cadence®

OrCAD Capture User Guide

Product Version 16.6
October 2012

Product Version 16.6
October 2012

Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

2. The publication may not be modified in any way.
3. Any authorized copy of the publication or portion thereof must include all original copyright, trademark, and other proprietary notices and this permission statement.
4. The information contained in this document cannot be used in the development of like products or software, whether for internal or external use, and shall not be used for the benefit of any other party, whether or not for consideration.

Contents

1

Before you begin

Welcome

OrCAD® Capture (henceforth referred to as Capture) is a schematic design tool set for the Windows environment. With Capture, you can draft schematics and produce connectivity and simulation information for printed circuit boards and programmable logic designs.
Capture is fully integrated in a number of different tool suites, including OrCAD PSpice, and PCB board layout tool set. Refer to OrCAD Product Installation Guide for Windows for information on hardware and software requirements.

How to use this guide

This guide is designed to make the most of the advantages of onscreen books. The table of contents and cross references provide instant links to the information you need. Just click on the text and jump.
If you find printed paper helpful, print only the section you need at the time. When you want an in-depth tutorial, print the example. When you want a quick reminder of a procedure, print the procedure.

Symbols and conventions

Our documentation uses a few special symbols and conventions.

Notation	Examples	Description
Bold text	**Import Measurements**, **Modified LSQ**, **PDF Graph**	Indicates that text is a menu or button command, dialog box option, column or graph label, or drop-down list option

Icon graphic		Shows the toolbar icon that should be clicked with your mouse button to accomplish a task
Lowercase file extensions	.aap, .sim, .drt	Indicates a file name extension

OrCAD Capture

OrCAD Capture can be used by designers to create schematics and produce connectivity and simulation information for printed circuit boards and programmable logic designs. You can use OrCAD Capture to create designs for other EDA applications by choosing to set up a PSpice project, PCB project, or programmable logic project when you start a new project. You can set your user preferences for the appearance of all designs on your system, and set up design options for each particular project or design you create.
Capture provides standard libraries that can be used to design schematics. You can also create your own library.
In Capture, you can drag and drop schematic folders and pages in the project manager in the session window. You can place parts and pins in Capture schematic editor, then connect the parts with buses, wires, off-page connectors, and more. Use a multitude of Capture tools to edit the design, including the part editor, the Edit menu, and the pop-up menu. Create your own parts and part packages, or use the standard libraries provided with Capture.
The property editor of Capture shows you properties of all or selected parts in your schematic design; one page at a time. You can use the property editor to add, change, or delete user properties and property values in your own custom filter for any design.
Capture also includes verification and reporting, printing, and netlisting features for a schematic page, a folder, or an entire design.

For more information about	**See**
OrCAD Capture	Capture User Guide
OrCAD Capture messages	Capture Messages Reference Guide
Libraries	OrCAD Supplied Libraries Reference Guide
Toolbar commands and shortcut keys	OrCAD Capture Quick Reference
PSpice simulations	PSpice User Guide
PCB design	Allegro PCB Editor User Guide

PCB design cycle	OrCAD Flow Tutorial

Related documentation

In addition to this guide, you can find technical product information on the Cadence website www.cadence.com/orcad. The table below describes the types of technical documentation provided with Capture.

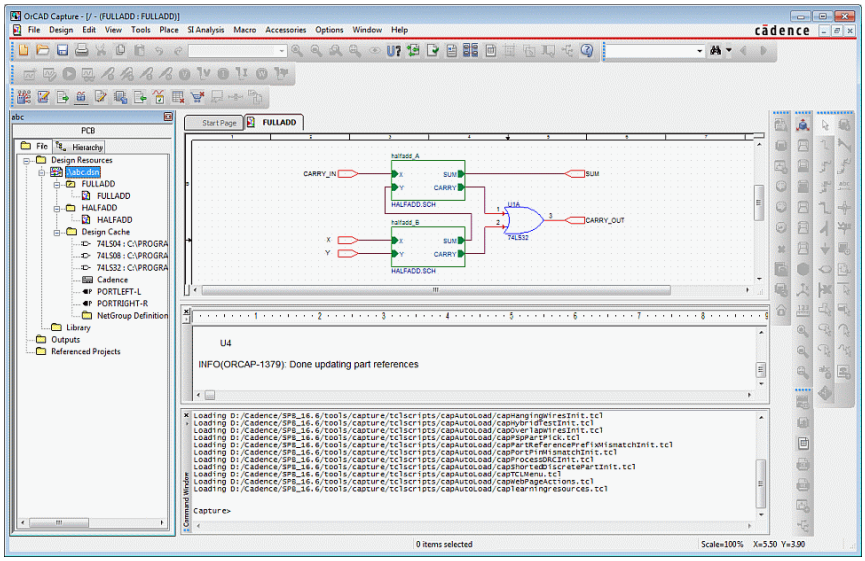
This documentation component . . .	**Provides this . . .**
This guide—OrCAD Capture User Guide	A comprehensive guide for understanding and using the features available in OrCAD Capture.
Help system (automatic and manual)	Provides comprehensive information for understanding the features in Capture and using them to perform schematic capture. Capture provides help in two ways: context-sensitive help and manual help. Context-sensitive help displays help topics that are associated with your current activity when you press the F1 key or click the Help button on the active dialog box or window within the Capture workspace and interface. It provides immediate access to information that is relative to your current task. The manual method gives you full navigational access to all topics and resources outside of the help system. Using either method, help topics include: • Explanations and instructions for common tasks • Descriptions of menu commands, dialog boxes, tools on the toolbar and tool palettes, and the status bar • Glossary terms

	• Reference information
Online interactive tutorial	A series of self-paced interactive lessons. You can practice what you've learned by going through the tutorial's specially designed interactive exercises. You can start the tutorial by choosing Learning OrCAD Capture from the Help menu and then clicking the OrCAD Capture Tutorial link.
OrCAD Capture Quick Reference Guide	Concise descriptions of the commands, shortcuts, and tools available in Capture

2

About the Capture Workspace

OrCAD Capture is a graphical user interface based application used for schematic design. The tool provides you with a large set of user-friendly tools to easily capture your schematic design.



The **Project manager** provides a hierarchical Windows explorer type access to the resources in your Capture design.

The **schematic page** and **part** editors provide a consistent and easy way to create and modify pages and parts that you require to build your design. The session windows logs the application messages.

The workspace also includes a set of **menus** that contain the commands (in the form of menu items) that you use for the operations in Capture. In addition to the menus, Capture also comes with a set of **toolbars** that provide shortcuts to many of the commonly used commands.

In this section:

The Start Page

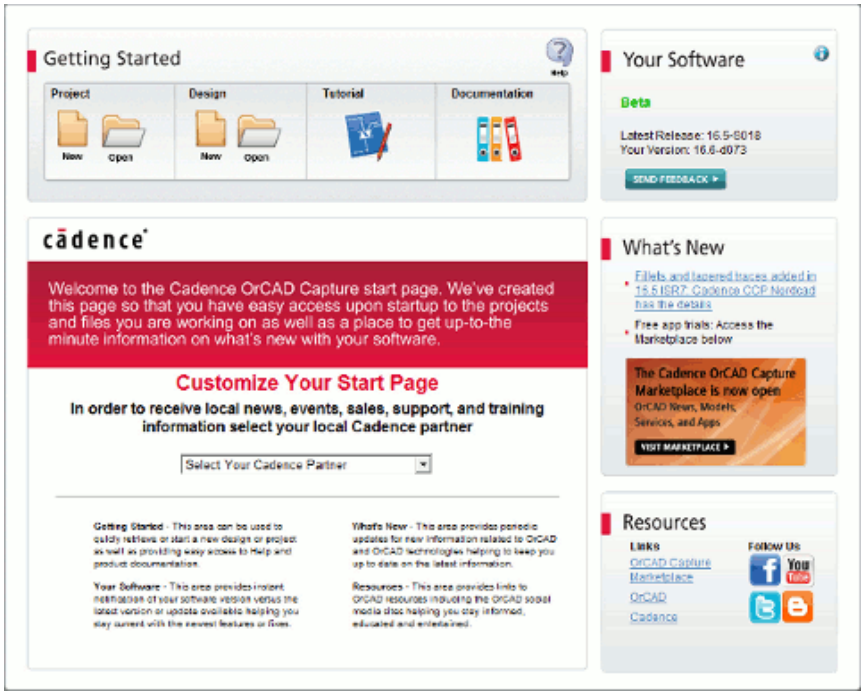
The OrCAD Capture Start Page is displayed each time you launch Capture.

Getting Started section

- This section includes quick links to Open and Create Projects and Designs.
- It also includes links to a Getting Started tutorial and the feature documentation.

Cadence Channel Partner section

- This section displays contents as sourced from your region Cadence Channel Partner.
- When you launch Capture for the first time, this section displays the Cadence default. However, you can Select Your Cadence Partner from the drop-down list.
- If you need to change the Channel Partner selection, use the Update Vendor Selection drop-down list in the *Resources* section,



Your Software section

- This sections describes the version of OrCAD Capture that is installed on your computer as well as the latest available version of OrCAD Capture.

What's New section

- This section provide quite links to new trends in the OrCAD space.
- It also provides a link to the OrCAD Marketplace.

The Update Vendor Selection drop-down list is available only after you have selected a Channel Partner.		**Resources section** • This section includes links to the OrCAD forums and communities such as Facebook, Twitter, and YouTube. • It also includes the Update Vendor Selection drop-down that allows you to change your Channel Partner selection. This drop-down is displayed after you select your Cadence Channel Partner.

The Project Manager

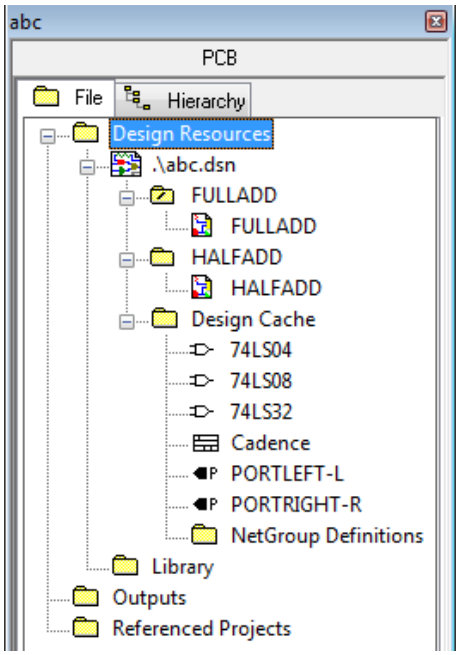
The project manager appears in the Capture session frame whenever you open or create a project. Use the project manager to collect and organize all the resources you need for your

project throughout the design flow. These resources include schematic design files, part libraries, netlists, VHDL or Verilog models, simulation models, timing files, stimulus files, and any other related information.

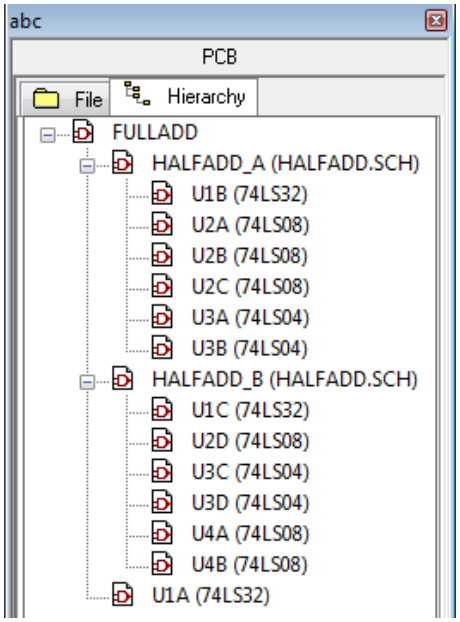
Views

The project manager provides two views of a project.

- File View
- Hierarchy View

File view
The file view is organized into folders and displays all the files included in the project.

These files may include VHDL models, netlists, schematic pages, simulation models, stimulus files, or any other files that contain information related to the project.

Hierarchy view
The Hierarchy tab shows the hierarchical relationship among the various design modules.



A design module is a structural block, typically represented as a distinct hierarchical entity, that defines the functionality of a particular portion of your design. A design module in Capture can be a VHDL or Verilog model or a schematic folder.

Each instantiation of a particular module appears in the hierarchy view as part of a hierarchical "tree." The hierarchical view of the design is derived from the files that exist in the Design Resources folder.

Project manager behavior

Within the project manager, you can expand or collapse the structure you are viewing by clicking on the plus sign or minus sign to the left of a folder. A plus sign indicates that the folder has contents that are not currently visible; a minus sign indicates that the folder is open and its contents are visible, listed below the folder.
When you double-click on a schematic folder, Capture displays the schematic pages within that folder. If the folder is a VHDL model, Capture displays each defined entity in that model. If the folder is a Verilog model, Capture displays each defined module in the model.
When you double-click on a schematic page, a VHDL entity, or Verilog model, you open that object in an appropriate editor. For example, double-clicking on a VHDL entity opens the VHDL model file at the location of that entity definition in Capture's VHDL editor.
Each project you open has its own project manager window. You can move or copy folders or files between projects by dragging them from one project manager window to another (as well as from the Windows Explorer). If you close a project manager window, you close the project.

One design for one project

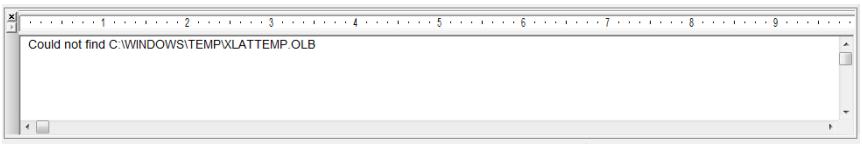
Each project may contain one design (.DSN). The design may consist of any number of schematic folders, schematic pages, or VHDL or Verilog models, but must have a single root module. The root module is the module that is defined as the top-level entity for the design. That is, all other modules in the design are referenced within the root module.

The Session Log

The OrCAD Capture session log contains a record of events that occur during the current session

of Capture. This window has a ruler with adjustable tabs, so you can format the way the information in the session log appears. This formatting only applies to the session log. It does not affect the way reports are formatted in other applications. You can set the session log ruler measurements to appear in U.S. or metric units by using the appropriate setting in the Regional Settings of your Control Panel.

The session log also includes results and messages from Capture utilities found on the Tools menu. If Capture reports an error or warning in the session log, you can get specific help on it by double-clicking on the message. In this case, Capture opens the file that contains the error and places the cursor at the location of the error. These files include netlists, CDS.LIB, HDL.VAR, and VHDL/Verilog models.



You can save the session log to any location on your system by clicking in the Session Log window and choosing File - Save As in OrCAD Capture. If Capture exits with a crash, the session log is saved as capture_crash_session.log in the %TEMP% folder, if defined, else it is saved in the current working directory.

To hide the session log

- Choose *Hide* from the pop-up menu of the left bar of the session log.

To show the session log

- Choose *Window - Session Log*

To clear the session log

- Choose *Clear Session Log* from the pop-up menu of the left bar of the session log.

The TCL Command Window

The Capture environment includes a Command window. You use this window to execute a TCL command. Also, when you perform an operation (function) in Capture, the associated command is registered with the TCL interpreter and the command is logged in the Command window.

To display the Command window

1. From the View menu, choose the Toolbar cascading menu.
2. From the Toolbar cascading menu, choose Command Window.
   The Command Window menu item is a toggle menu. This implies that if you select the menu again, you will hide the window.

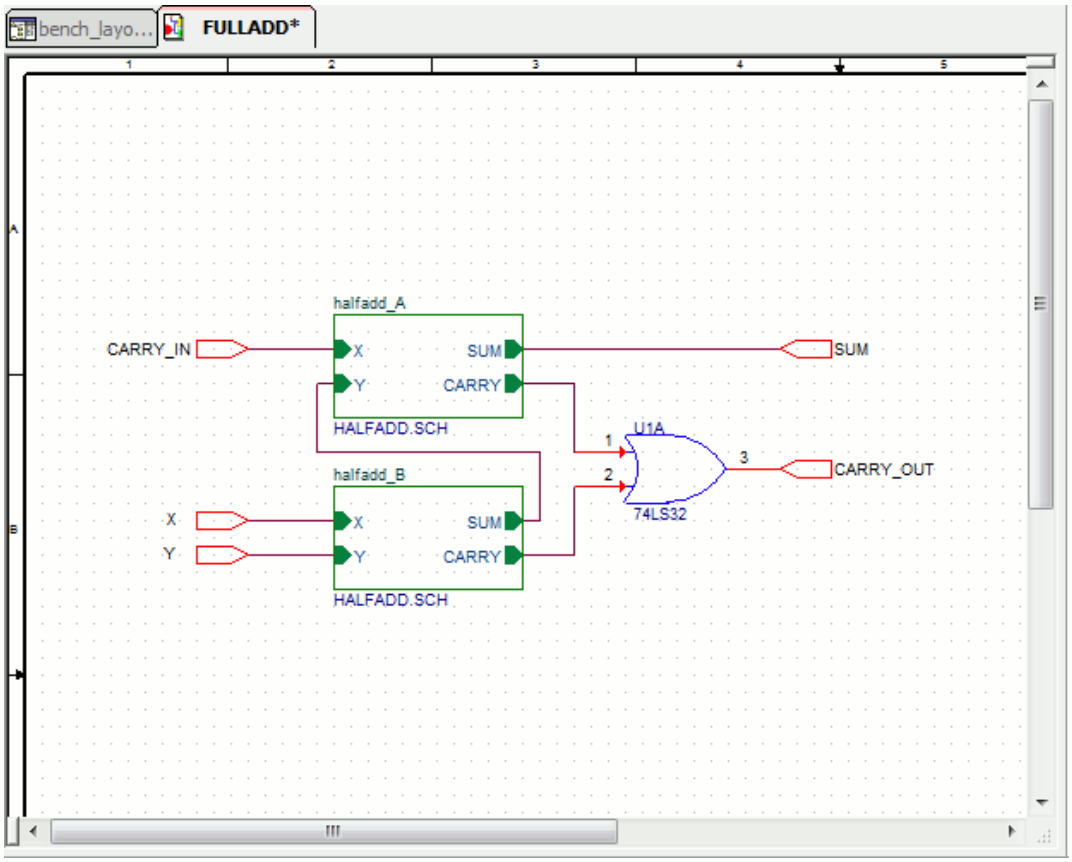
To clear the Command window

- In the Command prompt type *cls* and press Enter.

Schematic Page and Part Editors

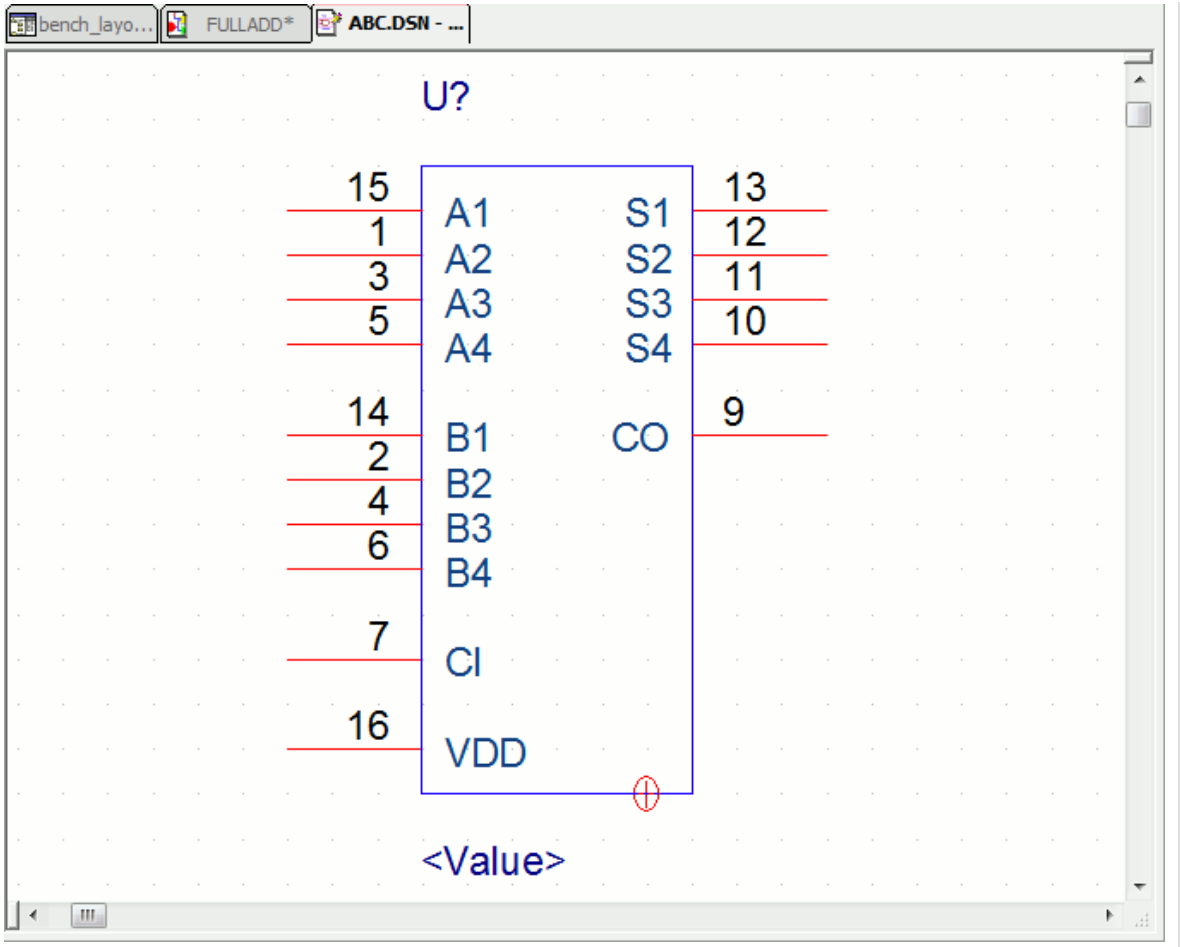
Capture includes a number of editors, including a text editor with features for creating VHDL models, a schematic editor, and a part editor. These editors mostly function in accordance with the general standards that one might expect in a Windows-based tool. However, there are certain unique traits (particularly with regard to zooming and scrolling) that distinguish the Capture editors from other Windows editors.

Schematic Page Editor



You edit schematic pages in the schematic page editor window. This window has two view splitters. The splitter at the upper right divides the view horizontally. The splitter at the lower left divides the view vertically. Each view has its own scroll bars, so you can view separate areas on the same page.

Schematic Part Editor



You edit parts and symbols in the part editor window. This window has two view splitters. The splitter at the upper right divides the view horizontally. The splitter at the lower left divides the view vertically. Each view has its own scroll bars, so you can view separate areas on the same part.

In this section:

- Moving around in the editors
- Manipulating Objects in the Editors
- Working with Text and Graphics

Moving around in the editors

In this section:

- Scrolling
- Panning
- Moving to a location, reference, or bookmark
- Using Zoom
- Setting a bookmark
- Non-Linear Editor (Fisheye)

Scrolling

In Capture, you can scroll up or down, or to the left or the right, to focus on a different portion of the active window. Even though some objects on the Place menu are attached to your pointer while you're placing them, you can still scroll.

- Click on either side of the scroll button to scroll the panning distance in the corresponding direction—up or down using the vertical scroll bar, right or left using the horizontal scroll bar.
- Click on the up, down, right, or left arrow to scroll one grid unit in the corresponding direction.
- Drag the horizontal or vertical scroll button to scroll the window dynamically.
- Press *Page Up* to scroll the panning distance up.
- Press *Page Down* to scroll the panning distance down.
- Press *Ctrl+Page Up* to scroll the panning distance to the left
- Press *Ctrl+Page Down* to scroll the panning distance to the right.
- Roll the mouse wheel up and down to scroll through vertically in the schematic page editor and property editor.
- Hold down the *Shift* key and roll the mouse wheel up and down to scroll through horizontally in the schematic page editor and property editor.
- Click the mouse wheel button and drag the mouse wheel:
    - To the right or left in the property editor and schematic page editor to scroll horizontally.
    - Up or down in the property editor and schematic page editor to scroll vertically.

Panning

When performing an action while the left mouse button is depressed (moving an object, drawing a selection area, and so forth) you can pan the display region by moving the cursor to the border of the active window. You can configure the distance by which the display changes (the panning distance) during a pan.
If you want to pan the display region while not performing any action (only for viewing), you can use the Scroll mouse button. Click on the scroll button and a Pan cursor appears. Now move the mouse anywhere across the page to pan the display area. As you move close to the edge of the display area, the area out of display will move into display.

To change the display region

1. While drawing, placing, or moving objects, or while drawing a selection area, move the pointer to the edge of the window. If there is more of the schematic page or part to display, the window scrolls in the corresponding direction.

To configure panning distance

1. Choose *Options – Preferences*, then choose the Pan and Zoom tab .
2. In the Scroll Percent text box, enter the percent of the window's horizontal or vertical dimension by which the display will scroll. Note that you can specify separate values for the schematic page editor and the part editor.
3. Click OK.

Moving to a location, reference, or bookmark

You can use the Go To command to move to specific locations (either grid coordinates, references or bookmarks) in a particular editor. The X and Y coordinates of your pointer's current location appear on the right-hand side of the status bar. Grid references appear on the left and upper edges of the schematic page in the schematic page editor.

To move to a specific location

1. Choose *View -- Go To*.
2. Select the Location tab.

3. Enter the X and Y values, select the Absolute option, then click OK. The coordinates are measured in inches or metric units, depending on what you specified in the Page Size tab of the Design Template / Design Properties dialog box. Your pointer moves to the new coordinates.

To move a specific distance

1. Choose *View -- Go To*.
2. Select the Location tab.
3. Enter the X and Y values that you want the pointer to move, select the Relative option, then click OK. The jump distance is measured in inches or metric units, depending on what you specified in the Page Size tab of the Design Template / Design Properties dialog box. Your pointer moves the specified distance.

To move to a specific grid reference

1. Choose *View -- Go To*.
2. Choose the Grid Reference tab.
3. Enter the horizontal and vertical information corresponding to the grid reference, then click OK.

To move to a specific bookmark

1. Choose *View -- Go To*.
2. Choose the Bookmark tab.
3. Enter the name of the bookmark, then click OK.

Using Zoom

In the schematic page editor and the part editor, you can zoom in to look closely at a particular area.

In this section:

Zooming In

When you press I to zoom in, Capture centers your view on the current pointer position, if possible. If the pointer is outside the window, or if you choose the Zoom In command or toolbar button, Capture centers your view on any selected objects. Otherwise, Capture zooms in on the center of the active window.

You can also zoom into a specific object on the page by using the right mouse button. Click on a blank area close to the object and keeping the mouse button pressed drag the area over the part you want to zoom into. The area is zoomed into as soon as you release the mouse button. The

zoom factor is 3.

To zoom in

Choose *View – Zoom*, then choose the Zoom In command. The current zoom scale is multiplied by the zoom factor. So, for example, a zoom factor of 2 causes the image to appear twice as large and displays half the area of the previous view.
Or
Hold down the CTRL key and roll up the mouse wheel.

To change the zoom factor

1. Choose *Options – Preferences*, then choose the Pan and Zoom tab.
2. In the Zoom Factor text box, enter the new zoom factor. Note that you can specify separate values for the schematic page editor and the part editor.
3. Click OK.

Shortcut

Toolbar: 
Keyboard: I

Zooming out

In the schematic page editor and the part editor, you can change your viewing perspective to increase the portion of the schematic page or part that is visible.

To zoom out

Choose *View – Zoom*, then choose the Zoom Out command. The current zoom scale is divided by the zoom factor. So, for example, a zoom factor of 2 causes Capture to halve the image size and show twice the area of the previous view.
OR
Hold down the Ctrl key and roll down the mouse wheel.

To change the zoom factor

> At certain zoom scales, Capture substitutes filled rectangles for text that is too small to appear. These placeholders are for display only—the text prints correctly.

1. Choose *Options – Preferences*, then choose the Pan and Zoom tab.
2. In the Zoom Factor text box, enter the new zoom factor. Note that you can specify separate values for the schematic page editor and the part editor.
3. Click OK.

Shortcut

Toolbar: 
Keyboard: O

Zooming to a specific scale

To view a part or schematic page at a specific scale

1. Choose *View – Zoom*, then choose the Scale command.
2. Select a preset scale or enter a custom scale, then click OK.

Changing the zoom factor

When you zoom in or out, the zoom scale is multiplied or divided by a zoom factor that you can set to suit your needs. Furthermore, you can set one zoom factor for the schematic page editor and another for the part editor.

To change the zoom factor

1. Choose *Options – Preferences*, then choose the Pan and Zoom tab.
2. Enter the new zoom factor, then click OK.

Viewing the entire schematic page or part

You can view the entire part or schematic page at once. For a schematic page, Capture uses the dimensions set in the Page Size tab in the Schematic Page Properties dialog box.

1. Choose *View – Zoom – All*. The entire schematic page or part is reduced to fit the window.

Shortcut

Toolbar: 

Viewing a specific area

To view an area of the part or schematic page

1. Choose *View – Zoom – Area*. The pointer appears as a magnifying glass.
2. Move the pointer to one corner of the rectangular area to enlarge.
3. Press and hold the left mouse button as you move the pointer to the opposite corner of the rectangular area.
4. Release the mouse button. The selected area fills the window.

Shortcut

Toolbar: 

Centering the view

In Capture, you can center the view on your pointer or you can focus the view on a specific object.

To center the view on a specific object

1. Select objects or an area.
2. Choose *View – Zoom – Selection*. The display scrolls so that the selected objects or selected area is in the center of the window. The zoom factor does not change.

To center the view on your pointer

1. Move the pointer over the area you want to be centered, then press *Shift+C* or just *C*.

Refreshing the display

To refresh the display

1. Go to the View menu.
2. From the Zoom cascading menu, choose the Redraw command.

Shortcut

Keyboard: F5

Setting a bookmark

If you find that you need to return repeatedly to a specific area of a schematic page, or if you need to direct attention to a particular location, a bookmark is very convenient. When you set a bookmark, you assign it a name. You can then use the Go To command to return to the location, and you can use the bookmark name to direct another member of your team to the location.

> The Go To command is always available on the right mouse button context-sensitive menus in the part editor and schematic page editor. The Go To command, with the Relative option selected, is particularly useful for precise placement and spacing.

To place a bookmark

1. From the *Place* menu, choose the Bookmark command.
2. Enter the name of the bookmark, then click *OK*.
3. Position the pointer where you want to place the bookmark and click the left mouse button. The bookmark appears in the selection color.
4. Choose End Mode from the right mouse button pop-up menu.
5. Click an area where there are no parts or objects to deselect the bookmark.

To rename a bookmark

1. Select the bookmark.
2. Choose *E dit – P roperties*. The Edit Bookmark dialog box appears.
3. Enter a new name in the text box, then click OK.

Non-Linear Editor (Fisheye)

The Fisheye feature in Capture allows you work on a schematic in a non-linear mode. The two basic features include the Fisheye focus and the Dynamic Fisheye View mode.

Fisheye focus allows you to set focus to specific objects on your schematic. Setting the Fisheye focus to one or more objects on the schematic ensures that these objects display in a magnified view. As this happens, the other visible objects are not moved off the page but demagnified. This ensures that you still have a view of the page, but with selected focus.

Note: The Fisheye mode is page specific and not design specific. Also, moving in and out of the Fisheye mode will retain the state of the previous mode. You can use all the zoom operations in addition to the focus feature of the Fisheye. All Capture features are available while in this mode.

Note: The Capture find functionality can be used in conjunction with the Fisheye feature. Finding an object on the page will set the focus to the object. If you press Shift + F11 (shortcut) will immediately set the Fisheye focus for the selected object.

In this section:

Fisheye view

You can use the Fisheye mode to zoom into only specific objects on your schematic.

To use the Fisheye features of Capture, you need to switch into the Fisheye mode.

To switch to the Fisheye mode

1. Right-click on the page.
2. Choose the Fisheye view menu item.

Fisheye focus

You can set the Fisheye focus to selected objects on your schematic, causing only these objects to zoom while the rest of the viewable area remain in view but is zoomed out.

To Set the Fisheye focus

1. Select one or more objects on the page. (Use Ctrl + Click to select multiple objects).
2. Right-click on the page.
3. Choose the Set Fisheye Focus menu item.

Shortcut

Keyboard: Shift + F11

To Remove the Fisheye focus

1. Right-click on the page.
2. Choose the RsSet Fisheye Focus menu item.

Shortcut

Keyboard: Ctrl + Shift + F11

Fisheye Dynamic Focus Mode

In the Dynamic Fisheye focus, the focus of the page shifts as you move the mouse pointer across the page. As the mouse pointer hovers over a part of the page, only that part of the page comes into focus. The focus area is magnified while the rest of the viewable area loses relative magnification.
To Set the Fisheye Dynamic focus mode

1. Right-click on the page.
2. Choose the Fisheye Dynamic Focus Mode menu item.

Shortcut

Keyboard: Q

Non-Linear Zoom

While in the Dynamic Fisheye mode, you can further zoom into or zoom out of the view to get higher or lower zoom factor as you pan across the page.

The magnification factor ranges from a minimum of 2 to a maximum of 10. This feature is used in conjunction with the Set Fisheye focus and the Dynamic Fisheye View modes to further zoom into or zoom out of the schematic in a non-linear manner.

To zoom in (non-linear)

Press Ctrl + + (Ctrl and plus key combination)

To zoom out (non-linear)

Press Ctrl + - (Ctrl and minus key combination)

Manipulating Objects in the Editors

In this section:

Moving objects

You can easily change the location of objects in the schematic page editor or the part editor. Immediately after you place an object, you need to select the selection tool or press `Esc` before you perform the steps below.

> Capture uses the location of the first pin on a part to snap to grid. If you move a part without pins, it will be on Fine grid unless you use the `Cut` command described below to place the part on Coarse grid.

To drag an object using the mouse

- Position the pointer on the object. Press the left mouse button and drag the object to the new location. Moving an object this way does not break any of its electrical connections, with the exception of pin or net symbol connections. Otherwise, electrically connected objects are rubberbanded to maintain connectivity.

> If you are dragging a part or wire to another location and that change will affect connectivity, Capture warns you with a changed cursor and temporary markers on your schematic. Visible and off-screen connectivity changes will be saved in the session log if you complete the operation.

To move an object using the mouse

1. Move the pointer over the object.
2. Press `Alt` and click, and then drag the object to the new location.
3. Release the mouse button. The object is placed at the new location. Nets previously

connected to the object are not moved.

To move objects using the Cut command

1. Select the object or objects.
2. From the *Edit* menu, choose the Cut command. The object is placed on the Clipboard.
3. If the object is to be moved to another window, open that window.
4. From the *Edit* menu, choose the Paste command. The object is attached to the pointer.
5. Move the pointer to the location where you wish to place the object and click the left mouse button. The object appears in the selection color.
6. Click an area where there are no parts or objects to deselect the object, or press `Esc`.

> When you move an object in this manner, all occurrence properties are cleared, but instance properties are retained. See Instances and occurrences for more information.

Copying objects

Capture uses the Windows Clipboard to support the standard Cut, Copy, and Paste functions. You can cut, copy and paste information across schematic page or part windows. You can copy text from other Windows applications and paste it into Capture text boxes using the Clipboard. You can also copy a section of your schematic page to another Windows application.

To copy objects using the mouse

1. Select the object or objects.
2. Press and hold both `Ctrl` and the left mouse button while you drag the object to its second location.
3. Release the left mouse button to place the copy.

> - If you are dragging a part or wire to another location and that change will affect connectivity, Capture warns you with a changed cursor and temporary markers on your schematic. Visible and off-screen connectivity changes will be saved in the session log if you complete the operation.
> - Copying projects using Ctrl+drag causes duplicate instances, which creates problems for EDIF netlisting. If you run an EDIF netlist on a design in which you have used this method of copying objects, be sure to use instances when you annotate the design.

To copy objects using the Copy command on the Edit menu

1. Select the object or objects.
2. From the *Edit* menu, choose the Copy command. The object is placed on the Clipboard.
3. If the object is to be copied to another window, open that window.
4. From the *Edit* menu, choose the Paste command. The object is attached to the pointer.
5. Move the pointer to the location where you wish to place the object and click the left mouse button. The object appears in the selection color.
6. Click an area where there are no parts or objects to deselect the object.
7. If you want to place another copy of the object, repeat steps 4, 5, and 6 above.

To copy text or graphics into other Windows applications

1. Select the text or graphic.
2. From the *Edit* menu, choose *Copy*. The selected objects are copied to the clipboard
3. Open the other Windows application and use that application's Paste command to place the clipboard contents.

Shortcuts

Toolbar: 

Rotating objects

Capture objects can be rotated by 90-degree increments. Some objects, such as images, cannot be rotated.

To rotate objects

1. Select the objects.
2. From the *Edit* menu, choose the Rotate command. The selection set rotates 90 degrees counterclockwise. If the `Rotate` command does not appear on the *Edit* menu, the objects cannot be rotated.

Mirroring objects

Capture objects can be mirrored horizontally, vertically, or both horizontally and vertically. Some objects, such as text and images, cannot be mirrored.

To mirror objects

1. Select the objects.
2. From the *Edit* menu, choose the Mirror command. If the commands of the pull-right menu are not available, the objects cannot be mirrored.
3. Choose Horizontally, Vertically, or Both. The objects flip in the indicated direction.

Selecting and deselecting objects

You select objects to edit, move, or alter them in any way. You can simultaneously alter multiple objects if they are all in the selection set. Objects that are selected appear in the selection color. You can also control the selection of objects in a schematic page when you drag the mouse pointer diagonally across the schematic page.

To select an object

- Position the pointer on the object, then click the left mouse button or press the space bar. The object appears in the selection color. Selection handles appear along the boundary box of an object selected in the schematic page editor. If the entire object is selected, all selection handles are the same size. A large handle indicates the point at which the object is selected.

- When you open the part editor from the schematic page editor, the part you are editing cannot be selected on the schematic page. After you close the part editor window, the part can be selected.
- You can re-size an object by selecting it at a single point and dragging.

To select objects that converge at a single location

- Click the point at which the objects converge to select all objects.

To add or remove an object from the selection set

- Position the pointer over the object and press `Ctrl` while you click the left mouse button. All objects in the selection set appear in the selection color. In the spreadsheet editor, this selection method is unavailable because the selection set is limited to contiguous cells.

To control the selection of objects during a mouse-drag operation

1. From the *View* menu, choose the Selection Filter command.
   The Selection Filter dialog box appears.
2. Select the check box corresponding to the object that you want to be selected during the mouse-drag operation.
3. Click *OK*.
   The next time you drag the mouse pointer diagonally across a schematic page, only these objects will be selected in the schematic page.

T o select all objects in an area

1. From the tool palette, choose the selection tool.
2. Move the pointer to one corner of the area. Press and hold the left mouse button while you drag the pointer to the opposite corner, then release the left mouse button. Every object in the selection set appears in the selection color and the set behaves as one object.

> You can specify whether the selection set includes all objects intersected by your selection rectangle or only those objects fully enclosed by the selection rectangle. From the Options menu, choose the
>
> Preferences command, and then choose the Select tab.

To select an entire contiguous polyline

1. From the tool palette, choose the selection tool.
2. Press and hold the left mouse button while you drag the pointer to select an area that includes some portion of the line.

To select all objects on a schematic page or a part

- From the *Edit* menu, choose the Select All command.

To select from among overlapping objects

- Position the pointer over the stack and press the `Tab` key while you click the left mouse button. This cycles through the objects in the stack.

To select all portions of a net on one schematic page

1. Select one segment of the net. The segment changes to the selection color.
2. Right-click to display a context-sensitive menu.
3. From the pop-up menu, choose the Select Entire Net command.

To deselect the selected objects

- Click an area where there is no object or part, or press the `Esc` key. Note that a part

occupies a rectangular area encompassing all its graphics and property text; this means that a part may occupy a larger area than is apparent.

> To deselect an object that you have just placed, you must select the selection tool before clicking the mouse or press ESC to end mode and press ESC again to deselect the object.

To change the selection color

1. From the *Options* menu, choose the Preferences command and then choose the *Colors* tab.
2. Click over the Selection color. The color palette window opens.
3. Select the new color and click *OK* to dismiss the color palette.
4. Click *OK* to dismiss the dialog box.

> When you click on a wire segment, only that segment and its two handles are selected.

Working with Text and Graphics

Text and graphics in schematics are not considered electrical components. That is, they do not have any effect on the netlist generated from the schematic. Text and graphics provide a way for you to document your schematic without effecting its connectivity.

In this section:

Creating graphics

You can create a wide variety of graphic shapes for your parts or to add to your schematic page. You can work with the snap-to-grid option turned on or turned off. For close work, you may want to try Zooming in on your graphic. To draw very precisely, use the Go To command on the *View* menu.

Before you begin drawing, you may want to specify default line and fill styles because all lines and shapes you draw adopt the current line style and closed shapes adopt the current fill style. You can use a variety of line types or fill styles for any schematic page or part.

To change the snap-to-grid option

- From the *Options* menu, choose the Preferences command, then choose the *Grid Display*

tab. You set the option separately for the schematic page editor and the part editor.

To set a default line style

1. From the *Options* menu, choose the Preferences command and then choose the *Miscellaneous* tab.
2. Click on the *Line Style* and *Width* drop box to display the options. Note that you can specify separate options for the schematic page editor and the part editor.
3. Select one of the options and click *OK*. Any lines or shapes you draw will have this line style.

To define a default fill

1. From the *Options* menu, choose the Preferences command and then choose the *Miscellaneous* tab.
2. Click on the *Fill Style* drop-down list to display the options. Note that you can specify separate options for the schematic page editor and the part editor.
3. Select one of the options and click *OK*. Any closed shapes you draw will have this fill style.

To edit line style or fill style of a placed object

1. Select the object.
2. From the *Edit* menu, choose the Properties command.
3. Select another line style or fill style in the dialog box that appears, then click *OK*.

To draw an object

1. From the *Place* menu, choose the appropriate drawing command or select the appropriate drawing tool from the tool palette.
2. Use the mouse to draw the object. To constrain the object by the orthogonality rules, press and hold the `Shift` key while you draw.

Drawing Arcs

You create a circular arc of any angle using the arc tool. Because it is a line, the arc adopts the current line style. For more information about setting line styles, see Creating graphics.
To draw elliptical arcs, see Drawing elliptical arcs.
To create a full circle, use the ellipse tool.

To draw a circular a rc

1. From the *Place* menu, choose the Arc command.
2. Move the pointer to the center of the arc and click.
3. Use the mouse to establish the radius of the arc; click to mark the start the arc.
   The arc is drawn counterclockwise from this start point.
4. Move the mouse along the path of the circle to draw the arc and click to mark the end of the arc.
   The arc appears in the selection color.
5. Choose the selection tool or press `Esc` to dismiss the arc tool.
   Or
   From the *Place* menu, choose the Arc command.
6. Move the pointer to the center of the arc and press the left mouse button.
7. Drag the mouse and then release the left mouse button to establish the radius of the arc and the location of the start of the arc.

The arc is drawn counterclockwise from this start point.

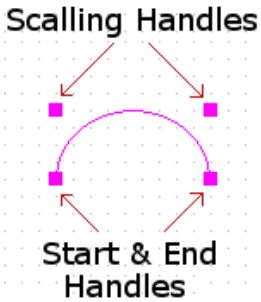
8. Move the mouse along the path of the circle to draw the arc and click the left mouse button to mark the end of the arc.
   The arc appears in the selection color.
9. Choose the selection tool or press `Esc` to dismiss the arc tool.

Shortcu t

Tool palette: 

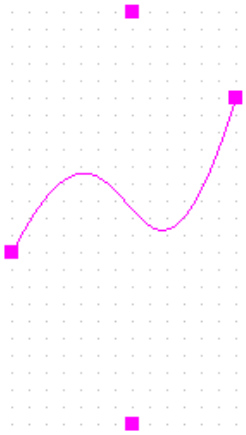
To edit an arc (circular or elliptical)

Use the start and end handles to increase or decrease the size of the arc. You the scaling handles to scale or re-shape the arc.



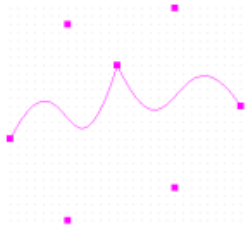
Drawing Bezier Curves

Bezier curves are defined using a start point, two control points and an end point. The two control points define the gradient of the curve. These two points control the shape of the curve. The entire curve is a blend of the four points that make up the curve.



To draw a Bezier curve

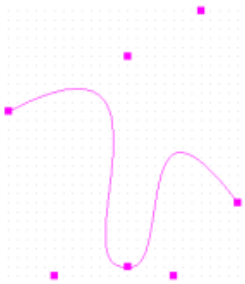
1. From the *Place* menu, choose the `Bezier` command
2. Click on the canvas to mark the start point of the curve.
3. Click to mark the first control point.
4. Click again to mark the second control point.
5. Click to mark the end point of the curve.;
6. Select the selection tool or press `Esc` to dismiss the Bezier curve tool.



Extending the four-point curve

After you have drawn the four points of the Bezier curve, you can mark a fifth point on the canvas. You will notice that the shape from the forth to fifth points is a straight line. Here the end point of the first curve is the start point of a second four-point curve. You can then continue to make any number of four-point Bezier curves, each starting and the end of the previous curve.

You will also notice that the point connecting two contiguous four-points curve forms a sharp edge. If required, select and move this point to smoothen out this edge. You can thus create a curve with any number of control points.



To edit a Bezier curve

1. Select and move the start or end points of the curve to alter the start or end positions of the curve.
2. Select and move the two control points of the curve to alter its gradient.

Shortcut

Tool palette: 

Drawing Ellipses and Circles

You use the ellipse tool to draw an ellipse or a full circle.
Because they are closed shapes, circles and ellipses will have the current fill style. They will also have the current line style. For information concerning line style and fill style, see Creating graphics.
For details on drawing circular or elliptical arcs, see Drawing arcs and Drawing elliptical arcs.

To draw an ellipse or a circle

1. From the *Place* menu, choose the Ellipse command.
2. Move the pointer to an edge of the intended ellipse.
3. Press and hold the left mouse button while dragging the mouse. The ellipse changes shape as you move the mouse. Release the left mouse button when you have the correct shape. To draw a circle, hold down the `Shift` key while you perform this step. The ellipse or circle appears in the selection color.

4. Choose the selection tool or press Esc to dismiss the ellipse tool.
5. Click on an area where there are no parts or objects to deselect the ellipse.

Or

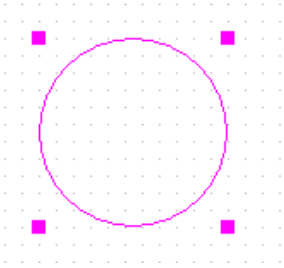
1. From the *Place* menu, choose the Ellipse command.
2. Move the pointer to an edge of the intended ellipse and click the left mouse button.
3. Move the mouse to a new location. The ellipse changes shape as you move the mouse. Release the left mouse button when you have the correct shape. To draw a circle, hold down the SHIFT key while you perform this step. The ellipse or circle appears in the selection color.
4. Choose the *selection tool* or press Esc to dismiss the ellipse tool.
5. Click on an area where there are no parts or objects to deselect the ellipse.

Shortcut

Tool palette: 

To edit an ellipse or a circle

When you select an ellipse, four handles are made visible around the shape. Click and drag any of these handles to alter the shape.



Drawing Elliptical Arcs

You create an elliptical arc of any angle using the elliptical arc tool. Because it is a line, the arc adopts the current line style. For more information about setting line styles, see Creating graphics.
To draw circular arcs, see Drawing arcs.
To create a full ellipse, use the ellipse tool.

To draw an elliptical arc

1. From the Place menu, choose the Elliptical Arc command.
2. Move the pointer to the top or bottom of the arc and click the left mouse button.
3. Move the mouse to the start point; click the left mouse button.
   The arc is drawn counter-clockwise from this start point.
4. Move the mouse along the path of the ellipse to draw the arc and click the left mouse button to mark the end of the arc.
   The arc appears in the selection color.
   Choose the selection tool or press ESC to dismiss the arc tool.
   OR

1. From the Place menu, choose the Arc command.
2. Move the pointer to the top or bottom of the arc and click the left mouse button.
3. Drag the mouse and then release the left mouse button to mark the start of the arc.
   The arc is drawn counterclockwise from this start point.

4. Move the mouse along the path of the ellipse to draw the arc and click the left mouse button to mark the end of the arc.
   The arc appears in the selection color.
5. Choose the selection tool or press ESC to dismiss the arc tool.

Shortcut

Tool palette: 
To edit an elliptical arc see To edit an arc (circular or elliptical).

Drawing lines

You use the line tool to draw a single line. The line you draw adopts the current line style. For information on setting the line style, see Creating graphics.
If you wish to draw a line with multiple contiguous segments, the polyline tool is very convenient.

To draw a line segment

1. From the Place menu, choose the Line command.
2. Move the pointer to the line's beginning.
3. Press and hold the left mouse button while moving the mouse to draw the line.
4. Release the left mouse button to end the line. The line appears in the selection color.
5. Select the selection tool or press ESC to dismiss the line tool.
6. Click an area where there are no parts or objects to deselect the line.

OR

1. From the Place menu choose the Line Command.
2. Move the pointer to the line's beginning.
3. Click the left mouse button
4. Move the mouse, and click the left mouse button again to end the line. The line appears in the selection color.
5. Select the selection tool or press ESC to dismiss the line tool.
6. Click an area where there are no parts or objects to deselect the line.

Shortcut

Tool palette: 

Drawing polylines

When you wish to draw a line with multiple contiguous segments, the polyline tool is very convenient. The line you draw adopts the current line style. Polygons can be created with the polyline tool; these polygons adopt the current fill style. For information on setting the line style, see Creating graphics.
Drawing polylines behaves like placing wires. Polylines automatically default to drawing with square corners. You can draw non-orthogonal polylines simply by holding SHIFT while you draw.

To draw a polyline

1. From the Place menu, choose the Polyline command.
2. Click the left mouse button to begin drawing, click to change directions, and double-click to end the final segment.
   To constrain the direction changes to multiples of 90 degrees, press SHIFT. After you

double-click, the polyline appears in the selection color.
3. Click an area where there are no parts or objects to deselect the polyline.
4. Select the selection tool or press ESC to dismiss the polyline tool.

To draw a polygon

- Follow the instructions above, ending the line with a single mouse-button click at the beginning point. The polygon adopts the current line and fill style.

Shortcut

Tool palette: 

Drawing Rectangles and Squares

You use the rectangle tool to create orthogonal shapes; if you wish to create a polygon, use the polyline tool.
Any rectangles or squares you create will have the current fill style and line style. For information concerning line type and fill style, see Creating graphics.

To draw a rectangle or a square

1. From the Place menu, choose the Rectangle command.
2. Move the pointer to one corner of the intended rectangle.
3. Press and hold the left mouse button while you drag the mouse.
4. The rectangle changes shape as you move the mouse.
5. Release the left mouse button when you have the correct shape.
6. To draw a square, hold down the `Shift` key while you perform this step. The rectangle or square appears in the selection color.
7. Choose the selection tool or press `Esc` to dismiss the rectangle tool.
8. Click on an area where there are no parts or objects to deselect the rectangle.

OR

1. From the Place menu, choose the Rectangle command.
   Move the pointer to one corner of the intended rectangle and click the left mouse button.
2. Move the mouse to a new location. The rectangle changes shape as you move the mouse.
3. Click the left mouse button when you have the correct shape.
4. To draw a square, hold down the `Shift` key while you perform this step. The rectangle or square appears in the selection color.
5. Choose the selection tool or press `Esc` to dismiss the rectangle tool.
6. Click on an area where there are no parts or objects to deselect the rectangle.

Shortcut

Tool palette: 

Placing IEEE symbols

You can place IEEE symbols directly onto your schematic to represent mechanical components. Again, remember that these symbols do not have an effect when you generate a netlist for your schematic.

To place an IEEE symbol

1. From the Place menu, choose the IEEE Symbol command. The Place IEEE Symbol dialog box appears.
2. From the Symbols list, select a symbol. The symbol appears in the preview box.
3. When the appropriate symbol is selected, click OK. The symbol is attached to your pointer.
4. Use the mouse to move the symbol and click the left mouse button to place the symbol.
5. Select the selection tool to dismiss the symbol tool or repeat step 4 to place additional symbols.

Shortcut

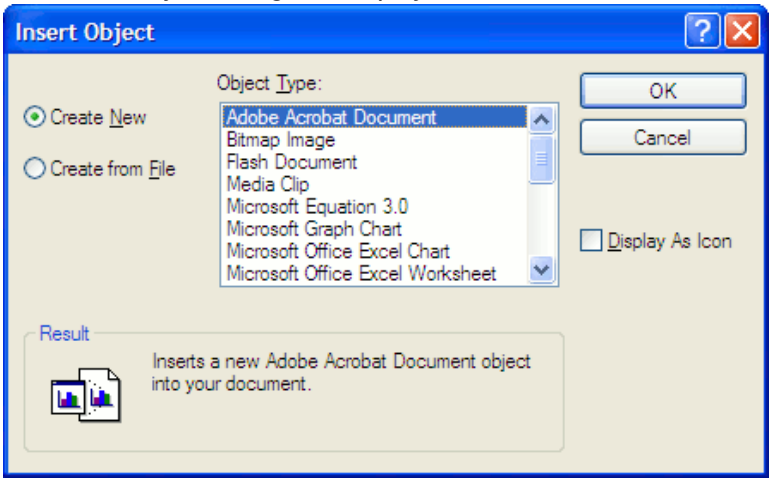
Tool palette: 

Placing OLE Objects

You can place objects of other applications on your schematic page using the OLE Object command. This allows to you embed or link an external application file into your schematic page. This provides the ability to package other files (or links to files) along with your schematic.
For example, you may want to embed a PDF document on a schematic page. Or you want to place and link an Excel document on a page.
Capture allows you to add an existing external file to your schematic as an OLE object. Alternatively, It allows you to add a new instance of an application file.

To place a new OLE object on a page

1. From the Place menu choose OLE Object.
   The Insert Object dialog box displays



   The Create New radio button is selected by default.
2. From the Object Type list, choose the new object type to embed on the schematic page. Choose only object types for which you have the associated application installed on your computer.
3. Click OK.
   The cursor changes into the cross-hair cursor indicating that Capture is now in the Place OLE Object mode.
4. Click the mouse button at the point where you want to start the object and drag the cursor to draw a rectangular area to contain the object.
   Notice that as soon as you release the mouse button, the Capture toolbar now includes the

toolbar for the application associated with the OLE object type. For example, if you select Bitmap Image type, the Capture toolbar includes the Microsoft Paint toolbars.
You can now make changes to the OLE object using the associated application toolbar and edit features from within Capture.

5. When you are done making changes to the new object, click anywhere on the schematic page outside the object.\
Notice now that the toolbars for the associated application are not available in Capture.

The OLE object is now embedded in your schematic page and the contents will be saved along with your schematic page.
To edit the OLE object, double-click on object in the schematic page, the associate application toolbars are included within Capture and you can now make changes to the object.

To place an existing OLE object on a page

1. From the Place menu choose OLE Object.
The Insert Object dialog box displays
The Create New radio button is selected by default.
2. Choose the Create From File radio option.
You can either enter the name of the file or you can browse for the file on the file system.
Choose the Link option to embed a reference of the file, else the file itself is embedded on your page.
Choose the Display as Icon option to display the icon for the application associated with the file.
3. Click OK.
The cursor changes into the cross-hair cursor indicating that Capture is now in the Place OLE Object mode.
4. Click the mouse button at the point where you want to start the object and drag the cursor to draw a rectangular area to contain the object.
Notice that as soon as you release the mouse button, the Capture toolbar now includes the toolbar for the application associated with the OLE object type. For example, if you select Bitmap Image file, the Capture toolbar will include the Microsoft Paint toolbars.
You can now make changes to the OLE object using the associate application toolbar and edit features from within Capture.
5. When you are done making changes to the new object, click anywhere on the schematic page outside the object.
Notice now that the toolbars for the associated application are not available in Capture.
Capture is now in the schematic page editor mode.

The file you selected is either embedded or linked on your schematic page.
If you choose to embed a file, any changes to the original file will not be reflected on the OLE object on the schematic page. However, if you link the file, any changes you make on the OLE object on the schematic page will be reflected on the file (available on the file system). Also, any change you make on the file on the file system will be reflected on the OLE object on your schematic page.

Placing Pictures

You can create a image in another application and place it on a schematic page or library part, or in a custom title block.

To place a n image

1. From the Place menu, choose the Picture command.
   The Open dialog box appears.
2. Select the image file. If the file is not listed in the File Name box:
   - In the Look in box, select a new drive, directory, or both.
   - In the Files of type box, select the type of file you wish to open.
3. Click OK. A rectangle representing the image is attached to the pointer.
4. Use the mouse to move the image and click the left mouse button to place the image at the desired location. If you wish to place multiple copies of the image, simply repeat this step.
5. Select the selection tool to dismiss the picture tool.

> Capture supports many different image file formats these include: BMP, JPEG, JPG, JPE, JFIF, GIF and PNG.

Placing Text

You can place text on a schematic page as a means of providing comments or descriptions on the page.

To place text on a schematic page

1. From the Place menu, choose the Text command.
   The Place Text dialog box appears.
2. Enter the text to place on the page.
   The Place Text dialog box also provides options to alter the text that you are placing on the page.
   a. Use the Color drop-down list to specify the text color.
   b. Use the Rotation option group to specify the orientation of the text on the page.
   c. Use the Change button in the Font group to change the font of the text.
3. Click OK.
   The text is immediately attached to the cursor.
4. Click on the schematic page where you want to place the text.

Shortcut

Tool palette: 

The Property Editor

OrCAD Capture includes a Property editor that allows you to view, edit and add properties to objects in a project.

Using the property editor, you can edit properties for instances or occurrences of:

- Parts (including hierarchical blocks)
- Nets (including constituent nets within buses)
- Pins
- Title blocks
- globals
- Ports
- Aliases

Each column in the property editor is a property. Each row is an instance or occurrence. Occurrence rows appear in yellow below their associated instance row. They only appear if you expand the instance by clicking the plus sign (+) to the left of the instance name, or if an occurrence property of an object is different from the instance. The cells in the property editor show the property values for each instance or occurrence. If a white cell contains hash marks, the corresponding property does not have an instance value causing the library definition of the property to "shine through" to the instance. If a yellow cell contains hash marks, the corresponding property does not have an occurrence value, causing the instance value to "shine through" to the occurrence.

The properties that appear in the property editor depend on the items selected in the schematic page. Also, these properties depend on the tab selection at the bottom of the property editor. For example, if the Parts tab is active, the properties for selected parts appear in the property editor.

Note: When you first start the property editor, all instance properties appear. Occurrence properties appear only if they have their own values assigned to them (independent of the instance property values).

You can also constrain the set of displayed properties by using the filters available in the drop-down list in the upper right of the property editor. A number of filters are available. These filters are sets of properties that are typically useful for particular project types. For example, the Actel Designer Part/Net Properties filter includes properties that are useful for constraining a PLD project for integration with Actel Designer software. The <Current properties> filter causes the property editor to display all properties that currently exist for the selected item.

When editing properties in the property editor:

- Property values that are applied to instances will "shine through" to all occurrences of those instances, unless an occurrence has a value (independent of the instance value) for a particular property.
- Occurrence property values override instance property values.
- When you delete an instance property, that property will no longer "shine through" to its occurrences.
- Deleting a property value from an occurrence causes the instance property value to "shine through" to that occurrence.
- Library definitions will "shine through" to the instance and occurrence of the object only if the instance or occurrence value is unedited.

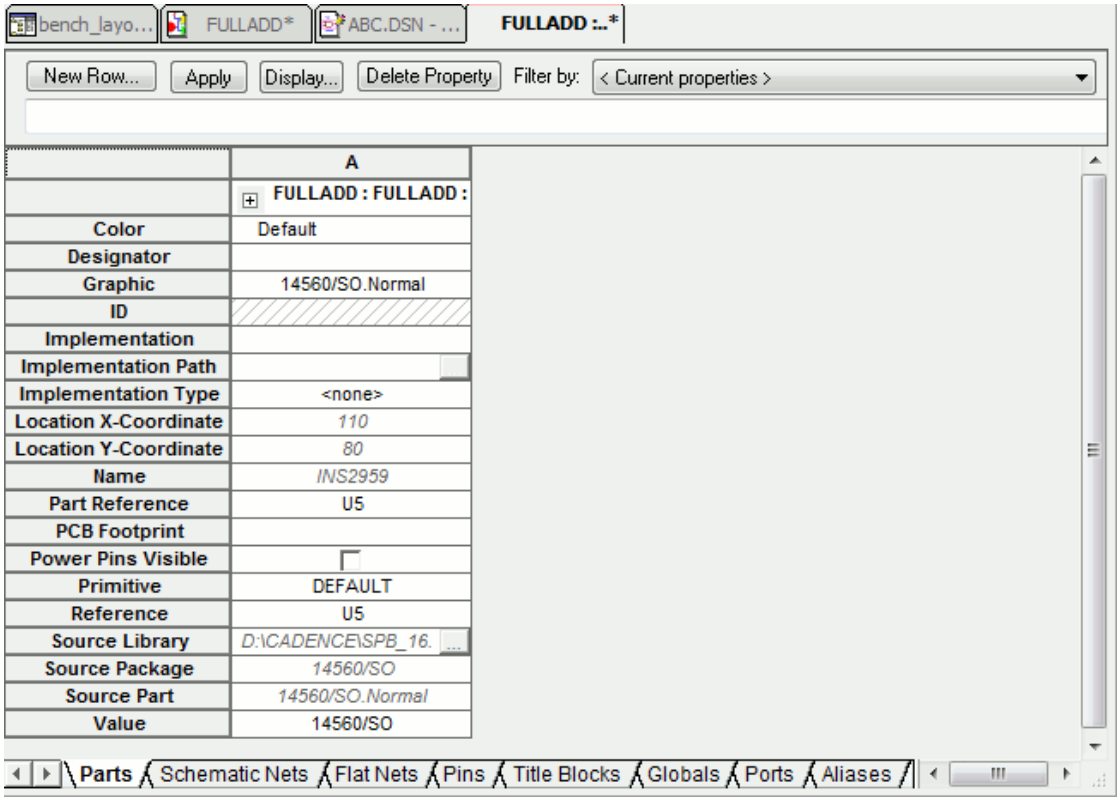
> To browse and edit properties for an entire design, see the Browse spreadsheet editor.

In this section:

The Property editor window

The property editor window appears when you select some combination of parts, nets, pins, title blocks, aliases and globals in the schematic page editor, and then choose Properties from the Edit menu or choose Edit Properties from the pop-up menu. You can use the property editor window to edit part, net, pin, title block, global, port, and alias properties. The property editor displays all

library definitions, instance properties, and occurrence properties for an object.



New Column or New Row

Displays the Add New Column or Add New Row dialog box, depending on the property editor orientation, to add a new property column or row. The property is added to a single object if a single cell is selected or to several objects if several cells are selected. To add a property to more than one object, either shift-select the cells or select an entire row/column by clicking on the property name. To add the property to an object, you must enter a property value for a given object.

> If no cell is selected on the sheet, then the value field is not activated in the Add New Column or Add New Row dialog box and the user is unable to add a new property unless the user manually enters values for the new fields in the property spreadsheet.

Apply

Applies the changes in the property editor to the schematic page. The Apply button does not dismiss the property editor. You can also apply the changes to the schematic page by closing the property editor.

Display

Displays the Display Properties dialog box to set the display option of the selected property and its value. You cannot display properties of an occurrence property using the Display Properties dialog box.

Delete Property

Deletes the editable property from the selected object or objects. (Properties that are not editable appear in italics.) If you select all of a property's cells and click the Delete Property button, the property will be removed from the selected objects but will remain in the filter. This is indicated by the hash marks that appear in the cell.

Filter by

Specifies a filter by which to view the objects. Use the property editor filter to constrain the available properties. For example, the Capture filter displays common schematic capture properties available to most parts, while the Capture-Allegro filter displays the properties needed to send a design to Allegro PCB Editor.The Layout filter displays properties needed to send a design to OrCAD Layout. For information about the Allegro PCB Editor properties, see document *Allegro Platform Properties Reference* guide (*propref.pdf*).
You can view all the properties available on the objects in the property editor by selecting the <Current properties> filter from the drop-down list.
Another example of constraining properties is using the *Allegro_Signal_Flow_Routing.* This filter setting lets you view signal flow properties, such as PROPAGATION_DELAY, RELATIVE_PROPAGATION_DELAY, RATSNEST_SCHEDULE, and DIFFERENTIAL_PAIR in the Flat Nets tab.

Column Value Editor

Displays and allows change of value in selected column. As a result, you do not need to resize the column widths to be able to view values that are larger than what can be accommodated in a column.

Parts

Displays the part properties of the selected objects. The Parts tab includes hierarchical blocks. You can use the Parts tab on the property editor to add and delete property instances and occurrences and to change their values.

Note: All property editor tabs provide one row of property information per instance or occurrence.

The Graphic property column provides the option to toggle the display of the part between Normal and Convert view. When you click the Graphic cell, a down arrow appears indicating a drop-down list. You can change the graphic's appearance on the schematic by clicking the down arrow and selecting a different view.

Schematic Nets

Displays the schematic net properties of the selected objects. This tab includes constituent nets within buses.

Pins

Displays the pins of the selected objects. This tab includes hierarchical pins in hierarchical blocks.

Title Blocks

Displays the title block properties of the selected objects.
With the Title Blocks tab selected, you can add a property to the Title Block instance on a schematic page that will display the full hierarchical path to the schematic.

Globals

Displays selected globals for simultaneous editing of multiple names.

Ports

Displays source symbol, source library, and type of port properties. Provides for simultaneous editing properties of multiple ports.

Aliases

Displays color, font, name, and rotation and other properties of net aliases. Use the Aliases tab to edit multiple aliases at one time.

Rows and columns

In the property editor, each row displays an instance or an occurrence of an object. Instance rows appear with a white background. Occurrences appear in yellow below their associated instance row. Occurrence rows automatically appear when one or more of the occurrence property values are different from the instance property values.
Each column is a placeholder that you can use to add properties. The cells in the property editor show the property values for each instance or occurrence. A cell with hash marks in indicates that the property does not exist on the object that the cell represents. You can add a value by clicking inside the cell, typing the value, and pressing ENTER or clicking the Apply button. A property value in italics is a read only property cannot be edited.

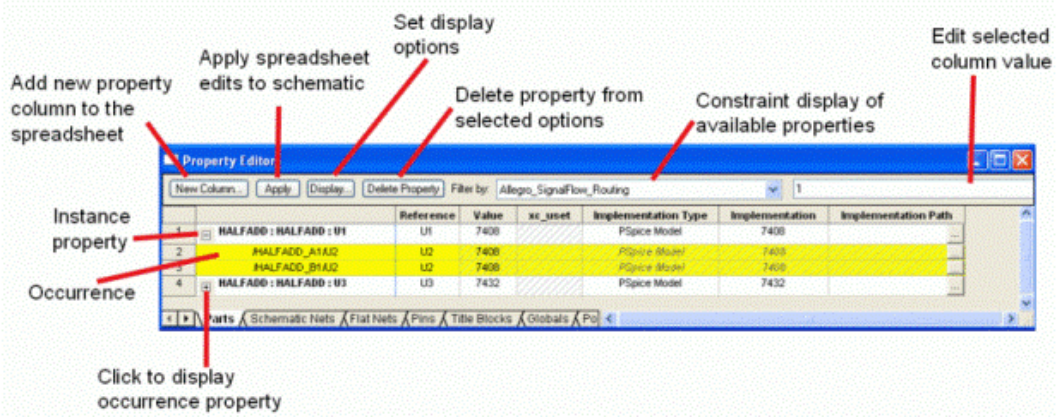
Roll the mouse wheel up and down to scroll through vertically in the Property Editor.

Hold down the CTRL key and roll the mouse wheel to zoom in and zoom out.

Hold down the SHIFT key and roll the mouse wheel up and down to scroll through horizontally in the Property Editor.

Click the mouse wheel button and drag the mouse wheel:

- To the right or left in the Property Editor window to scroll horizontally.
- Up or down in the Property Editor window to scroll vertically.



Changing the appearance of the property editor

You can edit the properties of a group of similar objects using a spreadsheet editor. The property editor is a spreadsheet editor you can use to edit instances or occurrence properties.
Any changes you make to the property editor spreadsheet appearance, such as sorting or moving columns or pivoting the spreadsheet, are saved to PREFPROP.TXT when you close the property editor window. The next time you open to that particular tab in that particular filter, you will see your last settings. This does not apply to any changes you make to the spreadsheet appearance while in the <Current properties> filter.

Pivot the Property Editor Spreadsheet

1. Right-click the empty cell in the top-leftmost position of the spreadsheet.
2. From the pop-up menu, choose Pivot.
   OR
   Double-click the empty cell in the top-leftmost position of the spreadsheet.

   - The default orientation of the spreadsheet shows property columns and instance and occurrence rows. You can add a new property by selecting the New Column button and entering a property name.
   - If you pivot the spreadsheet, instances and occurrences appear in columns across the top, and properties appear in rows. This may be advantageous if your selected object or objects have several properties.

> The Find command searches down columns in the spreadsheet, regardless of the spreadsheet orientation.

Display Or Hide all Occurrence Properties with a Single Key Stroke

- Press and hold the CTRL key while clicking on one of the plus (+) symbols in the leftmost column.
  If you press and hold the CTRL key while clicking on a dash (-) symbol in the left column, all the occurrences collapse so that you only see the instance properties of your design.

Move Columns in the Property Editor

1. Click the title cell of the column you want to move.
2. Release the left mouse button.
3. Click the title cell of the column again and drag the column to the new location.

> If the spreadsheet is pivoted, use these steps to move rows.

Sort Columns in the Property Editor

1. Right-click on the column heading. A pop-up menu will appear.
2. Select Sort Ascending or Sort Descending.
   OR
   If the spreadsheet is not pivoted, double-click on the column heading to toggle the sort order between ascending and descending.

Change Column Widths in the Property Editor

1. Move the cursor to the right edge of the title cell of the column you want to resize.
2. When the down arrow changes to a double-sided arrow, click and drag the column edge.

> You can hold down the shift key while dragging the column edge to resize all columns to the same width. When the property editor is pivoted, these changes are saved on the tab and filter you were using.

Create a New Column Or Row in the Property Editor

1. In the schematic page editor, select the object or objects for which you want to create the property.
2. On the Edit menu, choose the Properties command. Capture displays the property editor.
3. Click the New Column/New Row button. Capture displays the Add new column or row dialog box.
4. Enter a name for the new property and click OK. Capture adds a new column or row to the property editor. Adding values in the cells of that column or row adds the property to selected objects.

Edit a Property Value

1. In the property editor, select the cell or group of cells that contain the value you want to change.
2. Right-click and choose Edit from the pop-up menu. The Edit Part Properties dialog box appears.
3. Type in the new value and press ENTER. Note that changing an instance property value causes that value to "shine through" to all occurrences of the instance that do not have a value independent of the instance.

> You cannot delete some property values that have particular significance to the design. Properties that are not editable appear in Italics.

Globally Edit a Property Value On Selected Objects

1. In the property editor, click the top-leftmost cell to select the entire spreadsheet.
2. Right-click and choose Edit from the pop-up menu.
3. Select a property cell in the Edit Part Properties dialog box spreadsheet.
4. Type the new value and then click OK. The new property value appears on the spreadsheet for all selected objects.

Using the Filters menu in the property editor

The property editor filter is a powerful editing tool with which you can show or hide properties on selected objects. You can use the pop-up Filters menu on the spreadsheet to view the status of a property or edit columns, tabs, or the entire property editor spreadsheet.
You can add, delete, or change any filter except the <Current properties> filter. The <Current properties> filter displays all properties as undefined until you create or select another filter.
When you create a new filter, all properties appear undefined, just as in the <Current properties> filter. If you click the right mouse button on a column heading in the spreadsheet and point to Filters on the popup menu, you will see that each property is Undefined, and the filter specifies to Show Undefined.

Property Edit menu options

Show

The selected column will always appear when you use this filter, unless the filter is inverted.

Hide

The selected column will never appear when you use this filter, unless the filter is inverted.

Optional

The selected column will only appear if the property exists on one or more objects when you use this filter.

Undefined

The selected property is not defined. It is neither included in nor excluded from the filter.
You can control the display of undefined properties on individual tabs of the property editor with the next two choices on the Filters menu. Select any combination of the two.

Show Undefined

Specifies that any undefined property columns that are selected will appear when you use this filter. However, if you also select Invert Filter, these same selections will not appear.
Defined properties appear at the beginning of the spreadsheet (toward the left side) when you select Show Undefined.

Invert Filter

Specifies to show hidden property columns when you use this filter. Conversely, it will hide any property columns that you have specified to show. For example, if a property is optional and does not exist on any objects, you can use Invert Filter to show the property.
The last two menu choices affect all tabs on the property editor.

Add Filter

Specifies to add a new filter to all tabs. The default of a new filter is to show all properties as undefined.

Remove Current Filter

Specifies to delete the filter that displays in the Filter by list box from the list. You cannot undo this operation.
Your results will be more reliable if you use the property editor Filters menu to make changes rather than editing the PREFPROP.TXT file manually.
Changes to the filters are saved to PREFPROP.TXT when you close the spreadsheet. If you need to retrieve the original version, you can copy PREFPROP.TXT from the OrCAD installation CD in the Capture directory.
You can use the property editor filter to narrow the scope of properties it displays. Because you can have hundreds of properties assigned to your parts, nets, pins, and title blocks, it is more efficient to view only the properties you want to see. Capture provides a template that defines the properties that appear if you are targeting your design for PSpice or a board layout tool.

Use the Property Editor Filter

1. From the schematic page, choose Select All command from the Edit menu.
2. Click the right mouse button and choose the Edit Properties command from the pop-up menu. The property editor appears.
3. Click the Parts tab to display all the properties of the parts you selected.
4. Click the Filter by drop-down list down arrow to expand the list of filters, then select a filter.
5. If you chose PCB Editor, for example, the displayed properties change to include those properties you might want to apply to your parts for use in a PCB netlist.
6. Enter a value into one of the cells and click Apply. The property and new values apply to your part.

Create a New Filter in the Property Editor

1. Click the right mouse button on any column heading in the spreadsheet.
2. Point to Filters in the pop-up menu and choose Add Filter.
3. Type the new filter name in the Add Filter dialog box and click OK.
   The new filter will be saved in PREFPROP.TXT when you close the property editor.

Edit a Filter

1. Expand the Filter by drop-down list by clicking the down arrow.
2. Select a filter. The appearance of the properties on the spreadsheet may change when you change the filter.
3. Click the right mouse button on any column heading and point to Filters on the pop-up menu.
4. Use the Filters menu to change the property definitions and appearance on the spreadsheet.

Using the Spreadsheet Editor

The properties of a group of similar objects can be edited using a spreadsheet editor.

Capture provides three spreadsheet editors for editing properties: the property editor, the browse spreadsheet editor, and the package properties spreadsheet editor.

The property editor

Use the property editor from the schematic page editor to edit properties for instances or occurrences of the following objects:

- Parts (including hierarchical blocks)
- Nets (including constituent nets within buses)
- Pins
- Title blocks
- globals
- Ports
- Aliases

Roll the mouse wheel up and down to scroll through vertically in the Property Editor.

Hold down the CTRL key and roll the mouse wheel to zoom in and zoom out.

Hold down the SHIFT key and roll the mouse wheel up and down to scroll through horizontally in the Property Editor.

Click the mouse wheel button and drag the mouse wheel:

- To the right or left in the Property Editor window to scroll horizontally.
- Up or down in the Property Editor window to scroll vertically.

The Browse spreadsheet editor

You can display the Browse spreadsheet editor from the Edit menu of the project manager, schematic page editor, or the part editor.
You can edit the following properties using the Browse command from the Edit menu in the project manager:

- Hierarchical ports

- Off-page connectors
- DRC markers
- Bookmarks
- Parts (including hierarchical blocks)
- Net (including constituent nets within a bus) occurrences
- Pin properties
- Title block occurrences
- Flat nets

  In the schematic page editor, you can edit the following properties in the Browse spreadsheet by using the Properties command from the Edit menu:
- Off-page connectors
- DRC markers
- Bookmarks

  From the part editor (while in Part View), you can edit the following properties using the Browse spreadsheet:
- Pin properties

  The Browse spreadsheet editor browses the entire design for the objects you select, then displays their properties. Each property appears as a column heading in the spreadsheet. Each row is an object located by the editor.

  It is important to note that, in the Browse spreadsheet editor you can edit only occurrences. The only exception being in the part editor, where you can only edit instances. To edit instance properties, you must use the property editor.

To create a new property in the Browse spreadsheet editor

1. In the first column of the Browse spreadsheet, select the object or occurrence for which you want to create the new property.
2. From the Edit menu, choose Properties. Capture displays the object in a new Browse spreadsheet window.
3. Click New. Capture displays the New Property dialog box.
4. Enter a name and value for the new property, then click OK. Capture adds the property to the object or occurrence and displays the property in the original Browse spreadsheet.

To copy a value from one property to another property in the Browse spreadsheet editor

1. In the first column of the Browse spreadsheet, select the object or occurrence that has the property with the value you want to copy.
2. From the Edit menu, choose Properties. Capture displays the object in a new Browse spreadsheet window.
3. Select the cell that contains the value you want to copy.
4. Click Copy.
5. Select the cell that you want to contain the copied value.
6. Click Paste. Capture pastes the value into the selected cell.

> You can use the CTRL + C keys to copy a value from a cell and the CTRL + V keys to paste onto another cell in the Browse spreadsheet editor. Also, you can use the CTRL+ INSERT keys to copy a value from a cell in the Browse spreadsheet editor and paste it onto a cell in Microsoft Excel worksheet or use the SHIFT+ INSERT keys to paste values copied from Microsoft Excel onto a cell in the Browse spreadsheet editor.

To remove a user-defined property in the Browse spreadsheet editor

1. In the first column of the Browse spreadsheet, select the object or occurrence that has the property you want to remove.
2. Select the column heading for the property you want to remove.
3. Click Remove. Capture removes that property from the object.

> - Some properties cannot be removed as they are essential for creating a netlist. You can only remove user-defined properties.
> - If you remove a property from an occurrence for which there is a defined instance property, the occurrence property is not removed, but rather the instance property value "shines through" to the occurrence. To remove an instance property, you must use the property editor.

To replace property values

1. Select the objects whose properties you wish to edit. Note that the objects must be of the same type (for example, all pins or all hierarchical ports); otherwise, the Properties command is grayed out.
2. From the Edit menu, choose Properties. The Browse spreadsheet appears.
3. Double-click on a cell holding the value you wish to replace, then enter the new value.
4. Click the copy button.
5. Select the cells that are to receive the placement value.
6. Click the Paste button. The replacement value appears in the selected cells.
7. Click the OK button to close the Browse spreadsheet.

The Package Properties spreadsheet editor

Use the Package Properties spreadsheet editor to edit package properties of pins.
You can edit package properties using the Package Properties spreadsheet editor. The Package Properties spreadsheet editor is available in the part editor while in Package View. Use the Properties command on the Edit menu to display this spreadsheet. The spreadsheet displays all the package information on pins.
The Package Properties spreadsheet editor is similar to the Browse spreadsheet editor with the following differences:

- The Package Properties spreadsheet editor doesn't have New or Remove buttons. You cannot add properties to a package, or remove existing properties.
- The Package Properties spreadsheet has an Update button and a Validate button.
- The Package Properties spreadsheet displays all of the pins in the package, regardless of what is selected in the part editor.
- The package Properties spreadsheet displays two properties that do not show up in the Browse spreadsheet editor. These properties are PinGroup and Ignore.

Update

Use this to update the properties of all the pins in the package. This button is useful if you change a property on one pin, and need to change this property on the same pin in the other parts of the package. For example, say you have a four-part package. Each part in the package has a pin named IN. If you change this pin from a passive pin to an input pin in the A package part, you could use this button to update the type property for the IN pin in the B, C, and D package parts. The Update button updates all pins at once, without requiring that you click OK.

Validate

Use this button to check for duplicate pins. For example, suppose you have a pin 1, and then change another pin to pin 1. Using this button would detect the duplicate. This button checks for duplicate pin numbers, without requiring that you click OK.

Capture Toolbars

OrCAD Capture user interface includes the following toolbars that provide shortcuts to most of the commonly used commands in Capture

Capture toolbar

The Capture toolbar provides shortcuts for many of the most frequently used commands.

CIS Explorer toolbar

The CIS Explorer toolbar offers a quick and easy way to perform common tasks. This toolbar is active only when you open OrCAD Capture CIS.

Part Manager toolbar

The Part Manager toolbar offers a quick and easy way to perform common tasks. This toolbar is active only when you open OrCAD Capture CIS.

Footprint Viewer toolbar

The Footprint Viewer toolbar offers a quick and easy ways to view the footprint on the canvas in various different angles. It also provides access to the measure tool and the different zoom options in the footprint viewer.

Pspice toolbar

The PSpice toolbar provides shortcuts for many of the most frequently used PSpice commands. This toolbar appears only if you have PSpice license and open a project that uses PSpice.

FPGA toolbar

The FPGA toolbar offers a quick and easy way to simulate, synthesize, and compile vendor library projects. This toolbar is active only when you open a project of type Programmable Logic Design.

Search toolbar

The Search toolbar offers a quick and easy way to perform common search tasks in Capture. This toolbar is active only when you open OrCAD Capture CIS.

Draw toolbar

The Draw Toolbar provides shortcuts for commands to place components, pins, wires, bus, and drawing objects, such as arcs, polyline, ellipse, and text.

Analyze toolbar

The Analyze toolbar provides shorts for the commands to setup your design for use with Signal Explorer.

In this section:

Customizing toolbars

You can customize the toolbars In Capture to alter the look and feel of the toolbar buttons, to create custom toolbars or even add and remove buttons from existing toolbars.

In this section:

Docking Toolbars

The toolbars in Capture can be docked or made floating. This gives the flexibility of placing the toolbar anywhere on the screen. You can place a floating toolbar even outside the application area.
To make a toolbar floating, double-click on the toolbar area (ensure you do not click on any of the toolbar buttons).
To dock a floating toolbar, double-click on its title bar.

To move a toolbar

To move a floating toolbar, click on the title bar and keeping the left mouse button down, drag the toolbar.
You can also right-click on the title bar and choose the Move option. Now you can use the keyboard arrow keys to move the toolbar.

To hide a toolbar

To hide the toolbar, click on the close (X) button on the right side of the title bar.
You can also hide the toolbar by right-clicking the title bar and choosing the Hide option.

Working with multiple windows

In Capture, each document that you open is in a separate window. You may open as many windows as your computer's resources allow. For example, if you wish to work with three schematic pages or three parts, each opens in its own window. If you are working simultaneously with several projects, each opens in its own project manager window.

> All the open documents are tabbed windows. You can right-click the tabs to restore, minimize, maximize, save or close the windows. You can also right-click on the title bar of the Session Log and Project Manager to set them as docked, floating or MDI child windows.

Sometimes it is useful to have more than one view of a document. You might display different areas of the document at different zoom scales, or copy items from one location to another. Capture maintains and displays the selection set across all views of a part or schematic page. For example, you might open a schematic page and use the New Window command to open the schematic page in a second window. You could then shrink the second window and use the Zoom, then the All commands, to create a bird's eye view of the schematic page. Anything you select in the bird's eye view window is also selected in the original window. In the original window, you use the Zoom, then the Selection commands to zoom in on selected objects.
Another way to get a different view of your document is to use the window's splitter bars to split your view, then move objects across the splitter bar and place wires, buses, arcs, and polylines that span the splitter bar. This is very useful for working on large schematic pages.
The Capture work environment is windows based. That is, all documents or schematics appear in their own windows when open for editing.

To open a window on the active document

- Choose *Window – New Window*.

To open a window on another document

- Choose *File – New*.
  OR
- From the File menu, choose the name of a recently used file.
  In Capture, the windows in which you work have headings based upon the name of the open document.

To switch to a different open window

- Click the tab for the window. Alternatively, press CTRL+TAB
  OR
- From the Window menu, choose the window that you wish to make active.

Mouse-over a tab to see the name of the window as tool tip.

To switch to the project manager for the active document

- Click the Project Manager button on the toolbar.

> Click the tab for the project manager if the window is open. Mouse-over the tabs to identify the Project Manager window.

You can right-click the Project Manager title bar to specify the Project Manager to be docked to the main window.

To save all open windows

- From the project manager, choose *File – Save All*. All designs or libraries that have been modified are saved.

When you save a project, Capture automatically creates a backup with a .DBK file extension. When you save a library, Capture automatically creates a backup with a .OBK file extension. If you save only a schematic page or a part, no backup is generated.

To close all open windows

Choose *Windows - Close All Windows* to close all open windows.

To close all open tabs

Right-click on any open tab and choose *Close All Tabs*. This closes all schematic pages open for the project. The Close All Tabs option is not available for the project manager tab.

To close all open tabs except one

Right-click on the tab you do not want to close and choose *Close All Tabs But This*. This closes all schematic tabs for the project except the selected tab.

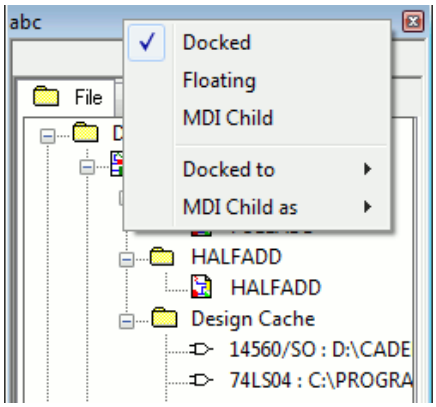
Setting the Window State

OrCAD Capture provides options to position and orient the different windows in the Capture workspace. This is a convenient feature when you are working with multiple windows simultaneously.

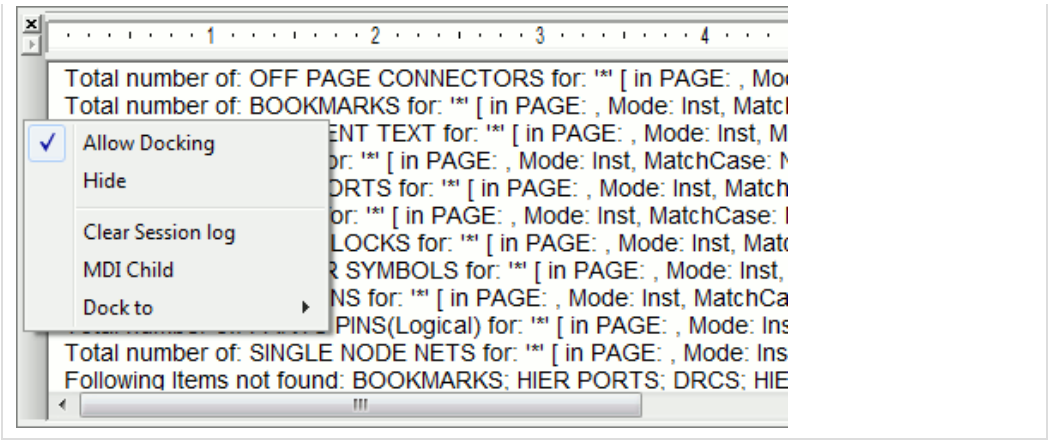
For example, you might need to have the Project manager, the Schematic page editor, and the session log. So might want to place the Project manager on the left, the schematic page editor to the right, and the session log at the bottom of your Capture workspace.

To allow you to move and arrange the Capture workspace, you can dock the windows, set them as floating, or as MDI child windows.

To access these menu options right-click on the title bar in the case of the Project manager, schematic page editor, schematic part editor, and the property editor window.



You can also access these options by right-clicking on the side-bar of the session log.



Searching in Capture

In Capture, you can search for specific comment text on a part, or you can search for a pin by name or by one of its property values.
Using the Find command and a part property value, you can locate a part in a schematic folder or on a schematic page. In the Find toolbar, you enter a property value string and specify that you want to find a part. Capture searches all the parts to find those with a property value that matches the string. You can use question marks (?) or asterisks (*) as wildcards in the property value string.

To locate an object in a project

1. In the project manager, select the schematic folders or schematic pages you want to search.
2. From the Edit menu, choose the Find command. The Find toolbar appears.
   OR
   Press Ctrl+F.
3. In the Text to Search text box, enter the property value string for the part you seek. You must use wildcard characters (standard "*" or "?") with a truncated search. For example, to search for resistors, enter "R*".
   You can also search for a part by property or use regular expressions in your search string.

   > Note that with Regular Expressions search feature enabled, Find performs a complete match for a search string containing alpha-numeric characters, underscore (_) or space unlike the standard regular expression search in TCL. For any other characters or patterns in the search string, standard TCL regular expression search behavior is observed.

4. The Search options drop-down list allows you to specify search criteria.
   You can choose a case-sensitive or case-insensitive search.
   You can highlight the first object found from the search.
   You can choose the type (or types) of objects to search.
5. Click the Search button to start the search.
   Object that have a property value matching the property value string in step 3 are listed in the Find window.
6. Double-click on the part in the Find window list to open the schematic page editor with that part displayed and selected.

Searching a Design Hierarchy

The find functionality in Capture allows you to search at different levels of the design hierarchy:

Design Level

1. In the Project manager, right-click on a design and choose Find.
2. In the Find text box, type the search string and press Enter.
   The search results displayed in the Find window include all objects found within the entire design.

Folder Level:

1. In the Project manager, right-click on a folder and choose Find.
2. In the Find text box, type the search string and press Enter.
   The search results displayed in the Find window include all objects found within the selected folder.

Page Level:

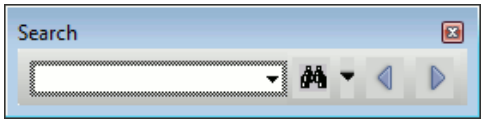
1. In the Project manager, right-click on a schematic page and choose Find.
2. In the Find text box, type the search string and press Enter.
   The search results displayed in the Find window include all objects found on the selected page.
   OR
3. Open the schematic page to search and choose Find from the Edit menu.

Multiple Object selection.

