as microwave specific geometries: transmission lines, bends, and tapers.

You use the microwave commands to create transmission lines, bends, and tapers to connect your components. When you choose *Tools – Microwave* in your cellview, the microwave commands appear in the layout editor *Create* menu.

Editing Transmission Lines

You can edit microwave objects with the Virtuoso® layout editor commands similar to the way you edit paths. You use these commands to

- Change the angle of paths
- Stretch sections of paths
- Reshape a path
- Use the snap mode with Bend and Trls

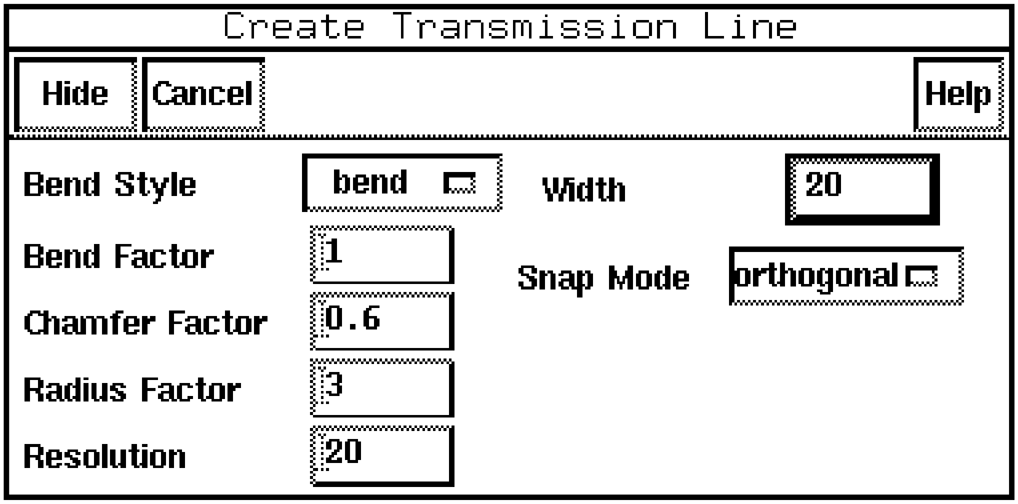
Creating Transmission Lines (Trl's)

The *Trl* command lets you create a transmission line.

To display the Create Transmission Line form,

➤ Choose *Create – Trl.*

About the Create Transmission Line Form



Bend Style sets the type of corner to use for the bends in the transmission line: bend, chamfer, or radial.

Bend Factor sets the maximum width allowed in a standard bend before the bend is automatically chamfered. (Available only when *Bend Style* is *bend*.)

Chamfer Factor sets the fraction of the corner that is removed in a chamfer bend. (Available only when *Bend Style* is *chamfer*.)

Radius Factor sets the ratio of the centerline bend width to the radial bend width. (Available only when *Bend Style* is *radius*.)

Resolution sets the number of segments used in a 180-degree radial bend.

Width specifies the transmission line width in user units (typically microns).

Snap Mode controls how transmission line segments snap to the grid.

Create Transmission Lines

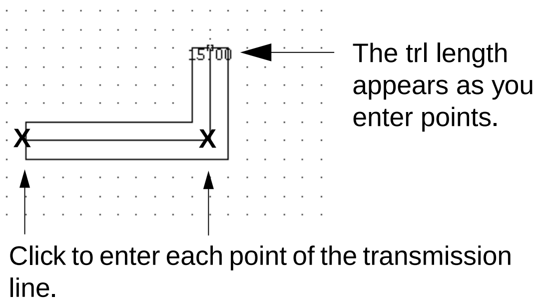
Transmission lines (trl's) show connections in microwave designs. You create trl's the same way you create paths.

To create a transmission line,

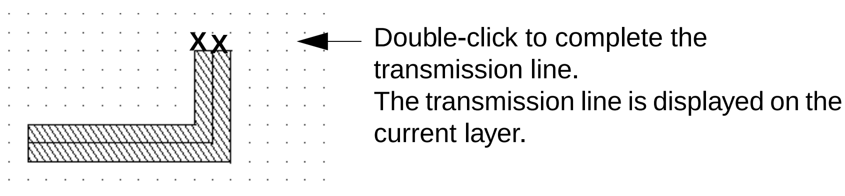
1. Choose *Create – Trl*.

The Create Transmission Line form appears.

2. Set *Bend Style* to *bend*.
3. Set *Bend Factor* to `1`.
4. Set the width to 2.
5. Click to create each point of the trl.



6. To complete the trl, double-click on the last point.



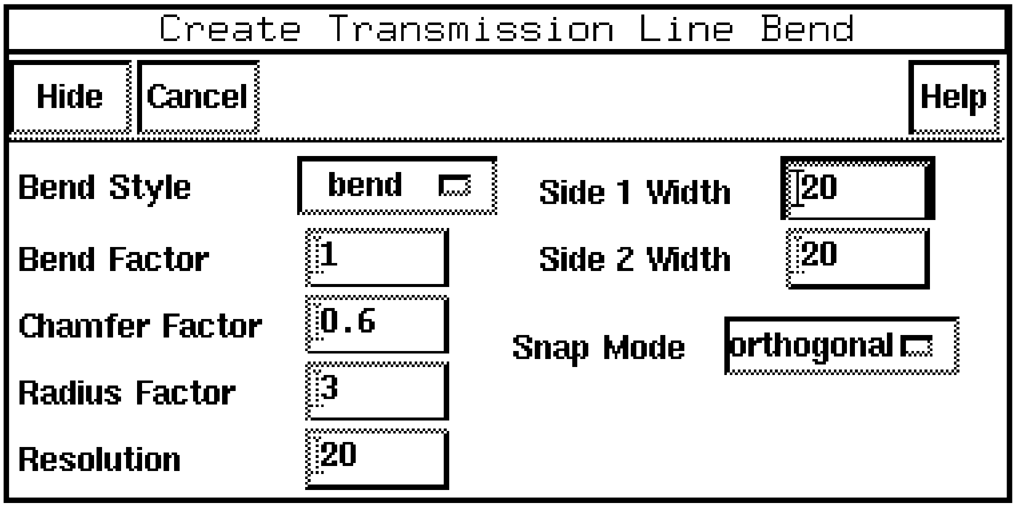
Creating Bends

The *Bend* command lets you create a two-segment transmission line with two different width settings.

To display the Create Transmission Line Bend form,

➤ Choose *Create – Bend*.

About the Create Transmission Line Bend Form



Bend Style sets the type of corner to use in the bend: bend, chamfer, or radial.

Bend Factor sets the maximum width allowed in a standard bend before the bend is automatically chamfered. (Available only when *Bend Style* is *bend*.)

Chamfer Factor sets the fraction of the corner that is removed in a chamfer bend. (Available only when *Bend Style* is *chamfer*.)

Radius Factor sets the ratio of the centerline bend width to the radial bend width. (Available only when *Bend Style* is *radius*.)

Resolution sets the number of segments used in a 180-degree radial bend.

Side 1 Width and **Side 2 Width** set the width of the two segments in user units (typically microns).

Snap Mode controls the shape of the transmission line segments.

Create Bends

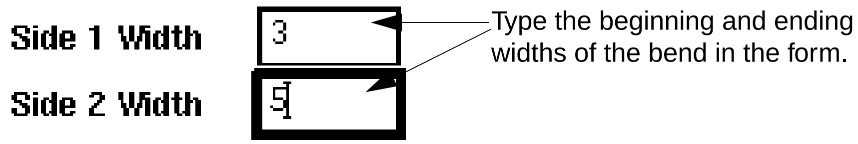
Bends connect objects of different sizes, using a right-angle segment instead of a straight line. You create bends similarly to the way you create paths.

To create a bend,

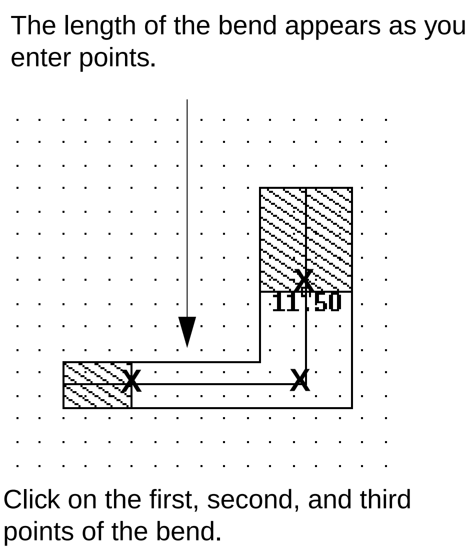
1. Choose *Create – Bend*.

The Create Transmission Line Bend form appears.

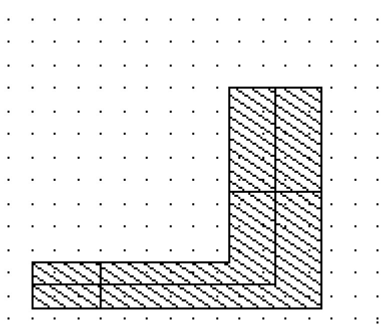
2. Type the beginning and ending widths of the bend.



3. Click to enter each point of the bend.



This is a completed bend.

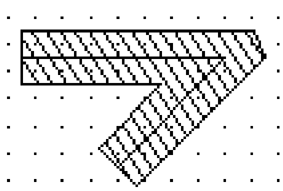


Creating Advanced Transmission Lines

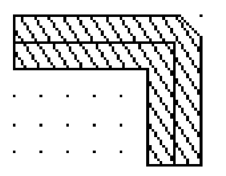
The *Bend Style* option lets you customize the corners of your transmission lines into standard bends, chamfers, and radial corners.

1. In the Create Transmission Line Bend form, or in the Create Transmission Line form, choose *Bend Style* and type the corresponding factor to use.

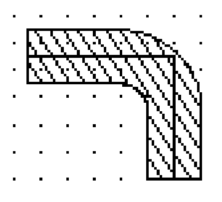
Bend Style: Bend
Bend Factor: 1



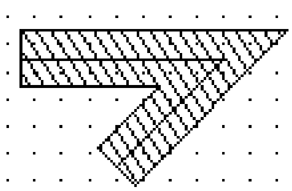
Bend Style: Chamfer
Chamfer Factor: .2



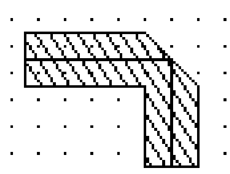
Bend Style: Radial
Radius Factor: 1



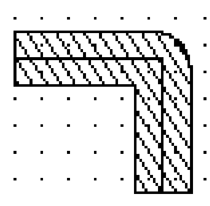
Bend Style: Bend
Bend Factor: 3



Bend Style: Chamfer
Chamfer Factor: .5



Bend Style: Radial
Radius Factor: .5



2. Click on each of the three points of the bend.

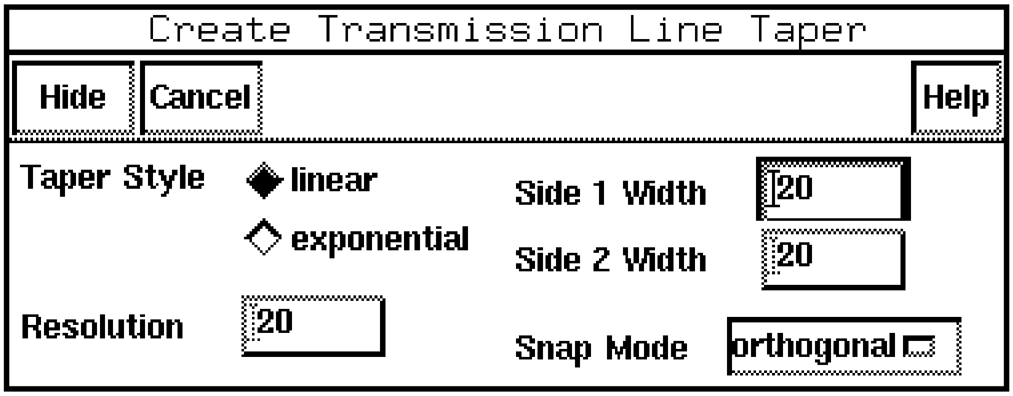
Creating Tapers

The *Taper* command lets you create a one-segment transmission line that connects two transmission lines of different width settings.

To display the Create Transmission Line Taper form,

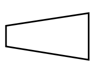
➤ Choose *Create – Taper*.

About the Create Transmission Line Taper Form

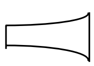


Taper Style sets the taper style you create:

- linear



- exponential



Resolution sets the number of segments used in an exponentially tapered line.

Side 1 Width and **Side 2 Width** specify the width of the taper ends in user units (typically microns).

Snap Mode controls how the points of the taper snap to the grid.

Create Tapers

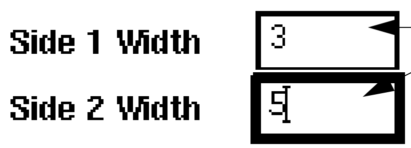
You create tapers similarly to the way you create paths.

To create a taper,

1. Choose *Create – Taper.*

The Create Transmission Line Taper form appears.

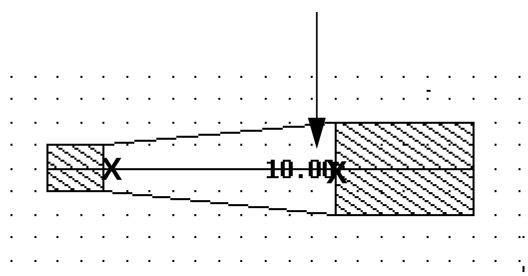
2. Type in the beginning and ending widths of the taper.



Type the beginning and ending widths of the taper in the form.