1. In the Project manager, use the Ctrl + mouse click combination to select multiple objects.
   You can select multiple folder or multiple pages or any combination of folders and pages.
2. Right-click on the selection and choose Find.
3. In the Find text box, type the search string and press Enter.
   The search results displayed in the Find window include all objects found within the selected items in the design hierarchy.

Find Toolbar

The find functionality in Capture is available through the Find toolbar.

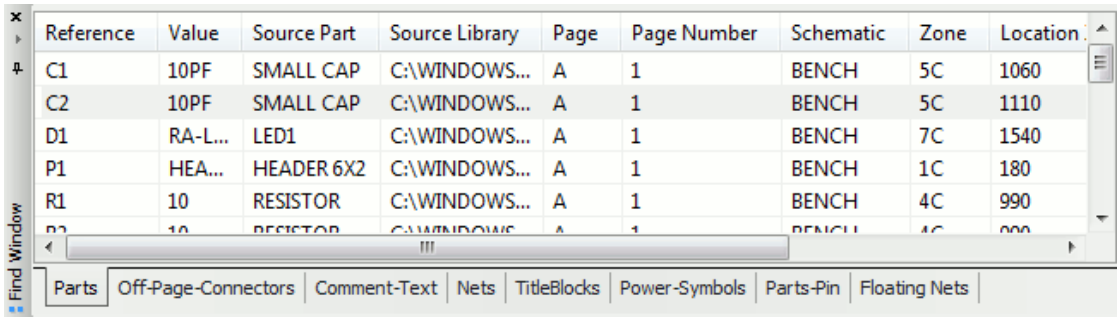


Find text box	Enter the text to search Wildcards: ? - Use the question mark wildcard character to denote one wildcard. E.g. The search for U?A will return U1A and U2A. But not U10A * - Use the asterisk to denote any number of characters. E.g. The search for U*A will return U1A, U2A and U10A.
Find	Run the search command

button	
Find options	This is a multiple selection pop-up list. It contains the search options that you can set to narrow down or broaden your search. This includes all the searchable object types on your schematic. So if you want to search only for parts, ensure that all the other objects types are unselected. Since it is a multiple select list, you can select multiple object types to search.
Find Next	Select the next item in the search list. The next find object will be selected on the open page. Or the page containing the next item is opened with the item selected.
Find Previous	Select the previous item in the search list. The previous find object will be selected on the open page. Or the page containing the previous item is opened with the item selected.

Find Window Search Results

After the search is complete and if it returned at least one result the result is displayed in the Find window. This is a tabbed dockable window.
Each result of the search will display as one line item in the window. A result line item contains other information besides the search object reference. This includes the page and schematic and properties specific to object types.
If the search returns multiple object types, each type displays in a different tab in the window.
If you double-click a line item in this window, the corresponding object is selected on the specific schematic page.
This window can be set as dockable or floating by double-clicking on the title bar. In the docked mode, use the pin icon to alternate the window from pinned to unpinned. In the unpinned state, the Find window remains docked but slides in and out of view as you move the cursor over the window icon.

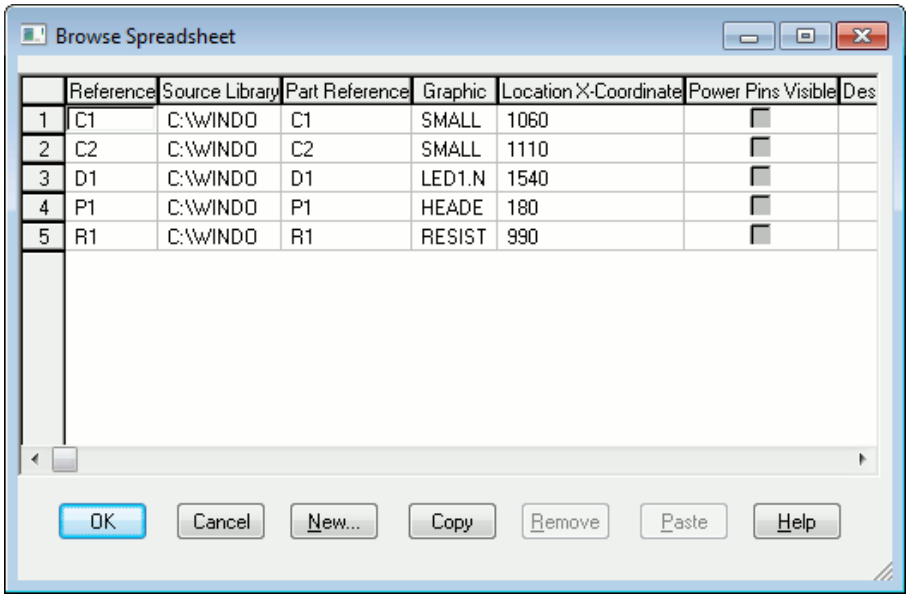


The pop-up menu on the Find window:

Edit Properties

You can open the Browse Spreadsheet window for a selected part in the Find window.

1. Right-click on a search line item.
2. Choose Edit Properties
   The Browse Spreadsheet window displays the editable part properties.



Save as HTML

You can also save your search results in HTML format.

1. Right-click on a search line item.
2. Choose Save as HTML.
   A message displays with the location and name of the exported HTML.

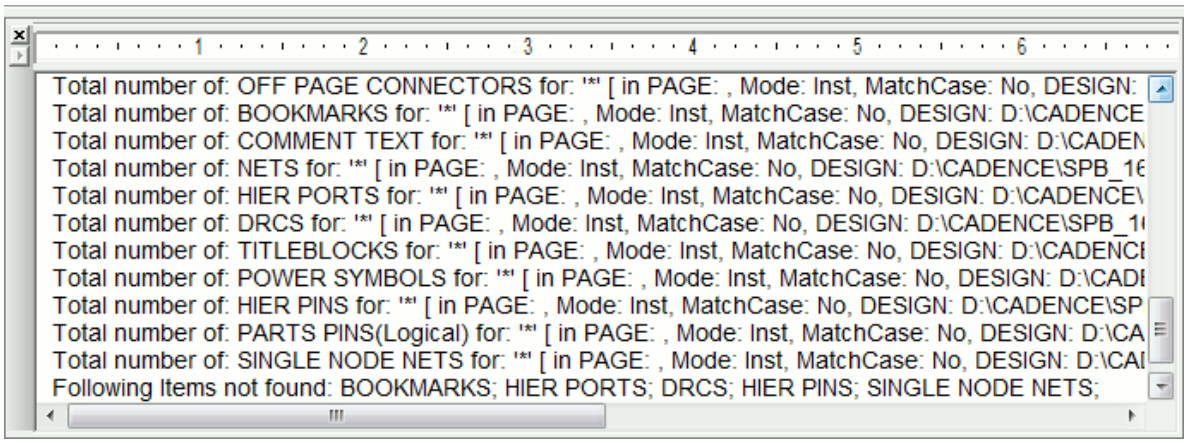
Save as CSV

You can also save your search results of the selected tab on the Find window in CSV format.

1. Right-click on a search line item.
2. Choose Save as CSV.
   A message displays with the location and name of the exported CSV.

Session Log Search Results

The session log displays a log of the search results. This log contains the hit count of the object types for each object selected in the Search options pop-up list. It also logs the pages that were searched.



Browsing in Capture

Using the project manager, you can list objects and sort them with the press of a button. This makes it easy to find, select, and edit objects.
For example, you can list the parts in your design and sort them by part reference or part value. You can list all objects by part value, then add a footprint property to all parts with the same value. When you are debugging your design, you can list all of the error markers and jump to them one by one.
In Capture, you can browse a design-wide list of all objects of one type; you can search for an object by name, or by one of its property values; and you can search a specific schematic page or an entire project.
In the project manager window, Capture will browse for the following object types:

- Parts (occurrence values only)
- Nets (occurrence values only)
- Flat netlist (nets as they appear in a netlist)
- Hierarchical ports
- Off-page connectors
- Title blocks (occurrence values only)
- Bookmarks
- DRC markers
- Variant Parts

The Find command searches for these object types or for comment text.
For information on how to use the Variant Parts option in the Find toolbar, see the *Searching for variant information on a schematic page* section of the *OrCAD Capture CIS User Guide*.

To browse a design

- From the Edit menu, choose Browse, then choose the browse category from the pull-right menu. For each category, the parameters given below appear in the browse window.

Parts	Reference, value, source part, source library, page
Nets	Name, netname, page, schematic folder
Hierarchical ports	Port name, page, schematic folder
Off-page connectors	Connector name, page, schematic folder
Bookmarks	Bookmark name, page, schematic folder
DRC markers	DRC error, DRC detail, DRC location, page, schematic folder

If you double-click an item in the browse window, the schematic page opens with that item selected. Or you can select several items, then choose the Properties command from the Edit menu to open the spreadsheet editor.

To display a list of parts in a library

- Open the library. A list of parts appears in the project manager.
  OR
- From the schematic page editor's Place menu, choose the Part command.

To display a list of parts in the design cache

- In the project manager, double-click on the Design Cache icon.

To list all objects of one type

1. In the project manager, select the documents you want to search. To search the entire design, select all schematic folders.
2. From the Edit menu, choose the Browse command, then choose the object type from the pull-right menu. The browse window displays a list of all objects of the selected type.
3. To display an object, double-click on the entry in the browse window. The schematic page editor opens and the object appears in the selection color.
   OR
   If you wish to edit the properties of one or more listed objects, then from the Edit menu choose the Properties command to display the spreadsheet editor.

To limit the list of objects

1. In the project manager, select the documents you want to search. To search the entire design, select all schematic folders.
2. From the Edit menu, choose the Find command. The Search toolbar displays.
3. In the Text to Search text box, enter a text string that defines the object you are searching. This could be the name, alias, or property value. You can use the standard "*" and "?" wildcard characters.
   The Search toolbar contains a drop-down list that displays the search options that allow you to further refine your search.
4. From the search options drop-down list verify that the Match Case option is as you want it.
5. From the search options drop-down list select the object type.
6. Press Enter or click the search button on the toolbar.
   The Find window displays a list of objects that meet the criteria you specified.
7. If you wish to display an object, double-click on the entry in the Find window. The schematic page editor opens and the object appears in the selection color.
   OR
   If you wish to edit the properties of one or more listed objects, from the Edit menu choose the Properties command to display the spreadsheet editor.
   ALSO
   If you wish to open the Browse Spreadsheet for a part in the Find window, right-click on the part and choose Edit Properties. The Browse Spreadsheet dialog displays.

Capture configuration

In this section:

Capture User Interface

Establishing the configuration for Capture requires that you determine the scope of each configuration setting. Capture provides different levels of configuration. Using commands on the Options menu, you can:

- Customize the working environment specific to your system (set preferences in the

Preferences dialog box).

- Create default settings for new designs (with theDesign Template command). These settings stay with the design even if it is moved to another system with different preferences.
- Override design template settings in individual designs (with the Design Properties command command) or individual schematic pages (with the Schematic Page Properties command).
- Create default settings for new parts (set part properties using the Part Properties command).
- Override default properties on individual parts (set package properties using the Package Properties command).

  No matter where you are in Capture, the Options menu always has a Preferences command and a Design Template command. In addition, the Options menu contains a command specific to the current active window. For example, the project manager's Options menu contains the Design Properties command, while the schematic page editor's Options menu contains the Schematic Page Properties command.

  The settings in the Preferences dialog box determine how Capture works on your system, and persist from one Capture session to the next. However, if you pass a design to another person, that person doesn't inherit your Preferences settings. This means that you can set colors, grid display, pan and zoom, and other things to your liking, and they won't change, even if you work on a design created on someone else's system.

  The Design Template / Design Properties dialog box determines the characteristics of all the designs created on your system. Because a new design inherits characteristics from the current design template, it's a good idea to check the design template settings before you create a new design.

  Once you begin working on a design, you can customize its particular characteristics by choosing the Design Properties command from the Options menu when you are in the project manager, or the Schematic Page Properties command when you are in the schematic page editor.

  Similarly, you use the Part Properties command and the Package Properties command on the part editor's Options menu to set default part properties and customize those settings for individual parts.

Capture.ini File

When Capture starts up, it uses a pre-defiend set of default values for the application settings. These default values are defined in the Capture configuration (Capture.ini) file.
If you are running Capture for the first time on a computer, it uses a pre-defined set of configurations to create the INI file. After this, every time you make any configuration changes, this file is updated when you close Capture.

Location of Capture.ini

By default, the Capture.ini file is created in your HOME directory at the location %HOME%/cdssetup/OrCAD_Capture/<release>. If a INI file exists in your installation, it will be copied to the HOME location. However, you have the option of specifying any other location for the Capture.ini. To ensure that Capture uses the Capture.ini file from an different location, you need to specify the location as a command line argument to the Capture.exe. For details on the command line arguments for Capture, see the Command Line Arguments & Switches section of the Capture Quick Reference guide.

Capture.ini Variables

The Capture.ini file contains a large set of initialization variables used by Capture. Since this file is a text-based file, you have the option to modify or delete the variables and sections in this file. However, any changes you make to this file can cause unexpected behavior in Capture. So you are advised to only make changes as recommened in the Capture Quick Reference guide.

Re-initializing Capture.ini

You can re-initialize all the configuration settings to the Capture default settings by deleting the Capture.ini file. However, these settings will only be available to you after you restart Capture.

Capture CIS Settings

The Capture.ini file also contains the Capture CIS configuration settings. These settings include the CIS parts database setup configuration settings. If you choose to re-initialize your Capture configuration by deleting the Capture.ini, the CIS configuration settings will also be deleted. However, Capture makes a backup of your CIS settigns in a separate file: **BackupCaptureCIS.ini** at the same location as the Capture.ini. This backup is maintained every time you close Capture. So the backup contains the most recent CIS configuration settings.
So, if you delete the Capture.ini and restart Capture, you are prompted with a message to choose if you want to use the backed up CIS configuration settings. If you choose to use the backed up settings, the CIS settings are copied from the **BackupCaptureCIS.ini** to the newly created Capture.ini.

Customizing Menus and Toolbars

You can now customize the menus and toolbars in OrCAD Capture. As a result, any TCL methods that you want to be able to run from the menus is possible. You can also specify your own icons for the menus or toolbars items.

> Pop-up or context-sensitive menus cannot be customized.

The resource files for menus and toolbars including the icons are located at:

```
<Cadence_installation>\share\orResources
```

To add a menu, you need to specify the following information:

- the menu label
- name of TCL method to be called on menu click
- location of the menu in existing menu items
- TCL method to enable and grey out the menu item
- an optional icon

Following is the syntax to add a new menu item:

```
<menuItem name="<menu name tag>">

<type>popup/action</type>

<label>Menu Label</label>

<enabled>true/false</enabled>

   <children>
```

```
      <menuItem name="<child menu tag>">
         <type>action/popup</type>
         <label>Child Label</label>
         <enabled>true/false</enabled>
         <action>TCL Action tag</action>
         <update>TCL Update tag</update>
         <image_16x16>image path</image_16x16>
         <image_24x24>image path</image_24x24>
      </menuItem>
   </children>
</menuItem>
```

You can also add menus dynamically using TCL commands.

3

Working with Projects

A project in Capture refers to the collection of design file, part libraries, report files, and other associated materials that exist, as a set, within the Capture environment.

In this section:

Creating a Project

A project file (.OPJ) is a container for the design file (.DSN). There can be only one design file in a project. The project file stores pointers for interacting with the design file, other referenced files, and outputs reports associated with the design file. The project file can also contain libraries, VHDL files, and information from the various Tools dialog boxes.
When the project is first created, the project manager creates a design file with the same name as the project. It also creates a schematic folder within the design file, and a schematic page within the folder. You can create a new design to replace the design created by the project manager. See Creating a new design.

Project types

When you create a project, you select the project type depending on the type of Capture project you want to create.

Project Type	Description
Analog or Mixed A/D	Analog or mixed signal circuit for designs to be used with OrCAD PSpice.
PC Board Wizard	PC board for designs to be used with PCB board layout tools.
Programmable	Programmable logic project for designs that may include Verilog or VHDL

Logic Wizard	models as part of the structure. Projects of this nature will often use simulation and synthesis tools as part of the design flow. When you create a programmable logic project, certain folders are added to the project. These are discussed in Designing for FPGA .
Schematic	Select this type of project if none of the other project types apply. Using this option, Capture creates a basic project containing only the design file.

To create a project with the project wizard

Capture includes project wizards that provide an easy method for creating specific project types, complete with library and simulation resources.

1. Choose *File – New*, then choose the Project command. The New Project dialog box appears.
2. Type a name for your new project in the name text box.

   > Do not specify a dot (.) in the project name. Also, while using the Save As dialog box to rename the project name, do not specify a dot (.) in the project file name.

3. Use the browse button to select a new directory.
4. Select a project type in the Create a New Project Using group box, and click OK. Continue with the steps below that are appropriate for the type of project you are starting.

> The project types available to you will depend upon which OrCAD programs you have installed. As a minimum, you will have the option to create a PC board or Schematic project type. Be sure to specify the appropriate type when you are creating your project.

Schematic (no simulation)

1. Click the Finish button.

Programmable logic project

1. Select the logic vendor and target family for the project (for example, Altera MAX5 Family).
2. Specify the VHDL models to be used as library models for the project. You can use the default models, provided by Capture, or your own custom VHDL models.
3. Click the Finish button.

Analog or mixed signal project

1. Select the vendor part symbol libraries that you wish to include in your project, and click the Add button.

2. Click the Finish button.

PC board project

1. Select the Enable project simulation check box if you intend to have simulation capabilities in your PCB design. If you selected Enable project simulation, go to step 2. Otherwise, go to step 3.
2. Select the type of simulation resources you want to include.
   - Analog or mixed signal
   - VHDL-based
   - Verilog-based
3. Click Next.
4. Continue with the steps below that are appropriate for the simulation resource you chose.

Analog or mixed signal simulation

1. Select a PSpice symbol library you want to include in your project and click the Add button (or double-click the library name). Continue this step until you have chosen all the libraries you want.
2. Click the Finish button.

VHDL-based simulation

1. Select the PCB part symbol libraries you want to include in your project, and click the Add button.
2. Click the Next button.
   Select the VHDL model libraries that you want to include in your project, and click the Add button.
3. Click the Finish button.

- Typically, referenced projects are FPGA projects that you want to include in your PCB project. This is useful for board simulation that includes the appropriate timing and functionality information for an FPGA that is included in your printed circuit board.
- Some symbol libraries do not have corresponding simulation models.

Changing the Project type

After you create the project with the specific project type, you can use the Change Project Type command to change the project type.

To change the Project type

1. In the Project manager, right-click on the design (.dsn) and choose Change Project Type.
    a. Select Project Type dialog displays the list of Capture project types.
    b. The selected project type is the project type for the current project.
2. To change the project type, select a type from the Associate Project To option group.
3. Click OK.

When you run the Change Project type command, Capture immediately closes the current project. If you want to continue working on the project, you will need to re-open the project.

> The Select Project Type dialog box also appears if you open a design in Capture that does not have an associated project. You then select the project type and Capture creates the project file (.opj) of the specific project type.

Setting Project Preferences

You set project preferences by using the Preferences dialog box. The settings in the Preferences dialog box determine how Capture works on your system, and persist from one Capture session to the next because they are stored in the Capture initialization (.INI) file on your system. If you pass projects to others, they won't inherit your preference settings. This means you can set colors, grid display options, pan and zoom options, and so on to your liking and be assured that your settings will remain, even if you work on a project created on another system.
Once you begin working on a project, you can customize its particular characteristics by choosing Design Properties from the Options menu when you are in the project manager, or Schematic Page Properties when you are in the schematic page editor.

In this section:

Setting colors for objects on the schematic page

Set up colors for objects such as off-page connectors, hierarchical blocks and ports, text, title blocks, and so on, and specify which objects will be printed or plotted. You can also change the background color and the color of the grid.

You control the color in which schematic page objects display by using the Colors/Print tab in the Preferences dialog box.

1. Choose *Options – Preferences*, then choose the Colors/Print tab.

2. Click the left mouse button on the color of an item. The color palette window opens.
3. Select a new color. Click OK to dismiss the color palette.
4. Click OK.

> The color that you select for Title Block is also the color used for borders and grid references.

Graphics objects (lines, polylines, and arcs) use the colors specified by Miscellaneous tab. If the color options in the Miscellaneous tab are set to Default color, then Capture uses the color specified for graphics by the Colors/Print tab.

Controlling the grid display

Select dots or lines for your grid, and whether to display or print your grid. You can select whether to have your pointer snap to grid as you place objects. You can set these options independently for the schematic page editor and the part editor.

Grid spacing is expressed as a fraction of pin-to-pin spacing, as follows:
1/n
where
n = an integer with a value of 1, 2, 5, or 10
So, for example, a setting of 1/2 specifies that the grid spacing on the schematic page is set to exactly half the specified pin-to-pin spacing.

> This setting applies only to the schematic page grid, not to the part and symbol grid.

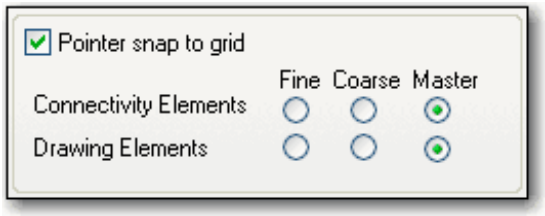
To control the grid

You can control whether Capture displays a grid independently in the schematic page editor and the part editor, and whether the grid uses dots or lines. You can also specify whether the pointer snaps to grid in each editor. Additionally, you can now specify whether the drawing objects, like Line, Polyline, Text, Rectangle, Ellipse, Arc, and Picture can be placed on fine grid.

1. Choose *Options – Preferences*, then choose the Grid Display tab.
2. For the schematic page editor and the part editor, specify:
   - Whether to display the grid.
   - Whether the grid uses dots or lines.
   - The grid spacing (that is, the space between each point on the grid).
   - Whether the pointer snaps to grid as you place objects.
     Note: The pointer by default snaps to the grid for connectivity and drawing objects.
3. Click OK.

Customizing placement and movement of objects on the schematic

OrCAD Capture allows you to customize the placement and movement of connectivity (part, symbol) and drawing objects (Line, Polyline, Text, Rectangle, Ellipse, Arc, and Picture) on coarse and fine grid in the schematic editor. You can use the options (see figure below) provided in the Grid Display tab of the Preferences dialog box to complete this task.



- These settings apply only to the schematic page grid; not to the part and symbol grid.
- The settings are saved in the CAPTURE.INI file and it is used whenever you start the next Capture session.

The connectivity and drawing objects can be individually configured to follow either coarse or fine grid. The following scenarios describe the usage of the above options:

If...	**Then...**
The *Master* option is selected for both Connectivity and Drawing Elements and the *Pointer snap to grid* check box is not selected	The connectivity and drawing objects can be placed and moved only on the fine grid.
The *Master* option is selected for both Connectivity and Drawing Elements and the *Pointer snap to grid* check box is selected	The connectivity and drawing objects can be placed and moved only on the coarse grid.
The *Master* option is selected for Connectivity Elements and the Fine option is selected for Drawing Elements and the *Pointer snap to grid* check box is not selected	The connectivity and drawing objects can be placed and moved only on the fine grid.
The *Master* option is selected for Connectivity Elements and the Fine option is selected for Drawing Elements and the *Pointer snap to grid* check box is selected	The connectivity objects can be placed and moved on the coarse grid and the drawing objects on the Fine grid.
The *Fine* option is selected for both Connectivity and Drawing Elements and the *Pointer snap to grid* check box is either selected or not selected	The connectivity and drawing objects can be placed and moved only on the fine grid.
The *Coarse* option is selected for both Connectivity and	The connectivity and drawing

Drawing Elements and the *Pointer snap to grid* check box is either selected or not selected	objects can be placed and moved only on the coarse grid.
The *Fine* option is selected for Connectivity Elements and the Coarse option is selected for Drawing Elements, and the *Pointer snap to grid* check box is either selected or not selected	The connectivity objects can be placed and moved on the fine grid and the drawing objects on the coarse grid.
The *Coarse* option is selected for Connectivity Elements and the Fine option is selected for Drawing Elements, and the *Pointer snap to grid* check box is either selected or not selected	The connectivity objects can be placed and moved on the coarse grid and the drawing objects on the Fine grid.
The *Coarse* option is selected for Connectivity Elements and the Master option is selected for Drawing Elements, and the *Pointer snap to grid* check box is selected	The connectivity and drawing objects can be placed and moved only on the coarse grid.
The *Coarse* option is selected for Connectivity Elements and the Master option is selected for Drawing Elements, and the *Pointer snap to grid* check box is not selected	The connectivity objects can be placed and moved on the coarse grid and the drawing objects on the fine grid.
The *Master* option is selected for Connectivity Elements and the Coarse option is selected for Drawing Elements, and the *Pointer snap to grid* check box is not selected	The connectivity objects can be placed and moved on the fine grid and the drawing objects on the coarse grid.
The *Master* option is selected for Connectivity Elements and the Coarse option is selected for Drawing Elements, and the *Pointer snap to grid* check box is selected	The connectivity and drawing objects can be placed and moved only on the coarse grid.

- However, if your selection contains both connectivity and drawing objects then precedence will be given to option set for Connectivity Elements.
- Ensure that the *Pointer snap to grid* check box is selected and the Connectivity Elements is set to Coarse while placing connectivity objects. Otherwise, your part pins may be placed on the fine grid, making it difficult to connect them properly.

You can configure Capture to hide the grid or display it as dots or lines, and to constrain the pointer to the grid.

To specify grid style, spacing and visibility

Choose *Options – Preferences* command, then choose the Grid Display tab.
Choose the grid style and spacing, and click the left mouse button on the Displayed or Printed option to change the visibility.
or (to set grid visibility only)
Choose *View – Grid*. The visibility of the grid toggles on or off.

To change snap to grid

1. Choose *Options – Preferences*, then choose the Grid Display tab.
2. Select or clear the Pointer snap to grid option.

OR

- Press CTRL+T.

Setting pan and zoom

Define how you want auto-scrolling to work, and what the zoom factor should be. You can set these options independently for the schematic page editor and the part editor.

Pan

When you hold the left mouse button down and move the pointer near the edge of the window while, the display scrolls to a different region of the document. This change is called panning. This only works if the full schematic is being displayed larger than screen.
The Auto Scroll Percent setting determines the percentage of the screen that changes when panning.

Zoom

When you zoom in or out, the view changes by the zoom factor.

To configure zoom factor and auto scroll percent

1. Choose Options – Preferences, then choose the “Pan and Zoom tab” on page 1050.
2. For the schematic page editor and the part editor, set these options:
3. Zoom Factor. Enter an integer to indicate the magnification or reduction of the objects shown in the window when you zoom in or zoom out. This number is a multiplier for each time you zoom in or out.
4. Auto Scroll Percent. Enter the percent of the window’s horizontal or vertical dimension by which the display will scroll when the pointer approaches the edge of the window with an object attached.
5. Click OK.

Defining selection options

Define whether you want to select objects enclosed by a selection rectangle or objects inside and intersecting a selection rectangle, the maximum number of objects to display at high resolution

while dragging, and whether to show the tool palette. You can set these options independently for the schematic page editor and the part editor.

You can specify whether objects are selected when the selection border intersects them or if the objects are selected only when they are completely enclosed in the selection area.
You can also change the maximum number of objects displayed at high resolution while dragging, and set tool palette visibility in both the schematic page editor, and the part and symbol editor.

To define selection options

1. Choose *Options – Preferences*, then choose the Select tab.
2. For the schematic page editor and the part editor, set these options:
    - Area Select. Specify whether to select objects that are inside and intersecting the selection border or only objects that are fully enclosed by the selection border.

      > If the Fully Enclosed option is selected and you select an object on a schematic page, make sure that you select the object along with its name and number. Otherwise, the object does not get selected.

    - Maximum number of objects to display at high resolution while dragging. If you drag more objects than you specify here, you will see rectangular placeholders for the objects as you drag them.
    - Show Palette. Select this check box to make the tool palette visible; deselect it to make the tool palette invisible.
3. Click OK.

Setting miscellaneous options

Define the default fill, line style and width, and color for graphic objects, define the font used in the project manager and session log, render TrueType fonts with strokes (for printing and plotting), and set whether to auto recover your project and how often. In addition, you can enable intertool communication, which is the method that Capture uses to communicate with other OrCAD software, such as PCB Editor.

Using the Miscellaneous tab:

- specify the default fill, line style and width, and color for graphics objects.
- define the font used in the project manager and session log.
- render TrueType fonts with strokes (for printing and plotting).
- set whether to enable auto recovery for your project and how often.
- enable inter-tool communication (the method that Capture uses to communicate with other OrCAD software, such as OrCAD PSpice and the PCB layout tools).

Fill Style

Specifies a fill pattern for rectangles, ellipses, and polygons.

Line Style and Width

Specifies both line style and line width for lines, poly-lines, rectangles, ellipses, and arcs.

Color

Specifies the color of lines, rectangles, and ellipses in the schematic page editor.

> This color is not the default color, but can be set to use the default color. This option does override the default color. However, changing this setting won't change the color of objects already placed in the schematic page editor.

Polylines and arcs use the default color of objects set in the Colors tab.

> You can change the fill style, line and width style, and color on individual objects using the Properties command on the Edit menu.

Project Manager and Session Log

Specifies the font for the project manager and session log. If you click on this box, a standard Windows Font dialog box for font selection appears. This option is neither a schematic page nor a part editor option.

Enable inter-tool communication

Enables intertool communication with other OrCAD products such as PSpice or the PCB layout tool. For more information about intertool communication, see Intertool communication. This option is not specific to either the schematic page editor or the part editor.

Text Rendering

The text rendering options affect how text on a schematic page appears on your screen, and how it is printed or plotted. The Render TrueType fonts with strokes option displays text as a series of lines, connected to resemble the outlines of the corresponding TrueType letters or numbers they represent.
Enabling the Fill text option causes the text outlines to be filled in.

Auto Recovery

You can specify any interval between five minutes and 120 minutes. When the time interval is up, any design, library, or VHDL file in your project that hasn't been saved, or has been modified since the last save, is saved as a temporary file (with an .ASP extension) in the WINDOWS/TEMP/AUTOSAVE directory.
When you close your project normally, the /AUTOSAVE directory and temporary files are deleted. In cases of power outages or system crashes, however, the temporary files are saved. When you

restart Capture, it loads the auto recovered files, showing “Restored” in their title bars. You must use the Save As command and provide a file name to have an auto recovered file overwrite the original file.

> Auto recovery is not an automatic saving feature. If you intentionally exit Capture without first saving your changes, they will be lost. Auto recovered files are automatically deleted when you exit Capture normally.

To set miscellaneous options

1. From the Options menu, point to Preferences, then choose the Miscellaneous tab.
2. For the schematic page editor and the part editor, set these options:
    - Fill Style. Select the fill pattern to be used when drawing rectangles, ellipses, and closed shapes drawn with the poly-line tool.
    - Line Style and Width. Select the line style and width used for lines, poly-lines, rectangles, ellipses, and arcs.
3. For the schematic page editor, set this option:
    - Color. Select the color used for graphic objects (rectangles, ellipses, and closed poly-lines).
4. Set the following options:
    - Project Manager and Session Log. Select a font for display text in the project manager and session log. If you select this option, a standard Windows dialog box for font selection appears. Select a font, style, and size from the dialog box, then click OK.
    - Text Rendering. The text rendering options affect how text on a schematic page appears on your screen, and how it is printed or plotted.
    - Auto Recovery. Select whether to enable auto recovery for your project and, if so, the interval between saves. You can specify any interval between five minutes and 120 minutes. When the time interval is up, any design, library, or VHDL file in your project that hasn’t been saved, or has been modified since the last save, is saved as a temporary file (with an .ASP extension) in the WINDOWS/TEMP/AUTOSAVE directory.
    - Auto Reference. Select whether to enable automatic annotating of reference designators when parts are placed.
    - Inter-tool Communication. Select whether to enable inter-tool communication (also known as ITC), so that you can test and display design information using other OrCAD software (such as the PCB layout tool and PSpice) in conjunction with Capture. Capture processes its tools faster when inter-tool communication is not selected.
5. Click OK.

Setting Text editor options

Define which (if any) VHDL keywords are highlighted, and the font and tab settings used within the text editor.

Capture text editor options include automatic highlighting of VHDL keywords, comments, or quoted strings. You can enable or disable the highlighting feature, and set the text editor font and tab spacing.

1. Choose *Options – Preferences*, then choose the Text Editor tab.
2. Set these options:
    - Syntax Highlighting. Select the color to use to highlight VHDL keywords, comments, and quoted strings. You can choose a different color for each.
    - Current Font Setting. Click Set to change the font setting for the text editor to values other than those displayed.
    - Tab Spacing. Set the tab spacing for the text editor.
3. Check the Highlight Keywords, Comments, and Quoted Strings option to have those VHDL items highlighted in the text editor. The colors used to highlight these items are the ones set in the Syntax Highlighting group box.
   Note: The Highlight Keywords, Comments, and Quoted Strings option must be enabled for Capture to use the syntax highlighting options.
4. If you want to reset the text editor options to the Capture default values, click the Reset button.
5. Click OK.

Setting up the Design Template

You set the design template using the Design Template / Design Properties dialog box. The Design Template dialog box determines the default characteristics of all the projects created on your system. Because a new project inherits characteristics from the current Design Template settings, it's a good idea to check the settings before you create a new project.
The options that you define in the Design Template dialog box are the default settings for all new projects, and for schematic pages you add to an existing project.

In this section:

Defining fonts for new designs

You can define the fonts for schematic page objects that contain text, such as part references and values. You can define the fonts assigned to the text associated with different schematic page objects in new designs. The fonts specified here do not affect existing designs.

> To change the fonts for an existing project, use the Fonts tab in the Design Properties dialog box. You can access this dialog box by choosing Design Properties from the project manager Options menu.

To assign fonts for new designs

Note: The default fonts were selected for optimal compatibility with SDT. Changing these fonts may result in less than optimal text sizes for translated projects.

1. From the Options menu, choose the Design Template command, then choose the Fonts tab.
2. Click the left mouse button on the font of an item. A standard Windows font dialog box appears.
3. Select a font, font style, and size. Click OK to dismiss the font dialog box.
4. Click OK.

Defining the title block

You can specify the text to appear in title block fields, as well as the path and filename of the library containing the title block. This affects new projects, as well as new schematic pages in existing projects. There are two types of title blocks: default and optional. Capture places one default title block—which you specify on the Title Block tab in the Design Template / Design Properties dialog box in the lower right corner of each new schematic page. You may place any number of optional title blocks anywhere on the schematic page, using the Title Block command on the Place menu.

Default title block

You specify the information that goes into the default title block in the Title Block tab of the Design Template dialog box. Capture places a default title block in the lower right corner of each schematic page (if a library and title block name is specified), and places the information you enter in the text fields in the Title Block tab into the title block.
This information is also used in reports created by the commands on the Tools menu. It affects new projects, as well as new schematic pages in existing projects.
You can set the default title block to be visible or invisible on an existing schematic page by changing the setting in the Grid Reference tab in the Schematic Page Properties dialog box.
Capture provides default title block symbols in the CAPSYM.OLB library.
Not all of the available default title blocks provide the same information. For example, TitleBlock0 doesn't provide any properties for the organization name and address, while TitleBlock5 provides

the organization name property and all five of the address properties. You must specify which title block you want for the default in the Design Template.
The default title block properties that are set in the Design Template dialog box are as follows:

Cage Code	Specifies the Cage Code
Design Create Date	Specifies the date of creation for the design
Design Create Time	Specifies the time of creation for the design
Design File Name	Specifies the path and file name of the design file
Design Modify Date	Specifies the date of the last modification to the design
Design Modify Time	Specifies the time of the last modification to the design
Design Name	Specifies the name of the design
Doc	Specifies the document number
Name	Specifies the name of the title block
OrgAddr1	Specifies the first line of the organization address
OrgAddr2	Specifies the second line of the organization address
OrgAddr3	Specifies the third line of the organization address
OrgAddr4	Specifies the fourth line of the organization address
OrgName	Specifies the organization name
Page Count	Specifies the number of schematic pages in the design
Page Create Date	Specifies the date of creation for the schematic page
Page Create Time	Specifies the time of creation for the schematic page
Page Modify Date	Specifies the date of the last modification to the schematic page
Page Modify Time	Specifies the time of the last modification to the schematic page
Page Number	Specifies the number of the schematic page. The page number determines when it will be printed in relation to the other schematic pages of the design

Page Size	Specifies the page size of the schematic page, as was set at creation time
RevCode	Specifies the revision
Schematic Create Date	Specifies the date of creation for the schematic
Schematic Create Time	Specifies the time of creation for the schematic
Schematic Modify Date	Specifies the date of the last modification to the schematic
Schematic Modify Time	Specifies the time of the last modification to the schematic
Schematic Page Count	Specifies the number of schematic pages in the given schematic
Schematic Page Number	Specifies the order of the schematic page within the schematic
Source Library	Specifies the path and file name of the library from where the title block was placed
Symbol Library	Specifies the name of the symbol for the title block in the Source Library
Title	Specifies the title

You can add the following property to display system generated information:

Path Name	Specifies the hierarchical blocks leading from the root to the child using the Name Property for each hierarchical block in the path

You can use the property editor to add the following property to display the hierarchical path of the schematic on an instance of a title block:

Schematic Path	Displays the full hierarchical path to the schematic visible and printable on the page

> You can create custom title blocks and store them in a library using the New Symbol command from the project manager Design menu. If you specify the name of the custom library and title block in the Symbol group box of the Design Template Title Block tab, the custom title block appears in the lower right corner of each new schematic page.

Optional title block	You can place any number of optional title blocks anywhere on the schematic page using the Title Block command on the Place menu. Optional title blocks display information that you define as property values for the title block symbol

To choose a title block and define the text it contains

1. Choose *Options – Design Template*, then choose the Title Block tab.
2. In the Text group box, enter the information you want to appear in the title block.
3. In the Symbol group box, enter the path and filename of the library containing the title block. The Library Name text box can be left blank if you are using title block from the CAPSYM.OLB library and CAPSYM.OLB has not been moved to a different directory from where it was installed.
4. Enter the exact name of the title block into the Title Block Name text box. Symbol names are case sensitive and space sensitive.
5. Click OK.

> You can create custom title blocks and store them in a library using the New Symbol command from the project manager Design menu. If you specify the name of the custom library and title block in the Symbol group box of the Design Template Title Block tab, the custom title block appears in the lower right corner of each new schematic page.

For Capture to automatically place the information you entered in the text fields into your custom title block, you must give your custom title block the appropriate properties.

You can use the Design Template command of the Options menu to specify information for the default title block. In addition, you can edit title block information in the schematic page editor.

To edit title block information

1. Select the information string on the title block that needs to change.
2. Choose *Edit – Properties*. The Display Properties dialog box appears.
3. Replace the old information with the new and click OK. The schematic page editor appears with the new information in the title block.

   OR
4. Select the title block, and choose *Edit – Properties*. The property editor appears.
5. Place the cursor in the cell of the property you want to change, and enter the new value.
6. Click Apply, and close the property editor. The schematic page editor appears with the new information in the title block.

To edit title block information on multiple pages

1. Select the design file in the project manager.
2. Choose *Edit – Browse – TitleBlocks*.

3. Click OK to dismiss the Browse Properties dialog box.
4. In the Browse window, select the name or names of the schematic pages that contain the title blocks you want to edit.
5. Choose *Edit – Properties* or press CTRL+E. The Browse spreadsheet editor appears. Use the Browse Spreadsheet to edit properties on one or more schematic pages at a time.

To change the display of title block information

1. Select the information string on the title block that needs to change.
2. Choose *Edit – Properties*.
3. In the Display Properties dialog box, choose the Change button. The Fonts dialog box appears.
4. In the Fonts dialog box, change the display properties, and click OK twice.

Shortcuts

Double-click on the information string to display the Display Properties dialog box, or double-click on the title block to display the Properties dialog box

Setting up a page

You can specify whether inches or millimeters are used as the unit of measure, the width and height of a schematic page, and the spacing between pins. For new projects, you can specify the default unit of measure, the default width and height of schematic pages, and the spacing between pins. The value you enter in the Pin-to-Pin Spacing text box defines how close together pins are placed in the part editor. It also defines the grid spacing.
Changing from Inches to Millimeters resets the page sizes to their defaults; therefore, if you make any changes to the standard page size dimensions, then change the units, the page size changes are not translated between the two types of units.

To set up the schematic page size

1. Choose *Options – Design Template*, then choose the Page Size tab.
2. In the Units area, select the default unit of measure for new projects. This setting only affects the schematic page editor, not the part editor.
   Note: Changing from Inches to Millimeters resets the page sizes to their defaults; therefore, if you make any changes to the standard page size dimensions and then change the units, the page size changes are not translated between the two types of units.
3. Select the default schematic page size for new projects. For each schematic page size (A, B, C, D, E, and Custom if the unit of measure is Inches; or A4, A3, A2, A1, A0, and Custom if the unit of measure is Millimeters) you can specify the width and height. The values that you enter in the Width and Height text boxes become the dimensions for each page size. You cannot change these dimensions for individual schematic pages, although you can select a different page size, or choose to define a custom size.
4. In the Pin-to-Pin Spacing text box, specify the default spacing between pins. The value you

enter in this text box defines how close together pins are when you place a part on a schematic page. It also defines the grid spacing (the space between grid dots or grid lines). You cannot change this value for existing projects or individual schematic pages.
Note: Part size will vary when copying and pasting parts between pages with different pin-to-pin spacings.

5. Click OK.

Defining the grid reference

For horizontal and vertical border grid references, you can set the number of border grid references to display in either direction, whether the grid references are alphabetic or numeric, whether they increment or decrement across the schematic page, and how wide grid reference cells are. You can also make the border, grid references, and title block visible or invisible. This affects new projects. You set the border grid references to display either horizontally or vertically, alphabetically or numerically, incrementally or decrementally across the schematic page, and the width of their cells. You can also make the border, grid references, and title block visible or invisible on the screen and on schematic pages you print.
The settings affect new projects.

> You can change these settings for existing schematic pages. Choose Schematic Page Properties from the schematic page editor Options menu, then choose the Grid Reference tab in the Schematic Page Properties dialog box.

To define the grid reference

1. Choose *Options – Design Template*, then choose the Grid Reference tab.
2. Specify the number of border grid references, whether they are alphabetic or numeric, whether the grid references increment (Ascending) or decrement (Descending) across the schematic page, and how wide the grid reference cells are.
   Note: The size of the Grid Reference font is tied to the width.
3. For the border, title block, and grid reference, select Displayed to have the item display on the screen or Printed to have the item appear on schematic pages you print. Select ANSI grid references to display the grid references in accordance with ANSI standards (see the glossary entry ANSI).
4. Click OK.

Specifying the default hierarchy option for new projects

For hierarchical blocks and part instances that have their Primitive property set to Default, you can specify if you want Capture to treat each as primitive (cannot descend into attached schematic folders) or nonprimitive (can descend into attached schematic folders). For hierarchical blocks and part instances that have their Primitive property set to Default, you can specify if you want Capture to treat each as primitive (cannot descend into attached schematic folders) or non-primitive (can

descend into attached schematic folders).
The Primitive and Nonprimitive options only affect new projects.

> This setting affects how the options on the Tools menu process projects.

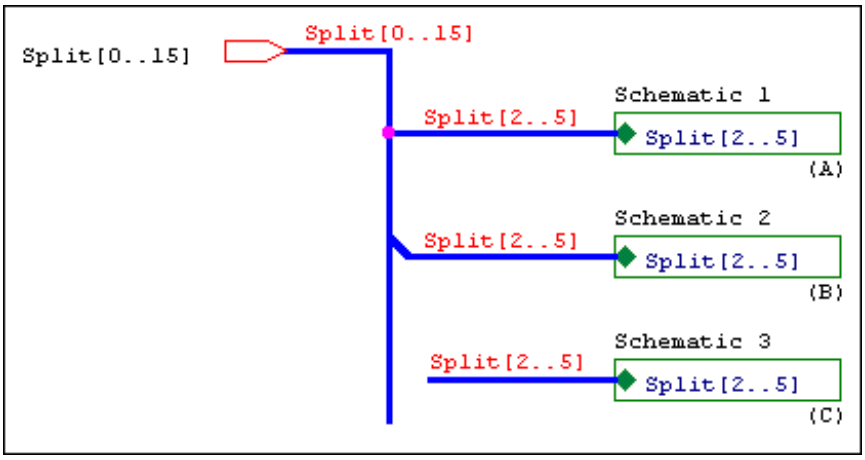
To define the default hierarchy option

1. Choose *Options – Design Template*, then choose the Hierarchy tab.
2. For hierarchical blocks and parts, select Primitive or Nonprimitive. All hierarchical blocks and part instances that have their Primitive property set to Default will use the setting selected here.
3. Click OK.

> You can change the hierarchy option for existing projects using the Hierarchy tab in the Design Properties dialog box. Choose Design Properties from the project manager Options menu.

Setting compatibility with OrCAD Schematic Design Tools

You can specify which Capture properties map to which OrCAD Schematic Design Tools (SDT) part fields when saving a project in SDT format. Capture uses the SDT compatibility options in the Design Template / Design Properties dialog box when you save a Capture design in SDT format. Capture sets the SDT compatibility options in the Design Properties dialog box when you open an SDT schematic folder (.SCH) file in Capture.
Capture uses a slightly different set of connectivity rules than SDT. The following cases explain the differences:
Case 1



The bus is split with like members connecting before and after the split.

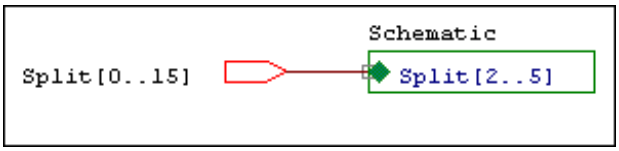
Situation A	The bus is split using a junction.

SDT	Yes
Capture	No—buses connected through a junction must contain the same number of signals

Situation B	The bus is split using a bus entry.
SDT	Yes
Capture	Ye

Situation C	The bus is split without any visible connection, but is connected through name.
SDT	Yes
Capture	Yes

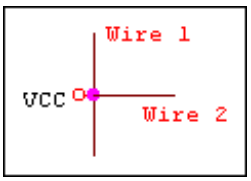
Case 2



The hierarchical port connects to the hierarchical block through a wire.

SDT	Yes
Capture	No—wires in Capture are for single signals only.

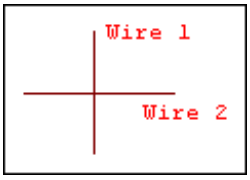
Case 3



The wire connects to the power symbol.

SDT	No
Capture	Yes

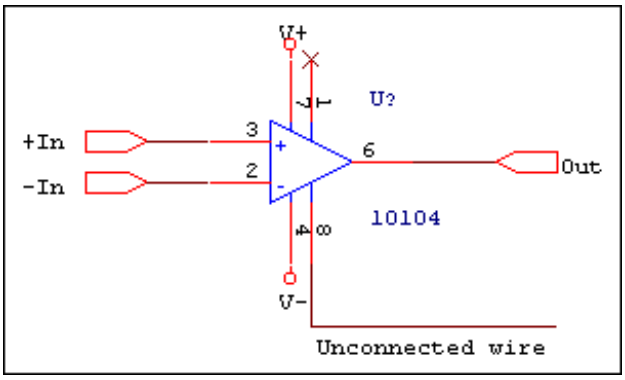
Case 4



Wire 1 connects to Wire 2 through a label hotpoint.

SDT	Yes
Capture	No—wires are connected only if they connect through a junction, or if they share an alias.

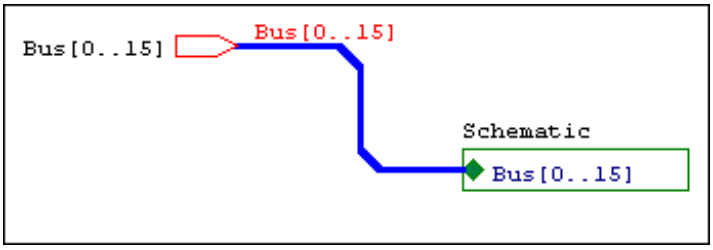
Case 5



The hanging wire connected to a pin causes a single node net in netlists.

SDT	No
Capture	Yes

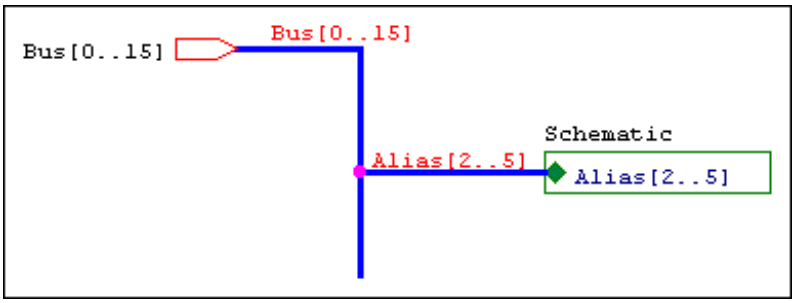
Case 6



Buses routed through bus entries are connected to the target object.

SDT	Yes
Capture	No—you should not use bus entries to route a bus to its target. Use the left mouse button to create turns in the bus route.

Case 7



Unlike bus members are connected.

SDT	No
Capture	Yes

You can specify which properties Capture stores in the eight SDT part fields when saving a project in SDT format. You can also use the part fields for mapping netlists that use part field information.

> To change the part field to property mapping for existing projects, use the SDT Compatibility tab in the Design Properties dialog box. (from the project manager Options menu, choose Design Properties)

To set up compatibility with OrCAD Schematic Design Tools

When you create a new design, the SDT compatibility options are inherited from the design template. Follow these steps to set up the design template for SDT compatibility:

1. Choose *Options – Design Template*, then choose the SDT Compatibility tab.
2. For each Capture property you want mapped to an SDT part field, specify the part field to contain the property value.
3. Click OK.

Changing the SDT compatibility options for a single design

When you save a design in SDT format, Capture uses the SDT compatibility options in the Design Properties dialog box . Follow these steps to change a design's SDT compatibility options:

1. With the program manager active, select the Design folder.

2. Choose *Options – Design Properties*, and choose the SDT Compatibility tab.
3. Specify the properties you want to map to the SDT part fields for the active design.

Translating part fields from SDT to Capture properties

Capture translates SDT part fields into properties. If you want to change the user property names before translation, follow these steps:

1. Open the design's SDT.CFG file in any text editor.
2. Locate the lines that specify the part field names, and change them to suit your needs.
3. Save your changes, and exit the editor.

Translating Capture properties to SDT part fields

You can specify properties for Capture to translate into SDT part fields by following these steps:

1. Choose *Options – Design Properties*, and then select the SDT Compatibility tab.
2. Specify the properties you want to map to the SDT part fields.

Dragging and dropping Folders, Pages, and Parts

You can use the standard Windows drag-and-drop operation to move or copy Capture documents in the project manager windows. If you wish to copy rather than move, you press and hold the CTRL key while you drag the document.
If you drag a part that has a part alias, the part alias also moves. In the context of dragging and dropping, a symbol behaves just as a part does—as shown in the table below, a symbol can be dragged from a design or a library and dropped in another library.
A document that is open in an editor, or one that contains any open elements, cannot be dragged. Documents can be dragged as indicated in the following table:

Drag from . . .	Part	Schematic Page	Schematic Folder
Design to design		X	X
Design to library	X	X	X
Schematic folder to schematic folder		X	
Library to design			X
Library to library	X		X

> If you copy or move a document from one design or library to another, you should save the destination design or library immediately. If you do not, you may lose data if you open the moved document in the schematic page editor or part editor and then close the editor without saving the document.

You can use the standard Windows drag-and-drop operation to move or copy schematic folders, schematic pages, libraries, and symbols between projects, in the project manager windows. If you wish to copy rather than move, press and hold the CTRL key while you drag the entity.

1. If you are moving or copying a folder or page, verify that:
   - for a folder, no Capture editor is open on any document in the schematic folder.
   - for a page, that it is not open in any Capture editor.
2. Open both projects in their respective project managers.
3. Select the schematic folder, page, library, or symbol that you want to move or copy, then drag (pressing the CTRL key to copy) the selection to the destination project manager entity.
4. For both projects, from the File menu, choose Save All.

> - If you copy or move a document from one design or library to another, you should save the destination design or library immediately. If you do not, you may lose data if you open the moved document in the schematic page editor or part editor and then close the editor without saving the document.
> - Deleting schematic folders, schematic pages, parts and symbols is permanent. You cannot use the Undo command to bring back deleted items from the project manager.
> - If you move or copy a parent schematic folder or schematic page from one project into a second project, Capture remembers the name and directory of the file containing the child schematic folder or folders. This information is stored in the Attach Implementation dialog box for each hierarchical block and nonprimitive part.

Capture Directory Map

This section details the OrCAD Capture install directory (%cdsroot%\tools\capture) structure. The section also provides a brief description of the files types associated with OrCAD Capture.

Capture directory contents

CAPTURE.EXE	The Capture executable. It appears in the OrCAD Desktop program group as Capture.

CAPTURE.INI	Capture's initialization file. You can specify a new location for Capture to create and modify the .INI file. Use one of the following command lines to specify the new location of the .INI file:
	
	`CAPTURE -I directory\\`
	`CAPTURE /I directory\\`
	
	where path is the directory where the .INI is located. For example:
	
	`CAPTURE -I C:\CAPTURE`

*.EXE, *.PIF, *.DLL, and *.NT	Executable files, program information files, and other files required by Capture.
Library directory	Contains the Cadence-supplied library files (.OLB), including the CAPSYM.OLB symbol library.
Netforms directory	Contains the netlist format files used by Capture.
Samples directory	Contains sample designs.

Capture file types

*.BCF	Binary SDT configuration file, used in translation.
*.BOM	Bill of materials report file.
*.CFG	SDT configuration file, used in translation.
*.CIR	SPICE netlist file.
*.DBK	Design backup file.
*.DRC	Design rules check file.
*.DSN	Design file.
*.DSF	VST Model netlist file.
*.EDN	EDIF netlist file.

*.ERR	DSN2MNL error text file.
*.EXP	Export property file.
*.INC	Bill of materials include file.
*.INF	VST file.
*.INS	Netlist creation file.
*.LIB	Layout or SDT library file.
*.MAP	SPICE map file.
*.MNL	Layout netlist file.
*.NET	Netlist file for most netlist formats.
*.OBK	Library backup file.
*.OLB	Library file.
*.OPJ	An OrCAD project file. It contains references to all other files included in the project.
*.PIP	Netlist creation file.
*.PLD	OHDL netlist file.
*.RES	Netlist creation file.
*.RPT	Update properties report file.
*.SCH	SDT schematic folder file.
*.SWP	Gate and pin swap file.
*.TXT	Session log text file.
*.UPD	Update properties file.
*.V	Verilog netlist file.

*.VHD or *.VHO	VHDL file.
*.XNF	XNF netlist file.
*.XRF	Cross reference file.

Opening a project

Once you have created a project in Capture, all library and schematic information defined therein, as well as any other included files, is recognized as being part of that project. When you open the project, all such data is automatically associated with the project.

To open an existing project

1. From the File menu, choose the Open command. A standard Open dialog box appears.
2. If the project you want to open is not listed in the File name text box, do one or more of the following:
3. In the Look in drop-down list, select a new drive or directory.
4. In the File name text box, enter a portion of the file name—you can use the standard "*" and "?" wildcard characters.
5. Select the project or type the name in the File name text box, then click OK. The project opens in a project manager window.

Shortcut

Toolbar: 

To open a recently used project

- From the File menu, choose the project either by name or by number. The project opens in a project manager window.

Saving a project, design, or library

When the project manager window is active, you can save a new or existing project, design, or library. The Save command saves all open documents referenced by the project, as well as the project itself.

> ⚠ Except for occurrence properties, the schematics of externally-referenced libraries and designs should not be edited. You should view them as read-only designs. Trying to edit, then save, these designs from within your schematic can introduce errors such as duplicate reference designators and other problems.

When saving schematics with externally-referenced libraries or designs, occurrence properties are saved but altered instance values are not. If you want to change externally-referenced libraries or designs you should first close the referencing design. Then, open the referenced library or design, make the necessary changes, and save and close the referenced library or design. At this point, you can reopen the original design and reference the modified design.

When you save a design or project created using an earlier release, the database format for the design or project is updated for any changes in the new release. However, only opening the design or project does not result in changes.

The Save As command saves files depending on what you have selected in the project manager.

- If one or more designs or libraries are selected, you are prompted to save each file in turn.
- If no top-level folders (Design Resources or Outputs) are selected, and items other than designs or libraries are selected, the Save As command is unavailable.
- If no designs or libraries are selected in the project manager, you are prompted to save the project.

> When you use the Save As command, you are prompted to choose the file type from the Save As Type list in the Save As dialog box. You can choose to save the file in the current design database schema version or in a schema version that is one version prior to the application version you are currently using.

To save a new design or library

1. With the design or library selected in the project manager, from the File menu, choose Save. The Save As dialog box displays.
2. Enter a name for the design or library in the File name text box, specify a location, then choose the Save button.
   The design or library is saved, and the project manager remains open. When you close the project, Capture prompts you to save the project file.

To save an existing project

- With the Design Resources or Output folder selected, choose Save from the File menu.
  The project is saved, and remains open in the Capture session frame.
  When you save a project, you are saving all the files residing in the project. If you have several pages open in schematic page editor windows, changes you have made to any of them are saved. In addition, changes made by the Capture tools are saved to disk.

To save one project

- From the File menu of the project manager, choose Save. If the project is new and has not yet been saved, the Save As dialog box appears, giving you the opportunity to specify a drive and replace the system-generated name.

> When you save a project, Capture automatically creates a backup with a .DBK file extension. When you save a library, Capture automatically creates a backup with a .OBK file extension. If you save only a schematic page or a part, no backup is generated.

A Capture a design file (.DSN) is always accompanied by a project file (.OPJ). Each time you use the Save As command from the File menu to save a design file to another name or directory, you should also use Save As for the project file. The following process saves a .DSN file and a .OPJ file into the same directory so you can continue editing the current project without altering the original files.

To save a project file along with the design file to a different location

1. In the project manager, select Design Resources or Outputs.
2. From the File menu, choose Save As.
3. Specify the destination directory and edit the project name if needed.
4. In the Settings tab, specify if you want the design to be copied.
5. In the Settings tab, specify if you want to copy referenced files.
6. Click OK.

Shortcut

Toolbar: 

Changes you make to a schematic page are temporary until you save the page or the project to disk using one of the commands of the File menu. If you save one schematic page, all of the pages in the schematic are saved. If you save a project while you have several pages open in schematic page editor windows, changes you have made to any of them are saved as well as any changes made by the Capture tools.

To save one schematic page

- From the File menu of the schematic page editor, choose the Save command. If the design is new and has not yet been saved, the Save As dialog box appears, giving you the opportunity to specify a drive and replace the system-generated name.

To save one design

- From the File menu of the project manager, choose the Save command. If the design is new and has not yet been saved, the Save As dialog box appears, giving you the opportunity to specify a drive and replace the system-generated name.

When you save a project, Capture automatically creates a backup with a .DBK file extension. When you save a library, Capture automatically creates a backup with a .OBK file extension. If you save only a schematic page or a part, no backup is generated.

Closing a Project

When the project manager window is active, you can close a project without quitting Capture. Alternatively, you can close and save your project as you quit.

To close a project:

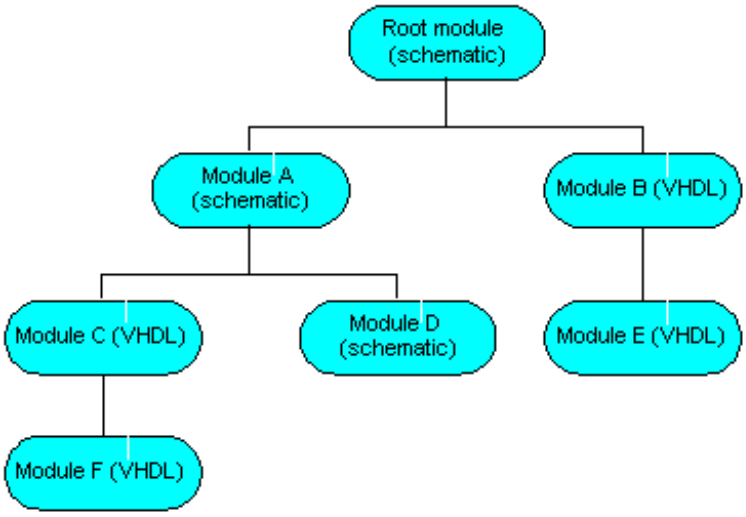
→ Choose *File – Close*.

A dialog box displays prompting you to save or reject the changes you made.

4

Working with Designs

Capture provides the means to create electronic designs in two media: as schematics or as VHDL models.
Schematic designs can include VHDL or Verilog models (one or the other, not both) as lower level hierarchical modules, but these models can only instantiate other models (of the same type) at lower levels in the hierarchy. Consider the following illustration:



Any schematic design module can include either schematics or VHDL/Verilog models as instantiated components. However, VHDL/Verilog design modules are limited to other modules of the same type as instantiated components. Hence, if the root module of your design is a VHDL model, all lower level modules must also be VHDL models.

- There can be only one design file in a project. If you create a new design file, or move or copy a different one into the project, the project manager will ask you if you want to replace the existing design file.
- If you haven't specified a root for your design, you cannot generate reports. Also, when folders are copied to a new design, the ROOT designation is lost and must be reestablished in the design.

In this section:

Creating a Design

When you create a project in Capture, a design with the same name as the project is immediately created for you. However, you also have the option of creating a design without first creating the project.

- Because a new design inherits characteristics from the settings in the Design Template dialog box, you should check those settings before you create a design.
- There can be only one design file in a project. If you create a new design file, or move or copy a different one into the project, the project manager will ask you if you want to replace the existing design fil

In this section:

During Project Creation

When the project is first created, the project manager creates a design file with the same name as the project. It also creates a schematic folder within the design file, and a schematic page within the folder. The schematic folder icon is marked with a backslash character (\) to signify that it is the root schematic folder.
A design file also contains a design cache, which is like an embedded library—it contains a copy of all the parts and symbols used on the schematic pages.
You can view the hierarchy using the File tab in the project manager.
You can create a new design file to replace the design created by the project manager.

A new design inherits characteristics from the settings in the Design Template dialog box. You should check those settings before you create a design.

Outside a Project

When you create a project, a design is immediately created with the project. However, you also have the option of creating a design without first creating a project.

To create a Design

- From the File menu, go to the New cascading menu cascading menu and choose Design.

The design with the name Design<n>.dsn is created in the Windows\temp directory. Capture checks if a file with the name Design1.dsn exists in the temp directory. If it exists, the design is designated the name Design2.dsn. This is continually incremented as you create designs using this procedure.

- When you create a design using the above procedure, you do not have the option to name the design.
- If you have a project currently open in Capture and you use the above procedure to create a design, a new design is created. The design in the current project is not affected.

Working with Title Blocks

Capture provides ANSI, and OrCAD title blocks in the CAPSYM.OLB library. In addition, you can create your own title block and store it in a library for future use.

There are two types of title blocks:

- A **default title block** is placed by Capture at the lower right corner of each schematic page.
- Any number of **custom title blocks** that can be placed at any location on the schematic page

The following properties are displayed for a title block in the property editor. Note that all properties might not be displayed depending on the type of block placed.:

Cage Code	Specifies the Cage Code. You can set the value for this property in the Title Block tab of the Design Template dialog box (Options - Design Template).
Design Create Date	Specifies the date of creation for the design. The value for this property is set automatically.
Design Create Time	Specifies the time of creation for the design. The value for this property is set automatically.
Design File Name	Specifies the path and file name of the design file. The value for this property is set automatically.
Design Modify Date	Specifies the date of the last modification to the design. The value for this property is set automatically.

Design Modify Time	Specifies the time of the last modification to the design. The value for this property is set automatically.
Design Name	Specifies the name of the design. The value for this property is set automatically.
Doc	Specifies the document number. You can set the value for this property in the Title Block tab of the Design Template dialog box (Options - Design Template).
Name	Specifies the name of the title block. By default, this is the name of the symbol in the library. You can manually update this field.
OrgAddr1	Specifies the first line of the organization address. You can set the value for this property in the Title Block tab of the Design Template dialog box (Options - Design Template).
OrgAddr2	Specifies the second line of the organization address. You can set the value for this property in the Title Block tab of the Design Template dialog box (Options - Design Template).
OrgAddr3	Specifies the third line of the organization address. You can set the value for this property in the Title Block tab of the Design Template dialog box (Options - Design Template).
OrgAddr4	Specifies the fourth line of the organization address. You can set the value for this property in the Title Block tab of the Design Template dialog box (Options - Design Template).
OrgName	Specifies the organization name. You can set the value for this property in the Title Block tab of the Design Template dialog box (Options - Design Template).
Page Count	Specifies the number of schematic pages in the design. This value is updated when you annotate a design.
Page Create Date	Specifies the date of creation for the schematic page. The value for this property is set automatically.
Page Create Time	Specifies the time of creation for the schematic page. The value for this property is set automatically.
Page Modify Date	Specifies the date of the last modification to the schematic page. The value for this property is set automatically.
Page Modify Time	Specifies the time of the last modification to the schematic page. The value for this property is set automatically.

Page Number	Specifies the number of the schematic page. The page number determines when it will be printed in relation to the other schematic pages of the design. This value is updated when you annotate a design.
Page Size	Specifies the page size of the schematic page, as was set at creation time.
RevCode	Specifies the revision. You can set the value for this property in the Title Block tab of the Design Template dialog box (Options - Design Template).
Schematic Create Date	Specifies the date of creation for the schematic. The value for this property is set automatically.
Schematic Create Time	Specifies the time of creation for the schematic. The value for this property is set automatically.
Schematic Modify Date	Specifies the date of the last modification to the schematic. The value for this property is set automatically.
Schematic Modify Time	Specifies the time of the last modification to the schematic. The value for this property is set automatically.
Schematic Page Count	Specifies the number of schematic pages in the given schematic. This value is updated when you annotate a design.
Schematic Page Number	Specifies the order of the schematic page within the schematic. This value is updated when you annotate a design.
Source Library	Specifies the path and file name of the library from where the title block was placed. The value for this property is set automatically.
Source Symbol	Specifies the name of the symbol for the title block in the Source Library. The value for this property is set automatically.
Title	Specifies the title. You can set the value for this property in the Title Block tab of the Design Template dialog box (Options - Design Template).

In this section:

Setting up the default title block

In Capture, each schematic page has a default title block in the lower right corner. Title block

information that you provide in the Design Template is written into the fields of the title block. You can choose the title block that Capture provides in the CAPSYM.OLB library or you can create a custom title block.

When you make a new default title block, you create its graphic symbol, add one or more properties to define the information fields, then you provide the information that will appear in the fields.
Selections that you make following these instructions are reflected in schematic pages you create subsequently, but do not affect existing schematic pages.

To specify information for title block fields

1. From the Options menu, choose the Design Template command, then choose the Title Block tab.
2. Enter the information for the nine fields that appear. This information appears in the title block and in reports generated by the Capture tools.

To select a default title block symbol

1. From the Options menu, choose the Design Template command, and then choose the Title Block tab.
2. Specify a path and library. You can use the Browse button to locate a title block.
3. Specify the name of the title block, maintaining the case of the original name.
4. Click OK.

- If Capture does not find a title block with the name you specify (the case of letters must match) in the library and path you specify, no default title block is placed.
- If you are using CAPSYM.OLB and you have maintained the directory structure that OrCAD provides, you can leave the Library Name text box empty.

To create fields that are automatically filled on a default title block

1. With the title block graphic symbol displayed in the part editor, double-click an area where there are no objects to display the User Properties dialog box.
2. Choose the New button. The New Property dialog box appears.
3. Enter one of the properties listed above in both the Name and Value text boxes, then click OK to dismiss the New Property dialog box.
4. In the Properties dialog box, choose the Display button, then select the Visible option.
5. Click OK to dismiss the Display Properties dialog box, then click OK again to dismiss the User Properties dialog box. The part editor appears with the property representation visible.
6. Use the mouse to move the property representation to the appropriate location and click

the left mouse button to place the representation. This property representation, which serves as a place holder, appears in the selection color.

7. Repeat steps 2 through 6 for each property of your title block.
8. Save the title block.

Creating a custom title block

You can place any number of **custom title blocks** at any locations you choose. The text that appears is a result of visible properties that you define when you create the symbol or after you place the title block.

When you make a custom title block, you create its graphic symbol, then define and place visible properties.

Property	Definition
Doc	Specifies the document number
RevCode	Specifies the revision
Cage Code	Specifies the Cage Code
Title	Specifies the title
OrgName	Specifies the organization name
OrgAddr1	Specifies the first line of the organization's address
OrgAddr2	Specifies the second line of the organization's address
OrgAddr3	Specifies the third line of the organization's address
OrgAddr4	Specifies the fourth line of the organization's address
Page Count	Specifies the number of schematic pages in the design
Page Number	Specifies the number of the schematic page. The page number determines when it will be printed in relation to the other schematic pages of the design

To create a custom title block symbol

1. In the project manager, open the library in which you will store the title block symbol.
2. From the Design menu, choose the New Symbol command. The New Symbol dialog box opens.
3. Enter a name and select Title Block as the Symbol Type, then click OK. The part editor opens with an empty part boundary box.
4. Use the graphics tools to create the title block. See Creating graphics. The part boundary box will stretch to accommodate your graphic objects.

To provide information for the fields of a custom default title block

1. From the Options menu, choose the Design Template command, then choose the Title Block tab.
2. For each of the nine properties listed above that you have added to your title block symbol, enter the text that will appear in your title block.
3. Enter the library name and the distinctive name of the title block.
4. Click OK.

Placing Multiple Title Blocks

Capture places a default title block at the lower right corner of every schematic page. In addition, you may place any number of custom title blocks at any location on the schematic page. Custom title blocks, unlike default title blocks, do not include information from the design template, but you can define and place visible properties on the title block.

Information placed as properties in custom title blocks will not appear in reports or netlists created by Capture; only the information in the default title block will be used.

To place custom title blocks

1. From the Place menu, choose the Title Block command. The Place Title Block dialog box appears.
2. In the Symbol text box, enter the name—you can use the standard "*" and "?" wildcard characters. Capture scans the selected libraries and lists all symbols that match the name or wildcard.
3. If the title block name is not listed, see Searching for a part in the libraries for further information.
4. Click on a title block name in the list for a preview, or double-click on it to place the title block. You can also select the title block name and click OK to place the title block.

Opening a Design

When you create a design in Capture, all schematic information defined therein, is recognized as being part of that project. When you open the design, all such data is automatically associated with it.

To open an existing design

1. From the File menu, choose the Open command. The standard Open dialog box displays.
2. If the design is not in the current folder, use the Look in drop-down list to navigate to the design folder.
3. In the Files of type drop-down list, select Capture Design (*.dsn).
4. Select the design from the files and folder list or enter the name of the design in the File name text box and click OK.

You can use the standard * (asterisk) or ? wildcard characters when typing in the File name text box.

The design opens in a Project manager window.

Shortcut

Toolbar: 

To open a recently used project

From the File menu, choose the project either by name or by number. The project opens in a project manager window.

Creating a new VHDL or Verilog file

You can create VHDL or Verilog files as part of your design's functionality description, or as test benches for simulation with NC VHDL or some other simulator. You can also instantiate lower level VHDL or Verilog files within a VHDL or Verilog file.

> Any model instantiations within a VHDL or Verilog file must be of the same type (that is, VHDL or Verilog) as the parent file.

To create a new VHDL or Verilog file

There are two ways to create a new VHDL or Verilog file in Capture:

1. From the File menu, choose New, then choose VHDL File or Verilog File, as appropriate.
2. A file of the appropriate type opens in Capture editor.

OR

1. With the project manager active, choose New VHDL File or new Verilog from the Design menu. The file opens in the appropriate editor and a dialog box appears, asking if you want to add the file to the project.
2. Choose the Yes button to add the file to the project that is currently open. The Save As dialog box appears.
   Note: If you choose the No button, Capture does not add the file to your project and you must save it yourself at a later point in time.
3. Select a directory for the file and supply a filename. By default, a VHDL file name is VHDLn.VHD (where n is an integer indicating the number of .VHD files created in the current session). Similarly, a Verilog file is named Verilogn.V.
4. Choose the Save button. Capture saves the file and places it in the Design Resources folder of your project.

Creating a Text File

To create a text file from the Project manager

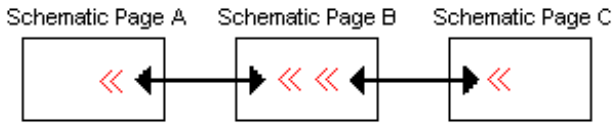
1. From the File menu, point to New and choose Text.
2. Type in the text that you want.
3. Close the text session when you are finished. Capture asks if you want to save the text file. If you want to save it, enter a name and location for storage.

Flat vs. hierarchical designs

In this section:

Flat designs

For schematic designs, flat designs are structures in which the output signals of one schematic page connect directly to the input signals of another schematic page in the same schematic folder though off-page connectors.



A flat design has no hierarchy (no hierarchical blocks, hierarchical ports, hierarchical pins, or parts with attached schematic folders). All schematic pages in a flat design are contained within a single schematic folder. Regardless of the number of schematic pages in a flat design, all parts appear at the same level of hierarchy in the Hierarchy tab.
Since you must manage all of the interconnections between the pages of a flat design using the names assigned to the off-page connectors, it is best to keep a flat design relatively small.
For VHDL models, flat designs are implemented in a single entity/architecture pair. The entire functionality of the design unit is described within the VHDL architecture. For example:

```
architecture behavior of Dtype is
begin
  process (ck)
  begin
  if (ck = '1') and ck'event then
  q <= d;
  end if;
  end process;
end behavior;
```

Hierarchical designs

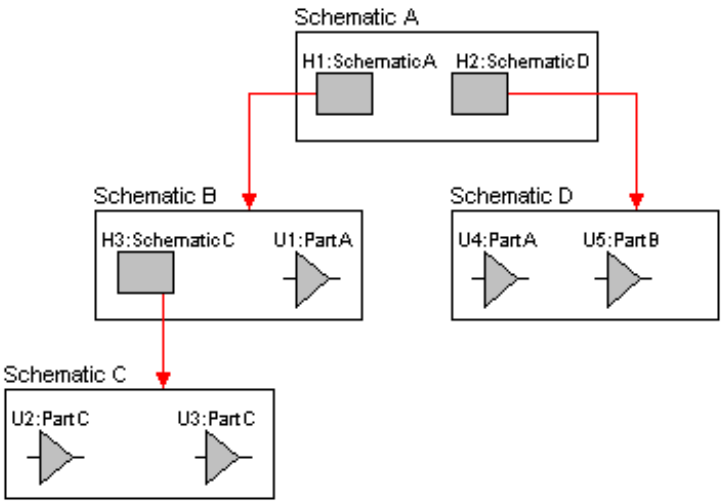
You can manage both schematic and VHDL design resources in a hierarchical manner. That is, you can create schematic pages containing hierarchical blocks or parts with schematic or VHDL implementations. The hierarchical block symbol (or part with an attached schematic page or model) in the schematic page editor is the primary mechanism you use to extend the scope of the design. Use hierarchical blocks to partition the major functional regions of your design using a block diagram.

Any schematic page can contain combinations of hierarchical blocks or parts that refer to other schematics or VHDL source files. This nesting structure can be many levels deep. VHDL source files may only instantiate VHDL models; you cannot refer to a schematic folder from within a VHDL source file.

The schematic folder or VHDL entity at the top of a hierarchy, which directly or indirectly refers to all other modules in the design, is called the root module. In the project manager's file tab, the root module has a backslash on its folder icon. The root module folder, as well as any other module folder, can contain as many schematic pages or VHDL models as you require. Capture also supports a combination of flat and hierarchical structures such that a schematic folder containing multiple schematic pages may be associated with a hierarchical block or part.

Simple hierarchies

A one-to-one correspondence between hierarchical blocks or parts and the schematic, EDIF or VHDL implementations they reference is called a simple hierarchy. The picture below is an example of a simple hierarchy typical of most PCB designs in Capture.

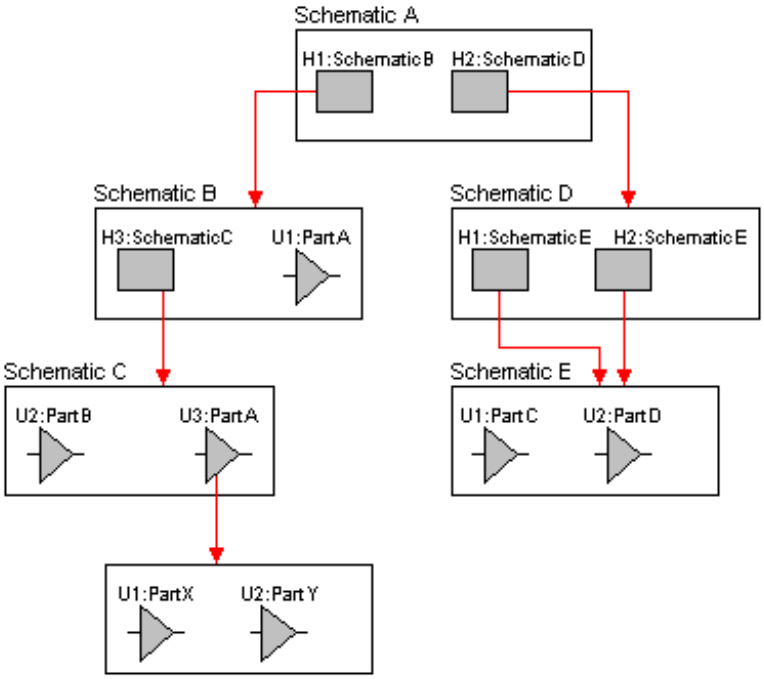


In a simple hierarchy, each hierarchical block or part with an attached schematic folder or VHDL model represents a unique design module. The project manager's hierarchy tab displays a simple hierarchical design as a tree of schematic pages.

Complex hierarchies

A complex hierarchy is one that includes a many-to-one correspondence between the hierarchical blocks or parts and their implementations (schematic, EDIF, or VHDL). The picture at right is an example of a complex hierarchy typical of most programmable logic designs in Capture.

As shown in the picture, two hierarchical blocks (H1 and H2 on schematic D) reference the same schematic (schematic E).



In this section:

- Editing Hierarchical block look and feel

Editing Hierarchical block look and feel

When you create a hierarchical block on a schematic page, the block displays the part name and the implementation. However, you may want to add a picture to the block that acts as a visual representation of the implementation below the block.

To place a graphical object on a hierarchical block

1. Click the hierarchical block.
2. Right-click on the block and choose Edit Part from the pop-up menu.
   The block opens in the Capture Part Editor.
   From the Place menu, select any object to place on the block.
   Note: To ensure that the object is visible on the part on the schematic, place the object on the block.
3. Save the changes and close the Part Editor to return to the schematic.

To change the look and feel of a hierarchical block

1. Click the hierarchical block.
2. Right-click on the block and choose Edit Part from the pop-up menu.
3. The block opens in the Capture Part Editor.

4. From the Place menu choose rectangle.
5. Click the crosshair cursor at one corner of the block and drag the cursor to the diagonally opposite corner to cover the entire block.
6. Click to select on the rectangle you created over the block in Step 4.
7. Right-click on the rectangle (any of the edges of the block) and choose Edit Properties from the pop-up menu.
8. From the Edit Properties menu, choose the properties form the drop-down lists to apply to the block.
9. Save the changes and close the Part Editor to return to the schematic.

> Even after closing the Part Editor, you can undo all the changes you made to the block by using the Capture Undo command (Edit - Undo or Ctrl + Z).

Renaming a Design

When you create a project, a design is created within the project. This design has the same name as the name of the project. However, you have the option to rename this design. Alternatively, you might have an existing design that you want to replicate and use as another design. Here too you can rename the design.

To rename a design

1. In the Project manager, select the design (.DSN) file.
2. From the File menu, choose Save As.
   The Save As dialog box displays.
   In this dialog, you specify an alternative name for the design.
   You can also specify an alternative directory location for the design.
3. Enter a name for the design and click OK.

> While using the Save As dialog box to rename the design, do not specify a dot (.) in the design file name.

Saving and Closing a Design

Any changes you make to a design are temporary until you save the design.

When you save a design, you save only the changed schematic pages and folders in the design. The changed design as well as any edited pages are marked by the asterisk (*) character.

To save a design

- From the File menu choose Save.

To save a design to a different location

1. From the File menu choose Save As.
2. In the Save As dialog, choose the name and location for the design.

To close a design

- From the File menu choose Close.

> If any part of the design is currently not saved, you are prompted to save un-saved changes or ignore the changes.

5

Working with Schematic Folders

When you create a design in Capture, you cannot place schematic pages directly in the design. To place and create schematic pages, you need to create schematic folders also referred to as schematics.

In this section:

Creating a Schematic Folder

When you create a project in Capture, a design is created within the project and a schematic folder, named SCHEMATIC1, is created within the design. You can also create more schematic folders in the current design.

To create a schematic folder:

1. In project manager, select the design.
2. From the *Design* menu, choose *New Schematic*.
3. In the New Schematic dialog, enter a name for the schematic folder.
4. Click *OK*.

> In Capture, the title of a window depends on the name of the currently open document.

Deleting a Schematic Folder

You can delete a schematic folder from a design even if the folder contains pages. However, this operation cannot be undone.

To delete a schematic folder:

1. From the *Design* menu, choose *Delete*.

You are prompted with a warning that the delete operation cannot be undone.

2. Click *OK* to delete the schematic folder.

- The delete command on a folder cannot be undone.
- You cannot delete a folder if any of the schematic pages within the folder are currently open. So, you need to close all the open schematic pages in the folder before you delete the folder.
- You cannot delete the root schematic folder. To delete this folder, you will first need to specify another folder in the design as the root folder.

Renaming a Schematic Folder

In Capture, the titles of the windows in which you work are based on the names of the open documents. When you create a new part, a symbol, a schematic folder, a schematic page, a project, or a library, you can specify a name or accept the unique name assigned by Capture.

To rename a schematic folder:

1. In project manager, select the schematic folder to rename.
2. Choose *Design – Rename*.
3. In the dialog box that appears, enter the new name and click *OK*.

The name is changed immediately.

Moving Schematic Folders

You can use schematic folders to organize a design by grouping schematic pages in ways that serve your purpose.

If you are working in one project and you want to use one or more schematic folders in another project, you can transfer schematic folders from one project to another, or you can create a copy for use in multiple projects. You cannot, however, move or copy a schematic folder into the design cache of any other design.

To move a schematic folder from one project to another:

1. Verify that no document is open in the schematic folder.
2. In project manager, select the schematic folder(s) to move.

3. Choose *Edit – Cut*. If you want to have a copy of the schematic folder in both the projects, choose *Copy* instead of *Cut*.

4. Open the project in which you want to paste the schematic folder(s).

5. Select the `filename.DSN` folder, and choose *Edit – Paste.*

6. Choose *File – Save All*. Do this for both the projects.

or

1. Verify that no document is open in the schematic folder.

2. Open both the projects in their respective project manager windows.

3. Drag and resize the two project manager windows so that both are visible.

4. Select the schematic folder(s) to move, and drag the schematic folder(s) to the `filename.DSN` folder in the second project manager window. To keep a copy of the schematic folder in both the projects, press and hold CTRL while you drag the folder.

5. Choose *File – Save All*. Do this for both the projects.

Attaching a Schematic Folder

You attach a schematic folder to extend net connections between schematic folders. The attached schematic folder is the child schematic folder in a hierarchy. A schematic folder can be attached to a non-primitive library part, a non-primitive part instance on a schematic page, or a hierarchical block.

To attach a schematic folder to a new hierarchical block:

1. Open the schematic page editor on the parent page.

2. Choose *Place – Hierarchical Block*.

3. Specify a name in the *Reference* field.

4. In the *Implementation Type* drop-down list, select *Schematic View*.

5. In the *Implementation name* text box, enter the name of the child schematic folder.

6. If the child schematic folder is not in the current design, specify the path and library where the schematic folder is located.

7. Click O*K*.

To attach a schematic folder to a new part:

1. Create a new part. For details, see Creating Parts .

2. Click the *Attach Implementation* button. The Attach Implementation dialog box appears.

3. In the *Schematic* text box, select *Schematic View* from the *Type* drop-down list box.

4. Specify the name of the child schematic folder in the *Name* text box.

5. If the child schematic folder is not in the current design, specify the path and library where the schematic folder is located.

6. Click *OK* twice.

To attach a schematic folder to an existing hierarchical block or part:

1. Select the hierarchical block or part on the parent schematic page, and choose *Edit – Properties*. The property editor appears.

2. Click the *Implementation Type property cell*, and choose *Schematic View* from the drop-down list.

3. Click the *Primitive* property cell, and select *No* from the drop-down list. Using this setting you can ascend and descend the hierarchy.

4. Enter a name in the *Implementation* property cell.

5. If the child schematic folder is not in the current design, specify the path and library where the schematic folder is located using the *Browse* button in the *Implementation Path* property cell.

6. Click *Yes* on the Undo Warning message box.

7. Click *Apply* and close the property editor.

Recommendations on Attaching a Schematic Folder

- It is recommended that, rather than editing parts in libraries provided by OrCAD, you copy the part and make changes in a custom library. If you do edit a library provided by OrCAD, it is important that you assign a new library name (choose *File – Save As*) so that your changes are not overwritten when you update or upgrade your software.

- Ensure that you do not create recursion in your design. Capture cannot prevent a recursion, and the *Design Rules Check* command does not report it.

- If you attach external schematic folders or other files to hierarchical blocks in a design or parts in a library, be sure to include the attachments when you pass the design or library to a board fabrication house or to another engineer. Attached schematic folder and other files are not carried along automatically when you copy or move a part, a schematic folder, or a schematic page to another library, design, or schematic folder. Only the “pointers” to the attached schematic folders and files— their names and the names of the designs or libraries that contain them— are carried along.

- Attached files work much like their counterparts in mail as they do not provide an alternative definition of the part (as do the attached schematic folders). If you attach a schematic folder to a homogeneous part, it is attached to each part in the package and not to the package itself. You cannot attach a schematic folder to a heterogeneous part.

- When you attach a schematic to a part or a hierarchical block, you can specify a full path and filename in the *Library* text box. So, although you can specify a library that has not been saved, you should not try to descend into the attached schematic folder until the library that contains the schematic folder has been saved.

- If you do not specify a full path and filename in the Library text box, Capture expects to find the attached schematic folder in the same design as the part of the hierarchical block to which it is attached. If the specific schematic folder does not exist in either the design or the library, Capture creates the schematic folder when you descend hierarchy on the part or hierarchical block.

- Capture preserves the case of the path and filename as you specify them in the Library text

box for compatibility with future versions of Windows.

6

Working with Schematic Pages

In this section:

Creating a Schematic Page

A schematic folder can contain one or more schematic pages. There can be multiple schematic folders in a project and each folder can contain multiple schematic pages.

When you create a project in Capture, a design is created immediately; also the root schematic folders are created within the design and one schematic page is created within the folder. You can also create multiple schematic pages within a schematic folder.

To create a schematic page:

1. On the *File* tab of project manager, select the schematic folder to which you want to add the new schematic page.

2. Choose *Design – New Schematic Page*.

   OR

   Right-click on the folder to which you want to add the new schematic page and choose *New Page* from the pop-up menu.

3. Specify a name for the new schematic page, and click *OK.*

Defining Schematic Page Characteristics

Using the design template, you can establish the characteristics of a schematic page for an entire project. You can also override these defaults and establish characteristics of a specific schematic page using the *Schematic Page Properties* or *Design Properties* command.

Capture creates a schematic page size to suit your printer or plotter. You can choose from the five standard page sizes or specify a custom size. The default title block symbol, default title block information, border, and grid references can all be established for each project. Title block visibility can be specified for the entire project or for each schematic page.

To define grid references for new designs and schematic pages:

1. Choose *Options – Design Template*, and then choose the *Grid Reference* tab.

2. Make selections for the horizontal and vertical grid references and then click *OK* to close the Design Template dialog box.

Until you change the Grid Reference tab, any designs or schematic pages you create reflect these selections.

To change grid references for an existing page:

1. Open the schematic page editor for the schematic page.

2. Choose *Options – Schematic Page Properties*, and then choose the *Grid Reference* tab.

3. Make selections for the horizontal and vertical grid references and then click *OK* to close the Schematic Page Properties dialog box. The selections are reflected in the active schematic page.

To define schematic page size for new designs and schematic pages:

1. Choose *Options – Design Template*, and then choose the *Page Size* tab.

2. Select *Inches* or *Millimeters* as the unit of measure.

3. Select a page size.

   -- If you choose inches, the choices are *A*, *B*, *C*, *D*, *E*, and *Custom*.

   -- If you choose millimeters, the choices are *A4*, *A3*, *A2*, *A1*, *A0*, and *Custom*.

4. Specify a value for pin-to-pin spacing, and click *OK* to close the Design Template dialog box.

Until you change the Page Size tab, any designs or schematic pages you create will reflect

these selections.

Working with Label States

During the design process, you may want to experiment with different “what if?” scenarios to determine the best implementation for your purposes. Capture provides a method to do this with *label states*. Label states allow you to define different states for a schematic page, perform edits, and then return to the defined state of the schematic before the editing occurred. Each schematic page can have its own set of label states. Further, each schematic page has two label states- *Start* and *End*-that are implicit. Label states are identified with a unique label consisting of one to five alphanumeric characters.

- Since *Start* and *End* are default label states for each schematic page, any label state that you define cannot use the identifiers Start or End as labels.
- Labels are not case-sensitive. *STATEA* and *stateA* are considered identical by Capture.

To set a label state:

1. From the *Edit* menu, choose *Label State*, and then choose *Set.* The Set Label State dialog box appears.

2. Specify a name for the label.

   Note: Labels must be one to five characters long, and must be unique per schematic page. The labels *Start* and *End* are reserved and cannot be assigned.

3. Click *OK.*

   The state of the schematic is labeled.

To go to a label state:

1. From the *Edit* menu, choose *Label State*, and then choose *Goto.* The Goto Label State dialog box appears.

2. Specify the label of the state to which you want to return.

3. Click *OK.*

Capture returns you to the labeled state of the schematic, removing any edits that occurred in the interim.