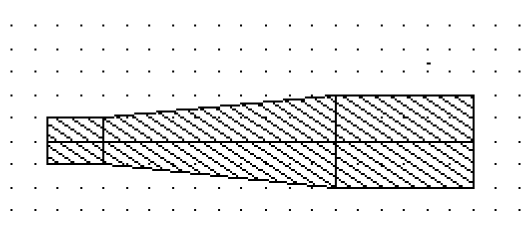
3. Click to create each point of the taper.

The length of the taper appears as you create.



Click on the first and last points of the taper.

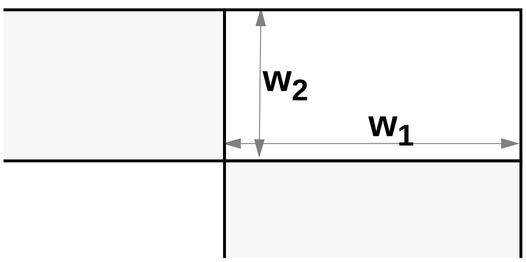
This is a completed taper.



Transmission Line Bend Examples

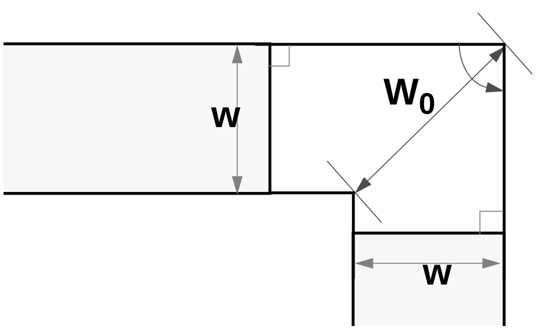
Standard Bend Examples

This illustration shows a standard bend. All angles in a standard bend are 90 degrees.



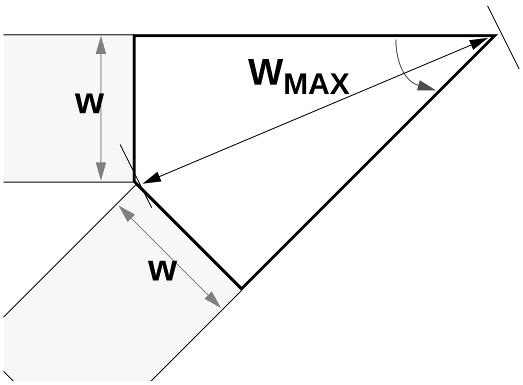
Bend Factor is the allowed ratio of the width (W_{MAX}) of the bend for a given angle to the width (W_0) of a 90-degree bend. If the width (W) of a bend is greater than W_{MAX}, the bend is chamfered so that the width is decreased to W_{MAX}. The effect of *Bend Factor* is shown in the following illustrations.

90-Degree Bend with Width W_0

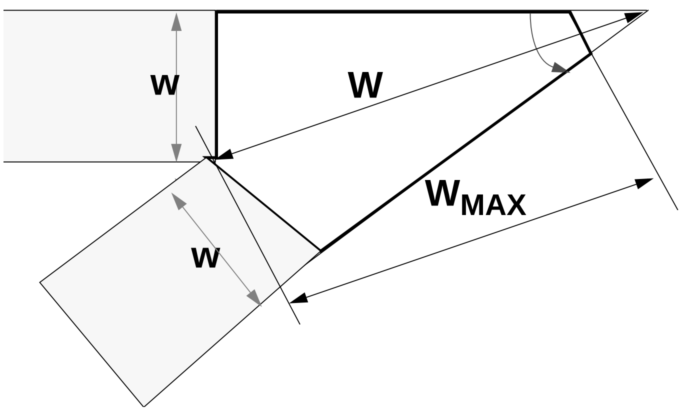


Specified Value of *Bend Factor* Defines W_{MAX}

For $W>W_{MAX}$, Bend Chamfered to Decrease W to W_{MAX}



$$\text{Bend Factor} = \frac{W_{MAX}}{W_0}$$



Chamfer Examples

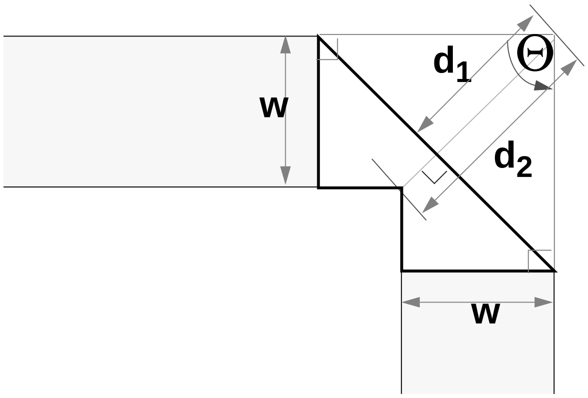
The types of chamfer bends and the effect of *Chamfer Factor* are shown in the following illustrations. The appearance of the corner is determined by the value of the chamfer factor, d_1/d_2.

Example 1

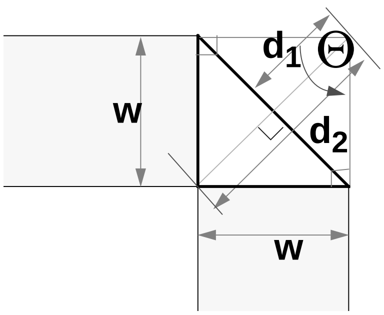
Example 2

Example 3

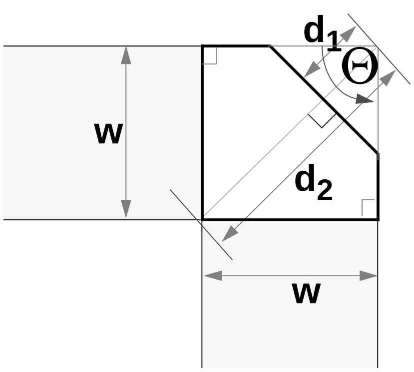
Example 4



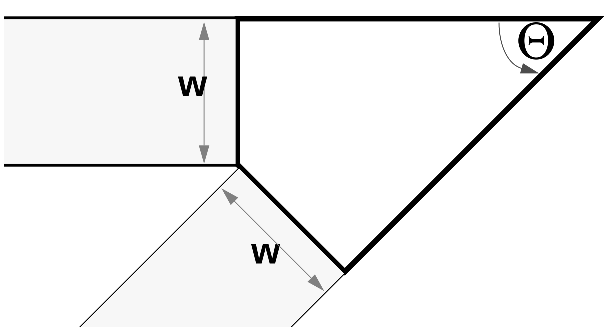
chamfer factor = d1 / d2 = .7



chamfer factor = d1 / d2 = .5



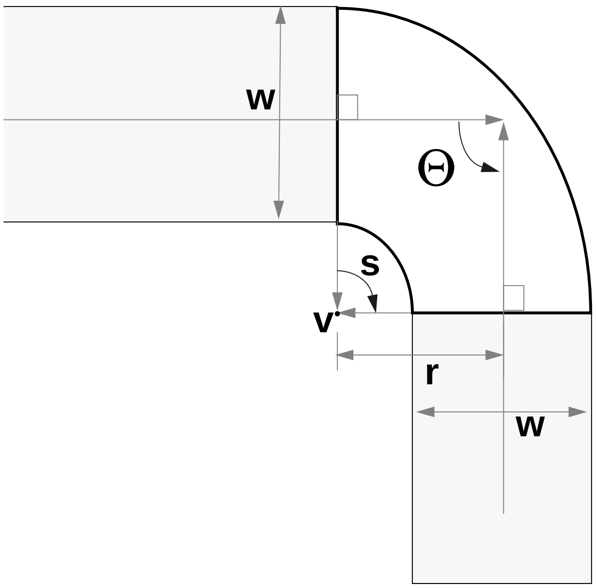
chamfer factor = d1 / d2 = .3



chamfer factor = 0.0

Radial Examples

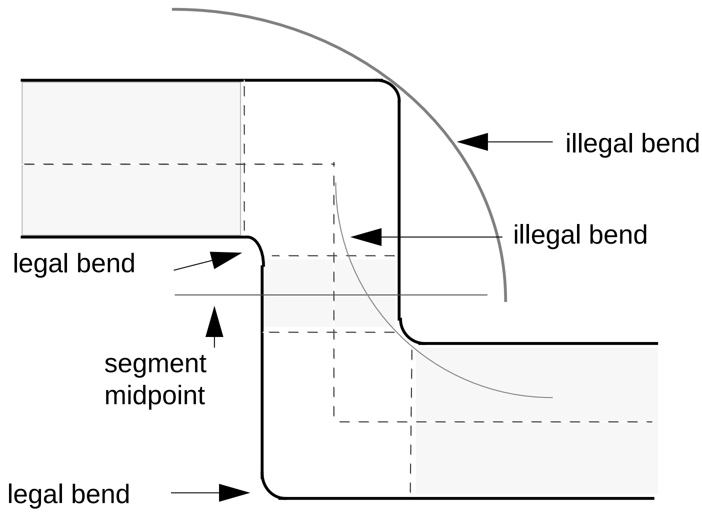
A radial bend has a rounded corner determined by the *Radius Factor.* The *Resolution* is the number of segments in a 180-degree bend.



r = (Radius Factor) * (width)

Bends must be properly formed. For example, segments adjacent to a bend must not be too short with respect to the bend. To prevent this, the layout editor does not allow a bend on adjacent segments to extend past the midpoint of the shortest adjacent segment.

In other words, the bend and the segment must coincide before or at the midpoint of the segment so that segments approaching the midpoint from opposite directions match at the midpoint. These segments must intersect the midpoint at 90 degrees. If necessary, the layout editor alters the bend to satisfy these conditions.



17

Resolving Verification Errors

This chapter contains these topics:

Working With Markers

When Assura™ interactive verification commands produce errors or warnings, the Virtuoso® layout editor provides information about the errors through the *Verify – Markers* commands. These commands help you find errors and get information about them. These commands appear under the *Verify* menu.

To get information about a marker, use these commands:

- *Verify – Markers – Explain* displays the reason for the error or warning marker in a text window
- *Verify – Markers – Find* searches for and highlights each error or warning marker

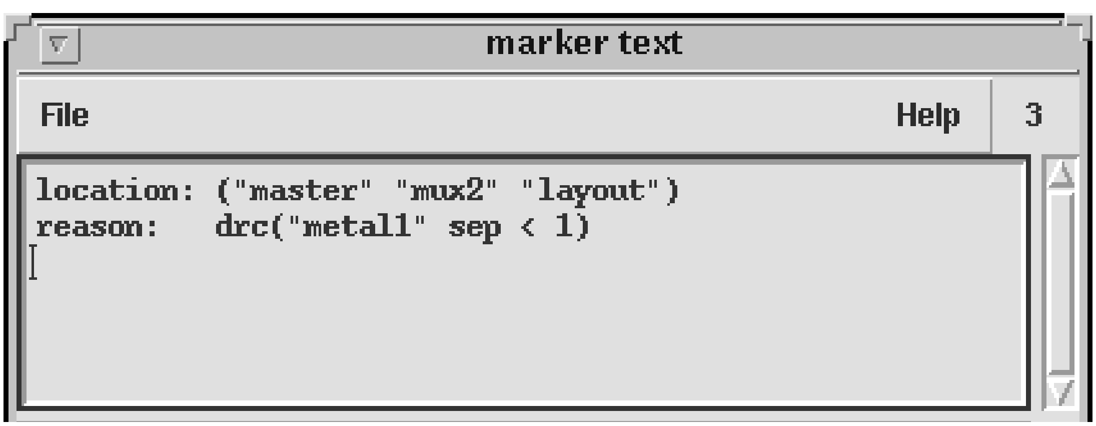
After you get the information you need, you can delete the marker using these commands

- *Verify – Markers – Delete* removes a specific marker
- *Verify – Markers– Delete All* removes all markers

Explaining Markers

You use *Verify – Markers – Explain* to obtain information about errors in your layout. The error information is displayed in a text window. Design Rules Check (DRC) and Extract generate markers of error severity only.

About the Marker Text Window



Location gives the library, cell, and cellview names.

Reason gives the reason associated with the marker. This information relates to the specific technology file rules that have been broken. In this example, the DRC found that *metal1* separation is less than 1, which breaks the physical rules for *metal1* as defined in the technology file.

Using Markers – Explain

You select markers either before or after you choose *Verify – Markers – Explain*.

To select markers,

1. Click on the marker or draw a selection rectangle around multiple markers.
2. Choose *Verify – Markers – Explain*.

   The marker text window appears and displays information about the error.
3. Continue selecting markers you want explanations for.

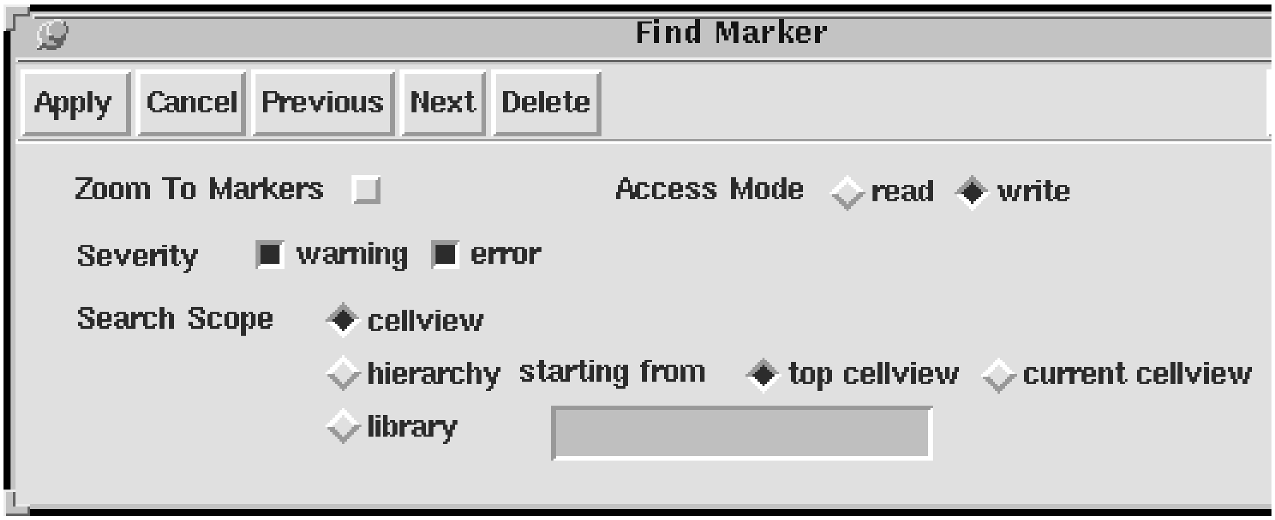
   The text scrolls in the marker text window so you can see it.
4. When you are finished viewing the errors, press `Escape` to end the command.
5. You can correct the errors and rerun the verification command.