To delete a label state:

1. From the *Edit* menu, choose *Label State*, and then choose *Delete*. The Delete Label State dialog box appears.
2. Specify the label for the state that you want to delete.
3. Click *OK*.

   The label state is removed and you can no longer return to it with the Goto Label State dialog box.

> ⚠ You cannot delete the default *Start* and *End* label states.

Working with VHDL and Verilog files

Capture provides editing capabilities for developing VHDL and Verilog files. You can use this capability to create functional models for design behavior, or testbenches for simulation.

In this section:

Creating a VHDL Model from a Hierarchical Block

In Capture, you can create VHDL models from hierarchical components on your schematic page. That is, you can create a hierarchical block on a schematic page, then create a VHDL model that defines its functionality. Creating VHDL models from hierarchical blocks in this manner is generally termed "top-down" design methodology.

To create a VHDL model from a hierarchical block:

1. Open the parent schematic page in the schematic page editor.
2. From the *Place* menu, choose the Hierarchical Block command. Capture displays the Place Hierarchical Block dialog box.

3. Assign a reference designator to the hierarchical block in the *Reference* text box

4. Choose *VHDL* as the implementation type from the Implementation Type drop-down list box.

5. Specify the entity name for the VHDL model in the Implementation name text box.

6. Specify a file name for the VHDL model that will define the behavior of the hierarchical block in the Path and filename text box.

   The file name must be a VHDL file (*.VHD).

7. Click *OK*.

   Capture returns to the schematic page editor.

8. Use the cursor to draw an outline for the block. Click at one corner of the desired area, drag the cursor to the opposite corner, and release it.

   Capture places the outline of the block on the schematic page.

9. With the new hierarchical block selected, choose the *Place Hierarchical Pin* button from the tool palette.

   Capture displays the Place Hierarchical Pin dialog box.

10. Assign a name to the pin, specify pin type and width, and click OK.

    Note: Do not move the cursor outside of the schematic page editor window before completing the next step.

11. Move the cursor to the desired location in the hierarchical block and click to place the pin on the block.

12. Right-click and choose *End Mode* from the pop-up menu.

13. Repeat steps 9 through 12 to place additional hierarchical pins as needed.

14. With the hierarchical block selected, right-click and choose *Descend Hierarchy* from the pop-up menu.

    Capture generates a VHDL model template for the block using the hierarchical pins as port names.

15. Specify the functional description of the block in the model architecture.

16. Close the VHDL editor.

17. When prompted to save the changes to the file, click *OK*.

    At this point, the hierarchical block is defined and ready to be *wired in* to the rest of the schematic.

Creating a Verilog Model from a Hierarchical Block

In Capture, you can create Verilog models from hierarchical components on your schematic page. That is, you can create a hierarchical block on a schematic page, and then create a Verilog model that defines its functionality. Creating models from hierarchical blocks in this manner is generally termed "top-down" design methodology.

To create a Verilog model from a hierarchical block:

1. Open the parent schematic page in the schematic page editor.

2. From the *Place* menu, choose the Hierarchical Block command.

   Capture displays the Place Hierarchical Block dialog box.

3. Assign a reference designator to the hierarchical block in the Reference text box

4. Choose *Verilog* as the implementation type in the *Implementation Type* drop-down list box.

5. Specify the module name for the Verilog module in the Implementation name text box.

6. Specify a file name for the Verilog model that will define the behavior of the hierarchical block in the Path and filename text box.

   The file name must be a Verilog file (*.V).

7. Click *OK*.

   Capture returns to the schematic page editor.

8. Use the cursor to draw an outline for the block. Click at one corner of the desired area, drag the cursor to the opposite corner, and release it.

   Capture places the outline of the block on the schematic page.

9. Choose the *Place Hierarchical Pin* button from the tool palette with the new hierarchical block selected.

   Capture displays the Place Hierarchical Pin dialog box.

10. Assign a name to the pin, specify pin type and width, and then click *OK*.

    Note: Do not move the cursor outside of the schematic page editor window before completing the next step.

11. Move the cursor to the desired location in the hierarchical block and click to place the pin on the block.

12. Right-click and choose *End Mode* from the pop-up menu.

13. Repeat steps 9 through 12 to place additional hierarchical pins as needed.

14. With the hierarchical block selected, right-click and choose *Descend Hierarchy* from the pop-up menu.

    Capture generates a Verilog model template for the block, using the hierarchical pins as port names.

15. Specify the functional description for the block in the model architecture.

16. Close the Verilog editor.

17. When prompted to save the changes to the file, click *OK*.

The hierarchical block is defined and ready to be *wired in* to the rest of the schematic.

Checking the syntax of VHDL or Verilog files

You can check the syntax of VHDL or Verilog files using the appropriate *Check Syntax* command. The tool checks the syntax in all the VHDL or Verilog files selected in the project manager window.

> You must have a project open to use the *Check Syntax* command. Error reporting for the tool requires some project resources that are not available unless a project is open.

To check the syntax of VHDL files:

1. In the project manager window, select the VHDL file for which you want to check the syntax. You can select multiple files using the Ctrl key.

2. From the *Edit* menu, choose the Check VHDL syntax command.

   The *Check Syntax* tool finds the first error in the file and highlights it so that you can fix it.

3. To continue checking the file, choose the *Check VHDL syntax* command from the *Edit* menu again.

Shortcuts

Pop-up menu: *Check Syntax*

To check the syntax of Verilog files:

1. In the project manager window, click to select the Verilog file for which you want to check the syntax. You can select multiple files using the Ctrl key.

2. From the *Edit* menu, choose the Check Verilog syntax command.

   The *Check Syntax* tool finds all the errors in the file and highlights them so that you can fix them.

   Further, Capture opens a text file, VAN.TXT, that contains details of all the errors in the netlist.

Shortcuts

Pop-up menu: *Check Syntax*

Moving Schematic Pages

You use schematic folders to organize a design, grouping schematic pages in ways that make the most sense for your requirements. If you change your mind, you can easily transfer schematic pages from one schematic folder to another. You can also place copies of pages in several schematic folders.

If one of your projects contains one or more schematic pages that solve a problem in a second project, you can transfer the pages from one project to another, or you can place copies in both the projects. Because all schematic pages must be contained in a schematic folder, a schematic folder to hold the pages must exist in the second project before you can place the pages.

If a document in a schematic folder is open in an editor, the document cannot be moved or copied.

To move schematic pages from one schematic folder to another:

1. Verify that the pages are not open in the schematic page editor or the spreadsheet editor.
2. In project manager, select the schematic pages you want to move.
3. Choose *Edit – Cut*.

   If you want to have a copy of the pages in both schematic folders, choose the *Copy* command.
4. Select the schematic folder to store the pages.
5. Choose *Edit – Paste*.

   OR

1. Verify that the pages are not open in the schematic page editor or the spreadsheet editor.
2. Select the schematic pages that you want to move, then drag them to the appropriate schematic folder.

   If you wish to have a copy of the pages in both the schematic folders, press and hold the Ctrl key while you drag.

To move schematic pages from one project to another:

1. Verify that the schematic pages are not open in the schematic page editor.

2. Open a project and select the schematic pages you want to move.
3. Choose *E dit – C ut*.

   If you want to have a copy of the pages in both projects, choose *Copy*.
4. Open the project in which you want to use the schematic pages.
5. Select the schematic folder to store the pages.
6. Choose *E dit – P aste*.
7. Choose *F ile – S ave All*. Do this for both the projects.

OR

1. Verify that the schematic pages are not open in the schematic page editor.
2. Open both projects in their respective project managers.
3. Drag and resize the two project manager windows so that each is visible.
4. Select the schematic pages that you want to move, then drag them to the appropriate schematic folder in the second project manager window.
5. If you want to have copies of the pages in both projects, press and hold Ctrl while you drag the pages.
6. Choose *F ile – S ave All*. Do this for both the projects.

- If you copy or move a document from one design or library to another, you must save the destination design or library immediately. Else, you may lose data when you open the moved document in the schematic page editor or part editor and then close the editor without saving the document.
- If you move or copy a parent schematic folder or schematic page from one project into a second project, Capture remembers the name and directory of the file containing the child schematic folder(s). This information is stored in the Attach Implementation dialog box for each hierarchical block and non-primitive part.

Renaming a Schematic Page

In Capture, the windows in which you work have headings based on the name of the open document. You can simplify your search through a set of open windows by using unique names for different documents. When you create a new part, symbol, schematic folder, schematic page, project, or library, you can specify a name or accept the unique name assigned by Capture.

To rename a schematic page:

1. In project manager, select the document you want to rename.
2. From the *Design* menu, choose *R ename*.

   OR

1. Right-click on the page in project manager and choose *Rename* from the pop-up menu.
2. In the Rename Page dialog box, specify a new name and click *OK.*

   The name change takes effect immediately.

Deleting a Schematic Page

To delete a schematic page:

1. In project manager, select the page(s) to delete.
2. From the *Design* menu, choose the Delete Command.

   OR

Press the Delete key.

- You cannot undo the *Delete Page* command.
- You cannot delete schematic pages that are currently open. You must close all the open schematic pages to be deleted.

Closing and Saving a Schematic Page

To close a schematic page:

- From the *File* menu choose *Close*.

If the page contains unsaved changes, you will be prompted to save or discard the changes.

To save changes to a schematic page:

- From the *File* menu, choose *Save*.

This saves only the current page and not the other pages in the design, even if you have edited them. However, if you save a page in a design that has been upreved from a previous release, you need to save the entire design. For upreved designs, click *Yes* to create a backup and save all the pages with changes.

If you open a 16.2 design with off-page connectors in 16.6, you must first save the design.

7

Working with Part Libraries

A library is a file that stores parts, symbols, title blocks, schematic folders or schematic pages. Capture provides more than 80 libraries; in addition, you can create custom libraries. If you edit a library provided by Capture, you should give it a custom name so that you do not copy over your changes when you receive updated libraries. You can, for example, create a library to hold all your programmable logic devices, or hold schematic folders that you use often. There is no need to create a library for a particular project, because the design cache holds all the parts and symbols used in the project.

Since a library is a file, you can work with it in the Windows File Manager as well as in Capture. It is recommended that, rather than editing parts in libraries provided by OrCAD, you copy the part and make the changes in a custom library. If you do edit a library provided by OrCAD, it is important that you assign a new library name (from the File menu, choose Save As) so that your changes are not overwritten when you update or upgrade your software.

> If you move a library after you place a part, the connection between the part and its library is broken. In this case, the Update Cache command will not find the library; you will need to use the Replace Cache command and specify the new path to the library.

In this section:

Adding a Library to your Project

When you add a library to a project, all the parts contained within the library become available for placement on the schematic pages.

To add an existing library to your project:

1. From the *File* menu, choose *Open - Library*.
2. Select the library you want to open.

   If the library file is not listed:

   - In the *Look in* drop-list box, select a new drive, a new directory, or both.
   - In the *Files of type* box, select the type of file to open.
3. Choose the *Open* button. The project manager window opens and the library parts appear

in project manager.

> If you open an SDT library, Capture prompts you to save the library.

To add a recently used library to your project:

- From the *File* menu, choose the name of the library you want to add.

> If you move a library after you place a part, the connection between the part and the library is broken. In this case, the *Update Cache* command will not find the library and you need to use the *Replace Cache* command and specify the new path to the library.

Creating a Library

In Capture, you can add as many libraries as required by specifying a name and storage location for each library. Each library is available to each project. The library size is limited only by the amount of space on your system's hard disk; however larger libraries take longer to load. If speed becomes an issue, consider creating several smaller libraries.

> Schematic folders and schematic pages cannot be created in a library, but can be copied or moved to a library from a project. Schematic folders and schematic pages can also be edited in a library.

When you create a new library, project manager adds an empty library to the project. To populate the library, you can create your own parts, or you can move or copy parts from another library. See Moving parts or symbols between libraries for details.

To create a new library from project manager:

1. Right-click on the new library icon and save the new library with a file name and location of your choice.

2. Populate the library by moving or copying files from other libraries. For more information, see Moving parts or symbols between libraries.

To create a new library:

1. Choose *File - New* and select *Library*.

   Capture adds a Library folder and a Library Cache folder to the project.

   OR

1. From the *File* menu, go to the New menu.

2. Select *Library* and click *OK*.

   If you have an instance of project manager currently open at the time, Capture adds the library to the project. Otherwise, Capture creates a new project for the library.

To rename a library:

1. In project manager, select the library.
2. Choose *File* - *Save As*.
3. Specify a new name in the *File Name* text box.
4. Click *OK* to return to project manager.

To specify the storage location for a library:

1. In project manager, select the library.
2. Choose *File* - Save As command.
3. Select the drive and directory in which you want to store the library.
4. Click *OK* to return to project manager.

Opening a Library

When you open a library, you can edit it in part editor.

To open a library:

1. Choose *File* - *Open*.

   The Open dialog box appears.

2. Select the library to open.

   If the library file is not listed:

   - In the *Look in* drop list box, select a new drive, a new directory, or both.
   - In the *Files of type* box, select the type of file to open.

3. Select the *Open* button.

   The Project Manager window opens; the library parts appear in Project Manager.

> If you open an SDT library, Capture prompts you to save the library.

To open a recently used library:

- From the *File* menu, choose the name of the library you want to open.

Shortcut

Tool palette: 

Editing a Library

In this section:

Copying a Part from the Design Cache to a Library

You can copy parts from the design cache to a library. This is useful if you have modified a part through the schematic page editor, and want a permanent copy of the part.

To save a design cache part in a library:

1. In project manager, open the design cache, and select the part to save.
2. Choose *Edit - Copy.*
3. Open the library where you want to store the part.
4. Choose *Edit - Paste*.

   Alternatively, you can open two instances of project manager, one for the design and one for the library, and drag-and-drop the part from the design cache to the library.

- If you edit an OrCAD-provided library, it is important to assign a new name to the library so that the changes you make are not overwritten when you upgrade or update your software.

Copying a Schematic Page to a Library

If you have a small circuit that is used in many projects, you can put the circuit on a separate schematic page, save it in a library, and attach it to a part that you can place in any design. It is recommended that you keep the attached schematic folder and the part in the same library.

To save a schematic page in a library, you must first move the page into a schematic folder. When you save a schematic page in a library, Capture puts the circuit's parts in the library cache. A schematic folder or schematic page that is open in an editor cannot be moved or copied.

Before editing a schematic page stored in a library, you should find out which projects call the schematic folder. Editing a library-stored schematic folder may create problems for other projects that use the library-stored schematic folder.

To save a schematic page in a library:

1. Verify that the page is not open in the schematic page editor.
2. In project manager, create a schematic folder to store the page.

3. Select the schematic page you want to save, and copy or move the page into the new schematic folder.

   Keep the new schematic folder selected.

4. From the *Edit* menu, choose *Copy.*

5. Open project manager for the library in which you want to store the schematic folder.

6. From the *Edit* menu, choose *Paste.*

OR

1. Verify that the page is not open in the schematic page editor.

2. Open project manager for the project and create a schematic folder to store the page.

3. Open project manager for the library in which you want to store the schematic folder.

4. Drag and resize the project manager windows for the project and the library so that each is visible.

5. In the project manager window for the project, select the schematic page you want to save, and copy or move the page into the new schematic folder. Keep the new schematic folder selected.

6. Press and hold Ctrl while you drag the schematic folder to the project manager window for the library.

> If you copy or move a document from one design or library to another, you must save the destination design or library immediately. Else, you may lose data if you open the moved document in the schematic page editor or part editor and close the editor without saving the document.

Copying a Schematic Folder to or from a Library

The behavior of schematic folders and schematic pages is almost identical in libraries and designs.

The primary differences are:

- Schematic folders and schematic pages cannot be created in libraries. If you want to add a schematic folder or a schematic page to a library, you must create it in a design and then move it to the library.

  In addition to this topic, see Copying a Schematic Page to a Library for more information.

- Schematic folders and schematic pages are limited to the library tool set, namely updating properties, exporting properties, and importing properties.

- The *Update Cache* and *Replace Cache* commands are not available when parts are

selected in the library cache.

- The *Annotate* command is unavailable for parts in schematic folders contained in libraries. You should use the Annotate command in a schematic folder prior to moving it to a library.

- You can open a library-stored schematic page in a schematic page editor, and edit it exactly the same way as if you had opened it from a design. However, it is recommended that you edit a schematic page in a design.

- If a schematic folder stored in a library is the child of another schematic folder in a design, the Descend Hierarchy command in the parent schematic folder opens the library containing the child schematic folder and a schematic page editor window containing the schematic page.

If you have a circuit that you use in many projects, you can put that circuit in a separate schematic folder, save it in a library, and attach it to a part that you can then place in any design. It is a good design practice to keep the part and the attached schematic folder in the same library. When you save a schematic folder in a library, Capture puts the circuit's parts in the library cache.

A schematic folder or schematic page that is opened in an editor cannot be moved or copied.

To save a copy of a schematic folder as a library object:

1. Verify that no part is opened in the schematic folder.
2. In project manager, select the schematic folder to save.
3. Choose *Edit* - *Copy*.
4. Open the project manager window for the library in which you want to store the schematic folder.
5. Select the library (.OLB) in the project manager window.
6. Choose *Edit* - *Paste*.

   Alternatively, you can open two project manager windows for the library and the design and drag-and-drop the schematic folder from the design to the library.

   When using the drag-and-drop procedure, you need to keep the *Ctrl* key pressed.
7. Choose *File* - Save command.

> If you copy or move a document from one design or library to another, you must save the destination design or library immediately. Else, you may lose data if you open the moved document in the schematic page editor or part editor and then close the editor without saving the document.

To copy a schematic folder from a library to a design:

1. In project manager, open the library containing the schematic folder that you want to copy and select the schematic folder.
2. Choose *Edit - Copy*.
3. Open the project in which you want to store the schematic folder.
4. Select the design (.DSN) in the project manager window.
5. Choose *Edit - Paste*.

   Alternatively, you can open two project manager windows for the design and the library and drag-and-drop the schematic folder from the library to the design.

   When using the drag-and-drop procedure, you need to keep the Ctrl key pressed.
6. Choose *File - Save*.

You can also attach a schematic folder to a part or hierarchical block without copying the schematic folder into the project or library, but this method affects design portability. For more information, see Attaching a schematic folder.

Moving Parts or Symbols between Libraries

You can store parts or symbols in any library. You can create libraries that serve a specific purpose. You can transfer some parts or symbols from one library to another and store some parts or symbols in multiple libraries. If you move a part with part alias from one library to another, the part alias is also moved to the library.

> A part that is open in an editor cannot be moved or copied.

To move or copy parts between libraries:

1. Verify that the parts are not open in the part editor or the spreadsheet editor.
2. In project manager, select the parts to move.
3. To move the part, choose *Edit - Cut*. To copy the part, choose *Edit - Copy*.
4. In project manager window, select the library (.OLB).
5. Choose *Edit - Paste*.

   Alternatively, you can open two instances of project manager for the two libraries and drag-and-drop the parts between the libraries. To copy the parts, keep the Ctrl key pressed while you drag-and-drop the parts.

> If you edit a library provided by OrCAD, it is important that you assign a new library name so that the changes are not overwritten when you upgrade or update your software.

To copy or create a part using an existing part:

1. Follow the procedure for moving parts between libraries to create a copy of the part in another library.

2. If you want to move the new part (created from an existing part) to the original library:

    a. Use the Rename command and specify a different name for the part.

    b. Follow the above procedure to drag the part to the original library.

Renaming a Library

To rename a library:

1. In project manager, select the libary (.OLB) file.

2. Choose *File - Save As*.

    The *Save As* dialog box displays. In this dialog, you specify an alternative name for the library. You can also specify an alternative directory location for the library.

3. Specify a name for the library and click *OK*.

> While using the Save As dialog box to rename the library, do not specify a dot (.) in the library file name. Renaming a library in this manner breaks the links between the library and the parts selected from it, and the *Update Cache* command does not work. Such libraries also are not listed in the *CAPTURE.INI* file for configuration.

Closing and Saving a Library

The changes you make to a part are temporary until you save the part or the library using one of the commands on the *File* menu. When you save a library, you save all the parts and symbols residing in the library. If there are several parts or symbols opened in the part editor, the changes you make to any of them are saved. If the library is new and has not yet been saved, the Save As dialog box appears, giving you the opportunity to specify a drive and replace the system-generated name.

To save a library:

- From the *File* menu in project manager, choose the *Save* command.

    The active library or the library that holds the active part is saved.

***To save a library* with a different file name:**

1. Open the project manager window for the design or the library.

2. Choose *F ile – S ave As*.

3. Specify a new file name in the *File Name* text box, and click *OK*.

For compatibility with future versions of Windows, Capture preserves the case of the path and filename as you specify them in the *Library* text box.

When you save a project, Capture automatically creates a backup with a *.DBK* file extension. When you save a library, Capture automatically creates a backup with a *.OBK* file extension. If you save only a schematic page or a part, no backup is generated.

To close a library:

- From the *File* menu in project manager, choose the *Close* command.

Using the Library Correction Utility

The library correction utility automatically verifies/corrects missing pin numbers and duplicate pin names in the Capture symbol library by scanning through all the parts in the library, finding out, and correcting the components that have missing pin numbers and duplicate logical pin names. This utility also converts components with lowercase pin names to uppercase and makes the Power pins visible for all the components in the library.

Library Correction Utility

Use the library correction utility to fix missing pin numbers and duplicate pin names issues in a design.

Starting the Library Correction Utility

To run the Library correction utility:

1. Choose *Accessories* - *LibCorrectionUtil* - Library Verification/Correction.
2. In the Library Correction Utility dialog box, browse to the library for correction in the *Select Library for Correction* field.

Verifying/Correcting Library Components With

Missing Pin Numbers

- Check the *Missing Pin Numbers* option to correct the components with missing pin numbers.

  The library correction utility scans through all the parts in the library and finds out the components with missing pin numbers. All the missing pin numbers are updated by their corresponding pin names. At the end of updating process, the following message pops-up:

  "Corrected <number of parts corrected> of Parts in <lib_path>/<lib_name>.olb for missing pin numbers" or "No Part in this library has missing pin numbers".

  The log file is generated at the location where the selected library resides. The naming convention used for log files is <Library_name>Miss.log. The log file lists all the parts and also the pin numbers for each corrected part in the library. If you choose the Verify option, only the log file is generated but the library is not updated.

Missing pins Log File Example:

List of verified / corrected packages

```
CONNECTOR DB25

              No missing Pin numbers

CAPACITOR NON-POL

              CAPACITOR NON-POL.Normal       1

              CAPACITOR NON-POL.Normal       2


      1 parts in library C:\TEMP\MISSINGPIN_NUMBERS.OLB have missing pin numbers.
```

The last line of the log file indicates that one part in the MISSINGPIN_NUMBERS.OLB library has missing pin numbers. The pin names for the missing pin numbers are 1, 2.

Duplicate Pin Names

- Check the *Duplicate Pin Names* option to correct the components with missing pin numbers.The library correction utility scans through all the parts in the library and finds out the components that have duplicate logical pin names.

  -- If a component has duplicate power pin names, those pins are not considered as duplicate pin names.

  -- To remove duplicate pin names, the library correction utility changes the duplicate pin names by appending "#" followed by the pin number to the duplicate pin names.

  -- The combination of the pin name and the pin number makes the pin name unique.

The first pin that this utility encounters does not get appended with # followed by its pin number.

At the end of updating process, the following message pops-up:

"Corrected <number of parts corrected> of Parts in <lib_path>/<lib_name>.olb for duplicate pin names" or "No Part in this library has duplicate pin names".

The log file is generated at the location where the selected library resides. The naming convention used for log files is <Library_name>Dup.log. The log file lists all the parts and also the corrected pin names for each of the corrected part in the library. If you choose the *Verify* option in the next step, only the log file is generated but the library is not updated.

Duplicate Logical pins Log File Example:

List of verified / corrected packages

```
ASSYMETRICAL

      ASSYMETRICALAA.Normal 2 A\#2

      ASSYMETRICALCC.Normal 8 A\\#8

74LS00

      No duplicate Pin names

1 parts in library C:\TEMP\DUPLICATE\_PIN\_NAMES.OLB have duplicate
pin names
```

The last line of the log file shown above indicates that one part in the DUPLICATE_PIN_NAMES.OLB library has duplicate pin names. The corrected pin names are A#2 and A\#8.

Verifying vs. Correcting

- Select *Verify.*

The library is verified for the missing pin numbers and/or duplicate pin names and a log file is generated.

Or

- Select *Correct*
  The library is corrected for missing pin numbers and/or duplicate pin names and a log file is generated.

When you opt for correction, the library is corrected. Therefore, if you want to have a copy of the old library, you need to back-up the library. In fact, it would be always safe to make a backup of the old library before correcting the library.

Changing Casing of Pin Names and Numbers

- Check the *Change the Pin name and number to uppercase* option.

The utility scans through all the parts in the library and converts all the components that have pin names appearing in lowercase to uppercase.

At the end of updating process, the following message pops-up:

"Changed all the Pin name to upper case. Please close the library and reopen to see the change."

Making Power Pins Visible

When you check this option, the utility will scan through all the parts in the library and change the Power pins settings for all the non-zero length pins in the library to *Visible*. Additionally, you can also make all zero length Power pins in the library visible. To do this, check the Change Zero Length Pins to check box and choose an appropriate pin shape option (Line/Short) from the list box.

> The Change Zero Length Pins to check box is available only, if the *Make All Power Pins Visible* check box is checked.

At the end of updating process this utility will pop up message: " Made all the Power Pins visible. Please close the library and reopen to see the change.“

Advantages of this Utility in the Flow

In the Capture-PCB Editor flow, new Capture-PCB Editor interface throws error, "Error ALG0031", if any of the components on your design have missing pin numbers. Old third party interface used to netlist the null pin number as pin name and take the component through flow. however, in that case the component was not updated at all.

In Capture-PCB Editor Flow, new Capture-PCB Editor interface throws error "Error ALG0050", if any of the components on your design have duplicate pin names.

Use Model for Capture-PCB Editor Flow

For New Designs:

Run this utility on library files, for correcting missing pin numbers and duplicate pin names, and use these libraries to create new designs. After correcting the libraries, you will not encounter errors, ALG0031 and ALG0050.

Correcting Existing Designs:

Run this utility on library files for correcting missing pin numbers and duplicate pin names, and then use the *Update Cache* command from the *Design* menu.

What if you do not have libraries for Existing Design?

Though this is not the preferred way, you can copy all the design cache components to a new library and then run this utility on that library. Now use the *Replace Cache* command from the *Design* menu. In this process, design cache points to a single library after replacing the cache. This affects the canonical path in the .pst files, as the source library name is embedded in the canonical path.

If you do not have all the libraries for the existing design, perform the followings steps:

1. Copy all design cache components to a new library.
2. Run the library correction utility on the new library, created in step 1.
3. Use Replace Cache command from Design menu.

Assumption:

This utility will put a pin name in place of an empty pin number. Matching footprints is your

responsibility as a user because in some cases footprint may not match with pin numbers as shown in the following example:

Pin number	Pin name
1	A B
3	C
4	D

The Library Correction utility will copy B in place of the missing pin number between 1 and 3. In this case, footprint should contain pin numbers as 1,B,3,4 instead of 1,2,3,4.

Known Limitation:

If all the pin types other than Power have duplicate pin names and missing pin numbers, the missing pin numbers functionality copies the pin names in place of missing pin numbers. This is followed by the duplicate pin names functionality which appends pin names to pin numbers followed by #. Even then pin names are not unique. This results in error "Error ALG0050". For these parts, you need to edit each part and create unique pin names for each pin type other than the Power pin type.

8

Working with Parts

Heterogeneous and Homogeneous Parts

Parts usually correspond to physical objects—gates, chips, connectors, and so on—that come in packages. Think of these packages as physical parts and the parts you place on a schematic page as logical parts. Physical parts that comprise more than one logical part are sometimes referred to as multiple-part packages. To keep it simple, both are referred to as parts in Capture.

Logical parts in a package may have different pin assignments, graphics, and user properties. If all the logical parts in a package are identical except for the pin names and numbers, the package is *homogeneous*. If the logical parts in a package have different graphics, numbers of pins, or properties, the package is *heterogeneous.* For example, a hex inverter is homogeneous--the six inverters are identical, except for their pin numbers. A relay--that has a normally opened switch, a normally closed switch, and a coil--is heterogeneous: the three physical parts differ in graphics, number of pins, and properties.

Creating a Heterogeneous Part

To create a heterogeneous part in a library:

1. Open the library and choose *New Part* from the *Design* menu.
2. Specify the part name.
3. Set the number of parts in the package.
4. Select *Heterogeneous* in the *Package Type* group box.
5. Enter the PCB Footprint if you want to assign one to the part at this time.
6. Click the *OK* button.

Capture will give you the logical part with the reference designator U?A.

1. Draw the part body and add the pins.
2. To get to the B package, choose *Next Part* from the *View* menu, or press CTRL+N.

Capture displays the U?B part for you to edit.

1. Again, draw the part body and add the pins.
2. Repeat this process until all the logical parts are created.

After creating the logical parts of your heterogeneous part in the library you need to assign a unique property to each one. For example, create a property named PACKAGE.

> Do not use GROUP as the name for this property. This may cause problems while annotating your design for a PCB Editor tool, like Allegro PCB Editor. The GROUP property is used in PCB Editor for a specific purpose.

To create this property:

1. Double-click on the empty space beside the logical part to get the User Properties dialog box.
2. Choose the New button, type in PACKAGE into the Property box and a 1 in the Value box.
3. Click twice to attach that property to the logical part.
4. Add this new property to each logical part in your package (part A, B, C, and so forth).
5. Save your library with the new part in it.

Open your schematic and place the A, B, C (and so on) logical parts of your heterogeneous parts appropriately in your design. After you place each logical part, double-click on the part to get the property editor. Edit the value of the PACKAGE property shown in the spreadsheet. Leave the Value of 1 on that property for each logical part of the first set you place; assign a Value of 2 to each logical part in the second set; assign a Value of 3 to each logical part in the third set, and so on. Capture uses these values to group the heterogeneous parts correctly when assigning reference designators.

When you get ready to annotate the design, you add that property name to the combined property string in the Annotate dialog box. Capture will use this property and the assigned values to annotate the parts correctly in the design. To do this, go to the project manager window, select the design name, and choose Annotate from the Tools menu. Select Update entire design; select Unconditional reference update (select Incremental if you have already partially annotated your design); and type in {PACKAGE} into the Combined property string box. This gives you a combined property string like {Value}{Source Package}{PACKAGE}. When you click OK, Capture then assigns the appropriate reference designators to all your parts in the design including the heterogeneous parts.

> Do not change the reference designators of heterogeneous parts for a complex hierarchical design manually. In case you want to change the reference designator for a part placed in the schematic page, delete it, and add it again. This way all the occurrences will get updated correctly.

Split parts

A split part is a multi-sectioned package. You may need to section a part for different reasons:

- Your design may include parts that have thousands of pins. Such large-sized parts may not fit in a single schematic page. To handle such parts, you can split them into multiple

sections based on your specification and can place different sections in different schematic pages. This will ease designing.
- You want to partition a large part based on its functionality and use sections individually. For example, you may like to create different sections for pins with the same voltage rating.

In this section:

Type of Packages

A package consists of one or more parts. Depending on the type of parts in a package, a package is classified as a Heterogeneous or Homogeneous part.

Placing Parts on a Schematic Page

OrCAD Capture ships with more than 400 part libraries. To place a part on your schematic, you will first need to search for the part, and then place the part on the schematic page.

In this section:

Searching for a Part

To search for a part you can go to the Place Part window. In this window, you can either select

one or more libraries to search. This method is useful if you know the library (or group of libraries) in which the part exists. Alternatively, you can search for a part from the libraries contained in a Windows directory.

To search for a Part in the Selected Libraries:

1. From the *Place* menu, choose *Part*.
   In the Place Part window, add the libraries in which you want to search for the part.
2. To add a library, click *Add Library*.
3. In the Browse File dialog, select `<the library to add>`.
   To add multiple libraries:
   a. Use the Ctrl + click or the Shift + click combinations.
   b. Choose one library at a time. Use this method if the libraries to add are in different directories.
4. In the Libraries list, choose the library (or libraries) within which you would search for the part.
   To select multiple libraries, use the Ctrl +click or the Shift + click combinations.
5. To use the *Filter* command, click the *Filter* button.
6. In the *Part* text box, enter `<the name of the Part>`.
   You can use the * (asterisk) or ? (question mark) wildcard characters to search.
   Notice that as you type, Capture auto-completes the Part name.
7. Press `Enter`.

To search for a Part in all the Libraries in a Directory:

1. From the *Place* menu, choose *Part*.
   In the *Place Part* window, add the libraries in which you want to search for the part.
2. Click the Plus sign to the left of *Search for Part* at the bottom of the *Place Part* window.
3. In the *Search For* text box, enter the name of the part to search.
   You can use the * (asterisk) or ? (question mark) wildcard characters to search.
4. Click the *Search* button.
   Capture scans the libraries in the selected directory, and lists all the parts that match the name or wildcard.
5. In the *Libraries* list, select the library containing the part you want, and click *Add*.
   If you need to know a part's library of origin, you can select the part in the project manager, then select *Replace Cache* from the *Edit* menu. The part name and the library and path are listed in the dialog box that appears.
6. Click *Cancel* to return to the project manager.

You can identify the library of origin for multiple parts by Creating a cross reference report.

Placing a Part

Parts are stored in libraries. In addition, you can create your own parts in custom libraries. Some library parts have a convert as well as the normal graphical representation. Many packages contain more than one part, in which case you may need to specify which of the parts to place.

To place a part:

1. From the *Place* menu, choose the Part command.
   The *Place Part* window is displayed. For more information, see Searching for a part in the libraries.
2. From the *Parts* list, select the part you wish to place.

   > When you select a part, the PSpice symbol ( ) is displayed for a part that can be simulated using PSpice and the layout symbol ( ) is displayed for a part that is supported for PCB Editor flow. The symbols are displayed below the Packaging box.

   Remember the following, when you are are placing.

   - To place a convert version of the part, select *Convert* in the *Graphic* group box.
   - If the package contains more than one part, select one of the parts from the drop-down list in the *Packaging* group box.
     The part appears in the preview box.
   - In case of a heterogeneneous part, you need to decide which part in the package you want at this time, because you will not be able to change the part in the package after the part is placed.

   > Whether the part is homogeneous or heterogeneous, if you are placing multiple copies of the same part in the package, you do not need to specify the part number at the time of placing the part. Capture will auto-increment the part numbers as you keep placing the parts on the schematic.

3. Click the *Place part* button.
   OR
   Press Enter.
   An image of the part is attached to the pointer.
   Press F6 to change the cursor to a crosshairs to place the part at a specific location.
4. Move the part image and click to place the part.
5. Press the ESC key or select another tool to dismiss the part that is attached to the pointer.

The first time a part is placed, a copy of the part is created in the design cache.

When you place a part off-grid, it remains off-grid throughout any cut-and-paste and drag-and-drop operations. If you place parts so that two pins meet end to end, the pins are connected.

OrCAD recommends that you connect the pins of the parts using a wire, and avoid placing parts in a way that two pins meet end to end. This is because, parts with direct pin-to-pin connections produce a system-generated net name to establish the connection and:

- Capture will not allow you to assign your own net name in the place of the system-generated net name.
- Searching for the system-generated net name can be difficult if you are not aware of the pin to pin connection.
- If you move the parts after creating the netlist, the system-generated net name might change. This may cause net name conflicts when you run back-annotation.

It is recommended that you do not connect a power symbol directly to a power pin. Connect the power symbol to the power pin using a wire.

You can place a part in the middle of a wire segment without redrawing the wire by placing the part over the wire such that two pins on the part connect with the wire segment. Then click over the part with the TAB key pressed until just the overlapping wire segment is selected. Finally, delete the wire segment.

Do not change the reference designators of heterogeneous parts for a complex hierarchical design manually. In case you want to change the reference designator for a part placed in the schematic page, delete the part and add it again. This way all the occurrences will get updated correctly.

When you place a part, make sure that the Automatically reference placed parts check box is selected in the Miscellaneous tab of the Preferences dialog box. This will ensure that the part references for the newly placed part are unique.

Shortcut

Tool palette: 

T o uniquely identify parts

1. In the project manager, select schematic folders or schematic pages, if you want to process only a portion of the design. If you want to process the entire design, leave the schematic folders or schematic pages unselected.
2. From the Tools menu, choose Annotate.
   The Annotate dialog box appears.
   Verify that the dialog box options are set the way you want them. For example, you specify

whether to update the entire design or only the schematic folders or pages selected in the project manager, whether to assign part references to all parts or to only those that have not been previously updated, or whether to return all the part references to the unassigned state (such as C? or U?A). Note that if you choose the Reset reference numbers to begin at 1 in each schematic option, it is possible that part references will be duplicated within a schematic folder that contains multiple pages.

3. Click OK.

If you copy a part into the Clipboard and then paste it onto a schematic page, Capture will automatically assign a unique reference designator to the pasted part when the following two conditions are met:

1. The Auto Reference option on the Miscellaneous tab of the Preferences dialog box is selected.
2. The pasted part has a reference designator assigned to it when it is copied to the Clipboard.

Capture assigns the reference designator, updated to the next available value (one greater than the highest value used on the schematic at that point). If the pasted part has a default reference (for example, R?) Capture does not assign a unique reference designator to it.

Shortcut

Toolbar: 

Creating hierarchical blocks

In this section:

- Attaching a schematic folder to a hierarchical block
- Creating a hierarchical block from a Verilog model
- Creating a hierarchical block from a VHDL model

Attaching a schematic folder to a hierarchical block

A hierarchical block is a representation of a schematic folder, which is attached to the hierarchical block. It provides vertical (downward-pointing) connection only. The hierarchical pins in a hierarchical block act as points of attachment for electrical connections between the hierarchical block and other connectivity objects in the attached schematic folder. A hierarchical block functions just like a part with an attached schematic folder.
Before you create or re-size a hierarchical block, make sure the Snap to grid option is turned on (from the schematic page editor's Options menu, choose Preferences). If the hierarchical block is on Fine grid, then hierarchical pins inside it are also on Fine grid—even if you change the Snap to grid setting before you place them—and it may be difficult to connect to these off-grid hierarchical pins.
A part with an attached schematic folder functions exactly as described for hierarchical blocks,

and pins on such a part function exactly as described for hierarchical pins within a hierarchical block. You can use the same attached schematic folder for either method of defining a hierarchy. The only difference between the two methods is that a part with an attached schematic folder is easier to reuse. See Creating parts and assigning properties for related information.
If you choose the Descend Hierarchy command on a non-primitive part or hierarchical block, and Capture cannot find the attached schematic folder, Capture creates a schematic folder in the active design.

> When you descend into an object that does not yet have a schematic folder or page associated with it, Capture creates the new schematic page and folder and gives it the same name as the hierarchical block. Because schematic names, schematic page names, part names, and symbol names are all limited to 31 characters, it is best to limit hierarchical block names to 31 characters.

When you place a hierarchical block, you must specify a reference. However, the reference need not be updated if the hierarchical block is primitive. For example, you could specify the reference to be "Halfadd?" when you place a hierarchical block. Then, when you run Annotate, the hierarchical block's reference is updated along with other parts.
If you attach an existing schematic folder to a hierarchical block, Capture automatically creates the hierarchical pins that correspond with the schematic folder's hierarchical ports. If you descend hierarchy on a hierarchical block whose schematic folder doesn't yet exist, then Capture automatically creates the hierarchical ports that correspond with the hierarchical pins of the hierarchical block.

> If you attach external schematic folders or other files to hierarchical blocks in a design or parts in a library, be sure to include the attachments when you pass the design or library to a board fabrication house or to another engineer. Attached schematic folders and other files are not carried along automatically when you copy or move a part, schematic folder, or schematic page to another library, design, or schematic folder. Only the "pointers" to the attached schematic folders and files —that is, their names and the names of the designs or libraries that contain them— are carried along.

Attached files work much like their counterparts in email—they do not provide an alternative definition of the part (as do attached schematic folders).

> When you attach a schematic folder to a part or hierarchical block, you can specify a full path and filename in the Library text box. So, although you can specify a library that hasn't been saved, you should not try to descend into the attached schematic folder until the library that contains the schematic folder has been saved.

If you don't specify a full path and file name in the Library text box, Capture expects to find the attached schematic folder in the same design as the part of hierarchical block to which it is

attached. If the specified schematic folder doesn't exist in either the design or library, Capture creates the schematic folder when you descend hierarchy on the part or hierarchical block. For compatibility with future versions of Windows, Capture preserves the case of the path and filename as you specify them in the Library text box.

> The Select Entire Net command is restricted to the active schematic page—it doesn't follow hierarchical blocks, hierarchical ports, or off-page connectors across schematic folders or schematic pages. For more information, see Tracing a net.

Remember that nets on a schematic page are electrically connected by name, by alias, or by connection to a named hierarchical port or off-page connector.

To attach a schematic folder to a hierarchical block

1. From the *Place* menu, choose *Hierarchical Block*.
2. Enter a name for the hierarchical block in the *Name* text box.
3. Choose *Schematic View* as the implementation type, in the *Implementation Type* list box.
4. Type the <name of the schematic folder in the *Implementation Name* text box.
5. If the schematic folder is not part of the current project, specify the path to the schematic folder in the *Library or File Pathname* text box.
6. Click *OK*.
7. Use the cursor to draw the boundaries of the hierarchical block on the schematic page. Capture creates the new hierarchical block and automatically places the hierarchical pins according to the ports that exist in the attached schematic.

> - Be careful not to create recursion in your design. Capture cannot prevent recursion, and the Design Rules Check command does not report it.
> - Recursion causes Capture to process infinitely as it tries to expand the design, resulting in the loss of any changes you've made to your design since it was last saved.

Creating a hierarchical block from a Verilog model

In Capture, you can create hierarchical blocks from Verilog models for inclusion on your schematic page. Creating hierarchical blocks in this manner is generally termed "bottom-up" design.

To create a hierarchical block from a Verilog model

1. Open the parent schematic page in the schematic page editor.
2. From the *Place* menu, choose Hierarchical Block (ALT, P, H).
3. Enter a name for the hierarchical block in the *Reference* text box.

4. Select *Verilog* as the implementation type in the *Implementation Type* drop-down list box.
5. Type the module name for the model in the *Implementation* name text box.
6. Specify the Verilog file for which you want to create a hierarchical block in the *Path and filename* text box. Make sure the file is a Verilog type file (*.V).
7. Click *OK*.
8. Use the cursor to draw the boundaries of the hierarchical block on the schematic page. Capture creates the new hierarchical block and automatically places the hierarchical pins according to the port list specified in the module section of the Verilog file.

> If the port names in the Verilog model have both upper and lower case characters in their identifiers, the property Vlog_Uppercase is attached to the resulting hierarchical block. For more information about Vlog_Uppercase, see the discussion in Verilog tab.

At this point, the hierarchical block is defined and ready to be "wired in" to the rest of the schematic.

Creating a hierarchical block from a VHDL model

In Capture, you can create hierarchical blocks from VHDL models for inclusion on your schematic page. Creating hierarchical blocks in this manner is generally termed "bottom-up" design.

To create a hierarchical block from a VHDL model:

1. Open the parent schematic page in the schematic page editor.
2. From the *Place* menu, choose Hierarchical Block.
3. Enter a name for the hierarchical block in the *Reference* text box.
4. Select *VHDL* as the implementation type in the *Implementation Type* drop-down list box.
5. Type the entity name for the model in the *Implementation name* text box.
6. Specify the VHDL file for which you want to create a hierarchical block in the *Path and filename* text box. Make sure the file is a VHDL type file (*.VHD).
7. Click *OK*.
8. Use the cursor to draw the boundaries of the hierarchical block on the schematic page. Capture creates the new hierarchical block and automatically places the hierarchical pins according to the port list specified in the VHDL entity.

At this point, the hierarchical block is defined and ready to be "wired in" to the rest of the schematic.

Creating Parts

In Capture, you can create a part and add it to a new or existing library. The part may be a single part or a multiple-part package. It can contain graphics, which must be inside the part-body border, as well as IEEE symbols and text, which can be either inside or outside the part-body border. For

any part that you create, you can also create a part convert.

Creating a part involves three processes:

1. defining the part in the *Part Properties* dialog box (including defining the part package as either heterogeneous or homogeneous)
2. defining the part body
3. placing pins on the part body

In this section:

> You can use an existing part as a model for a new part by moving a copy of the part to a second library and then editing the copy. If you wish to have the new part in the original library, rename the new part, then move it to the original library.

Creating a Part Body

When you create a new part, the part editor opens with an empty, rectangular visible part outline (the part-body border) visible. The part body border expands to accommodate the graphic elements of the part body, and pins are constrained to the part-body border.

If you want to change the size or shape of the part-body border, you can select the border and drag the selection handles until the part-body border appears as you want it.

Pins, when you place them, are constrained to the part-body border. If the edge of the part-body coincides with this border, the pins are directly attached to the part-body, but if the part-body is inside this border, you must draw a line from the pin to the part-body. You may place individual pins, or you may place an array of pins.

You define the part-body using the tools available on the tool palette. All of these tools are also available on the Place menu. Using the Selection tool, you can select a placed object for editing.

> You can draw part bodies thicker than pins and the rest of the part by adjusting the line style for the graphic objects you want to be thicker.

When you place a pin, you can describe it completely. To place pins, you need to be in the Part view of the part editor.
If you want to place several identical pins that are not sequentially numbered on the part-body border, the Pin tool is ideal. See *To place an array of pins* section, if you wish to create multiple identical pins that are numbered sequentially on the part body border.

When you place a pin in one view (normal or convert), Capture places an identical pin in the other view to prevent the parts from getting out of sync. The same is true about deleting pins. Changing the name of a pin in one view doesn't cause the name to change in the other. However, if you change a pin number in one view, Capture changes the pin number in the other view so the two views stay in sync.

If the part you are creating includes a series of pins that vary only in pin number, placing a pin array is very convenient. A pin array is defined by a single set of electrical characteristics. This tool is ideal if you wish to create multiple pins with identical properties and place them so that the pin numbers and names are sequentially arranged on the part body border, this tool is ideal.

Both homogeneous and heterogeneous parts may have shared pins. A common use of shared pins is for supply (power or ground) pins, which are referred to in Capture as "power pins."
On heterogeneous parts, power pins can be visible on every part in the package. If the pins are visible, they must be placed on at least one part in the package, and that part must be placed in the design for the power connections to appear in the netlist. Invisible power pin types must also be in a part that is placed in the design for them to appear in a netlist.

On homogeneous parts, power pins appear on every part in the package. The pin names are filled in automatically, but you must specify the pin numbers. For the pins to be shared, verify that both the pin names and pin numbers are the same for every part in the package.

- If you place the same pin on multiple parts in a package, you can inadvertently short two nets. Use caution to avoid this problem, and always run Design Rules Check before creating a netlist.
- Pin names are shared, but pin numbers are not.

You may have comment text, in the font of your choice, on a schematic page or a part. Use the text tool to document your schematic folder or to place the logic definition for a programmable logic device.

At certain zoom scales, Capture substitutes text that is too small to appear with filled rectangles.. These placeholders are only for display—the text prints correctly.

Before you begin drawing, you may want to specify default line and fill styles because all lines and shapes you draw adopt the current line style, and closed shapes adopt the current fill style. You can use a variety of line types or fill styles for any schematic page or part.

To change the snap-to-grid option

- From the *Options* menu, choose the *Preferences* command, then choose the Grid Reference tab.
  You set the option separately for the schematic page editor and the part editor.

To set a default line style

1. From the *Options* menu, choose the *Preferences* command and then choose the Miscellaneous tab.
2. Click on the *Line Style and Width* drop-down list to display the options.
   Note that you can specify separate options for the schematic page editor and the part editor.
3. Select one of the options and click *OK*.
   Any lines or shapes you draw will have this line style.

To define a default fill:

1. From the *Options* menu, choose the *Preferences* command and then choose the Miscellaneous tab.
2. Click on the *Fill Style* drop-down list to display the options.
   Note that you can specify separate options for the schematic page editor and the part editor.
3. Select one of the options and click *OK*.
   Any closed shapes you draw will have this fill style.

To create the part body you use the drawing objects in the Place *menu* (or the Draw toolbar). These include line, polyline, rectangle, arcs.

Notice that when you are in the Part editor, most of the options on the *Place* menu are disabled.

To draw an object:

1. From the *Place* menu, choose the appropriate drawing command or select the appropriate drawing command from the *Draw* toolbar.
2. Use the mouse to draw the object.
   To constrain the object by the orthogonality rules, press and hold the `SHIFT` key while you draw.

To edit line style or fill style of a placed object

1. Select the object.
2. From the *Edit* menu, choose the Properties command.
3. Select another line style or fill style in the dialog box that appears, then click *OK*.

To add comment text to a part or a schematic page:

1. From the *Place* menu, choose *Text*.
2. Enter the text in the dialog box that appears.
3. Complete the dialog box selections; you can specify the font, color, or rotation.
4. Click *OK* to dismiss the dialog box. A rectangle representing the text is attached to the

pointer.

5. Use the mouse to move the text.
6. Click to place the text at the desired location.

- You can place multiple copies of the text. Just click at each location where you would like text. When you are through placing text, select the selection tool or press ESC.
- You can create multiple lines within a text object by pressing Ctrl+Enter to create the new line. This is useful for creating piped PLD commands without having to place multiple lines of text. Piped SPICE commands must be placed as separately placed lines of text.

Shortcut

Tool palette: 

To edit text display properties:

1. Select the text.
2. From the *Edit* menu, choose the *Properties* command.
3. In the dialog box that appears, change the font, color, or rotation, then click *OK*.

Shortcut

Mouse: Double-click the text to edit.

To place a pin:

1. From the *Place* menu, choose *Pin*. The *Place Pin* dialog box appears.
2. Edit the values as necessary.

Name	The name can be up to 128 characters long and may include any character. If you place multiple copies of the pin and the name ends with a numeric component, that final numeric component increments by one with each successive pin you place. Note: If you are using Capture design with PCB Editor, make sure that the pin names do not exceed 255 characters.
Number	The number can be up to 128 characters long and may include any character. If you place multiple copies of the pin and the number ends with a numeric component, the final numeric component increments by one with each successive pin you place.
Shape	Select one; the choices are CLOCK, DOT, DOT CLOCK, LINE, SHORT, SHORT CLOCK, SHORT DOT, SHORT DOT CLOCK and ZERO LENGTH. If

	you select a pin type of POWER, the pin shape is set to ZERO LENGTH automatically. Note: You can also specify a user-defined pin shape if the pin shape is available in the CAPSYM.OLB library.
Type	Select one; the choices are 3STATE, BIDIRECTIONAL, INPUT, OPEN COLLECTOR, OPEN EMITTER, OUTPUT, PASSIVE, and POWER. The Design Rules Check tool uses pin type to check electrical rules.
Width	Select Scalar or Bus. If you choose Bus, the pin name must be of the form basename[m..n] where m..n specifies a range of decimal integers representing the number of bus members. For more information, see Naming Conventions.
Visibility	If you are placing a power pin, you can select the Pin Visible box to cause the pin to cancel the pin's global attribute. This is useful if you want to create an isolated power net. If a power pin is visible, it must be connected to a wire. For more information see About power and ground pins or Isolating power or ground. Some netlist formats do not accept certain characters in pin names. See the description for the netlist format you want to use.

3. When you have defined the pin, click *OK*.
   The pin appears attached to the periphery of the part.
4. Use the mouse to move the pin to its intended location and click to place it.
   The pin appears in the selection color until you move the pointer.
5. If you want to place additional pins, repeat step 1-4.
   As you place successive pins, any final numeric component of the pin name or pin number increases by one.
6. If you need to edit pin properties, leave the pin selected, then from the *Edit* menu choose *Properties*, make the changes in the *Pin Properties* dialog box, then click *OK*.
7. When the pins are placed, select the selection tool, or press ESC to dismiss the pin tool.
8. If the part body does not coincide with the part-body border, draw a line from the pin's connection point to the part body. You may need to temporarily turn off the *Snap to grid* option (*Options* menu - *Preferences* command - *Grid Display* tab) while you draw the line.

If you want an overbar over a signal name, follow each character in the name with a backslash (\).

You can edit every pin in the package using the *Package Properties* spreadsheet editor. For more information, see Using the spreadsheet editor.

You may want to use the *Pin Array* tool for placing large numbers of pins even though the properties or pin numbers vary. After placing the pins, you can select those you want to change,

then from the *Edit* menu, choose *Properties*; this brings up the spreadsheet editor in which you can edit the properties of many pins.

You cannot move pin names or pin number text when you are creating or editing a part, but you can make all the pin names on a part invisible. In the part editor, double-click within the part editor, but not on any graphic element, to bring up the *Edit Part* dialog box. Change *Pin Names Visible* from `True` to `False`.

Shortcut

Tool palette: 

To place an array of pins

1. Verify that you are in the *Part* view of the part editor.
2. From the *Place* menu, choose *Pin Array*.
   The *Place Pin Array* dialog box appears.
3. Edit the values as necessary.

Starting name	The starting name can be up to 128 characters long and may include any character. The leftmost or upper pin of the array is assigned the starting name. If the starting name ends with a numeric component, that component increments by the increment amount from top to bottom and from left to right.
Starting number	The starting number can be up to 128 characters long and may include any character. The leftmost or upper pin of the array is assigned the starting number. If the final component of the starting number is numeric, the pin numbers change by the increment amount from top to bottom and from left to right.
Number of pins	An integer
Increment	A positive or negative integer. Pin names and pin numbers that end with a numeric component are affected by the increment value. The numeric portion of the pin name or pin number changes by this amount from top to bottom or from left to right.
Pin spacing	A positive value. The distance between the pins is measured in grid units.
Shape	Select one; the choices are CLOCK, DOT, DOT CLOCK, LINE, SHORT, ZERO LENGTH. If you select a pin type of POWER, the pin shape automatically is set to ZERO LENGTH.
Type	Select one; the choices are 3STATE, BIDIRECTIONAL, INPUT, OPEN COLLECTOR, OPEN EMITTER, OUTPUT, PASSIVE, POWER. Pin type is

	used by the Design Rules Check tool to check electrical rules. Note: Some netlist formats do not accept certain characters in pin names. See the description for the netlist format you want to use.

4. When you have completely defined the array, click OK. The array is attached to the periphery of the part; the part body border automatically increases in size if necessary.
5. Use the mouse to move the array to its intended location and click to place it. The array appears in the selection color until you move the pointer.
6. Select the selection tool to dismiss the pin tool.
7. If the part body does not coincide with the part body border, draw a line from the pins' connection points to the part body. You may need to temporarily turn off the Snap to grid option (Options menu, Preferences command, Grid Display tab) while you draw the lines.

Shortcut

Tool palette: 

To connect a pin to a non-rectangular part body:

1. Place the pin on the part body border.
2. From the *Options* menu, choose *Preferences*, then choose the *Grid Display* tab.
3. In the *Part and Symbol Editor* group box, disable the *Cursor Snap to Grid* option, then click *OK*.
4. Draw a line between the pin and the part body.
5. If the line does not look like the pin, edit the line's style and width.
6. From the *Options* menu, choose *Preferences*, then choose the *Grid Display* tab.
7. In the *Part and Symbol Editor* group box, enable the *Cursor Snap to Grid* option, then click *OK*.

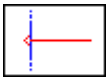
> The size of a part or a symbol is limited to 32 by 32 inches.

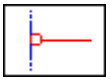
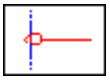
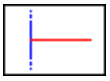
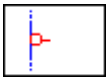
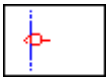
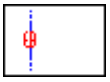
Pin Shapes

When you place pins on a part body, you can specify the shapes of the pins. You can use the predefined System-Defined Pin shapes or you can create your own Pin Shape.

System-Defined Pin shapes

Capture provides a list of system-defined pin shapes that you can use when you create a new part or edit the pins shapes on an existing part.

	**Clock** Clock symbol

	**Dot** Inversion bubble
	**Dot-Clock** Clock symbol with inversion bubble.
	**Line** Normal pin with lead three grid units in length.
	**Short** Normal pin with lead one grid unit in length.
	**Short Clock** Clock symbol with lead one grid unit in length.
	**Short Dot** Inversion bubble with lead one grid unit in length.
	**Short Dot-Clock** Clock symbol with inversion bubble with lead one grid unit in length.
	**Zero length** Normal pin with lead zero grid units in length.

User-Defined Pin Shapes

You can create your own pin shapes in Capture. You can then use these pin shapes on new or existing parts.

To create a pin shape:

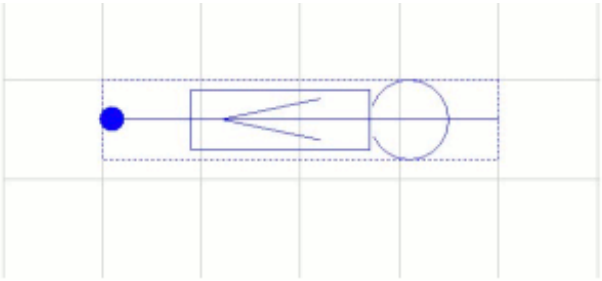
1. Open the `CAPSYM.OLB` library from the following installation path: `%CDSROOT%\tools\capture\library\capsym.olb`.
   (see Opening a library)
2. Select the library (`capsym.olb`) in the Project manager and choose *Design - New Symbol*.
   OR
   Right-click on the library (`.olb`) and choose *New Symbol* from the shortcut menu.
3. Enter a name for the new pin shape.
   A symbol name length cannot exceed 31 characters.
4. Choose the *Pin Shape* option in the *Symbol* type group and click *OK*.
   The *Part Editor* page opens with an empty rectangle defining the boundary of the pin

shape.

5. Draw the pin shape using the available shapes on the *Draw* toolbar.
   The new pin shape is now available in the selected library.
   You can now use this pin shape by placing it on a part (see *To place a pin* section).

When **Creating User-Defined pin shapes**:

- You can choose Line, Arc, Polyline, Bezier curve, Rectangle, Ellipse, Elliptical arc shapes to create a user-defined pin shape around a dot connection point.



- The bounding box of the pin shape determines the length of the pin. If the bounding box does not completely enclose the graphics then you may see your pin offset from the boundary of the symbol. For a correct adjustment, ensure the bounding box completely encloses the graphics.
- The Capture Part and Schematic Page editors require a dot connection point to be on the grid so for the pin start point to touch the body of the symbol, you must keep the end points on the grid.
- You should not draw objects enclosing the dot connection point. That may not show the pin shape correctly on the Capture Part and Schematic Page editor.
- IEEE symbols and pictures are not supported in user-defined pin shapes.

When **Adding User-Defined pin shapes** to parts in your designs:

- To place a user-defined pin shape on a part, the pin shape must be available in the `CAPSYM.OLB` library in the path `%CDSROOT%\tools\capture\library\capsym.olb`.
  If you create your pin shape in a user-defined library, you need to copy that symbol into the `CAPSYM.OLB` library.
  Capture reads the user-defined pin shapes from thise location and populates the pin shape names in the *Shape* cell of the *Pin Properties* dialog in Part Editor.
- If you place a new part in a design that has pin shapes assigned to pins, Capture will search for the pin shapes in the CAPSYM.OLB library. When the part is placed on the design and the pin shapes are found in the CAPSYM.OLB library, the pin shapes are cached to the design cache. Once you save the design, the pin-shapes will be read from the design cache.

- If a pin shape is not available in the CAPSYM.OLB library, the default pin shape, which is a line, will display on the part.
- If you downgrade a design containing user-defined pin shapes from any Capture version greater than 16.2 to 16.2 or any previous version, the user-defined pin shapes will be removed from the design cache. You can still open the design in 16.2 but the pins will show with default pin shape, line. However, if you reopen this 16.2 design in any version greater than 16.2, Capture will seach the CAPSYM.OLB library for the user-defined pin shapes used in the design. The pin shapes found in the library will display on the schematic. For pin shapes not found in the library, the default, line, and shape, will display.
- If you run the Replace cache command on a user-defined pin shape, say, A with B in the design, all instances of the user-defined pin shape A will be replaced with B in the design, and the Pin Shape property on the pins will be updated to the new pin shape value. This property is an instance override. At any time, if you want to revert to the library-level pin shape value, you can use the Delete property in the Property editor and it will delete the instance override and this will be reflected on the schematic. However, if you do an Edit part, then it will still show the part level user-defined pin shape and not the instance override that exists in the schematic.

P in types

3 State	A 3-state pin has three possible states: low, high, and high impedance. When it is in its high impedance state, a 3-state pin looks like an open circuit. For example, the 74LS373 latch has 3-state pins.
Bidirectional	A bidirectional pin is either an input or an output pin. For example, pin 2 on the 74LS245 bus transceiver is a bidirectional pin. The value at pin 1 (an input) determines the active type of pin 2, as well as others.
Input	An input pin is one to which you apply a signal. For example, pins 1 and 2 on the 74LS00 NAND gate are input pins.
Open Collector	An open collector gate omits the collector pull-up. Use an open collector to make "wired-OR" connections between the collectors of several gates and to connect with a single pull-up resistor. For example, pin 1 on the 74LS01 NAND gate is an open collector gate.
Open Emitter	An open emitter gate omits the emitter pull-down. The proper resistance is added externally. ECL logic uses an open emitter gate and is analogous to an open collector gate. For example, the MC10100 has an open emitter gate.
Output	An output pin is one to which the part applies a signal. For example, pin 3 on the 74LS00 NAND gate is an output.

Passive	A passive pin is typically connected to a passive device. A passive device does not have a source of energy. For example, a resistor lead is a passive pin.
Power	A power pin expects either a supply voltage or ground. For example, on the 74LS00 NAND gate, pin 14 is VCC and pin 7 is GND. It is not a good idea to use overbars above power pin names; if you do, any netlists that you create will have invalid power pin names. Power pins are invisible.

Creating a Part Convert

You can store a part convert with a library part, then place either the normal view of the part or its convert.

To add a convert while you are creating a part

1. From the *Design* menu of the library's project manager window, choose *New Part*.
   The *Part Properties dialog* box appears.
2. In the *Edit Part Properties* dialog box, enable *Create Convert View*.
3. Fill out the rest of the dialog box and click *OK*.
   For more information, see *Creating a part*.
4. From the *View* menu, choose the *Convert* command.
5. Use the part editor to define the equivalent part and add the pins.
6. Choose *View - Package* to see the convert.
7. Choose *File - Save*.

To view a part convert

- From the part editor's *View* menu, choose *Convert*.
  If you place the regular view of a part and you need to use the convert view instead, you can use the *Graphic* property in the property editor to change the view.

Creating a Part Alias

A part may come with several speed ratings or be made by several manufacturers. If all the variations have a common graphic and PCB footprint, you may not have the time and space to create and store a different library part for each variation. Instead, create a single library part and assign it multiple aliases.

To create or add a part alias

1. Open the *New Part Properties* dialog box.
   a. In the *Project manager*, select the design (.OLB) file.
   b. From the *Design* menu, choose *New Part*

   OR

Open the Edit Part Properties dialog box.

a. In the *Project Manager*, select the part to edit.
b. From the *Edit* menu, choose *Part*.
c. From the *Options* menu, choose *Package Properties*.

2. In the *New Part Properties* dialog box or the *Edit Part Properties* dialog box, click *Part Aliases*.
3. In the *Part Aliases* dialog box, click *New*.
4. In the *New Aliases* dialog box, enter the name of the alias and click *OK*.
5. Click *OK*, as needed, to dismiss the remaining open dialog boxes.

Creating a Part from a Spreadsheet

You can use the *New Part Creation Spreadsheet* to create new parts (multi-section/single-section). The *New Part Creation Spreadsheet* has a spreadsheet-like interface that allows you to paste contents copied from a part data sheet to the spreadsheet.
Each row in the *New Part Creation Spreadsheet* corresponds to a pin while each column corresponds to properties associated with the pins. The property names are listed as the column header.

> Before you copy and paste part information (*pin number*, *pin name*, *pin type*, and so on) from a data sheet, make sure that you arrange the part information in the same column header format/sequence as it appears in the New Part Creation Spreadsheet.

To create a new part from spreadsheet

1. Select a library (.OLB) file that will contain the new part in the project manager.
2. Select the *Design* menu and choose the New Part from Spreadsheet command.
   OR
   Right-click on the library file and select *New Part from Spreadsheet* from the pop-up menu.
   The New Part Creation Spreadsheet is displayed.
3. Specify a name for the new part in the *Part Name* text field.
4. If you want to create a multi-section part, specify the number of sections you want to have in your new part in the *No. of Sections* text field. The *New Part Creation Spreadsheet* creates single-section parts, by default.

   > The *Section property* column changes to a list box displaying the number of sections you specified in the *No. of Sections* text field.

   > ***Number of sections cannot be less than one.***

5. Specify a part reference prefix for the part in the *Part Ref Prefix* text box.

6. Select *Numeric* or *Alphabetic* in the *Part Numbering* group.
   If you select *Alphabetic*, an alphabet (between A to Z) will be added as a suffix to the current part reference for each of the new parts. If you select *Numeric*, a number (between 1 and 1024) will be added as a suffix to the current part reference for each of the new parts. The *Section* property column changes based on your selection in the *Part Numbering* group. For example, if *Alphabetic* is selected, the Section property column displays "A".
7. Specify a pin number in the *Number* property column.
   To sort on any property, double-click its name in the column header.
8. Specify a name for the pin in the *Name* property column.
9. Select the type of pin from the *Type* property column list box.
   You can select the *Type* cells for multiple pins simultaneously using the Shift+down Arrow keys and then enter the pin type. The selected *Type* cells get populated with the pin type of your choice.
   Alternatively, you can:
   a. Select the *Type* cells for multiple pins simultaneously using the Shift+click, then press the Ctrl key, and then select a pin type of your choice from the list box.
      The selected Type cells get populated with the pin type of your choice.
   b. Click the first cell of the range, and then drag to the last cell, and then enter the pin type of your choice.
      The selected Type cells get populated with the pin type of your choice.
      (You can use these methods to make selections in the *Shape*, *Position*, and *Section* property column list boxes also).
10. Select a shape for the pin from the *Shape property* column list box.
    You can hide or unhide a property column in the *New Part Creation Spreadsheet*. To do this, right-click the property column header you want to hide and select Hide from the pop-up menu. The selected property column will now not appear. To unhide a property column, right-click the property column header next on the right-hand side of the hidden property column and select *Unhide* from the pop-up menu. The hidden property column appears in the New Part Creation Spreadsheet.
    Alternatively, you can unhide a property column by:
    - Double-clicking the column handle () of the property column header.
    - Dragging the column handle of the property column header.

    (only the last two methods can be used to unhide a property column, which is the last column in the *New Part Creation Spreadsheet*)
    You can change the order in which the property columns and rows appear in the *New Part Creation Spreadsheet*. To do this, select the property column/row header you want to move and drag and drop it to the location where you want it in the New Part Creation Spreadsheet.
11. Specify a value for a swappable (input) pin of the part in *PinGroup* text field.
12. Select the position where you want this pin to appear in the part for the *Position* property

column list box.

13. Select a section number you want to associate with the pin from the *Section* property column list box.
    You can select Section cells for multiple pins simultaneously using the Shift+down arrow keys and enter the section number. Alternatively, you can:
    - Select the Section cells for multiple pins simultaneously using the Shift+click, then press the Ctrl key, and then select a section number of your choice from the list box. The selected Section cells get populated with the section number of your choice.
    - Click the first cell of the range, and then drag to the last cell, and then enter the section number of your choice. The selected *Section* cells get populated with the section number of your choice.

    You can select alternate Section cells for multiple pins simultaneously using the Ctrl+Left mouse button click and enter the section number.

    > The pins will be added in the same order in the new part as the order in which the pins appear in the spreadsheet

14. Click the *Save* button to save the new part.
    If any warnings are generated during the save operation, a message box appears asking you whether you want to view the warnings.
    To view the warnings, click the *View Warnings* button. The New Part Creation Spreadsheet expands and displays a grid showing warnings messages. If you select the *Continue* button, the part is saved as is.

    > Click the *Hide Warnings* button to hide the warning messages or the *Show Warnings* button to display the warning messages again.

> If the new part you created contains one part per package then the part created would be homogeneous otherwise the part will be heterogeneous.

To add a pin

1. From the *New Part Creation Spreadsheet*, click the *Add Pins* button.
   The *Add Pins* dialog box appears.
2. Specify the number of pins you want to add in the *Number of Pins* text field.
3. Click *OK*.
   The desired number of blank rows gets added at the end of the current row set in the *New Part Creation Spreadsheet*.

To delete a pin

1. Select a row in the *New Part Creation Spreadsheet*.

2. Click the *Delete Pins* button.
   A message box appears asking you to confirm the deletion.
3. Click *OK* to confirm deletion.
   The selected row is deleted from the *New Part Creation Spreadsheet*.

Editing and Renaming a Part

If you have a library part that is nearly perfect, you can tailor the original part so that it suits your project. You can edit the part properties and you can change its graphical representation or its pins.

To edit a library part

1. Open the library containing the part.
2. In the *Project Manager*, double-click on the part.
   The *Part Editor* opens with the part displayed.
3. Make changes to the part body definition using graphics, text, and images.
4. Make the needed changes to pins.
   You can move pins after you select them or add pins. To edit pin properties, double-click on the pin.
   You cannot move pin names or pin number text when you are editing a library part. You can move pin names and pin numbers only when you are editing a part instance on a schematic page. For more information, see *Moving pin name and pin number text* section.
5. From the *File* menu, choose *Save*.

To edit a part instance on a schematic page

1. Select the part in the schematic page editor.
2. From the *Edit* menu, choose the Part command.
   The part editor opens with the selected part displayed.
3. Edit the part as required.
4. From the *File* menu, choose the *Close* command.
   The *Save Part Instance* dialog displays:
   Update All replaces the old part in the design cache with the newly edited part and breaks the link with the original library.
   Update Current creates a new part in the design cache. The new part has no link to the original library.

The *Part Editor* window closes and the updated part appears in the schematic page editor.

The edited part does not exist in a library, so the only way to place a copy of it is to use the *Copy* and *Paste* commands on the schematic page editor *Edit* menu.

The edited part has no link with the original library, so it is not affected by the *Update Cache*

command. To restore its link with the original library, choose the *Replace Cache* command from the Project Manager *Design* menu. For more information, see Replacing and Updating the cache .

When you open the part editor from the schematic page editor, the part you are editing cannot be selected on the schematic page. After you close the part editor window, the part can be selected.

You can move pin names and pin number text when you are editing a part instance on a schematic page. For more information, see *Moving pin name and pin number text* section.

When you edit a part's graphic representation on a schematic page, you break the connection between the part and the library; if you want to reverse your edits, you use the Replace Cache command of the *Design* menu.

M oving pin name and pin number

You can move pin names and pin numbers only when you are editing a part instance on a schematic page.

- To move a pin name or pin number, select the pin name or number text and drag it to the desired location.
- To reset a pin name movement, select the pin name text you had moved and from the *Edit* menu, choose *Reset Location*, then *Pin Name*.
- To reset a pin number movement, select the pin number text you had moved and from the *Edit* menu, choose *Reset Location*, then *Pin Number*.

> You cannot move pin names or pin number text when you are creating or editing a library part.

To rename a part

1. In the Project Manager, select the part.
2. From the *Design* menu, choose *Rename*.
   The *Rename* dialog box appears.
3. Enter the new name and click *OK*.

Part Properties

When you need to edit the properties of individual parts, follow the instructions below; when you need to edit properties for several parts, you can save time by using the spreadsheet editor, the *Update Properties* command or the Export Properties command.
Once you have added properties to a part on a schematic page, its properties no longer match the properties of the same part residing in the library. This part is unique in that it has properties assigned specifically to it that are not inherited from the library part definition.

If you add a user-defined property to one part in a homogeneous multiple-part package, all parts in the package inherit the property and its value. If you add a user-defined property to one part in a heterogeneous multiple-part package, the other parts in the package are not affected.
You can also edit properties on part packages, in which case the changes appear on every part in the package, and on every part instance.
When you are editing the properties of a library part, you use the Edit Part Properties dialog box. In this dialog box you can add or remove user-defined properties or change the values for the following properties:

- Part Value
- Part Reference
- Primitive
- Graphic
- Packaging
- PCB Footprint
- Power Pin Visibility
- User properties

In this section:

Assigning Properties to a Part

Assigning a reference designator

When you place parts on a schematic page, all parts of the same type are assigned the same part reference. For example, C? is assigned to all capacitors. Regardless of the ultimate purpose of your design, each part needs a unique identifier. You can assign part references by editing individual parts in the part editor, or, for PCB designs, by[creating a swap file|Designating pins_44 gates_44 or packages for swapping#1490563] to use with the Back Annotate tool. For uniquely identifying parts, it is more convenient to use the Annotate command on the Tools menu.
The Annotate tool assigns unique alphanumeric part references. For PCB designs, it assigns individual parts to a package and assigns unique pin numbers to each part in a multiple-part package. References are assigned in order from top to bottom and left to right; parts located at the top of the page have the lowest numerical designation. If two parts share a vertical coordinate, the part further to the left has the lower numerical designation. If you add parts after you have assigned part references, you can easily remove part reference assignment and run Annotate again.
In general, you use Annotate after you have placed all parts and before you use other Capture

tools. See Annotating the Design for an overview of the design processing tools.
You can update references incrementally, so that previously assigned part references are not changed, or you can update unconditionally.
Capture automatically updates either instances or occurrences depending upon the type of design you are working with. In general, you should update instances for FPGA and PSpice projects, and update occurrences for PCB and Schematic projects.

> If you want to preserve a reference designator during annotation, choose *User Assigned Flag - Set* from the pop-up menu of the part on the schematic page. You can also open the part in *Properties Editor* and from the pop-up menu of *Reference,* choose *User Assigned Flag - Set*. The User Assigned Flag is set by the tool if the reference designator is changed using the *Property Editor* or *Schematic Editor* or through *Backannotation*.
>
> Note that if the property does not exist on the occurrence, the *User Assigned Flag* is set for the instance and thus will be used for all occurrences of that part unless the property is explicitly overridden on the occurrence. If you want to set the *User Assigned Flag* for the occurrence value, set the property on the occurrence from the property editor and then set the flag.

Assigning other properties

You can assign other properties to your library parts just as you would assign them to instances or occurrences in your schematic design. See Assigning properties for more information.

To globally change the visibility of pin names or pin numbers

1. With the part open in the part editor, double-click on it to bring up the User Properties dialog box .
2. Select the *Pin Names Visible* property to control pin name visibility.
3. Select the *Pin Numbers Visible* property to control pin number visibility.
4. Set the property to FALSE to hide the pin names (or numbers).
5. Set the property value to TRUE to show the pin names (or numbers).
6. Click *OK* to apply the change.

Property tables

You can choose a user-defined property from the property tables that follow and assign a value to the property for use with another tool (such as PCB Editor or Layout).

Part property name	Example value	Description
COMPFIXED	YES	If the value is YES, the part (such as an edge connector) is permanently fixed to the board.

COMPGROUP	2	An integer value that assigns the part to a group for placement. The value must be numeric, and between 0 and 100.
COMPKEY	YES	Used to designate a component as the key component in a given group. The key component is placed first, with all the other components in the group placed in proximity to it.
COMPLOC	[1000, 1000]	Part location on the board as X and Y coordinates. Use the following format [X, Y], where X and Y represent the coordinates. Both must be integers in mils or microns.
COMPLOCKED	YES	If the value is YES, the part is temporarily locked in position.
COMPROT	270.00	Part rotation in degrees and minutes counterclockwise from the orientation defined in the Layout library. Use a period (.) to separate degrees and minutes.
COMPSIDE	BOT	Determines which side of a board a part will reside on, TOP or BOT.
FOOTPRINT	DIP24	An explicit definition of the footprint name to attach to the component.
FPLIST	DIP24\400	Comma-delimited list of alternate footprints to attach to components, to ease switching between footprints.
GATEGROUP	1	Identifies gate swapping restrictions within a component. To be swapped, two gates must belong to the same gate group.
PARTNUM	489746	A customer part number that is generally unique for each customer and identifies the exact part, including the manufacturer and case type.
PARTSHAPE	74LS04	A generic part number (such as 74LS04 or CK05) that represents a certain part throughout the industry, but may not identify the manufacturer or case type. If no footprint is defined, or the correct footprint is not found, PARTSHAPE value is compared to the data in

		SYSTEM.PRT (in ORCADWIN/LAYOUT/DATA) and the footprint listed in SYSTEM.PRT is used.
POWERPIN	YES	Defines non-wired pins (such as unusual voltages) as belonging to a particular net. POWERPIN is typically used to override the standard GND or VCC attachments to particular pins of an IC.
CONNWIDTH	10	Sets the track width, leaving MINWIDTH and MAXWIDTH at their defaults.
HIGHLIGHT	YES	If the value is YES, the net is highlighted.
MAXWIDTH	12	Sets the maximum track width.
MINWIDTH	8	Sets the minimum track width.
NETGROUP	2	Identifies grouped nets. Use this to select or color nets as a group in Layout (for editing or routing).
NETWEIGHT	60	Integer between 1 and 100 assigning relative priority to the net. The default value is 50.
PLAINLAYERS	GND	Assigns a net to a plane layer. Values are GND and POWER.
RECONNTYPE	ECL	Specifies the reconnect rules for each type of reconnect. Values are STD, HORZ, VERT, NONE, or ECL.
ROUTELAYERS	GND,PWR	Restricts the layers on which this net can route.
SPACINGBYLAYERS	TOP=13,BOT=8	Net spacing for one or more layers.
TESTPOINT	YES	If the value is YES, a test point is automatically assigned to the net.
VIAPERNET	VIA1	Via types allowed for net.
WIDTH	12	Track width value assigned to the MINWIDTH, MAXWIDTH, and CONNWIDTH properties unless overridden.
WIDTHBYLAYER	TOP=6,BOT=12	Net width for one or more layers.

Renaming and deleting part properties

Capture allows you to rename and delete a part property in every part instance that contains the property for an entire design.

To rename a part property

1. In the project manager, select the design file or the schematic page.
2. Select the Rename Part Property command from the *Edit* menu.
   The Rename Part Property dialog box is displayed.
3. In the *Find User Property* text field, type the current property name.
4. In the *Replace with User Property* text field, type the new property name.
5. Click *OK*.

To delete a part property

1. In the project manager, select the design file or the schematic page.
2. Select the Delete Part Property command from the *Edit* menu.
   The Delete Part Property dialog box appears.
3. In the *Property Name* text box, type the property name.
4. Click *OK*.

Update Properties using an Update File

If you need specific information, like stock number or supplier, on your parts, you can add a property and set the property value in the library. Later, when you place the part in a project, the information is present; there's no need to add information to the placed part.
You can edit parts individually in the part editor, but when you wish to update properties of a number of parts, the Update Properties tool is very convenient. You can use the Update Properties tool to edit any properties except part value and part reference.
Before you run the Update Properties tool, you create an update file as described in Creating an update file. To identify the parts you wish to update, you specify an identifying property or combination of properties called a combined property string. For each combined property string you use, you must create a separate update file and run the Update Properties tool.

To update part properties

1. In the Project manager, select schematic folders or schematic pages if you want to process only a portion of the design.
   If you want to process the entire design, leave the schematic folders or schematic pages unselected.
2. Choose *Tools - Update Properties* command.
   The Update Properties dialog box appears. Verify that the dialog box options are set as required.

For example, you specify, among other things, whether you are updating parts or nets, whether you want to overwrite existing property values, and the name and location for the update file.
Unless you select the button to unconditionally update the property, properties that currently hold a property value are not updated.

3. In the *Property Update File* box, enter the name and location of the update file.
   OR
   Use the *Browse* button to navigate to the file.
4. Click *OK*.

If you edit a library provided by OrCAD, it is important that you assign a new library name so that your changes are not overwritten when you upgrade or update your software.

Capture preserves the case of part names and net names, but ignores the case when comparing names for electrical connection. That means you may use upper-case or lower-case letters as you wish and you need not remember the case.

Capture report files are text files and can be opened in any text editor. You may want to use the tab alignment capability of your word processor to line up reports correctly. Spreadsheets will automatically align the columns of Capture-generated report files.

Replacing and Updating the cache

When you place the first instance property of a part in a project, a copy of the part is created in the design cache. The design cache stores a copy of every part used in the design. (You can think of it as an embedded library.) All instances of the part normally refer to this copy in the design cache. A cache part also retains a link to the library part on which it is based, so you can update or replace all of the parts in the design cache to synchronize them with the parts in the libraries.
The project manager updates the display of the design cache every time you open the cache. Just click on the design cache icon to close or open the design cache.

Using the Replace Cache and Update Cache commands

When you use the Update Cache command or the Replace Cache command with the option to preserve schematic part properties, the part retains all instance and occurrence properties. This means you will not lose any changes made to pin properties after the part was placed, including those made by the *Back Annotate* or the *Annotate* tools.
The *Replace Cache* and *Update Cache* commands are quite similar. However, there are a couple of significant differences between the two commands.
You can modify a part's link to the library (part name, path, and library) with *Replace Cache*, but not with *Update Cache*. *Update cache* only brings in new data when the path has changed. Another difference is that if the path and library names do not change, Replace Cache reloads the part definition into the design. However, if *Update Cache* finds that the part name and the library names are the same, it does not bring in part changes.

Replacing parts in a design

If you need to replace a part in your project, you could open the schematic page editor to find and delete each instance of the part, then place the replacement part. If your design includes many instances of this part, you can more easily achieve the same end with the Replace Cache command.

Restoring parts in a design

When you delete a part, you also delete all of its properties. When you use the *Replace Cache* command to restore the part, the properties of the part instance are attached to the replacement part, but the pin properties are not restored.
You can also use the *Replace Cache* command if you want to undo edits to a part on a schematic page and restore the part's link to its library.

You would also use the *Replace Cache* command after you edit a part on a schematic page if you want to undo the edits and restore the part's link to its library.
When you delete a part, all of its properties are also deleted. When you use the *Replace Cache* command, properties of the part instance are attached to the replacement part; however, the pin properties are lost.

> When you use the *Update Cache* command or the *Replace Cache* command with the option to preserve schematic part properties, all instance and occurrence properties of the schematic part are retained. This means you will not lose any changes made to pin properties after the part was placed, including those made by the *Back Annotate* or the *Annotate* tools.

To replace a part throughout a design

1. From the design cache, select the part you want to replace.
2. Choose *Design - Replace Cache*.
3. In the dialog box that appears, enter the name of the replacement part and the library that contains it.
4. Select the *Replace schematic* part properties action if you want to completely replace the part and its properties. Otherwise, use the default action to preserve schematic part properties.
5. Click OK. When the project manager appears again, double-click on the design cache to verify that the replacement part is listed instead of the original part.
   If you need to know a part's library of origin, you can select the part in the project manager, then select *Replace Cache* from the *Design* menu. The part name and the library and path are listed in the dialog box that appears.
6. Click *Cancel* to return to the project manager.

You can discover the library of origin for multiple parts by Creating a cross reference report.

Part Packages

Parts are the basic building blocks of a design. For PCB designs, part may represent one or more physical components; or it may represent a function, a simulation model, or a text description for use by an external application. The part's behavior is described somehow, whether by a SPICE model, an attached schematic folder, HDL statements, or other means.

Parts in PCB designs usually correspond to physical objects—gates, chips, connectors, and so on—that come in packages of one or more parts. "Multiple-part packages" are physical objects that comprise more than one part.
Each logical part has graphics, pins, and properties that describe it. As you place the parts in a package to suit your design requirements, Capture maintains the identity of the part package for back annotation, netlisting, bills of materials, and processes that require it. Parts inherit this information from the package.
You specify packaging information when you create a part. You can also change it in the Part Editor (from the View menu, choose package; then, from the Options menu, choose the Package Properties command.
The parts in a package may have different pin assignments, graphics, and user properties. If all the parts in a package are identical except for the pin names and numbers, the package is homogeneous. If the parts in a package have different graphics, numbers of pins, or properties, the package is heterogeneous.

For example, a hex inverter is homogeneous: the six inverters are identical, except for their pin numbers. A relay, which has a normally opened switch, a normally closed switch, and a coil, is heterogeneous: the three physical parts differ in graphics, number of pins, and properties.

When you place a part on a schematic page, you actually create an instance of the part. A part instance is like a "snapshot" of the part in the library; that is, it inherits all the properties of the library part. Once a part instance is on the schematic page, you can edit the properties of that instance without changing the properties of any other instance. The instance values of those properties supercede the values of any identical properties that exist on the library part. For more information, see Instances and occurrences.

When using multiple-part packages in your design, you must either make all shared pins visible and connect them to power and/or ground nets, or you must make them all invisible, in which case they are connected according to their pin names. See Making power pins visible for more information.

In this section:

- Creating a Package
- Editing, Deleting, and Viewing a Package

Creating a Package

Creating a package

A part or hierarchical block may have an underlying hierarchical description, such as an attached schematic folder. If it does, it's called a non-primitive. A part or hierarchical block that has no underlying hierarchical description is called a primitive. In Capture, this characteristic is defined in a property, called Primitive, on every part instance. You can change the Primitive property as often as you like during the design process. When a part or hierarchical block is marked as primitive, all of Capture's tools treat it as such. You cannot descend into a part or hierarchical block that is marked as primitive, even if it has an attached schematic folder.

For example, you might create a part and attach a schematic folder that describes its gates and wiring, and then attach schematic folders to some of those parts to describe their transistors. Before you create a netlist for simulation, you should specify those parts as non-primitive, so Create Netlist can descend far enough to find the transistor-level descriptions. Before you create a netlist for board layout, you should specify the parts as primitive, so Create Netlist stops at the gate-level descriptions. Bill of Materials and Cross Reference work similarly.

For part instances that have their Primitive property set to Default, you can configure Capture to treat them as either primitive or non-primitive on a design-wide basis, using the Design Template and Design Properties commands on the Options menu. This is useful when you are describing and simulating your design at varying levels of abstraction (as in a top-down design).

If you attach a schematic folder to a homogeneous part, it is attached to each part in the package, not the package itself. You cannot attach a schematic folder to a heterogeneous part.

When you attach a schematic folder to a part or hierarchical block, you can specify a full path and file name in the Library text box. So, although you can specify a library that has not been saved, you should not try to descend into the attached schematic folder until the library that contains the schematic folder has been saved.

If you do not specify a full path and file name in the Library text box, Capture expects to find the attached schematic folder in the same design as the part of hierarchical block to which it is attached. If the specified schematic folder does not exist in either the design or library, Capture creates the schematic folder when you descend the hierarchy on the part or hierarchical block. For compatibility with future versions of Windows, Capture preserves the case of the path and filename as you specify them in the Library text box.

Using the package view

In this view, you can add and edit package properties, though parts are not available for editing. Use this view if you want to add properties that are the same for all parts in the package.
All packages include the following properties:

- Name
- Alphabetic or numeric part numbering
- Homogeneous or heterogeneous type
- Number of parts per package

- Part reference prefix
- Printed Circuit Board footprint
- Aliases

Forcing multiple parts into a single package

For PCB (Printed Circuit Board) designs, if you need to make sure that two or more parts in your design are in the same package, you use the Update Part Reference tool. First, choose a property that all the parts share and verify that the parts all have the same value for that property, then use the Update Part Reference tool.

For example, you might have several NANDs in a schematic folder and four that are in close proximity in the final product. For each of the four NANDs, create a user-defined property named `COMPGROUP` and set the property's value to `1`. In the `Combined Property String` text box in the *Update Part Reference* dialog box, enter {`COMPGROUP`}.

To specify ignored package pins

The `IGNORE` property, available for package pins, provides a method for you to specify that certain pins on a part are ignored when the part is placed on a schematic page. Pins that have the `IGNORE` property assigned to them do not appear on the schematic page. These pins will also not be included on the part footprint for any downstream PCB layout tools. Also, note that ignored pins will not be included in any back annotation from a layout tool.

1. In the Part Editor, Choose *View - Package*.
2. Choose *Edit - Properties*.
   The *Package Properties* dialog box appears.
3. Select the pins you wish to ignore, and select the `IGNORE` property for those pins.
4. Click *OK*.

To force multiple parts into a single package

1. Choose one property that the parts share and assign the same value to that property for each part. You may want to add a user-defined property to each part.
2. In the project manager, select schematic folders or schematic pages if you want to process only a portion of the design. If you want to process the entire design, leave the schematic folders or schematic pages unselected.
3. Choose *Tools - Annotate*.
   The *Annotate* dialog box appears.
4. In the Combined Property String text box, enter the property name. The name must be enclosed in braces: "{" and "}".
5. Verify that the remaining dialog box options are set the way you want them. For example, specify, among other things, whether you are unconditionally updating all references or only those that are set to the unassigned (?) reference.

6. Click OK.

Shortcut

Toolbar: 

To create a multiple-part package

1. Open the library that will hold the part.
2. Choose *Design - New Part*.
   The *New Part Properties* dialog box appears.
3. Enter the number of parts in the package (up to 128), and specify whether they are all the same (homogeneous) or different (heterogeneous).
4. Fill in the other fields in the dialog box and click *OK*.
   The part editor opens with an empty part outline.
5. Use the graphics tools to define the part body.
6. Use the *Place Pin* dialog box to add pins to the part.
   You can share pins such as power and ground pins for each part in the package as described in Creating a shared pin.
7. If you are creating a heterogeneous package, choose *View - Next Part*, then repeat steps 4 and 5 for each part.
   OR
   If you are creating a homogeneous package, choose *View - Next Part*, then assign pin numbers for each part.
8. Choose *File - Save*.
   If you are creating the part in a new library that has not yet been saved, the *Save As* dialog box appears, giving you the opportunity to name the library file.

- If you edit a library provided by OrCAD, it is important that you assign a new library name so that your changes are not overwritten when you upgrade or update your software.
- After a part is created, you can add to or decrease the number of parts in the homogeneous package, even if the part starts out as a package of one. However, you can neither increase nor decrease the number of parts in a heterogeneous package.

Editing, Deleting, and Viewing a Package

Editing package properties

As with part properties, you can edit package properties in the library or on the schematic page. If

you edit package properties on the schematic page, the changes affect only the parts in the project; you are, in effect, creating a new part that is not stored in a library.
Package properties are inherited by every part in the package and by every part placed on a schematic page. Packages do not support user-defined properties.