Finding Markers

You use *Verify – Markers – Find* to narrow your search by limiting the list box display to a particular severity level. You can also use the command to expand your search by expanding the scope.

Verify – Markers – Find searches for and highlights each error or warning marker in a layout to help you correct errors. You can search for markers anywhere in the design hierarchy. You can also display the previous or next marker.

About the Find Marker Form



Zoom To Markers when on, enlarges the position of the marker in the window by zooming in on the specified marker; when off, displays the marker but does not zoom in.

Access Mode specifies the mode for opening a cellview. You base your selection on what you want to do in the cellview.

read displays the cellview but does not permit you to edit and save.

write displays the cellview and permits you to edit and save.

Severity specifies the type of markers displayed.

warning displays warning markers.

error displays error markers.

Search Scope specifies what part of the design hierarchy you want to search.

cellview searches only for objects in the cellview displayed in the current window.

hierarchy starting from searches all cellviews specified in the design hierarchy. The hierarchy either starts with the top cellview or the current cellview.

library searches all cellviews in the specified library. You must type in the library name, and the library must exist in the current search path.

Using Markers – Find

To find markers in a layout,

1. Choose *Verify – Markers – Find*.

   The Find Marker form appears.

2. Change the options on the form as desired.
3. Click *Apply*.

   The first error or warning marker is highlighted and the reason for the marker is displayed in the marker text window.

   Note: You can change the options on the form at any time and click *Apply*.

4. To find the next marker, click *Next* on the form, or to look at the previous marker, click *Previous* on the form. Then check the marker text window for the reason for the marker.
5. To delete the current marker, click *Delete*.

Note: There is a similar *Find Marker* command available in the Virtuoso schematic composer. The command options are a little different than the layout editor version and provide additional functionality. The only way to run the schematic editor *Find Marker* command in a layout window is by typing the Cadence® SKILL language function `geHiCommonFindMarker()` in the Command Interpreter Window (CIW). For more information on this command, see the *Virtuoso Schematic Composer User Guide*.

Deleting Individual or Multiple Markers

To delete individual or multiple markers in the design window,

1. Indicate which marker you want to delete by doing one of the following:
   - Click the marker(s) you want to delete individually.
   - Draw a selection box around multiple markers.
   - Choose the message on the Find Marker form that corresponds to the marker you want to delete.
2. Choose *Verify – Markers – Delete*.

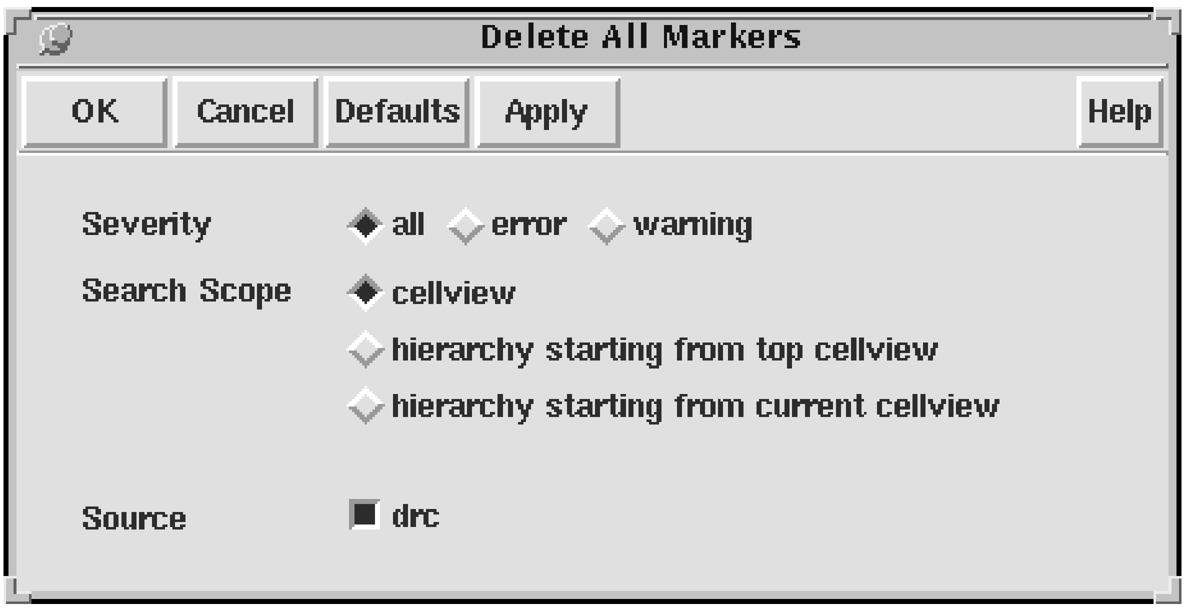
   The message(s) and the corresponding marker(s) are deleted.

   If you chose the command before you selected the markers, press `Escape` to end the command.

Deleting All Markers

You can find and delete all the markers of a particular severity level, and in a particular cellview or hierarchy, using *Verify – Markers– Delete All*.

About the Delete All Markers Form



Severity defines which level of markers the system deletes in the list box and in your design.

all deletes all markers.

error deletes all markers with error messages.

warning deletes all markers with warning messages.

Search Scope defines the scope of the deletion.

cellview deletes markers from the current cellview only.

hierarchy starting from top cellview deletes markers throughout the entire hierarchy.

hierarchy starting from current cellview deletes markers throughout the hierarchy starting with the current cellview.

Source specifies the category of markers that are to be deleted in terms of the program that generated them.

drc deletes Design Rules Check markers.

lvs deletes Layout Versus Schematic markers.

none appears when there are no messages in the list box.

A

Bindkey Keyboard Map

F keys						
F1	F2	F3	F4	F5	F6	F7
Help	Save		Toggle Partial Select	Open Design	Maintain Connec-tions	Guided Path
F8	F9	F10	F11	**F12**		
Guided Path Create	Filter Size					

Alphabet keys				
Key to Map: 1=Top row is Control + key 2=Middle row is Shift + key 3=Bottom row is key				
A	**B**	C	**D**	**E**
1 Select All	1	1 Interrupt	1 Deselect All	1
2 Select Area	2 Return	2 Chop	2 Deselect Area	2 Display Ops.

3 Select	3 Go to Level	3 Copy	3 Deselect	3 Edit Options
F	**G**	**H**	**I**	**J**
1 View 0	1		1	
2 View 32	2 Zoom To Grid		2	
3 Fit All	3 Toggle Gravity		3 Create Instance	
K	**L**	**M**	N Snap mode options:	O
1	1	1	1 diagonal	1
2 Clear Rulers	2	2 Merge	2 orthogonal	2 Rotate
3 Draw Rulers	3 Label	3 Move	3 L90XFirst	3 Create Contact
P	**Q**	**R**	S	**T**
1 Create Pin	1	1 Redraw	1 Split	1 Zoom to Set
2 Create Polygon	2 Design Prop	2 Reshape	2 Search	2 Tree
3 Create Path	3 Object Prop	3 Create Rectangle	3 Stretch	3 Layer Tap
U	**V**	**W**	**X**	**Y**
1	1 Type in CIW	1 Close	1 Fit Edit	1 Cycle Select
2 Redo	2	2 Next View	2 Descend	2 Paste
3 Undo	3 Attach	3 Previous View	3 Edit-In-Place	3 Yank
Z	**Esc**	**Tab**	**Delete**	**Back Space**
1 Zoom In x2	Cancel	Pan	Delete	Undo Point
2 Zoom out x2				
3 Zoom In				
Return Enter last point				

Arrow keys				
Control + key: Fit cell to portion of window Shift + key: Move Cursor Key: Pan to portion of cellview				
R7 Home up left	**R8** ↑ top	**R9** PgUp up right		
← **R10** left	R11 center	R12 → right		
R13 End down left	R14 ↓ bottom	**R15** PgDn down right		

Symbol keys on arrow key pad					
= R4	**/ R5**	*** R6**	-	**+**	**Enter**
Moves cursor .5 grid points when used with Shift and arrow keys	Moves cursor 1 grid point when used with Shift and arrow keys	Moves cursor 2 grid points when used with Shift and arrow keys	Delete reference point	Set reference point	Enter point

Glossary

A

abutment

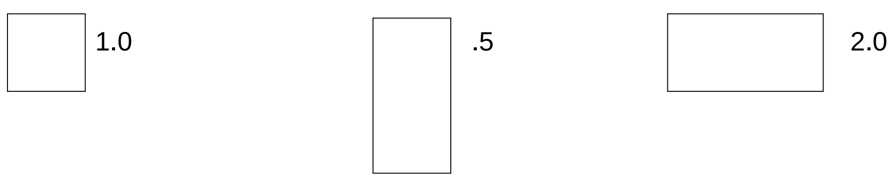
Abutment is the ability to overlap, partially or completely, instances of cells to make an electrical connection without introducing design rule check or connectivity errors. The two instances must include pins connected to the same net.

AEL expressions

Analog Expression Language (AEL) expressions used to define design parameters as functions of variables. These expressions are used by the mixed signal and analog netlisters.

aspect ratio

The width-to-height ratio of a layout. Setting a 1:1 ratio results in a square shape.



A (continued)

autoLayout

A cellview that shows the physical design hierarchy at a given hierarchical level. Block Ensemble™ and other place-and-route tools use this cellview. The software automatically creates autoLayout cellviews when you flatten the logical hierarchy.

Top-level designs and soft blocks have autoLayout cellviews. A flat autoLayout has only one hierarchical level and one autoLayout cellview. A hierarchical autoLayout has one top-level autoLayout and one autoLayout for each soft block in the hierarchy.

The autoLayout cellview references the abstract or autoAbstract cellviews for the instances it contains but does not show all physical details of the instances.

B

banner

In the Cadence® software, a bar across a window that contains the names of menus.

bbox. *See* bounding box

bindkey

In the Cadence® software, a key combination or mouse button that executes one or more Cadence® SKILL commands.

block

A collection of instances that forms a hierarchy.

boundary. *See* design boundary

bounding box (bbox)

A rectangle that encompasses all of the shapes in an instance (or a selected set).

bus

A named collection of nets. Bundling nets together in a bus makes it easier to route nets with similar routing patterns. You cannot use buses in Gate Ensemble® and Cell3 Ensemble® designs.

C

Cadence design environment

The Cadence® software: programs running within the operating system and X Window System environment.

Cadence software

The programs that contain graphic design capabilities and a graphic user interface from Cadence Design Systems.

Cadence system

A computer system with the Cadence® software installed.

CDF. *See* Component Description Format

.cdsinit

The file that defines the startup environment of the Cadence® software.

cell

A component of a design; a collection of different aspects (representations) of component implementations, such as its schematic, layout, or symbol representations. A design object consisting of a set of views that can be stored and referenced independently. A cell can include other cells, forming a hierarchical design. A cell is an individual building block of a chip or system. In the database, a cell contains all the cellviews of that cell.
An inverter and a buffer are examples of a small cell. A decoder register, arithmetic logic unit (ALU), memories, complete chips, and printed circuit boards are examples of large cells.

C (continued)

cellview

A specific representation (view) of a cell. A particular representation of a particular component, such as the physical layout of a flip-flop or the schematic symbol of a NAND gate. A database object containing all the information unique to a particular representation of a particular component. Cellviews are classified by their viewType. Each cellview has a view name and can have one or more versions. *See also* view.

cell boundary

The outside boundary of a layout cellview; a rectangle on any user-defined layer of function property *cellBoundary* or the reserved layer *boundary*.

For user-defined layers, there is an implicit zero-valued enclosure rule from the boundary to any object contained inside. For the cell boundary layer, all objects are kept inside the boundary.

chaining

Chaining, of transistors, is the ability to automatically abut a list of MOS transistors (or the fingers of folded transistors) with one another in a specified order. Transistors to be chained must be set up for abutment.

CIW. *See* Command Interpreter Window

click

Rapidly press and release a mouse button.

C (continued)

Command Interpreter Window (CIW)

The window that launches any Cadence® design framework II application. The CIW logs your design session and reports messages.

compactor. *See* Virtuoso® compactor

compacted view

A compacted layout that contains transistors, contacts, routing, and net connectivity.

component

For the purposes of Virtuoso XL, a component is a pin or a device.