1. Choose *View - Package* in Part Editor.
2. Choose *Options - Package Properties*.
   The *Edit Part Properties* dialog box appears.
3. Make your changes in the dialog box and click *OK*.
   You can change, among other things, the number of parts in the package, the PCB footprint, and the part reference prefix. The changes are reflected in the part editor, but they are not permanent until you save the design.

To edit pin-specific package properties

1. Choose *View - Package* in Part Editor.
2. Choose *Edit - Properties*.
   The Package Properties dialog box appears.
3. Make your changes and click *OK*.
   Capture updates pin property information and checks for duplicate pin numbers. The Validate and Update buttons also update pin property information and check for duplicate pin numbers.

Using part or net properties in a package

Using properties, you can conveniently store part and net information. To change part properties on a single part (or on every instance of the part in a design), edit the part in the schematic page editor. To set part properties on every instance of the part that you place, edit the part in the part editor. If you want to make changes to a number of parts or nets, the *Update Properties* command on the *Tools* menu is a convenient method. You can use *Update Properties* to edit any properties **except** part value, part reference, and netname, and you can update the properties of parts in a design or in a library.

Before you run *Update Properties*, you create an update file. To identify the parts or nets you want to update, you specify an identifying property or combination of properties. For each identifier you use, you must create a separate update file and run *Update Properties*.

You can update references incrementally, so that previously assigned part references are not changed, or you can annotate unconditionally, changing all the parts across all the schematic pages processed.
Capture automatically selects to update either instances or occurrences depending upon the type of design. In general, you should update instances for FPGA and PSpice projects, and update occurrences for PCB and Schematic projects.

To delete a part in a package of a heterogeneous part

1. Select the part that you want to delete.
2. From the Edit menu, select Delete.

> ☑ You cannot undo the deletion of part from a package. Once the part is deleted from a package, all information regarding the part is lost.

Viewing a Package

To view all parts in a package

- From the View menu, choose the Package command. An image of all parts in the package appears.

To switch to a different part in the package

1. Choose *View - Package*.
2. Double-click on the part you wish to edit.
   OR
   Choose *View - Part*.
3. Choose *View - Next Part* or *Previous Part* command.

Synchronizing parts

Capture gives you the power to place parts in a design, alter the parts in the library, then update the design so that all of the altered parts reflect the changes you have made in the library. You can use the Update Cache command on the Design menu to do this.
When you use the *Update Cache* command, properties of the part are retained; however, the pin properties of the part instance are lost.

To update a part in the design cache so it matches a part in the library

1. Using the part editor, edit the parts in the library to suit your needs.
2. Open the project manager window for the design.
3. Select the parts in the project manager.
4. Choose *Design - Update Cache* command.
   The design cache and all part instances are updated to match the library parts.

When you copy pages from one design or library to another, parts displayed on the copied pages may appear different due to differences in each design or library cache. If a part is not already in the destination design cache, Capture will copy it from the source design's cache. Otherwise, it will use the part already present in the destination design's cache.

When you use the *Replace Cache* or *Update Cache* command, all properties of the part are retained, but the pin reflects the properties of the library part. This means you lose any changes

made to pin properties after the part was placed, including those made by the *Back Annotate* or the *Annotate* tools.

The *Replace Cache* and *Update Cache* commands are quite similar, but they have the following differences:

- You can have only one part selected in the design cache when you use *Replace Cache*; you can have multiple parts selected in the cache when you select *Update Cache*.
- You can modify the part's link to the library (part name, path, and library) with *Replace Cache*; you cannot modify the part's link with *Update Cache*.

If you move a library after you place a part, the connection between the part and its library is broken. In this case, the *Update Cache* command will not find the library; you will need to use the *Replace Cache* command and specify the new path to the library.

Part Instances and Occurrences

A part instance is a part you have placed on a schematic page. A part occurrence is created each time the part instance occurs in a schematic that is within the design hierarchy. So, for example, placing a part on a schematic page in your design creates both an instance and an occurrence. You can assign properties to the part instance, in which case the property (and its associated value) will "shine through" to each part occurrence unless the value is specifically replaced by a value on an individual part occurrence. The Property editor window graphically illustrates the state of instance and occurrence properties.

The instance property values shines through to the occurrence as long as the occurrence property values have not been edited in any way. When you explicitly edit an occurrence property value or when Capture modifies it via one of its tools, the occurrence values overrides the instance value. Only the occurrence value will be placed in the netlist.

If you...	**The result will be...**
place a part on an unused page in your design	only instance properties on that part.
place one part in the root schematic	one set of instance properties and one set of occurrence properties on that part.
do not modify the occurrence properties on an object	instance properties will "shine through" on the occurrence. The instance properties themselves may be shining through from the library definition (or design cache if the two differ
edit a part in a library	no effect on the part in any project.

	Use the Update Cache command or Replace Cache command to bring library changes into a design.

How Capture uses instance and occurrence properties

The type of property you update or use in Capture depends on the type of project in which you are working. If you are working with an FPGA project or a PSpice project, Capture allows you a choice, but defaults to update instances when you use the Annotate, Update Properties or Export Properties commands. It is best that you use instances to create Cross Reference and Bill of Materials reports, as well.
When you work with a PCB or schematic project, it is best to update occurrences when you use the Annotate, Update Properties, and Export Property commands. In these projects, Capture also uses occurrences to create reports with Cross Reference and Bill of Materials.

> While modifying occurrence properties, open only one occurrence at a time.

The EDIF 2 0 0, VHDL, and Verilog netlist formats generate true hierarchical netlists. Capture uses the instance property values on nets and parts when it generates a netlist for a design with one of these formats. All other netlist formats in Capture produce flat netlists and use occurrence property values.

Preferred modes for design processing

Capture determines and recommends a preferred mode for processing your design based on the type of project flow, type of design with respect to complex or simple hierarchy, and whether occurrence properties already exist on the design.
The Cross Reference Parts dialog box is an example of where you can choose to use instances or use occurrences as the preferred mode.

Use instances	If you are in an FPGA, PSpice, or digital simulation project, or if your design does not have occurrence properties.
Use occurrences	If your design has at least one schematic used multiple times or a schematic from an external library. Using occurrences is also the preferred mode if the design already has some occurrence properties.

Removing part reference assignments

If you want to incrementally update a design in which some of the schematic pages have already been updated, you can use the Annotate command to remove part references from those schematic pages.

To remove part references

1. In the project manager, select schematic folders or schematic pages if you want to process only a portion of the design. If you want to process the entire design, leave the schematic folders or schematic pages unselected.
2. From the Tools menu, choose Annotate. The Annotate dialog box appears.
3. In the Action group box, select Reset part references to "?," ;select other options as appropriate; then click OK.

Shortcut

Toolbar: 

Generating library parts

Generating a part from a PSpice model library

You can use the Generate Part dialog box to create a part that has pins matching all the hierarchical ports on the root schematic of the design for design reuse. If a destination library file does not exist, Capture creates a library part file (.OLB) that contains the newly generated part and a copy of the top-level schematic. The new .OLB appears in the Project manager's Outputs directory along with the new part and schematic. The new part has a reference back to the design where the source schematic resides so that when you descend the hierarchy from the placed part, it will open the source schematic from the destination library.

> To create a pin on a symbol using the Generate Part utility, the pin must have a pin to port mapping in the pin file.

To generate a part from a PSpice model library

1. From the Tools menu, choose Generate Part to open the Generate Part dialog box.
2. In the Netlist/source file text field, enter the path and name of the file you want to use to generate the part, or click the Browse button to browse for it. Be sure to change the type of files in the Browse File dialog box to PSpice Model Library Files (*.cir, *.net, *.lib).
3. Select the Implementation name from the drop-down list box in the Implementation options. If more than one model is available, you can choose one from the drop-down list, or you can keep the default setting of <ALL> that appears in the Implementation name and the Part name text boxes. Choosing <ALL> will create a part from all available models.
4. In the Destination part library text field, you can choose to accept the default path and file name, enter the path and name of the file you want to use for the new library, or click the Browse button to search for another one.
5. Click OK.
   The Microsim Schematics to Capture Translator translates a single model or all of the models in the library for reuse in Capture.

Generating a part from a schematic or library

You can use the Generate Part dialog box to create a part that has pins matching all the hierarchical ports on the root schematic of a design and use the design as a[reuse module.|Design reuse models#1271334]
If a destination library file doesn't exist, Capture creates a library part file (.OLB) that contains the newly generated part and a copy of the top-level schematic (if the Copy schematic to library option is selected). The new .OLB appears in the project manager Outputs directory along with the new part and schematic. The new part has a reference back to the design where the source schematic resides so that when you descend the hierarchy from the placed part, it will open the source schematic from the destination library.

To generate a part from a schematic design for reuse

1. In the project manager, click to select a schematic folder from which you want to create a part. The schematic can be in a design (.DSN) file or a library (.OLB) file.
   Note: The design file you use for the source must contain ports or the Generate Part operation will fail.
2. From the Tools menu, choose Generate Part. Capture fills in default settings in the Generate Part dialog box.
3. Browse to change the Netlist/source file or keep the default settings. You can generate a part from an external design or from the current design in the active project manager window.
4. Select the Netlist/source file type. For example, choose Capture/Schematic Design if you are generating a design reuse module.
5. Keep the default Part name or enter another name from the design.
   - If the source design has more than one schematic and no schematic is selected, or if more than one schematic is selected, the Part name setting defaults to the root schematic.
   - If the source design is not opened in the current project manager window, its root schematic will be the default Part name.
6. Browse to change the Destination part library or keep the default settings. The default setting creates a new .OLB file that matches the name of the source .DSN file. If the source design is a library (.OLB), the default part library name will be the same name as the source file.
7. Set Primitive property of the new part using the Primitive radio buttons. The default value is No. You can descend from the placed part to its schematic if the Primitive is set to No. For a design reuse module check Default.
8. Select the Copy schematic to library check box if you want to include a copy of the schematic in the new .OLB file along with the new part. This option is unavailable if the source and destination files are the same.

> If you are generating a design reuse module, do not check the Copy schematic to library option. If you do, you will lose occurrence properties that are critical in the design reuse module. Also, checking this option will overwrite the custom library, so make sure to specify an used library name unless you intend to overwrite.

9. Click OK.

> If the source schematic is copied and becomes locally available in the library, the implementation path is the same as the destination library. If the source schematic is not copied, the implementation path is the same as the source design or library.

To reuse an existing hierarchical design

1. Create a new project or open a working design.
2. In the project manager from the Tools menu, choose Generate Part.
3. Use the Browse button to pick a source design file that has a reusable design and is useful in the working design.
4. Select the schematic to reuse.
5. Enter a resulting library name or keep the default.
6. Select the Copy Schematic to Library Check box and click OK to create a part.
7. Select the schematic page editor window to place instances of the part to reuse its schematic design.
8. Click OK.

Creating a split part

Capture allows you to split a part using the Split Part Section Input Spreadsheet.

To split a part

1. Select the part that you want to split from the Library folder in the project manager.
   Note: You need to select a single-sectioned part from a library. You can split a multi-sectioned part only when it has already been split using the Split Part Section Input Spreadsheet.
2. Select the Tools menu and choose the Split Part command.
   OR
   Right-click on the selection and choose Split Part from the pop-up menu.
   The Split Part Section Input Spreadsheet appears displaying all the pins and their corresponding properties for the selected part.

> The Part Name and Part Ref Prefix fields display the name of the selected part and its part reference. These fields are view-only.

3. Select Numeric or Alphabetic in the Part Numbering group. If you select Alphabetic, an alphabet (between A to Z) will be added as a suffix to the current part reference for each of the split parts. If you select Numeric, a number (between 1 and 1024) will be added as a suffix to the current part reference for each of the split parts.
   The Section property column changes based on your selection in the Part Numbering group. For example, if Alphabetic is selected, the Section property column displays “A”.
   Resize the Split Part window by dragging the borders as per your requirement.
4. Specify the number of sections you want to have in the split part in the No. of Sections text box. The Section property column changes to a list box displaying the number of sections you specified in the No. of Sections text box.

> Number of sections cannot be less than one.

> If you select alphabetic numbering, then you can create up to a maximum of 26 sections only. If you select numeric numbering, then you can create up to a maximum of 1024 sections.

5. Click on a Section cell for a specific pin and select a section number from the list box.
   OR
   Click on a Section cell and enter the new section number.
   The selected Section cell displays the new section number.

> You can select Section cells for multiple pins simultaneously using the Shift+Down Arrow keys and enter the section number.
>
> OR
>
> Select the Section cells for multiple pins simultaneously using the Shift+Click, then press the Ctrl key, and then select a section number of your choice from the list box. The selected Section cells get populated with the section number of your choice.
>
> Click the first cell of the range, and then drag to the last cell, and then enter the section number of your choice. The selected Section cells get populated with the section number of your choice.

> You can select alternate Section cells for multiple pins simultaneously using the Ctrl+Click and enter the section number.

To sort on any property, double-click its name in the column header.

The pins will be added in the same order in the split part as the order in which the pins appear in the spreadsheet.

6. Click the Save button to save the changes to the current part. If any warnings are generated during the save operation, a message box appears asking you whether you want to view the warnings.
   To view the warnings, click the View Warnings button.
   The Split Part Section Input Spreadsheet expands and displays a grid showing warnings messages.
   If you select the Continue button, the split part is saved as is.

   - Click the Hide Warnings button to hide the warning messages or the Show Warnings button to display the warning messages again.
   - Use Save As to retain the original part and save the changed part as a new part in the same library.

You can hide or show a property column in the Split Part Section Input Spreadsheet. To do this, right-click the property column header you want to hide and select Hide from the pop-up menu. The selected property column will not appear now. To show a property column, right-click the property column header next on the right-hand side of the hidden property column and select Unhide from the pop-up menu. The hidden property column appears in the Split Part Section Input Spreadsheet. Alternatively, you can show a property column by:

- Double-clicking the column handle () of the property column header.
- Dragging the column handle of the property column header.
- (only the last two methods can be used to show a property column, which is the last column in the Split Part Section Input Spreadsheet).

If the number of Sections specified in the Split Part Section Input Spreadsheet is greater than one, then the resulting part will be of heterogeneous package type.

You can change the order in which the property columns and rows appear in the Split Part Section Input Spreadsheet. To do this, select the property column/row header you want to move and drag and drop it to the location where you want it in the Split Part Section Input Spreadsheet.

You can use standard copy and paste features to copy all the data from the Split Part Section Input Spreadsheet to MS Excel. You can later use the MS Excel file for archiving or documentation. It is recommended that you avoid using MS Excel to paste information into the Split Part Section

Input Spreadsheet.

Adding pins to Split Part Section Input Spreadsheet

You can add more pins to the Split Part Section Input Spreadsheet. The pins are added at the end of the current set of pins. The pins numbers for the added pins is generated automatically.

To add a pin

1. From the Split Part Section Input Spreadsheet, Click the Add Pins button. The Add Pins dialog box appears.
2. Specify the number of pins you want to add in the Number of Pins text box.
3. Click OK. The required pins are added at the end of the current pin set in the Split Part Section Input Spreadsheet.

> All the pins added are populated with default property values. The default value for the Section property column is one.

Deleting pins from Split Part Section Input Spreadsheet

You can delete pins along with their corresponding properties in the Split Part Section Input Spreadsheet.

To delete a pin

1. Select a row in the Split Part Section Input Spreadsheet.
2. Click the Delete Pins button. A message box appears asking you to confirm the deletion.
3. Click OK to confirm deletion. The selected row containing the pin information is deleted from the Split Part Section Input Spreadsheet.

> Once you delete a pin from the Split Part Section Input Spreadsheet, you cannot retrieve it later.

Viewing split part properties

When you split a part, Capture assigns the following properties to it:

- `SWAP_INFO`—This property is assigned to each section and its value is determined by the number of sections you specified in the No. of Sections text field in the Split Part Input Spreadsheet. For example, if you split a part into 3 sections then all the 3 sections will be assigned the `SWAP_INFO` property with value (S1+S2+S3).
- `SPLIT_INST=TRUE`

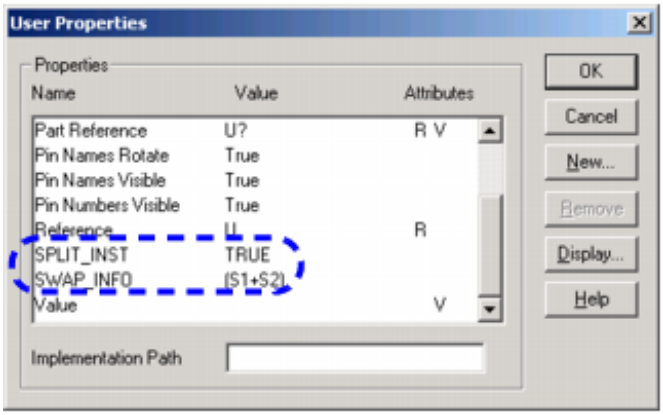
Saving a multi-section split part will remove any existing SPLIT property. Instead, it will add the SPLIT_INST property.

Saving a multi-section split part will overwrite any existing SWAP_INFO property value with the value (S1+S2+S3+...Sn), where n is the number of sections specified.

Ensure that you use '(' and ')' brackets for defining single sections. For example, If you have a split part having 6 sections (S1 to S6) and wish to add the SWAP_INFO property such that swapping happens only between sections S5 and S6, you need to add the SWAP_INFO value as (S1), (S2), (S3), (S4), (S5+S6).

To view split part properties

1. Double-click a split part from the Library folder in the Project manager. The Part editor window displays.
2. Select the Options menu and choose Part Properties. The User Properties dialog box is dispalyed. The following figure shows a split part with the two properties assigned to it.



The SWAP_INFO and SPLIT_INST properties appear in the User Properties dialog box for only those parts that are heterogeneous.

Viewing split part package and its properties

A split part is a multi-sectioned package. You can view a split part in package view.

To view split part package

- From the part editor, select the View menu and choose the Package command. An image of all the sections in the package appears.
  OR
- You can also double-click a split part from the Library folder in the project manager to open

the part editor window.

> The part reference for each of the sections contain a suffix entry. For example, J?A, J?B, and J?C, where A, B, C are the section numbers.

To view split part package properties

1. From the part editor, select the View menu and choose the Package command.
2. Select the Options menu and choose the Package Properties command.
   The Edit Part Properties dialog box displays.

The Multi-Part Package group displays the properties for your split part package, like parts per package, package type, and part numbering.

Deleting a Part

To delete a part

1. Open the library containing the part to delete.
2. In the Project manager, select the part.
3. Right-click on the part and choose Delete.
4. To save the changes to the library, choose Save from the File menu.

9

Working with Properties

Capture uses properties to describe objects. Imagine a brown, ceramic capacitor that measures 6 millimeters in height. Type, color, and height are properties, while ceramic, brown, and 6 millimeters are property values. Every Capture object is made up of such name-and-value pairs.

> Capture Release property names are not case sensitive. If you have a legacy design that contains two properties with the same name, Capture will append _# to one of two properties, where # is the counting number that makes the name unique on the object.

Inherent and user-defined properties

Some properties, called inherent properties, are an essential part of an object; others, called user-defined properties, are not used by Capture, but may be used by another tool. For example, if you want to include the supplier and the price per hundred for the objects on your schematic pages, you create two user-defined properties for the objects. You can add as many user-defined properties to objects as you like, and you can remove user-defined properties when you find that you don't need them. Graphic objects such as lines, ellipses, and rectangles do not support user-defined properties.

Creating and adding properties

When you create a user-defined property with the Edit Part Properties dialog box, you can make the property name visible and specify the font and location of the property value text, even before you specify the property value. The property name acts as a placeholder until you supply the property value.
When you add a user property to an existing object using the property editor, you can assign the property value at the time you create the new property or later. The additional benefit to using the property editor is the flexibility with which you can edit all properties on an object or group of objects on a schematic page.
Once you've added properties to a part on a schematic page, its properties no longer match the properties of the same part residing in the library. This part on the schematic page is unique, in that it has properties assigned specifically to it that are not inherited from the library part definition.
The property editor window shows you all available properties in the new single view. You can use the tabs to edit properties of all selected objects from the property editor. The property editor also displays all library definitions, instance properties, and occurrence properties for an object.

Properties in the design cache

When a part is first placed on a schematic page, a copy of the part and its library properties is put into the design cache from the library. A few of the library properties, such as PCB Footprint and

Value, are also copied as instance properties onto the placed part. The rest of the library properties “shine through” from the cache to the instance and occurrence of the part property. “Shine through” is indicated by hash marks in the cell. In the property editor, you can assign an instance or occurrence value, creating an instance or occurrence property. This instance or occurrence property then will override the shine-through definition.

Instance property

An instance property is a user property applied to the placed instance of a part or symbol in the design. This includes PCB Footprint, Value, and Name properties of each placed part or symbol in a design. This is the same as the user properties displayed and editable from the Capture Logical view.
An instance property will "shine through" to all occurrences of that instance unless it is overridden by occurrence properties that you have edited. A change using any of the tools, like Annotate, also may update the instance property.

Occurrence property

An occurrence property is a user property applied to a particular occurrence of a placed instance of a part or symbol in a design. This is the same as the user properties displayed and editable from the Capture v7.2 Physical view.
The spreadsheet will expand to display occurrence properties if values are different from the instance shine through value; otherwise, the rows are hidden from view. To quickly hide or display all the occurrence properties, press and hold the CTRL key while clicking on one of the plus (+) symbols in the property editor.

> ⚠ If you are working on multiple occurrences of a block, Capture provides a Save message only while closing the last occurrence. It is recommended you work on only one occurrence at a time.

Push Occurrence Properties into Instance Utility

Suppose that you copied a circuit or part of a circuit from design A and pasted it in design B. You might see occurrence and instance level properties with different values on the pasted parts in design B. In previous releases of Capture, you had to invoke the property editor on each part, copy and paste the occurrence property values of the Part Reference, PCB Footprint property and any other property as instance property values, and remove occurrence properties.
The Push Occurrence Properties into Instance Utility allows you to automatically do this. It automatically:

- transfers occurrence property values of the part reference and PCB footprint properties as instance level property values
    - removes all occurrence properties from the design and sets the preferred mode of the design to instance (if you select the Remove occurrence level properties check box).

- transfer occurrence property values of flat nets to schematic nets.
  To run this utility, from the Accessories menu in Capture, choose *Push Occ. Prop into Instance*, then choose *Transfer Occ. Prop. to Instance*.
  See Push Occ. properties to instance dialog box for more information.

Combined property strings

With many of the tools in Capture, such as Create Netlist and Annotate, you use combined property strings to convey information to the tool or to limit the tool's action.
A combined property string consists of one or more property names, enclosed in braces, and can also contain literal text. Capture combines the values of the named properties with any literal text to create a string. An example is:

```
{Value}{Reference}
```

where "Value" and "Reference" are property names. Using this combined property string and a part with a part value of 74LS32 and a part reference of U?A, Capture creates the string:

```
74LS32U?A
```

You can include spaces and other characters in the combined property string, as in this example:

```
Part: {Value} ({Reference})
```

Using this combined property string and the same part, Capture creates the string:

```
Part: 74LS32 (U?A)
```

Different tools use combined property strings in different ways. For example, Annotate uses one to compare parts---if one part's combined property string matches another part's combined property string, it packages the parts together.

Bill of Materials and other commands on the Tools menu that generate an output file based on a combined property string may produce errors if you include extra curly braces.

You can include tabs in combined property strings, so that the output file can be manipulated in a spreadsheet or database application. Tabs also help format report files, such as those created by the Bill of Materials command.

Wherever you want to have a tab in the output file, insert the characters \t (a backslash and a lowercase "t") in the combined property string.

Certain properties can be edited, but not removed. These are called inherent properties, and are listed in the table below.

Object type	Properties
Arcs	Line style and width, color
Bookmarks	Name
Images (pictures)	(None)
Bus entries	ID, net name
Buses	(None)
DRC markers	(None)
Ellipses	Fill style, line style and width, color
Ground symbols	Name
Hierarchical pins	Name, pin type, pin width
Hierarchical ports	Name, pin type
Hierarchical blocks	Color, implementation path, implementation type, implementation, name, primitive, part reference, value
IEEE symbols	(None)
Junctions	(None)
Lines	Line style and width, color
Net aliases	Alias name, color, rotation, font

No connect objects	(None)
Off-page connectors	Name
*Pictures (*images)	(None)
Pins in part editor	Name, number, width, shape, type
Pins in schematic page editor	Is no connect, name, net name, number, order, swap ID, type
Polygons	Fill style, line style and width, color
Polylines	Line style and width, color
Power symbols	Name
Rectangles	Fill style, line style and width, color
Title blocks	Design create time, design file name, design modify time, design name, page create date, page modify date, page size, schematic create date, schematic modify time, schematic name, schematic page count, schematic page number, source library, source symbol
Text	Text content, color, rotation, font
Visible properties	Value (of most properties), visibility, color, font, rotation
Wires	ID, net name

In this section:

Defining properties

All objects are described by properties to which you can assign values to suit your needs. In addition, you can add to the set of properties for the object types listed below.

- Parts
- Hierarchical blocks
- Pins on library parts
- Buses
- Wires
  For nets, you actually select a wire segment and add a property, but the property exists on the net rather than on the individual wire segment.

> You cannot add properties to graphic objects, bookmarks, IEEE symbols, no-connect objects, net aliases, power and ground symbols, off-page connectors, hierarchical ports, or bus entries.

You can add a property and specify the color, visibility, and font of the property text without assigning a value. The property name, which serves as a placeholder, appears next to the object and is enclosed in braces.

To add a user-defined property

1. Select an object on a schematic page.
2. Right-click on the object and choose Edit Properties from the pop-up menu.
   The Property Editor for the object displays.
   Note: The Property Editor displays a number of properties of the object. These are the system-defined properties for the object.
3. To add a user-defined property, click Add Row (or Add Column) button on the top left corner of the Property Editor window.
4. In the Add New Row (or Add New Column) dialog, enter a name and value for property.
   Note: A property value is not mandatory. This means that you can create a property with only a name.
5. To save this property and add a new property click Apply.
   OR
   Click OK to save this property and close the dialog.
   The new property is added in the Property Editor window.
6. To change the display properties of a property, right-click on the property Row (or Column) and choose Display.
   The Display Properties dialog displays.
   In this dialog, you can choose the display options:
   a. Not to display the property on the schematic page

b. Display only the value or only the name
c. Display both value and name

You can also choose to display font, font color and orientation. However, these options are unavailable if choose not to display the property.

7. Click OK.

Editing properties

All objects in Capture have attributes, or properties. Examples include the type of object (such as text, wire, or title block), part reference, color, font, and visibility. Some properties, called inherent properties, are essential to Capture—you cannot remove these inherent properties, though you can change the values of some of them. Other properties, called user-defined properties, may be used by external tools—they are not used by Capture, so you can add and modify these user-defined properties to suit your needs.
For information on editing the properties of a set of objects, see aUsing the spreadsheet editor. If you want to edit the properties of a part, see Assigning properties to the part.

> Except for occurrence properties, the schematics of externally-referenced libraries and designs should not be edited. You should view them as read-only designs. Trying to edit, then save, these designs from within your schematic can introduce errors such as duplicate reference designators and other problems.

When saving schematics with externally-referenced libraries or designs, occurrence properties are saved but altered instance values are not. If you want to change externally-referenced libraries or designs you should first close the referencing design. Then, open the referenced library or design, make the necessary changes, and save and close the referenced library or design. At this point, you can reopen the original design and reference the modified design.

To edit properties associated with a part, a wire, or a pin

1. In the schematic page editor, select the object.
2. From the Edit menu, choose Properties. The property editor appears.
3. Change the properties.
4. Click Apply, and close the property editor.

To edit the properties of a pin

1. In the Part Editor
    a. Select the pin.
    b. From the Edit menu, choose Properties. The Pin Properties dialog box appears.
    c. Change the properties.
    d. Click OK to close the Pin Properties dialog box.
2. In the Schematic page Editor

a. Select the pin
b. Right-click on the pin and choose Edit Properties.
c. In the Property Editor window, edit the properties of the pin.
d. Click the Apply button in the top left corner of the Property Editor window and close the window.

> Property changes made from the schematic page editor are limited to FLOAT and no-connect properties.

To edit the properties of multiple pins

1. In the part editor, hold the Ctrl key while you click to select each pin.
2. From the Edit menu, choose Properties command. The Browse spreadsheet appears.
3. Change the properties.
4. Click OK to close the Browse spreadsheet.

To edit properties associated with comment text

1. Select the text to edit.
2. From the Edit menu, choose Properties.
   The Display Properties dialog box appears.
3. Make the changes to the text, or its font, rotation, color, and visibility, then click OK.

To edit the name of a property

1. Double-click on the object. The property editor appears.
2. Select the property, edit its name in the Name property cell.
3. Click Apply, and close the property editor.

To delete a property

1. Select the object.
2. Right-click on the object and choose Edit Property.
   The property editor appears.
3. Select the property and click the Delete Property button.
   Only user-defined properties can be deleted.
4. Click Apply, and close the property editor.

> To delete an occurrence property from a flat net, replace the property value with <null> in the property file. When you import the file, the property will be deleted. You cannot export instance properties using this method.

Importing part and pin properties

After you create a property file using the Export Properties command, you can use a spreadsheet, database, or word processing application to edit property values or to add or delete properties. See Assigning properties to the part for important information about the format and contents of a property file.

Capture does not import all part and pin properties. So if a property is exported using the Export Properties command, it does not imply that you can change it's value and then import the change back into the part or pin. The thumb rule is that any property that is editable in the Property Editor in Capture and is exported can be imported. For example, the pin type property of a pin is exported by the Export command. However, if you EDT the value in the export file and import the file back into Capture, the pin type does not change.

When you import the edited properties, Capture expects to find the schematic pages and parts unchanged. After you export properties, do not edit the project or library from which the properties were exported until after you import the changed properties. If you do, the Import Properties command will fail, and you will have to export and edit the properties again.

It is a good idea to update (annotate) part references before you export properties.

Because various popular spreadsheet and database applications behave differently, Capture can import properties with or without enclosing quotation marks around each field in the property file. The fields must be tab-delimited, though—all other characters, including commas and leading and trailing spaces, are treated as part of a field's text. Be sure your spreadsheet or database program can save in this format.

> If the import file was created using a version prior to 16.3, it might consist of blank properties resulting in error during the import process. Edit any instances of blank properties before importing the file.

You can change reference designators by editing the Part References column of the property file.

If Capture finds errors in the property file, the project or library remains unchanged. There is no risk that some parts will be changed and others not.

The Import Properties command allows you to update the pin-type, pin-numbers and user-defined pin properties of the part.

EXP files exported from version 16.2 cannot be imported in previous versions of Capture. However, EXP files exported from previous versions will import correctly in 16.2.

To import part properties or part and pin properties

1. Open the project with the design or library holding the parts.
2. From the Tools menu, choose the Import Properties command. The Import Properties dialog

box appears.

3. Select the property file. If the property file is not listed, do one or more of the following:
4. In the Look in drop-down list box, select a new drive, a new directory, or both.
5. In the Files of type box, select the type of file you wish to open.
6. Click OK to apply the properties.

> If you edit a library provided by OrCAD, it is important that you assign a new library name so that your changes are not overwritten when you upgrade or update your software.

> While importing or exporting a design, make sure that the property values do not contain a quote ‘ " ' symbol.

Exporting part and pin properties

The Export Properties and Import Properties commands provide a means to edit properties of parts and pins in a spreadsheet or database program, or in a text editor that preserves tab characters. You first export the properties to a property file, edit the property file in the application of your choice, then import the edited properties. See Editing Property Files below for important information about the format and contents of a property file.

When you import the edited properties, Capture expects to find the schematic pages and parts unchanged. After you export properties, do not edit the project or library from which the properties were exported until after you import the changed properties. If you do, the Import Properties command will fail, and you will have to export and edit the properties again.

It is a good idea to update (annotate) part references before you export properties.

Because various popular spreadsheet and database applications behave differently, Capture can import properties with or without enclosing quotation marks around each field in the property file. The fields must be tab-delimited, though—all other characters, including commas and leading and trailing spaces, are treated as part of a field's text. Be sure your spreadsheet or database program can save in this format.

You can change part references by editing the References column of the property file.

The Export Properties command outputs the device information for the part. So if a homogenous part has 2 sections, it will output information corresponding to both the sections. These sections can recognized in the EXP file through designator prefix.

Pin-numbers will be output for each device/section of the part.

For a design, you can export all parts or just the parts in selected schematic folders and schematic

pages. For a library, you can export all parts or just selected parts, but you cannot export part aliases. If you export and edit properties of a part that has aliases, the aliases reflect the changes. You cannot select parts in the design cache or library cache for export.
You can add comments to document a property file; any text to the right of a semicolon (;) is ignored by the Import Properties tool.

Editing property files

When you export properties, Capture creates a tab-delimited list of keywords, identifiers, and properties, each of which is enclosed in double quotation marks. You can edit this file in a spreadsheet or database program, or even in a text editor (as long as the editor doesn't convert the tabs to spaces). Depending on which tool you use, you may see the property file as rows and columns of cells or fields or as lines of text.
The property file starts with a line identifying the document as either a design or library. Most subsequent lines begin with a keyword and an identifier.

> Do not make changes in the ID, Net Name, and Net ID fields in the property file (*.exp). The changes will not reflect in your design when you use the Import Properties command.

Making certain changes to the property file will cause the column headers and fields to be out of sync and invalidate your design You must not:

- change or delete the first line.
- delete the first field in any line.
- delete a field from a HEADER line without also deleting the corresponding fields from subsequent lines.

You can make these changes...	**with these results.**
Add a field to a HEADER line and subsequent lines (add a column).	This adds a property and pins with a value in this field. The name of the property is the string in the HEADER line. The value assigned to the part or pin is the string in the corresponding field. If the corresponding field is empty, Capture adds a property with no value and displays the property name as a placeholder.
Change a property value to <null>.	This deletes any existing property.
Delete a field from a	This has no effect on any part or pin. Deleting columns for properties you don't want to change may make the property file easier to edit. If you delete a

HEADER line and subsequent lines (delete a column).	field from a HEADER line without also deleting the corresponding fields from subsequent lines, Capture reports an error when you import the property and does not process any changes.
Change the value of a field.	Resets the value of the property on the object to which it refers.

The following table illustrates what happens when a part or pin has a property or does not have a property when a field is in various conditions:

Condition	Part or pin has the property	Part or pin does not have the property
Field is not <null>	Property value changes to specified value.	Property is added with specified value.
Field is <null>	Removes existing property.	Object is not affected.
Field is empty	Capture shows {Property Name} as place holder (when the property is visible).	Capture shows {Property Name} as place holder when the property as visible).

Capture property files contain the following keywords:

DESIGN	Identifies the property file as describing a project; specifies the path and filename of the project; identifies the mode active when the properties were exported.
LIBRARY	Identifies the property file as describing a library; specifies the path and filename of the library.
HEADER	Lists all the properties for the parts and pins in the design.
PAGE	Identifies the current schematic page.
PART	Identifies the current part and lists its properties. A column without a value signifies that the property does not apply to the part, or that no value is assigned.
PIN	Identifies a pin on the current part and lists its properties. A column without a value represents either a missing property or a property with an empty string for the value.

	A column without a value signifies that the property does not apply to the pin, or that no value is assigned.
SYMBOL	Identifies a symbol and lists its properties. A column without a value signifies that the property does not apply to the symbol, or that no value is assigned.

Note: It is a good idea to update (annotate) part references before you export properties.

Because various popular spreadsheet and database applications behave differently, Capture can import properties with or without enclosing quotation marks around each field in the property file. The fields must be tab-delimited, though—all other characters, including commas and leading and trailing spaces, are treated as part of a field's text. Be sure your spreadsheet or database program can save in this format.

To export part properties or part and pin properties

1. Open the project with the design or library holding the parts.
2. For a design, select the schematic folders or schematic pages containing the part you want to export.
   OR
   For a library, select the parts to export.
3. From the Tools menu, choose Export Properties command. The Export Properties dialog box appears.
4. In the dialog box, specify whether the property file is to include the entire design or a selected portion, whether you want to export properties for pins as well as parts or flat nets, and whether you want to export instance or occurrence properties.
5. Before you click OK in the dialog box, note the location of the export file in the Export File text box.
6. Click OK to create the property file.

If you edit a library provided by OrCAD, it is important that you assign a new library name so that your changes are not overwritten when you upgrade or update your software.

While importing or exporting a design, make sure that the property values do not contain a quote ' " ' symbol.

For projects

Capture reports a part once for each place it is used in the design. One HEADER line applies to the entire project.

> This abbreviated sample of a property file is formatted for presentation. As plain text, the columns do not actually line up as shown. In a spreadsheet or database program, fields may wrap or appear to be truncated.

```
"DESIGN" "C:\CAPTURE\SAMPLES\4BIT.DSN" "PHYSICAL"
"HEADER" "ID" "Part Reference" "Value"
"PART" "32" "fulladd_1" "FULLADD"
"PIN" "0{X}"
"PIN" "1{Y}"
"PIN" "2{SUM}"
"PIN" "3{CARRY_IN}"
"PIN" "4{CARRY_OUT}"
"PART" "152" "fulladd_4" "FULLADD"
"PIN" "0{X}"
"PIN" "1{Y}"
"PIN" "2{CARRY_OUT}"
"PIN" "3{CARRY_IN}"
"PIN" "4{SUM}"
"PART" "272" "fulladd_3" "FULLADD"
"PIN" ...
 :
"PART" "392" "fulladd_2" "FULLADD"
"PIN" ...
 :
"PART" "54" "halfadd_B" "HALFADD"
"PIN" ...
 :
"PART" "103" "halfadd_A" "HALFADD"
"PIN" ...
 :
"PART" "69" "U?" "74LS04"
"PIN" ...
 :
"PART" "74" "U?" "74LS08"
"PIN" ...
 :
"PART" "80" "U?" "74LS04"
"PIN" ...
 :
"PART" "85" "U?" "74LS32"
"PIN" ...
 :
"PART" "174" "halfadd_B" "HALFADD"
"PIN" ...
```

```
 :
"PART" "223" "halfadd_A" "HALFADD"
"PIN" ...
 :
"PART" "189" "U?" "74LS04"
"PIN" ...
 :
"PART" "194" "U?" "74LS08"
"PIN" ...
 :
"PART" "200" "U?" "74LS04"
"PIN" ...
 :
"PART" "294" "halfadd_B" "HALFADD"
"PIN" ...
 :
"PART" "343" "halfadd_A" "HALFADD"
"PIN" ...
 :
"PART" "309" "U?" "74LS04"
"PIN" ...
 :
"PART" "314" "U?" "74LS08"
"PIN" ...
 :
"PART" "414" "halfadd_B" "HALFADD"
"PIN" ...
 :
"PART" "463" "halfadd_A" "HALFADD"
"PIN" ...
 :
"PART" "429" "U?" "74LS04"
"PIN" ...
 :
```

For libraries

Capture reports each part in the library.

> This abbreviated sample of a property file is formatted for presentation. As plain text, the columns do not actually line up as shown. In a spreadsheet or database program, fields may wrap or appear to be truncated.

```
"LIBRARY" "C:\CAPTURE\SAMPLES\INTEL.OLB"
```

```
"HEADER" "ID" "Part Reference"
"PART" "80186.Normal"
"PIN" "X1"
"PIN" "X2"
"PIN" ...
 :
"PART" "80188.Normal"
"PIN" "X1"
"PIN" ...
 :
"PART" "80286.Normal"
"PIN" ...
 :
"PART" "80287.Normal"
"PIN" ...
 :
"PART" "8031.Normal"
"PIN" ...
 :
"PART" "8032.Normal"
"PIN" ...
 :
"PART" "80386.Normal"
"PIN" ..
 :
"PART" "80386SX.Normal"
"PIN" ...
 :
```

10

Managing Projects

In this section:

Archiving a Project

You can save the project (.OPJ) and all the related files (design, library, output files, and referenced projects) in a different directory and also create a zip file of this directory for archival purposes. You can also specify any additional files or directories that you may want to be archived along with your project files. For example, you can archive external designs, global PSpice model libraries, and global include files along with your PSpice project or data sheets for the parts. For more information, see To add additional files and directories to the archive.
You can use the Archive Project command on the File menu to archive your project. This command will allow you to save all the files related to your project in the directory you specify for archival and zip the directory into a single zip file, which will have a *.zip* extension. You can use the WinZip software to unzip zip archives created using Capture.

> The path names of the files and directories in the archive file are relative to the archive directory. This implies that when the archive file is unzipped on a different machine the files and directories retain the original directory structure of the archive directory.

To archive a project:

1. Make sure that the project you want to archive is active and the schematic pages for the project are not open.
2. From the *File* menu, choose *Archive Project*.
   The Archive Project dialog box displays.
3. Select the files you want to be archived with your project. If you do not select any of the options (Library files, Output files, or Referenced projects), Capture by default archives only your project (.OPJ) and design (.DSN) files.

For PSpice projects, the simulation profiles and local files (.LIB, .STL, .INC) will all be archived along with the project. The Output files option also archives simulation output files such as .DAT and .OUT for PSpice projects. The archiving methodology for PSpice model libraries is as follows:

- Profile-level model libraries are archived under their respective profiles and referenced as .\<*library_name*>*.lib*.
  For example, when a profile; AC containing a model library *diode.lib* is archived, the *diode.lib* is copied under the directory *AC* and the simulation settings is modified as: `.\diode.lib`.
- Design-level model libraries are archived under `.\<design_name-pspicefiles>\<design_name>\<library_name>.lib`.
  For example, when a design called *histo* containing a model library *bipolar.lib* is archived, the model library *bipolar.lib* is copied under directory `histo-pspicefiles\histo` and the simulation settings is modified as: `.\histo-pspicefiles\histo\bipolar.lib`.
- In case of global-level model libraries:
  - a copy of model library is created under the existing _<design_name>.lib (_if exists).
  - a new <*design_name*>*.lib* file is created and a copy of model library is added to the <*design_name*>*lib* and the simulation setting is modified as design-level library.

4. Click the **...**button to find the directory to which you will save your files.
   The Select Directory dialog box displays.

You can also enter the relative path of the archive directory in the Archive directory text box. This path is treated as relative to the project being archived.

5. Find and select the directory in which you want your project archived and click OK. If required, create the directory.
6. Click OK in the Archive Project dialog box.
   Capture archives your project with all the selected files to the specified directory and displays information/errors in the Session Log.

- The working directory does not change to the newly set archive directory.
- The settings you specify in the Archive Project dialog box get saved in the CAPTURE.INI file. These settings are used whenever you start the next archive session.

To create a zip archive for the project:

1. Make sure that the project you want to archive is active and the schematic pages for the project are not open.
2. From the *File* menu, choose *Archive Project*. The Archive Project dialog box appears.
3. Set the directory to which you want to save all project files as described in "To archive a project" [on page 464|Archiving#1217948].
4. Select the Create single archive file check box to activate the *File* name text box.
5. Specify a file name for the zip archive file in the File name text box.

   The default file name for the zip archive file is <projectname-current date>. The file extension (.zip) is automatically added to the zip archive file.

6. Click OK. Capture zips all the files in the specified directory and generates a zip file with .zip extension. The Session Log displays all the events that occur during the archiving process and report whether the process completed successfully or with errors.

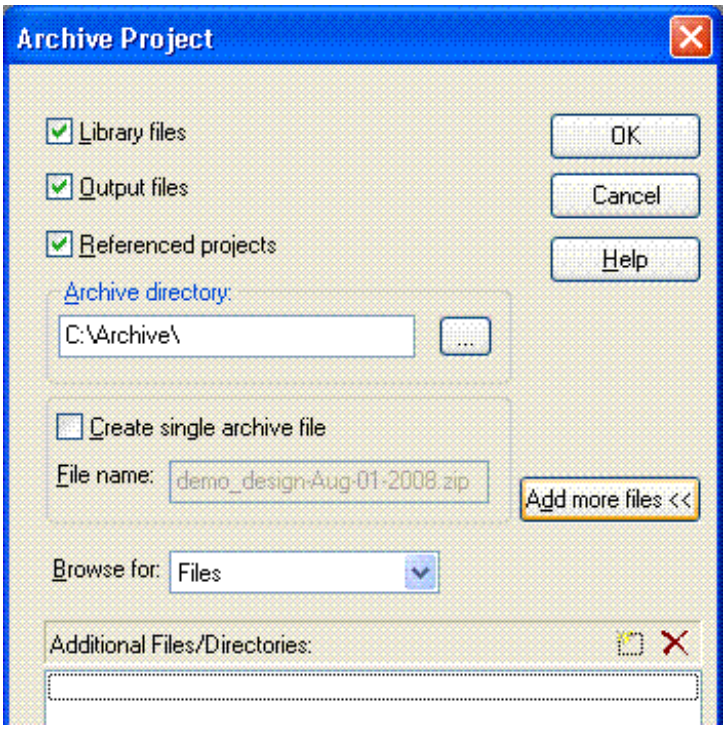
- The working directory does not change to the newly set archive directory.
- You can unzip the zip archive using the WinZip software.
- Archiving to a single file with a .zip extension does not compress the contents.

To add additional f iles and directories to the archive:

1. Make sure that the project you want to archive is active and the schematic pages for the project are not open.
2. From the File menu, choose Archive Project. The Archive Project dialog box appears.
3. Set the directory to which you want to save all project files as described in *To archive a project* section.
4. Click the Add more files >> button. The Archive Project dialog box expands and displays a grid where you can add more files and directories.

> Click the Add more files << button to revert the Archive Project dialog box to its default state.

5. Select an option from the Browse for list box. Select the Directories option to add directories or the Files option to add more files to your archive. The Files options is the default selection.
6. Click the button or press the Insert keyboard key. An edit box with a blinking cursor appears in the Additional Files/Directories list.



7. Click the **...** button to locate the files or directories you want to add in the archive. A file or directory selection dialog box appears depending on your selection in the Browse for list box. For example, if you selected the Directories option, then the Select Directory dialog box appears. Otherwise, the Select File (s) dialog box appears.

8. Find and select the file or directory to be added to your archive. The location path of the selected file or directory is added in the Additional Files/Directories list.

   Do not enter relative path for files or directories in the Additional Files/Directories list.
   You can use the standard CTRL or SHIFT keys to select multiple files in the Select File (s) dialog box.
   You can also select multiple files by dragging the left mouse button over the files you want to select in the Select File (s) dialog box. You cannot use this method to select multiple

directories in the Select Directory dialog box.
If you finished adding files and want to add directories now and vice-versa, you must select an appropriate option from the Browse for list box.
The Additional Files/Directories list displays information based on your selection in the Browse for list box, that is, if your selection is Directories then only the directories added to the list are displayed and vice-versa.
Use the  button or press the Delete key to delete the files or directories you do not want in the Additional Files/Directories list.
The archiving mechanism ensures that duplicate files or directories do not get added to the Additional Files/Directories list.

9. Click OK in the Archive Project dialog box. Capture archives your project with all the selected additional files and directories to the specified directory. The working directory does not change to the newly set archive directory. For information on how to create a zip archive file, see To create a zip archive for the project

- The files and directories you add using the Additional Files/Directories list are added to the archive directory under a separate sub-directory called *Additional files.*
- The archived project (.OPJ) file does not contain references to the additional files and directories added using the Additional Files/Directories list.
- The settings you specify in the Archive Project dialog box get saved in the CAPTURE.INI file. The settings are used whenever you start the next archive session except for the files and directories list in the Additional Files/Directories list.

Auto Recovery

The auto recovery feature of Capture is designed to protect you from losing work as a result of a system crash or power outage. Capture automatically saves the state of the open design and library files at the end of each interval set in the Miscellaneous tab of the Preferences dialog box. Capture removes auto-saved files from your system when a project is closed and when you exit Capture normally. The Preferences dialog box appears when you choose *Tools – Preferences*.

Auto recovery is not an automatic saving feature intended to replace the Save commands. If you intentionally exit Capture without first saving your changes, they will be lost. Auto recovered files are automatically deleted when you exit Capture normally.

Auto-Saving files

The FILES.ASL auto-recovery file

Capture maintains a file called FILES.ASL in the \WINDOWS\TEMP directory that has a list of all design and library files that have been opened. Capture updates FILES.ASL when a project is opened or closed. If your system goes down without exiting Capture, FILES.ASL has a list of design and library files that were open for editing when the system failure occurred.

Auto recovery works on a timer that you can set to specify how often (if at all) you want Capture to auto-save the currently open and modified design and library files.

The frequency can be from five minutes to every 120 minutes (2 hours).

When the interval is reached:

1. Capture examines all open design and library files.
2. If a file that is part of the project is open for editing, Capture examines it to see if it has been modified since the last auto-save.
3. If it has, it is saved to the \WINDOWS\TEMP directory.
4. If the project itself has been modified, or one of its files was auto-saved, the project itself is saved to the \WINDOWS\TEMP directory.
5. When this happens, the auto recovered version of the project file is updated such that the path of any auto-saved designs or libraries change to point to the \WINDOWS\TEMP directory.
6. Any paths of ".\" are changed to point to the directory the project was loaded from.
7. Capture can then open the project from the \WINDOWS\TEMP directory with all paths pointing to the correct location: the \WINDOWS\TEMP directory for those files that were auto-saved, and the original location for those files that were not modified and therefore not auto-saved.

When a project is closed normally, all auto recovered files in the \WINDOWS\TEMP directory for that project are deleted.

Restoring auto-saved files

- When you start Capture, it checks for the FILES.ASL file.
- If Capture finds the file, it reads it.
- If Capture exited abnormally the last time it was used, and there are design and library files listed in FILES.ASL, Capture opens the design and library files from their original locations.
- Capture then appends "(Restored)" to the project manager's title to indicate that the project was loaded as a result of a previous system failure.
- Capture also looks for design and library files that were auto recovered to the \WINDOWS\TEMP directory.

- If any are present, Capture opens them, changes their name to a default name (like PROJECT1), adds "(Restored)" to their project manager's titles, removes the file name and directory from the project manager window, and marks the project as modified.
- Capture then looks in the project for any files that reference the \WINDOWS\TEMP directory.
- Any files that do are auto recovered to the Windows temporary directory, are opened, their filenames and directories are removed from their project manager windows, and they are marked as modified.

Adding and Deleting Project files

Adding files to a Project

Typically, the files you add to your project will be files that have specific functions in the design process. For example, you might add a standard delay file to provide timing information for simulation with NC VHDL or some other simulation tool. However, you can add any files you want to a project, including documentation files (perhaps a functional specification) or waveform files (to show the results of a simulation).

To add a file to the project:

1. In project manager, select the folder to which you want to add a file.
2. Choose *Edit – Project*. The Add File to Project Folder dialog box displays.
3. Select the file you want to add and choose the Open button. The file is added to the project. Alternatively, drag-and-drop the file from the Windows Explorer into the folder in the project manager.

> You can also add files to your project interactively. When you create a design using the New command on the File menu, it is placed in the project manager Design Resources folder.

Deleting Files from a Project

You can delete files from your project just as you would delete files in Windows Explorer. That is, just select the file and press the Delete key.

To delete a file from the project:

1. In project manager, select the file you want to delete.
2. Press the Delete key. The file is removed from the project.

- You cannot delete schematic pages or schematic folders if those schematic pages (or schematic pages within those folders) are currently open in Capture. You must first close the schematic pages in question.
- Deleting schematic folders, schematic pages, parts and symbols is permanent. You cannot use the Undo command to bring back deleted items from the project manager.

Undoing and Repeating commands

If you make a mistake, you can use the Undo command. If you change your mind again, you can use the Redo command. Undo and Redo functionality is available in the schematic editor, the part editor and the property spreadsheet.

You can use Undo/Redo for:

- Object creation/deletion activities (for example, Cut, Copy, Paste, and Place commands).
- Object manipulation (for example, Move, Resize, Rotate, and Mirror commands)
- Property value modifications.

Capture creates an “undo/redo” cache as you perform commands in the schematic editor and the property editor. Note that when you perform commands in the schematic editor, the undo/redo cache for the property editor is cleared, and vice versa. Also, note that certain other operations, such as Synchronize Up/Down cause the undo/redo cache to be cleared.

Multiple undo/redo operations

You can undo/redo commands:

- Sequentially, exactly one command at a time.
- By setting label states. Label state enables you to tag the schematic at different stages of design. You can later use these tags to go to a particular stage of the design and then undo/redo a number of steps that were performed at that point in the design.

For example, suppose you had performed the following actions on a schematic page:

1. Place a part.
2. Label the schematic as stage1
3. Rotate the part by 90 degrees.
4. Wire one of the pins on the part to another part on the schematic page.
5. Place another part.
6. Label the schematic as stage2.

You could then use the Undo command sequentially to return the schematic page to its state at any point in this sequence. Alternately, suppose after step 5 you decide that instead of rotating the part by 90 degree, it would make better sense to rotate the part by 270 degrees, and change the wire-pin connections as well. In such a case, instead of undoing a number of steps, you can jump to the label state stage1. This will take you to the state of the schematic described in step 1. You can then do the modifications as required.

> Also, note that the part editor does not include an undo/redo cache. Therefore, in the part editor, you can only undo/redo a single command.

Using Undo/Redo for designs and schematic pages

You can use Undo/Redo independently for each schematic page in your design. That is, a separate cache of undo/redo data is maintained by Capture for each schematic page.

In complex hierarchical designs, there can be more than one occurrence of a particular schematic. When editing in this environment, objects and annotations are handled by separate mechanisms. For objects, the edits are reflected in all the pages open on different occurrences of that page. If there are two pages open on two different uses of a schematic and you move an occurrence on one page, the occurrence will also move in the second page. When the user performs an UNDO, the state of the objects is restored on all occurrence pages identically.
For annotations edits are reflected uniquely in the occurrence pages. The annotation displayed by a view is selected by filter. UNDO in this case will restore the annotation value only on the particular occurrence page

For example consider the case of a reused instance with two occurrences H1/U1 and H2/U1. This instance has different annotations for these two occurrences. When the page is open on H1/U1, the Reference Designator displays as U25; when the page is open on H2/U1, the Reference Designator displays as U72.The behavior of UNDO will be as shown in following sequence:

Command	**Undo/redo**	**Schematic state**
Open root schematic	disabled	Schematic displayed
Open occurrences H1/U1 and H2/U1	disabled	Occurrence pages displayed
Move a component in H1/U1	Undo Move	Component moved on both H1/U1 and H2/U1
Change reference designator on H1/U1 to from U25 to U50	Undo Text (Redo disabled)	H1/U1 (only) is changed.

Change reference designator on H2/U1 from U72 to U80	Undo Text (Redo disabled)	H2/U1 (only) is changed.
Undo	Undo Text (Redo enabled)	H2/U1 reference designator returned to U72 from U80.
Undo	Undo Move (Redo enabled)	H1/U1 reference designator returned to U25 from U50.
Undo	disabled	Component moved to its previous position on both H1/U1 and H2/U1.

Clearing the Undo/Redo cache

There are a number of operations that will clear the undo/redo cache (that is, these operations cannot be undone, nor can the schematic page be returned to a state that existed previous to the execution of these operations):

- Choosing Update Current or Update all after editing a part on a schematic page
- Replace Cache, Cleanup Cache, or Update Cache
- Edit Properties through the Browse/Find commands or through third-party tools
- Annotation, back-annotation, Update properties, Import Properties, or Cross Reference operations

To undo an action

- From the *Edit* menu, choose the Undo command.

> Deleting schematic folder, schematic pages, parts, and symbols is permanent. You cannot use the Undo command to bring back deleted items from the project manager.

Shortcut

Toolbar: 

To undo an Undo command

- From the Edit menu, choose the Redo command.

Shortcut

Toolbar: 

Graphical Operation (GOp) Locking

A schematic page often contains a large number of different types of objects like parts, pins, buses, wires. A user often needs to perform operations like adding new objects, changing object properties, moving, creating and deleting objects. All these operations require extensive user interaction with the Capture interface. Also, with the increasing complexity of designs, the number of objects on a page and pages in a design has increased exponentially. All these issues raise the need in Capture for providing a methodology to lock the state of a design at a particular point of the design process. For example, a designer should have the ability to maybe lock the layout of a schematic page.

To address such issues, Capture includes a graphical operation locking (GOp) feature that allows you to lock objects (like components, pages, folders and even design) in a Capture project.

When you graphically lock an object, the graphical aspects of the object are locked. This implies that non-graphical aspects of an object such as its properties are still editable. For example, if you lock a part on a schematic page, you cannot delete, or move the part but you can change, say, the PCB footprint of the part.

GOp locking allows you to lock any object in a Capture design. You can lock the design, the schematic folders within the design, the pages within the schematic folders and the objects on the schematic pages.

In this section:

Locking and Unlocking Objects in a Design

You can lock (and subsequently unlock) any object in a Capture project. You can lock the objects

on a schematic page, the pages in a schematic folder, the folders in a design, and the design in a project.

To lock an object in Capture:

1. Select the object to lock.
   For schematic page objects, you can use the multi-select feature on the schematic page to select and lock multiple objects simultaneously.
   For project manager objects (pages, folders and designs), you select the objects in project manager.

   > ⚠ You cannot lock multiple Project manager objects simultaneously.

2. From the *Edit* menu choose the Lock menu item.
   Alternatively, you can right-click on the object (on the page or in the Project manager) and choose the Lock item in the pop-up menu. When you lock a schematic page object the look-and-feel of the object when it is selected is changed. When you lock a project manager object, a lock symbol appears over the icon of the object in the Project manager.

> ⚠
> - After locking (or unlocking) one or more objects on a design, the lock (or unlocked) state of the objects must be saved. For exmaple, say you lock one or more objects on a schematic page. After locking the objects, if you close the page without saving changes, the lock state of these objects is lost.
> - If you lock one or more objects in a design, export the design and then again import the design, all the locks on the imported design are lost.

To unlock an object in Capture:

1. Select the object to unlock.
   For schematic page objects, you can use the multi-select feature on the schematic page to select and unlock multiple objects simultaneously.
   For Project manager objects (pages, folders and designs), you select the objects in the Project manager. However, you cannot unlock multiple Project manager objects simultaneously.
2. From the Edit menu choose the UnLock menu item.

> ⚠ The lock and unlock menu items (in the Edit menu or the pop-up menu) are disabled or enabled depending on the lock state of the selected object (or objects).

Features of a Locked Schematic Page object

- The object cannot be deleted or cut.
- The object cannot be moved to another part of the page (using a cut-and-paste operation or a mouse drag operation).
- The object cannot be moved to another page (using a cut-and-paste operation or a mouse drag-and-drop operation).
- The object can be copied to another page or as another instance on the current page. However, the copied instance of the object is locked as soon as you paste it on the page.
- If the locked object is a part, the part editor for the object is inaccessible. This means that the menu option to open the part editor for a locked part is unavailable.
- The replace or update cache operations will fail if they effect a locked part. Say a design contains multiple instances of a part where some instances are locked and some are unlocked. In this case, the replace or update cache operations on a part with multiple instances will fail if these operations effect the locked instances. This means that these operations will not update even on the unlocked part instances.
- The Update All operation on an unlocked part instance (executed in the edit part procedure) will fail if this operation effects a locked part instance. This means that this operation will not effect even on the unlocked part instances.
  Since, this type of locking is graphical so you are still permitted to edit the properties of a locked part. This implies that you can open the property editor for a locked object and add, modify or delete properties on the part.

Features of a Locked Schematic Page

- The page cannot be deleted or cut.
- The page cannot be moved to another schematic folder (using a cut-and-paste operation or a mouse drag-and-drop operation).
- The page cannot be renamed.
- Schematic page objects cannot be added to the page. This implies that an object cannot be placed on a locked page using the Place command. Also, an object cannot be placed on a locked page by copying the object from another page and pasted it onto the locked page.
- The page can be copied to another folder. However, the copied page is locked as soon as you paste it to the destination folder.
- All the page objects are locked as soon as the page is locked. Also, objects on a locked page may be explicitly unlocked. For details on the cascading and roll-up effects of locking pages see the section Cascading and roll-up effects of Locking.

Features of a Locked Schematic Folder

- The folder cannot be deleted or cut.
- The folder cannot be moved to another design (using a cut-and-paste operation or a mouse drag-and-drop operation).

- The folder cannot be renamed.
- Schematic pages cannot be added to the folder.
- The make root property of a locked schematic folder cannot be modified.
- The folder can be copied to another design. However, the copied design is locked as soon as you paste it to the destination design.
- All the pages are locked as soon as the folder is locked. Also, pages in a locked folder may be explicitly unlocked. For details on the cascading and roll-up effects of locking folders see the section Cascading and roll-up effects of Locking.

Features of a Locked Design

- Schematic folders cannot be added to the design.
- All the folders are locked as soon as the design is locked. Also, folders in a locked design may be explicitly unlocked. For details on the cascading and roll-up effects of locking designs see the section Cascading and roll-up effects of Locking.
- Design operations, netlisting, annotations, DRC and permitted on a locked design. You can also simulate a locked design.
  Note: However, if you run these commands on a locked design, and this causes a graphical change in the design, Capture allows the change but it will immediately be locked onto the design.
  Say you run the DRC on a locked design (or the DRC effects locked objects in the design). If the design has any DRC errors or warnings, Capture allows the process to place the markers even on locked object of the design. However, if a marker is placed on a locked page, the marker is immediately locked and you will need to either unlock the page or the marker if you need to remove the marker.

Cascading and Roll-up effects of Locking

When you lock a container object (a page, a folder, or a design), all the objects within the container are also locked. Also, this process cascades down to the lowest level object.
So, if you lock a page, all the objects on the page are locked. If you lock a folder, all the pages contained in the folder are locked. In addition, the objects on each of the pages are locked.

> Unlocking has the same cascading effect on a container and the objects within the container.

When you lock a container object, you can unlock specific objects within the locked container by explicitly unlocking these. However, since locking and unlocking does not cause a roll-up effect, the unlock operation does not unlock the object container.

When you lock a container object, a lock symbol appears over the container icon in the Project manager. Now, if you explicitly unlock one or more objects within the locked container, the lock

symbol remains but it changes to an open lock. This indicates that the container is locked but one or more objects within the container are unlocked.

The locking operation on an object within a container is specific to the object. This implies that the lock (or unlock) operation on an object overrides the operation on the object container. Consider the example of a folder, SCHEMATIC1, containing two pages, PAGE1 and PAGE2.

1. Lock PAGE2.
2. Lock SCHEMATIC1.
   Since locking is a cascading operation, locking SCHEMATIC1 effects the lock status of its pages. In this case, since PAGE2 is assigned locked state, so the cascading operation will effect only PAGE1.
   The lock operation did not effect PAGE2 not because the page was already locked but because the lock (or unlock) state on an object overrides the locked (or unlocked) state of the container.
3. Unlock SCHEMATIC1.
   Again, due to the cascading effect of unlocking, the pages within SCHEMATIC1 are unlocked.
   However, since PAGE2 was locked specifically and not as part of the cascading lock on SCHEMATIC1, the cascading lock operation will not effect the lock state of PAGE1.

Locking (or unlocking) a container does not necessarily imply that the state of the entire contents of the container will be effected by operation. So objects within the container are assigned their own lock (or unlock) state.

11

Establishing connectivity

In this section:

Wire connectivity

In Capture, you can establish connectivity with wires or aliases.
Two perpendicular wires or buses are connected if:

- They form a "T" intersection, either by dragging an object or placing a wire.
- A junction is placed where they cross.

If they simply cross at 90 degrees, they are not electrically connected unless you place a junction at the intersection manually.
You can also use an alias to connect a signal from one area of your schematic page to another. For example, suppose you have placed a part on your schematic page and you want to connect it to another part at the opposite corner of the schematic page. Instead of drawing a wire from the first part to the second part, you can assign a single net alias to a wire connected to both parts.
You can connect two crossing nets, after they have been placed, by placing a junction where they cross. For more information about placing junctions see Placing junctions.
A net can have any number of aliases plus one optional net name. The only purpose of the net name is to give the highest priority to one of a net's aliases. When you assign a name to a net, you force Capture to resolve netname conflicts in favor of a particular alias.

As you place buses and wires:

- A bus and a wire can be connected only by name.
  If you begin or end a bus segment on a segment of a wire, a vertex is added to the wire and a junction appears, but the bus and wire are not electrically connected.
  If you begin or end a wire segment on a segment of a bus, a vertex is added to the bus and a junction appears, but the wire and bus are not electrically connected.
- Two buses or two wires can be connected physically.
  If you begin or end a bus segment on a segment of another bus, a vertex is added to the second bus, and a junction appears—the buses are connected.

If you begin or end a wire segment on a segment of another wire, a vertex is added to the second wire, and a junction appears—the wires are connected.

If you place parts so that two pins meet end to end, the pins are connected.
OrCAD recommends that you connect the pins of the parts using a wire, and avoid placing parts so that two pins meet end to end. This is because, parts with direct pin-to-pin connections produce a system generated net name to establish the connection and:

- Capture will not allow you to assign your own net name in the place of the system generated net name.
- Searching for the system generated net name can be difficult if you are not aware of the pin-to-pin connection.
- If you move the parts after creating the netlist, the system generated net name might change. This may cause net name conflicts when you run back-annotation.

Capture preserves the case of part names and net names, but ignores the case when comparing names for electrical connections. That means you may use upper-case or lower-case letters as you wish, but you need not remember the case.

In this section:

- Auto-Wiring in Capture

Placing and naming wires

The purpose of wires is to create connections. When you connect a wire to a pin, Capture provides visual confirmation: the unconnected pin box on the pin disappears. If two wires cross at 90 degrees, they are not electrically connected unless you create a junction by clicking on one wire as you draw the other to it, or by placing a junction over two crossing wires.
When a wire forms a "T" intersection with a pin or another wire, a visible junction is shown. If the two objects don't intersect, like when a wire ends at a pin or where the next wire begins, then a junction does not appear.
When you place a wire, it is assigned a system-generated net name. You can replace the system-generated name by assigning an alias or a net name. If you connect a wire to an existing net, the wire assumes the name of that net.

In this section:

To place a wire

1. From the Place menu, choose Wire.
   Or Press the W key on the keyboard.
2. Click on the schematic page to start the wire.
   You can press F6 to change the cursor to crosshair to start the wire from a specific location.
3. Use the mouse to draw the wire. Click to place a vertex and change directions. The vertex is constrained to multiples of 90 degrees.
4. If the wire ends at a pin or another wire, click to end the wire. The wire appears in the selection color.
   OR
5. Double-click to end the wire.
6. Select the selection tool to dismiss the wire tool.
   OR
7. Press the Esc key on the keyboard.

To place a non-orthogonal w ire

- Hold down the Shift key while you draw the wire. There is no constraint on vertex angles.

To attach a wire to a net

1. Begin or end the wire on the net.
   OR
   Click as you draw the wire over the net.
   OR
   Create a net alias, as described below, assigning the alias of the net to this wire. Within a schematic page, wires with the same name or alias are electrically connected.

As you place buses and wires:

- A bus and a wire can be connected only by name.
  - If you begin or end a bus segment on a segment of a wire, a vertex is added to the wire and a junction appears, but the bus and wire are not electrically connected.
  - If you begin or end a wire segment on a segment of a bus, a vertex is added to the bus and a junction appears, but the wire and bus are not electrically connected.
  - If you begin or end a wire segment on a segment of another wire, a vertex is added to the second wire, and a junction appears—the wires are connected.
- Two buses or two wires can be connected physically.
  - If you begin or end a bus segment on a segment of another bus, a vertex is added to the second bus, and a junction appears—the buses are connected.

> If you place parts so that two pins meet end to end, the pins are connected.

OrCAD recommends that you connect the pins of the parts using a wire, and avoid placing parts so that two pins meet end to end. This is because, parts with direct pin-to-pin connections produce a system-generated net name to establish the connection and:

- Capture will not allow you to assign your own net name in the place of the system-generated net name.
- Searching for the system generated net name can be difficult if you are not aware of the pin-to-pin connection.
- If you move the parts after creating the netlist, the system-generated net name might change. This may cause net name conflicts when you run back-annotation.

- OrCAD recommends that you do not connect a power symbol directly to a power pin. Connect the power symbol to the power pin using a wire.
- When you click on a wire segment, only that segment and its two handles are selected.
- Capture preserves the case of part names and net names, but ignores the case when comparing names for electrical connection. That means you may use upper-case or lower-case letters, and need not remember the case.

Shortcut

Tool palette: 

Placing junctions

You create junctions on wires while placing wires, or after the wires have been placed. If you are in the process of placing wires, clicking on another wire creates a junction for the two wires. Use the Place junction tool palette to place junctions on existing wires that cross each other, but do not connect.
Use junctions on buses the same way as on wires.

To create a junction while placing a wire

1. From the Place menu, choose Wire.
2. Click to start the wire. Click the left mouse button to change the direction of the wire as needed.
3. Move the pointer over the wire segment or wire segment vertex you want to connect with, and click. Capture creates a junction where the two wires meet, and ends the wire you are currently placing.
   OR
   If you are connecting with a wire or pin and continuing across it, double-click when you reach the intersection. Capture creates a junction at the intersection and continues the wire you are currently placing.

To place a junction on existing wires

1. From the Place menu, choose Junction.
   OR
   Choose the Place junction button on the Draw toolbar.
2. Move the pointer over two wires that cross, but do not connect.
3. Click.
   Capture places a junction where the two wires cross.

You can also specify the size of the junction dots to be placed on your schematic. The size is specified in the Miscellaneous tab of the Preferences dialog box. You can choose from Small, Medium, Large and Very Large sizes.

To select a junction

- Hold the s key and click to select one or more junctions.

To remove a junction

1. From the Place menu, choose Junction.
   OR
   Choose the Place junction button on the draw toolbar.
2. Move the pointer over the junction you want to remove.
3. Click. Capture removes the junction. The two wires no longer connect.
   OR
   Hold the s key while you move the pointer over the junction you want to remove, and click to select it.
4. From the Edit menu, choose Delete.
   OR
   Press the DELETE key. The two wires no longer connect.

- If the component is deleted, junction dots residing on the pin-stubs will also get automatically deleted in case that junction is not serving as a connection point to other wires/pins.
- When a wire is dragged, un-necessary junctions will not get created unless the drag results in a junction on wire break only.

Shortcut

Tool palette: 

In this section:

Junction Dot Formation

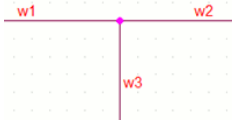
The creation of a junction dot when displaying connectivity in Capture uses the concept of **Junction on straight wire-break only**.

Junction on straight wire-break only

If a horizontal or vertical straight-wire is broken at a point between the end-points of the wire because of a connection with one or more connection objects (another wire, port, off-page connector, global, part pin or hierarchical pin), a junction dot is created at the point of connection.

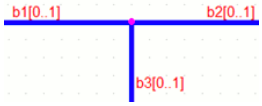
1. Consider a horizontal (or vertical) scalar wire joined with another scalar wire originating at the same connection point and drawn in the opposite direction. In this case, a junction dot is created if any of the following objects is placed at the point of connection of these two wires:
2. An orthogonal scalar wire
3. An off-page connector
4. A scalar schematic port
5. A scalar global.
6. A scalar pin
7. A scalar hierarchical port
8. A bus entry point. However, in the case of a bus entry point, the junction dot will not display.

Example: Say the wire w1 is joined with the wire w2, originating at the same connection point and drawn in the opposite direction. A junction dot is created if an orthogonal wire w3 is placed at the point of connection of these two wires.



1. As in the case of scalar wire, consider a horizontal (or vertical) bus or NetGroup joined with another bus or NetGroup originating at the same connection point and drawn in the opposite direction. A junction dot is created if any of the following objects is placed at the point of connection of the two objects:
2. An orthogonal bus or NetGroup
3. A bus or NetGroup off-page connector
4. A bus or NetGroup schematic port
5. A bus or NetGroup global
6. A bus or NetGroup pin
7. A bus or NetGroup hierarchical port
8. A bus entry point. However, in the case of a bus entry point, the junction dot will not display.

Example: Say the wire bus or NetGroup (b1[0..1]) is joined with the bus or NetGroup (b2[0..1]), originating at the same connection point and drawn in the opposite direction. A junction dot is created if an orthogonal bus or NetGroup (b3[0..1]) is placed at the point of connection of these two objects.



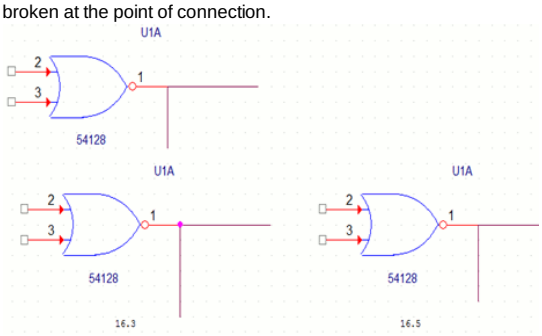
1. Since a junction is the attribute of a wire, it will appear only on a wire-end. If three other types of
   For example, if a port, an off-page connector, and a ground are shorted together, a junction dot is not created at the point of connection. connection objects are shorted together, no junction is created.
2. A junction dot is created on a bus or NetGroup wire irrespective of the widths of the buses or NetGroups constituting the connection.
   For example, a junction is created if the buses or NetGroups b1[0..1], b2[0..3] and b3[0..5] are shorted together.
3. A junction dot can be explicitly added or removed by a user only in the case of crossing wires.

Junction Dot Formation v16.5 and prior releases

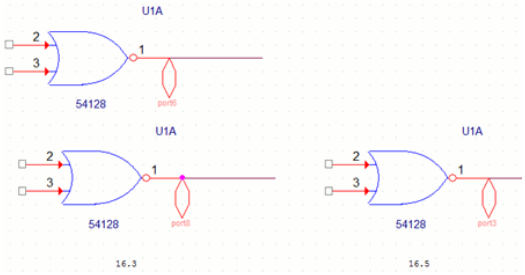
With the implementation of Junction on straight wire-break only, there are some differences in the creation of junctions in v16.5 and later releases of Capture. However, all designs created in releases prior to v16.5 will open with all the existing junctions intact. The enhanced junction

creation logic will apply only when any object contributing to a junction is edited on the schematic page or the page is completely re-evaluated by the system.

1. In Capture v16.5, a junction cannot be placed on slanting crossing wires. This is allowed in prior release.
2. In Capture v16.5, a junction dot cannot be removed from a T-section. This is allowed in prior releases.
3. Depending on the order of the creation of objects, in prior releases, a junction dot may or may not be created when two orthogonal wires are connected with a part pin or any other connectivity object. In v16.5, a junction dot will not be created because no straight wire is broken at the point of connection.



4. Depending upon the order of creation of objects, in prior releases, a junction dot may or may not be created when one wire is connected with two other connectivity objects. In v16.5, a junction dot will not be created, because no straight wire is broken at the point of connection.



5. In prior releases of Capture, if a user drags one of the opposite wires of the T-formation to form a slanting wire, the junction dot remains intact. In v16.5, the junction dot will be removed because no straight wire is broken at the point of connection.

Auto-Wiring in Capture

Use the wire command to draw wires between various objects and provide connectivity on the schematic. This is often a tedious and time-consuming task. To automate the task of wiring the components on a schematic page, use the Auto-Wire feature in Capture.
This feature allows you to wire two or more points on the schematic page. It also allows you to wire multiple points on your schematic to a bus.
When wiring the parts on a page, you wire two (or more) pins on different parts or the same part (for shorting). You can create a net between two (or more) wires and you can also create a net between any number of pins and wires on a page.

In this Section:

Auto-Wire two points

The auto-wire feature in Capture allows you to connect any two points (part pins and / or wires) on a schematic page.

To wire two points on a page (pin-to-pin, pin-to-wire or wire-to-wire).

1. From the Place menu, choose Auto Wire then choose Two Points.
   Or click the Auto Connect two points button on the Draw toolbar.
   Capture is now in the Auto-Wire mode. Notice the cursor changes to the Auto-Wire cursor.
2. Click the pin or wire to start the net.
   As you move the cursor across the page, notice a wire, from the start pin or wire, is formed. And the wire stretches as you move across the page.
3. Click the pin or wire to end the net.
   A wire is created between the start and end points.
4. Choose the selection tool to exit the Auto-Wire mode or go back to step 2 to Auto-Wire

other pairs of pins and wires on the page.

Use this feature to short two pins on a part.

Shortcut

Draw Toolbar: 

Auto-Wire Multiple Points

The Auto-Connect feature of Capture allows you to connect any number of points (pins or wires) on your schematic with an easy-to-use multi-select auto-wiring feature.

To auto-wire multiple points on a page

1. From the Place menu, choose Auto-Wire then choose Multiple Points.
   Or click the Auto Connect multiple points button on the Draw toolbar.
   Capture is now in the Auto-Wire mode. Notice the cursor changes to the Auto-Wire cursor.
2. Click the pin or wire to start the net.
3. Click the next pin or wire on the net.
4. Continue to click on as many pins or wires as required to create the complete net.
   Note: Since you are in the Multiple Point mode, you do not need to press the Ctrl key to multi-select points on the page.
   Note: Since Capture is in the Multiple Point mode a wire is not dragged as you move the cursor across the page. This is unlike the Two-Point Auto Wire mode.
5. Finally, right-click anywhere on the schematic page and choose Connect.

OR

1. Click the pin or wire to start the net.
2. Press and hold down the Ctrl key and click the next pin or wire on the route.
3. Continue to click on as many pins or wires as required to create the complete net.

   > You need to keep the Ctrl key pressed with each new pin or wire selected.

4. With all the pins and wires in the net selected, right-click anywhere on the schematic page and choose Connect.

Shortcut

Tool palette: 

Auto-Wire to Bus

You can use the auto-wiring feature to connect the pins on a part to a bus. You can also connect pins and wires from across the page to a single bus.
In this feature, you simply need to select the pins (and wires) to connect to the bus, choose the Connect to Bus command and finally provide a net alias and all the connections to the bus are made.

To connect Part Pins and / or wires to a Bus

1. From the *Place* menu, choose *Auto Wire* then choose *Connect to Bus*.
   Or click the Auto Connect to Bus button on the Draw toolbar.
   Capture is now in the Auto-Wire mode. Notice the cursor changes to the Auto-Wire cursor.
2. Select any number of pins and / or wires to be connected to the bus.

   > Since you are in the Connect to Bus mode, you do not need to press the `Ctrl` key to multi-select points on the page.

3. Select the bus.

   > As soon as you select the bus, the wire connections between the selected points on the page and the bus are created. Notice that the bus entries for these connections are also mad

   When all the connections to the bus are made, you are prompted for the net alias.
   This net alias will be used for all the connections to the bus. So you need to provide a alias name prefix followed by a numeric range in square brackets. So that each net alias in the connections will use name prefix followed by the sequenced numeric.
   Take the example of the following alias name prefix and number range:
   AD [9-0]
   The net aliases will be named AD9, AD8, AD7 through to AD0.

> If you the numeric range that you provide is greater than the number of objects to be connected to the bus, Capture will discard the un-necessary number values.

4. Enter the net alias name prefix followed by the numeric range.
   All the connections to the bus are complete along with the number sequenced net aliases.

The auto-connect to bus feature is extremely sensitive to the exact location of the mouse click on the objects (wires or pins) on the schematic. When you use the feature to connect wires to a bus, you need to ensure that you click at the precise end of the wires. Alternatively, when using this feature, you will find it easier to connect the bus directly to the pins on the part.

Shortcut

Tool palette: 

> When using the Auto-Connect to Bus feature, ensure that there is at least one grid spacing between pins on the component and the bits on the bus.

Auto-Wire to NetGroup

You can use the auto-wiring feature to connect the pins on a part to the component signals on a NetGroup wire. You can also connect pins and wires to the signals on a NetGroup block.

To auto-connect Part Pins and / or wires to a NetGroup wire

1. From the Place menu, choose Auto Wire then choose Connect to Bus.
   Or click the Auto Connect to Bus button on the Draw toolbar.
   Capture is now in the Auto-Wire mode. Notice the cursor changes to the Auto-Wire cursor.
2. Select any number of pins and / or wires to be connected to the NetGroup wire.

   > Since you are in the Connect to Bus mode, you do not need to press the Ctrl key to multi-select points on the page.

3. Select the NetGroup wire.
   As soon as you select the NetGroup wire, the Select Nets dialog box displays the list of all the signals contained in the selected NetGroup with checkbox to the left of each signal. And a checkbox to the left of the NetGroup name.
4. Check each of the signals in the NetGroup that you want to connect to the pins.
   In the Select Nets dialog, you can check the items at any level. This implies that you can select individual signals, NetGroups or buses. If you select a NetGroup or a bus, all the contained signals are selected.
5. Click OK.
   The nets that connect the part to the bus are assigned aliases depending on the signals in the NetGroup that are connected to the pins on the part.
   For example, if you create a NetGroup N1 that contains two signals S1 and S2. If these signals are connected to a part, the aliases that are assigned to the connecting nets are N1.S1 and N1.S2. However, if you rename the net alias defiend for the NetGroup or signal within the NetGroup, Capture will auto-rename the signals.
   So if you rename the NetGroup net alias to, say, P1, Capture will immediately rename the net aliases on the connecting nets to P1.S1 and P1.S2.
   Also, if you rename the NetGroup signals to, say, T1 and T2, Capture renames the net aliases to P1.T1 and P1.T2. For details on renaming signals in a NetGroup, see To rename a NetGroup member.
   Further, say you connect a NetGroup hierarchical port HP1 to the NetGroup wire N1. The connectivity of the NetGroup hierarchical port will override the connectivity of the NetGroup wire. So, if you now auto-wire the NetGroup to the part, the aliases on connecting nets are HP1.S1 and HP1.S2. This means that renaming the NetGroup hierarchical port will affect the net aliases names. However, renaming the NetGroup wire will not affect the net wire will not affect the names of the net alias names.

Consider the above scenario, where you chose a combination of a NetGroup wire and a NetGroup hierarchical port. Say you auto-wire the part to the NetGroup wire. Immdiately the net aliases are set to N1.S1 and N1.S2. Now, the net aliases naming (and auto-renaming) is associated with the NetGroup wire. So, if after this you connect the NetGroup hierarhcial port to the NetGroup wire the net aliases are not renamed to the port. Note that this is only for purposes of net alias naming. The connectivity of the nets still depends on the precedence of objects. So the pins on the part are shorted to the signals on the NetGroup port. This can easily be verified by checking the flat net names of the connecting nets.

To auto-connect Part Pins and / or wire to the signals in a NetGroup block

1. From the Place menu, choose Auto Wire then choose Two Points.
   Or click the Auto Connect two points button on the Draw toolbar.
   Capture is now in the Auto-Wire mode. Notice the cursor changes to the Auto-Wire cursor.
2. Click the pin or wire to start the net.

As you move the cursor across the page, notice a wire, from the start pin or wire, is formed. And the wire stretches as you move across the page.

3. Click the NetGroup entry pin on the NetGroup block.
   A wire is created between the pin / wire and the NetGroup entry.
4. Choose the selection tool to exit the Auto-Wire mode or go back to step 2 to Auto-Wire other pairs of pins and wires on the page.

Bus connectivity

A bus is a group of scalar signals (wires), and is never connected to a net. Once a bus acquires a valid name or alias, that name or alias defines the signals carried by the bus and connects those signals to the corresponding nets. For example, the alias A[0:3] defines a four-signal bus that connects the four bus signals A[0], A[1], A[2], and A[3] to the individual wires named A0, A1, A2, and A3. Net aliases on wires do not use brackets.

You can place one pin on a part that represents all the pins for a bus. Such a pin is called a bus pin. Bus pins use the same naming convention as buses.

You can use bus pins in most of the cases where you can use scalar pins. Examples of these would be:

- Off-page connectors.
- Hierarchical ports.
- Hierarchical pins of nonprimitive parts and hierarchical blocks.

Bus pins will only work with the VHDL netlist format. No other netlist format understands them.

Do not use bus pins in the following situations:

- Pins on primitive parts.
- Any design that you intend to use with PCB Editor.

Like wires, buses can acquire names and aliases by two means:

- Direct application of a valid bus name
- Electrical connection to a hierarchical port, off-page connector, or global bus pin with a valid bus name or alias

In addition to the rules by which netnames are resolved, bus names and aliases follow these general rules:

- If one alias defines a subset of the signals defined by another, like-named signals are connected. For example:
  Given aliases A[0..2] and A'[0..5]:
  A[0] connects to A'[0], A[1] connects to A'[1], and A[2] connects to A'[2].
- If the base names differ, or if neither alias defines a subset of the signals defined by the other, signals are connected in a bit-wise manner (m to m,. . . , n to n). For example:
  Given aliases A[0..2] and A'[1..3]:
  A[0] connects to A'[1], A[1] connects to A'[2], and A[2] connects to A'[3].
  Given aliases A[0..2] and B[5..0]:
  A[0] connects to B[5], A[1] connects to B[4], and A[2] connects to B[3].

As you place buses and wires, remember the following points:

- A bus and a wire can be connected only by a name.
  - If you begin or end a bus segment on a segment of a wire, a vertex is added to the wire, but no junction appears—the bus and wire are not connected.
  - If you begin or end a wire segment on a segment of a bus, a vertex is added to the bus, but no junction appears—the wire and bus are not connected.
- Two buses or two wires can be connected physically.
  - If you begin or end a bus segment on a segment of another bus, a vertex is added to the second bus, and a junction appears—the buses are connected.
  - If you begin or end a wire segment on a segment of another wire, a vertex is added to the second wire, and a junction appears—the wires are connected.

> Capture preserves the case of part names and netnames, but ignores the case when comparing names for electrical connection. That means you may use uppercase or lowercase letters as you wish, but you need not remember the case.

To make connections to a bus, you label the bus, label the signals that are members of the bus, and assign an alias to each signal entering and leaving the bus. Each signal bears an alias that is within the bus range. For example, if the bus alias is ADDR[0..3], the four bus members must bear aliases ADDR0, ADDR1, ADDR2, and ADDR3.
In Capture, you can use an alias to connect a signal from one area of your schematic page to another without placing a bus between the areas. For example, suppose you have placed the bus TIMING[1..4] on your schematic page and you want to connect it to another object at the opposite corner of the schematic page. Instead of drawing a bus from TIMING[1..4] to the other object, you can assign the alias TIMING[1..4] to the other object.
To provide a visual cue that a signal is tied to a bus, you can physically connect the signal to the

bus. It is recommended that you use a bus entry for this connection. The advantage of using a bus entry is that two bus entries can be connected at the same point on a bus without connecting the signals. If two wires are run directly to a bus at the same location, the signals are connected.

As you place buses and wires:

- A bus and a wire can be connected only by name.
  - If you begin or end a bus segment on a segment of a wire, a vertex is added to the wire and a junction appears, but the bus and wire are not electrically connected.
  - If you begin or end a wire segment on a segment of a bus, a vertex is added to the bus and a junction appears, but the wire and bus are not electrically connected.
- Two buses or two wires can be connected physically.
  - If you begin or end a bus segment on a segment of another bus, a vertex is added to the second bus, and a junction appears—the buses are connected.
  - If you begin or end a wire segment on a segment of another wire, a vertex is added to the second wire, and a junction appears—the wires are connected.

To create a bus

1. From the Place menu, choose Bus.
2. Click to start the bus.
3. Use the mouse to draw the bus.
4. Click to place a vertex and change directions. The vertex is constrained to a multiple of 90 degrees.
5. Double-click to end the bus.
6. Select the selection tool to dismiss the bus tool or repeat from step 2 to place additional buses.

To place a non-orthogonal bus

- Hold the Shift key while you draw the bus. There is no constraint on vertex angles.

To name a bus

1. From the Place menu, choose Net Alias.
   Capture displays the Place Net Alias dialog box.
2. Following the naming conventions for buses, enter the net alias in the dialog box that appears, then click OK. The bus appears in the selection color.
3. Use the mouse to move the net alias and click on the bus to place the net alias. The net alias appears in the selection color. The tip of the pointer must be touching the net for you to place the net alias.
4. Select the selection tool to dismiss the net alias tool. The alias is added to the alias list for the net.

Shortcut

Tool palette: 

To connect single-signal nets to a bus

1. Place the bus and assign it a name.
2. From the Place menu, choose the Bus Entry command. The bus entry symbol is attached to your pointer.
3. If the entry is not at the angle you need, then from the Edit menu, choose the Rotate command to rotate the entry 90 degrees counterclockwise.
4. Use the mouse to position one end of the entry on the bus, then click to place the bus entry.
5. Repeat step 4 until all bus entries are placed. If you place the bus entries at regular intervals, you can simplify connecting the single-signal nets to the bus entries.
6. Place a wire to connect the first bus entry to one net, and place an alias, taking care to assign this bus member the lowest value in the bus range.
7. Select the entire wire and press Ctrl and drag the wire so that it connects the next net to the bus. Note that the wire is copied to the new location and the alias value is increased by one. However, if you select only an end of the wire (and not the entire wire) and then drag the wire keeping Ctrl pressed, the wire is moved to the new location along with the original alias value.
8. From the Edit menu, choose the Repeat command. The wire and the incremented alias are placed at the specified distance from the previous set.
9. Repeat step 8 for every net in the bus or repeat steps 7 and 8 as needed, then select the selection tool to dismiss the set.

> You can place one pin on a part that represents all the pins for a bus. Such a pin is called a bus pin. Bus pins use the same naming convention as buses.

Shortcut

Tool palette: 

In this section:

- Naming Conventions

Naming Conventions

For a Bus

A bus name must have the form basename[m..n] where m..n specifies a range of decimal integers representing the signal numbers of bus members. There are (n - m + 1) wires in the bus. You can use two periods (..), a colon (:), or a dash (-) between m and n.
Examples:

ADDR[0..31]	(32 members)
DATA[16:31]	(16 members)
CONTROL[4-1]	(4 members)
A[100..190]	(91 members)

Do not add any space between the basename and the left bracket ([), because this can cause problems during the netlist operations.
Also, note that you should not end a bus name with a numeric character (0-9), because this can cause problems during the netlist operations. However, numeric characters can occur at other places in the bus name, however. For example, BUS2A will work, but BUSA2 could cause problems when you generate the netlist.

For a Bus Member

The name of a bus member must have the form basename N where N is the bus member's signal number in the bus. The signal can have additional aliases, but it must have this name to be connected to the bus. Also, bus members cannot have a preceding zero in their name. For example, A0 is a legal name for a bus member, but A00 is not.

Modifying Wires and Buses

In this section:

- Labeling wires and buses
- Editing wire and bus look and feel
- Moving Connectivity Objects
- Deleting wires and buses

Labeling wires and buses

You use aliases to connect electrical objects.