Component Description Format (CDF)

A Cadence® design framework II feature that lets you assign attributes, properties, and parameters to libraries and cells for such purposes as assigning parameter names and values, allocating units and default values, checking that values lie within specified ranges, dynamically changing how parameters are displayed depending on predefined conditions, and executing Cadence® SKILL programming language functions whenever certain information is changed (callback functions). A cell CDF lets you store information specific to a cell with that cell. For more information, see the *Component Description Format User Guide.*

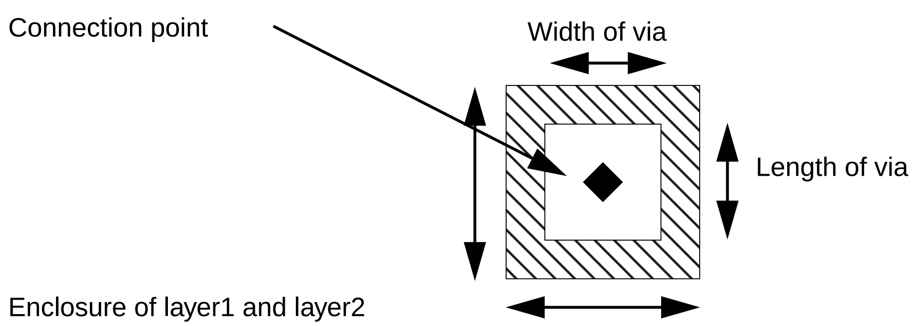
C (continued)

constraint

A restriction on the placement or size of an object, or a timing requirement for a net or delay path.

contact

A connection point between wires of different types (generally metals and non-metals) that connects the lowest metal layer and the conducting layer below it. The structure contains the two conducting layers and contact cut.



current window

The Cadence® window in which you do something, such as click on a title bar or border, pull down a menu, or work on a design. The window remains the current window until you do something in another design window.

current directory

The directory in which you started the Cadence® software.

C (continued)

cyclic field

A button on a form that displays a list of valid options when you click a mouse button on it. For example, a cyclic field button called *Units* lets you select the units of measurement and gives you a choice of inches, centimeters, mils, or microns.

D

default value

The value used by the software unless you specify otherwise. The default is frequently the initial state.

delay

The time interval between the manifestation of a signal at one point and the manifestation or detection of the same signal at another point.

delay path

An ordered series of instance-pin pairs that forms a connected signal path.

design

A window holding a cellview. A composite of cells and views, usually hierarchical.

D (continued)

design boundary

The outside boundary of a layout cellview; a rectangle on the reserved layer boundary *(prboundary)*. For user-defined layers, there is an implicit zero-valued enclosure rule from the boundary to any object contained inside. For the *boundary* layer, all objects are kept at half the maximum design rule inside the boundary. See also *cell boundary*.

Design framework II

The Cadence® framework system that provides a common interface for schematic capture, layout, floorplanning, place and route, and verification tools.

design library

A library that contains data for the current design. It is usually in the designer's own directory or in the design group's directory.

device

A design element that has both a symbol view and a layout view, with corresponding pins.

double-click

Press the mouse button twice, rapidly.

drag

Press and hold the mouse button while moving the mouse.

D (continued)

drain

The receiving end of the connection channel of a MOS transistor.

E

environment

The hardware and software setup and conditions within which the system operates.

externally connected pins. *See* must-connect pins

F

feedthrough pin

A pin that forms connections by passing through a cell or instance.

filled button

A darkened button on a form indicating an option is selected.

fingering

Fingering, of transistors, is the ability to divide a transistor pcell by setting the "number of Fingers" in the Property List Editor. All multiple fingers are part of the same pcell. The purpose is to split a transistor, but keep all fingers adjacent to one another under all conditions.

first point

The starting position of a graphic.

fixed menu

A menu with columns of buttons that appears in some application windows. Icons on the buttons represent selected pull-down menu commands.

floorplanning

Creating a rough plan to estimate whether a design meets timing and routability criteria.

F (continued)

folding

Folding, of transistors, is the ability to divide a MOS transistor pcell by the width into two or more instances (as specified) that can be kept adjacent or separated to achieve an interleaved configuration. You can fold two series transistors and interdigitate the folded legs using abutment. Transistor folding can be specified during device generation, when picking devices from the schematic, and interactively by selection.

font

A complete set of characters in one size, typeface, and style, such as 12-point Times Roman.

form

A window or dialog box that lets you specify information and options for a specific command or menu item in the Cadence® software. The options take effect when you click *OK* or *Apply.*

form field

A rectangle in a form into which you type information, or a cyclic field in a form that gives you a fixed group of options you can choose.

G

gate

A transistor that is formed when poly dissects diffusion. A gate is a device element for combinational logic, for example, AND gates and OR gates.

gate array

Gates placed in a prefabricated matrix where you provide the design for the interconnect. Gate array designs are cheaper to manufacture because only the interconnects have to be custom made.

global router

A router that divides the total routing area into parts and assigns the nets to be connected locally in each of the parts.

H

hierarchy

Nested design levels, such as instances within a cell. By default, you open the top level in the hierarchy when you open a cellview.

home directory

The directory in which you are placed when you log into a computer and to which you have read and write permission.

I

icon

A small graphic symbol that represents a cell, window, or application.

iconify

Change a window to an icon.

instance

A database object that represents a master cellview. You can have several instances of the same cellview in a design.

internal pin

A pin inside a cell that can be connected to the cell boundary through a specific access direction. Internal pins help connect blocks by allowing over-the-cell routers access to all pins in the layout.

internally connected pins. *See* strongly connected pins

interrow routing areas

The routing area between rows in the layout.

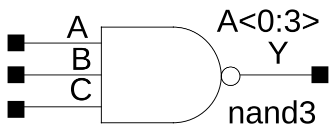
intrarow routing areas

Routing areas within each row and between the N- and P-diffusion strips.

I (continued)

iterated instances

A compact way of displaying repeated instances of a symbol in a schematic, particularly useful in bus-type or data-flow architectures that have identical structures to handle each bit on the bus. To add several instances of the same type, you can express multiple unique names with an iterative expression. For example, A<0:3> generates one graphic representing four instances: A0, A1, A2, and A3.



L

layer

In the Cadence® software, a physical or other design entity used as a visual representation of different types of information, such as mask geometries and interconnection in schematics. Each layer has its own colors, highlighting, menus, and design objects.

Layer Selection Window (LSW)

A window that lets you choose the design layer for objects in the layout, make design objects visible or invisible, or make design objects selectable or unselectable.

layout cellview

A layout view of a cell. Layout views include placed, uncompacted, and compacted views.

le_ex_# net

The notation used by Virtuoso XL to indicate a shape that is not part of a defined net in the design–usually an open or floating shape.

L (continued)

library

A logical collection of cells, views, and technology information. A physical collection of files and directories that can reside anywhere in the file system. A library can be shared by all users and controlled by a single person.

Each library has associated files, such as the technology file, catalog file, foreign database files, and an audit trail. The catalog file keeps track of the logical names of design objects and their physical location. The technology file governs mask layer names and colors, design rules, symbolic device definitions, and parameter values.

Library Manager

The Cadence® tool that displays the list of available libraries in your search path as well as any open libraries. You can use the Library Manager to search through your libraries, cells, views, cellviews, and versions.

list box

Fields in forms that you can scroll up or down to view and select items from a list.