To place an alias

1. From the Place menu, choose Net Alias command.
2. Enter the net alias text, following the naming conventions for buses and bus members, then click OK. A rectangle representing the alias text is attached to the pointer. The tip of the pointer must be touching the net for you to place the net alias.
3. Use the mouse to move the alias text and click directly on the wire or bus. The alias text appears in the selection color.
4. Select the selection tool or press Esc, to dismiss the net alias tool.

To label a series of bus members

1. Use the Repeat command to place the bus members at regular intervals.
2. On the first bus member, place one alias, taking care to assign this bus member the lowest value in the bus range.
3. Place a net alias, using the left mouse button, on each member of the series.
4. Select the selection tool, or press Esc, to dismiss the net alias tool.

Shortcut

Tool palette: 

To edit net alias text

1. Select the net alias.
2. From the Edit menu, choose the Properties command.
3. In the dialog box that appears, you can change the color, the font, the rotation, or the alias.
4. Click OK to dismiss the dialog box.

To move net alias text

- Select the net alias text and drag it to another location.

Editing wire and bus look and feel

You can change the look and feel of a wire or a net on a schematic page by changing the color,

line style or line width.

To edit the look and feel of a wire or bus

1. Click on a wire or bus on the page.
2. Right-click the wire or bus to display the pop-up menu.
3. Click the Edit Wire Properties option.
   The Edit Properties dialog box displays.
   The dialog box contains three drop-down lists to edit the line style, line width and color of the wire.
4. Make the required selections in the drop-down lists and click OK.

> To change the style of the wire to the OrCAD Capture default style, choose the Default option in any of the drop-down lists.

To edit the look and feel of a net

1. Click on a net on the page.
2. Right-click the net to display the pop-up menu.
3. Click the Edit Net Properties option.
   The Edit Properties dialog box displays.
   The dialog box contains three drop-down lists to edit the line style, line width and color of the wire.
4. Make the required selections in the drop-down lists and click OK.

> To change the style of the wire to the OrCAD Capture default style, choose the Default option in any of the drop-down lists.

Combining wire and net look and feel

If you change the look and feel of a wire in a net and then change the look and feel of the net, the properties of the wire will override the properties of the net.

Scenario 1

- Say a net contains two sections (wires).
- Select one wire in the net and change the color property of the wire to red.
- The color of the other wire in the net is not affected.
- Now select the net and change the color of the entire net to blue.
- The color of the second wire in the net changes to Blue. However, the color of the first net remains red.
- This is because the custom properties of a wire will override the properties of a net.

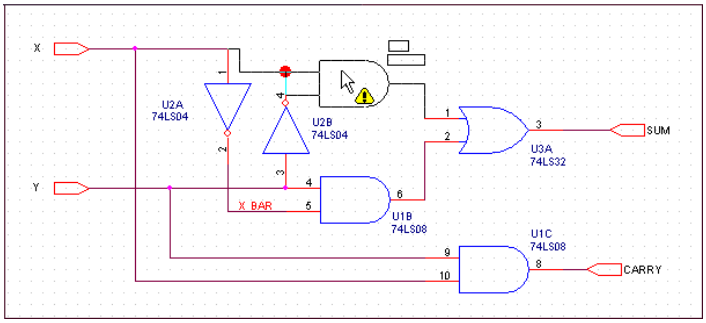
Scenario 2

- Say a net contains two sections (wires),
- Select one wire in the net and change the color property of the wire to red.
- The color of the other wire in the net is not affected.
- Now select the net and change the color of the entire net to blue.
- The color of the second wire in the net changes to blue. However, the color of the first net remains red.
- Again, change the color of the wire to Default.
- The wire color changes to blue.
- This is because the wire now inherits the color of the net.

Moving Connectivity Objects

Connectivity and orthogonal drag

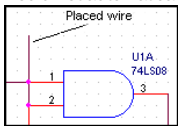
Connectivity, of course, is of vital importance to an electronic design. Therefore, it is important to understand how connectivity is maintained when you move objects on a schematic page.
Capture draws wires that maintain connectivity with a moved object in a stair step (orthogonal) fashion. When you reposition an object, connectivity may be affected.
Capture warns you of connectivity changes as you drag the object by placing markers at the connectivity change points visible on the page. At the same time, the cursor changes to an exclamation point as shown in the following schematic, and the status line warns of net connectivity changes.



Because some connectivity changes may not be visible on screen, most connectivity changes for which you see an alert are documented in the session log.
Capture automatically places junctions to create electrical connections at "T" intersections where wires abut and do not cross. Also, Capture places a junction for you where a wire crosses a pin.

Note: Before you edit a design created in an earlier version of Capture in Capture Release 9.1, you should run a Design Rules Check to show where Capture will place junctions in your design. If you do not want electrical connections at "T" intersections and on pins where wires cross, you can adjust the design as necessary by using an earlier version of Capture before you edit in Capture Release 9.1.

In Capture Release 9 and earlier, you could place wires as shown in the figure below and Capture would not automatically place junctions for you at "T" intersections or pins.



If you complete the operation, connectivity change warnings will appear in the session log as shown below.

```
---------------------------------------------------
The following 2 points have been identified as net connectivity change
points from the last operation
---------------------------------------------------
(2.60, 1.80)
(2.60, 2.00)
```

The orthogonal drag feature eliminates most unintentional disconnects. As such, Capture will no longer warn you of some connectivity changes caused by dragging objects. If you drag the end of a wire or wire segments that are connected to pins or net symbols, they will disconnect. If you drag wire-to-pin or wire-to-net symbol disconnects back into place, you will see connectivity change indicators. Wire-to-wire connections stretch to maintain connectivity, and junctions maintain connections when you drag wire segments, objects, or entire nets.
The following table shows when Capture provides a connectivity change alert:

Object Type	Drag	Place	Pasting	Resizing
Part	Yes	Yes	Yes	N/A
Hierarchical Block	Yes	No	Yes	N/A
Wire	Yes	Yes	Yes	Yes
Bus	Yes	Yes	Yes	Yes
Net Symbol	Yes	Yes	Yes	N/A
Bus Entry	Yes	No	Yes	N/A

To drag a segment orthogonally

- Select the segment and drag it to the new location. The wire or bus stretches orthogonally to maintain connectivity.

To drag a segment non-orthogonally

- Hold the Shift key while you select the end point and drag the wire. The selected end of the wire moves in any direction while the opposite end remains in place.

When you click on a wire segment, only that segment and its two handles are selected.

When you drag a part or wire to another location and that change affects connectivity, Capture flags a warning with a changed cursor ( ) and temporary markers ( ) on your schematic.

Visible and off-screen connectivity changes are saved in the session log, if you complete the operation.

Capture now provides you the following UI options to control the drag operation:

- A check box named *Allow component move with connectivity changes* in the Miscellaneous tab of the Preferences dialog box.
- A toolbar button with the following toggle states:
  - If the check box is selected or the toolbar button is in the  state, then Capture will allow you to drag and place the selected part or wire on the schematic, even if it results in connectivity changes. Also, Capture will flag a warning with a changed cursor and will show the temporary markers.
  - However, if the check box is not selected or the toolbar button is in the  state, then the selected part or wire attaches to the cursor and does not get placed on the schematic, if it results in connectivity changes. Also, Capture flags only a warning with a changed cursor and does not show the temporary markers.

To move a wire or bus

1. Select the wire or bus.
2. Hold the Alt key while you move the wire or bus. The wire or bus segment breaks connectivity with the rest of the net.

> Moving pins connected to wires may cause wires to drag, but moving a wire always causes disconnection from pins, ports, and other objects.

To copy a wire

- Hold the Ctrl key while you drag the wire.

To move a vertex

1. Select a wire segment next to the vertex.
2. Drag the vertex to the new location.
   One segment of the wire or bus stretches to the new location.
   The other segment breaks connectivity.

Deleting wires and buses

To delete a wire or bus segment

1. Select the segment.
2. From the Edit menu, choose Delete.

To delete a net

1. Select one segment of the net.
2. Right click. A pop-up menu appears.
3. From the pop-up menu, choose the Select Entire Net command.
4. From the Edit menu, choose Delete.

Shorting Part Pins

You short the pins on a part to connect these together. Capture provides the Connect command to short multiple pins on a part.

To short pins on a part

1. Multi-select the pins to be shorted.
   You can do this by clicking on the pins to short while keeping the Ctrl keep pressed.
   If the pins are on one side of the part, click outside the part and drag the mouse over the pins to be shorted.
2. Right-click and choose Connect.

All the selected pins are shorted.

Placing power, ground, and no connect symbols

In this section:

No connect symbols

The Design Rules Check tool checks for unconnected pins. If you intentionally leave a pin unconnected in a schematic page, it needs a no connect symbol. The Design Rules Check tool ignores unconnected pins with no connect symbols.
If a pin with a no connect symbol is connected to a net, the no connect symbol has no effect on the pin and becomes invisible. If the pin is later disconnected from the net, the no connect symbol becomes visible again.

To place a no connect symbol

1. Press Shift, and then X keys.
2. Position the mouse pointer over the pin, and click. The end of the pin changes from a square (unconnected) to an X (not connected).
   OR
3. From the Place menu, choose No Connect.
4. Position the mouse pointer over the pin, and click. The end of the pin changes from a square (unconnected) to an X (not connected).
   OR
5. Select the pin.
6. From the Edit menu, choose Properties. The property editor appears.
7. Change the filter to <All>.
8. Select the Is No Connect property.
9. Click Apply, and close the property editor.
10. Click the left mouse button in any open space on the schematic page. The end of the pin changes from a square (unconnected) to an X (not connected).

No connects cannot be deleted with the Delete command.

To remove a no connect symbol

1. From the Place menu, choose No Connect.
2. Position the mouse pointer over the pin, and click. The end of the pin changes from an X (not connected) to a square (unconnected).
   OR
3. Select the pin.
4. From the Edit menu, choose Properties. The property editor appears.
5. Change the filter to <All>.
6. Clear the Is No Connect property.
7. Click Apply, and close the property editor.
8. Click in any open space on the schematic page. The end of the pin changes from a square (unconnected) to an X (not connected).

Power and ground symbols

When you place a part that has power and ground pins, the power and ground pins of the part are automatically connected to like-named global power and ground nets of the schematic folder. This happens because, when you place the part, the power and ground pins of the part are assigned a net name that is the same as the pin name. If you need to isolate one power or ground pin from the others, you can assign it a unique net name.
Power and ground pins are invisible and global by default. This means that they are connected, on a project -wide basis, to all pins, power objects, and nets of the same name.
If you need to isolate a power or ground net, do one of the following:

- make the pin visible and connect it to another net or power object
- display the invisible power pin and connect it to a net or power object

For information on making power pins visible and on displaying invisible power pins, see Making power pins visible.

To place power or ground symbols

1. From the *Place* menu, choose Power
   OR
   Ground.
   The Place Power or Place Ground dialog box appears.
2. In the Place Power dialog box, select a power symbol and click *OK*.
   OR
   In the Place Ground dialog box, select a ground symbol and click *OK*.
3. Use the mouse to move the symbol to the appropriate location and click. The symbol appears in the selection color.
4. Select the selection tool, or press Esc, to dismiss the power or ground tool.
5. Click an area where there are no parts or objects to deselect the symbol.

> To place DC ground ('0') symbols in your PSpice designs, see Placing PSpice ground 0 symbols for PSpice simulations.

Shortcut

Tool palette: 

To rotate power or ground symbols

1. Select the symbol.
2. From the Edit menu, choose the Rotate command. The symbol rotates 90 degrees counterclockwise.
3. Repeat step 2 as necessary.

4. Click an area where there are no parts or objects to deselect the symbol.

To create a power or ground symbol

1. Open the library that is to hold the new symbol, and select the library in the project manager.
2. From the *Design* menu, choose the New Symbol command. The New Symbol dialog box appears.
3. Enter a name and select Power as the Symbol Type, then click *OK*. The part editor opens with an empty part boundary box.
4. Use the graphics tools to create the symbol; the part boundary box dimensions change to accommodate the graphic elements.

To isolate a power net to a schematic folder

- Place a power symbol and attach it to a hierarchical port.

To isolate a power net to a schematic page

1. Place a power symbol and attach it to an off-page connector.
   When Capture resolves net name conflicts, the name of the off-page connector takes precedence over the name of the power object, and the scope of the off-page connector is limited to the schematic folder. All pins on the same page that are connected by name or by wire to the power symbol are connected to the isolated power net.
   For example, say you want to isolate your analog and digital grounds and then connect them at one point when you make a printed circuit board. You place your analog circuitry on a separate schematic folder. On each page in the analog schematic folder, you place a ground symbol with the name GND. This implicitly connects all the pins named GND to ground. Then you connect that power symbol to an off-page connector named AGND. To connect AGND to the digital ground (GND), you can create a part whose footprint is a strip of copper with two pads, GND and AGND.

Working with Power Pins

In this section:

Browsing for Power Pins

In Capture, power and ground supply pins are referred to generically as "power pins". Normally, power pins are invisible, and global—this means they are shorted by name or they are connected to like-named power pins, power objects, and power nets throughout the schematic folder. However, if you need to change the shorting behavior on the pins, you would need to make them visible and then work on the pin properties.
Alternatively, you can view the list of all the power pins in a schematic design and then edit the power names or define them as NC pins.

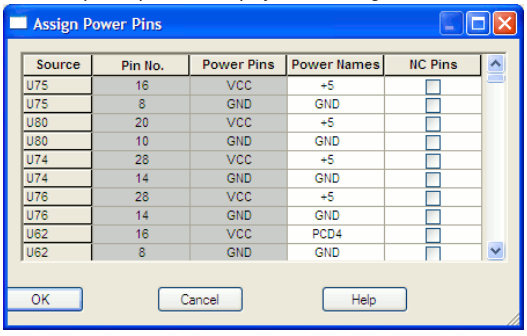
To browse the power pins in a design

1. Select the design in the Project manager.
2. On the Edit menu choose Browse – Power Pins.
3. In the Browse Properties dialog box, you can choose the mode as occurrences or instances and click OK.

This list displays all the power pins in this design.
If you double-click on a line item in this list, you are directed to the component containing the selected power pin. However, since the pins are all invisible, you are still not able to make any changes to the pin properties.

To change the properties of a power pin

1. Click the power pin list.
2. On the Edit menu choose the Properties option (Or Press Ctrl + E).
   The list of power pins now displays in the Assign Power Pins dialog.



You can now use the Power Name property on a pin to change the shorting behavior on the pin. This implies that you can change the power name of a pin.

To change the power name of a pin

1. Pick another power name from the drop-down list on the Power Names cell of a selected Power pin.
   OR
   Type an alternative power name in the power name cell.

You can also define a power pin as an NC pin by checking the NC Pin option.
This functionality allows you to work with and change the power name and NC Pin property of power pins without having to first make the pins visible.

> If you set the NC Pins property of a part, the property Editor for the part will reflect this change by adding an NC_PINS property line item.

Making power pins visible

In Capture, power and ground supply pins are referred to generically as "power pins". Normally, power pins are invisible, and global—that is, they are connected to like-named power pins, power objects, and power nets throughout the schematic folder. You can override this default connection by making a power pin visible and connecting it to a wire or other connectivity object. If you connect a power pin to a net using a hierarchical port, or off-page connector, then the pin is no longer global.
Capture can also display invisible power pins on individual part instances or throughout a design. Merely displaying an invisible power pin does not change its global nature; however, connecting a wire or other connectivity object to an invisible power pin isolates it from the design-wide (global) net. If there are duplicates of the pin in the devices of a multi-part package, then all of the pins should be made visible, then wired.

To display invisible power pins

Invisible power pins always appear in the part editor. The method by which you display invisible power pins in the schematic page editor determines whether you can connect wires and other connectivity objects to them

On a part instance

1. Select the part in the schematic page editor.
2. From the Edit menu, choose Properties.
3. Find the Power Pins Visible property column on the property editor Parts tab and select the check box, then close the property editor.

Connecting a wire or other connectivity object to a power pin made visible by this method isolates that pin from the design-wide power net.

Throughout a design

1. In the Project manager Options menu, choose Design Properties.
2. In the Design Properties dialog box, choose the Miscellaneous tab.
3. Select the Display Invisible Power Pins option (for documentation purposes only).
4. Click OK.
   You cannot connect to a power pin made visible by this method.

To make power pins visible

A power pin is by default connected to a global net that has the same name as the power pin. You can override this default connection by making the power pin visible by either of the methods below and connecting it to a wire or other connectivity object.

On a new part

1. In the Part editor Place menu, choose Pin.
2. In the Place Pin dialog box, change the Type to Power.
3. Verify that the Pin Visible option is selected.
4. Click OK.
5. Place the pin.

For a power pin that is already placed, select the pin in the part editor. From the part editor's Edit menu, choose Properties. In the Pin Properties dialog box, verify that the Type is Power and that the Pin Visible option is selected, then click OK.

On a part instance

1. Select the part in the schematic page editor.
2. From the Edit menu, choose Part.
3. For each power pin you want to make visible, select the pin, then choose Properties from the Edit menu.
4. In the Pin Properties dialog box, change the Type to Power.
5. Verify that the Pin Visible option is selected.
6. Click OK.

7. When you finish, close the part editor window.
8. In the Save Part Instance dialog box, choose whether to apply your changes to all instances of the part in the design or only the selected (current) instance.
9. Click OK.

When you edit a part's graphic representation on a schematic page, you break the connection between the part and the library; if you want to reverse your edits, you use the

Replace Cache command of the Design menu. For more information, see Replacing and Updating the cache .

Placing off-page connectors

Off-page connectors provide connections between schematic pages within the same schematic folder. An off-page connector is connected by name to other off-page connectors within the same schematic folder.

- Like-named off-page connectors in different schematic folders are not connected.
- The Select Entire Net command is restricted to the active schematic page—it does not follow hierarchical blocks, hierarchical ports, or off-page connectors across schematic folders or schematic pages. For more information, see Tracing a net.
- Remember that nets on a schematic page are electrically connected by name, by alias, or by connection to a named hierarchical port or off-page connector.
- To connect an off-page connector to a bus, name the off-page connector with the same name and range as that of the bus. For example, to connect an off-page connector to a bus named `ABC[0:3]`, name the off-page connector as `ABC[0:3]`.

To connect schematic pages laterally (within the schematic folder)

1. From the Place menu, choose Off-Page Connector.
2. Select a symbol (standard or user-created), enter a name, and choose OK.
3. Place the symbol anywhere on the schematic page.
4. Repeat steps 1 through 3 for the other schematic pages (within the same schematic folder) that you wish to connect.
   The size of a part or a symbol is limited to 32 by 32 inches.

To create a hierarchical port or off-page connector

1. Open the library that will hold the new symbol.
2. From the Design menu, choose New Symbol. The New Symbol Properties dialog box appears.
3. Enter a name and select off-page connector or hierarchical port as the symbol type, then click OK. The part editor opens with an empty part boundary box.
4. Use the graphics tools to create the symbol. The symbol dimensions expand automatically to accommodate the graphics.
5. From the File menu, choose Save. If you are creating the symbol in a new library that has not yet been saved, the Save As dialog box appears, giving you the opportunity to name the library file.

- If you edit a library provided by OrCAD, it is important that you assign a new library name so that your changes are not overwritten when you upgrade or update your software.
- When you save a project, Capture automatically creates a backup with a .DBK file extension. When you save a library, Capture automatically creates a backup with a .OBK file extension. If you save only a schematic page or a part, no backup is generated.

Adding hierarchical ports

Hierarchical ports and hierarchical pins provide connection between levels of hierarchy on a schematic page.
Inside a hierarchical block, a hierarchical pin provides only vertical (downward-pointing) connection. It is connected by name to hierarchical ports on schematic pages within the attached schematic folder or to the appropriate signals in the VHDL entity port definitions. You can think of its function as bringing a net "up" from the attached implementation into the hierarchical block (but not out onto the schematic page).
Outside a hierarchical block, a hierarchical port provides vertical (upward pointing) and lateral connections. Its connected vertically to the like-named hierarchical pin inside any hierarchical block to which it is attached. Its connected laterally to like-named nets, hierarchical ports, and off-

page connectors within the same schematic folder. You can think of its function as carrying a net out of the schematic folder.
Before you create or re-size a hierarchical block, make sure the Snap to grid option is turned on (from the schematic page editor's Options menu, choose Preferences). If the hierarchical block is on Fine grid, then hierarchical pins inside it are also on Fine grid—even if you change the Snap to grid setting before you place them—and it may be difficult to connect to these off-grid hierarchical pins.
A part with an attached schematic folder functions exactly like that for hierarchical blocks, and pins on such a part function exactly as described for hierarchical pins within a hierarchical block. You can use the same attached schematic folder for either method of defining a hierarchy. The only difference between the two methods is that a part with an attached schematic folder is easier to reuse. See Creating parts and assigning properties for related information.
If you choose the Descend Hierarchy command on a non-primitive part or hierarchical block, and Capture cannot find the attached implementation, Capture creates a schematic folder or VHDL model in the active design.
If you attach an existing implementation to a hierarchical block, Capture automatically creates the hierarchical pins that correspond with the schematic folder's hierarchical ports or the VHDL models port definitions. If you descend the hierarchy on a hierarchical block whose implementation does not yet exist, then Capture automatically creates the hierarchical ports (for schematics) or port definitions (for VHDL models) that correspond with to the hierarchical pins of the hierarchical block.

> If you attach external schematic folders or other files to hierarchical blocks in a design or parts in a library, be sure to include the attachments when you pass the design or library to a board fabrication house or to another engineer. Attached schematic folders and other files are not carried along automatically when you copy or move a part, schematic folder, or schematic page to another library, design, or schematic folder. Only the "pointers" to the attached schematic folders and files —that is, their names and the names of the designs or libraries that contain them— are carried along.

Attached files work much like their counterparts in email—they do not provide an alternative definition of the part (as do attached schematic folders).

> When you attach a schematic folder to a part or hierarchical block, you can specify a full path and file name in the Library field. So, although you can specify a library that has not been saved, you should not try to descend into the attached schematic folder until the library that contains the schematic folder has been saved.

If you do not specify a full path and file name in the Library field, Capture expects to find the attached schematic folder in the same design as the part of hierarchical block to which it is attached. If the specified schematic folder does not exist in either the design or library, Capture creates the schematic folder when you descend the hierarchy on the part or hierarchical block.

For compatibility with future versions of Windows, Capture preserves the case of the path and filename as you specify them in the Library field.

> Note: The Select Entire Net command is restricted to the active schematic page—it does not follow hierarchical blocks, hierarchical ports, or off-page connectors across schematic folders or schematic pages. For more information, see Tracing a net.

Remember that nets on a schematic page are electrically connected by name, by alias, or by connection to a named hierarchical port or off-page connector.

To connect pages vertically (through a hierarchical block)

If necessary, place the hierarchical block and attach the implementation (see Creating hierarchical blocks for instructions). Complete these steps to bring the net into the hierarchical block:

1. Bring the net into the hierarchical block:
2. Select the hierarchical block.
3. From the Place menu, choose Hierarchical Pin.
4. Enter a name and choose OK.
5. Place the symbol within the boundaries of the selected hierarchical block.
   This hierarchical pin is downward pointing—it is connected to any like-named hierarchical port on any schematic page in the attached schematic folder.
   To complete the procedure, carry the net up to the hierarchical block.
6. Open a schematic page contained in the schematic folder attached to the hierarchical block mentioned above.
7. Make sure no hierarchical block is selected.
8. From the Place menu, choose Hierarchical Port.
9. Select a symbol, enter the name used in step 3 of the preceding sequence, and choose OK.
10. Place the symbol anywhere (except inside a hierarchical block) on the schematic page.
    This hierarchical port is upward pointing—it is connected to any like-named hierarchical pin inside any hierarchical block to which it is attached.

11. If necessary, use off-page connectors to carry the net to other schematic pages in the same schematic folder (see Placing off-page connectors for instructions).

> - Be careful not to create recursion in your design. Capture cannot prevent recursion, and the Design Rules Check command does not report it.
> - Recursion causes Capture to process infinitely as it tries to expand the design, resulting in the loss of any changes you've made to your design since it was last saved.
> - You can use the copy and paste keyboard shortcuts (CTRL+C and CTRL+V) to enter the same name in the Name text field of both dialog boxes.

To connect hierarchical ports or off-page connectors with nets

- Extend the net to the hierarchical port or off-page connector by placing a wire or bus.
  OR

1. Select the hierarchical port or off-page connector and choose Properties from the Edit menu.
2. In the Name or Value field, type the name of the net, and click OK.
   OR
3. Select the hierarchical port or off-page connector's name and choose Properties from the Edit menu.
4. In the Name or Value field, type the name of the net, and click OK.

Establishing connectivity in schematic pages

In Capture, you connect schematic folders and schematic pages by extending nets between them, using off-page connectors, hierarchical blocks, and hierarchical ports. Off-page connectors carry nets between schematic pages within a single schematic folder. Hierarchical blocks and hierarchical ports carry nets between schematic folders.
A part with an attached schematic folder functions exactly as described for hierarchical blocks, and pins on such a part function exactly as described for hierarchical ports within a hierarchical block. You can use the same attached schematic folder for either method of defining a hierarchy. The only difference between the two methods is that a part with an attached schematic folder is easier to reuse. See Creating parts and assigning properties for related information.
If you choose the Descend Hierarchy command on a non-primitive part or hierarchical block, and Capture cannot find the attached schematic folder, Capture creates a schematic folder in the active design.

> The Select Entire Net command is restricted to the active schematic page—it does not follow hierarchical blocks, hierarchical ports, or off-page connectors across schematic folders or schematic pages. For more information, see Tracing a net.

Remember that nets of a schematic page are electrically connected by name, by alias, or by connection to a named hierarchical port or off-page connector.

> Capture preserves the case of part names and net names, but ignores the case when comparing names for electrical connection. That means you may use uppercase or lowercase letters as you wish, but you need not remember the case.

When you attach a schematic to a part or a hierarchical block, you can specify a full path and filename in the Library text box. So, although you can specify a library that has not been saved, you should not try to descend into the attached schematic folder until the library that contains the schematic folder has been saved.
If you do not specify a full path and filename in the Library field, Capture expects to find the attached schematic folder in the same design as the part of the hierarchical block to which it is attached. If the specific schematic folder does not exist in either the design or library, Capture creates the schematic folder when you descend the hierarchy on the part or hierarchical block. For compatibility with future versions of Windows, Capture preserves the case of the path and file name as you specify them in the Library field.

To extend a net across schematic pages within a single schematic folder

1. Open the schematic page editor on a page that contains the net.
2. From the Place menu, choose the Off-Page Connector command.
3. Select a symbol and enter a name in the Name field; then click OK.
4. Connect the off-page connector to the net, either by name or by wire.
5. For each schematic page on which the net resides (and within the same schematic folder), repeat steps 1 through 4, using the same name for each off-page connector you place.

To extend a net through a hierarchy

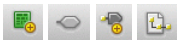
1. Open the schematic page editor on the parent page.
2. Place a hierarchical block, then assign a name to the hierarchical block.
   OR
   Place a non-primitive part.

3. If necessary, attach a schematic folder to the hierarchical block or part.
4. If you placed a hierarchical block in step 2, then from the Place menu, choose the Hierarchical Pin command and assign the pin a name.
5. Open a schematic page in the attached schematic folder.
6. Place a hierarchical port using the Hierarchical Port command with the same name of the hierarchical pin you used in step 4, then place wires to connect the hierarchical port to the net.
7. Repeat steps 4 through 6 for each hierarchical pin in the hierarchical block, or for each pin on the part.

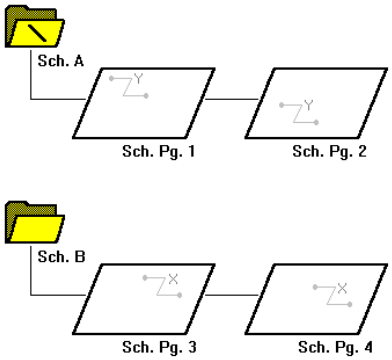
- Be careful not to create recursion in your design. Capture cannot prevent recursion, and the Design Rules Check command does not report it.
- If you attach external schematic folders or other files to hierarchical blocks in a design or parts in a library, be sure to include the attachments when you pass the design or library to a board fabrication house or to another engineer. Attached schematic folder and other files are not carried along automatically when you copy or move a part, schematic folder, or schematic page to another library, design, or schematic folder. Only the "pointers" to the attached schematic folders and files—that is, their names and the names of the designs or libraries that contain them—are carried along.

Attached files work much like their counterparts in email—they do not provide an alternative definition of the part (as do attached schematic folders).

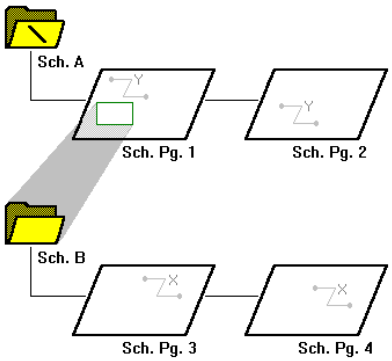
Shortcuts

Tool palette: 

EXAMPLE

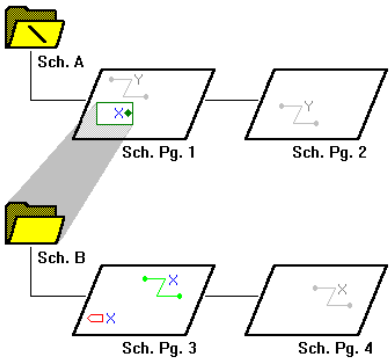


This figure shows two schematic folders, A and B, with two schematic pages each. The schematic folder marked with a backslash (\) is called the root schematic folder. In this demonstration, you see how to create a simple hierarchy.



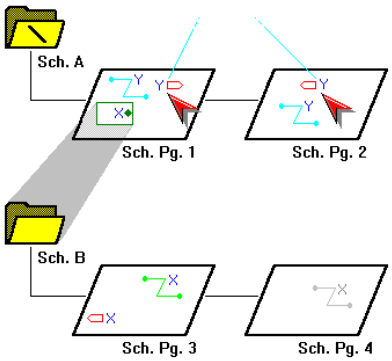
To establish the hierarchy with schematic folder A "above" schematic folder B:

1. Place a hierarchical block on schematic page 1.
2. Attach schematic folder B.

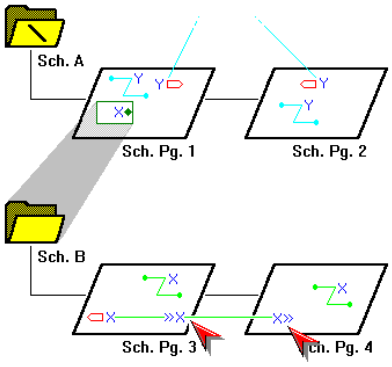


To carry a net between schematic folders A and B:

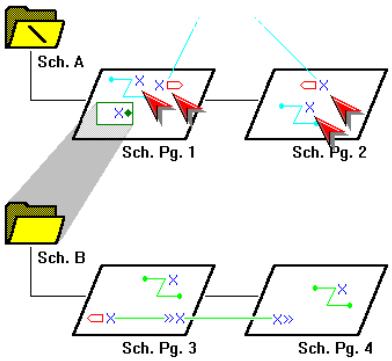
1. Select the hierarchical block on schematic page 1 and place a hierarchical pin named X inside it.
   This hierarchical pin is a point of attachment for electrical connections between the hierarchical block and other objects on schematic page 1.
2. Place a hierarchical port named X on schematic page 3.
   This hierarchical port is a point of attachment for electrical connections between schematic page 3 and other schematic pages. It is connected by name to the hierarchical pin inside the hierarchical block on schematic page 1.



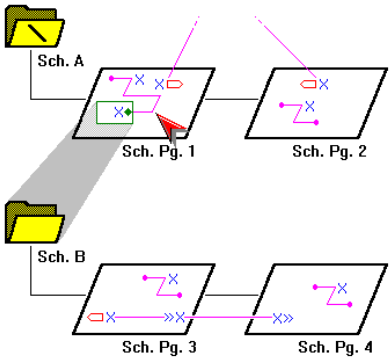
Hierarchical ports generally carry a net "up" through a hierarchy. In the root schematic folder, they usually represent external signals such as physical connectors on a PC board.
Note that hierarchical ports in schematic folder A are electrically connected by name, so any like-named connectivity objects on schematic pages 1 and 2 are part of a single net named Y. You could make either one (but not both) of these hierarchical ports an off-page connector without affecting the electrical connections.



To connect the schematic pages in schematic folder B, place an off-page connector named X on both schematic pages 3 and 4. Any like-named connectivity objects on schematic pages 3 and 4 are part of a single net named X.



To connect the X and Y nets, it is not enough simply to rename one set of objects, as shown here. Again, the hierarchical pin does not bring the "green" net X out of the hierarchical block and onto the schematic page.



When you physically connect any part of the "blue" net X to the hierarchical pin inside the hierarchical block, the nets are joined.

Using intersheet references

Intersheet references indicate the source and destination of schematic page and schematic signals in your design, making it easier to trace signals and find errors in the electrical connectivity of your design. Compare this to off-page connectors, which are used for signals between schematic pages within the same schematic, or hierarchical ports, which are used for signals between schematics.
An intersheet reference for an input signal indicates all the schematic pages from which the signal originates; an intersheet reference for an output signal indicates all the schematic pages to which the signal travels.
For example, an output hierarchical port with intersheet references 35, 42, and 61 indicate that the signal goes to schematic pages 35, 42, and 61.

> Intersheet references work with a flat design, simple hierarchy or complex hierarchy.

In this Section:

Creating Intersheet references

To add intersheet reference s

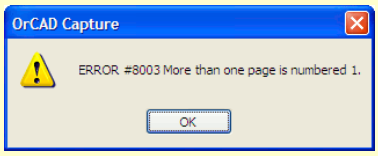
1. From the Tools menu, choose Intersheet References.
   The Intersheet References dialog box displays.
   You can also access this dialog box by choosing the Annotate option on the Tools menu. Then in the Annotate dialog box select the Add Intersheet References option and click OK.
2. Select the Place On Off-Page Connectors option if you want the intersheet references placed on off-page connectors.
3. Verify that the value used for X Offset is correct. This option adjusts the horizontal spacing between the port name and the intersheet reference. Increase the value to increase the separation.

4. Select or clear the option in the Port Type Match Matrix group box as required to specify how port types are to match each other when creating intersheet references.
5. Click OK. Capture performs error checking on your design while it generates intersheet references.
6. To generate a report of the interesheet references of the selected design, select the View Report option and specify the name of a CSV file in which you want to create the report.

> Use the Intersheet references command, with the option to generate the .csv output file. This file will be available in the Outputs folder in the Project manager.

7. If any design errors are encountered during the creation of intersheet references, a message box appears asking if you want to view the errors or warnings in the session log. Choose either the Yes button or the No button, as appropriate. If no design errors are encountered during report creation, the intersheet references are added to your design.

> When running the Intersheet References command you encounter the following error. This is caused if the page numbers (as defined by the Page Number property of the page title block) are duplicated. You can resolve this issue either by re-annotating your design (choose the Annotate command from the Tools menu) or by manually editing the Page Numbers on the title blocks on the pages in your design.



To remove intersheet references

1. From the Tools menu, choose Annotate. The Annotate dialog box appears.
2. Select Delete Intersheet References, and click OK. All intersheet references are removed from the design.

Guidelines for Creating Intersheet References

When generating intersheet references for a design, Capture uses a number of rules. The following set of guidelines will help you understand the details of how the intersheet references are generated and the points you need to keep in mind when running this command.

1. **Same name Off-page connector and Port**
   If an off-page connector and a port on the same page have the same name, no intersheet reference will be generated if another off-page connector or port of this name does not exist on another page. Also, two warning messages will be logged for the two un-connected signals.
2. **Hanging Off-page Connectors**
   If a page contains two off-page connectors that are not connected to any pin (know as hanging off-page connectors), the intersheet references for these two off-page connector will be generated. However, if these hanging off-page connectors exist on different pages, the Intersheet references will not be generated. Also, two warning messages will be generated for the two un-connected signals.
3. **Placing Intersheet References**
   When placing an intersheet reference, if the net symbol is a left port or off-page connector, the IREF will be displayed to the right of the graphical lines. If the net symbol is a right port or off-page connector, the IREF will be displayed to the left of the graphical lines.
4. **Zone Information for Port**
   The zone information will not be displayed for a port that is inherited from a hierarchical pin. However, the zone information will be displayed if that signal is routed to another page.
5. **Multiple Occurrence Port**
   If a port has multiple occurrences, then all the pages are appended to the IREF property for each occurrence of the port.
6. **Bus Intersheet References**
   If a design has two buses and the hierarchical ports on these bus are EN2[7..0] and EN2[0..7], then one a intersheet reference will be displayed for each port, showing connectivity between the two buses. However, a warning will be displayed in the session log stating that the hierarchical port EN[7..0] appears twice on the same page.
   If an off-page connector with the name A[15:0] is connected to a hierarchical port with the pin name A[0:3], then pin A0 on the hierarchical block is mapped to A15. This implies that the flatnet name for this pin is A15 and not A0.
   If the off-page connector name is A[15:0], then intersheet references are generated on pages that have flatnets with names A0, A1 through A15.
   If an off-page bus is placed on the top page of a design and another off-page bus on

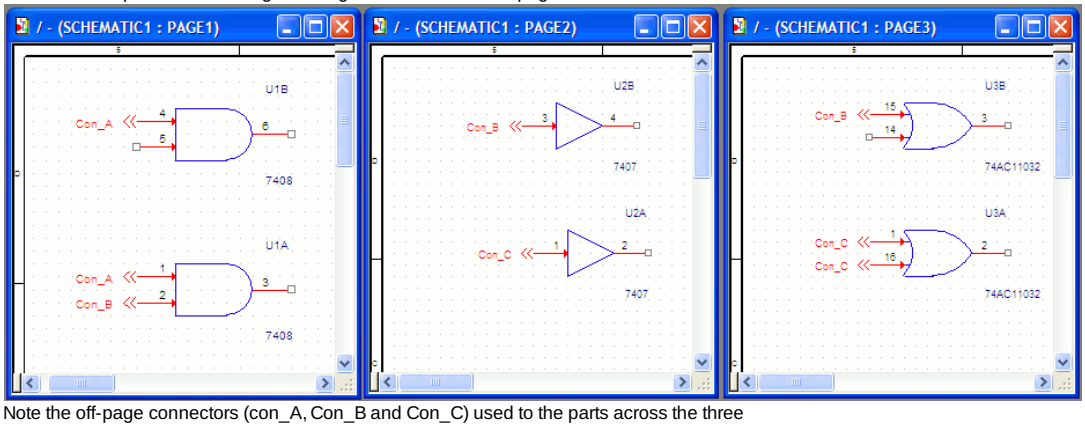
another page of the design, then the intersheet references will be generated only if the appropriate bus bits are specified on the off-page defined in the latter page of the design. To verify this, you need to open the property editor for each bus bit and check the schematic net and flatnet name defined for the bit.

7. **Grid Display Option**
   If an off-page connector is connected to a hierarchical pin, in the Intersheet References grid display option, the grid information is not displayed with the port.
8. **Port Page Number Append**
   The port page number will be added to an off-page connector intersheet reference only when the net connected to the off-page is connected to a part pin and a is port connected to the same net on the other page.
9. **Generate Intersheet References**
   The following matrix provides a snapshot of the scenarios for Intersheet References:

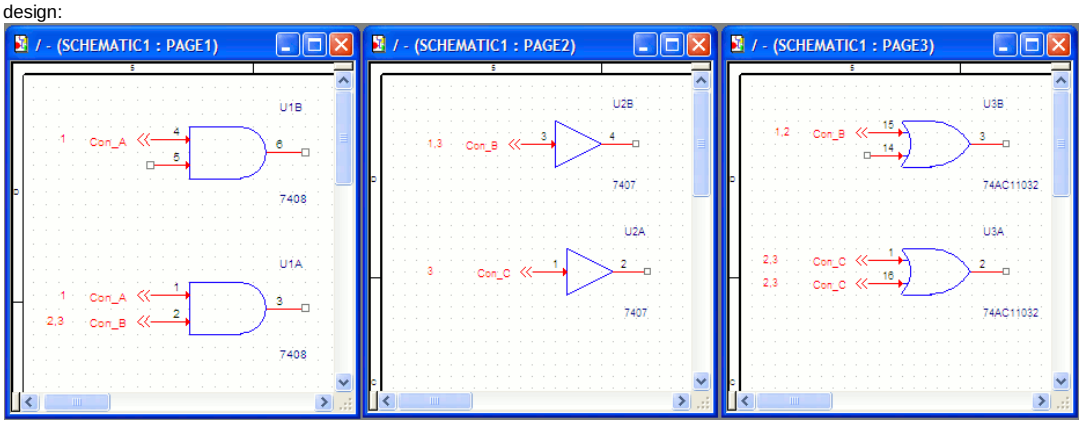
Scenario	IREF Generated
Port is a bus, bus bit is a port or net name equals to bus bit	YES
Port is a bus, bus bit is an off-page connector or net name equals to bus bit	NO
Off-Page connector is a bus, bit is an off-page connector or net name equals to bus bit	NO
Port is a bus and a bus with same name exists	YES
Off-page connector is bus and a bus with same name exists	YES
Same port name exists on same page	YES
Same off-page connector name exists on the same page	YES

Intersheet references in a flat design

For every off-page connector on the pages of a flat design, the intersheet reference of its port will be attached to that connector.
Take the example of the following flat design that contains three pages.



Note the off-page connectors (con_A, Con_B and Con_C) used to the parts across the three pages.
Notice the output if you create intersheet references to trace the signals across the pages of this design:



Con_A:

Since the two connectors lie on page 1, the intersheet references are defined as 1 for both connectors.

Con_B

On page 1 the reference is defined as 2,3. This implies that the signals for this connector exist on page 2 and page 3.
Similarly, if you see the Con_B reference on page 2 is 1, 3 and the reference on page 3 is 1,2.

Con_C

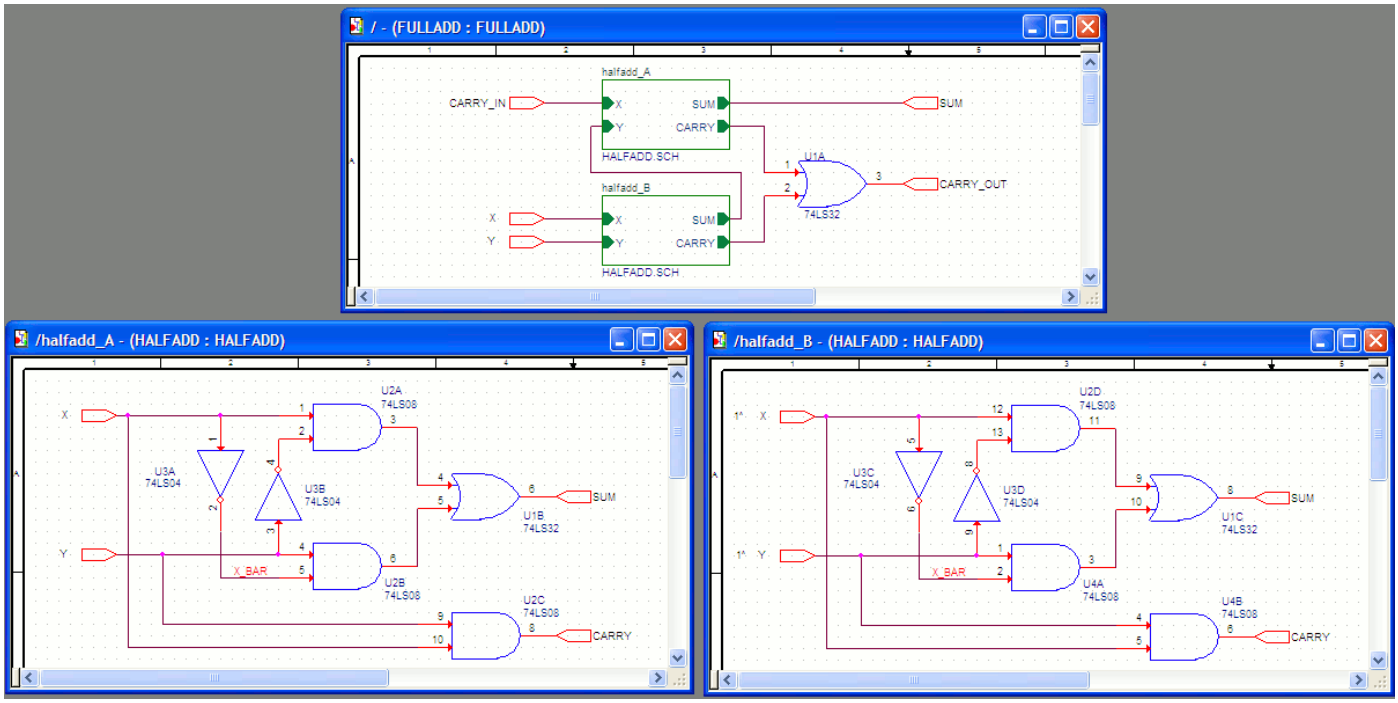
The page 2 reference for Con_B is 3 since the signals for this exist on page 3. Now the page 3 reference for both Con_C connectors on page 3 show as 2,3. This implies a signal exists on page 2. Also, another signal for this connector exists on page 3 itself.

> The page numbers defined in an intersheet reference are derived from the page numbers defined in the page title block.

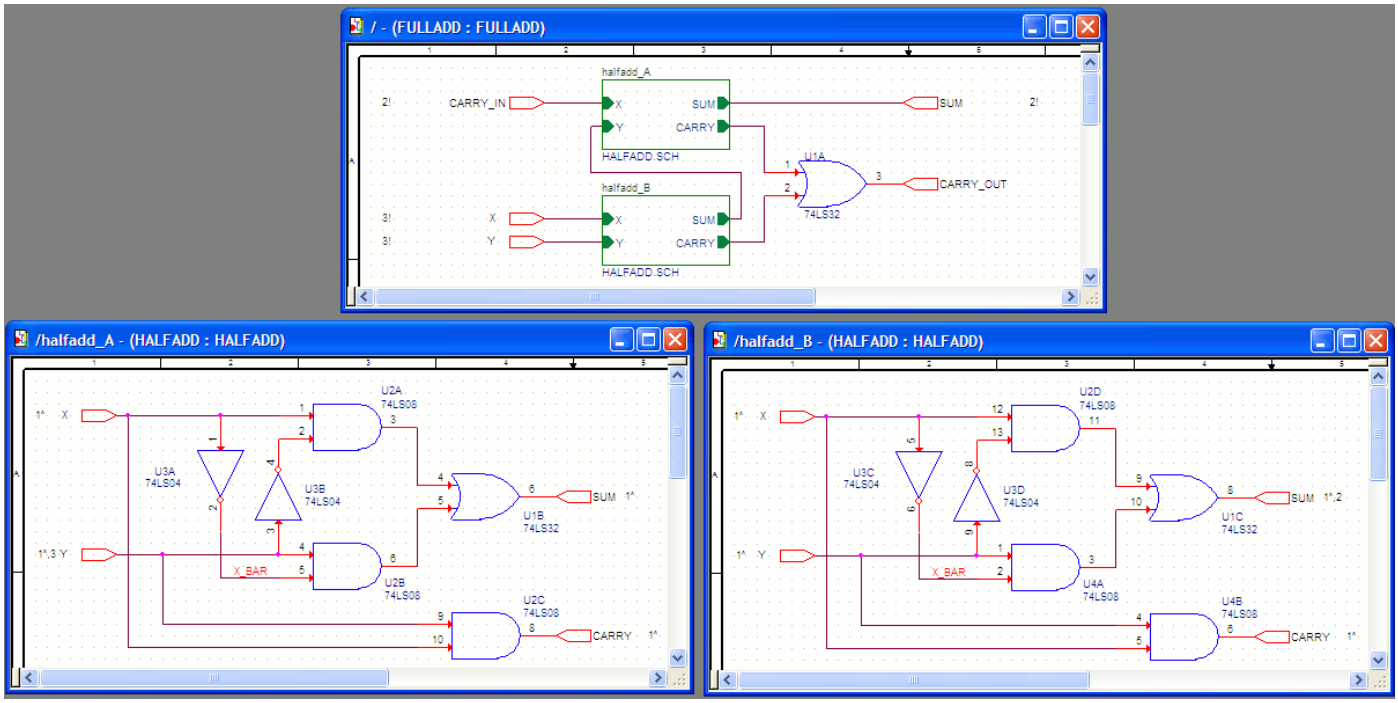
In a design containing a large number of pages and signals, you use the signal navigation facility in Capture to navigate connected signals across pages on your design.

Intersheet references in a hierarchical design

Take the example of the FULLADDER design that contains two occurrences of the HALFADD hierarchical blocks.



Note that the hierarchical blocks on the FULLADD page are connected to the two HALFADD pages via hierarchical ports.
Notice the output if you create intersheet references to trace the signals across the pages of this design:



CARRY_IN (FULLADD)

The intersheet reference for the CARRY_IN port on the FULLADD page is defined as 2!. This implies that the signal for this port is on page two of the design (halfadd_A). Also, the exclamation (!) symbol implies that the connected signal exists one level lower in the hierarchy.

X (FULLADD)

The reference for the X port on the FULLADD page is defined as 3!. This indicates that the connecting signal is on page 3 (halfadd_A), and the exclamation symbol implies that the connecting signal exists one level lower in the hierarchy.

X (halfadd_A)

The intersheet reference for the X port on the halfadd_A page is defined as 1^. This indicates that the connected signal exists on page 1 (FULLADD). Also, the caret (^) symbol implies that the connecting signal exists one level higher in the hierarchy.

Y (halfadd_A)

The reference for the Y port on the halfadd_A page *1^,3* defines that one connecting signal exists on page 1 (FULLADD) and the caret symbol indicates that this signal exists one level higher in the hierarchy. The second connecting symbol exists on page 3 (halfadd_B). Similarly, the intersheet reference for the SUM port on the halfadd_B page is 1^,2.
In a complex design containing a large number of pages and signals, you use signal navigation facility in Capture to navigate connected signals across pages on your design.

> A port must be connected to a pin of a hierarchical block to ensure that the ^ and ! symbols will be displayed with the intersheet references for the port. These symbols will not be displayed if the port is connected to a part that is not primitive.

Reporting Intersheet References

When you create the intersheet references for a design, the Intersheet References dialog box contains an option to create a report file (in the .csv format). This report file provides a complete list of all the connected signals on your design.
The following report file sample is generated from the FULLADD design.

	A	B	C	D	E	F	G	H	I	J
1	Name	Type	Page	Page Number	Schematic	PartPin	LocationX	LocationY	Zone	IREF
2	SUM	Output	FULLADD	1	FULLADD	halfadd_A.SUM	560	240	5B	5B2!
3	SUM	Output	HALFADD	2	HALFADD	U1B.6	540	200	5B	1^
4	X	Input	FULLADD	1	FULLADD	halfadd_B.X	220	340	2B	1A3!
5	X	Input	HALFADD	3	HALFADD	U2D.12,U3C.5,U4B.5	60	130	1A	1^
6	Y	Input	FULLADD	1	FULLADD	halfadd_B.Y	220	360	2C	1B3!
7	Y	Input	HALFADD	3	HALFADD	U3D.9,U4A.1,U4B.4	60	250	1B	1^
8	CARRY_IN	Input	FULLADD	1	FULLADD	halfadd_A.X	220	240	2B	1A2!
9	X	Input	HALFADD	2	HALFADD	U2A.1,U3A.1,U2C.10	60	130	1A	1^
10	CARRY	Output	HALFADD	2	HALFADD	U2C.8	520	320	5B	1^
11	CARRY	Output	HALFADD	3	HALFADD	U4B.6	520	320	5B	1^
12	CARRY_OUT	Output	FULLADD	1	FULLADD	U1A.3	560	310	5B	Property Not Present
13	SUM	Output	HALFADD	3	HALFADD	U1C.8	540	200	5B	1^,1B2
14	Y	Input	HALFADD	2	HALFADD	U3B.3,U2B.4,U2C.9	60	250	1B	1^,5B3

As an example, see the following two selected rows from the report:

4	X	Input	FULLADD	1	FULLADD	halfadd_B.X	220	340	2B	1A3!
5	X	Input	HALFADD	3	HALFADD	U2D.12,U3C.5,U4B.5	60	130	1A	1^

Name	name of the port or off-page connector
Type	signal type
Page	the page on which the port or off-page connector exists
Page Number	defined by the Page Number property of the title block on that respective page
Schematic	schematic folder containing the specific pages
PartPin	the part and pin combination connected to the port or off-page connector
LocationX	the X-axis location on the schematic page grid for the port or off-page connector
LocationY	the Y-axis location on the schematic page grid for the port or off-page connector
Zone	the zone location on the schematic page grid for the port or off-page connector
IREF	is the reference of the connected signal. Notice in the first item the reference is 1A3!. This means that the connected signal exists on the 1A zone of page 3 of the design and is one level lower in the hierarchy

After you run the Intersheet References command, with the option to generate a .csv output file, this file will be available in the Outputs folder in the Project manager.

Signal Navigation in Capture

You can use the signal navigation feature in Capture to navigate the connected signals on a design. This feature allows you to select a signal that you want to trace. Capture then browses for all the connected signals on the design. Finally, you can select and highlight the signals from the browse list.

To find and navigate the signals on a design

1. Select the off-page connector, hierarchical port, net or bus to find its connecting signals.
2. Right-click and choose the Signals option from the pop-up menu.
3. A browse list appears with all the signals that are connected to the currently selected signal.

Object ID	Name	Page	Page Number	Schematic	Part Pin
CARRY_IN(Port)	CARRY_IN	FULLADD	1	FULLADD	halfad...
halfadd_A/X(Port)	CARRY_IN	HALFADD	2	HALFADD	U2A.1...

4. Double-click on a signal in this list to navigate to the connected signal.

When you select to view the signals of a bus, the signal list contains the bus entries on the selected bus. However, the signal navigation does not connect to any bus with the same name on different pages. For example, if you choose a bus with the off-page connector defined as D[0..5] and the bus contains the bits D1 to D5, the list will display these bits. However it will not show any connectivity to another bus with the same off-page connector (D[0..5]) defined.

Working with nets

A net comprises the wires, buses, parts, and symbols that are logically connected via net names, net aliases, off-page connectors, and hierarchical ports.

In this section:

- Assigning net aliases
- Net operations
- Tracing a net

Assigning net aliases

A net is not required to have an alias, but by using an alias, you can establish connectivity.

Within a schematic page, a net with an alias is connected to any net with the same alias, or to any off-page connector, hierarchical port, or global pin with the same name.
A net alias differs from a net name in that a net can have numerous aliases, but it can have only one name. When the Create Netlist tool resolves the conflict between the various aliases attached to a net, the net alias has the highest priority; so by assigning a net name, you can determine the final name of your net.
When you place a wire, it is assigned a system-generated name. When you place a net alias on the wire, the system-generated name is replaced by the alias.
A net alias is visible at the location where you place the alias, You may find it useful to label the net throughout your project.

Anchor | 1102975 ***To create a net alias***

1. From the Place menu, choose the Net Alias command.
2. Enter the net alias text in the dialog box that appears, then click OK. A rectangle representing the net alias is attached to the pointer.
3. Use the mouse to move the net alias and click the left mouse button on the wire to place the net alias. The net alias appears in the selection color. The tip of the pointer must be touching the net for you to place the net alias.
4. Select the selection tool to dismiss the net alias tool. The alias is added to the alias list for the net.

Shortcut

Tool palette: 

To assign a net name

1. Select the wire.
2. From the Edit menu, choose the Properties command. The Property editor window opens to the Schematic Nets tab.
3. Change the entry in the Name column to one of the existing net aliases and close the property editor.

To edit a net alias

1. Select the net alias.
2. From the Edit menu, choose the Properties command.
3. In the dialog box that appears, you can change the color, the font, the rotation and the alias itself.
4. Click OK.

To move net alias text

1. Select the net alias text on a net. A handler appears around the net alias text.
2. Drag the net alias text handler to the location (on the same net) where you want to place it. Note: You can not move a net alias outside of a net segment.
3. Drop the net alias handler. The net alias text appears in the new location on the net.

To display the net alias at multiple locations

1. Select the portion of net where you want the alias to be visible.
2. From the Edit menu, choose the Properties command. The property editor appears.
3. Click the New button. The Add New Property dialog box appears.
4. Assign a name, such as NAME1, to the new property. Do not assign a value at this time. Click OK to dismiss the Add New Property dialog box.
5. Select the cell of the new property, and click the Display button.
6. Select the Value Only Display Format option, and click OK.
7. Click Apply, and then close the property editor.
8. Repeat steps 1 through 7 for each location where you want the alias to appear, assigning another property name (NAME2, NAME3 . . .) at each location.
9. Use the Update Properties tool (see To update part or net properties) to assign the net's alias as the value to the properties NAME1, NAME2, NAME3 . . .

> Capture preserves the case of part names and net names, but ignores the case when comparing names for electrical connection. That means you may use uppercase or lowercase letters as you wish, but you need not remember the case.

Net operations

A net is one or more wires that are physically connected or that have been connected by a net alias, a hierarchical port, or an off-page connector. In addition, all like-named power pins, power objects, and attached wires throughout the project constitute a net, unless they have been isolated.
You can edit a discrete wire, a wire segment, or you can edit the net as a whole. You can also easily edit or add to the properties of multiple nets. See To update part or net properties for more information.

Find and Select a Net

1. In the project manager, select the schematic folder or schematic pages that you wish to search.
2. From the Edit menu, choose the Browse command, and then select the Nets command from the pull right menu. The browse window displays a list of all nets by name and by alias.
   OR
   From the Edit menu, choose the Browse command, and then select the Flat Netlist command from the pull right menu. The browse window displays a list of the nets that appear in netlists.
   OR
   From the Edit menu, choose the Find command, and then type an asterisk (*) in the Text to Search field and click the Find button. The Find window displays a list of all the nets by name and by alias.
3. From the list, double-click on the name of the desired net. The schematic page editor opens with the net appearing in the selection color.
4. Right-click to display the pop-up menu.
5. From the menu, choose the Select Entire Net command.
   All net segments on the active page appear in the selection color.

> The Select Entire Net command is restricted to the active schematic page—it does not follow hierarchical blocks, hierarchical ports, or off-page connectors across schematic folders or schematic pages. To do this you can use the Signal Navigation in Capture or by Tracing a net.

The status bar displays the net name of a selected net or wire.

Remember that nets on a schematic page are electrically connected by name, by alias, or by connection to a named hierarchical port or off-page connector.

Edit Net Properties

1. Select a segment on the net.
2. From the Edit menu, choose the Properties command. The property editor appears.
3. Change the filter to Capture.
4. Use the property editor to edit, add, or remove properties as necessary.

Delete a Net

1. Select the net.
2. Right-click on the net.
3. From the context-sensitive menu, select Select Entire Net.
4. From the Edit menu, choose Delete.

Tracing a net

When you need to trace a net, you may not know all the net aliases or how many schematic pages the net touches. Using Capture, you can overcome these problems and find every portion of the net. You will need to start with a portion of the net selected in the schematic page editor, or with a net name, an off-page connector name, or a hierarchical port name. If you start with a name, use thePower Pins command of the project manager to locate a portion of the net.
The actions involved in tracing a net can be done in any order. Typically, you locate a part of the net, highlight all portions of the net on the same schematic page, follow the net onto other schematic pages in the same schematic folder, and then follow the net into other schematic folders.

> You can trace a signal in a design by using the Signal Navigation in Capture.

Find a Net Using a Name

1. In the project manager, select the schematic folder that holds the name. If you do not know which schematic folder holds this portion of the net, select all schematic folders (press Ctrl while you click on a schematic folder to add it to the selection set).
2. From the Edit menu, choose the Find command. The Find toolbar displays.
3. In the Text to Search field, enter the name, with wildcards if you wish, in the Search options drop-down list and specify that this is the name of a net, an off-page connector, or a hierarchical port.
4. Click the Search button to initiate the search. A list of all objects that match your search criteria appears in the Find window.
5. Double-click on an item in the Find window. The schematic page editor opens with the net or off-page connector or hierarchical port selected.

Locate and Highlight all Wires of a Net on a Single Page

1. Click over a wire of the net to select the wire.
2. Right-click to display the pop-up menu.
3. From the menu, choose the Select Entire Net command.
   All wires of the net appear in the selection color. You may need to zoom out to see the

entire net.

> The Select Entire Net command is restricted to the active schematic page—it doesn't follow hierarchical blocks, hierarchical ports, or off-page connectors across schematic folder or schematic pages.

Nets on a schematic page are electrically connected by name, by alias, or by connection to a named hierarchical port or off-page connector.

Trace a Net Across Pages of a Schematic Folder

1. Locate and highlight all wires of the net on one page.
2. Scan the selected net for off-page connectors and for hierarchical ports not inside a hierarchical block. For each off-page connector or hierarchical port,
3. Note the name.
4. In the Project manager, select the current schematic folder.
5. From the Edit menu, choose the Find command.
6. The Find toolbar displays with the Find field selected.
7. In the Text to Search field, enter the name, select Off-Page Connectors, then click the Find button. The Find window displays a list of off-page connectors with the specified name.
8. For each entry in the Find window, double-click on the entry. The schematic page editor opens with the off-page connector appearing in the selection color.
9. Repeat step 2, selecting Hierarchical Ports in the Find pop-up list on the Find toolbar.

To Trace a Net Between Schematic Folders

1. Locate and highlight all the wires of the net on one page.
2. Scan the selected net for hierarchical ports not inside a hierarchical block. For each port:
3. Note the name.
4. In the Project manager, select all schematic folders except the active one.
5. From the Edit menu, choose the Find command.
6. In the Text to Search field, enter the name, select Hierarchical Ports, then click Find. The Find window displays a list of hierarchical ports with the specified name.
7. For each entry in the Find window, double-click on the entry. The schematic page editor opens with the hierarchical port appearing in the selection color.

As you place buses and wires:

- A bus and a wire can be connected only by name.
  If you begin or end a bus segment on a segment of a wire, a vertex is added to the wire and a junction appears, but the bus and wire are not electrically connected.
  If you begin or end a wire segment on a segment of a bus, a vertex is added to the bus and a junction appears, but the wire and bus are not electrically connected.
- Two buses or two wires can be connected physically.
  If you begin or end a bus segment on a segment of another bus, a vertex is added to the second bus, and a junction appears—the buses are connected.
  If you begin or end a wire segment on a segment of another wire, a vertex is added to the second wire, and a junction appears—the wires are connected.

Capture preserves the case of part names and net names, but ignores the case when comparing names for electrical connection. That means you may use uppercase or lowercase letters as you wish, but you need not remember the case.

When you attach a schematic folder to a part or hierarchical block, you can specify a full path and file name in the Library text box. So, although you can specify a library that has not been saved, you should not try to descend into the attached schematic folder until the library that contains the schematic folder has been saved.

If you do not specify a full path and filename in the Library field, Capture expects to find the attached schematic folder in the same design as the part of hierarchical block to which it is attached. If the specified schematic folder does not exist in either the design or library, Capture creates the schematic folder when you descend the hierarchy on the part or hierarchical block.

For compatibility with future versions of Windows, Capture preserves the case of the path and filename as you specify them in the Library field.

Using NetGroups

A NetGroup is a collection of nets. The nets in a NetGroup can be scalar, vector or a combination of both. You can create a NetGroup that consists only of nets (like a bus). You can also create a NetGroup that consists of nets (scalar and / or vector), consists of buses and consists of other NetGroups.

In this section:

Introducing NetGroups

By definition, a NetGroup is a completely heterogeneous collection of nets. Unlike a bus, which is a homogeneous collection of nets (scalar or vector), a NetGroup provides a greater flexibility in grouping nets together.
For example, you can collect together a large number of signals on a page of a schematic into a NetGroup. You create an off-page connector and then connect all the signals on the NetGroup to the signals on another page.

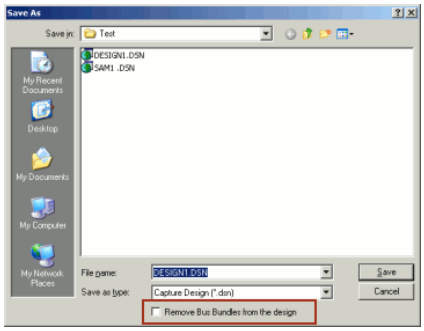
> While a NetGroup provides greater flexibility for net grouping than a bus, there will be many situations where a bus will be a sufficient implementation of the required functionary. So care should be taken not to assume that the NetGroup completely overrides the functionality and value of a bus.

To create a NetGroup, you use the following objects (that are NetGroup-aware):

- Off-page connector
- Port
- Bus Wire
- Pin
- Global

When you create a NetGroup and define a property for the NetGroup, this property is inherited by all the constituent nets of the NetGroup.
You can create two types of NetGroups, a **named** NetGroup or an **unnamed** NetGroup.

When you add a NetGroup to a design, the design database version is upgraded to v16.5. This implies that the design can now not be opened in any version of Capture prior to v16.5. However, you can choose to remove all the NetGroups from a design (if you need to open the design in a previous version), by choosing Save As from the File menu and clicking the Remove NetGroups from the design checkbox.

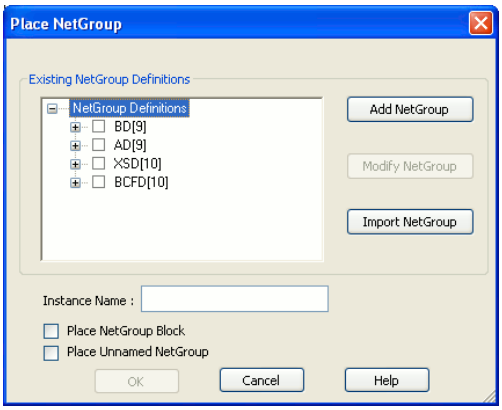


Named NetGroup

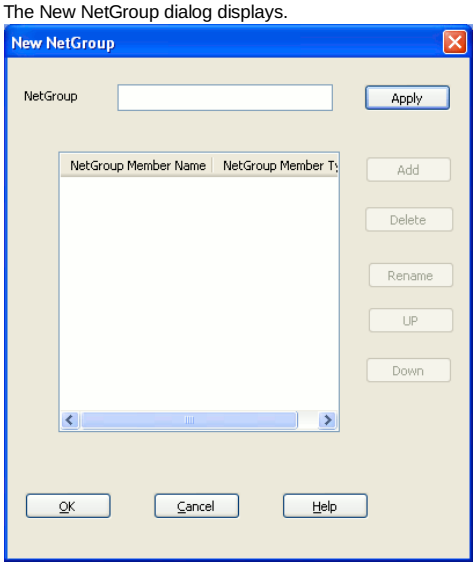
When you create a named NetGroup, you need to specify the associated NetGroup definition and then specify a name for the NetGroup. This type of NetGroup is persistent and can be instantiated across a design. The NetGroup can also be exported as a library and then instantiated and used in other designs. However, if you need to create a NetGroup for one-time use, you create an unnamed NetGroup. For details on unnamed NetGroups, see Unnamed NetGroup.

To create a named NetGroup

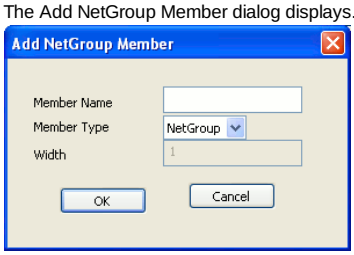
1. Choose NetGroup from the Place menu in Capture.
   The Place / Create NetGroup dialog displays.
   You use this dialog box to build associated NetGroup definitions that you can use anywhere in your design.



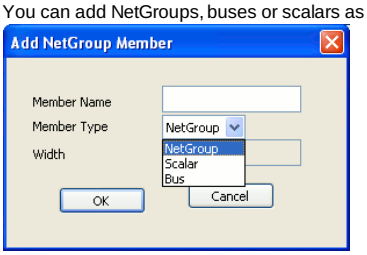
2. To specify a new associated NetGroup definition, click the Add NetGroup button.
   The New NetGroup dialog displays.



3. In the NetGroup Name text box, enter the name of the NetGroup and click Apply.
4. To add a new NetGroup member, click the Add button.
   Note:The Add button is disabled until you specify the NetGroup name and click Apply.
   The Add NetGroup Member dialog displays.



   You can add NetGroups, buses or scalars as members of a NetGroup.



   If you add a NetGroup or a bus, you need to also specify the width of the member. For example, if you add a NetGroup named AD that contains five signals, you will specify the name as AD[0..4] or AD[0-9]. The same holds true for buses.
   When you add a bus or a NetGroup as a member of a NetGroup definition, you need to specify the bus or NetGroup member using the normenclature [LSB..MSB]. The **least significant byte** followed by the **most significant byte**.

If you add a NetGroup as a member to a NetGroup, the new NetGroup is also available for use as

a NetGroup on its own. So in the Place NetGroup dialog, you will see the new associated NetGroup definition.

In the New NetGroup dialog, you can also **rename** or **delete** existing NetGroup memebrs. You can also **move** the existing memebrs **up** and **down**.

You move the positions of the members of a NetGroup, up or down to specify the horizontal position they will display in when you place the NetGroup on a page.

To rename a NetGroup member

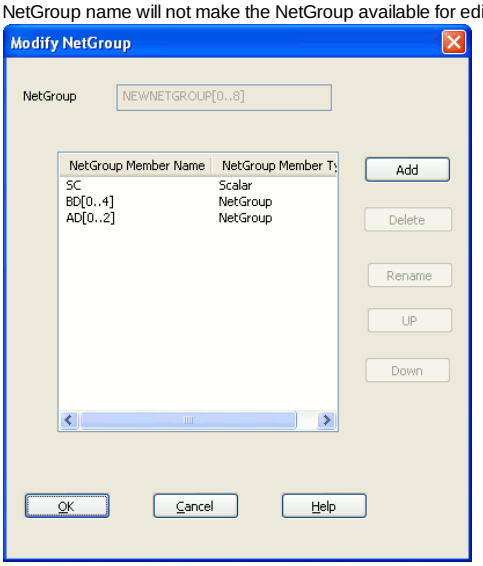
1. In the New NetGroup or Modify NetGroup dialog, select the NetGroup member to rename.
2. Click the Rename button.
   The Rename NetGroup Member dialog displays.
3. Type a new name for the member and click OK.

When you rename a NetGroup member that is either a NetGroup or a bus, you need to ensure that you only change the name of the member. This implies, in the rename procedure, you cannot change the width of the member.

When you rename a NetGroup member that is a NetGroup, you need to rename the NetGroup to the name of an existing NetGroup not already contained within the current NetGroup.

To modify a named associated NetGroup definition

1. To open the NetGroup dialog, choose NetGroup from the Place menu.
2. Click the check-mark next to the NetGroup you want to modify and click the Modify button.
   Note: You need to click the check-mark next to the name of the NetGroup. Selecting the NetGroup name will not make the NetGroup available for edit.



When you modify an associated NetGroup definition, you can add, delete or rename members. You can also move the positions of the existing members up and down. However, you cannot change the name of the member.

To delete a named NetGroup

1. Choose NetGroup from the Place menu in Capture.
2. In the Place NetGroup dialog, choose the NetGroup to delete.
3. Click Delete NetGroup.

- You cannot delete a NetGroup that is used in the design.
- You cannot delete a NetGroup that is defined in another NetGroup.
- If you delete a NetGroup (say, N1) that contains other NetGroups (say, SubN1 and SubN2)), the top level NetGroup (N1) is deleted. However, the contained NetGroups (SubN1 and SubN2) are not deleted.

Copy and paste NetGroups across designs

You can use the Capture Copy and Paste commands to copy a NetGroup from one design to another. This functionality is the same as for other schematic page objects.
When you copy a NetGroup to another design, the associated NetGroup definition is also copied. This means that if you open the NetGroup dialog, you will see the associated NetGroup definitions now available for use in the new design.
Also, if you copy and paste a NetGroup into another design and NetGroups of the same name exist in the destination NetGroup, the NetGroups in the destination design will not be overwritten.

Exporting and Importing Associated NetGroup Definitions

You can export the NetGroups defined in a design to an associated NetGroup definition xml file. These NetGroups can then be imported and used in any other design.

To export an Associated NetGroup to a Definition Xml

1. From the Place menu choose NetGroup
2. Open the NetGroup dialog
3. Click Export NetGroup.
4. In the Save As dialog, specify the name and destination of the NetGroup definition xml.

When you run the Export NetGroup command in Capture, all the named NetGroups in the current design are exported to the definition xml.

This implies the unnamed NetGroup definitions are not exported.

To import an Associated NetGroup Definition Xml

1. From the Place menu choose NetGroup.
2. Open the NetGroup dialog.
3. Click Import NetGroup.
   The Associated NetGroup Definitions xml files dialog box displays.
4. Enter the full path of the associated NetGroup definition xml file or browse to and select the associated NetGroup definition xml file and click Open.
   The NetGroups contained in the file are imported into the current design.

Placing a named NetGroup on a page

After you define a NetGroup, you can place the NetGroup on the page of the design.
You can place a named NetGroup on a page as:

- a block
- a hierarchical port
- an off-page connector

To place a named NetGroup as a block

1. To place a NetGroup as a block you need to go to the NetGroup dialog box. This box is open if you have just created a new associated NetGroup definition or you can open this from the NetGroup menu item on the Place menu.
   When placing a NetGroup on a page, you have the option to place the entire NetGroup or place only selected members of the NetGroup.
2. To place the entire NetGroup on the page, click the check box to the left of the NetGroup name.
   (To place only selected members of the NetGroup, expand the NetGroup node and click the check boxes to the left of the selected members.)
   Notice, when you click the NetGroup name check box or any one of the NetGroup member check boxes, the Name field is filled with the name of the NetGroup. This is the default name given to the named NetGroup instance, and you have the option to edit this instance name.
3. To place the NetGroup as a block, choose the Place NetGroup Block check box and click OK.
   The cursor changes to a crosshair icon.
4. Draw a block to contain the NetGroup. This is done the same way as when drawing a hierarchical block on a page.

When you place a NetGroup as a block on a schematic page, you have the option to synchronize this block (up or down) the same way you synchronize a hierarchical block in a design. For details about synchronizing a block, see Synchronize Up command and Synchronize Down command.

To place a named NetGroup as a hierarchical port

1. To place a named NetGroup as a hierarchical port, you need to go to the Place Hierarchical Port dialog.
   To open this dialog, choose the Hierarchical Port menu item from the Place menu.
2. Enter a symbol for the port.
3. To place the hierarchical port as a NetGroup port, choose the NetGroup Port check box.
4. From the drop-down list, choose the NetGroup.
5. Click OK and place the port on the page.

The look and feel of a NetGroup port is different from that of a hierarchical port.

Also, if you place the cursor over the NetGroup port, the tooltip displays the associated NetGroup definition.

To place a named NetGroup as an off-page connector

1. To place a named NetGroup as an off-page connector, you need go to the Place Off-Page Connector dialog box.
   To open this dialog box, choose the Off-Page Connector menu item from the Place menu.
2. Enter a symbol for the connector.

3. To place the off-page connector a NetGroup connector, choose the NetGroup Port check box.
4. From the drop-down list choose the NetGroup.
5. Click OK and place the connector on the page.

The look and feel of a NetGroup connector is different from an off-page connector.

Also, if you place the cursor over the NetGroup connector, the tooltip displays the associated NetGroup definition.

To add or remove pins from a netgroup

1. Select the NetGroup on the design.
2. Choose Add/Remove Pins on NetGroup Block from the pop-up menu.
3. Select the pins to add or remove selection from pins that you want to remove.
4. Click OK.
   You can click to add the pins.

To assign a NetGroup to a bus

1. Choose Place - Net Alias.
   The Place Net Alias dialog box appears.
2. In the Alias field, specify a name for the alias.
3. Select NetGroup Aware Aliases.
4. Select a NetGroup from the list. You can also edit to specify a new NetGroup name.
5. Click OK.
   The alias has the same width as the specified NetGroup.
6. Click on a Bus in the design to assign the NetGroup.

Unnamed NetGroup

An unnamed NetGroup allows you to create an associated NetGroup definition for one-time usage. This means that the associated NetGroup definition is built dynamically and the NetGroup cannot be further instantiated across the design.
A benefit of an unnamed NetGroup is that you first create a empty definition and then add signals, as required. While, you cannot instantiate the associated NetGroup definition elsewhere in your design (or page), you can, however, reference the NetGroup on other pages within the same schematic.
An unnamed NetGroup can contain a scalar or a bus. However, unlike a named NetGroup an unnamed NetGroup cannot contain another NetGroup.

To create an unnamed NetGroup

1. Choose NetGroup from the Place menu in Capture.
   The Place / Create NetGroup dialog displays.
2. To specify the NetGroup as unnamed, choose Place UnNamed NetGroup checkbox.
   The Instance Name field displays the default name for the NetGroup (@@UNNG). You can edit this name as required.
3. Click OK.
4. Draw a block to contain the unnamed NetGroup. This is done the same way as drawing a hierarchical block on a page.

> Because an unnamed NetGroup can be reference across a design, the name you specify must be unique across the pages of a design.

Placing an unnamed NetGroup as a block

- *Using the Place NetGroup dialog*
- *Using the Schematic page edit pop-up menu*
- *To add a scalar member to an unnamed NetGroup*
- *To add a bus member to an unnamed NetGroup*
- *To delete a member from an unnamed NetGroup*
- *To place an unnamed NetGroup as a hierarchical port*
- *To place an unnamed NetGroup as an off-page connector*

Using the Place NetGroup dialog

1. Choose NetGroup from the Place menu.
   The Place NetGroup dialog displays.
2. To place the NetGroup as a block, choose the Place Unnamed NetGroup Block check box and click OK.
   The cursor changes to a crosshair icon.
3. Draw a block to contain the NetGroup. This is done the same way as when drawing a hierarchical block on a page.

Using the Schematic page edit pop-up menu

1. Select the nets, NetGroups and buses on the page that you want to add to the unnamed NetGroup.
2. Right-click on the selection and choose Create UnNamed NetGroup.

The cursor changes to a cross-hair icon.

3. Draw the NetGroup on the page.
   The entry pins of the NetGroup define the input signals for the signals included in the NetGroup.
   When you place an unnamed NetGroup as a block on a page (Using the Place NetGroup dialog), the NetGroup is empty. You need to now add members (scalars or buses) to the NetGroup.

To add a scalar member to an unnamed NetGroup

1. Select the unnamed NetGroup.
2. From the Place menu choose the Hierarchical Pin menu item.
   The Place Hierarchical Pin dialog displays.
3. Enter a name for the scalar member.
4. Specify the width as scalar by selecting the Scalar radio button in the Width group.
5. Click OK.
   The pin is attached to the cursor.
6. Place the pin on one of the edges of the unnamed NetGroup.

> As you keep adding members to the NetGroup, the size of the NetGroup is dynamically increased. Similarly, if you delete a member from a NetGroup the size will reduce dynamically.

To add a bus member to an unnamed NetGroup

1. Select the unnamed NetGroup.
2. From the Place menu choose the Hierarchical Pin menu item.
   The Place Hierarchical Pin dialog box displays.
3. Enter a name for the bus member.
   Note: The name of the bus must also define the size of the bus.
4. Specify the width as bus by selecting the Bus radio button in the Width group.
5. Click OK.
   The pin is attached to the cursor.
6. Place the pin on one of the edges of the unnamed NetGroup.

To delete a member from an unnamed NetGroup

- Select the member to delete and press the Delete key.

> As you keep adding and deleting members of an unnamed NetGroup, the size will increase and decrease dynamically.
>
> In the case of an unnamed NetGroup, the order of the members depends on the order in which they are added to the NetGroup. Also, you can see the ordered list of members defined to the right of the NetGroup.

To place an unnamed NetGroup as a hierarchical port

1. To place an unnamed NetGroup as a hierarchical port, you need go to the Place Hierarchical Port dialog.
   To open this dialog, choose the Hierarchical Port menu item from the Place menu.
2. Enter a symbol for the port.
3. To place the hierarchical port as an unnamed NetGroup port, choose the Show UnNamed NetGroup check box.
4. From the drop-down list choose the unnamed NetGroup.
5. Click OK and place the port on the page.

> The look and feel of a NetGroup port is different from that of a hierarchical port. Also, if you place the cursor over the NetGroup port, the tooltip displays the associated NetGroup definition.

To place an unnamed NetGroup as an off-page connector

1. To place an unnamed NetGroup as an off-page connector, you need to go to the Place Off-Page Connector dialog.
   To open this dialog box, choose the Off-Page Connector menu item from the Place menu.
2. Enter a symbol for the connector.
3. To place the hierarchical port as an unnamed NetGroup port, choose the Show UnNamed NetGroup check box.
4. From the drop-down list choose the unnamed NetGroup.
5. Click OK and place the connector on the page.

> The look and feel of a NetGroup connector is different from that of an off-page connector. Also, if you place the cursor over the NetGroup connector, the tooltip displays the associated NetGroup definition.

To reorder pins in an unnamed NetGroup

1. Select the unnamed NetGroup on the design.
2. Choose Reorder pins for UnNamed NetGroup from the pop-up menu.
   The Reorder UnNamed NetGroup Pins dialog box appears.
3. Select any of the listed pins and click or Down to change the order.
4. Click OK.

NetGroup Reuse (Hierarchical Part Creation)

You can reuse a NetGroup in other designs by creating a hierarchical part of the NetGroup and then instantiating the part in other designs.

To create a hierarchical part of a NetGroup

1. Open a design (or create a new design).
2. Create the NetGroup that you need to reuse.
   You can create either a named or an unnamed NetGroup.
3. Place the NetGroup as a hierarchical port on a page of the design.
   For details, see *Named NetGroup* or *Unnamed NetGroup*.
4. Save the design.
5. Go to the Generate Part dialog (from the Tools menu choose Generate Part).
6. In the Netlist/Source file field, enter the full path for the design. You can also browse to the design file.
7. If required, in the Netlist/Source file type drop-down list, choose the Capture Schematic / Design option.
8. if required, you can change the name and destination of the library (OLB) containing the hierarchical part.
9. Click OK.
   The Split Part Section Input Spreadsheet displays.
10. Click Save.

The part is created in the library. Also, you can view the OLB file and the part in the Outputs folder in the Project manager.
If you double-click the part in the Project manager, the Part editor displays. You can also make changes to the part.
The NetGroup contained in this part is now available for use in any design.

Components of a NetGroup Block

When you place a NetGroup block on a page, the block consists of a number of different parts and definitions that are different from a hierarchical block. The parts help in using the NetGroup for connectivity. These definitions help to understand the constituent members of the NetGroup.

NetGroup Entry/Exit Pin

When you define a NetGroup, you specify members of the NetGroup. You will then need to connect signals on your page to these members. On a NetGroup block these members appear as ports and are referred to as **NetGroup pins**. The nomenclature for a NetGroup pin is <NetGroupInstanceName>.<MemberName>.

NetGroup Pin

The benefit of a NetGroup is that you do not need to create as many exit points as the number of entry points. When you define the NetGroup, one exit point is created that holds the signals for all the entry points. This exit point is referred to as the **NetGroup entry pin**.

Note: The NetGroup port displays the NetGroup instance name and the size of the NetGroup instance.

NetGroup Wire

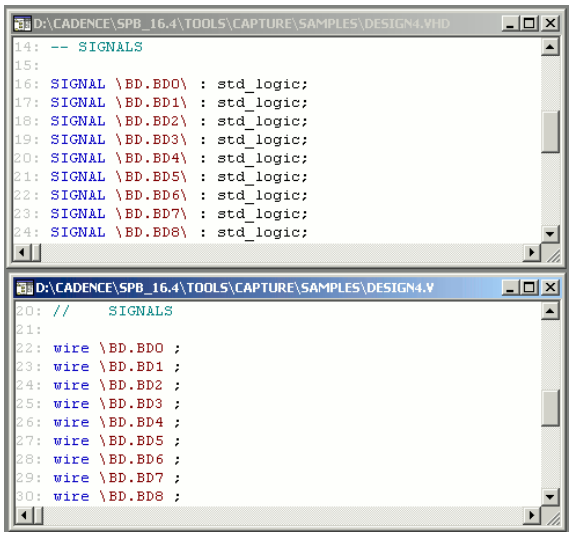
When you connect the entry / exit pins of the NetGroup to a wire, you are effectively connecting all the signals that constitute the NetGroup into one wire. This wire is known as the **NetGroup wire**.

Netlising NetGroup Designs

When you netlist a design, the signals in the design are mapped on the output netlilst.
A NetGroup is a hetrogeneous group of nets and hence the NetGroup as it is, cannot be translated as a NetGroup onto a netlist. The Capture netlisting command netlists the signals in a NetGroup by extracting out the signals when creating the netlist.
However, when the signal names display on the netlist, the names must also contain the name of the NetGroup. This prevents the possibility of duplicate signal names on the netlist. To define a NetGroup-signal name combination on the netlist, Capture uses <NetGroup Name><Separator><Signal Name>.

> The default NetGroup-signal combination, <NetGroup Name>.<Signal Name>, uses the dot notation.

For example, if you create a VHDL or Verilog netlist out of a design that contains NetGroups, the output uses the dot notation to handle the signals contained in the NetGroup.



Other Netlists

While most of the netlists generated through the Capture netlist command support the dot notation to signify NetGroup signals, some netlisters (available in the Other Netlists tab of the Create Netlist dialog) do no support the dot in a net name. To handle this, Capture provides a TCL script (**capCorrectNetnamesONL.tcl**) that defines an alternative separator in the netlist depending on the netlister.

Formatter	Name Separator
orVstmodel.dll	_ (underscore)
orOhdlnet.dll	_ (underscore)
orPcadnlt.dll	_ (underscore)
orEdif.dll	_ (underscore)
orCbds.dll	- (hyphen)
orCalay90.dll	- (hyphen)
orCalay.dll	- (hyphen)

This TCL script runs during the netlisting procedure. This implies that if you have a custom netlister not included in Capture, and the formatter does not support the default dot separator you can update this script to specifiy an alternative separator.

NetGroup Connectivity

As the name suggests, a NetGroup allows you to group together a heterogeneous group of signals. You can use this feature to easily connect a large number of signals on a page, across pages in a design and even across a hierarchy. This section describes NetGroup connectivity using four scenarios. Two scenarios for named NetGroups and two scenarios for un-named NetGroups.
To connect signals on the same page to a NetGroup, you can use the Auto-connect to NetGroup feature of Capture . For details, see Auto-Wire to NetGroup.
If you short buses and NetGroups together, the order of preference depends on factors like the width of the bus or NetGroup. This preference defines the resultant object (bus or NetGroup), Winning bus, and the flat nets generated out of the short. For details (covering a set of scenarios), see Net Generation Scenarios.

In this section:

- Named NetGroup Connectivity
- Un-named NetGroup connectivity

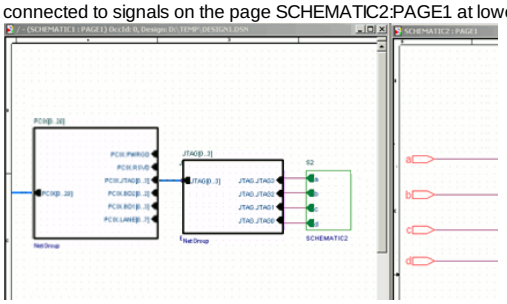
Named NetGroup Connectivity

Using named NetGroups you can connect signals across pages in a hierarchical design. You can also use named NetGroups to connect signals across a page at the same level of a hierarchical design or across pages in a flat design.

Using a Named NetGroup to connect signals across a hierarchy

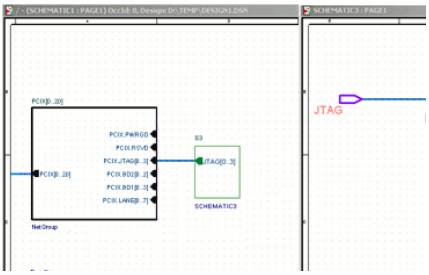
A NetGroup enables the connectivity of signals across the levels of a hierarchical design.

Scenario 1
In this example, the signals at the top level (SCHEMATIC1:PAGE1) of the design need to be connected to signals on the page SCHEMATIC2:PAGE1 at lower levels of the hierarchy.



1. The signals on SCHEMATIC1:PAGE1 are first tapped into a large NetGroup, PCIX. This NetGroup contains the signals of this page that needs to be connected to pages across the hierarchy.
   Four signals (a, b, c & d) need to be connected from the top level to the signals on SCHEMATIC2:PAGE1.
2. These four signals are first placed in a NetGroup JTAG that is then placed, as a member, in the large NetGroup PCIX.
3. Next, the JTAG NetGroup is placed separately onto the SCHEMATIC1:PAGE1 page.
4. The PCIX.JTAG NetGroup port is connected to the separate JTAG NetGroup.
5. Each NetGroup port of the JTAG NetGroup is then connected to the corresponding ports, a, b, c & d of the hierarchical block of SCHEMATIC2.
6. Finally, on the page SCHEMATIC2:PAGE1, four off-page connectors, a, b, c & d, are created to tap out the corresponding signals from SCHEMATIC1:PAGE1.

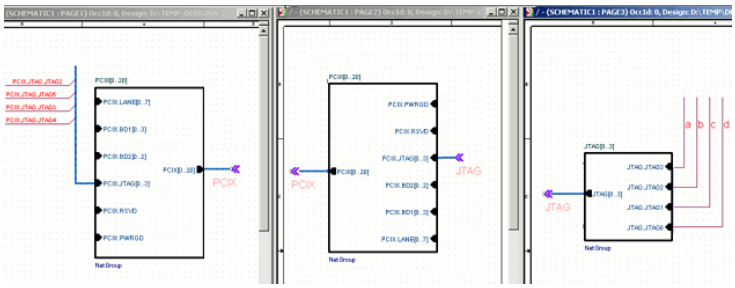
Scenario 2
In this example, the four signals, a, b, c & d are connected to signals on SCHEMATIC3:PAGE1 using a NetGroup off-page connector.



1. Again, the signals on SCHEMATIC1:PAGE1 are first tapped into a large NetGroup, PCIX, containing 19 signals.
2. In this case, the PCIX.JTAG NetGroup port is connected to the JTAG hierarchical port of the SCHEMATIC3 hierarchical block.
3. Finally, on SCHEMATIC3:PAGE1, the JTAG signals are tapped out by placing a JTAG NetGroup connector.

Using a Named NetGroup to connect signals across pages in a design

A NetGroup also increases the ease of the connectivity of signals across the pages of a design. In this example, you will connect signals across pages in a flat design.



1. The signals on SCHEMATIC1:PAGE1 are first tapped into a large NetGroup, PCIX, containing 19 signals. This associated NetGroup definition contains all the signals of this page that need to be connected to pages across the design.
   Four signals need to be connected from PAGE1 to PAGE3 on SCHEMATIC1.
2. On PAGE1, these signals are connected via a bus to the PCIX.JTAG NetGroup entry point.
3. The PCIX NetGroup off-page connector creates the outlet for the signals of the NetGroup.

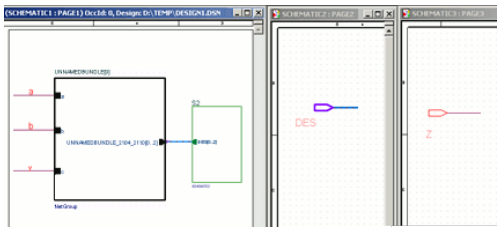
4. Another instance of the NetGroup is placed on SCHEMATIC1:PAGE2 to tap out the signals from SCHEMATIC1:PAGE1. To build the connectivity of these signals on SCHEMATIC1:PAGE2, place a NetGroup off-page connector PCIX on the page.
5. The signals are part of the JTAG NetGroup (included within the PCIX NetGroup). To tap out these signals, connect a NetGroup off-page connector to the PCIX.JTAG NetGroup entry in the PCIX NetGroup on SCHEMATIC1:PAGE2.
6. Next, on SCHEMATIC1:PAGE3, place an instance of the JTAG NetGroup.
7. Also, place a JTAG off-page connector to the JTAG NetGroup port.
8. Finally, tap out the signals from the NetGroup entry points to complete the signal connectivity.

Un-named NetGroup connectivity

Similar to named NetGroups, you can also use un-named NetGroups to connect signals across pages in a hierarchical design. Again, as in named NetGroups, in un-named NetGroups, you can connect signals across a page at the same level of a hierarchical design or across pages in a flat design.

Using an Un-named NetGroup to connect signals across a hierarchy

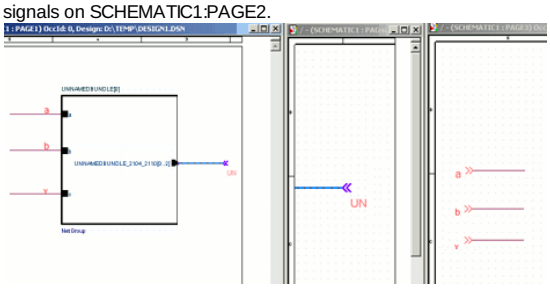
Along with using named NetGroups to connect signals on different levels of a hierarchical design, you can also use un-named NetGroups. We use unnamed NetGroup to group multiple signals on the fly. This implies that we create (in the design) any signals we want to include in the NetGroup.
In this example, the signals, a, b, and y, at the top level (SCHEMATIC1:PAGE1) of the design are to be connected to corresponding signals on the page, SCHEMATIC3:PAGE3 at a lower level of the hierarchy.



1. An unnamed NetGroup is first placed on SCHEMATIC1:PAGE1.
   This is an empty associated NetGroup definition so we still need to add members to this NetGroup.
2. Specify three scalar members for the NetGroup. These will be used as entry points from the signals a, b, and y to the unnamed NetGroup. To this, add three hierarchical pins and name them a, b, and y.
3. Then connect the three signals to the NetGroup entry points.
4. The NetGroup port is then connected to a hierarchical port, named DES, of the SCHEMATIC2:PAGE1 hierarchical block placed on SCHEMATIC1:PAGE1.
5. The NetGroup port DES is placed on SCHEMATIC2:PAGE1.

Using a Un-named NetGroup to connect signals across pages in a design

An unnamed NetGroup can also be used to connect signals across pages at the same level of a design.
In this example, the signals on SCHEMATIC1:PAGE1 of the design need to be connected to signals on SCHEMATIC1:PAGE2.



1. An unnamed NetGroup is first placed on SCHEMATIC1:PAGE1.
   This is an empty associated NetGroup definition so we still need to add members to this NetGroup.
2. Specify three scalar members for the NetGroup. These will be used as entry points from the signals a, b, and y to the unnamed NetGroup. To this, add three hierarchical pins and name them a, b, and y.
3. Then connect the three signals to the NetGroup entry points.
4. Since we are connecting signals across pages at the same level of a design, the NetGroup is connected to a NetGroup off-page connector UN.
5. To build the connectivity of these signals on SCHEMATIC1:PAGE2, place a NetGroup off-

page connector UN on the page.

6. Finally, on SCHEMATIC3:PAGE3, create off-page connectors for the signals, a, b, and y.

NetGroup and Bus Member Net Generation

In a design that contains multiple NetGroups and buses, if you short two of these objects together, the signal output of the short depends on the definition of the objects.

If you short together buses and NetGroups, the resultant signal will be generated by the object with higher significance (as described in this section). In this section, the term **Winning Bus** is used to specify the object (bus or NetGroup) that defines the resultant signals in the connectivity of a design. This term is used in the case of a bus as well as a NetGroup.

In this section:

Net Generation Scenarios

When you short a bus/NetGroup to a bus/NetGroup, the short will result in:

- a resultant object (bus or NetGroup)
- a winning bus
- the flat nets generated from the short
- the associated NetGroup definition (in case a NetGroup is involved in the short)

This table describes scenarios that you encounter when you short together NetGroups and buses in a Capture design. The table is followed by one example for each of the described scenarios.

Short	**Generated Object (** *Bus or NetGroup* **)**	**Winning Bus**	**Generated Flat Nets**	**Defintion**
Bus & Bus (different width)	Bus	Higher width bus	Follow lexicographic order	NA
EXAMPLE Bus A[0:3] - Members: A0, A1, , A2, A3 Bus B[0:5] - Members: B0, B1, B2, B3, B4, B5				
	B[0..5]	B[0..5]	A0 A1 A2 A3 B4 B5	NA
NetGroup & NetGroup - physical short (different width)	NetGroup	Higher width NetGroup	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
EXAMPLE NetGroup S[0..5] - Members: S.JTAG0, S.JTAG1 ,S.JTAG2 ,S.JTAG3, S.JTAG4, S.JTAG5 NetGroup B[0:2] - Members: B.B0, B.B1, B.B2				
	S[0..5]	S[0..5]	S.JTAG0, S.JTAG1, S.JTAG2, S.JTAG3, S.JTAG4, S.JTAG5	S[0..5]
NetGroup & NetGroup - logical short	NetGroup	Higher width NetGroup	Winning Bus defines flat	associated NetGroup

(different width)			nets	definition should match winning bus
EXAMPLE NetGroup S[0..5] - Members: S.JTAG0, S.JTAG1 ,S.JTAG2 ,S.JTAG3, S.JTAG4, S.JTAG5 NetGroup B[0:2] - Members: B.B0, B.B1, B.B2				
	S[0..5]	S[0..5]	S.JTAG0, S.JTAG1, S.JTAG2, S.JTAG3, S.JTAG4, S.JTAG5	S[0..5]
In the case of a logical (named) connection: • if the NetGroups aliases are different, the NetGroups are shorted together. • if the associated NetGroup definitions are the same, the NetGroups are shorted together.				
NetGroup & NetGroup (same width)	NetGroup	Lexicographically smaller NetGroup	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
EXAMPLE NetGroup S[0..2] - Members: S.JTAG0, S.JTAG1 ,S.JTAG2 NetGroup B[0..2] - Members: B.B0,B.B1 and B.B2				
NetGroup & NetGroup - logical short (same width)	NetGroup	Lexicographically smaller NetGroup	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
EXAMPLE NetGroup S[0..2] - Members: S.JTAG0, S.JTAG1 ,S.JTAG2 NetGroup B[0..2] - Members: B.B0,B.B1 and B.B2				
In the case of a logical (named) connection, if the aliases names of both the NetGroups is the same, only then will				

the two NetGroups be shorted together.				
	B[0..2]	B[0..2]	B.B0,B.B1 and B.B2	B[0..2]
Bus & NetGroup (NetGroup width higher)	NetGroup	NetGroup	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
EXAMPLE NetGroup S[0..5] - Members: S.JTAG0, S.JTAG1, S.JTAG2, S.JTAG3, S.JTAG4, S.JTAG5 Bus B[0:2] - Members: B.B0, B.B1, B.B2				
	S[0..5]	S[0..5]	S.JTAG0, S.JTAG1, S.JTAG2, S.JTAG3, S.JTAG4, S.JTAG5	S[0..5]
Bus & NetGroup (Bus width higher)	Hybrid Bus (Bus+NetGroup)	Higher width NetGroup	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
EXAMPLE NetGroup S[0..2] - Members: S.JTAG0, S.JTAG1, S.JTAG2 Bus B[0:5] - Members: B.B0, B.B1, B.B2, B.B3, B.B4, B.B5				
	B[0..5]	B[0..5]	B.JTAG0, B.JTAG1, B.JTAG2, B3, B4 and B5	S[0..2]
Bus & NetGroup (same width)	NetGroup	NetGroup	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
EXAMPLE NetGroup S[0..2] - Members: S.JTAG0, S.JTAG1 ,S.JTAG2 Bus B[0:2] - Members: B.B0, B.B1, B.B2				
	S[0..5]	S[0..5]	S.JTAG0, S.JTAG1, S.JTAG2	S[0..5]
NetGroup wire & NetGroup OPC/GLOBAL/PORT	NetGroup	NetGroup OPC/GLOBAL/PORT	Winning Bus defines flat nets	associated NetGroup definition should

(different widths)				match winning bus
EXAMPLE NetGroup S[0..5] - Members: S.JTAG0, S.JTAG1 ,S.JTAG2, S.JTAG3, S.JTAG4, S.JTAG5 NetGroup OPC B[0..2] - Members: B.B0, B.B1, B.B2				
	B[0..2]	B[0..2]	B.B0, B.B1, S.JTAG2, S.JTAG3, S.JTAG4, S.JTAG5	B[0..2]
Bus & NetGroup OPC/GLOBAL/PORT (NetGroup width higher)	NetGroup	NetGroup OPC/GLOBAL/PORT	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
EXAMPLE NetGroup OPC S[0..5] - Members: S.JTAG0, S.JTAG1, S.JTAG2, S.JTAG3, S.JTAG4, S.JTAG5 Bus B[0:2] - Members: B.B0, B.B1, B.B2				
	S[0..5]	S[0..5]	S.JTAG0, S.JTAG1, S.JTAG2, S.JTAG3, S.JTAG4, S.JTAG5	S[0..5]
Bus & NetGroup OPC/GLOBAL/PORT (bus width higher)	NetGroup	NetGroup OPC/GLOBAL/PORT	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
EXAMPLE NetGroup S[0..5] - Members: S.JTAG0, S.JTAG1, S.JTAG2, S.JTAG3, S.JTAG4, S.JTAG5 Bus OPC B[0:2] - Members: B.B0, B.B1, B.B2				
	S[0..2]	S[0..2]	S.JTAG0, S.JTAG1, S.JTAG2, B3,B4,B5	S[0..2]
NetGroup wire and Bus OPC/GLOBAL/PORT (NetGroup width higher)	Bus	Bus OPC/GLOBAL/PORT	Winning Bus defines flat nets	associated NetGroup definition should match winning bus

EXAMPLE NetGroup S[0..2] - Members: S.JTAG0, S.JTAG1, S.JTAG2 Bus OPC B[0:5] - Members: B.B0, B.B1, B.B2, B.B3, B.B4, B.B5				
	S[0..2]	S[0..2]	B0,B1, B2, S.JTAG3, S.JTAG4, S.JTAG5	S[0..5]
NetGroup wire and Bus OPC/GLOBAL/PORT (bus width higher)	Bus	Bus OPC/GLOBAL/PORT	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
EXAMPLE NetGroup S[0..2] - Members: S.JTAG0, S.JTAG1, S.JTAG2 Bus OPC B[0:5] - Members: B.B0, B.B1, B.B2, B.B3, B.B4, B.B5				
	B[0..5]	B[0..5]	B0,B1, B2, B3,B4,B5	S[0..2]
NetGroup wire and NetGroup connector	NetGroup	NetGroup connector	Winning Bus defines flat nets	associated NetGroup definition should match winning bus
NetGroup connector and NetGroup connector	NetGroup	Lexicographically smaller NetGroup connector	Winning Bus defines flat nets	associated NetGroup definition should match winning bus

12

Automation in Capture

This chapter covers the automation features available in Capture. The topics covered in this section are:

- Working with Macros
- Working with Capture TCL

Working with Macros

The macro recording capability of Capture is a subset of the macro functionality provided by Visual Basic for Applications. In Capture, when you record macros in a schematic page editor, a .BAS file is written. You can either edit this file later, or write your own.

While working with macros in Capture, you must take care of the following:

- Each macro must be written in a single MAIN function. Do not use IF-THEN-ELSE subroutines.
- Capture provides a macro command for each menu command in the schematic page editor. In some cases, a macro command cannot specify all of the values presented in the dialog box it opens. For example, you cannot add or edit properties using the PlacePart macro command.
- Commands that change the active window are not available to macros. The following commands are unavailable to macros:
  - Ascend Hierarchy
  - Descend Hierarchy
  - Edit Part (on the pop-up menu)
  - Undo
  - Redo
- In a macro file, anything following a single quotation mark (') is ignored as comment text. The pause button automatically adds the following comment line:

```
PAUSE...
```

> For more information on macro language, see Language Reference (ENDUSER.doc) in your Capture installation directory.

The topics covered in this section are:

- Recording and Saving a Macro
- Playing a macro
- Macro input boxes
- Creating macro shortcut keys
- Macro Command Reference

Recording and Saving a Macro

To ensure that the macro recorded by you in the current session of Capture, is available for future use, you need to record and save the macro.

Recording a Macro

In the schematic page editor, you can record a series of editing actions as a macro.

To record a macro

1. Click the left mouse button on the schematic page to set a location to begin recording the macro.

2. From the Macro menu, choose Record. Capture displays the macro recorder tool palette.

OR

1. From the Configure Macro dialog box, choose Record.
2. Perform the series of edits that you want to record as a macro, using the three macro record buttons as necessary. The buttons perform the following tasks:
3. Use the left button to stop recording the macro.
4. Use the center button to pause recording. The pause mode is in effect until you choose the center button again.
5. Use the right button to cause a command to record in a "with dialog" mode. If a command is recorded in this mode, the value you enter while recording the macro is not saved. Instead, when the macro is run, the command displays a dialog box so that you can fill in a value. When recording the "with dialog" mode is in effect until you choose the right button again.
6. Choose the left macro record button to stop recording the macro.

Saving a Macro

When you record a macro, Capture assigns it a temporary name, and it is treated as a temporary macro. Temporary macros can be run during the current Capture session, but they are not saved for use in subsequent Capture sessions. You can make a temporary macro permanent by assigning it a name in the Configure Macro dialog box.

Once you give a macro a name and save it, it will automatically appear in the macro name list box in the Configure Macro dialog box. The text you enter as the menu entry appears in the Macro menu, along with the macro's access key definition, if you specified one. The text you enter as the description appears in the Description text box in the Configure Macro dialog box when you select the macro name.

To save a configured macro

1. From the Macro menu, choose the Configure command.
   Capture displays the Configure Macro dialog box.
2. In the Macro Name text box, enter a name for the macro.
3. Click Save. Capture displays the Macro Name dialog box.
4. If you want to assign an access key, enter an access key or key combination (for example, CTRL+7) in the Keyboard Assignment text box.

   - Do not assign the access key combination CTRL+ALT+DEL to a macro. Capture's macro system won't override this combination to restart your system.
   - Macro shortcuts may be any alphanumeric character or a function key (like F7), and may additionally use the CTRL, SHIFT, and ALT keys. For example, CTRL+SHIFT+Q and F6. For more information, see Creating macro shortcut keys.

5. If you want the macro to appear as an entry in the Macro menu, enter the appropriate text (for example, "Name Wires") in the Menu Assignment text box.

   You can make an access key for a macro by placing an ampersand character (&) in front of one letter in the Menu Assignment text box of the Macro Name dialog box. For example, "&Name Wires" creates the access key combination ALT, M, N to use a macro that is configured to appear as "Name Wires" in the Macro menu.

6. If you want to further describe the macro, enter the appropriate text in the Description text box.
7. Click OK. The *Save As* dialog box appears.
8. Select an appropriate file location and file name, then click *Save*.

9. In the Configure Macro dialog box, click the *Close* button.

To add a macro

- From the Macro menu, choose the Configure command. Capture displays the Configure Macro dialog box.

To remove macro

- From the Macro menu, choose the Configure command. Capture displays the Configure Macro dialog box.

The following macro file searches for three specific kinds of parts and adds a Cost property with a value for each one.

```
Macro commands are reserved words, and should not be used for
macro function or procedure names.
SUB Cost
'MACROMENU &Cost
            'MACROKEY CTRL+1
            'MACRODESCRIPTION Adds a price to a known set of
parts.
FindParts "74ls04", FALSE
            SetProperty "Cost", "$0.70"
            DisplayProperty "Cost", "Arial", 9, FALSE, FALSE,
48, 0
FindParts "74ls08", FALSE
            SetProperty "Cost", "$0.75"
            DisplayProperty "Cost", "Arial", 9, FALSE, FALSE,
48, 0
FindParts "74ls32", FALSE
            SetProperty "Cost", "$0.80"
            DisplayProperty "Cost", "Arial", 9, FALSE, FALSE,
48, 0
END SUB
```

The first line in each set finds and selects all parts of a specific type. Once the parts are selected, the second line of each set changes the value of Cost to the specified value. If the part doesn't have the Cost property, then the property is created. The final line of each set makes the value of the Cost property visible.

The first set of lines search for and edit 74LS04 type parts. The second set of lines search for and edit 74LS08 type parts. The last set of lines search for and edit 74LS32 type parts. When this macro is configured, it will appear in the lower portion of the Macro menu. The name of the command on the menu is "Cost". In addition to being chosen from the menu with the mouse, the macro can be reached with the access keys ALT, M, O and the shortcut CTRL+1.

Playing a macro

To play the currently selected macro

1. Click the left mouse button on the schematic page to set a starting location for your macro.
2. From the *Macro* menu, choose the Play command.

OR

1. Click the left mouse button on the schematic page to set a starting location for your macro.
2. From the Configure Macro dialog box, choose Play.

To play a macro

1. Click the left mouse button on the schematic page to set a starting location for your macro.
2. If the macro appears in the *Macro* menu, choose it from the menu.

OR

1. Click the left mouse button on the schematic page to set a starting location for your macro.
2. If the macro appears in the Macro menu and has an access key, press ALT, M, X where X is the letter or number corresponding to the macro command. For example ALT, M, N for a macro called "Name Wires."

OR

1. Click the left mouse button on the schematic page to set a starting location for your macro.
2. If the macro has a shortcut key or keys, press the shortcut key or keys. For example, if "Name Wires" uses CTRL+F6 then press CTRL+F6.

Macro input boxes

Macro input boxes create a dialog box as you execute a macro. Use input boxes if you only want to prompt the user for one variable of a macro command.
In the following macro command description, **Magenta** indicates any character string. Command description variables enclosed in brackets ("[" and "]") are optional. If a variable can be a character string, it must be immediately followed by the dollar sign ($). These possibilities are shown in **Blue**.

```
InputBox$(prompt $[, [title $][, [default $], [X, Y]]])
```

A prompt for the input box is specified by **prompt** . The input box's title is specified by **title** . The default value for the input box is specified by **default** . The horizontal location of the input box is specified by **X** , and the vertical location of the input box is specified by **Y** .

```
Z$ = "Line Coordinates"
X1 = 0
X2 = 0
Y1 = 0
Y2 = 0
PlaceLine InputBox("What is X1?", Z$, X1), InputBox("What is
Y1?", Z$, Y1), InputBox("What is X2?", Z$, X2), InputBox("What is Y2?", Z$, Y2)
```

Creating macro shortcut keys

You can make an access key for a macro by placing an ampersand character (&) in front of one letter in the Menu Assignment text box of the Macro Name dialog box. For example, *"&Name Wires"* creates the access key combination `ALT, M, N` to use a macro that is configured to appear as "Name Wires" in the *Macro* menu.

You can also create a shortcut key for a macro command, using the Keyboard assignment text entry box in the Macro Name dialog box. Use the following form to assign a macro shortcut key, where **blue** indicates a required value and **red** indicates an optional value.

modifier* + *keydescription

The value for ***modifier*** can be one of the following:

- `CTRL`
- `ALT`
- `SHIFT`
- `CTRL+ALT`
- `CTRL+SHIFT`
- `ALT+SHIFT`
- `CTRL+ALT+SHIFT`

The value for ***keydescription*** can be one of the following:

- A..Z. Any alphabetic character.
- 0..9. Any numeric character.
- F1..F24. Any one function key.

The following examples are possible combinations:

- F1
- CTRL+SHIFT+5
- ALT+B

-- Do not assign the access key combination CTRL+ALT+DEL to a macro. Capture macro system will not override this combination to restart your system.

-- The ESC key can be used as a shortcut key, but cannot be used in junction with any other key. For example, SHIFT+ESC is not a valid shortcut key.

-- The keys "." and "=" do not work in macro key sequences.

Macro Command Reference

The commands covered in this section are:

Selection commands

Use the macro selection commands to find and select objects. If Capture finds multiple objects that meet the selection criteria, it places them in a rectangular selection set, and the upper left corner of the rectangle is indicated as the Current Location. If Capture finds only one object, it selects the object and changes the Current Location value to the upper left corner of the object. If no objects are found, nothing is selected, and the current location is left unchanged.

Selection macro commands are listed in the following table. In the descriptions, **Red** indicates a decimal number value. **Blue** indicates a value of either TRUE or FALSE. **Magenta** indicates any character string.

Macro commands are reserved words, and should not be used for macro function or procedure names.

Function	Syntax	Description	Example
FindBookmarks	**FindBookmarks "*value*", *matchcase***	Use this command to select bookmarks with a property name or property value that matches **value**. If ***matchcase*** is TRUE, the search for ***value*** is case sensitive.	`FindBookmarks "Bookmark1", TRUE`
FindDRCMarks	**FindDRCMarks "** *value* **",** ***matchcase***	Use this command to select all DRC markers with a property with a value of ***value*** . If ***matchcase*** is TRUE, the search for ***value*** is case sensitive.	`FindDRCMarks "74LS32", TRUE`
FindHierarchicalPorts	**FindHierarchicalPorts "** *value* **",** ***matchcase***	Use this command to select hierarchical ports with a property with a	`FindHierarchicalPorts "A", FALSE`

		value of ***value*** . If ***matchcase*** is TRUE, the search for ***value*** is case sensitive.	
FindNets	**FindNets "** *value* **",** ***matchcase***	Use this command to select all nets that have a property with a value of ***value*** . If the selection consists of a single net, then CurrentLocation is set to the net's upper left vertex. If ***matchcase*** is TRUE, the search for ***value*** is case sensitive.	`FindNets "Carry", FALSE`
FindOffPageConnectors	**FindOffPageConnectors "value",** ***matchcase***	Use this command to select off-page connectors with a property name or property value that matches **value**. If ***matchcase*** is TRUE, the search for ***value*** is case sensitive.	`FindOffPageConnectors "A", FALSE`
FindParts	**FindParts "** *value* **",** ***matchcase***	Use this command to select all part instances that have a property value that matches ***value*** . If ***matchcase*** is TRUE, the search for ***value*** is case sensitive.	`FindParts "74LS00", FALSE`
FindText	**FindText "** *value* **",** ***matchcase***	Use this command to select text with a property value that matches ***value*** . If ***matchcase*** is TRUE, the search for ***value*** is case sensitive.	`FindText "a line", TRUE`
SelectAll	**SelectAll**	Use this command to select all objects in the	`SelectAll`

		page. CurrentLocation is unaffected.	
SelectBlock	**SelectBlock** ***X1***, ***Y1*** , ***X2*** , ***Y2*** , ***SelectID***	Use this command to select the set of objects contained in the area identified by the rectangle at the coordinates CurrentLocation + (***X1***, ***Y1***) and CurrentLocation + (***X2***, ***Y2***). Object selection is controlled using the rules chosen on the Select tab in the Preferences dialog box. If ***SelectID*** is TRUE, then previously selected objects remain selected. If ***SelectID*** is FALSE then previously selected objects are deselected.	`SelectBlock -1.00, -1.00, -5.00, -5.00, TRUE`
SelectObject	**SelectObject** ***X*** , ***Y*** , *SelectID*	Use this object to select the object as the CurrentLocation + (***X***, ***Y***), or deselect any currently selected objects at that location. If an object is selected, the CurrentLocation is set to the upper left corner of the object. If no object is selected at the location (***X***, ***Y***), then the selection set is empty and the CurrentLocation is set to CurrentLocation + (***X***, ***Y***). If ***SelectID*** is TRUE, then previously selected	`SelectObject -1.00, -1.00, FALSE`

		objects remain selected. If ***SelectID*** is FALSE then previously selected objects are deselected.	

Macro Editing Commands

Use the macro editing commands to edit the selected objects. Except where noted, these commands do not change the current location of an object, and the selection set.

In the following table, **Red** in the macro command description section indicates a decimal number value. **Blue** indicates a value of either TRUE or FALSE. **Dark Cyan** indicates an integer value. **Magenta** indicates any character string.

> Macro commands are reserved words, and should not be used for macro function or procedure names.

Function...	Syntax	Description	Example
Copy	**copy**	Use this command to copy all selected objects to the Clipboard.	copy
Cut	**Cut**	Use this command to remove all selected objects from the schematic page, and place them on the Clipboard. After the operation, no objects are selected.	Cut
Delete	**Delete**	Use this command to delete all of the selected objects, without placing them on the Clipboard. After the operation, no objects remain selected.	Delete
Drag	**Drag X, Y, dragflag**	Use this command to drag all selected objects from CurrentLocation to CurrentLocation + (**X** , **Y**). CurrentLocation is changed to CurrentLocation + (**X** , **Y**). If **dragflag** is TRUE, then nets will stretch to maintain connectivity to selected parts, and remain orthogonal. If **dragflag** is FALSE, then select will stretch to maintain connectivity, but not remain orthogonal.	`Drag 2.50, 3.10, TRUE`
Duplicate	**Duplicate X, Y**	Use this command to create a copy of the selected object at CurrentLocation + (**X** , **Y**). CurrentLocation is changed to CurrentLocation + (**X** , **Y**), and the new object is	`Duplicate -2.50, -4.60`

		selected.	
MirrorHorizontal	**MirrorHorizontal**	Use this command to mirror all selected objects horizontally.	`MirrorHorizontal`
MirrorVertical	**MirrorVertical**	Use this command to mirror all selected objects vertically.	`MirrorVertical`
Move	**Move *X* , *Y* , *moveflag***	Use this command to move all selected objects from CurrentLocation to CurrentLocation + (**X** , **Y**). CurrentLocation is changed to CurrentLocation + (**X** , **Y**). If **moveflag** is TRUE, then nets will stretch to maintain connectivity to selected parts, and remain orthogonal. If **moveflag** is FALSE, then select will stretch to maintain connectivity, but not remain orthogonal.	`Move 2.50, 3.10, TRUE`
Paste	**Paste *X*, *Y***	Use this command to paste the contents of the Clipboard to CurrentLocation + (**X** , **Y**). CurrentLocation is changed to CurrentLocation + (**X** , **Y**). All of the objects pasted from the Clipboard are selected.	`Paste 4.00, 5.50`
ReplacePart	**ReplacePart "*libname* ", "*pkgname* ", "*view* ", "*devicedesignator* "**	Use this command to change the selected parts from their normal view to their convert view, or from their convert view to their normal view. The part is specified by ***pkgname*** , and the reference is specified by ***devicedesignator*** . *The library containing the part is specified by* ***libname*** *. The view of the part is specified by* ***view*** . Check that the library path and name specified is correct. You can use this macro to change the library, part, or both, similar to the Replace Cache command. If you do this, be certain that the new part's pin arrangement matches that of the one being replaced.	`ReplacePart "C:\PROGRAM FILES\ORCAD\ CAPTURE\LIBRARY\TTL.OLB", "74LS08", "74LS08.Normal", "D"`
Rotate	**Rotate**	Use this command to rotate all selected	`Rotate`

		objects. CurrentLocation is set to the new upper left corner of the selection set.	
SetColor	**SetColor** ***colorID***	Use this command to set the color property to ***colorID*** for all selected objects. Objects that do not have a color property are unaffected. Capture ColorID to RGB mapping.	SetColor 1
SetFillStyle	**SetFillStyle** ***StyleID***	Use this command to set the fill style property to ***styleID*** for all selected objects. Objects that do not have a fill style are unaffected.	`SetFillStyle 3`
SetFont	**SetFont "*fontID* ", *colorID* , *boldID* , *italicID***	Use this command to set the font and font style for the selected objects. The font is specified by ***fontID*** . The text's color is specified by ***colorID*** . *If* ***boldID*** is TRUE, then the text appears as bold, otherwise it appears non-bold. If **italicID** is TRUE, then the text appears italicized, otherwise it appears non-italicized. Capture ColorID to RGB mapping.	`SetFont "Arial", 13, FALSE`
SetHatchStyle	**SetHatchStyle** ***StyleID***	Use this command to set the hatch style property to ***StyleID*** for all selected objects. Objects that do not have a fill style are unaffected.	`SetHatchStyle 3`
SetLineStyle	**SetLineStyle** *StyleID*	Use this command to set the line style property to ***StyleID*** for all selected objects. Objects that do not have a line style are unaffected.	`SetLineStyle 2`
SetLineWidth	**SetLineWidth** ***widthID***	Use this command to set the line width property to ***widthID*** for all selected objects. Objects that do not have a line width are unaffected.	`SetLineWidth 2`

Placement commands

Use the macro placement commands to place objects in the schematic page editor. Some of the placement commands can be declared as WithDialog. For example, PlaceBlock can be used as PlaceBlockWithDialog. Macro commands using WithDialog prompt you to provide information, like the name of a hierarchical block, when executed.

In the following macro command descriptions, **Red** indicates a decimal number value. **Blue** indicates a Boolean value where zero means FALSE and non-zero numbers mean TRUE.

Magenta indicates any character string. Command description variables enclosed in brackets ("**[**" and "**]**") are not used when the command is declared as WithDialog.

> Macro commands are reserved words, and should not be used for macro function or procedure names.

The macro placement commands are:

- PlaceArc
- PlaceBlock
- PlaceBookmark
- PlaceBus
- PlaceBusEntry
- PlaceEllipse
- PlaceGround
- PlaceJunction
- PlaceLine
- PlaceNetAlias
- PlaceNextPolygonPoint
- PlaceNextPolylinePoint
- PlaceNoConnect
- PlaceOffPage
- PlacePart
- PlacePicture
- PlacePin
- PlacePolygon
- PlacePolyline
- PlacePort
- PlacePower
- PlaceRectangle
- PlaceText
- PlaceTitleblock
- PlaceWire

PlaceArc

Function:	**PlaceArc** ***X1*** , ***Y1*** , ***X2*** , ***Y2*** , ***X3*** , ***Y3*** , ***X4*** , ***Y4*** Use this command to place an arc specified by the rectangle with the coordinates CurrentLocation + (***X1*** , ***Y1***) and the CurrentLocation + (***X2*** , ***Y2***). The starting point of the arc is specified by the coordinate (***X3*** , ***Y3***), and is terminated at the coordinate specified by (**X4** , **Y4**). The current location is changed to the upper left corner of the arc's defining rectangle or the coordinate (**X4** , **Y4**). The arc becomes the only object selected.
Example:	`PlaceArc -0.01, -1.12, 2.03, 0.92, 1.91, -0.60, 0.00, 0.00`

PlaceBlock

Function:	**PlaceBlock** *X1* , *Y1* , *X2* , *Y2* , *["libname", " schematicname* ", "*blockname* ", "*primitiveflag* "] Use this command to place a hierarchical block defined by CurrentLocation + (**X1**, **Y1**) and CurrentLocation + (**X2**, **Y2**). The hierarchical block's name and reference are specified by ***blockname*** . The current location is changed to the upper left corner of the hierarchical block, and the hierarchical block becomes the only selected object. If ***blockname*** is not specified, Capture will assign it a unique internal name. If ***blockname*** is specified, but not unique, the command will fail and no objects will be selected. If the hierarchical block has an attached schematic folder, it is specified by ***schematicname*** . The path and library name containing the schematic folder are specified by ***libname*** . The hierarchical block's primitivity is

	specified by ***primitiveflag*** , which is set to YES, NO, or DEFAULT.

Example:	`PlaceBlock -0.10, -0.60, 0.40, -0.30, "C:\PROGRAM FILES\ ORCAD\CAPTURE\Design\Fulladd\Fulladd.OLB", "HALFADD", "Halfadd_A", "DEFAULT"`
	
	`PlaceBlockWithDialog 0.30, -0.30, 0.80, 0.00`

PlaceBookmark

Function:	**PlaceBookmark *X* , *Y* , ["*name* "]** Use this command to place a bookmark at the current location added to the offset (***X*** , ***Y***). The bookmark name is specified by ***name*** . The current location is changed to its current value added to the offset (***X*** , ***Y***), and the bookmark becomes the only selected object.

Example:	`PlaceBookMark 0.20, 0.00, "C:\PROGRAM FILES\ORCAD\CAPTURE\ DESIGN1.DSN", "B"`
	
	`PlaceBookMarkWithDialog -1.80, 0.50`

PlaceBus

Function:	**PlaceBus** *X1* , *Y1* , *X2* , *Y2* Use this command to place the first segment of a bus with the starting point at CurrentLocation + (***X1*** , ***Y1***), and the endpoint at CurrentLocation + (***X2*** , ***Y2***). CurrentLocation is set to CurrentLocation + (***X2*** , ***Y2***). The bus becomes the only selected object.
Example:	`PlaceBus -1.40, 1.00, 0.10, 1.00`

PlaceBusEntry

Function:	**PlaceBusEntry *X* , *Y* , *rotate*** Use this command to place a bus entry at CurrentLocation + (***X*** , ***Y***). If **rotate** is FALSE, the bus entry appears as a forward slash (/). If **rotate** is TRUE, then the bus entry appears as a backward slash (\). CurrentLocation is set to CurrentLocation + (***X*** , ***Y***). The bus entry becomes the only selected object.
Example:	`PlaceBusEntry -3.00, -1.00, FALSE`

PlaceEllipse

Function:	**PlaceEllipse *X1* , *Y1* , *X2* , *Y2*** Use this command to place an ellipse specified by the points CurrentLocation + (***X1*** , ***Y1***), and CurrentLocation + (***X2*** , ***Y2***). CurrentLocation is changed to CurrentLocation + (***X2*** , ***Y2***), and the ellipse becomes the only selected object.
Example:	`PlaceEllipse 0.00, 0.00, 1.20, 1.30`

PlaceGround

Function:	**PlaceGround *X* , *Y*, *["libname", "symbolname", "name"]*** Use this command to place an instance of a ground symbol at CurrentLocation +

	(***X*** , ***Y***). The ground symbol is specified by ***symbolname*** , and is given the name ***name*** . The library containing the ground symbol is specified by **libname** . CurrentLocation is set to CurrentLocation + (**X** , **Y**), and the ground symbol becomes the only selected object. If there is no symbol called ***symbolname*** in a library named ***libname*** , then the command fails and no objects are selected.

Example:	`PlaceGround 0.30, 0.40, "C:\PROGRAM FILES\ORCAD\CAPTURE\ LIBRARY\CAPSYM.OLB", "GND FIELD SIGNAL", "GND"`
	
	`PlaceGroundWithDialog -1.90, -0.30`

PlaceJunction

Function:	**PlaceJunction** ***X*** , ***Y*** Use this command to place a junction specified by the points CurrentLocation + (***X*** , ***Y***). If a junction already exists at this location, then it is removed.
Example:	`PlaceJunction -0.10, -1.30`

PlaceLine

Function:	**PlaceLine** *X1* , *Y1* , ***X2*** , ***Y2*** Use this command to place a line specified by the points CurrentLocation + (***X1*** , ***Y1***), and CurrentLocation + (***X2*** , ***Y2***). CurrentLocation is changed to CurrentLocation + (***X2*** , ***Y2***). The line becomes the only selected object.
Example:	`PlaceLine -0.10, 0.40, 1.20, 0.40`

PlaceNetAlias

Function:	**PlaceNetAlias** *X* , *Y* , ["alias"] Use this command to place a net alias at CurrentLocation + (***X*** , ***Y***). The name of the net alias is specified by **alias** . CurrentLocation is set to CurrentLocation + (***X*** , ***Y***). The net alias becomes the only object selected. If CurrentLocation + (***X*** , ***Y***) is not a wire or bus, the command fails. If CurrentLocation + (***X*** , ***Y***) is on a bus, but **alias** is not a valid bus name, the command fails. No objects are selected if the command fails.

Example:	`PlaceNetAlias -3.00, -1.30, "A[1..5]"`
	
	`PlaceNetAliasWithDialog 2.00, 1.00`

PlaceNextPolygonPoint

Function:	**PlaceNextPolygonPoint** *X* , *Y* Use this command to add a point to the currently selected polygon. The point is specified by CurrentLocation + (***X*** , ***Y***). CurrentLocation is then changed to CurrentLocation + (***X*** , ***Y***), and the polygon becomes the only selected object. If a polygon is not selected, the command fails and no objects are selected.
Example:	`PlaceNextPolygonPoint 1.30, 3.20`

PlaceNextPolylinePoint

Function:	**PlaceNextPolylinePoint** *X* , *Y*

	Use this command to add a point to the currently selected polyline. The point is specified by CurrentLocation + (***X*** , ***Y***). CurrentLocation is then changed to CurrentLocation + (***X*** , ***Y***), and the polyline becomes the only selected object. If a polyline is not selected, the command fails and no objects are selected.
Example:	`PlaceNextPolylinePoint 1.30, 3.20`

PlaceNoConnect

Function:	**PlaceNoConnect** ***X*** , ***Y*** Use this command to place a no connect symbol at CurrentLocation + (***X*** , ***Y***). CurrentLocation is then changed to CurrentLocation + (***X*** , ***Y***). Nothing remains selected. If CurrentLocation + (***X*** , ***Y***) is not located at the end of a pin, the command fails.
Example:	`PlaceNoConnect 3.40, 5.20`

PlaceOffPage

Function:	**PlaceOffPage** *X* , *Y* , ["*libname* ", "*symbolname* ", "*name* "] Use this command to place an instance of an off-page connector at CurrentLocation + (***X*** , ***Y***). The off-page connector is specified by ***symbolname*** , and is given the name **name** . The library containing the off-page connector is specified by **libname** . The current location is set to CurrentLocation + (***X*** , ***Y***), and the off-page connector becomes the only selected object. If there is no off-page connector called **symbolname** in a library named **libname**, then the command fails and no objects are selected.

Example:	`PlaceOffPage 2.00, 1.20, "C:\PROGRAM FILES\ORCAD\CAPTURE\ LIBRARY", "OFFPAGELEFT-L", "Carry"`
	
	`PlaceOffPageWithDialog -0.50, 0.00`

PlacePart

Function:	**PlacePart** *X* , *Y* , ["*libname* ", "*pkgname* ", "*devicedesignator* ", *convert*] Use this command to place a part instance at CurrentLocation (***X*** , ***Y***). The part is specified by **pkgname**, and the reference is specified by **devicedesignator**. The library containing the part is specified by **libname**. If **convert** is TRUE, then Capture uses the part's convert view rather than the normal view. After the part is placed, it becomes the only selected item. CurrentLocation is changed to CurrentLocation + (***X*** , ***Y***). This command can fail for the following reasons: • There is no library with the name ***libname*** . • There is no package in the specified library with the name ***pkgname***. • There is no device with the specified ***devicedesignator*** in the package. • The ***convert*** parameter has been set to TRUE, but there is no convert for the specified device. If the command fails, CurrentLocation changes to CurrentLocation + (**X**, **Y**).

Example:	`PlacePart 1.60, 0.00, "C:\PROGRAM FILES\ORCAD\CAPTURE\ DESIGN\FULLADD\FULLADD.OLB", "74LS08", "A", FALSE`
	
	`PlacePartWithDialog -1.90, -0.90`

PlacePicture

Function:	**PlacePicture *X* , *Y* , "*filename* "** Use this command to place a picture or graphic. The picture's filename and path are specified by **filename** , and the picture's location is specified by the current location added to the offset (**X** , **Y**). The current location is changed to the picture's location, and the picture becomes the only selected object. If the specified picture file cannot be opened, the command fails and no objects are selected.
Example:	`PlacePicture 1.50, 0.10, "C:\Pictures\Logo.bmp"`

PlacePin

Function:	**PlacePin *X* , *Y* , ["*name* ", "*pintype* ", "*isbus* "]** Use this command to place a hierarchical pin on the selected hierarchical block at CurrentLocation + (**X** , **Y**). The hierarchical pin name is specified by ***name*** , and the pin type is specified by ***pintype*** . *If* ***isbus*** is FALSE, a scalar pin is created, otherwise a bus pin is created. CurrentLocation is changed to CurrentLocation + (**X** , **Y**), and the hierarchical pin is the only selected object. The following situations will cause the command to fail: • CurrentLocation + (**X** , **Y**) does not lie on the boundary of a hierarchical block. • If is ***isbus*** TRUE and name is not a valid bus name. • If is **isbus** FALSE and name is a valid bus name. • If the command fails, the current location is unaffected, and no objects are selected.

Example:	`PlacePin 3.20, 1.20, "A[0..2]", "output", TRUE`
	
	`PlacePinWithDialog 3.10, 2.20`

PlacePolygon

Function:	**PlacePolygon X1, *Y1* , *X2* , *Y2*** Use this command to place the first segment of a polygon. The starting point is defined by CurrentLocation + (**X1, *Y* 1**). The endpoint is defined by CurrentLocation + (**X2, *Y* 2**). CurrentLocation is changed to `CurrentLocation + (X2, Y 2`), and the polygon becomes the only selected object.
Example:	`PlacePolygon 1.30, 1.30, 3.20, 2.40`

PlacePolyline

Function:	**PlacePolyline *X1* , *Y1* , *X2* , *Y2*** Use this command to place the first segment of a polyline. The starting point is defined by CurrentLocation + (***X1*** , **Y1**). The endpoint is defined by CurrentLocation + (**X2** , **Y2**). CurrentLocation is changed to CurrentLocation + (**X2** , **Y2**), and the polygon becomes the only selected object.
Example:	`PlacePolyline 1.30, 1.30, 3.20, 2.40`

PlacePort

Function:	**PlacePort *X* , *Y* , ["*libname* ", "*symbolname*", " name"]**

	Use this command to place an instance of a hierarchical port at CurrentLocation + (**X**, **Y**). The hierarchical port is specified by **symbolname**, and is given the name ***name*** . The library containing the hierarchical port is specified by ***libname*** . CurrentLocation is set to CurrentLocation + (**X**, **Y**), and the hierarchical port becomes the only selected object. If there is no hierarchical port called ***symbolname*** in a library named ***libname*** , then the command fails and no objects are selected.

Example:	`PlacePort 0.00, 0.20, "C:\PROGRAM FILES\ORCAD\CAPTURE\ LIBRARY\CAPSYM.OLB", "PORTBOTH-L", "High"`
	
	`PlacePortWithDialog -1.50, -1.40`

PlacePower

Function:	**PlacePower *X* , *Y* , ["*libname* ", "*symbolname* ", "*name* "]** Use this command to place an instance of a power symbol at CurrentLocation + (***X*** , ***Y***). The power symbol is specified by ***symbolname*** , and is given the name ***name*** . The library containing the power symbol is specified by ***libname*** . CurrentLocation is set to CurrentLocation + (***X*** , ***Y***), and the power symbol becomes the only selected object. If there is no symbol called ***symbolname*** in a library named ***libname*** , then the command fails and no objects are selected.

Example:	`PlacePower 0.00, 1.20, "C:\PROGRAM FILES\ORCAD\CAPTURE\ LIBRARY\CAPSYM.OLB", "VCC_ARROW", "VCC_ARROW"`
	
	`PlacePowerWithDialog 0.60, 0.00`

PlaceRectangle

Function:	**PlaceRectangle *X1* , *Y1* , *X2* , *Y2*** Use this command to place a rectangle specified by the points CurrentLocation + (***X1*** , ***Y1***), and CurrentLocation + (***X2*** , ***Y2***). The current location is changed to CurrentLocation + (***X2*** , ***Y2***), and the rectangle becomes the only selected object.
Example:	`PlaceRectangle 1.30, 4.20, 5.40, 2.20`

PlaceText

Function:	**PlaceText *X1* , *Y1* , *X2* , *Y2* , ["*text* "]** Use this command to place comment text in the rectangle defined by CurrentLocation + (***X1*** , ***Y1***), and CurrentLocation + (***X2*** , ***Y2***). The string of text is specified by **text** . CurrentLocation is changed to CurrentLocation + (***X2*** , ***Y2***), and the text becomes the only selected object.

Example:	`PlaceText -2.10, 0.30, -1.35, 0.42, "|+ 200k 300K 200uf"`
	
	`PlaceTextWithDialog -0.05, 0.68, 0.15, 0.80`

PlaceTitleblock

Function:	**PlaceTitleblock *X* , *Y* , ["*libname* ", "titleblockname", "name"]** Use this command to place a title block symbol at CurrentLocation + (***X*** , ***Y***). The title block is specified by **titleblockname** , in a library specified by ***libname*** . The title block's name is specified by **name**. CurrentLocation is changed to CurrentLocation + (***X*** , ***Y***), and the title block becomes the only selected object. If the specified title block doesn't exist in the specified library, the command fails. If the command fails, no objects are selected.

Example:	`PlaceTitleblock 3.40, 5.20, "C:\PROGRAM FILES\ORCAD\CAPTURE\LIBRARY\CAPSYM.OLB", "TitleBlock0", "TitleBlock0"`
	
	`PlaceTitleBlockWithDialog -1.80, 0.50`

PlaceWire

Function:	**PlaceWire *X1* , *Y1* , *X2* , *Y2*** Use this command to place the first segment of a wire with starting point at CurrentLocation + (***X1*** , ***Y1***), and an endpoint at CurrentLocation + (***X2*** , ***Y2***). CurrentLocation is set to CurrentLocation + (***X2*** , ***Y2***). The wire segment becomes the only selected object.
Example:	`PlaceWire -1.00, -1.00, 2.00, 2.30`

Property commands

Use the macro property commands to add, remove, and display properties. These commands do not affect the current location or the selection set.

In the following macro command descriptions, **Blue** indicates a value of either TRUE or FALSE. **Dark Cyan** indicates an integer value. **Magenta** indicates any character string.

> Macro commands are reserved words, and should not be used for macro function or procedure names.

Function	Syntax	Description	Example
DisplayProperty	**DisplayProperty "*name* ", "*fontID* ", *sizeID* , *boldID* , *italicID* , *colorID* , *rotate***	Use this command to display the value of a property currently not visible on the selected object. The name of the property is specified by ***name*** . The font for the property is specified by ***fontID*** . The font size for the property is specified by ***sizeID*** . If ***boldID*** is TRUE, then the property value appears in bold. If ***italicID*** is TRUE, then the property value appears in italics. The property value's color is specified by ***colorID*** . The number of 90-degree rotations is specified by ***rotate*** . Capture ColorID	`DisplayProperty "Part Reference", "Arial", 9, FALSE, FALSE, 48, 0`

		to RGB mapping.	
RemoveDisplayProperty	**RemoveDisplayProperty "** *name* "	Use this command to hide a display property of an object. The property to hide is specified by ***name*** .	`RemoveDisplayProperty "Part Reference"`
RemoveProperty	**RemoveProperty "** *name* "	Use this command to remove the property, specified by ***name*** , from the selected object or objects.	`RemoveProperty "Location"`
SetProperty	**SetProperty "** *name* ", "*value* "	Use this command to change a property value or add a new property with a specified value to the selected object. The property to be changed or created is specified by ***name*** . The new property value is specified by ***value*** . If more than one object is selected, then all objects are affected.	`SetProperty "Value", "74LS04"`
GetProperty	**GetProperty "** *name* ", "***value*** "	Use this command to fetch a property value for the selected object. The parameter ***name*** specifies the name of the property to be fetched. The parameter ***value*** is the variable that will store the property value.	`GetProperty "Cost", Temp$` where, • Cost is the property name. • Temp$ is the variable that stores the property value. ⚠ Make sure that the variable you define is followed by a $ symbol.

Viewing commands

Use the macro viewing commands to display Capture tools, go to a specific area of the schematic page, and zoom in or out of an area. Except where noted, these macro commands have no effect on the current location or the selection set.

In the following macro command descriptions, **Red** indicates a decimal number value. **Blue** indicates a Boolean value where zero means FALSE and non-zero numbers mean TRUE. **Magenta** indicates any character string.

Macro commands are reserved words, and should not be used for macro function or procedure names.

Function	Syntax	Description	Example
GoToAbsolute	**GoToAbsolute** ***X***, ***Y***	Use this command to set CurrentLocation to the coordinate (***X***, ***Y***).	`GoToAbsolute 0.00, 1.20`
GoToBookmark	**GoToBookmark "*name*"**	Use this command to set CurrentLocation to the upper left corner of the bookmark specified by the name ***name*** .	`GoToBookmark "Marker1"`
GotoGridReference	**GoToGridReference "*horizontal*", "*vertical*"**	Use this command to set CurrentLocation to the point specified by the horizontal grid reference ***horizontal*** , and the vertical grid reference ***vertical*** .	`GoToGridReference "A", "B"`
GotoRelative	**GoToRelative** *X*, *Y*	Use this command to set CurrentLocation to CurrentLocation + (***X***, ***Y***).	`GoToRelative 0.00, 1.20`
ViewGrid	**ViewGrid** ***on***	Use this command to display the grid. If ***on*** is TRUE, then the grid becomes visible. Otherwise, the grid is hidden.	`ViewGrid TRUE`
ViewGridReference	**ViewGridReference** *on*	Use this command to display the grid references. If ***on*** is TRUE, then the grid references become visible. Otherwise, the grid references are hidden.	`ViewGridReference FALSE`
ZoomAll	**ZoomAll**	Use this command to see the entire schematic page.	`ZoomAll`
ZoomArea	**ZoomArea** ***X1*** , ***Y1*** , ***X2*** , ***Y2***	Use this command to	`ZoomArea 2.40, 0.50, 5.10, 1.70`

		zoom in on the area specified by the rectangle with the coordinates the current location added to the offset (***X1***, ***Y1***), and the current location added to the offset (***X2*** , ***Y2***).	
ZoomIn	**ZoomIn**	Use this command to zoom in at CurrentLocation. The zoom is specified by the schematic page's zoom factor.	`ZoomIn`
ZoomOut	**ZoomOut**	Use this command to zoom out from CurrentLocation. The zoom is specified by zoom factor of the schematic page.	`ZoomOut`
ZoomSelection	**ZoomSelection**	Use this command to zoom in on the rectangle that contains all of the objects selected on the schematic page. If nothing is selected, then the command does nothing.	`ZoomSelection`

Working with Capture TCL

OrCAD Capture includes a scripting functionality that allows you to execute a Capture command through a command prompt. Capture also provides the facility to store and later replay the command.

Every Capture command is logged in the form of a TCL script command. This command that logged is registered with a TCL interpreter. When the command is played back, Capture uses the TCL interpreter to retrieve the command and execute it in Capture. However, this process is completely abstracted from the Capture. This makes logging and replaying of a set of commands an intuitive and simple task.

The user resources are then better utilized to actually identify the type of script (set of steps) that is required for a specific task.

The topics covered in this section are:

- Using the Capture Command window
- Creating a Capture TCL Script
- Executing a Capture TCL script

Using the Capture Command window

The Capture environment includes a Command window. You use this window to execute a TCL command. Also, when you perform an operation (function) in Capture, the associated command is registered with the TCL interpreter and the command is logged in the Command window.

To Open the Command window

1. Right-click on the Capture menu bar.
2. Choose Command Window from the short-cut menu.

The Command window displays with the capture> prompt.

> Any operation you perform in Capture is registered with the TCL interpreter and logged by Capture even if the Command window is closed.

To know about executing TCL command, see:

- Registering and Logging TCL Commands
- Executing a TCL Command

Registering and Logging TCL Commands

When you perform an operation in Capture, the associated command is registered with the TCL interpreter and the command is logged in the Command window.

Menu Commands

In Capture, you perform operations that include menu commands. These commands are logged with the TCL interpreter as menu commands.

For example, to open a new design you choose *File - New - Design*.

As soon as you complete the menu selection, the associated command is registered with the TCL interpreter and the command is logged in the command window as:

```
Menu “File::New::Design”
```

Similarly, the command to open the part editor for a selected part on a schematic is:

```
Menu “Edit::Part”
```

Notice, when you perform any operation that is associated with a Capture menu, the associated command syntax includes the text Menu followed by the menu selection.

You can use this procedure to easily identify the TCL command associated with any operation in Capture.

Page Editor Commands

You can also log and register any operation in a schematic page.

For example, if you select or un-select all the objects on a schematic page, the associated TCL commands are:

```
SelectAll
```

```
UnSelectAll
```

Similarly, the following command will place a wire with the co-ordinates of the start point at 1, 3 and end point at 6, 3:

```
PlaceWire 1 3 6 3
```

Project Manager Commands

You can log and register command for operations performed on the Project manager. For example, the following command will select the project item with the name Page1:

```
SelectPMItem "Page1"
```

The command to open a page Page1 in the schematic folder BENCH is:

```
OPage “Page1” “BENCH”.
```

Similarly, to identify a command, you perform the required operation in the Project Manager and the command in logged in the Capture Command window.

Executing a TCL Command

You use the Command window in Capture to execute a TCL Command. To execute a TCL Command in the Command window, you simply type the command in the Command window and press Enter. If an error occurs in executing the command, the error message displays in the Command window.

> ☑ All TCL commands are case-sensitive.

Besides the Capture TCL commands, you can also execute all the native TCL commands from the Capture Command window.

For example, the following native TCL command returns the absolute path of the current (present) working directory:

```
pwd
```

And to change the working directory, use:

```
cd
```

Creating a Capture TCL Script

You can create a Capture TCL script manually, in any text editor, or by executing the series of steps in Capture, and allowing Capture to record the associated commands. When you allow Capture to record TCL commands corresponding to the Capture commands, you can create a:

- TCL Script for all Capture commands
- TCL script for specific Capture commands

TCL Script for all Capture commands

When you perform an operation in Capture, the associated TCL command is registered with the TCL interpreter. The command is logged to a Capture TCL file. In addition, the command displays in the Capture Command window.

To enabled logging each command that you execute in Capture to a TCL script file, you need to execute the SetOptionBool command in the Capture Command window.

```
SetOptionBool Journaling TRUE
```

This is because, in Capture, by default, the TCL logging feature is disabled.

Also, if you are in the TCL command logging mode, you can view each executed Capture command in the Command window. Use the SetOptionBool command to enable viewing of the corresponding TCL command for each Capture command in the Command window.

```
SetOptionBool DisplayCommands TRUE
```

If you enable the display of TCL commands, the TCL commands, for each Capture command, are displayed in the session log. However, one Capture command corresponds, in some cases, to a set of multiple TCL commands. Also, the Capture operation is performed only after the corresponding TCL commands are displayed in the session log. This implies that if a Capture command corresponds to a number of TCL commands, this may cause a delay in completing the Capture command. To avoid this display, you can turn off the display functionality using the `SetOptionBool` command using the FALSE argument.

```
SetOptionBool DisplayCommands FALSE
```

For this reason,command display in the Command window is disabled, by default.

Location of the TCL file

If you enable TCL logging in Capture (using the SetOptionBool command), a TCL file, for each Capture session, is created in the Temp\CAPTURELOG directory.
The location of the Temp directory is defined in the following order of preference:

- The path specified by the TMP environment variable
- The path specified by the TEMP environment variable
- The path specified by the USERPROFILE environment variable
- The Windows directory

This directory contains the Capture TCL file OrCaptureLogFile.captcl along with support files required to execute the script.

Capture Dialog settings in TCL

When you perform an operation in Capture that involves a dialog box (like Options - Preferences), the Capture TCL logging feature saves the complete dialog settings to an Xml file. This file is placed in the same location as the current session TCL file.

For example, to annotate a design you use the Annotate dialog box.

The associated TCL command for this operation is:

```
Menu "Tools::Annotate" | DialogBox "OK" "C:/Temp/CAPTURELOG/Wed_Nov_18_14_34_45_2009/Packaging_6.xml"
```

Notice that an Xml file is one of the arguments in the TCL command. Also, notice the location of the Xml file is the same as the location of the current Capture session TCL file.

The location of the CAPTURELOG directory in the above example is `C:\Temp`. However this may change depending on the conditions as described in the *Location of the TCL file* section.

TCL script for specific Capture commands

When you turn on TCL script logging, the TCL commands for every Capture command that you execute are logged to the TCL session file. However, you can also create a TCL script with a specific set of Capture commands.

To create a script of a set of Capture commands

1. Right-click in the Capture Command window and choose Clear All.
2. Perform the Capture steps for which you want to record the corresponding TCL commands.
3. After completing the set of Capture steps, right-click in the Command window and choose Save.
4. In the Save TCL Script dialog, specify the name and location of the script.

The resultant script also includes the final steps to save the TCL file. You can open the script in a text editor to edit any extra steps that may have been recorded.

If you save a specific set of Capture commands, any support files required to execute these commands are not saved along with this script. For example, the dialog settings that are saved to an Xml will not be saved along with the script. In this case, the settings `Xml` is saved to the session TCL file location. For information on TCL file locations, see TCL Script for all Capture commands.

Executing a Capture TCL script

After creating a Capture TCL script (for details see Creating a Capture TCL Script), you can execute the script to replay the set of commands defined in the script. You can execute a TCL from the Capture Command window or from the Windows Command line.

- Execute TCL script in Capture
- Execute TCL script in Windows

Execute TCL script in Capture

You can execute a TCL script directly from the Capture Command window by the source command.

```
source <Script Path and Name>
```

For example, to execute the script, `D:\Cadence\tclsamples\selectObjs.tcl`

```
source D:/Cadence/tclsamples/selectObjs.tcl
```

- To specify a file location in a TCL command argument, you need to use the / (forward slash) path separator.

  Alternatively, if you use the \ (backward slash) path separator, you can enclose the argument in curly braces to ensure that argument is treated as a literal.

```
source {D:\Cadence\tclsamples\selectObjs.tcl}
```

- If a script file path contains spaces, you need to enclose the path in double-quotes. Also, you must use the / (forward slash) path separator and exclude the curly braces.

If you execute a TCL script in Capture that contains errors, the script exits after it encounters the first error and the error is displayed in the Capture session log. Also, you can use the ? alias to get details of the last TCL error. When you type ? in the Command window and press Enter, the last error encountered by the TCL interpreter is displayed in the Command window.

Execute TCL script in Windows

You can execute a TCL script directly from the Windows command line, without explicitly launching Capture to execute the script.

For this, you need to pass the script name and path as a command line argument to Capture. This will ensure that Capture will open and all the steps defined in the script will execute.

To execute the script from the command line:

```
capture <Script Path and Name>
```

After the script executes, it opens as a text file in edit mode in Capture.

The path separator rules to execute TCL script from the Windows command line are the same as the rules for executing a script from the Capture command window. For details see Execute TCL script in Capture

You can create nested TCL scripts by including the TclScript command to call a TCL script from within another TCL script.

13

Processing the Design

The topic covered In this section are:

- Annotating the Design
- Generating Reports
- Printing and plotting
- Netlisting a Design

Annotating the Design

By annotating your design (that is, by assigning reference designators and net names to unnamed parts and electrical connections in your design), you provide the means by which to pass it "downstream" to other layout design tools (PCB Editor, for example) that take it beyond the schematic capture phase of the design. Possible downstream applications include PCB layout tools, simulators, or logic optimizers.

For hierarchical PCB designs, you can choose to perform design level annotation by selecting the Design Level (Only PCB Designs) option in the Miscellaneous tab of the Preferences dialog box. This option is not selected by default.

If you want to preserve a reference designator, choose *User Assigned Flag – Set* from the pop-up menu of the part on the schematic page. You can also open the part in Properties Editor and from the pop-up menu of Reference choose *User Assigned Flag – Set*. The User Assigned Flag is set by the tool if the reference designator is changed using the Property Editor or Schematic Editor or through Back annotation.

Note that if the instance and occurrence values are the same in a hierarchical design, the User Assigned Flag is not set for the occurrence value. If you want to set the User Assigned Flag for the occurrence value, edit the Reference and then set the flag.

The topics covered are:

- Customizing Part References in a Design
- Back Annotating
- Forward Annotating Schematic Information
- Choosing the Annotate Sequence
- Designating Pins, Gates, or Packages for Swapping
- Creating an update file
- Creating a Combined Swap and Update File

Customizing Part References in a Design

You can customize the way Capture assigns part references in your design. You can specify a range of part reference values that Capture will use to annotate a schematic page or a hierarchical block in your design. Use the Annotate dialog box to complete this task.

> This functionality works independently from the existing annotation behavior of Capture.

Schematic page-wise and hierarchical block-wise annotation

In schematic page-wise annotation (recommended for flat design), you can set a part reference range for each schematic page that exists in the root schematic folder of your design. All the schematic pages in the root schematic folder are displayed in a grid in the Annotate dialog box in the following format: *Schematic_Folder_Name:Schematic_Page_Name* .

In hierarchical block-wise annotation (recommended for hierarchical design), you can set a part reference range for each hierarchical block that exists in root schematic folder of your design. All hierarchical blocks in the root schematic folder are displayed in a grid in the Annotate dialog box in the following format: *Schematic_Folder_Name:Hierarchical_Block_Name*.

If you set the design for schematic page-wise or hierarchical block-wise annotation and also use the:

- Incremental reference update option, then all the part references in your design are updated incrementally within the specified part reference range. However, this does not affect the already annotated parts in your design.
- Unconditional reference update option, then all the part references in your design are updated unconditionally from the start value specified in the part reference range.

Additionally, Capture will flash error and warning messages, if it encounters any invalid operation while using this functionality. For example, same part reference range for more than one schematic page or hierarchical blocks in your design or parts in your design that are outside the specified part reference range.

To perform schematic page-wise or hierarchical block-wise annotation

1. In the project manager, select the design file, schematic folder, or a schematic page.

   > Capture currently does not support the specifying of part reference range for a portion of the design, for example, a specific schematic page or a schematic folder. You should specify a part reference range for all the schematic pages or hierarchical blocks in your design.

2. From the *Tools* menu, select the *Annotate* command. The Annotate dialog box appears.

   > To perform regular annotation, see To uniquely identify parts.

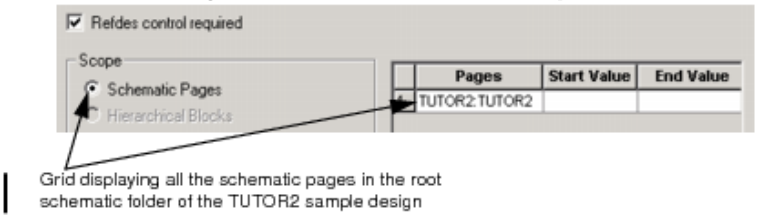
3. Select the *Refdes control required* check box, if you want to specify a part reference range for each schematic page or a hierarchical block in your design. The Scope options in the dialog box changes to Schematic Pages and Hierarchical Blocks. Also, a grid appears on the right-hand side of the dialog box displaying all the schematic pages or hierarchical blocks in the root schematic folder of your design depending on whether your design is a

flat design or a hierarchical design (see figure: *Annotate Dialog Box - Refdes control required*).

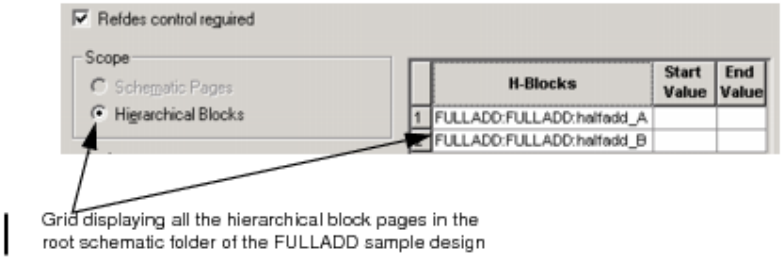
> The *Hierarchical Blocks* option is not available, if there are no hierarchical blocks in the root schematic folder of your design.

The grid is divided into rows and columns. Each row has a number, a schematic page name or a hierarchical block name, and cells to specify the Start and End values for the part reference range.
For example, the *TUTOR2.dsn* sample design, which is a flat design, contains a schematic page defined at the root level of the design. When you select the Refdes control required check box, then the grid displays the following entry in the Pages column: *TUTOR2:TUTOR2*, where the first *TUTOR2* is the root schematic folder name and the second *TUTOR2* is the schematic page name.

Figure: **Annotate Dialog Box - Refdes control required**



Take the example of a hierarchical design. The *FULLADD.DSN* sample design, which is a hierarchical design, contains two hierarchical blocks defined at the root level of the design. When you select the Refdes control required check box, then the grid displays the following two entries in the H-Blocks column: *FULLADD:FULLADD:halfadd_A* and *FULLADD:FULLADD:halfadd_B*, where *FULLADD* is the root schematic folder name and *FULLADD:halfadd_A* and *FULLADD:halfadd_B* are the reference to the root-level hierarchical blocks in the design.



> The grid displays only the schematic pages or hierarchical blocks on the root schematic of the design.

4. Enter a numeric value greater than 0 in the Start Value and End Value columns corresponding to each schematic page name or the hierarchical block name.

– Use the Tab key to move from the Start Value column to the End Value column.

– You can also use the Arrow keys to move around in the grid.

– You can use the column handle ( ) to resize the rows and columns in the grid.

– A valid range must have both the Start and End Values, and the End Value must be greater than the Start Value.

5. Specify all the other desired settings in the Annotation dialog box.
6. Click *OK*.
   Part references in each schematic page or hierarchical block get updated according to the range specified for them in the Annotate dialog box.

- Capture saves the annotation preferences for the project when you click OK. These settings will be used whenever you open the project the next time.
- If you are using the *Refdes control required* option for a project, then the Auto reference placed part option in the *Miscellaneous* tab of the *Preferences* dialog box will not honor the range specified in the grid.
- If any changes are made to the root schematic of a design, then all the part reference range values specified in the grid for the design will be lost. However, this does not affect the already annotated parts in the design.

Back Annotating

When you need to transfer packaging information to your schematic folder from other EDA tools, use the Back Annotate tool. When you need to back annotate properties, use the Update Properties tool (see To update part or net properties). Using Back Annotate, you can import changes created by external tools such as PCB layout packages. Capture uses a simple file format to provide support for gate swapping, for pin swapping, and for changing or adding properties on parts, pins, or nets. If the external tool creates a back annotation file, edit the file to match the format described in Designating pins, gates, or packages for swapping.

You might use Back Annotate, after you have completed your schematic design and while you are routing a printed circuit board, you discover that you could greatly reduce the via count, track length, or routing complexity by exchanging two of the gates or pins on a part. You could then use your board layout application to rewire the board, exchanging the connections of U1A and U1B. To ensure that your schematic design reflects the changes, you use a text editor to create a swap file, then run the Back Annotate command. The next time you look at the design, you see that U1A and U1B have traded places.

Back annotating board file information to your schematic design is a matter of creating a report file and reading it back into Capture.

To back annotate schematic information

1. After you have made changes to the design in the layout tool, choose *Reports* from the

File menu.
The Generate Reports dialog box appears.

2. If you re-annotated the names of parts, or altered parts or nets, choose OrCAD Back annotation File (.SWP) to create a combined swap and update file. Click *OK* to create the report.
3. From the project manager's *Tools* menu, choose *Back Annotate*.
   The Back Annotate dialog box appears.
4. Use the Browse button to find the file (.SWP) you created in step 2, then click *OK*.

Capture updates the schematic design.

To back annotate part packaging information

1. Using a text editor, create a swap file.
   See Designating pins, gates, or packages for swapping for details.
2. In the project manager, select schematic folders or schematic pages if you want to process only a portion of the design. If you want to process the entire design, leave the schematic folders or schematic pages unselected.
3. From the *Tools* menu, choose *Back Annotate*.
   The Back Annotate dialog box displays.
4. Select the tab (either PCB Editor or Layout) that is appropriate to your downstream PCB tool. Verify that the dialog box options are set the way you want them.
   For example, the name and location for the netlist directory and the swap file.
5. Click *OK*.

Shortcut

Toolbar: 

Forward Annotating Schematic Information

If you make changes to your board design in Capture, you can bring those changes into PCB Editor. In addition, you must save your Capture design before you can create a netlist.

To f orward annotate schematic information from Capture

Capture includes functionality with which you can forward annotate your schematic data, such that it can be included in a PCB Editor board design.

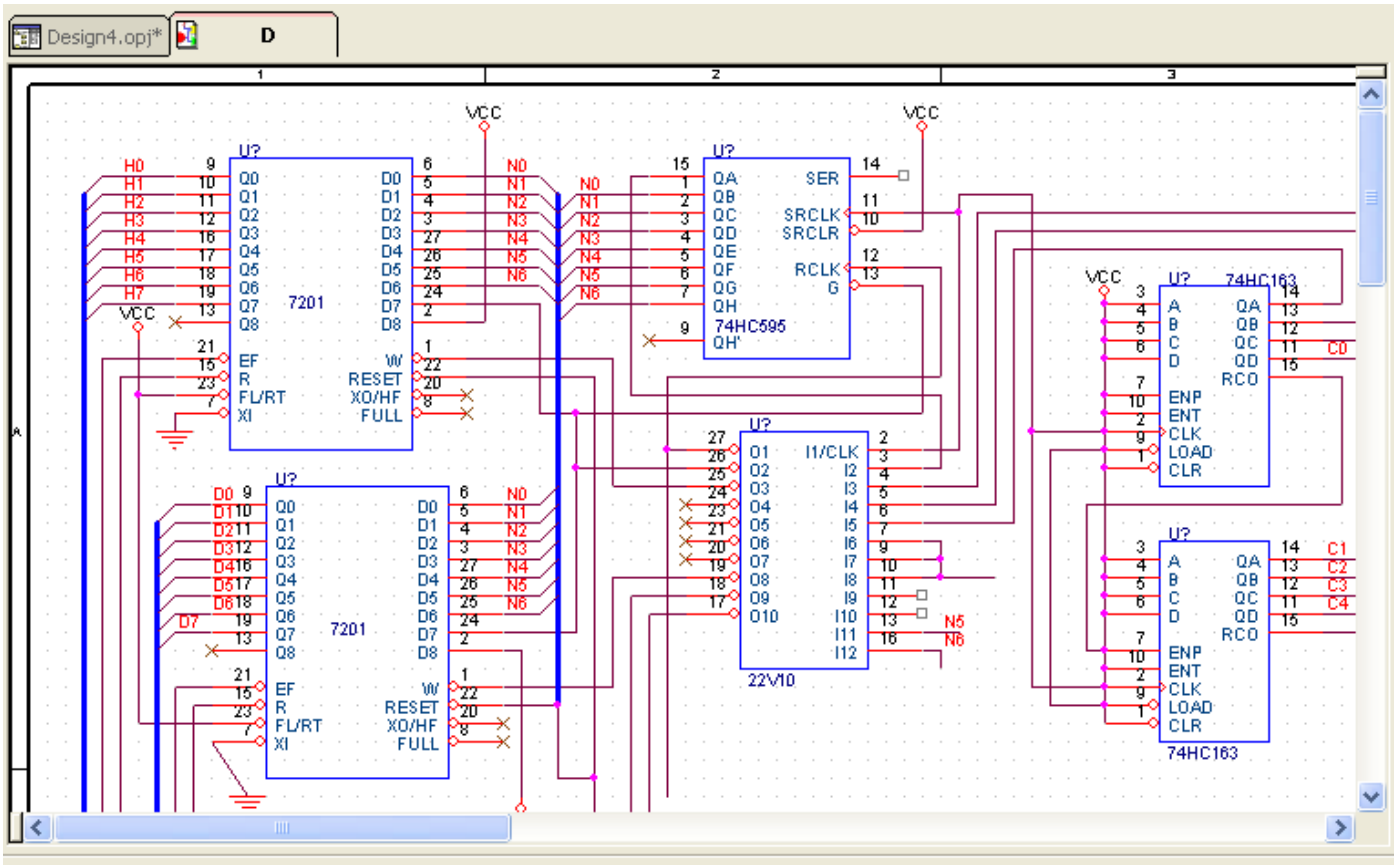
1. Open the design for which you are going to create a netlist.
2. From the *Tools* menu, choose *Create Netlist*.
3. In the Create Netlist dialog box, choose the *PCB Editor* tab.
4. In the Input Board File box, enter a name for the input file using a .BRD file extension.
5. In the Output Board File box, enter a name for the output file using a .BRD file extension.
6. Click *OK* to create the .BRD file.

Choosing the Annotate Sequence

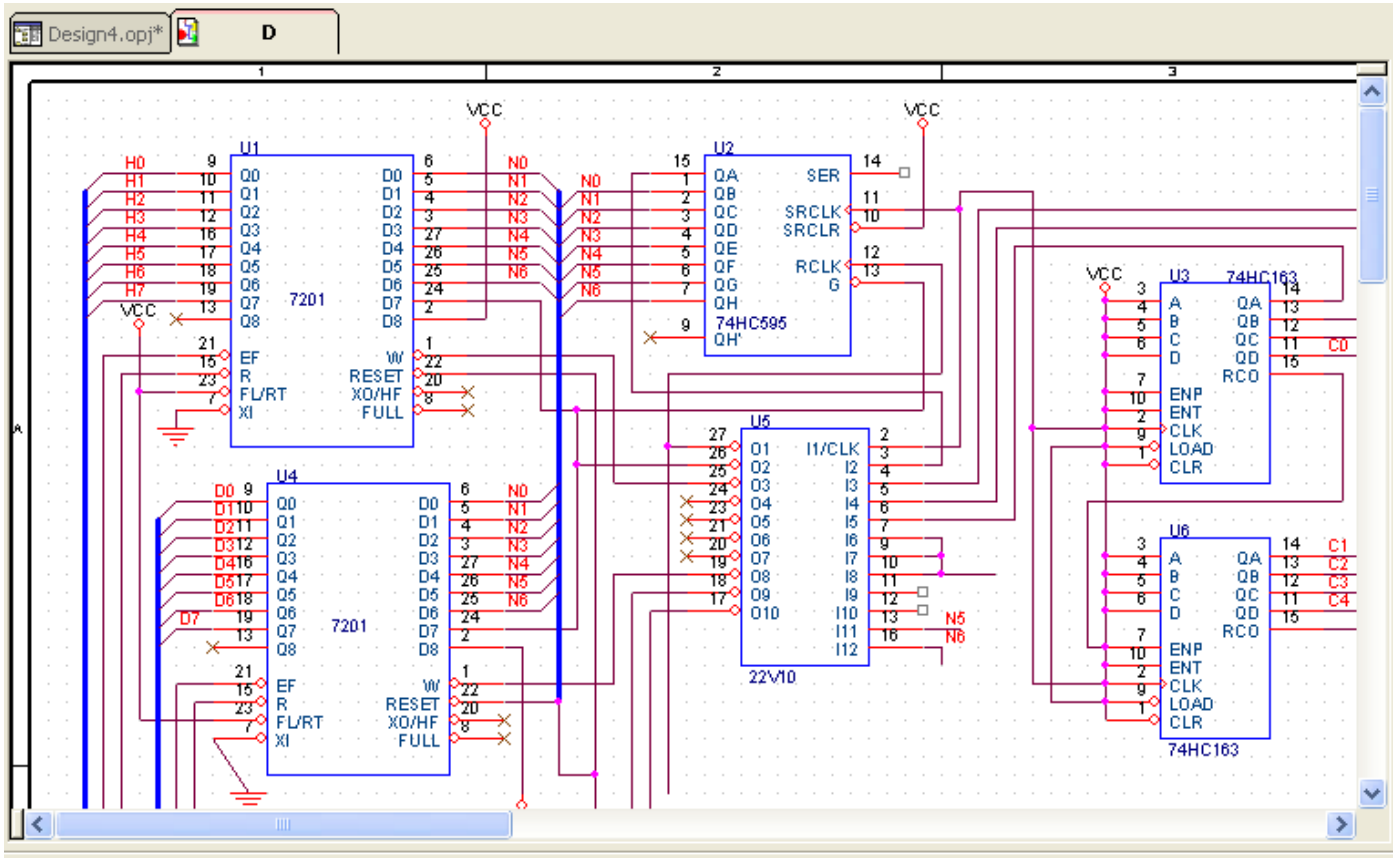
You can choose the sequence in which the components of your design are annotated.

The Annotation Sequence list contains three options that you can use to decide the sequence in which the objects on your design are annotated — Default, Left to Right, and Top to Bottom.

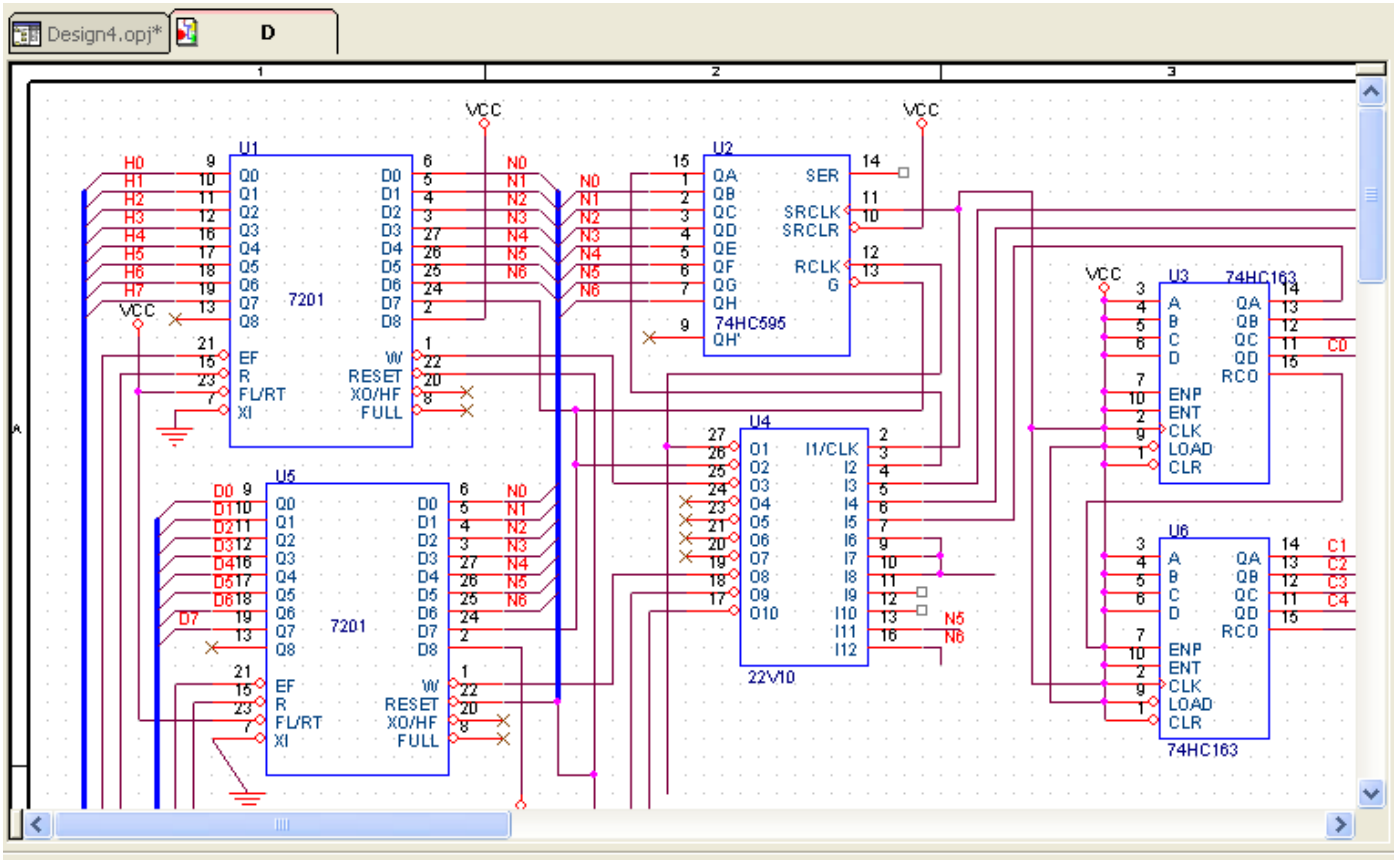
To differentiate between the way these sequences work on a design, we will use the example of a dense design that contains a number of large components clustered together.



If we run the annotate command on this design using the Default option, the results are as follows:

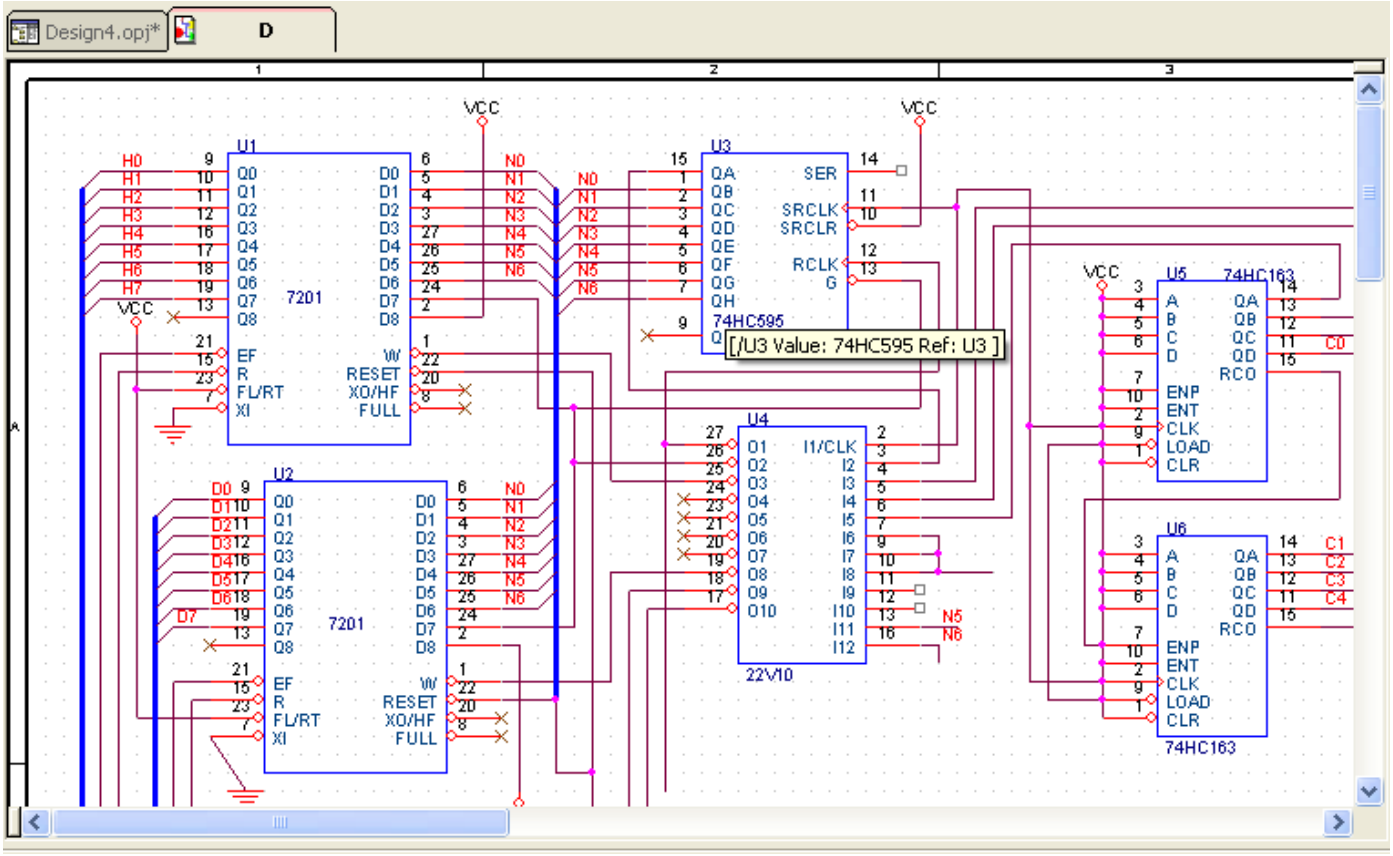


Notice the sequence of annotation of the components on the design.
Next, we will select the Left - Right option and annotate the design again.



The annotation procedure using this option scans the page grid from left to right one grid line at a time. Then moving downwards to the next grid line. In effect, the sequence is left to right and top to bottom.

We will now annotate the design using the Top - Bottom option.



In this case, the procedure scans the pages from top to bottom one grid line at a time. Then moving towards the left of the screen.

Designating Pins, Gates, or Packages for Swapping

Creating a swap file (.SWP)

For PCB designs, a swap file is a text file containing old and new part references for use with the Back Annotate command. Swap files are typically created by another application, such as PCB Editor. You can also create a swap (.SWP) file using any text editor that saves files in the ASCII format. The file can include comments; the Back Annotate tool ignores any text to the right of a semicolon.

In a swap file, each line (unless preceded by a semicolon) causes one action. The elements of each line may be separated with any number of space or tab characters. In general, the first element of the line specifies the type of swap. If no swap type is specified, CHANGEREF is assumed. The other swap types are GATESWAP and PINSWAP.

When you are creating a swap file, include only the changes from the present state of the design to the state you want it to have. For example, you might place a part as U1 in the design, and change it in a PCB layout package first to U2, then to U3. The swap file should reflect the change from U1 to U3; do not include the intermediate step involving U2.

For gate swaps, make sure that the gates being swapped are of the same type. If they are not, you may get incorrect results.

For pin swaps, an additional element — the part reference — must be specified before the old and new values. Pin swap is limited to pins of the same type and shape on the same part. For example, you can swap data pins on U5B, but you cannot swap a pin on U5B with a pin on U5C.

CHANGEREF

Changes the specified part's reference.

Examples

```
CHANGEREF U1 U2 ; Change part reference U1 to U2
CHANGEREF U1A U1B ; Change part reference U1A to U1B
U1C U2B ; Change part reference U1C to U2B
```

GATESWAP

Swaps the specified parts or packages. If U1 and U2 are multiple-part packages, then all the devices in U1 will change to U2, and vice versa---U1A, U1B, and U1C change to U2A, U2B, and U2C, respectively; U2A, U2B, and U2C change to U1A, U1B, and U1C, respectively.

Examples

```
GATESWAP U1 U2 ; Change part U1 to U2 and
  ; part U2 to U1
GATESWAP U1A U1B ; Swap gates A and B on U1
GATESWAP U1C U2B ; Swap gates U1C and U2B
```

> Back Annotate does not check part types before performing the specified swap. If you swap gates between parts of different types (as shown in the following example), you may see unwanted results in your design.
>
> GATESWAP U1C U2B ; Swap gates U1C and U2B

PINSWAP

Swaps two pins on the specified part. Only pins of the same type and shape on the same part can be swapped. The pins are identified by name or number. Pin names must be enclosed in double quotation marks. PINSWAP can be used multiple times on the same pins in a swap file.

Examples

```
PINSWAP U5B "D0" "D1" ; Swap the pins named D0 and D1 on U5B
PINSWAP U3 5 6 ; Swap pins 5 and 6 on U3
```

CHANGEPIN

Changes the first pin with the second pin. Only pins of the same type and shape on the same part can be swapped. The pins are identified by name or number. Pin names must be enclosed in double quotation marks. CHANGEPIN can only be used once on each pin in a swap file.

Examples

```
CHANGEPIN U5B "D0" "D1" ; Changes pin D0 to D1 on U5B
CHANGEPIN U3 5 6 ; Changes pin 5 to pin 6 on U3
```

To designate pins, gates, or packages for swapping

To specify that pins, gates, or packages in your PCB design are eligible for swapping (in order, to improve board routing) follow these steps.

1. Using a text editor, create a swap file. See *Creating a swap file (.SWP)* for instructions.
2. In the Project Manager, select schematic folders or schematic pages if you want to process only a portion of the design. If you want to process the entire design, leave the schematic folders or schematic pages unselected.
3. From the *Tools* menu, choose *Back Annotate*.
   The Back Annotate dialog box displays.
4. Verify that the dialog box options are set the way you want them. For example, you specify whether you want to process the entire design or only the selected schematic folders or schematic pages, and the name and location for the swap file.
5. Click *OK*.

Shortcut

Toolbar: 

Example

```
CHANGEREF U1 U2 ; Change part reference U1 to U2
CHANGEREF U1A U1B ; Change part reference U1A to U1B
  U1C U2B ; Change part reference U1C to U2B
GATESWAP U1 U2 ; Change part U1 to U2 and part U2 to U1
GATESWAP U1A U1B ; Swap gates U1A and U1B
PINSWAP U7 1 2 ; Swap pins 1 and 2 on U7
PINSWAP U5B "D0" "D1" ; Swap the pins named D0 and D1 on U5B
PINSWAP U3 5 6 ; Swap pins 5 and 6 on U3
```

Creating an update file

The update file is used by the Update Properties tool to determine which objects to change, which of the objects' properties are affected, and what values those properties receive. You can create an update (.UPD) file using any text editor that saves files in ASCII format. The file can include comments; any text to the right of a semicolon is ignored by the Update Properties tool. Strings in the update file (except for comments) must be enclosed in quotation marks and cannot exceed 124 characters. You can use spaces and tab characters to format the update file in rows and columns, as shown in the example below.

The first line of the update file is a header line. It starts with a combined property string that identifies which properties to compare. In the example, only the Net Name property is compared. The other strings on the first line specify which properties to update when a match is found. The rest of the file contains lines for each match string to be compared and the values to be recorded in the updated properties. In the following example, the combined property string is {Net Name}. For every object whose Net Name property value matches one of the strings in the first column, the object's Track Width, Net Spacing, and Routing Priority properties are updated with the corresponding values. For example, every object whose Net Name property is set to VCC will be updated as follows: the Track Width property is set to 0.04, the Net Spacing property to 0.035, and

the Routing Priority property to 3.