LSW. *See* Layer Selection Window

M

master cell

Any layout cell you have placed in another cell. The placed copy of the cell is called a “cell instance.”

master symbol

A representation of a design, such as an arithmetic logic unit (ALU) or register. In a schematic, an instance is linked to the master symbol; it is not a copy of the master symbol.

You can create a master symbol for a design and then place virtual copies (instances) of it throughout other designs.

maximize

Enlarging a window to fill the entire screen.

menu banner

The rectangular area across the top of a window. It contains menus, such as *File,* which display lists of commands when you click on them.

minimize

Changing a window to an icon.

must-connect pins

Pins that must be connected outside a cell and do not require routing between them inside the cell. Sometimes called “externally connected pins.”

M (continued)

.mwmrc

The Motif window manager file that defines the Motif environment. Your system might have a default `.mwmrc` file or you might have your own in your home directory.

N

net

A logical signal connection between a set of pins on different instances. After routing, a net consists of routed wires on the routing layers.

NLP (Netlist Processor) Expressions

Expressions used to define design parameters as functions of variables. These expressions are used by the Open Simulation System (OSS) in netlisting.

O

object

A library, cell, view, cellview, version, cell category, or basic design object –such as a wire, via, or model–or a display object–such as an array or label. Each type of object, such as each type of path, is drawn on a different layer. Objects can have different functions or features, depending on the layer purpose.

options form

A window or dialog box that lets you specify information and various options while you are using a command. The options take effect immediately.

P

pan

To view a design by moving it in the window.

parameterized cell

A master cell that has parameters such as length and width. When creating an instance, you can change these parameters without changing the master cell. Parameterized cells are often called “pcells.”

path

The course over which electrical current flows in a circuit. Also, the route through directories the system must take in the directory hierarchy to access a file.

pcell. *See* parameterized cell

pin

A physical implementation of a terminal. You can place pins on any layer. Pins can be feedthrough or internal. *See also* terminal.

pop-up

A menu that appears when you press the middle mouse button.

P (continued)

postselecting

Selecting a command first and then selecting the objects on which the command operates.

preselecting

Selecting objects for a command to operate on first and then selecting the command.

property

A characteristic of a design object or cellview that affects the appearance of the object. A property can be edited and deleted. Certain properties are mandatory for certain applications. Properties are defined and managed by the application.

R

radio button

A button on a form that lets you select only one of several choices. *See also* toggle button.

reference point

The point used to measure the exact distance between objects or line segments as you draw. The distance between reference points is shown in the *Dist* field at the top of the cellview.

reference library

A library that contains design data for cells placed in the current design, usually a well-verified collection of design objects shared and existing in a public system library.

relative path

The path to a file or directory from the current location within the file system.

restore

Change an icon back to a window.

routing

Physically connecting objects in a design according to design rules set in the reference library or technology file.

R (continued)

row

A rectangular area in a layout that contains a power and ground net, an N- and P-diffusion strip, and intra-row routing. Areas between rows in a layout are called inter-row routing areas.

S

schematic cellview

A cellview that describes the connectivity, gate widths, and gate lengths of transistors.

SDF (Standard Delay Format)

An ASCII format for writing out delay information and for passing the data between different Cadence® tools.

search path

The list of directories the software searches for files, libraries, and commands.

select

Choose an object by clicking on it or by enclosing it in a bounding box.

SKILL

The Cadence® SKILL language is a high-level programming language. Menu commands execute SKILL commands. SKILL is based on the LIS Processing (LISP) programming language but uses a C-like syntax. It is accessible only from within Cadence software.

source

The sending end of the connection channel of a MOS transistor.

S (continued)

stack

A group of abutted transistors.

strongly connected pins

Pins that are connected inside a cell and do not require external connection. Sometimes called "internally connected pins."

substrate contact

A contact that connects the substrate to power.

symbolic contact

A contact defined with the `tfcDefineSymContactDevice()` statement in the technology file. You can set the size and property values for these contacts. Symbolic contacts have symbolic views that contain contact information. Symbolic contacts are used as input and output terminals on lower-level blocks.

T

technology file

An ASCII file that uses Cadence® SKILL functions to define all the physical information required for a design. Technology files define information such as layers and layer properties, colors, display and plotter devices, views and view properties, physical design rules for compaction, and devices.

template

Information entered through the Layout Generation Options form and saved in an ASCII file that tells Virtuoso XL how to format a new layout.

terminal

The electrical input or output of a net. *See also* pin.

text entry window

A window in which you can enter commands or other information. A form is one type of text entry window.

text viewing window

A window in which you can view the contents of a file without leaving another application.

toggle

The action of switching between two states, such as on and off.

T (continued)

toggle button

A button on a form that lets you turn any number of choices on or off. *See also* radio button.

toolkit

A group of programming utilities.

top-down design

An approach to hierarchical design that uses estimates and floorplanning to start at the top level of a design.

transistor chaining. *See* chaining, of transistors

transistor fingering. *See* fingering, of transistors

transistor folding. *See* folding, of transistors

U

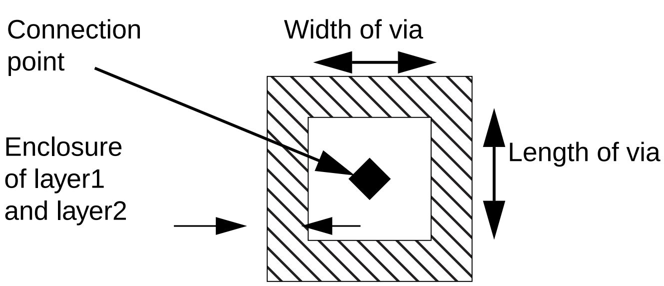
unplaced transistors

Components placed outside the cell boundary.

V

via

A connection point between wires of different types.



view

A specific representation of a cell, such as schematic, geometric, symbolic, logical, or routing. In the database, a view contains all cellviews of that view. Each view can have a *viewType* property that associates it with a specific application. For example, the view named "XYZ" could be a *viewType* "layout."

view name

A cellview property that defines the name for a cellview. A cellview has a default view name based on its view type, but you can give the cellview any name you want.

V (continued)

viewType

A property of a view that associates it with a particular application. The Cadence® design framework II software recognizes a set of registered *viewTypes*, such as the schematic view and the layout view.

Virtuoso® compactor

A Cadence® tool you can use to compact a layout. For more information about the compactor, see the Virtuoso Compactor Reference Manual.

W

weakly connected pins

An external connection between a pin or group of pins in a net to avoid specific internal connections (typically ones with high-resistance paths).

window

In the X environment, a rectangular area on a graphics workstation that emulates a terminal and runs an application separate from the applications in other windows. Usually you can have several windows on your screen at one time.

wire

A connection etched into the polysilicon, diffusion, or metal routing layers on an integrated circuit when it is fabricated.

X

.Xdefaults

An X Window System startup file that defines the X environment. Your system might have a default `.Xdefaults` file, or you might have your own `.Xdefaults` file in your home directory.

.xinitrc

An X file that defines the starting X environment. Your system might have a default `.xinitrc` file, or you might have your own `.xinitrc` file in your home directory.