Combined property strings

With many of the tools in Capture, such as Create Netlist and Annotate, you use combined property strings to convey information to the tool or to limit the tool's action. A combined property string consists of one or more property names, enclosed in braces, and can also contain literal text. Capture combines the values of the named properties with any literal text to create a string. An example is:

```
{Value}{Reference}
```

where "Value" and "Reference" are property names. Using this combined property string and a part with a part value of 74LS32 and a part reference of U?A, Capture creates the string:

```
74LS32U?A
```

You can include spaces and other characters in the combined property string, as in this example:

```
Part: {Value} ({Reference})
```

Using this combined property string and the same part, Capture creates the string:

```
Part: 74LS32 (U?A)
```

Different tools use combined property strings in different ways. For example, Annotate uses one to compare parts—if one part's combined property string matches another part's combined property string, it packages the parts together.

- Bill of Materials and other commands on the Tools menu that generate an output file based on a combined property string may produce errors if you include extra curly braces.
- You can include tabs in combined property strings, so that the output file can be manipulated in a spreadsheet or database application. Tabs also help format report files, such as those created by the Bill of Materials command. Wherever you want to have a tab in the output file, insert the characters \t (a backslash and a lowercase "t") in the combined property string.
- Do not use {GROUP} as a property name in combined property strings. This may cause problems while annotating your design for a PCB Editor tool, like Allegro PCB Editor. The GROUP property is used in PCB Editor for a specific purpose.
- Except for the header line, update files for Capture have the same format as the update files used by Update Field Contents in SDT. Once you add the appropriate header line, you can use an SDT update file without modification to change individual property values using Update Properties. You can also build on an SDT update file for a more elaborate property update, because Capture can update multiple properties in a single pass.

Example :

```
"{Net Name}" "Track Width" "Net Spacing" "Routing Priority"
"VCC" "0.04" "0.035" "3" ; Any text to the right
"CLK" "0.01" "0.025" "1" ; of a semicolon is
"CLR" "0.01" "0.025" "3" ; ignored by the
"RESET" "0.01" "0.025" "3" ; Update Properties
"GND" "0.04" "0.035" "2" ; tool.
```

> ⚠ Do not create an empty string (two consecutive double quotation marks without intervening characters) in the header line. Update Properties reports an empty string as an error.

Creating a Combined Swap and Update File

You can create a file that combines the swap file and update file information. Run *Back Annotate* to use a combined swap and update file. Swap and update files should have the same .SWP file extension as normal swap files.
A swap and update file is divided into sections. Each section uses the following general form:

```
.label utility-name utility-parameters
section-information
.End
where:
```

label

Specifies the name of a section. A label can be any string combination of letters and numbers, but must always start with a dot (.).

utility-name

Specifies the utility Capture uses for the section.

Section	Definition
Flags	Marks the view of the design, and the design name and path.
GateAndPinSwap	A section using this utility behaves the same as a Back Annotate file.
UpdateProperties	A section using this utility behaves the same as an Update Properties file.

utility-parameters

Specifies the parameters of the utility for the section. All utility parameters only apply to the UpdateProperties utility.

Parameter	Defintion
Parts	Specifies that only parts are updated. If both Parts and Nets are specified as parameters, Nets overrides Parts.

Nets	Specifies that only nets are updated. If both Parts and Nets are specified as parameters, Nets overrides Parts.
OnlyStuffEmpties	Specifies that only empty properties are updated. If a property already contains a value, then it is not modified. If this parameter is not specified, UpdateProperties unconditionally updates all properties.
UppercaseCombined	Specifies that a case insensitive match is to be attempted when comparing the match string with the combined property string.
UppercaseStuffString	Specifies that the update string will be changed to upper case before updating the property, but after string matching.

section-information

Specifies the actions the utility performs for the section. If the utility is GateAndPinSwap, these lines use the normal Gate and Pin Swap file format. If the utility is UpdateProperties, these lines use the normal Update Properties file format. If the utility is Flags, the section contains the following lines:

```
View = design-view
DesignName = path
```

where design-view is either Logical or Physical, and path specifies the design file name and path. In Capture Release 9 and later, the Logical design-view corresponds to instances, and Physical corresponds to occurrences.

Example

```
.Label1 Flags
  View = Physical
  DesignName = C:\ORCADWIN\CAPTURE\DESIGN\FULLADD.DSN
End
.Label2 GateAndPinSwap
  GateSwap R1 R2
  PinSwap R3 "1" "2"
  ChangeRef R4 R10
End
.Label3 UpdateProperties Parts
  "{Part Reference}" "Notes"
  "R10" "This used to be R4"
  "R3" "Notice pins 1 and 2 are swapped"
  "R1" "This used to be R2"
  "R2" "This used to be R1"
End
.Label4 UpdateProperties Nets
  "{Net Name}" "Trace Width"
  "VCC_WAVE" "0.040"
  "GND Power" "0.040"
End
```

Generating Reports

In this section:

If you have not specified a root for your design, you cannot generate reports.

Creating an Include File

You can use an include file to have the `Bill of Materials` command add information that is not in the schematic folder to the final bill of materials. You can create an include (.INC) file using any text editor that saves files in the ASCII format.

The first line of the include file is a header. The bill of materials is normally keyed to the part value, so the first line begins with a pair of single quotes with no spaces or other characters between them. The rest of the first line contains any information you want to include to make the file and the bill of materials more readable—this usually consists of headers for the values in the rest of the file.

The rest of the file contains a separate line for each part. Each line must begin with the property value (as specified in the Combined property string field within the Include File group box in the Bill of Materials dialog box) enclosed in single quotes. Following the property value (and on the same line) is the information that you want added to the bill of materials. You can separate the part value from additional information by inserting any number of spaces or tab characters—Capture will align the first non-blank character in each line when it creates the bill of materials report.

You must separate the items in the Combined property string field in the Bill of Materials dialog box exactly as they are separated in the include file. For example, if you use a space to separate the part values, descriptions, and part orders, then the combined property string should look like this:

```
{Value}
```

Screws, washers, and other hardware appear in a bill of materials, but not in a netlist. Netlists include only objects with pins.

Example

```
''     DESCRIPTION             PART ORDER CODE
'1K'   Resistor 1/4 Watt 5%    10000111003
'4.7K' Resistor 1/4 Watt 5%    10000114703
'22K'  Resistor 1/4 Watt 5%    10000112204
'1uF'  Capacitor Ceramic Disk  10000211006
'.1uF' Capacitor Ceramic Disk  10000211007
```

Include files for Capture have the same format as the include files used by Create Bill of Materials in OrCAD's SDT 386+. You can use an SDT 386+ include file without modification to create a bill of materials in Capture.

Creating a Bill of Materials

A bill of materials is a composite list of all the elements you need for your PCB design. Using the Bill of Materials command, you can create a standard tab-delimited part list, or you can create a custom bill of materials showing properties that you specify. With either of these formats, you can add information about any part by merging an include file with your bill of materials. A standard bill of materials includes the item, quantity, part reference, and part value.

Capture automatically elects to use either instances or occurrences for generating reports, depending upon your type of design. In general, you should use instances for FPGA and PSpice projects, and use occurrences for PCB and Schematic projects.

You can create non-electrical parts—such as screws, washers, and sockets—that will appear in a bill of materials report but not in a netlist because the non-electrical parts do not have pins. Any part without pins is considered non-electrical.

You can specify any header information you want. The header of a bill of materials usually contains information such as the design name, date, document number, revision code, report name, page number, and the time the report is created. If the Header field contains only a single space character, the header is left blank.

Bill of Materials and other commands on the Tools menu that generate an output file based on a combined property string may produce errors if you include extra curly braces {}.

Capture report files are text files, and can be opened in any text editor. You may want to use the tab alignment capability of your word processor to line reports up correctly. Spreadsheets will automatically align the columns of Capture-generated report files.

To create a Bill of Materials

1. In the project manager, select schematic folders or schematic pages if you want to process only a portion of the design. If you want to process the entire design, leave the schematic folders or schematic pages unselected.
2. From the *Tools* menu, choose the Bill of Materials command.
   The Bill of Materials dialog box appears.
   Verify that the dialog box options are set the way you want them.
   For example, you specify whether you want to process the entire design or only the selected schematic folders or schematic pages, and the name and location for the report file.
   (If you want to customize the information contained in the bill of materials report, see *To create a custom bill of materials* below.)
3. Click *OK*.

The Bill of Materials command generates report files with .BOM extensions.

To merge information from an external database

1. Create an include file (see Creating an include file).
2. Perform the steps listed in *To create a bill of materials* section and set these additional options in the Bill of Materials dialog box.
    - Select the Merge an include file with report option.
    - Specify a combined property string.
    - Specify the path and name of the include file.

To create a custom bill of materials

- Perform the steps in *To create a bill of materials* section and set these additional options in the Bill of Materials dialog box:
    - In the Header field, enter the column headings you want in the report. If you leave the Header field blank, there are no column headings in the report.
    - In the Combined property string field, enter the names of the properties you want in the report. If you want the property values separated by literals, include the literals in the field. See Defining properties for more information.

To import the bill of materials in Microsoft Excel

1. Select the *Open in Excel* check box.
2. Click *OK*.
   The bill of materials is displayed in the Microsoft Excel spreadsheet.

Shortcut

Toolbar: 

Example

1 Bit Full Adder Hierarchy (COMPLEX) Revised: March 31, 1999 OrCAD
Bill Of Materials March 31, 1995 16:50:31 Page 1

Item	Quantity	Reference	Part
1	1	U1	74LS32
2	3	U2	74LS08
		U2	74LS08
		U2	74LS08
3	2	U3	74LS04

		U3	74LS04

Creating a Cross Reference Report

The Cross Reference tool creates a report, indexed by schematic page, of all the parts with their part references, part names and libraries. You may specify that the report should also list the unused parts in multiple-part packages and the coordinates of all parts.
Capture automatically selects either instances or occurrences for generating reports depending upon the type of design with which you are working. In general, you should use instances for FPGA and PSpice projects, and use occurrences for PCB and Schematic projects.

To create a cross reference report

1. In the project manager, select schematic folders or schematic pages if you want to process only a portion of the design. If you want to process the entire design, leave the schematic folders or schematic pages unselected.
2. From the *Tools* menu, choose the Cross Reference command.
   The Cross Reference Parts dialog box displays.
3. Verify that the dialog box options are set the way you want them. For example, you specify, among other things, whether you want to process the entire design or only the selected schematic folders or schematic pages, whether the parts are sorted by part value or by part reference, and the name and location for the report file.
4. Click *OK*.

- The Cross Reference command generates report files with .XRF or .CSV extensions.
- Capture report files are text files and can be opened in any text editor. You may want to use the tab alignment capability of your word processor to line reports up correctly. Spreadsheets will automatically align the columns of Capture-generated report files.
- The path listed in the Library column of the Cross Reference report represents where the part placed was found when it was originally placed. If you change machines, move or delete the library, or rename the part, the information in the report won't correspond to the current path and file name.

Shortcut

Toolbar: 

Example

```
1 Bit Full Adder Hierarchy (COMPLEX) Revised: March 31, 1995 OrCAD
Design Name: C:\CAPTURE\DESIGN\FULLADD.DSN
Cross Reference March 31, 1995 16:15:54 Page 1
```

Item	Part	Reference	SchematicName	Sheet	Library
1	74LS04	U3A	HALFADD	1	C:\WINDOWS\TEMP\TTL.OLB
2	74LS04	U3B	HALFADD	1	C:\WINDOWS\TEMP\TTL.OLB
3	74LS08	U2A	HALFADD	1	C:\WINDOWS\TEMP\TTL.OLB
4	74LS08	U2B	HALFADD	1	C:\WINDOWS\TEMP\TTL.OLB
5	74LS08	U2C	HALFADD	1	C:\WINDOWS\TEMP\TTL.OLB
6	74LS32	U1B	HALFADD	1	C:\WINDOWS\TEMP\TTL.OLB

Creating a Placement Report

You can generate a report of the X and Y locations of the placements of the parts on a schematic. This report, generated as a .CSV file, provides the following details of the parts:

- reference designator
- part name
- schematic name
- sheet number
- file system location of the part library
- X co-ordinate location
- Y co-ordinate location

To Create a Placement report

1. You can create this report at any level of a design hierarchy: design, folder or page. Therefore, you need to select the appropriate level of the hierarchy in the part manager.
2. Right-click on the Project manager.
3. Point to Reports in the pop-up menu and choose the Export Placement item.

The Placements.csv file is created at the same file system location as the current design and simultaneously opened for viewing.

> While the report is generated at any level of the design hierarchy, you can also multi-select parts of the hierarchy to generate the report. For example, you can select multiple pages to generate a placement report of only the parts on the selected pages.

Creating a Find Result Report

You can create a report for the results of the Find command. This report can be output in either the CSV or HTML formats.
Also, when you run the Find command to search for different types of objects, the search results appear in different types of the Find window. In this case, you can export the data from each tab.

To create a Find result report

1. Execute the *Find* command.
2. Right-click on any line item in the Find window.
3. From the pop-up menu, choose, *Save as HTML* or *Save as CSV*.
   A pop-up dialog displays the file system location of the report.
4. Click *OK*.

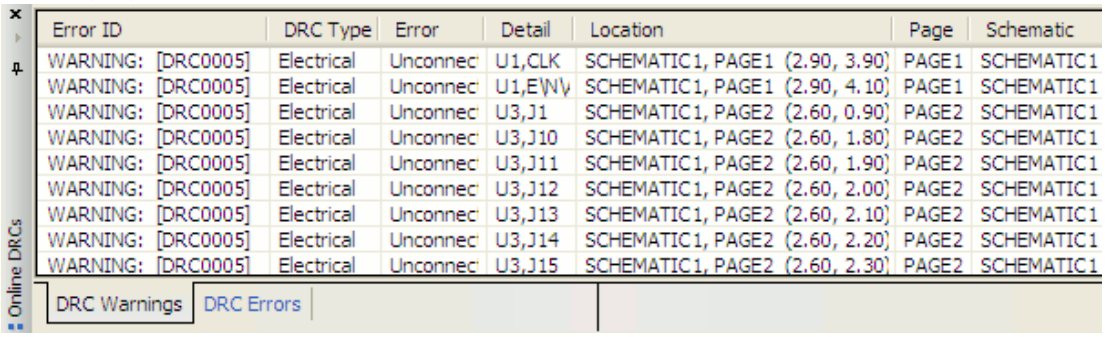
> The report does not open when you run the command. You need to go to the specified location and open the report in the associated application.

Check and Save

After you run the design rule check at least once on a design, you can then use the Check and Save command. This command executes the Design Rules Check on the current set of design rules that you define in the Design Rules Check dialog box.

To run the Check and Save command

1. Click the *File* menu.
2. Choose the *Check and Save* option.
   The Online DRCs window displays two tabs with the listings of the DRC errors and warnings generated from the DRC check.



Error ID	DRC Type	Error	Detail	Location	Page	Schematic
WARNING: [DRC0005]	Electrical	Unconnec	U1,CLK	SCHEMATIC1, PAGE1 (2.90, 3.90)	PAGE1	SCHEMATIC1
WARNING: [DRC0005]	Electrical	Unconnec	U1,E\N\	SCHEMATIC1, PAGE1 (2.90, 4.10)	PAGE1	SCHEMATIC1
WARNING: [DRC0005]	Electrical	Unconnec	U3,J1	SCHEMATIC1, PAGE2 (2.60, 0.90)	PAGE2	SCHEMATIC1
WARNING: [DRC0005]	Electrical	Unconnec	U3,J10	SCHEMATIC1, PAGE2 (2.60, 1.80)	PAGE2	SCHEMATIC1
WARNING: [DRC0005]	Electrical	Unconnec	U3,J11	SCHEMATIC1, PAGE2 (2.60, 1.90)	PAGE2	SCHEMATIC1
WARNING: [DRC0005]	Electrical	Unconnec	U3,J12	SCHEMATIC1, PAGE2 (2.60, 2.00)	PAGE2	SCHEMATIC1
WARNING: [DRC0005]	Electrical	Unconnec	U3,J13	SCHEMATIC1, PAGE2 (2.60, 2.10)	PAGE2	SCHEMATIC1
WARNING: [DRC0005]	Electrical	Unconnec	U3,J14	SCHEMATIC1, PAGE2 (2.60, 2.20)	PAGE2	SCHEMATIC1
WARNING: [DRC0005]	Electrical	Unconnec	U3,J15	SCHEMATIC1, PAGE2 (2.60, 2.30)	PAGE2	SCHEMATIC1

3. Double-click on a error or warning in the listings to go to the DRC marker on the specific schematic page.

> You need to setup and run the design rules check at least once before using the Check and Save command.

Printing and plotting

Whether you wish to send output to a printer, a plotter, or an encapsulated PostScript file, you work with the standard Windows dialog boxes. Capture can send output to any driver that Windows supports.
In the Print dialog box, you make choices for a print job. The choices you establish are used for previewing and for creating output. Each time you open the Print dialog box to request a print job, all choices will be reset to the default settings.

For additional information on printing and plotting, see your Windows documentation.

> At certain zoom scales, Capture substitutes filled rectangles for text that is too small to appear. These placeholders are for display only—the text prints correctly.

To print or plot

1. Open the schematic page, part, or symbol you wish to print.
2. From the *File* menu, choose the *Print* command.

In this section:

Printer or Plotter Setup

There are some considerations you should take into account when printing or plotting from Capture.

Special considerations for plotters

Plotters do not support bitmaps directly. If you are sending Capture output to a plotter, your bitmaps will not be plotted.
The Capture setup command may not give you access to all your plotter's setup options. For access to additional printer settings, use the Printers icon in the Windows Control Panel.
Many plotters do not have drivers that ship with Windows. If you do not see the plotter you are looking for in the list of available drivers, contact your plotter manufacturer and ask for a Windows driver. If your plotter will emulate HPGL, an alternative solution is to use the HPGL driver.

Plotter pen colors

The plotter driver maps your color choice to the closest available pen color as established in your plotter driver configuration. See your plotter's driver setup and documentation for more details. For access to additional printer settings, use the Printers icon in the Windows Control Panel.

To set up a printer or plotter

1. From the Print dialog box, choose the *Setup* button to select a different printer or plotter or to change printer settings.
2. If you need to set up your printer or plotter, see the documentation that accompanies the printer or plotter. For access to additional printer settings, use the Printers setting on the *Start* menu.

Setting Print Options for Schematic Page Objects

You can specify various options for determining how specific objects on the schematic page are printed.

To define if an object is printed or plotted

1. From the *Options* menu, choose *Preference*.
2. Go to the *Colors/Print* tab.
3. Select the check box located beside the color block for the object that you want to be able to print or plot.
   Clear the check box if the object is not to be printed or plotted.

> ⚠ Objects are always displayed on the schematic page, regardless of the setting of their check boxes.

Printing Documents

Documents can be printed, as indicated in the following table:

Document to print	From Project Manager	From Schematic Editor	From Part Editor	From Text Eindow
Single schematic page	Yes	Yes	---	---
Multiple schematic pages	Yes	---	---	---
Entire schematic folder	Yes	---	---	---
Single part or symbol	Yes	---	Yes	---
Multiple parts and symbols	Yes	---	Yes (a)	---
Entire package	Yes	---	Yes (b)	---
Text	Yes (c)	---	---	Yes

> ⚠ (a) Multiple-part package in package view, only.
>
> (b) In package view.
>
> (c) Not including the session log.

To print a document

1. In the Project manager, select the document to print
   To print more than one document, use the *Ctrl+click* combination.
2. From the *File* menu, choose the *Print* command.

3. Choose either to print the highlighted text or the entire document.
4. Click *OK*.
   The "Printing" message appears. Capture prints the window.

> When printing a multi-page schematic, make sure that the pages do not have multiple Title Blocks with different page numbers. Otherwise, the pages will not be printed in the correct order. If you change the page numbers in the Title Blocks manually, then make sure that the *Do not change the page number* checkbox is checked in the Annotate dialog box.

To print from a text editor window

1. Open the text editor window to print.
2. From the *File* menu, choose *Print*.
3. Choose either to print the highlighted text or the entire document.
4. Click *OK*.
   The "Printing" message appears. Capture prints the window.

Shortcut

Toolbar: 

Printing or plotting a schematic page

With the schematic page editor active and open to a specific schematic page, you can create a print or a plot of that page. You can also print a page from the project manager.
Windows normally sets the printer to the Portrait mode. You can use the Print Preview command to check the output before sending it to the printer or plotter.

Behavior of offsets in printing and plotting

Your entire schematic page will be output to the printing or plotting device, regardless of the use of offsets.

The following rules define the number of output pages that will be printed or plotted:

- The device and its driver determine the dimensions of the printed page area.
- The number of pages is calculated from the physical dimensions of the schematic page and the driver-provided area dimensions.
- A positive offset shifts the entire schematic page to the right in the X direction, and down in the Y direction. Additional pages are output as required, therefore, the entire schematic is printed. No truncation takes place.
- A negative offset shifts the schematic page left, and up. The effect of a negative offset will be to start the drawing on a “previous” page. Previous pages are “pre-pended” so drawing can start at the starting portion of the schematic. No truncation takes place.

Only the number of pages required to print or plot the schematic page will be printed. Extra “blank” pages are omitted.

To print or plot one page

1. If you are working in the schematic page editor, open the window for the page you wish to print.
   OR
   If you are working in the project manager window, then select the schematic page.
2. From the *File* menu, choose the *Print* command. The Print dialog box opens.
3. Select the scale, the print quality, and the number of copies, then click *OK*.

Shortcut

Toolbar: 

Previewing Print Output

Using the Print Preview command, you can make sure that your schematic folder or schematic page is complete and that its appearance is what you want before you commit it to paper.

To preview print output

1. In the project manager window, select the documents you wish to print or plot.
   Or
   Select the entire design or library you wish to print or plot.
   Or
   Open the single part or schematic page you wish to print or plot.
2. From the *File* menu, choose *Print Preview*. The Print dialog box appears.
3. Edit the values as necessary.
4. Click *OK* to begin. The "Printing Now" message appears and after a moment, the Print Preview window opens. If the document requires multiple printer pages, scroll through them using the scroll bar.
5. Use the Previous page and Next page buttons to look at additional printer pages of the document.
6. To zoom in, move the magnifier pointer to a specific area and click.
7. When you finish, choose the *Close* button to dismiss the Print Preview window.

To preview print output from a text editor window

1. Open the text editor window you wish to print or plot.
2. From the *File* menu, choose the *Print Preview* command. The window displays the Print Preview and Print Setup dialog boxes.
3. Use the window's vertical scroll bars to view the other pages, if the document extends beyond one page.
4. When you finish, choose *Print Preview* from the *File* menu to return the text editor window to its normal editing state.

Printing or Plotting a Part

With the part editor active and open to a specific part, you can create a print or a plot of that part. You can also print a part from the project manager.

To print or plot a part

1. Select the part in the project manager window.
   Or
   Open the part to print.
2. From the *File* menu, choose the *Print* command.
   The Print dialog box displays.
3. Select the scale, the print quality, and the number of copies, then click *OK*.

To print or plot a multiple-part package

1. Select the part in the project manager window.
   OR
   Open the part, and from the *View* menu choose the Package command.
2. From the *File* menu, choose the *Print* command.
   The Print dialog box displays.
3. Select the scale, the print quality, and the number of copies, then click *OK*.

Shortcut

Toolbar: 

Scaling a Print or Plot

You can manually scale or have Capture automatically scale prints and plots to fit the paper size you choose.

To scale a print or a plot of a schematic page or part

1. From the *File* menu, choose *Print*. The Print dialog box appears.
2. Select one of the three radio buttons in the Scale box.
   - The *Auto scale* option scales each schematic page to fit a single sheet of paper.
   - The *Scale to page size* option scales your schematic pages to the page size you select in the Scale to size box. This will result in multiple sheets of paper if you select a sheet size larger than your printer paper.
   - The *Scale by factor* option scales your schematic pages to a factor of your choice. The acceptance range of factors is 0.100 to 10.000.
3. If you select the *Scale to sheet size* option above, the Scale to size list becomes available. Your schematic page is scaled to the sheet size you select. This will result in multiple sheets of paper if you select a sheet size larger than your printer paper.
4. Click *OK* to send the image to the output device.

Shortcut

Toolbar: 

Netlisting a Design

You netlist a design after you place parts, update part references, and check for design rule

violations.

You can choose from more than 30 industry-recognized netlist formats. Your choice of netlist format is determined by the application that you intend to use.
The EDIF 2 0 0, VHDL, PSpice and Verilog netlist formats generate true hierarchical netlists. When a design is netlisted with one of these formats, the instance property values on nets and parts are used. All other netlist formats in Capture produce flat netlists, and use occurrence property values. If you have translated a design with multiple schematic folders, use Annotate (and check for duplicate references) before netlisting.

- Run Design Rules Check to verify your design before you generate a netlist. This allows for more efficient netlist creation, and you can concentrate on netlist-specific problems if they should occur during the Create Netlist process. Design Rules Check warns you if certain conditions exist in your design. The severity of the specific problem may prevent completion of the design. Other conditions are subject to your judgment, and may be of no consequence. If you are satisfied with the results of design tests such as Design Rules Check, then proceed with the creation of a netlist.
- Design Rules Check uses the decision matrix located in the ERC Matrix tab located in the Design Rules Check dialog box. It also uses a set of pre-determined rules, which are part of the executable code.
- Use Design Rules Check as a guide to verify the integrity of your design. It is only a guide. It is possible to generate a valid netlist even if Design Rules Check reports errors.
- The value, if any, you create for the PCB footprint depends on the particular netlist format you want to produce. Different applications require netlists with different types of PCB footprints. If you do not specify this property, the PCB footprint will be set to the part value.

Net name resolution

When you are creating schematic pages, you can assign a variety of aliases to signals that are ultimately connected, but the netlist needs exactly one name for each net.

If Create Netlist encounters multiple names for a single net, higher priority aliases override lower-priority aliases. Priority is determined by the source of the name, ranked as follows:

Lowest:	System-generated names Aliases Power object names Off-page connectors Hierarchical port names
Highest:	Named nets

Any remaining conflicts among netnames are resolved according to the following rules:

- The net name closest to the "root" of the project takes precedence over those further away.

- If the net is a bus, the net alias assigned to the greatest number of bus members has the highest priority.
- Among net names of equal precedence, priority follows an alphabetical order.
  As you can see, a net may change names several times as Create Netlist works. For example, the net may start with an alias of Battery on one page, be renamed ToBattery from an off-page connector, change again to become DC as a port is encountered, and finally change to BatteryBackup when Create Netlist finds a named net closer to the root schematic folder. Once the netlist is created, you can select any piece of the net anywhere in the design and see the net's name as it is recorded in the netlist (BatteryBackup), not as it appears at that particular location.

PACK_SHORT Property

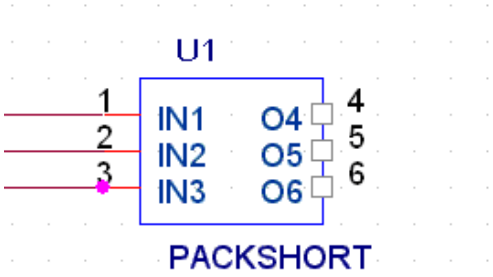
Capture includes a PACK_SHORT property that lets you map one logical pin to two or more physical pins. Take the example of an SMA connector. It has one signal pin and 4 shielding pins connected to Ground. The standard Capture symbol has only one visible Ground pin. So what is required is to short all the 4 GND pins in the Allegro netlist with the net that is connected to the visible GND pin.
To do this, you can either add the other three Ground pins as NC or with a different net name depending on whether they have been added as invisible or visible with zero stub length.
Alternatively, you can use the PACK_SHORT property in Capture to hide the PACK_SHORT pins or make them visible with zero length pin stub.
Using this property, multiple groups of pins, each group having two or more pins, can be shorted.

> You cannot use the PACK_SHORT property to short invisible pins. Only visible pins can be part of the PACK_SHORT property.

Syntax:
`PACK_SHORT=(<group1>)(<group2>)[<group3>]`
Where: <group> indicates (logicPin1, logicPin2 ... [logicPinN])

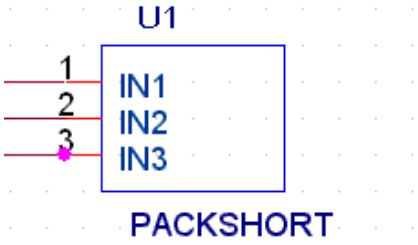
Example

Consider the assignment, PACK_SHORT = (A1, B1, Y1) (A2, B2) shorts together. The nets attached to logic pins A1, B1, and Y1 are shorted with each other and the nets attached to pins A2 and B2 are shorted with each other.
The PACK_SHORT property is implemented so you can either hide the pins to be shorted by checking the ignore checkbox in Package Properties dialog box or you can make them zero length pins and show them as unconnected as illustrated below.
Method 1:



The above part has O4, O5 and O6 added as PACK_SHORT pins (IN1,O4,O5,O6) that will inherit the pin IN1 net set as zero-length but their stub will still be visible. If you instantiate this part and generate an Allegro netlist, IN1, O4, O5 and O6 will inherit the same net.
If IN1 is left unconnected, all PACK_SHORT pins will be marked as NC.

Method 2:



If you have chosen to hide the pins to be shorted by checking the *Ignore* checkbox in Package properties, and also defined those pins as part of the PACK_SHORT property, the hidden pins will be written to the netlist. Also, its net will be inherited from the master pin that is visible on the symbol instance.

You can also use the PACK_SHORT property in conjunction with the PSpiceOnly property to specify shorting for nets attached to the instance. If a PACK_SHORT device is connected with two different nets across the PACK_SHORT pins, the net connected to the first pin defined in PACK_SHORT property will be written to the netlist.

Create Netlist dialog box tabs

OrCAD provides a number of netlist format files. You choose a netlist format in the Create Netlist dialog box.

- PCB Editor tab
- EDIF 2 0 0 tab
- INF tab
- Layout tab
- PSpice tab
- SPICE tab
- Verilog tab
- VHDL tab
- Other tab

> The Capture netlist format files are not the same as those shipped with SDT 386+. It is important that you keep both versions of the netlist format files installed if you plan on using both Capture and SDT 386+. Capture netlist formats files are supplied as .DLL files, while SDT netlist format files are provided as .EXE files.

In this section:

Creating a Netlist

To create a netlist

1. In the Project Manager, select the design file for which you want to create a netlist.
2. From the *Tools* menu, choose the Create Netlist command to display the Create Netlist dialog box.
3. Select a tab corresponding to the netlist format you want to use.
4. In the Netlist File field, enter a name for the output file. If the selected format creates an additional file (such as a map file or pinlist file), enter the filename in the appropriate field.
   If necessary, set the Part Value and PCB Footprint combined property strings to reflect the information you want in the netlist. For more information about combined property strings, see *Combined property strings*.
   If necessary, set other format-specific options in the Options group box.
5. Click *OK* to create the netlist.

Creating a Flat Netlist

To Create a Flat Netlist

1. In the project manager, select the design file (.DSN) to netlist.
2. From the *Tools* menu, choose *Create Netlist*.
3. Go to the *PSpice* tab.
4. Leave all the check boxes in the Options group blank.
5. In the Use Template list box, select the netlisting template(s) to apply.
   For details, see Specifying an alternate netlist template.
   Check whether you want to place DRC markers for Errors and Warnings.
6. In the Netlist File field, enter a name for the output file, or click the *Browse* button to assign a filename.
   If required, click the *View Output* check box to display the netlist after it is generated.
7. Click *OK*.

Working with Hierarchical Netlists

In this section:

- Using SUBPARAM

Creating a Hierarchical Netlist

To create a hierarchical netlist

1. In the project manager, select the design file (.DSN) you want to netlist.
2. From the *Tools* menu, choose *Create Netlist*.
3. Go to the *PSpice* tab.
4. In the Options group, click *Create Hierarchical Format Netlist*.
5. Click *Settings* to customize the format of the hierarchical netlist.
6. Click *Create Subcircuit Format Netlist* to specify how subcircuits will be netlisted.
7. In the Use Template list box, select the netlisting template(s) you wish to apply.
8. Check whether you want to place DRC markers for Errors and Warnings.
9. In the Netlist File text box, type a name for the output file, or click the *Browse* button to assign a filename.
   If required, click the *View Output* check box to display the netlist after it has been generated.
10. Click *OK*.

Customizing a Hierarchical Netlist

You can also customize the format of the subcircuit definition and reference text in the netlist. These settings, once defined, persist and will apply to all subsequent PSpice netlists whether the netlist is invoked from the Tools menu in the project manager or directly from within the schematic editor. Of course, you can always change the settings in the Settings dialog box.

Two groups of settings are saved: PSpice and "Layout versus Schematics" (or LVS). Having two groups makes it easy to switch between netlisting for PSpice and netlisting for an LVS compatible format. You can specify which group of settings is active for the netlister by using the Products list box.

> The settings you define are project specific. If you want to save the settings globally, click the Save as Default Project Settings button.

To customize the hierarchical netlist

1. In the PSpice tab of the Create Netlist dialog box, under the Options frame, click *Create Hierarchical Format Netlist*.
2. Click *Settings*, then enable or specify the following options, as desired:
3. Click *OK*.

Passing parameters to subcircuits

Hierarchical netlists have the advantage of allowing parameters to be passed from the top level schematic to any subcircuit schematics. To take advantage of this feature, you must use the SUBPARAM part in the SPECIAL.OLB library.
To learn more about setting up parameterized subcircuits for hierarchical netlists, click Using SUBPARAM.

Hierarchical netlists do not support cross-probing from a subcircuit, nor do they support Probe markers in a subcircuit. Cross-probing only works on the top-level (root) schematic.

Creating Subcircuit Netlists

You can specify how subcircuits in a hierarchical design are processed and defined in the simulation netlist.
You cannot directly simulate a subcircuit netlist; it defines a model that can be called by another circuit being simulated. The models of parts in the PSpice libraries such as op amps and regulators, which have multiple constituent components, are implemented as subcircuits.
A subcircuit implementation may consist of a single schematic, or a hierarchy of schematics.

To create a subcircuit format netlist

1. In the project manager, select the design file (.DSN) you want to netlist.
2. From the *Tools* menu, choose *Create Netlist* to display the Create Netlist dialog box.
3. Select the *PSpice* tab.
4. In the *Options* group, click *Create Subcircuit Format Netlist*, then click one of the following options, as required:
   - Descend: This generates a definition of a hierarchical design that includes the top level circuit as well as its subcircuits. (This option is only available if *Create Subcircuit Format Netlist* is enabled.) If the Create Hierarchical Format Netlist is not checked, then this option combination is equivalent to creating a flat netlist.
   - Do Not Descend: This generates a definition of a hierarchical design that includes only the top level circuit, without any of its subcircuits. (This option is only available if *Create Hierarchical Format Netlist* and *Create Subcircuit Format Netlist* are enabled.)

To define a subcircuit

1. Place hierarchical ports on the top level of the subcircuit for each node that interfaces to the circuit that will use it. (You should place all such ports on a single page of a multi-page schematic.)
2. Add a sequence property to each port, assigning values of 1, 2, and so forth. When you select the *Create Subcircuit Netlist Format* option on the PSpice tab of the *Create Netlist* dialog box, it generates a header line of the form:
   .SUBCKT LM317 IN ADJ OUT
   The example above comes from a schematic with ports named IN, ADJ, and OUT. The three ports were assigned Sequence property values of 1, 2 and 3 respectively.

To use this subcircuit in another schematic

1. Place a hierarchical symbol, or draw a hierarchical block.
2. Set the Implementation Type to PSpice Model and the Implementation to LM317. The symbol or block must be primitive and have a PSpice Template such as the following:

```
X^@REFDES %IN %ADJ %OUT @MODEL
```

The PSpice netlister, under guidance of the PSpiceTemplate, produces a line of the form:

```
X_U1 INNET ADJNET OUTNET LM317
```

INNET, ADJNET and OUTNET refer to the IN, ADJ, and OUT ports of the defining circuit. Because the correspondence is by position, the port order in the subcircuit definition must match the net order in the reference. The order in which netnames appear in the reference is controlled by the PSpiceTemplate property. As mentioned above, the Sequence property added to each port determines the order in which port names appear in the subcircuit definition.

For more information on the use of subcircuits, see the section on the .SUBCKT command in the online PSpice Reference Manual.

Using SUBPARAM

You can pass parameters from the top-level schematic to a subcircuit schematic using the SUBPARAM part. This allows you to explicitly define the properties and default values to be used during netlisting and simulation.
Any part in the subcircuit (child) schematic can reference the properties in its PSPICETEMPLATE. The PSpice subcircuit mechanism supports parameterizing:

- constants specified on device statements
- model parameters
- expressions consisting of constants
- parameters
- functions

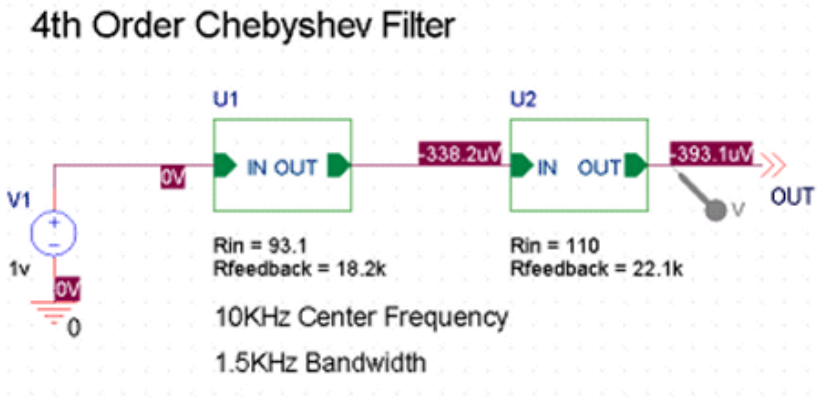
To set up parameter passing to a subcircuit using SUBPARAM

1. Make the subcircuit your active schematic page in the Capture schematic page editor.
2. From the *Place* menu, choose the *Part* command.
3. Select the part SUBPARAM from the PSpice library SPECIAL.OLB and place it on the subcircuit.
4. With the SUBPARAM part still selected, from the *Edit* menu, choose *Properties*.
   The property editor appears.
   In the spreadsheet, on an instance-by-instance basis, define the names and default values for the properties that can be changed.
5. To view property names and values at the same time, select a property cell, and click the *Display* button.
6. The Display Properties dialog box displays.
7. Check Name and Value format for Display Format, then click *OK*.
   In the top-level schematic, use the property editor to edit the properties of the hierarchical part or block that references the subcircuit (child) schematic so they match the properties you defined in Step 5.

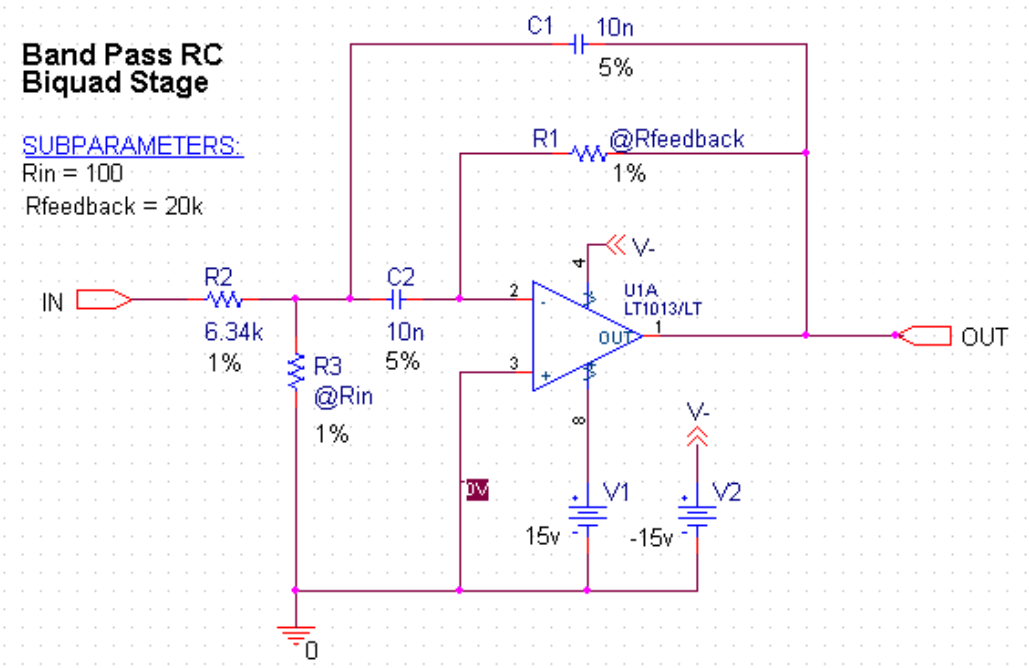
Example

This fourth-order Chebyshev filter schematic illustrates how the SUPARAM part may be used to pass design parameters from a top-level (parent) schematic to a subcircuit (child) schematic in a

hierarchical design.
In the top-level schematic, you explicitly define the parameter values you want to pass to a subcircuit. In this case, the parameters and their corresponding values for the U2 subcircuit are Rin = 110 and Rfeedback = 22.1k.



Descending into the hierarchy of the U2 subcircuit, you see the Rin and Rfeedback parameters listed with their corresponding default values of 100 and 20k, respectively, in the SUBPARAM part (under SUBPARAMETERS).



Notice the Rin and Rfeedback values are preceded with @ symbols, indicating that these parameters are substituted with values passed down from the top-level schematic.
Here is the hierarchical netlist generated in Capture for this design:

```
* source HISTO
V_V1 N00023 0 AC 1v
X_U1 N00023 N00030 OneStage PARAMS: RIN=93.1 RFEEDBACK=18.2k
X_U2 N00030 OUT OneStage PARAMS: RIN=110 RFEEDBACK=22.1k
.SUBCKT OneStage IN OUT PARAMS: RIN=100 RFEEDBACK=20k
```

```
V_V2 V- 0 -15v
R_R2 IN N00024 R_R2 6.34k
.model R_R2 RES R=1 DEV=1%
R_R1 N00034 OUT R_R1 {Rfeedback}
.model R_R1 RES R=1 DEV=1%
R_R3 0 N00024 R_R3 {Rin}
.model R_R3 RES R=1 DEV=1%
C_C1 N00024 OUT C_C1 10n
.model C_C1 CAP C=1 DEV=5%
C_C2 N00024 N00034 C_C2 10n
.model C_C2 CAP C=1 DEV=5%
X_U1A 0 N00034 N00594 V- OUT LT1013/LT
V_V1 N00594 0 15v
.ENDS
```

Here, the U2 and U1 subcircuits are referenced in the source (top-level) circuit as X_U1 and X_U2 with the explicit values defined for the parameters that are passed to from the top-level. A .SUBCKT models both the U1 and U2 subcircuits and the default parameters are listed as placeholders. When PSpice simulates the design, the parameters passed from the top-level circuit are the ones used in the subcircuits.

Specifying an Alternate Netlist Template

To specify an alternate netlist template

1. In the Capture project manager, select the design file (.DSN) you want to netlist.
2. From the *Tools* menu, choose *Create Netlist*.
3. Select the *PSpice* tab.
4. In the Use Template list box, add the name of the template to use.

14

Simulating Capture Designs Using PSpice

In this section:

Overview of Simulation Using PSpice

PSpice is a simulation program that models the behavior of a circuit. You can use PSpice to test and refine your design before manufacturing the physical board or IC. PSpice1 supports analyzes that can simulate analog-only, mixed-signal, and digital-only circuits.

To simulate your design in PSpice:

1. Create an Analog or Mixed A/D project:
   The Capture project must be of *Analog or Mixed A/D* type to simulate using PSpice. You can simulate other projects only if you convert them to Analog or Mixed A/D type. (See Creating Projects for Simulation)
2. Create a design using parts that can be simulated:
   While creating the designs for simulation, you need to use parts that PSpice can simulate. These parts are available in the PSpice part libraries. You can also associate PSpice models to Capture parts to be able to simulate the parts. A part needs to have an associated model (Implementation type = PSpice Model) and property defined to simulate the part in PSpice. (See Preparing a Design for Simulation)
3. Create a simulation profile:
   After completing the design circuit, you can create a simulation profile using the Simulation Settings dialog box. While creating a simulation profile, specify the types of analysis type and its various parameters, configuration file settings, and so on.
4. Run PSpice and view results:
   When you run PSpice from Capture, a netlist is generated and the PSpice probe window appears. The netlist contains the connectivity information and the analyzes type. To view

the output of a simulation, you can:

- place markers in Capture
- add traces in the PSpice probe window

(See Running a Simulation and Viewing Results Using PSpice)

Creating Projects for Simulation

To simulate a Capture design with PSpice, you must begin the project as an analog or mixed-signal type intended for simulation. Existing projects in Capture cannot be simulated without special modifications. To learn how to simulate an existing project with PSpice, see Setting up an existing project for simulation.

The remaining topics cover:

- Creating a new project
- Setting up an existing project for simulation

Creating a new project

To create a new project for simulation:

1. In Capture, choose *File – New - Project*.
   The New Project dialog box appears.
2. In the Name field, enter the name for the new project.
3. Select *Analog or Mixed A/D*.

   > You must create a project (not a design) and select the Analog or Mixed A/D option to be able to simulate the new design with PSpice.

4. In the Location text box, enter the path where you want the new project files to be stored, or use the Browse button to locate the directory.
5. Click *OK*.
   The Create PSpice Project dialog box appears.
6. Select one of the two available options: Create it based upon an existing project or Create a blank project.
   If you select Create it based upon an existing project, choose a project file (*.OPJ*) with either the appropriate drop-down menu or the Browse button.
   PSpice comes with a set of design templates covering basic electronics circuits and SMPS topologies. These design templates cover the range of analog, digital, and mixed designs. You can use the design templates, which are a combination of design and simulation profiles, as a starting point for new designs. These templates are also suitable for learning and demonstration purposes.

You can click Browse to locate and open any of the available templates. The templates are available in the *<Cadence_installation>\tools\pspice\capture_samples\design_examples* directory.

7. Click *OK* to create the new project directory and open the schematic page editor.

Setting up an existing project for simulation

If you have a design in Capture that was not set up initially as a project that can be simulated by PSpice, you need to do the following modifications in order to be able to simulate it. The basic process involves creating a new project intended for simulation and then copying the existing design into it.

> ⚠ Be sure to use a name different than that of existing project for the new project.

Once you have created the new PSpice project, use the Copy and Paste commands in Capture to copy the entire preexisting design into the new project. After copying, be sure to setup the simulation profile and prepare the parts for simulation as described in Preparing a Design for Simulation.

Preparing a Design for Simulation

A design that is targeted for simulation has:

- parts for which are simulation models available and configured
- sources of stimulus to the circuit

When creating designs for both simulation and printed circuit board layout, some of the parts you use are for simulation only (simulation stimulus parts like voltage sources), and some of the parts you use have simulation models that only model some of the pins of a real device.

The parts that are to be used for simulation, but not for board layout, have a SIMULATIONONLY property.

You can add this (or any) property to your own custom parts to make them simulation-only.

The remaining topics cover:

- Adding PSpice and parametrized libraries and parts
- Defining part properties needed for simulation
- Guidelines and best practices for specifying values for part properties
- Placing PSpice Ground 0 Symbols for PSpice Simulations
- Using the FLOAT property for unconnected pins
- Using Global Parameters and Expressions
- Associating PSpice model to a Capture symbol
- Importing PSpice Schematic Projects in Capture

- Defining Stimuli

Adding PSpice and parametrized libraries and parts

The PSpice part libraries (`.OLB` files) are located in the `tools\capture\library\pspice` directory, under your main installation directory. The simulation model libraries (`.lib` files) are located under the `tools\pspice\library` directory.

If you wish to add more PSpice part libraries to your design, you can do so by using the Place Part dialog box (choose *Place – Part* or use the Place Part toolbar button). Then add the library you need by selecting it from the `tools\capture\library\pspice` sub-directory. The PSpice libraries located in the `pspice` sub-directory include:

- vendor-supplied parts
- passive parts
- breakout parts
- behavioral parts
- special simulation-only parts

> When you select a part in the Place Part dialog box, the PSpice symbol ( ) appears for a part that can be simulated.

At minimum, a part that you can simulate has these properties:

- A simulation model to describe the part's electrical behavior; the model can be:
  - explicitly defined in a model library,
  - built into PSpice, or
  - built into the part (for some kinds of analog behavioral parts).
- A part with modeled pins to form electrical connections on your schematic.
- A translation from the design part to the netlist statement so that PSpice can read it in.

> Not all parts in the libraries are set up for simulation. For example, connectors are parts destined only for board layout and do not have these simulation properties. The libraries contained in the pspice subfolder are the only ones set up for simulation.

> You must use the 0 (zero) ground part in designs intended to be simulated by PSpice. If you have used other ground parts, you can rename them to 0 so that they will be accepted by PSpice. (See Placing PSpice Ground 0 Symbols for PSpice Simulations)

Special simulation-only parts

The PSpice part libraries also include special parts that you can use only for simulation. These include:

- stimulus parts to generate input signals to the circuit (see Defining Stimuli)
- ground parts required by all analog and mixed-signal circuits, which need reference to the ground
- simulation control parts to do things like set bias values
- output control parts for generating tables and line-printer plots to the PSpice output file

Vendor-supplied parts

The PSpice libraries provide an extensive selection of manufacturers' analog and digital parts. Typically, the library name reflects the kind of parts contained in the library and the vendor that provided the models. For example, MOTOR_RF.OLB and MOTOR_RF.LIB contain parts and models, respectively, for Motorola-made RF bipolar transistors.

Two types of libraries are provided with PSpice:

- Standard PSpice libraries
- PSpice Advanced Analysis libraries

Standard PSpice libraries

The standard PSpice libraries are installed in the following locations in the installation directory:

- Capture symbols for standard PSpice libraries at \tools\capture\library\pspice
- Standard PSpice model libraries at \tools\psice\library\

> To find out more about each model library, read the comments in the .LIB file header.

PSpice Advanced Analysis libraries

The PSpice Advanced Analysis libraries contain over 4,300 analog parts. The Advanced Analysis libraries contain parameterized and standard parts. The majority of the parts are parameterized. The parametrized parts have tolerance, distribution, optimizable and smoke parameters that are required by the PSpice Advanced Analysis tools. Standard parts in the Advanced Analysis libraries are similar to parts in the standard PSpice libraries.

The parametrized parts are associated with template-based PSpice models. An important advantage of using the template-based PSpice models is that you can pass simulation parameters as properties from the schematic editor. For example, if a template-based model is associated with a part, the simulation parameters that you specify on an instance of the part in

your design will be passed to the model. There is no need to edit the model itself to change a parameter value. This is unlike the standard PSpice parts that are associated with device characteristic curve-based PSpice models, where you need to edit the model to change a simulation parameter. For more information on template-based and device characteristic curve-based PSpice models, see Chapter 4, "Creating and editing models", in the online PSpice User's Guide.

Use parametrized parts from Advanced Analysis libraries if you want to analyze the part with an Advanced Analysis tool.

Advanced Analysis tool	Part Parameters
Sensitivity	Tolerance parameters
Optimizer	Optimizable parameters
Smoke	Smoke parameters
Monte Carlo	Tolerance parameters, Distribution parameters (default parameter value is Flat / Uniform)

> You can use a mixture of standard and parametrized parts in your design, but Advanced Analysis is performed on only the parameterized components.

The Advanced Analysis libraries are installed in the following locations in the installation directory:

- Capture symbols for Advanced Analysis libraries at

  \tools\capture\library\pspice\advanls

- PSpice Advanced Analysis model libraries at

  \tools\pspice\library

The parts in the Advanced Analysis libraries are listed in the online PSpice Advanced Analysis Library List. For information on finding parts using the online PSpice Advanced Analysis Library List, see Finding the part that you want. To find out more about each model library, read the comments in the .lib file header.

Part naming conventions

The part names in the PSpice libraries usually reflect the manufacturers' part names. If multiple vendors supply the same part, each part name includes a suffix that indicates the vendor that supplied the model.

Finding the part that you want

If you are having trouble finding a part, you can search the libraries for parts with similar names by using either:

- the search option in the Place Part dialog box in Capture and restricting the parts list to those names that match a specified wildcard text string, or
- the online PSpice Library List or the PSpice Advanced Analysis Library List and searching for the generic part name using capabilities of the Adobe Acrobat Reader.

To find parts using the parts browser

1. In Capture, choose *Place – Part*.
2. In the Part Name field, type a text string with wildcard characters that approximates the part name that you want to find. Use this syntax:
   <wildcard><part_name_fragment><wildcard>
   where <wildcard> is one of the following:

*	match zero or more characters
?	match exactly one character

   The parts browser displays only the matching part names.

   > This method finds any part contained in the current part libraries configuration, including parts for user-defined models.

To find parts using the online PSpice Library List:

If you want to find out more about a part supplied in the PSpice libraries, such as the name of the manufacturer or whether you can simulate it, then search the online library lists.

Separate library lists are provided for the standard PSpice libraries and Advanced Analysis libraries. The parts in the standard PSpice libraries are listed in the online PSpice Library List. The parts in the Advanced Analysis libraries are listed in the online PSpice Advanced Analysis Library List.

1. From the *Windows Start* menu, choose *Cadence* and the release and then the Cadence Help shortcut.
   The Cadence Help window appears.
2. Click the *PSpice* category to show the documents in the category.
3. Double-click PSpice Library List or PSpice Advanced Analysis Library List.
   This opens the library list. Click the PDF Viewer icon on the Cadence Help toolbar if required.
4. Follow the instructions in the second page of the library list to use the library list.

If you are unsure of the device type, you can scan all of the device type lists using the Acrobat search capability. The first time you do this, you need to set up the across-list index. To find out more, refer to the online Adobe Acrobat manuals.

Passive parts

The PSpice libraries supply several basic parts based on the passive device models built into PSpice. These are summarized in the following table.

To find out more about how to use these parts and define their properties, look up the corresponding PSpice device letter in the Analog Devices chapter in the online PSpice Reference Manual, and then see the Capture Parts sections.

Available Parts	Device Type	PSpice Device Letter...
C C_VAR	capacitor	C
L	inductor	L
R R_VAR	resistor	R
XFRM_LINEAR K_LINEAR	transformer	K and L
T	ideal transmission line	T
TLOSSY	Lossy transmission line	T
TnCOUPLED TnCOUPLEDX* KCOUPLEn*	coupled transmission line	T and K

*For these device types, the PSpice libraries supply several parts. Refer to the online PSpice Reference Manual for the available parts.

Breakout parts

The PSpice libraries supply passive and semiconductor parts with default model definitions that define a basic set of model parameters. This way, you can easily:

- assign device and lot tolerances to model parameters for Monte Carlo and sensitivity/worst-case analyses
- define temperature coefficients
- define device-specific operating temperatures

These are called breakout parts and are summarized in the following table.

> To find out more about how to use these parts and define their properties, look up the corresponding PSpice device letter in the Analog Devices chapter in the online PSpice Reference Manual, and then see the Capture Parts sections.

Breakout Part	Device Type	PSpice Device Letter
BBREAK	GaAsFET	B
CBREAK	capacitor	C
DBREAKx	diode	D
JBREAKx*	JFET	J
KBREAK	inductor coupling	K
LBREAK	inductor	L
MBREAKx*	MOSFET	M
QBREAKx*	bipolar transistor	Q
RBREAK	resistor	R
SBREAK	voltage-controlled switch	S
TBREAK	transmission line	T
WBREAK	current-controlled switch	W
XFRM_NONLINEAR	transformer	K and L
ZBREAKN	IGBT	Z

* For this device type, the PSpice libraries supply several breakout parts. Refer to the online PSpice Reference Manual for the available parts.

Behavioral parts

Behavioral parts allow you to define how a block of circuitry should work without having to define each discrete component.

Analog behavioral parts

These parts use analog behavioral modeling (ABM) to define each part's behavior as a mathematical expression or lookup table. The PSpice libraries provide ABM parts that operate as math functions, limiters, Chebyshev filters, integrators, differentiators, and others that you can

customize for specific expressions and lookup tables. You can also create your own ABM parts.

Digital behavioral parts

These parts use special behavioral primitives to define each part's functional and timing behavior. These primitives are:

LOGICEXP	to define logic expressions
PINDLY	to define pin-to-pin delays
CONSTRAINT	to define constraint checks

Many of the digital parts provided in the PSpice libraries are modeled using these primitives. You can also create your own digital behavioral parts using these primitives.

Defining part properties needed for simulation

If you want to use a part for simulation, then your part should have the PSPICETEMPLATE property defined for it:

You can also add other simulation-specific properties for digital parts: IO_LEVEL, MNTYMXDLY, and PSPICEDEFAULTNET.

Here are the things to check when editing part properties:

- Does the property specify the correct number of pins/ nodes?
- Are the pins/ nodes in the specified property in the proper order?
- Do the pin/ node names in the property match the pin names on the part?

For examples of how to use the property, see PSPICETEMPLATE examples.

Editing simulation properties

To edit a property needed for simulation

1. In the schematic page editor, select the part to edit.
2. From the Edit menu, choose Properties to display the Parts spreadsheet of the Property Editor.
3. Click on the cell of the column you want to change, or click the New button to add a property (and type the property name in the Name field).
4. If needed, type a value in the Value text box.
5. Click Apply to update the design, then close the spreadsheet.

PSPICETEMPLATE property

The PSPICETEMPLATE property defines the PSpice syntax for the part's netlist entry. When creating a netlist, Capture substitutes actual values from the circuit into the appropriate places in the PSPICETEMPLATE syntax, then saves the translated statement to the netlist file.

Any part that you want to simulate must have a defined PSPICETEMPLATE property. These rules apply:

- The pin names specified in the PSPICETEMPLATE property must match the pin names on the part.
- The number and order of the pins listed in the PSPICETEMPLATE property must match those for the associated .MODEL or .SUBCKT definition referenced for simulation.
- The first character in a PSPICETEMPLATE must be a PSpice device letter appropriate for the part (such as Q for a bipolar transistor).

PSPICETEMPLATE syntax

The PSPICETEMPLATE contains:

- regular characters that the schematic page editor interprets verbatim, and
- property names and control characters that the schematic page editor translates.

Regular characters in template

Regular characters include the following:

- alphanumerics
- any keyboard part except the special syntactical parts used with attributes (@ & ? ~ #)
- whitespace

An identifier is a collection of regular characters of the form:

alphabetic character [any other regular character]*.

Property names in templates

Property names are preceded by a special character as follows:

[@ | ? | ~ | # | &]<identifier>

The schematic page editor processes the property according to the special character as shown in the following table.

Syntax*	Replacement
@<id>	Value of <id>. Error if no <id> property or if no value assigned.

&<id>	Value of <id>, if <id> is defined.
?<id>s…s	Text between s…s separators, if <id> is defined.
?<id>s…ss…s	Text between the first s…s separators, if <id> is defined, else the second s…s clause.
~<id>s…s	Text between s…s separators, if <id> is undefined.
~<id>s…ss…s	Text between the first s…s separators, if <id> is undefined, else the second s…s clause.
#<id>s…s	Text between s…s separators, if <id> is defined, but delete rest of template if <id> is undefined.

*s is a separator character.

Separator characters include commas (,), periods (.), semi-colons (;), forward slashes (/), and vertical bars (|). You must always use the same character to specify an opening-closing pair of separators.

You can use different separator characters to nest conditional property clauses.

The caret (^) character in templates

The schematic page editor replaces the caret (^) character with the complete hierarchical path to the device being netlisted.

The new line (\n) character sequence in templates

The part editor replaces the new line (\n) character sequence with a new line. Using newline character sequence (\n), you can specify a multiline netlist entry from a one-line template.

The percentage (%) character and pin names in templates

Pin names are denoted as follows:

%<pin name>

where pin name is one or more regular characters.

The schematic page editor replaces the %<pin name> clause in the template with the name of the net connected to that pin.

The end of the pin name is marked with a separator. To avoid name conflicts in PSpice, the schematic page editor translates the following characters contained in pin names.

Pin name character	Replacement
<	l (L)
>	g
=	e
\XXX\	XXXbar

To include a literal percentage (%) character into the netlist output, type the percentage symbol twice (%%) in the template.

Recommended scheme for netlist templates

Templates for devices in the part library start with a PSpice device letter, followed by the hierarchical path, and then the reference designator (REFDES) property. We recommend that you adopt this scheme when defining your own netlist templates.

PSPICETEMPLATE examples

Simple resistor (R) template

The R part has two pins (1 and 2) and two required properties, REFDES and VALUE.

The template for the resistor is:

R^@REFDES %1 %2 @VALUE

A sample translation of the template is:

R_R23 abc def 1k

where REFDES equals R23, VALUE equals 1k, and R is connected to nets abc and def.

Voltage source with optional AC and DC specifications (VAC) template

The VAC part has two properties, AC and DC, and two pins, + and -.

The template is:

V^@REFDES %+ %- ?DC|DC=@DC| ?AC|AC=@AC|

A sample translation of the template is:

V_V6 vp vm DC=5v

where REFDES equals V6, VSRC is connected to nodes vp and vm, DC is set to 5v, and AC

is undefined.

Another sample translation of the template is:

V_V6 vp vm DC=5v AC=1v

where, in addition to the settings for the previous translation, AC is set to 1v.

Parameterized subcircuit call (X) template

This example supposes a subcircuit Z that has two pins (a and b) and a subcircuit parameter G, where G defaults to 1000 when no value is supplied.

To allow the parameter to be changed on the schematic page, treat G as property in the template.

The template is:

X^@REFDES %a %b Z PARAMS: ?G|G=@G|

~G|G=1000|

An equivalent template (using the if...else form) is:

X^@REFDES %a %b Z PARAMS: ?G|G=@G||G=1000|

A sample translation of the template is:

X_U33 101 102 Z PARAMS: G=1024

where REFDES equals U33, G is set to 1024, and the subcircuit connects to nets 101 and 102.

Another sample translation of the template is:

X_U33 101 102 Z PARAMS: G=1000

where the settings of the previous translation apply except that G is undefined.

Digital stimulus parts with variable width pins template

For a digital stimulus device template (such as that for a DIGSTIM part), a pin name can be preceded by an asterisk (*) character. This signifies that the pin can be connected to a bus and the width of the pin is set to be equal to the width of the bus.

The template is:

U^@REFDES STIM(%#PIN, 0) %*PIN

\n+ STIMULUS=@STIMULUS

where #PIN refers to a variable width pin.

A sample translation of the template is:

U_U1 STIM(4,0) 5PIN1 %PIN2 %PIN3 %PIN4

+ STIMULUS=mystim

where the stimulus is connected to a four-input bus, a[0-3].

Pin callout in subcircuit templates

The number and sequence of pins named in a template for a subcircuit must agree with the definition of the subcircuit itself--that is, the node names listed in the .SUBCKT statement, which heads the definition of a subcircuit. These are the pinouts of the subcircuit.

IO_LEVEL property

The IO_LEVEL property defines the level of interface subcircuit model PSpice must use for a digital part that is connected to an analog part.

If you are creating a digital part, you need to

1. Add the IO_LEVEL property to the part and assign a value shown in the table below.

Value	Interface subcircuit (level)
0	circuit-wide default
1	AtoD1 and DtoA1
2	AtoD2 and DtoA2
3	AtoD3 and DtoA3
4	AtoD4 and DtoA4

2. Use this property in the property definition (IO_LEVEL is also a subcircuit parameter used in calls for digital subcircuits).

MNTYMXDLY property

The MNTYMXDLY property defines the digital propagation delay level that PSpice must use for a digital part.

If you are creating a digital part, you need to do the following

1. Add the MNTYMXDLY property to the part and assign a value shown in the table below.

Value	Propagation delay

0	circuit-wide default
1	minimum
2	typical
3	maximum
4	worst-case (min/max)

2. Use this property in the property definition (MNTYMXDLY is also a subcircuit parameter used in calls for digital subcircuits).

PSPICEDEFAULTNET property

The PSPICEDEFAULTNET pin property defines the net name to which a power or ground (invisible) pin is connected.

For example, if the power and ground pins on a digital part are connected to the digital nets $G_DPWR and $G_DGND, respectively, then the properties are defined as follows:

PSPICEDEFAULTNET=$G_DPWR

PSPICEDEFAULTNET=$G_DGND

If you are creating a digital part, you need to do the following

1. For each power pin, create a PSPICEDEFAULTNET property and assign the name of the digital net to which the pin is connected.
2. Use the appropriate pin name in the property definition.

Guidelines and best practices for specifying values for part properties

Note the following when specifying values for part properties:

- Do not leave a space between the value and its unit, if the unit is a scale symbol. For example, specify 5K instead of 5 K.
- For a listing of the scale symbols, see the section on Numeric value conventions in the Before you begin chapter of the online PSpice Reference Guide.
- Do not use the European notation, where the decimal point is omitted and replaced by the unit symbol, for specifying values. For example, if you specify 3K3 (the European notation for 3.3K), PSpice reads the value as 3K. Use 3.3K instead.
- Specify tolerance values as percentages. If you specify an absolute value, the tolerance value will be read as an absolute number. For example, if you specify the value of the POSTOL property as a percentage, say 10%, on a 10K resistor, the distribution values will

be taken in the range of 10K?1K. If you specify the tolerance value as an absolute number, say 10, the distribution values will be taken in the range of 10K?10?

Placing PSpice Ground 0 Symbols for PSpice Simulations

For PSpice analog simulation to run, your design must have a PSpice ground (0) symbol. The CAPSYM.OLB, which is the default library in Capture, includes the PSpice ground (0) symbol. Use the 0 symbol to place a PSpice ground 0 symbol in your design.

To select the 0 symbol:

1. Choose Place – Ground (or use the Place Ground toolbar button). The Place Ground dialog box appears.
2. Select the CAPSYM part library from the Libraries list (if it is not already selected).

> You can also place the 0 symbol from the Source part library. To do this, add the Source part library to the Libraries list using the Add Library button; SOURCE.OLB is located in the \TOOLS\CAPTURE\LIBRARY\PSPICE subdirectory under your installation directory.

3. Select the 0 symbol (if not already selected).
4. Click OK to place the PSpice ground 0 symbol.

> Alternatively, you can place any ground symbol, open the Property Editor, and change its name to 0.

> While generating the PSpice netlist, if Capture does not find a PSpice ground (0) symbol in your design, then a warning message is flagged in the Session Log. You may ignore the warning, if the design will be used for running the digital PSpice simulation. However, for running analog simulation, the design must have at least one PSpice ground 0 symbol.

> If you are starting a new analog PSpice design, then it is recommended that you use the PSpice project template, AnalogGNDSymbol.opj. This project by default has the PSpice ground 0 symbol needed for your analog designs.

Using the FLOAT property for unconnected pins

When preparing a circuit for simulation with PSpice, it is important that all pins for all parts are connected properly. If a pin is meant to remain unconnected intentionally, you need to use the PSpice pin property FLOAT, rather than a No Connect symbol. Otherwise, the circuit may not netlist correctly for PSpice.

The pin property FLOAT may have one of the following three values:

Value	Description
Error	The pin will not netlist. An error message will be returned when the PSpice simulation netlist is generated. Use Error when you want to be reminded that this pin is a "no connect" and should be treated in a special way. Error is the default value.
RtoGND	The pin is connected to a virtual resistor, whose opposite pin is tied to GND. The resistor has a value of 1/GMIN. This value allows the simulation netlist to be created and allows PSpice to perform the analysis. The virtual resistor will not be processed as part of a layout netlist or appear in a BOM.
UniqueNet	The pin, when left unconnected, is attached to a unique node when the PSpice simulation netlist is generated. Use UniqueNet when you want the pin to remain unconnected but correspond to the Probe data associated with its part.

The FLOAT property can either be defined in the part editor when creating a new part, or you can edit a pin on an existing part using the property editor.

To define the FLOAT property using the property editor

1. In Capture, double-click on the pin to open the property editor spreadsheet.
2. Click on the Pins tab.
3. Click New Column and type FLOAT (upper case) in the Name text box.
4. Type the property value you want to use, then click OK.
5. Click Apply or close the spreadsheet to have the changes take effect.

Using Global Parameters and Expressions

In addition to literal values, you can use global parameters and expressions to represent numeric values in your circuit design.

Global parameters

A global parameter is like a programming variable that represents a numeric value by name.

Once you have defined a parameter (declared its name and given it a value), you can use it to represent circuit values anywhere in the schematic; this applies to any hierarchical level.

Some ways that you can use parameters are as follows:

- Apply the same value to multiple part instances.
- Set up an analysis that sweeps a variable through a range of values (for example, the DC sweep or parametric analysis).

When multiple parts are set to the same value, global parameters provide a convenient way to change all their values for "what-if" analyses.

For example, if two independent sources have a value defined by the parameter VSUPPLY, then you can change both sources to 10 volts by assigning the value once to VSUPPLY.

Declaring and using a global parameter

To use a global parameter in your design, you need to:

- define the parameter using a PARAM part from SPECIAL.OLB.
- use the parameter in place of a literal value somewhere in your design.

To declare a global parameter

1. Place a PARAM part in your design.
2. Double-click the PARAM part to display the Parts spreadsheet.
3. Do the following for each global parameter:
    1. Click New, then enter NAMEn in the Property Name field, then click OK. This creates a new property for the PARAM part, NAMEn in the spreadsheet.
    2. Click on the cell below the NAMEn column and enter a default value for the parameter.
    3. While this cell is still selected, click Display. In the Display format frame, select Name and Value, then click OK.

        > System variables have reserved parameter names. Do not use these parameter names when defining your own parameters.

    4. Click Apply to update all the changes to the PARAM part, and then close the spreadsheet.

For example, to declare the global parameter VSUPPLY that will set the value of an independent voltage source to 14 volts, place the PARAM part, and then create a new property named VSUPPLY with a value of 14V.

To use the global parameter in your circuit

1. Find the numeric value that you want to replace: a component, model parameter, or other property value.
2. Replace the value with the name of the global parameter using the following syntax:
   { global_parameter_name }
   The curly braces tell PSpice to evaluate the parameter and use its value.

To avoid errors, always include parameter variable name in curly braces when it is assigned to the parameter/property on the part.

Expressions

An expression is a mathematical relationship that you can use to define a numeric or Boolean (TRUE/FALSE) value.

PSpice evaluates the expression to a single value every time:

- it reads in a new circuit
- a parameter value used within an expression changes during an analysis.

An example of this would be a parameter that changes with each step of a DC sweep or parametric analysis.

Specifying expressions

To use an expression in your circuit

1. Find the numeric or boolean value you want to replace: a component value, model parameter value, other property value, or logic in an IF function test.
2. Replace the value with an expression using the following syntax:
   { expression }
   where {expression} can contain any of the following:
   - standard operators (listed in the table below)
   - built-in functions (listed in the PSpice User's Guide)
   - user-defined functions
   - system variables (listed in the PSpice User's Guide)
   - user-defined global parameters
   - literal operands

The curly braces tell PSpice to evaluate the expression and use its value.

Operator class	Operator	Operation
arithmetic		
	+	addition or string concatenation
	-	subtraction
	*	multiplication
	/	division

	**	exponentiation
logical		
	~	unary NOT
	|	boolean OR
	^	boolean XOR
	&	boolean AND
relational*		
	==	equality test
	!=	non-equality test
	>	greater than test
	>=	greater than or equal to test
	<	less than test
	<=	less than or equal to test

*Logical and relational operators are used within the IF() function; for digital parts, logical operators are used in Boolean expressions.

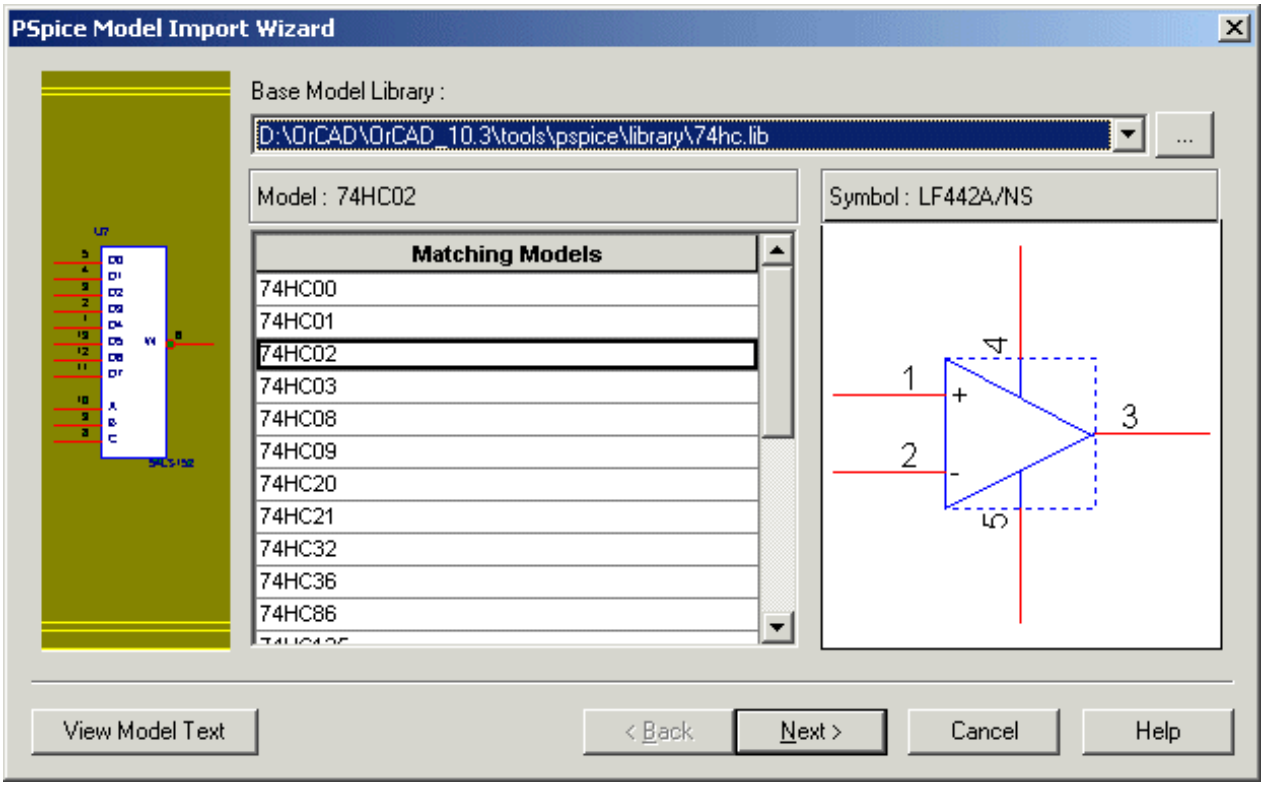
For lists of system variables and functions in arithmetic expression, refer to your PSpice User's Guide.

Associating PSpice model to a Capture symbol

You can associate a PSpice model to an existing Capture symbol using the Model Import wizard.

To associate a PSpice model to an existing Capture symbol:

1. Open the library (.OLB) containing the symbol for which the PSpice model is required.
2. Choose Tools and choose the Associate PSpice Model command.
   or
   From the library file pop-up menu, choose Associate PSpice Model.
   The Model Import Wizard appears.



If the symbol is open in Capture, the Associate PSpice Model command will appear disabled in the pop-up menu.

If the selected symbol in the .OLB file has a Convert view, it will be ignored by the Model Import wizard. The PSpice model gets attached to the symbol in the normal view.

If you are reusing any existing symbol, you might get an warning stating that the Implementation property is already defined. Ignore the warning and click OK.

3. Specify the name and location of the library containing the required PSpice model.
   You can either select the library from the Base Model Library drop-down list box or can browse to the library location.
   Once you have selected the .lib file, the Model Import wizard lists the model, then matches the symbol, in the Matching Model list box.
4. In the Select Matching page of the wizard, choose a model from the Matching Models list box and click Next.
5. In the Define Pin Mapping page of the wizard, map each of the model terminal to a Symbol pin. The pin names on the selected symbol appear in the Symbol Pin drop-down list box.

For mapping you may want to view the model definition. For this use the View Model Text button at the bottom-left corner of the wizard.

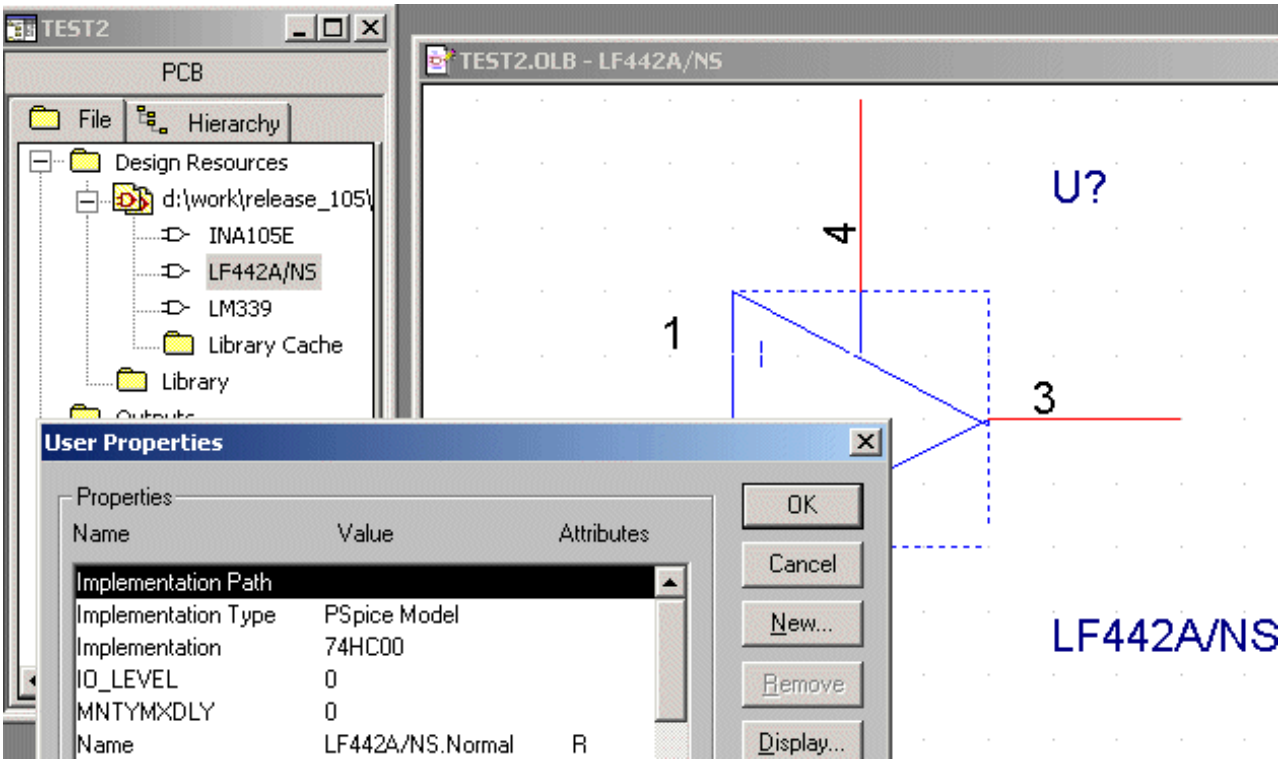
6. To complete attaching a PSpice model to the selected symbol and to close the Model Import wizard, click Finish.
   A message box appears indicating that the a PSpice model is now attached to the selected symbol.

The procedure covered in this section can be used to attach PSpice models only to homogeneous part symbols.

When you use Model Import Wizard to attach a model to a symbol, the following symbol properties are updated.

- Value of the IMPLEMENTATION TYPE property
- Value of the IMPLEMENTATION property

The value of the IMPLEMENTATION property attached to the part symbol, should match the name of the PSpice model as it appears in the .MODEL or .SUBCKT definition.



- Value of the property (not required for template-based models).

Importing PSpice Schematic Projects in Capture

You can import a MicroSim Schematics project to an OrCAD Capture project.

To convert a Schematics project to Capture:

1. In Capture, choose File - Import Design.
2. In the PSpice tab, select the source schematic (Open), the destination project, (Save As) and the PSPICE.INI (Schematic Configuration File) file for existing settings.
3. Click OK to start the automated translation utility.

> For more detailed information about using the Schematics-to-Capture Translator, see the online guide Converting MicroSim Schematics Designs to OrCAD Capture Designs (SCH2CAP.PDF) located in the \doc directory of your OrCAD installation, or if you installed the documentation to your hard disk, you can access it by choosing Cadence Help from the Cadence - Release <version> options of the Start menu.

Defining Stimuli

To simulate your circuit, you need to connect one or more source parts that describe the input signal that the circuit must respond to .

The PSpice libraries supply several source parts that are described in the tables that follow. These parts depend on:

- the kind of analysis you are running,
- whether you are connecting to the analog or digital portion of your circuit, and
- how you want to define the stimulus: using the Stimulus Editor, using a file specification, or by defining part property values.

Analog stimuli

Analog stimuli include both voltage and current sources. The following table shows the part names for voltage sources.

Input	Part for voltage
DC bias	VDC or VSRC
AC magnitude and phase	VAC or VSRC
exponential	VEXP or VSTIM
periodic pulse	VPULSE or VSTIM
piecewise-linear	VPWL or VSTIM
piecewise-linear that repeats forever	VPWL_RE_FOREVER or VPWL_F_RE_FOREVER
piecewise-linear that repeats n times	VPWL_N_TIMES or VPWL_F_N_TIMES*
frequency-modulated sine wave	VSFFM or VSTIM
sine wave	VSIN or VSTIM

*VPWL_F_RE_FOREVER and VPWL_F_N_TIMES are file-based parts; the stimulus specification is saved in a file and adheres to the PSpice netlist syntax.

To determine the part name for an equivalent current source, in the table of voltage source parts, replace the first V in the part name with I. For example, the current source equivalent to VDC is IDC, to VAC is IAC, to VEXP is IEXP, and so on.

Using VSTIM and ISTIM

You can use VSTIM and ISTIM parts to define any kind of time-based input signals. To specify the input signal itself, you need to use the Stimulus Editor.

If you want to specify multiple stimulus types

If you want to run more than one analysis type, including a transient analysis, then you need to use either of the following:

- time-based stimulus parts with AC and DC properties
- VSRC or ISRC parts

Using time-based stimulus parts with AC and DC properties

The time-based stimulus parts that you can use to define a transient, DC, and/or AC input signal are listed below.

VEXP	IEXP
VPULSE	IPULSE
VPWL	IPWL
VPWL_F_RE_FOREVER	IPWL_F_RE_FOREVER
VPWL_F_N_TIMES	IPWL_F_N_TIMES
VPWL_RE_FOREVER	IPWL_RE_FOREVER
VPWL_RE_N_TIMES	IPWL_RE_N_TIMES
VSFFM	ISFFM
VSIN	ISIN

In addition to the transient properties, each of these parts also has a DC and AC property. When you use one of these parts, you must define all of the transient properties. However, it is common to leave DC and/or AC undefined (blank). When you give them a value, the syntax you need to use is as follows.

Property	Syntax
DC	DC_value[units]
AC	magnitude_value[units] [phase_value]

> For the meaning of transient source properties, refer to the I/V (independent current and voltage source) device type syntax in the Analog Devices chapter in the online PSpice Reference Guide.

Using VSRC or ISRC parts

The VSRC and ISRC parts have one property for each analysis type: DC, AC, and TRAN. You can set any or all of them using PSpice netlist syntax. When you give them a value, the syntax you need to use is as follows.

Property	Syntax
DC	DC_value[units]

AC	magnitude_value[units] [phase_value]
TRAN	time-based_type (parameters)

where time-based_type is EXP, PULSE, PWL, SFFM, or SIN, and the parameters depend on the time-based_type.

> If you are running only a transient analysis, it is recommended that you use a VSTIM or ISTIM part if you have the standard package, or one of the other time-based source part that has properties specific for a waveform shape.

Digital stimuli

Input	Part
For transient analyses	
DIGSTIMn	signal or bus (any width)
DIGCLOCK	clock signal
STIM1	1-bit signal
STIM4	4-bit bus
STIM8	8-bit bus
STIM16	16-bit bus
FILESTIMn	file-based signal or bus (any width)

Using Partial Design Simulation

Using the Partial Design Simulation feature, you can:

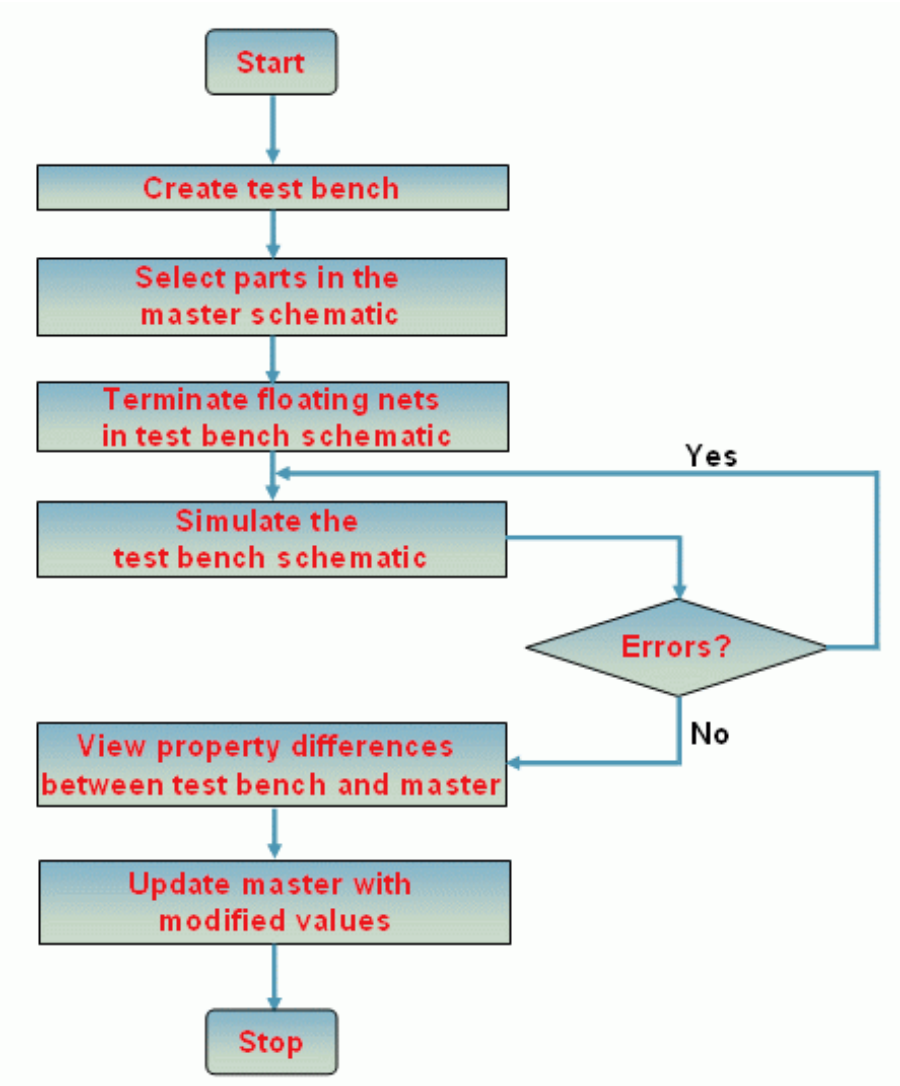
- Identify individual components of any design and simulate only selected portions
- Simulate different circuits in the design with different simulation profiles
- Create netlist of only a particular portion of the design
- Compare and merge portions of a design quickly

  To use this feature, you select a portion called test bench of a master design. You create one or more test benches using the *Tools – Test Bench – Create Test Bench* menu of OrCAD Capture. The test benches are listed in the Project Manager window of the master design. You can add components from the design to a test bench by selecting them from the master schematic canvas, and then add profiles and simulate the test bench. You can also synchronize the master design with a test bench, to propagate any changes you make in the test bench design.

The OrCAD Capture CIS license is required to use this feature.

You can use the Partial Design Simulation in the following flow, as shown in the figure:

1. Create a test bench.
2. Select parts in the master schematic.
3. Terminate floating nets in test bench schematic.
4. Simulate the test bench schematic.
5. View Property differences between test bench and master.
6. Update the master with modified values.



The remaining sections explain these steps in detail.:

- Working with a Test Bench
- Comparing and Updating Master Design

Working with a Test Bench

A test bench is like any other new project created in Capture. When you create a test bench, it is listed under the *TestBenches* node in the project manager of the master project. All simulation profiles and parameters or variables in the master project are copied by default to the test bench project. The components in the different schematics are grayed out. You can choose to activate the components to create a partial design.
You might need to add terminations and other parts to the partial design of the test bench because a test bench design must be complete in itself. You can also make edits to your test bench to prepare it for simulation by adding a stimulus or simulation profiles. You can simulate a test bench even if the master design is not a PSpice project. If the master project is a PSpice project, the test bench can inherit the simulation profiles in the master project.

Creating a Test Bench

1. Select the DSN file in the Project Manager
2. Choose *Tools – Test Bench – Create Test Bench*
   The Test Bench field appears.
3. Enter a name in the *Enter Test Bench Name* field.

You can set a default test bench name by adding the *Default Test Bench Name* property in the `[TEST BENCH]` section of `capture.ini`. For example, to set the default test bench name to `MyTestBench`, add the following section in `capture.ini` :

```
[TEST BENCH]

Default Test Bench Name=MyTestBench
```

1. Click *OK*.
   The test bench is added under *TestBenches* in the Project Manager. The created test bench contains all the designs in the master project.
   The components in the schematic pages of the test bench are grayed out. You need to add components to the test bench to be able to work on a partial design.

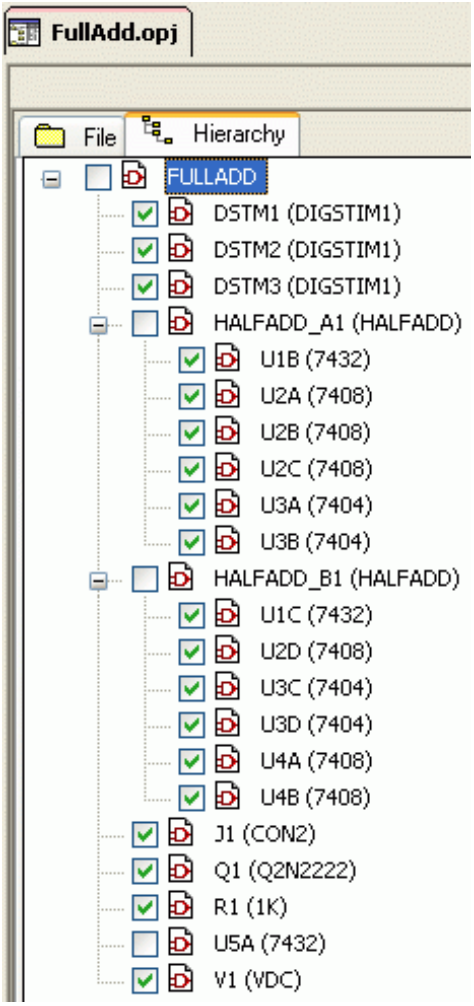
Note: You can activate a test bench by right-clicking on the test bench in the Project Manager under Test Benches and choosing Make Active.

Activating Components

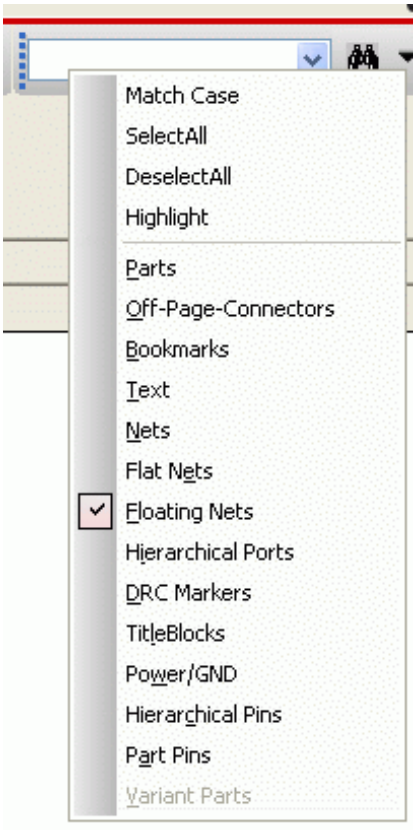
You can activate components in a test bench by using any one of the options; context-menu for selected parts in the master design, conext-menu for selected parts in the test bench design, or from the hierarchy editor.
To activate components from the master design:

1. Select the components in the master design.

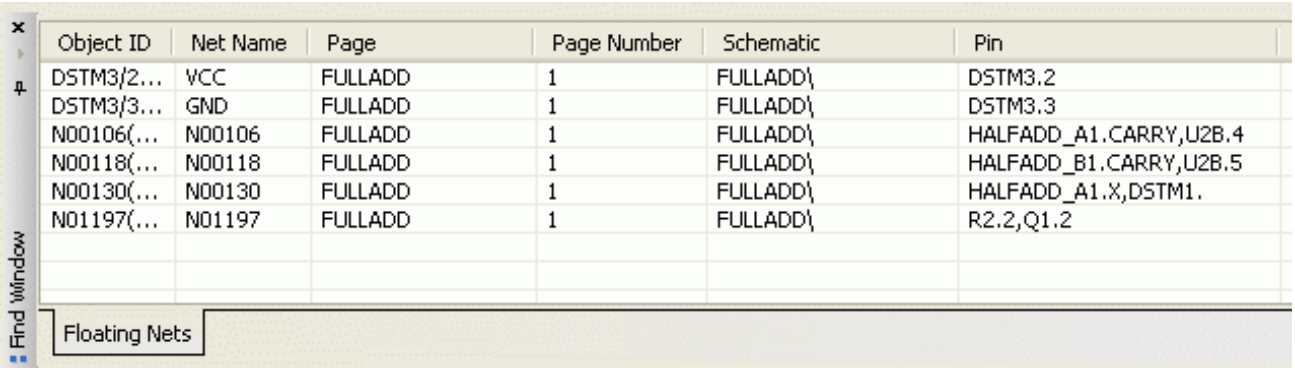
2. Right-click and choose *Test B ench – Add Part(s) to Active TestBench*.
   To activate components from the test bench design:
   Select the components in the test bench design.
   Right-click and choose *TestBench – Add Part(s) to Self* .
   To activate components using the hierarchy editor, check the components to be added in the hierarchy editor of the master design, as shown in the figure.



   Similarly, you can inactivate a component from the test bench design from the context-menu or the hierarchy editor. Capture will ignore all components that are inactive. As a result, these inactive components will not be processed, for example, for simulation.
   When you activate only a portion of the design, many nets might be floating because they are not terminated. You can easily resolve this problem by making a floating net search. To perform this search:
3. Select the test bench design in Capture
4. From the search menu, select *Floating Nets* as shown in the figure



5. Click the Find button.
   All the floating nets requiring terminators are listed in the Floating Nets tab of the Find Window.



Object ID	Net Name	Page	Page Number	Schematic	Pin
DSTM3/2...	VCC	FULLADD	1	FULLADD\	DSTM3.2
DSTM3/3...	GND	FULLADD	1	FULLADD\	DSTM3.3
N00106(...	N00106	FULLADD	1	FULLADD\	HALFADD_A1.CARRY,U2B.4
N00118(...	N00118	FULLADD	1	FULLADD\	HALFADD_B1.CARRY,U2B.5
N00130(...	N00130	FULLADD	1	FULLADD\	HALFADD_A1.X,DSTM1.
N01197(...	N01197	FULLADD	1	FULLADD\	R2.2,Q1.2

6. Double-click a row in the Find Window to select it in the design.

Comparing and Updating Master Design

You can compare the schematics in the master project and the test bench to highlight the differences using the SVS utility. This utility displays the differences and uses color code to highlight the different types of changes. The result window has two panels, the left panel represents the test bench. The differences listed are for the categories: unmatched object (yellow by default), missing objects (red by default), and matching objects (white in color). You can check any of the listed differences on the test bench panel, and propagate the changes to the master

design. However, you cannot update a master design for missing objects.

You can click Settings () to open the Settings dialog box and change the default colors in the *Color* tab. You can also filter for different objects, if you do not want them to be listed.
To compare and propagate changes:

1. Select the master DSN in Project Manager.
2. Choose *Tools – Test Bench – Diff and Merge*.
   The differences between the master design and the test bench design are displayed.

   To update the master design with the test bench differences, check the differences you want to update in the *SVS* tab and click Accept Left ().

Running a Simulation and Viewing Results Using PSpice

The following section discusses how to use PSpice from Capture to create a simulation profile, run a simulation, and view the simulation results in the PSpice Probe window.

The remaining topics cover:

- Creating a new simulation profile
- Creating a simulation netlist
- Viewing a simulation netlist
- Running a simulation
- Viewing the results as the simulation progresses
- Viewing the most recent simulation results
- Viewing the output file
- Editing simulation settings
- Placing markers
- Showing, hiding, and deleting markers
- Simulating and viewing the results of multiple profiles
- Making a simulation profile active

Creating a new simulation profile

A simulation profile (*.SIM) saves your simulation settings for an analysis type so you can reuse them easily.

You can create a new simulation profile from scratch or import the settings from an existing simulation profile. Importing settings from existing simulation profiles allows you to reuse the settings from other simulation profiles.

Capture allows you to create a new simulation profile by importing settings from a simulation profile that exists in the same project or in another project.

To create a new simulation profile

1. Choose *PSpice - New Simulation Profile*.
   The New Simulation dialog box appears.
2. In the Profile Name text box, type a name for the profile (such as the name of the analysis type for the new profile).
3. You may want to import the simulation settings from an existing profile to the new profile. To do this, from the Inherit From drop-down list, select the profile from which you want to import the settings.
   The Inherit From drop-down list lists all the profiles existing in the current project. Click browse to select a profile from another project.
4. Click *Create* to create the profile and display the *Simulation Settings* dialog box.
   The Simulation Settings dialog box appears.
5. In the *General* tab, specify the analysis type.
   Fill up the relevant settings in the General and other tabs.

   > Check whether you have the nom.lib added as a GLOBAL library under the Library category of the Configuration Files tab. This "master library" file calls out the other libraries that Cadence supplies along with the installation. It takes time for PSpice to scan each library file. PSpice creates an index file, called <filename>.IND, to speed up the search process. The index file is re-created whenever PSpice senses that it might be invalid.
   >
   > If this nom.lib is not there then Capture-PSpice interface will not be able to detect the Cadence-supplied PSpice libraries to be used in the simulation, so add this globally. The nom.lib resides in the <install dir>/tools/pspice/library folder.

6. Click *OK* to save the settings and close the dialog box.

After creating a new profile, you can edit the settings with the *PSpice – Edit Simulation Settings* command.

Shortcut

Keyboard: ALT+S+N

Creating a simulation netlist

When generating a PSpice netlist, you can choose between two types of netlist formats:

- Flat netlist
- Hierarchical netlist

Use the PSpice tab on the Create Netlist dialog box to generate a customized PSpice netlist.

> While generating the netlist, if Capture does not find a PSpice ground (0) symbol in your design, then a warning message is flagged in the Session Log. You may ignore the warning, if the design will be used for running digital PSpice simulation. However, for running analog simulation, the design must have at least one PSpice ground 0 symbol.

Viewing a simulation netlist

You can view the most recent simulation netlist for a selected design, or the current design.

To view a simulation netlist:

1. In the project manager, select the design for which you want to create a netlist, or open a schematic page.
2. From the *PSpice* menu, choose *View Netlist*.

Running a simulation

You can simulate your Capture design using PSpice, provided that there are PSpice models for the parts in your design. PSpice and Capture are fully integrated.

To run a simulation:

1. In the project manager, select a design to simulate, or open a schematic page.
2. In the project manager, select a simulation profile.
3. From the PSpice menu, choose Run or press the F11 function key.

PSpice does the following:

- Checks design rules for your design.
- Creates a simulation netlist for PSpice.
- Opens PSpice using the netlist created from your design.

PSpice creates an output file (.OUT) as the simulation progresses. It contains bias point information, model parameter values, error messages, and so on. If the simulation fails, you can view the output file to see the error messages.

If the simulation completes successfully, PSpice produces a data file (.DAT). This is the file PSpice uses to display the simulation results. To see marker simulation results, the schematic must be open.

Viewing the results as the simulation progresses

You can choose to view results as a simulation progresses or after a simulation is completed.

To view results as a simulation progresses:

1. From the *PSpice* menu, choose *Edit Simulation Settings*.
   The Simulation Settings dialog box appears.
2. In the *Probe Window* tab, select the *Display Probe* window check box.
3. Select the during simulation option.
4. Click *OK*.

Viewing the most recent simulation results

You can view the most recent simulation results for a schematic. If the schematic was simulated with more than one profile, you can choose which profile results to view.

To view the most recent simulation results:

1. Open the schematic for which you want to view simulation results. You must do this to see marker results.
2. In the project manager, select the simulation profile you want to be active.
3. From the PSpice menu, choose View Simulation Results, or press the F12 function key.

Viewing the output file

To view the most recent output file:

1. In the project manager, choose the simulation profile for which you want to see the output file.
2. From the *PSpice* menu, choose *View Output File*.

Editing simulation settings

Simulation profiles can be edited in Capture and PSpice.

To edit simulation settings from Capture

1. Choose *PSpice – Edit Simulation Settings*. The Simulation Settings dialog box appears.
2. Click the tab for the settings you want to change.

3. Edit the settings and click *Apply*.
4. Repeat steps 2 and 3 until you have changed all the settings you need.
5. Click *OK*.

Shortcut

Keyboard: ALT+S+E

Placing markers

To view the markers in the simulation results, the schematic must be open.

> Marker types on the Advanced command's submenu are only available after defining a simulation profile for an AC Sweep/Noise analysis.

To place markers in your design:

1. From the *PSpice* menu, choose *Markers*.
2. Select the marker you want to place.
3. Drag the marker symbol attached to the cursor to the location where you want to place it.
4. Click to place the symbol.
5. Repeat steps 3 and 4 until you have that you want.
6. Press the Esc key to end the marker mode, or right-click and select the *End Mode*.

Showing, hiding, and deleting markers

You can show all, hide all, or delete all markers. Showing or hiding markers in the schematic also shows or hides the trace results in the Probe window.

To show all, hide all, or delete all markers:

1. From the *PSpice* menu, choose *Markers*.
2. Select the *Show All*, *Hide All*, or *Delete All* option.

> When you move a wire that has a voltage marker placed on it, you might find that the voltage marker stays at its original place and no longer points to the wire. When you try to move the marker on the moved wire, you get the following warning message: "Voltage/digital level marker will be ignored unless connected to a wire, bus, or a pin." To avoid this warning, after moving the wire that already has a marker placed, choose the Show All command from the Markers submenu of the PSpice menu. This action rearranges all the markers to their corresponding node/net at the new locations.

Simulating and viewing the results of multiple profiles

You can select one profile or multiple profiles to be simulated or viewed. If you select only a single profile for simulation, it is handled as though you chose the Run command. If you select a file for viewing, it is handled as though you chose the View Simulation Results command.

> If you select multiple profiles, simulations for all selected profiles are performed using the simulation queue. You must then open the .DAT files to view the results.

To simulate multiple profiles

1. In the project manager, choose the simulation profiles you want to simulate.
2. From the *PSpice* menu, choose *Simulate Selected Profile(s)*.

PSpice opens and processes the profiles using the simulation queue.

To view the results:

1. Close the simulation queue, but leave PSpice active.
2. The Probe window is active, but no traces are visible.
3. From the PSpice *File* menu, choose *Open*.
4. Select the .DAT files that you want to view and click the Open button. A tab for each of the .DAT file you selected appears at the bottom of the Probe window.
5. From the PSpice *Window* menu, choose *Display Control*.
6. Select a profile for which you want to display the results.
7. Click *Restore*.
8. Repeat steps 4, 5, and 6 for all the profiles you want to view.

You can then click each tab to view the displayed results.

Making a simulation profile active

To simulate a design with a specific simulation profile, or to view the most recent results of a specific simulated profile, you must activate the profile.

To activate a simulation profile:

1. In the project manager, choose the simulation profile you want to activate.
2. From the *PSpice* menu, choose *Make Active*.

Files Needed for Simulation

To simulate your design, PSpice needs to know about:

- parts in your circuit and how they are connected,
- what analysis to run,
- the simulation models that correspond to the parts in your circuit, and
- the stimulus definitions with which to test.

This information is provided in various data files. Some of these are generated by the design entry program such as Capture or Design Entry HDL, others come from libraries (which can also be generated by other programs like the Stimulus Editor and the Model Editor), and still others are user-defined.

The remaining topics cover:

- Files that Design Entry Programs generate
- Other files that you can configure for simulation
- Files that PSpice generates

Files that Design Entry Programs generate

When you begin the simulation process, the design entry programs first generate files describing the parts and connections in your circuit. These files are the netlist file and the circuit files that PSpice reads before doing anything else.

Netlist file

The netlist file contains a list of device names, values, and how they are connected with other devices. The name that design entry program generate for this file is DESIGN_NAME-DESIGN_NAME.NET. The netlist file is located in the directory:

<project_directory>\worklib\<design_name>\cfg_analog\

Other files that you can configure for simulation

Before starting simulation, PSpice needs to read other files that contain simulation information for your circuit. These are model files, and if required, stimulus files and include files.

The simulation profile contains references to the other user-configurable files that PSpice needs to read.

You can create these files using PSpice programs like the Stimulus Editor and the Model Editor. These programs automate file generation and provide graphical ways to verify the data. You can also use the Model Text view in the Model Editor (or another text editor like Notepad) to enter the data manually.

Model library

PSpice uses this information in a model library to determine how a part will respond to different electrical inputs. These definitions take the form of either a:

- model parameter set, which defines the behavior of a part by fine-tuning the underlying model built into PSpice
- or a subcrcuit netlist, which describes the structure and function of the part by interconnecting other parts and primitives.

The most commonly used models are available in the PSpice model libraries shipped with your programs. The model library names have a.LIB extension.

If needed, however, you can create your own models and libraries, either by:

- manually using the Model Text view in the Model Editor (or another text editor like Notepad),

or

- automatically using the Model Editor.

Stimulus file

You can create a stimulus file either by:

- manually using the text editor in PSpice (or a standard text editor) to create the definition (a typical file extension is .STM)

- or -

- automatically using the Stimulus Editor (which generates an .STL file extension).

Include file

An include file is a user-defined file that contains:

- PSpice commands,
- supplemental text comments that you want to appear in the PSpice output file.

You can create an include file using any standard text editor. Typically, include file names have a .INC extension.

> An include file can contain definitions, using the PSpice .FUNC command, for functions that you want to use in numeric expressions elsewhere in your design.

Configuring model library, stimulus, and include files

PSpice searches model libraries, stimulus files, and include files for any information it needs to

complete the definition of a part or to run a simulation.

The files that PSpice searches depend on how you configure your model libraries and other files. Much of the configuration is set up for you automatically, however, you can do the following yourself:

- Add and delete files from the configuration.
- Change the scope of a file: that is, whether the file applies only to a profile, a design (local) or to any design (global).
- Change the search order.

To configure these, edit the simulation profile by using the Configuration Files tab in the Simulation Settings dialog box.

Files that PSpice generates

After reading the circuit file, netlist file, model libraries, and any other required inputs, PSpice starts the simulation. As simulation progresses, PSpice saves results to two files--the data file and the PSpice output file.

Probe data file

The data file contains simulation results that can be displayed graphically. PSpice reads this file automatically and displays waveforms reflecting the circuit response at nets, pins, and parts that you marked in your design (cross-probing). You can set up your simulation so that PSpice displays the results as the simulation progresses or after the simulation completes.

After PSpice has read the data file and displayed the initial set of results, you can add more waveforms and perform post-simulation analysis of the data.

There are two ways to add waveforms to the display:

- From within PSpice, by specifying trace expressions.
- From within the design entry program, by cross-probing.

PSpice output file

The PSpice output file is an ASCII text file that contains:

- the netlist representation of the circuit
- the PSpice command syntax for simulation commands and options (like the enabled analyses)
- simulation results
- warning and error messages for problems encountered during read-in or simulation

Its content is determined by:

- the types of analyses you run
- the options you select for running PSpice
- the simulation control parts (like VPRINT1 and VPLOT1) that you place and connect to nets in your design

Running PSpice in Batch Mode

Simulations in PSpice can be run in batch mode in two ways:

- Interactive Mode
- Non-Interactive Mode

Interactive Mode

In this mode PSpice is invoked from command line and the simulation circuit file is loaded. See the PSpice Reference Guide for information on the command line options.

To run a set of simulation circuit files (*.cir), the following command can be used on Windows command prompt:

<path to pspice.exe/pspice.exe /r <path to cir file>

For example:

<CDS_INST_DIR>\tools\pspice\pspice.exe /r C:/mysim/test1.cir

To set up multiple simulations, a batch file can be created with each command calling a different .cir file.

Non-Interactive Mode

In this mode the simulation runs in the background and pspice is not invoked.To run this, the psp_cmd.exe executable is called from command line.This executable resides in the same location as pspice.exe. For example:

<CDS_INST_DIR>\tools\pspice\psp_cmd.exe /r <path to cir file>

The background simulation has the advantage that in case of non-convergence, the simulation waits for user intervention for 10s and if there is no action it proceeds to the next available command .

SPICE netlist format

Generic, flat SPICE format netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not

checked for length.

- Node numbers are limited to five characters.
- If the Use net names option is selected, legal characters for node names are limited to:

```
0..9 A..Z a..z $ _ (underscore)
```

If you select this option, Capture uses the node names you placed on the schematic page (via aliases and hierarchical ports), where available. Not all versions of SPICE support alphanumeric node names. Check your SPICE manual for details. If your version of SPICE does not allow alphanumeric node names, you can still give them numeric names such as "17." These numeric names do not interfere with the ones generated by Capture, since the node numbers it generates begin at 10000 (except GND, which is always 0).

- All ASCII characters are legal except as noted for node names.

  For more information on flat SPICE netlists, see the flat SPICE netlist example.

The Spice formats

Capture provides two SPICE netlist formats. The first format produces either hierarchical or flat netlist output, depending on your project structure and the active view. It is accessible from the SPICE tab on the Create Netlist dialog box. The second format produces only flat netlists, and is accessible through the Other tab on the Create Netlist dialog box.
Use the SPICE tab if:

- You want to include net, part, and pin properties.
- You want a hierarchical netlist.

  Use the Other tab if:
- You want a flat netlist of a simple hierarchical design.

Hierarchical designs in SPICE

For hierarchical designs, the SPICE format produces netlists with subcircuit (.SUBCKT) definitions for schematic pages in the hierarchy. These subcircuits are called by the X command (subcircuit call). Since SPICE does not require subcircuits to be defined before use, the hierarchy appears in normal form in the netlist with the root page at the top of the file.

According to the PSPICE manual, the X subcircuit general form is:

```
X name [nodes] subcircuit-name
```

where:
name Specifies a unique name for the device
nodes Specifies the list of nets that attach to the device in the same order as the .SUBCKT definition used by the device
subcircuit-name Specifies the name of a .SUBSCKT definition that the X device uses

The following is an example of an X subcircuit call:

```
XBUF 13 15 UNITAMP
```

For more information on hierarchical SPICE netlists, see the hierarchical SPICE netlist example.

SPICE netlist constraints

- Capture can create netlists larger than what most PC-based SPICE programs accept. Consult your SPICE manual for the limits. If your PC meets SPICE's memory requirements, you can generate the largest allowed netlist.
- The part value is used to pass modeling information to the netlist. For instance, resistor RS1 in the example flat schematic folder has a value of 1K Ohms; in the example hierarchical schematic folder R1 has a value of 6.8K Ohms.
- Use the special PSPICE.OLB or SPICE.OLB libraries supplied by OrCAD when generating a SPICE netlist. These libraries already have pin numbers on the parts and are compatible with most versions of SPICE. The PSPICE.OLB contains many specific part types, such as a 2N2222 NPN transistor, that are not provided in the generic SPICE.OLB.
- All library part pin names should be changed to reflect the model node index. To find out the proper node ordering, see your SPICE manual.

As an example of what to change, the OrCAD-supplied NPN transistor has the pin names defined as base, emitter, and collector in the DEVICE.OLB library. For SPICE to understand the nodal information, the pin names must be changed from base, emitter, and collector to 2, 3, and 1 (as defined in the SPICE manual). Therefore, the library source file for an NPN transistor that is compatible with the SPICE pin numbering convention is as follows:

```
'NPN'
REFERENCE 'Q'
{X Size =} 2 {Y Size =} 2 {Parts per Package =} 0
L1 SHORT IN '2'
B2 SHORT IN '3'
T2 SHORT IN '1'
{ 0}..##.#
{ 1} ## #
  .
  .
  .
```

SPICE map files

In addition to the netlist file, Capture also creates a map file when you select the SPICE format. The node numbers created by Capture are placed in the .MAP file so you can cross-reference the SPICE node numbers with the node names that you specified on your schematic page. You must

enter the map filename in the Map File text box in the Create Netlist dialog box.

> If you select the Use net names option, the map file may contain erroneous results.

For more information on SPICE map files, see the flat and hierarchical map file examples.

SPICE pipe commands

You can place lines of text on your schematic page to be included in the SPICE netlist. Select the Text command on the Place menu to place the text on your schematic page.
Each line of text must start with the pipe character (|). The first line must be:

```
|SPICE
```

This tells Capture to extract the information in the following lines of text when generating a SPICE netlist. The remaining lines can contain any information you want to include in the netlist. The lines following |SPICE are placed at the top of the netlist.

15

Physical Layout in PCB Editor

After creating a schematic and verifying the logic by simulating the design in PSpice, the next step in the design process is to create the physical layout of the PCB board in PCB Editor; the Cadence tool for designing physical layout of a PCB board.
Capture offers full integration with Cadence® PCB Editor tool suite, allowing you to use all of Capture schematic design capabilities to enter your PCB projects, then export the information to PCB Editor for layout and routing.
While the actual board design tasks are performed in PCB Editor, there are a few tasks that must be performed in Capture to prepare the schematic for the layout.
This chapter lists the design tasks and the best practices that must be followed during the schematic capture stage to ensure that process of exporting data to PCB Editor is completed smoothly.
The tasks covered in this chapter are:

Preparing the Schematic for Layout

Before you design the physical layout of your schematic in PCB Editor, you should validate your design to ensure the that the object names used in schematic follow the object naming convention required in PCB Editor. This section list some of the recommendations or best practices to be

followed in Capture to ensure that schematic is successfully exported to PCB Editor.

Best practices for Capture-PCB Editor flow

- Avoid using parenthesis "( )" in schematic names. If you still want to use parenthesis in schematic name then make sure that you map it with a valid character while generating a netlist and do reverse mapping while generating a `.swp` file.
- Naming nets, parts, or pins:
  - Keep the maximum length of a net name or alias up to 31 characters.
  - Limit part and pin names to 31 characters.

> If limit of 31 characters cannot be achieved, ensure that before you generate PCB Editor netlist, you modify the this limit in the Setup dialog box. This dialog box is invoked when you choose Setup in the PCB Editor tab of the Create Netlist dialog box.

  - Use only upper case characters for part/symbol names, reference designators, and pin names. Do not use lower case characters.
  - Do not use special characters in a net names, part names, reference designators, or pin names.
- Do not use "0" as a pin number.
- Do not use duplicate names for pins other than power pins.
- For multiple power pins with the same pin name, do not make some pins as visible and others as invisible.
- Avoid using "Power Pins Visible" property at design-level.
- Use net to connect pins.
  - Leave room for assigning a net name. Pin-to-pin connection changes the net name when a user moves a component.
- Run the Capture DRC command before generating PCB Editor netlist.
- Set path for PCB Editor footprint before running Netrev.
- Do not use {GROUP} as property name in combined property strings. This may cause problems while annotating your design for a layout in PCB Editor. The GROUP property is used in PCB Editor for a specific purpose.
- Do not use a part in your design, which does not contain a logical pin and contains only a power pin. You will not be able to create an Allegro netlist for the design.
- Capture allows you to assign the SIGNAL_MODEL property and pass it to the back-end tool. However, you cannot use it to create XNets. You can create XNets only in Allegro PCB Editor.

Unsupported Capture-PCB Flow field values

You should avoid the use of the following special characters when defining **pin names**, **net names** or **signal names** in the Capture - PCB Editor flow:

- leading or trailing white spaces
- ! (exclamation mark)
- ' (single-quote)

⚠	Both the backslash (\ ) and underscore (_) characters in net names interfere with cross probing. Also, the design name must not contain period (.).

Assigning Physical Properties to a Schematic Design

While preparing a design for layout, you can add some physical design properties either as Instance properties or as Net properties. On generating the physical netlist the values for some of the properties are dumped in the netlist. This is then read by PCB Editor for board creation.

⚠	Only the properties that have values assigned to them and are listed in the configuration file are exported to PCB Editor. To know more about property flow from OrCAD Capture to Allegro PCB Editor, see Property Flow from Capture to PCB Editor.

This section lists the properties specific to the Capture-PCB Editor design flow and can be assigned to the Capture part.

Methods to assign properties in Capture for use in PCB Editor

There are several ways to make property assignments if the parts are already placed in your design:

- In the schematic design editor, select the part to which properties are to be added, then right-click and choose Edit Part from the pop-up menu to activate the part editor. Choose the Part Properties command from the Options menu to add properties such as PCB Footprint and CLASS. After modifying the part and closing the User Properties window, click the Update All button when you close the property editor window to make sure the properties are propagated to every like part in the design. You can use the Capture-Allegro filter in the Property Editor to view typical properties that may be assigned in Capture to be used in Allegro PCB Editor. (This list of properties comes from the PREFPROP.TXT file,

placed in your Capture folder during installation). For information about the Allegro PCB Editor properties, see document *Allegro Platform Properties Reference* guide (*propref.pdf*).

- Modify your parts in the library and then use the Replace Cache command on the Design menu with the Preserve schematic part properties option. If you don't have the original library, you can create a library and copy the parts from the design cache to the new library, then modify the parts in the new library.
- Add or edit properties manually in the Property Editor. You can access the property editor by double-clicking on the part or selecting a part and then choosing Edit Properties from the pop-up menu. If you change the occurrence values on the part, occurrence values are used instead of the values found in the library. If you have not changed the occurrence values, then the values are those from the original library.
- Use the Update Properties command from the Tools menu with an update (.UPD) file which adds properties and corresponding values to one or more components.

Assigning Footprint Properties

Why footprint?

In order to correctly map a logical component to the physical board environment, you need to specify the footprint information for all design parts that are to be included in the PCB. This is an instance-level property that is assigned on all schematic components that are to be included in the physical board design.

Schematic-only parts, such as offpage connectors, do not have the footprint information assigned to them. If you have simulated your design in PSpice, you might have simulation-only parts that should not have footprint information associated with them. For more information on such parts, see Physical Layout of a Simulation Design.

Locating Footprints in Install Hierarchy

With PCB Editor installation, Cadence ships a library of footprints and padstacks. These can be accessed from <*install_hierarchy*>/share/pcb/pcb_lib/symbols.

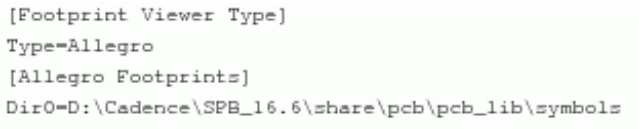
If you need to use custom footprints or padstacks, you can copy them at

<*install_hierarchy*>/share/local/pcb/symbols or <*install_hierarchy*>/share/local/pcb/padstacks, respectively. Copying footprint symbols at this location ensures that the custom footprints are accessible during component placement, without making any modifications to the PCB Editor setup.

In case you do not want to copy the footprints, you need to modify the PCB Editor setup to include the location of custom footprints and custom padstacks in psmpath and padpath variables, respectively. For this complete the following steps in PCB Editor.

1. Choose *Setup - User Preferences* or use the enved command.
2. From the Categories list in the User Preferences dialog box, select *Paths - Library*.
3. Modify the padpath and psmpath to include the location of custom footprints.

Once you have finalized the footprint location, update the capture.ini to ensure that it has a section on Allegro Footprints. The variables in this section define the directory locations for the psm and pad files. The relevant section of the capture.ini is shown in the following figure.



If you are using the custom footprints, the capture.ini must be updated such that Dir0 or DIR1 variables point to the folder containing the footprints. This section of capture.ini file is also read by the 3D Footprint Viewer, to display component footprint in Capture.

Do all parts in my design have footprint information?

Capture provides multiple ways to verify whether or not a design component has footprint information associated with it. Depending on factors such as design size and the numbers of components, you can use the method convenient to you.

To check if a part has footprint information, launch Property Editor using one of the following methods.

- Double-click on the part
  Or
- Right-click on the part and select Edit Properties.

The footprint information is displayed against the PCB Footprint property.

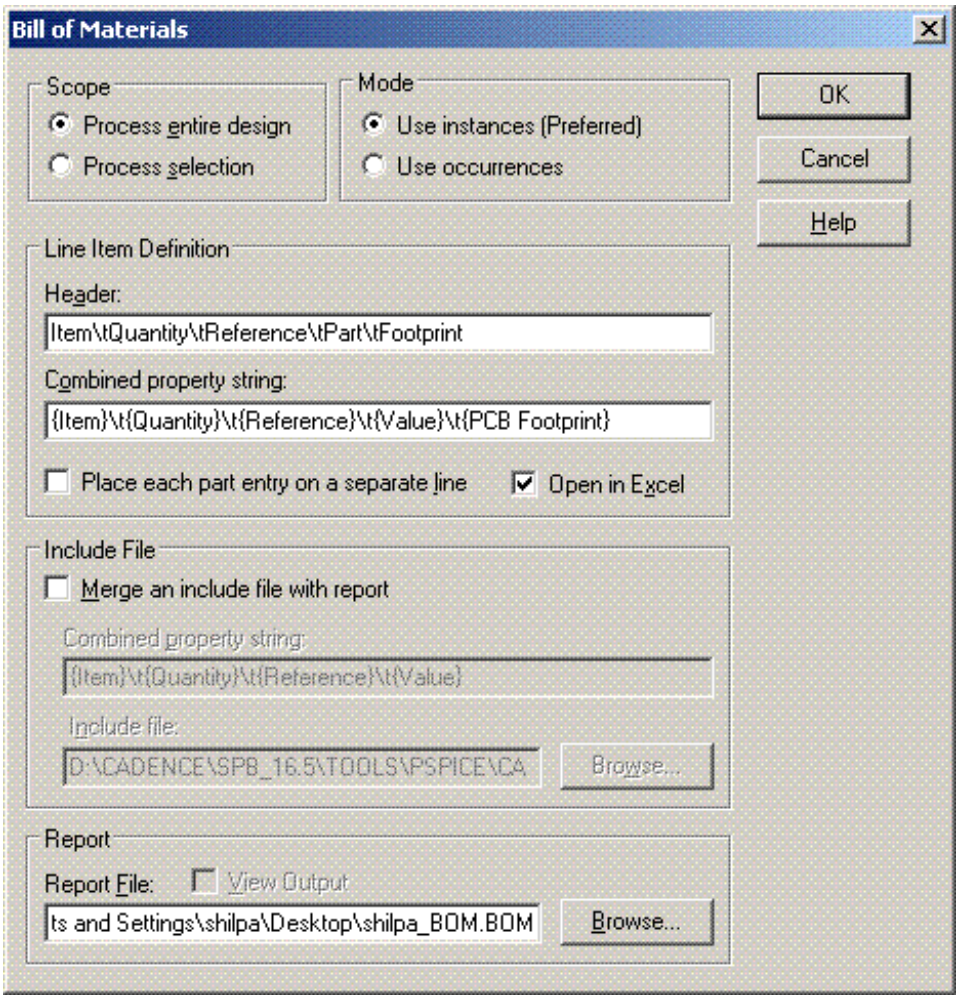
To simultaneously check if multiple parts have footprint information, select the parts on the schematic page and launch Property Editor. The PCB Footprint row displays the value for each selected part.

To ensure that all parts in your design have footprint information, enable the *Check missing/illegal PCB Footprint property* DRC in the Physical Rules tab of the Design Rules check dialog box.

If required, you can also include the footprint information for the parts in the Bill Of Materials (BOM) generated for your design. To generate a custom BOM with footprint information, complete the

following tasks.

1. In Project Manager, select the design file.
2. Choose *Tools - Bill of Materials*.
3. Modify the default entry in the Line Item Definition section of the dialog box
   In the Header text box, append \tFootprints.
   In the Combined property string value text box, append `\t{PCB footprint}`.
4. Generate BOM.



The generated BOM has footprint information added in the last column.

Assigning footprints to components

While assigning a footprint to a schematic symbol, ensure that the number of pins in the schematic symbol match the number of the PCB footprint symbol, and the pin numbers in the schematic symbol match the pin numbers in the footprint.

Assigning Footprint to one component

Using the Property Editor, you can assign or change the footprint data associated with a part.

1. Select the part and launch Property Editor.
2. Enter the required value in the text box next to the PCB Footprint property.

Assigning same footprint to multiple components

1. Select the components and launch Property Editor.
2. Select the PCB Footprint row (or column), right-click and select Edit.
3. Add the required value in the Edit Property Value dialog box.

For creating a board in PCB Editor, it is must to assign a valid PCB Footprint property for each part in the design. This includes all mechanical parts in your design, although, in this case, you can use a "dummy" value for the PCB Footprint property. While designing a board using Capture-PCB Editor flow, combined property strings can be used to pass user-defined properties as PCB Footprint property for PCB Netlist generation. This gives you the flexibility of defining a user-defined PCB Footprint property specifically for the PCB Editor flow. As a result, you can define different PCB Footprint properties for different PCB flows.

Specifying Alternate Footprints

While specifying component footprint, the ALT_SYMBOLS property can be used to specify a list of alternate footprint names that can be used to substitute the primary footprint during interactive placement in PCB Editor. The ALT_SYMBOLS property can be assigned to a package or to a component either at the library-level or at the instance-level.

The syntax of the ALT_SYMBOLS is as follows:

```
ALT_SYMBOLS '(Subclass:Symbol,...;Subclass:Symbol,...)'
```

where, Subclass can either be TOP (or T) for top layer, or BOTTOM (or B) for bottom layer and Symbol is a standard footprint name.

For example, you can assign the component 7400 the following ALT_SYMBOLS value, T:dip14_3;B:dip14_3

Viewing Footprints in Capture

While working in Capture CIS, you can view the footprint information associated with the part in the PCB Editor 3D Footprint viewer. This viewer provides a three dimensional view of the footprint symbol of a selected part on the schematic or the part editor. Along with the footprint symbol, the viewer also displays pin numbers and pin names.

	Footprint Viewer is available with Capture CIS only.

3D Footprint Viewer

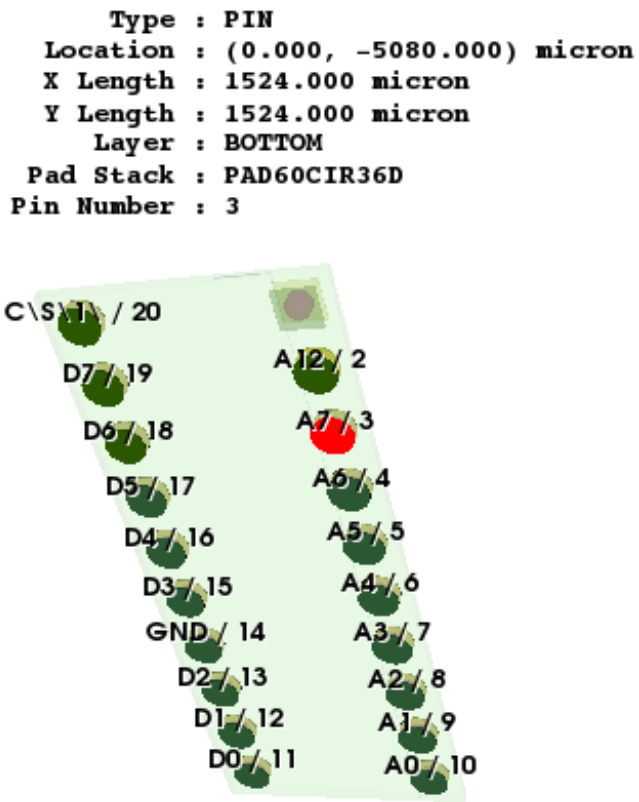
The viewer is available from within the schematic, the part editor or CIS Explorer.

- To view the footprint, right-click on the part and choose *Show Footprint*.

The footprint viewer opens with the three dimensional view of the part displayed.

⚠	If the footprint information for the part is not available, the viewer displays a blank canvas with an error in the session log.

Footprint viewer can also be used to display footprint and pin details. To view the footprint details, right-click on the footprint in the viewer. The details are displayed at the top of the viewer. Similarly, to view the pin details, in the footprint viewer, right-click on the pin. Pin details are displayed at the top of the viewer.



Footprint Views

The Footprint Viewer can be used to display the standard as well as non-standard footprint views. Use the Footprint Viewer Toolbar to view the standard footprint views.
To view the footprint from any non-standard angle, in the viewer window:

1. Select the footprint viewer and hold the left mouse button down.
2. Keeping the mouse button pressed, move the mouse within the viewer window.

As you move the mouse button, the footprint rotates within the viewer.

> From any non-standard view position, you can always use the toolbar view buttons to go back to any standard view.

Moving the footprint

You can move (reposition) the footprint anywhere within the viewer.

- Select the footprint, keeping the Shift key pressed, drag the footprint to any point on the canvas.

Mapping symbol pins with footprint pins

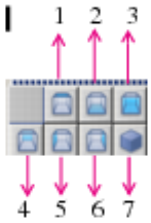
With the footprint viewer displayed, select a symbol pin on the part. The corresponding pin is highlighted in the footprint viewer, and the pin details are also displayed at the top of the viewer.

Footprint Viewer Toolbar

To display the PCB Footprint toolbar, choose *View - Toolbars - Footprint Viewer*.

Table 2-1 Footprint Viewer Toolbar

ToolBar Button	Usage
Toggle Footprint Viewer 	You can toggle the footprint viewer between show and hide mode by using the Show footprint viewer button. This switch only hides the viewer. So the footprint remains on the canvas in the hide mode.



The PCB footprint in the viewer can be viewed from different perspectives. The Footprint Viewer toolbar provides shortcuts to the following standard views:

1. Top - displayed on selecting the *Show Top* button
2. Bottom - displayed on selecting the *Show Bottom* button
3. Front - displayed on selecting the *Show Front* button
4. Back - displayed on selecting the *Show Back* button
5. Left - displayed on selecting the *Show Left* button
6. Right - displayed on selecting the *Show Right* button
7. Isometric (the angles between the projection of the x, y, and z axes are all the same, or 120°) - displayed on selecting the *Show Isometric View* button

View Axis

Displays the x, y and z axis in the bottom left corner of the viewer. When you are in a non-standard view position, you may need to identify the actual angle or preoperative of the footprint. This may not always be very easy to identify simply by looking at the view. To easily view the probative of the footprint, switch to the View Axis mode. The x, y and z axes are displayed in the same perspective as the footprint.

As you change the perspective of the footprint the perspective of the axis changes. This is enables you to identify the current propitiate of the footprint. The axis view especially useful when you use the mouse to change the view out of the standard views.

Measure Tool

Used to measure any distance on the footprint viewer canvas. An added feature to this tool is that is allows you to measure across different axis. This means that you can rotate the tool to any standard or non-standard view and use this tool to measure distances between any two points.

1.
    a. Select the Measure Tool.
    b. Click the start point of the distance you want to measure.
    c. Click the end point.
    The distance (in microns) displays at the top of the viewer. Also, as you move the mouse from the start point to the end point, a ruler displays (and extends with the mouse movement) on the canvas between the start and end points.
    d. To reposition either the start or end point of the ruler, click on the point to reposition.
    Notice a wireframe appears at the point you clicked.
    e. Click the wireframe and reposition the start or end point of the ruler.
    You can move this point back and forth along the same direction to measure in the line. You can also move the this point in any other direction (on any axis) to measure in other directions or even across

axis.

f. Click the other end of the ruler to make this end also movable. Now you can reposition either ends of the ruler.

> If you place the ruler on the canvas, the ruler view will change as you change the view of the footprint either using the standard or non-standard viewing.

Footprint Viewer Zoom 	Use the zoom buttons on the Footprint Viewer toolbar to zoom the footprint on the canvas. If you are using a scroll mouse, you can use the scroll wheel to zoom in and out of a footprint on the canvas.

Troubleshooting Footprint Viewing Errors

Problem: Footprints not displaying in the 3D viewer.
Solution: When viewing a footprint of a part, the footprint viewer selects the footprint to be displayed in the following order of preference:

- PCB Footprint property defined on the part instance.
- PCB Footprint defined in the CIS table, if the part is a linked database part.
- PCB Footprint property defined in the package properties of the part.

If none of the above properties/values are found, the viewer displays a blank canvas with an error in the session log.

- *ERROR(ORCAP-1732): Could not retrieve PCB Footprint property. Unable to show footprint in viewer.*

  Ensure that for the selected component, a valid footprint value is defined for the PCB Footprint property.

- *ERROR(ORCAP-1731): Allegro Footprints section could not be located in Capture.ini. Unable to show footprint %s in viewer.*

  Check the capture.ini to ensure that it has a section on Allegro Footprints. The variables in this section define the directory locations for the psm and pad files. These files are used by the viewer to display the footprint and pin information, respectively, in the viewer. The relevant section of the capture.ini is shown in the following figure.

```
[Footprint Viewer Type]
Type=Allegro
[Allegro Footprints]
Dir0=D:\Cadence\SPB_16.6\share\pcb\pcb_lib\symbols
```

For details on creating and editing the variables (including the Allegro Footprint variables) in the Capture.ini see the *OrCAD Capture Quick Reference*.

Instance-Level Properties for Physical Design

Besides the footprint property, there are other instance-level properties that can be defined on schematic components but are used during physical design. This section talks about some of these properties. Unlike footprint property, which is a must for board created, using the properties covered in this section is optional.

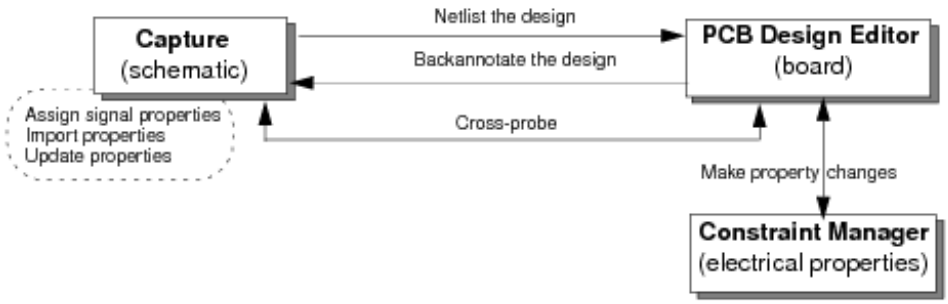
ROOM Property

The ROOM property is a component instance property that is used to group a set of components on the PCB board. For example, to ensure that a set of components are placed together in a particular section of the PCB board, add ROOM property on the schematic components and also define a room in PCB Editor, components with same value of the ROOM property are placed in the defined area in the physical board design.

In Capture, use Property Editor to specify a value for the ROOM property on the schematic component.

Defining Properties on Nets

While designing a schematic in Capture, you can specify high-speed constraints as net properties and take them through a complete front-to-back flow. As net properties are passed to the physical netlist generated by Capture, these constraints are also passed to the PCB Editor. In PCB Editor, you can modify these constraints by launching Constraint Manager. Following figure shows the flow of signal properties.

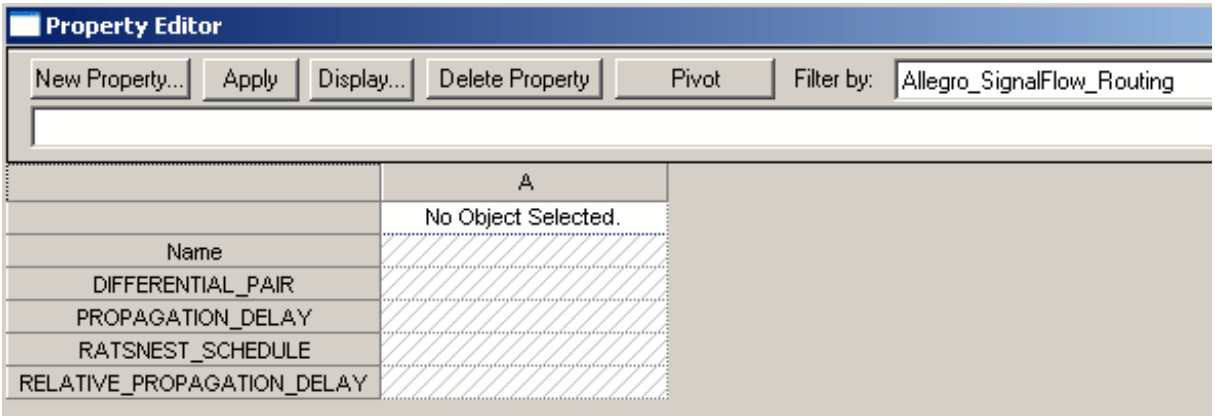


To enable cross-probing between OrCAD Capture and Constraint Manager, enable highlighting in PCB Editor. When you highlight a component in PCB Editor, PCB Editor sends cross-probing messages to OrCAD Capture. You can disable cross-probing messages in Capture by deselecting Enable Intertool Communication under Intertool Communication in the Miscellaneous tab of the Preferences dialog b

To view the list of high speed signal properties supported by Capture:

- Launch Property Editor, and from the *Filter by:* drop-down list, select *Allegro_SignalFlow_Routing*.

The properties are listed in the Flat Nets tab.



For more information about the function of signal properties, see the Cadence *Allegro Platform Properties Reference* documentation.

Use the Schematic Nets tab to display the net properties on schematic. Adding or modifying high-speed signal properties is supported using the Flat nets tab of the Property Editor.

Signal Property Flow

This section lists the top-level design tasks to be performed for a design that has high-speed signal properties defined as net properties.

1. Make a schematic.
2. Select the net on which high-speed signal properties are to be assigned.

3. Assign one or more of the following signal properties.

    - PROPAGATION_DELAY
    - RATSNEST_SCHEDULE
    - RELATIVE_PROPAGATION_DELAY
    - DIFFERENTIAL_PAIR

4. (Optional) Import properties.
   If you have properties recorded in a text file in a previous session in the Property Editor, you can import those properties.
5. (Optional) Update properties
   If you have a large design and want to avoid manual property assignment, you can update properties using the update property text file.
6. Export properties
   It is a good practice to export properties to a text file. These files can be used for archival purpose and as backup in the unlikely event of accidental data corruption or deletion.
7. Perform a design rule check on the design.
8. Netlist the design.
9. Open PCB Design Editor and launch Constraint Manager.
    a. Open the board file in the PCB Editor.
    b. Click *Setup - Electrical Constraint Spreadsheet* to open the Constraint Manager.
    c. Expand the *Routing* spreadsheet.
    d. Select the *Min/Max Propagation Delays* tab to view the PROPAGATION_DELAY property or the Relative Propagation delay tab to view the RELATIVE_PROPAGATION_DELAY property.
    e. Select the *Wiring* tab to view the RATSNEST_SCHEDULE property. This property appears in the *Schedule* column in the *Topology* section.
10. Backannotate property changes.
    If you make any signal property changes in Constraint Manager, you need to backannotate those changes to Capture.
    a. Open the board file in the PCB Editor and open the Constraint Manager.
    b. Make the desired property changes in Constraint Manager.
    c. In Capture, select *Tools - Back Annotate.*
       The Backannotate dialog box appears.
    d. Select the Allegro tab, set different backannotation options and click *OK*.
       The Progress window reports the details of backannotation.
    e. Open the Property Editor and verify any changes.

Limitations of the signal property flow

The signal property flow has the following limitations:

- You cannot define extended nets.
- You cannot use multi match group power.

PROPAGATION_DELAY

This property defines the minimum and maximum propagation delay constraint between any pair of pins in a net. By assigning this property to nets, you can make the router restrict the length of interconnect to meet timing margin. This property often is best applied to a common clock sourced designed bus.

To specify the propagation delay on a net, launch Property Editor and select the filter to display the signal flow properties. To edit or specify the value for the propagation delay, select the Flat Nets tab, and perform the following steps.

1. Select the grid corresponding to the PROPAGATION_DELAY property.
2. Choose *Edit - Invoke UI* or press the CTRL+U shortcut keys.
   The Propagation Delay dialog box appears.

> Alternatively, you can right-click on the grid corresponding to the PROPAGATION_DELAY property and from the pop-up menu, select the Invoke UI command.

3. To specify the pin-pair, select one of the following options in the *Pin Pair* field.
   - Longest/Shortest pin-pair(L:S)--To apply minimum delay to the shortest pin-pair and maximum delay to the longest pin-pair.
   - Longest/Shortest Driver/Receiver(D:R)--To apply minimum delay to the shortest driver/receiver pin-pair and maximum delay to the longest driver/receiver pin-pair.
   - All Drivers/All Receivers(AD:AR)--To apply Min/Max constraints to all driver/receiver pin-pairs.
4. To create a new pin-pair, click the *Add Pin Pair* () button or press the ALT+A shortcut keys.
   The Create Pin Pairs dialog box appears. Select the first pin for the pin-pair, then select the second pin, and click *OK.*
   A pin-pair is created. The new pin-pair appears as a row in the Propagation Delay dialog box. You can define constraints for it
   You can use the following methods to select multiple consecutive pins in the Create Pin Pairs dialog box:
   - Using Shift+Down Arrow keys
   - Using Shift+Left mouse button click
   - Dragging the mouse pointer diagonally across the pins appearing in the combo box to select them
     Similarly, you can use the CTRL+Left mouse button click to select multiple

nonconsecutive pins in the Create Pin Pairs dialog box.

5. Enter a value specifying the minimum allowable propagation delay/length for the pin-pairs in the *Min* field.
6. To specify the unit for minimum constraint, select one of the following options in the *Min Rule* field:
    - DELAY in ns
    - %MANHATAN
    - LENGTH in mills (mils), micron (um), millimeter (mm), centimeter (cm), and inches (in)
7. Enter a value specifying the maximum allowable propagation delay/length for the pin-pairs in the *Max* field.
8. Specify the unit for the maximum constraint by selecting unit value in the *Max Rule* field.
9. Click OK *in the Propagation Delay dialog box.* The PROPAGATION_DELAY property is seeded in the PROPAGATION_DELAY grid for the corresponding column.
10. Click the Apply button in the Property Editor to apply the PROPAGATION_DELAY property on the nets. Moreover, if you have manually entered the PROPAGATION_DELAY syntax, then Capture performs syntax validation and appends any syntax violations to the Session Log.

⚠	To delete an existing pin-pair, select the left-most cell of the pin-pair row and click the *Delete Pin Pair* () button or press the ALT+D shortcut keys.

You can use the User Properties dialog box to assign the PROPAGATION_DELAY property to all the bits of a bus at the same time. Make sure that you use the correct syntax for specifying a value for the PROPAGATION_DELAY property. The syntax is:

```
<Pin_pair>:<min_value>:<max_value>
```

The pin-pairs can only be:

- L:S
- D:R
- AD:AR

You can also manually enter values in the grids corresponding to the PROPAGATION_DELAY property. After you enter a value in the PROPAGATION_DELAY property grid and click the Apply button, Capture performs syntax validation and if there is a syntax violation, the property is not applied and the details of the violation are appended to the Session Log.

You can populate multiple consecutive or nonconsecutive grids of the PROPAGATION_DELAY property at the same time. To do this, select the grids you want to populate and press the CTRL+E

shortcut keys. The Edit Property Values dialog box appears. Specify the value that you want to be populated across all the selected grids in the dialog box. You can also use the shortcut keys CTRL+C and CTRL+V to perform standard copy/paste operations in the PROPAGATION_DELAY property grids.

RATSNEST_SCHEDULE

This property specifies the type of ratsnest calculation that Constraint Manager performs on the net.

By using the RATSNEST_SCHEDULE property, you can meet a balance between time margin and noise margin. Based on your design need, you can define the configuration as MIN_TREE, MIN_DAISY_CHAIN, SOURCE_LOAD_DAISY_CHAIN, FAR_END_CLUSTER or STAR. This property useful in defining the placement of receiver or driver in multi-drop buses and asynchronous signals.

To specify the RATSNEST_SCHEDULE property:

1. Select the grid corresponding to the RATSNEST_SCHEDULE property.
2. From the drop-down list, select any of the following values:
    - MIN_TREE--Indicates that the net rat should be displayed with the minimum spanning tree algorithm. Selecting this option can lead to formation of Ts at pins.
    - MIN_DAISY_CHAIN--Indicates that a minimum length daisy-chain schedule is formed.
    - SOURCE_LOAD_DAISY_CHAIN--Indicates that a source-to-load ECL daisy-chain schedule is used.
    - FAR_END_CLUSTER--Automatically places a single Tpoint in a schedule at a calculated location.
    - STAR--Specifies a ratsnest similar to FAR_END_CLUSTER without the Tpoint added.

RELATIVE_PROPAGATION_DELAY

This property is an electrical constraint attached to pin-pairs on a net. It specifies a group of pin-pairs that are required to have interconnect propagation delays matching a specified delta (offset) and tolerance with respect to the target pin pair. A RELATIVE_PROPAGATION_DELAY group has a pin-pair against which all other pin-pairs in the group are compared. You can apply the RELATIVE_PROPAGATION_DELAY property to a source synchronous bus design, such as DDR interfaces.

To specify RELATIVE_PROPAGATION_DELAY property on a net, perform the following steps.

1. In the *Flat Nets* tab of the Property Editor, select the grid corresponding to the RELATIVE_PROPAGATION_DELAY property.
2. Select the Edit menu and choose the Invoke UI command.
   The Relative Propagation Delay dialog box appears.

	Alternatively, you can right-click on the grid corresponding to the RELATIVE_PROPAGATION_DELAY property and select the Invoke UI command from the pop-up menu or press the CTRL+U shortcut keys.

3. To specify the pin-pair, select one of the following options in the *Pin Pair* field.
   - Longest/Shortest pin-pair--To apply minimum delay to the shortest pin-pair and maximum delay to the longest pin-pair.
   - Longest/Shortest Driver/Receiver--To apply minimum delay to the shortest driver/receiver pin-pair and maximum delay to the longest driver/receiver pin-pair.
   - All Drivers/All Receivers--To apply Min/Max constraints to all driver/receiver pin-pairs.
4. Select the scope as global or local. Select the scope as global to define the RELATIVE_PROPAGATION_DELAY property between different nets of same match group. Select the scope as local to define the RELATIVE_PROPAGATION_DELAY property between different pin-pairs of same net.
5. Enter the relative value from the target net that all nets in the group should match in the *Delta* field.
6. To specify the unit for delta, select *Delay* in ns or *Length* in mills (mils), micron (um), millimeter (mm), centimeter (cm), and inches (in) in the *Delta Units* field.
7. Enter a value that specifies the maximum allowable propagation delay/length for the pin-pairs in the *Tolerance* field.
8. To specify the unit for Tolerance, select one of the following options in the *Tol. Units* field:
   - %
   - DELAY (ns)
   - LENGTH (mils, mm, cm, in)
9. To create a new pin-pair, click the *Add Pin Pair* () button or press the ALT+A shortcut keys. The Create Pin Pairs dialog box appears.
10. Select the first pin for the pin-pair, then select the second pin, and click *OK*.
    A pin-pair is created. The new pin-pair appears as a row in the Propagation Delay dialog box. You can define constraints for it.
    You can use the following methods to select multiple consecutive pins in the Create Pin Pairs dialog box:
    - Using Shift+Down Arrow keys
    - Using Shift+click
    - Dragging the mouse pointer diagonally across the pins appearing in the combo box to select them

    Similarly, you can use the Ctrl+click to select multiple nonconsecutive pins in the Create Pin Pairs dialog box.

11. To delete an existing pin-pair, select the pin-pair row by clicking its left-most cell, and click the *Delete Pin Pair* () button or press the ALT+D shortcut keys.
12. To set a pin-pair as the target net, select the pin-pair row and click the *Set Target* () button or press the ALT+S shortcut keys.
    Target Pin Pair name is displayed and the Delta and Tolerance fields for the target pin-pair is set to'0'.
13. To delete the target status from a pin-pair, select the pin-pair row and click the *Delete Target* () button or press the ALT+T shortcut keys.
14. To change the match group:
    - Select a group from the list box.
    - Type a new match group name.

    Based on the match group selected, all nets contained in it will display in the *Nets Attached* box.

15. Click OK *in the Relative Propagation Delay dialog box.* The RELATIVE_PROPAGATION_DELAY property is seeded in the RELATIVE_PROPAGATION_DELAY grid for the corresponding column.
16. Click the Apply button in the Property Editor to apply the RELATIVE_PROPAGATION_DELAY property on the nets. Moreover, if you have manually entered the RELATIVE_PROPAGATION_DELAY syntax, then Capture performs syntax validation and appends any syntax violations to the Session Log.

You can use the User Properties dialog box to assign the RELATIVE_PROPAGATION_DELAY property to all the bits of a bus at the same time. Make sure that you use the correct syntax for specifying a value for the RELATIVE_PROPAGATION_DELAY property. The syntax is:
For the target pin-pair:

```
<match_group>:<scope>:<pin-pair>::
```

where <pin-pair> has the following syntax:
<pin1>:<pin2>
valid values of <scope> are L for local and G for global.

For non-target pin-pairs:

```
<match_group>:<scope>:<pin-pair>:<delta>:<tolerance>
```

The pin-pairs can only be:

- AD:AR
- L:S

- D:R

You can also manually enter values in the grids corresponding to the RELATIVE_PROPAGATION_DELAY property, and click *Apply*. When you select the Apply button, Capture performs syntax validation and if there is a syntax violation, the property is not applied and the details of the violation are appended to the Session Log.
You can use the shortcut keys CTRL+C and CTRL+V to perform standard copy/paste operations in the RELATIVE_PROPAGATION_DELAY property grids.

⚠	You can populate multiple consecutive or nonconsecutive grids of the RELATIVE_PROPAGATION_DELAY property at the same time. To do this, select the grids you want to populate and press the CTRL+E shortcut keys. The Edit Property Values dialog box appears. Specify the value that you want to be populated across all the selected grids in the dialog box.

DIFFERENTIAL_PAIR

This property represents a pair of flat nets that will be routed in a way that the signals passing through them are opposite in sign with respect to the same reference. This ensures that any electromagnetic noise in the circuit is cancelled out.

Creating Differential Pairs using Property Editor

To specify the DIFFERENTIAL_PAIR property for flat nets:

1. Right-click the design in the project manager and select Edit Object Properties. The Property Editor window appears.
2. Click the Flat Nets tab at the bottom in the Property Editor window.
3. Select the first flat net for which you want to create a differential pair.
4. Select the grid corresponding to the DIFFERENTIAL_PAIR property and specify a name for the differential pair.
5. Select the second flat net for which you want to create a differential pair.
6. Specify the same differential pair name you specified for the first net.
7. Click the Apply button. A differential pair between both the nets is created.

⚠	For more information about the DIFFERENTIAL_PAIR property, see Cadence document *Allegro Platform Properties Reference*

Create Differential Pair Command

In addition to creating a differential pair using the DIFFERENTIAL_PAIR property in the property editor, you can use the _ Create Differential Pair command to create a differential pair between two flat nets in your design. You can also modify or delete a differential pair from your design.

1. In the project manager, click the design file (.dsn) or a schematic page file.
2. Select the Tools menu and choose the Create Differential Pair command. The Create Differential Pair dialog box appears.
3. Ensure that the Net option is selected from the drop-down list. All the flat nets in the design are listed in All Nets column in a sorted order (all net names starting with a numeric character will be displayed first and then all net names starting with an alphabet).

> To view nets of a particular type, specify the initial letters of the net in the Filter text box. All the nets of that particular type will appear in the All Nets column. For example, if you want to view all nets starting with the letter "A", then enter "A" in the Filter text box. All the nets starting with letter "A" will appear in the All Nets column.

4. Select a net from the All Nets column and click the button or double-click the net. The selected net appears in the Selections column.
5. Repeat step 4 for the second net to be included in the differential pair.
   To remove a net from the Selections column, double-click on the net name or select a net and click the button.
   Once two nets are available in the Selections list-box, the Create button is enabled.
6. Specify a name for the differential pair in the Diff Pair Name text box.
7. Click the Create button. The differential pair is created between the selected nets. The differential pair name appears in the Selections column adjacent to the net name.

> If the selected nets are not of the same type (for example, a power net and a non-power net) or they differ in the total number of pins in each selected net, then a message appears asking you to confirm the creation of a differential pair between the selected nets.

8. Click the Close button to close the Create Differential Pair dialog box.

For steps on how to create a differential pair using the property editor, see Creating Differential Pairs using Property Editor.

- The DIFFERENTIAL_PAIR property column is automatically updated with the differential pairs you create using the Create Differential Pair dialog box.
- An Auto Differential pair can also be created for a bus. To do so, you need to put *n* & *p* as prefix and the Auto command creates differential pairs for all bits in the bus.

To view a differential pair

1. In the Create Differential Pair dialog box, select the Differential Pair option from the drop-down list. All the differential pairs you created in your design appear in Diff Pairs column.

   To view differential pairs of a particular type, specify the initial letters of the differential pairs in the Filter text box. All the differential pairs of that particular type will appear in the Diff Pairs column. For example, if you want to view all differential pairs starting with the letter "DP", then enter "DP" in the Filter text box. All the differential pairs starting with letter "DP" will appear in the Diff Pairs column.

2. Select a differential pair from the Diff Pairs column and click the (>) button or double-click the differential pair. The Selections column will display the name of the two nets associated with the selected differential pair.

   You can use the CTRL or SHIFT keys to move multiple differential pairs to the Selections column and view the nets associated with the selected differential pairs.

3. Click the Close button to close the Create Differential Pair dialog box.

To modify a differential pair

1. In the Create Differential Pair dialog box, select the Differential Pair option from the drop-down list. All the differential pairs you created in your design appear in Diff Pairs column.
2. Select the differential pair you want to modify from the Diff Pairs column and click the (>) button or double-click the differential pair. The selected differential pair along with the associated nets appears in the Selections column.

⚠	In case, you select a wrong differential pair for modification and want to revert back, double-click the differential pair in the Selections column. The differential pair is removed from the Selections column, but is available in your design.

3. Specify a new name for the differential pair in the Diff Pair Name text box.
4. Click the Modify button. The new differential pair name is assigned to the selected nets.
5. Click the Close button to close the Create Differential Pair dialog box.

To delete a differential pair

1. In the Create Differential Pair dialog box_ , select the Differential Pair option from the drop-down list. All the differential pairs you created in your design appear in Diff Pairs column.
2. Select the differential pair you want to delete from the Diff Pairs grid and click the (>) button. The selected differential pair along with the associated nets appear in the Selections column.

⚠	If you accidentally selected the wrong differential pair for deletion and want to revert back, double-click the nets in the Selections column. The differential pair is removed from the Selections column, but is available in your design.

3. Click the Delete button. The differential pair set on the selected nets is deleted.

⚠	When you click the Delete button, the differential pair is deleted from the Selections column and the Diff Pairs column.

4. Click the Close button to close the Create Differential Pair dialog box.

Creating differential pairs between multiple pairs of flat nets simultaneously

Instead of creating differential pairs between two nets individually, you can quickly create differential pair between multiple pairs of flat nets simultaneously.

To create multiple differential pairs simultaneously

1.	In the Create Differential Pair dialog box, click the Auto Setup button. The Differential Pair Automatic Setup dialog box appears displaying all the flats nets and the corresponding differential pairs in the All Nets column.

✅	To view nets of a particular type, specify the initial letters of the net in the Filter text box. All the nets of that particular type will appear in the All Nets column. For example, if you want to view all nets starting with the letter "A", then enter "A" in the Filter text box. All the nets starting with letter "A" will appear in the All Nets column.

2.	Specify a string (numeric or alphabet) that you want to precede the differential pair name. For example, if you specify "A" in the Prefix text box, then all the differential pair names that will be created will be preceded with "A".
3.	Specify the last digit of the first net name in the + Filter text box. For example, all net names ending with 1.
4.	Specify the last digit of the second net name in the - Filter text box. For example, all net names ending with 4.
5.	Click anywhere inside the Differential Pair Automatic Setup dialog box

Capture displays a list of all differential pairs that can be created between all the nets that qualify the criteria set in the + Filter and - Filter text boxes. Also, the Differential Pair Name is preceded with the prefix specified in the Prefix text box.
The +Net and -Net grid displays the two nets associated with a differential pair.

⚠	If you do not want a specific differential pair to be created, select the row containing the differential pair and click the Remove button or double-click the row containing the differential pair. The selected row disappears.

⚠	If the nets forming a differential pair are of the type DP+ and DP-, the name of the differential pair is set to DP. For other pairs of nets, the name of the differential pair is of the type DP

6.	Click the Create button. All the differential pairs displayed in the Selections column are created. For information on how to view the differential pairs, see "To view a differential pair".
7.	Click the Close button to close the Differential Pair Automatic Setup dialog box and go back to the Create Differential Pair dialog box.

The Voltage Property

Besides the high-speed signal properties, another commonly used property assigned on a design net is the Voltage property.

When assigned on a part as an instance-level property, is not transferred to the PCB Editor netlist. Only when the Voltage property is applied to a flat net, it is passed to PCB Editor netlist as an electrical constraint.

The Voltage value assigned should be a numeric value. For example if you specify the value as 10mV, the non-numeric part is ignored. By default the property value is in volts. Therefore, say you need to specify the value of the Voltage property as 3mV, the value you need to enter is 0.003 and not 3mV.

Properties on Power Pins

While preparing your design for physical layout, it is important that the power pins in your design are handled properly, because depending on the part type, power pins may be shared across a package.

This section details properties related to POWER PINS that can be used in the Capture – PCB Editor design flow to ensure that the design is successfully netlisted.

POWER_PINS property

Power pins are usually defined when you generate a part using Part Editor. For example, during part generation you can define the pins 1, 3, and 7 as VCC pins; and 2, 4, 5, and 6 as GND. However, after you have instantiated a part in the schematic, if required, you can use override the part-level assignments by using the POWER_PINS property.

To use the POWER_PINS property

1.	Launch Property Editor on the selected part.
2.	Click New Row.
3.	In the Add New Row dialog, enter the property name as *POWER_PINS*.
4.	In the Value text box, type the power pins assignments. The syntax to be followed is:

<pin_type>:<comma seperated list of pinnumbers>; <pin_type>:<list of pinnumbers>

Example1:

To assign the pins 4, 5, and 7 to the VCC net, the value assigned to the POWER_PINS property should be (VCC:4,5,7)

Example2:

To assign the pins 4, 5 and 7 to the VCC net, and the pins 16, 18, 21 to the GND net, the power pins assignment is defined as (VCC:4,5,7;GND:16,18,21)

	Capture does not support duplicate name properties. So, in Example 2, you must define the VCC power pins assignment and the GND power pins in the same POWER_PINS property. To do this, use the semi-colon to differentiate between the assignments.

The power pin assignments that you perform through the POWER_PINS property are defined at the schematic level, and the changes are not reflected on the part. As a result, on opening the selected part in the Part Editor, the modified pin assignments are not reflected. However, in the Capture - PCB Editor flow, the POWER_PINS property on the part takes precedence over the part instance pin assignments.

Assigning Power Nets to Invisible Power Pins

It is a common practice to create parts with invisible power pins. When these parts are instantiated in schematic, Capture connects invisible power pins to a default power net that shares a name with the pin. For example, if pin number 14 and 28 are marked as VCC pins, these are by default connected to the global VCC net in the design.

Capture provides you a mechanism to override this default behavior and assign different values to different power pins. You can achieve this using one of the following methods.

- Using Assign Power Pins command
- Using the POWER_GROUP property

Using Assign Power Pins command

The Assign Power Pins command can be used on a component or on a design to view the invisible power pins of a single component or of all components in the designs.

1.	Select the component instance or the root design.
2.	From the Tools menu, choose *Assign Power Pins*.

The Assign Power Pins dialog displays.

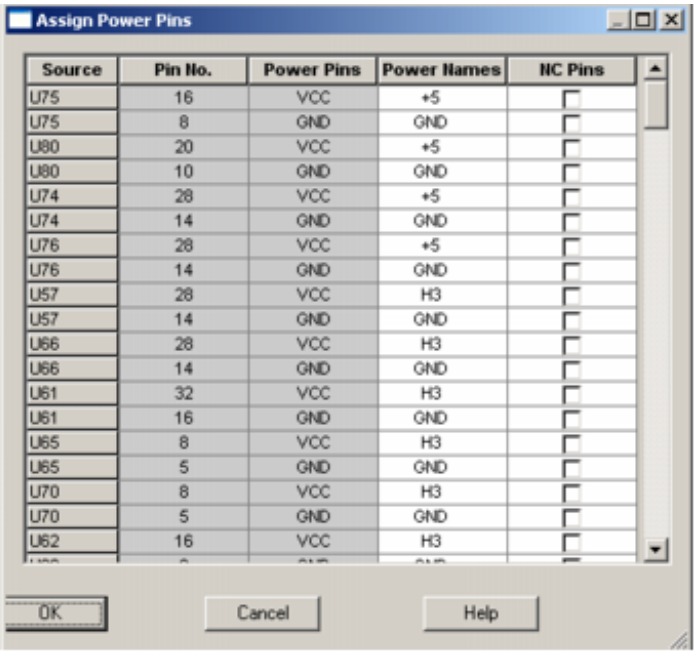
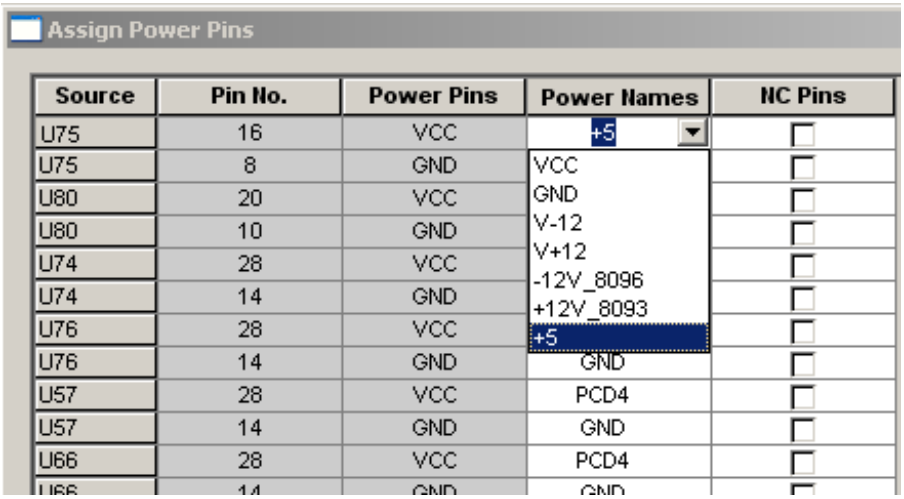


Figure 1-1 Assign Power Pin dialog for a design

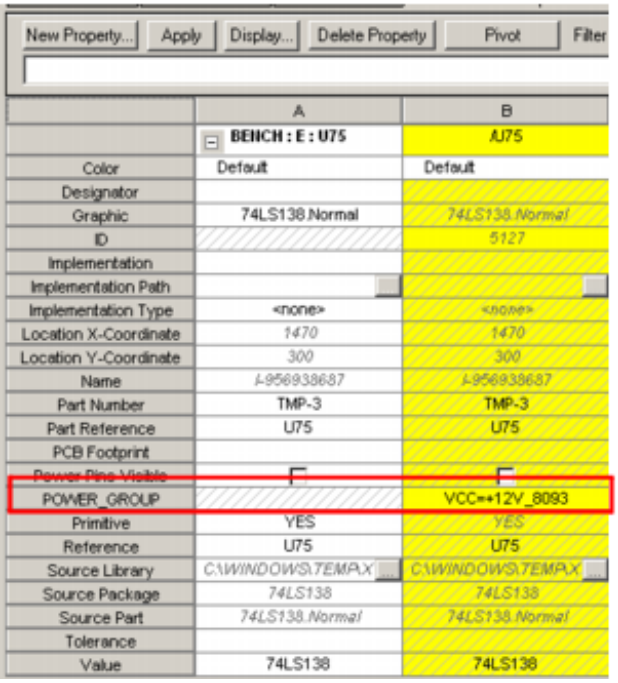
The Power Names column lists the power net connected to the power pin by default.

3.	To connect the power pin to a different power net, click on the Power names grid.
4.	From the drop-down list, select the power net to be associated with the power pin.



5.	Make changes for all the required power pins and click OK.

If you now save the design and launch Property Editor, the new value is assigned to the POWER_GROUP property on the component.



Using the POWER_GROUP property

This section discusses how to use the POWER_GROUP property to change the default power net associated with an invisible power pin of a component. This is a component definition property that circumvents the need to edit the pin properties for each invisible power pin on a particular component. Using this property, you can assign different values for the power net used on a particular design, instance or occurrence.
POWER_GROUP can be assigned with a unique value for multiple occurrences in the design. Multiple occurrences of a part can have different values for the POWER_GROUP property, allowing you to control power net connections at the occurrence level.
With the POWER_GROUP property added to parts with invisible power pin(s), you can overwrite power pins with the new power pin name at the instance level. If this property is used, POWER_GROUP is added to your combined property string so that you can annotate correctly.
The POWER_PINS property determines the power net in your design, to which invisible pins are connected. In order to connect multiple (invisible) power pins to the same net, you assign the POWER_GROUP property to each component that includes these pins.

To avoid using visible power pins in the Capture-PCB Editor flow

1.	Create a part with invisible power pins.
2.	Place the part in your design. By default, Capture connects any power pins on the part to the corresponding global nets. So, for example, if the part included (invisible) power pins VCC, VDD, GND, and AGND, each of these would be connected to a corresponding global net of the same name.
3.	In order to change the default power connections, select the part and add the

POWER_GROUP property to reassign the connections for any of the power pins. So, for example, to change the connection for VDD to VCC, you would add the POWER_GROUP property as follows:

```
POWER_GROUP VDD=VCC
```

(To change the value of this property, open the Property spreadsheet, then go to the Capture-PCB Editor Property filter and enter "VCC=VDD" as the value of the POWER_GROUP property).
In this example, pins in the netlist that would have been assigned to the VDD net are instead assigned to VCC and the netlister correctly reflects this change into the netlist.

The POWER_GROUP property is handled at the individual net level. So, if you have a part that includes power pins VCC, VDD, VPP, AVCC, GND, AGND, you can reassign one, some, or all of these depending on your requirements. For example:

```
POWER_GROUP : VCC=AVPP; GND=HGND
POWER_GROUP : AGND=GND
```

Reassigning power pin connections

When reassigning nets, Capture uses precedence rules for the POWER_GROUPS property in the same way they are used for other component definition properties. Therefore, when a higher level property value overrides property values at a lower level, even if there are multiple occurrences at the lower level, each of these occurrences is replaced by the higher-level properties.

For example, assume you have an instance with various power signals VCC, VDD, VPP, and VSS. If you want all these signals to be shorted to VCC, then assign the following POWER_GROUP property in the Attribute form of the Property Spreadsheet:

POWER_GROUP = VDD=VCC;VPP=VCC;VSS=VCC

This results in the following assignments:

On the part	On the schematic instance	In the PSTCHIPS.DAT file
Pin numbers 14,28 = VCC Pin numbers 13,12=VDD Pin numbers 10,11	POWER_GROUP= VDD=VCC; VPP=VCC; VSS=VCC	POWER_PINS='(VCC:14,28,13,12,10,11,8,9)'; POWER_PINS='(GND:7,21)'

=VPP Pin numbers 8,9=VSS Pin numbers 7,21 =GND		

The POWER_PINS property is an PCB Editor property that defines the various power nets for layout purposes. The value of POWER_PINS is derived from the value of the POWER_GROUP property when you create an PCB Editor netlist for your design

Unconnected Pins in Capture-PCB Editor Flow

While designing a schematic, it is a good design practice to ensure that there are no unconnected pins in the schematic. To ensure that an unconnected pin in a schematic page, is by design and not an oversight, no connect (NC) symbols must be used. You should always connect a no connect symbol to open pins for better readability of the design. The Design Rules Check tool ignores unconnected pins with no connect symbols.

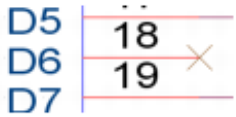
If a pin with a no connect symbol is connected to a net, the no connect symbol has no effect on the pin and becomes invisible. If the pin is later disconnected from the net, the no connect symbol becomes visible again.

Assigning "no connect" pins

To successfully create the physical layout of a schematic, in PCB Editor, you need to ensure that for all components, the number of pins in the schematic must equal the number of pins specified in the PCB Editor footprint.
When you generate the PCB Editor netlist for a capture schematic, the information about the number of pins in a component is included in the PSTCHIP.DAT file. The total number of component pins includes regular pins, power pins, and NC (no-connect) pins if present.
If you have numbered through-hole pins or non-electrical pins on the board then you must do one of the following to the part in your design:

- Add a NC property to the part. For the value of the NC property, use the pin numbers of the non-electrical pins separated by commas. For example, if you had an 8-pin footprint with the two through-holes being pins 7 and 8, then you would have a 6-pin part on your design with an NC property containing the value of 7,8.
- Place a No Connect symbol on pins that you do not want to be connected to anything. To do this, launch Property Editor to display the component properties. In the Pins tab, select the Is No Connect check box for the pin to be marked as NC pin.

	V
	BENCH : A : U4 : D6
Is No Connect	☑
Name	D6
Net Name	
Number	18
Order	23
Swap Id	-1
Type	3 State



In the design, you will see an X symbol on the pin. So, for the previous example you would have an 8-pin part in your design with No Connect on pins 7 and 8.

An error message is generated if there are missing pins on a symbol. After adding a No Connect property on a part, use the Update Cache command to update the part in your design (if you realize that the part is missing pin numbers).

- Do not connect any nets to the non-electrical pins on the part. So, in the two options above, you would have an 8-pin part with nothing connected on pins 7 and 8.

Do not do a combination of these three options on the same part or the netlister will issue a fatal error.

All pins that are not connected to a net (whether with a No Connect symbol, or otherwise) appear in the PSTXNET.DAT file, as nets with the name "NC." Therefore, you should avoid the net alias "NC" in your design. The NC property discussed in the first option, above, appears in the PSTCHIP.DAT file in the NC_PINS line rather than being added to the NC net in PSTXNET.DAT/PSTXPRT.DAT. Therefore, all pins connected to the NC net are unconnected on the PC board.

You have to account for unconnected pins of multi-section parts, such as mounting holes of multi-row connectors. To do so, however, you do not want to make the part heterogeneous with the mounting holes as pins on one section or distributed among the sections. Instead, make the part homogeneous and add a NC property to each section of the part, with the same pins listed for the NC property on all sections. In the part editor, you can add an NC property to each part in turn by choosing Previous Part from the View menu and placing the same NC property on all sections.

⚠	During netlisting, multi-section, heterogeneous parts are treated as single-section parts.

Assigning "no connect" power pins

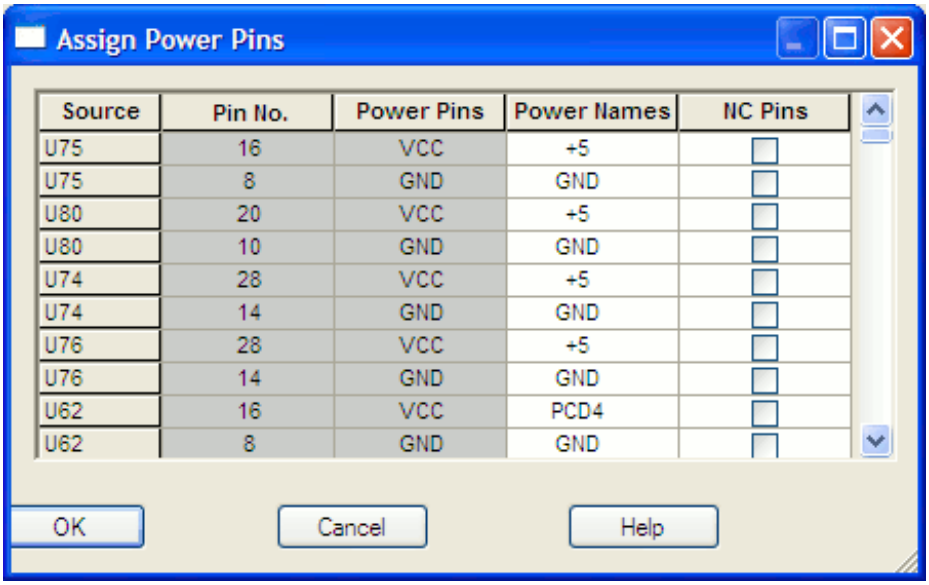
When you take your design to the board, by default, all powers pins will be shorted together (all VCC pins are shorted and all GND pins are shorted). This means that on your board you will need to route all the power pins. In the case of large pin devices (like FPGAs), this can be a tedious and time-consuming task.

To overcome this, you can select the power pins on your design that you want to route and then set all the other power pins as NC pins. This ensures that you then only need to route the power pins that are not set as NC pins.

Since a design may potentially have a large number of schematic folder and pages, Capture allows you to specify the NC pin command for power pins at multiple levels of the design. At the design level, you can view the list of the power pins in the entire design. You can also select and view the power pins at schematic folder level, schematic page level or even down to the level of one or more selected objects on a schematic page.

To specify a power pin as an NC pin

1. In the Project Manager, select the design, schematic folder or schematic page that contains the power pins you need to set as NC.
   Or
   To specify a power pin as an NC pin at the object level in a schematic page, select an object on the schematic page.
2. From the *Tools* menu choose *Assign Power Pins* to view the list of invisible power pins. The Assign Power Pins dialog displays listing all the invisible power pins in the object (or objects) selected in the Project manager.



If the selected object contains no invisible power pins, a warning is displayed in the Session window.

3. To specify a power pin as a no connect pin, select the corresponding NC Pins check box. On selecting the NC Pins check box, the corresponding Power Names field is disabled. The field lists the default Power net connected to the power pin.

4. Click OK to save your changes and to close the dialog box.

You can also open the Assign Power Pins dialog for objects selected in the Project manager by selecting *Edit - Browse - Power Pins*. To modify a power pin properties, select a line item in the power pin list and from the Edit menu, choose *Properties*.

Property Flow from Capture to PCB Editor

When you netlist a Capture schematic, not all properties defined in Capture are transferred to PCB Editor. For a property to flow from Capture to PCB Editor it needs to be included in the configuration (.cfg) file.

The configuration file specifies net, part (function), and component instance and component definition properties. This mapping determines what properties may be netlisted from Capture to PCB Editor or back annotated from PCB Editor to Capture. If a Capture property is not included in the configuration file it is not passed to PCB Editor. Similarly, if an PCB Editor property is not listed in the file, it does not get back annotated to Capture.

PCB Editor configuration file

The default configuration file installed with Capture is available at *<install_hier>*/tools/capture/allegro.cfg. This file has a predefined set of default properties for components, functions, nets, and pins, listed under different sections of the configuration file. By default, allegro.cfg is divided into five sections, written in a Windows.INI format.

- **ComponentDefinitionProps** - PCB Editor component definition properties, output in PSTCHIP.DAT file
- **ComponentInstanceProps** - PCB Editor component instance properties, output in PSTXPRT.DAT file
- **netprops** - PCB Editor net properties and all pin level properties, output in the PSTXNET.DAT file
- **functionprops** - PCB Editor function properties, output in the PSTXPRT.DAT file
- **pinprops** - pin-level properties transferred between Capture and PCB Editor

Following figure shows a section of the default configuration file included in the Capture installation.

Figure 2-2 Section of allegro.cfg

```
[ComponentDefinitionProps]
ALT_SYMBOLS=YES
CLASS=YES
PART_NUMBER=YES
TOL=YES
VALUE=YES
POWER_GROUP=YES

[ComponentInstanceProps]
GROUP=YES
ROOM=YES
VOLTAGE=YES

[netprops]
ASSIGN_TOPOLOGY=YES
BUS_NAME=YES
CLOCK_NET=YES
DIFFERENTIAL_PAIR=YES
DIFFP_2ND_LENGTH=YES
DIFFP_LENGTH_TOL=YES
ECL=YES
ELECTRICAL_CONSTRAINT_SET=YES
...
...
[functionprops]
GROUP=YES
HARD_LOCATION=YES
NO_SWAP_GATE=YES
NO_SWAP_GATE_EXT=YES
NO_SWAP_PIN=YES
ROOM=YES

[pinprops]
NO_DRC=YES
NO_PIN_ESCAPE=YES
NO_SHAPE_CONNECT=YES
NO_SWAP_PIN=YES
```

How properties are netlisted from Capture to PCB Editor

Not all properties in the configuration file show up as properties in PCB Editor. Some of these properties are used in generating portions of the netlist PST*.DAT files.

In PCB Editor, component properties (package properties in Capture) take precedence over function properties (part properties in Capture). So in the netlist, a package property value is used if both a part and package have values for the same property. Capture always uses the occurrence values in the netlist.

Package (component) properties, which are found in the PSTCHIP.DAT and PSTXPRT.DAT netlist files, can be viewed in PCB Editor using the Show Element command on a component. For example, PCB Editor has defined VALUE as a Component Definition property so it appears under this heading in the Show Element dialog box. Other properties such as CLASS or JEDEC_TYPE are also listed.

Part (function) properties are found in the PSTXNET.DAT file and can appear as Component Definition properties if they are predefined in PCB Editor and if you list them in the [ComponentDefinitionProps] section of the configuration file. Function properties are listed in the [functionprops] section of the configuration file.

Net properties appear in the PSTXNET.DAT/PSTXPRT.DAT file under the NET_NAME section.

For a design, you can have multiple configuration files. However, the configuration file to be used while generating PCB Editor netlist needs to be specified in the Setup dialog box launched from the PCB Editor tab of the Create Netlist dialog box.

User-Defined Properties in Capture - PCB Editor Flow

If your Capture design has user-defined properties, that you want should be transferred to PCB Editor, perform the following steps.

1. In Capture, add the property on the required design object.
2. Modify the configuration file to include the name of the custom property in the appropriate section of the configuration file.
   Following snapshot shows a user-defined property added on the ComponentInstanceProps section.

```
[ComponentInstanceProps]
GROUP=YES
ROOM=YES
VOLTAGE=YES
CDS_FSP_LIB_PART_MODEL=YES
CDS_FSP_IS_FPGA=YES
CDS_FSP_INSTANCE_NAME=YES
CDS_FSP_INSTANCE_ID=YES
SIGNAL_MODEL=YES
NO_XNET_CONNECTION=YES
MYPROP=YES
```

3. While specifying the options for generating the PCB Editor netlist, in the Create Netlist dialog box, ensure the following.

a. The file modified in step 2 is specified as the configuration file.
b. The Allow User Defined Property option is selected.

If you now generate the PCB Editor netlist, the custom property is pushed to PCB Editor. The property is visible in PCB Editor. Any changes made to the property value in PCB Editor will be updated in the Capture schematic, when you run the back-to-front flow.

Guidelines for Updating Configuration File (allegro.cfg)

Following should be kept in mind while modifying a configuration file:

1. User-defined property names are case sensitive. PCB Editor properties consist of all capitalized letters. If you have problems seeing properties netlisted or back annotated, check the spelling and the case of the property names.
2. The component definition properties VALUE, ALT _SYMBOL, JEDEC_TYPE, and CLASS are not back annotated since they cannot be changed in PCB Editor.
3. Do not use NO after the = sign for a property. NO becomes the property alias. If you do not want a property to be passed, you must delete it from the configuration file.
   In Figure 2-2, if a value after the = sign were to be anything other than YES, it would specify an alias under which the property will be output instead of its real name.
4. Some properties listed in the configuration file are not applicable in all versions of PCB Editor. Including these properties in the configuration file is not a problem since they would not be used.
   A list of typical properties used with PCB Editor may be found in the Capture-PCB Editor filter of the property editor.
   This filter is built on the PREFPROP.TXT file, which is copied to your Capture directory during installation.

Aliasing properties in configuration file

While migrating an old design to the latest version of the product, you may need to use aliases in the configuration file to map an old PCB Editor property to a new one. Table 2-2 lists some PCB Editor properties where aliasing would be required.

Table 2-2 Examples of Property Aliasing	
Old property name	**New property name**
ASSIGN_TOPOLOGY	ELECTRICAL_CONSTRAINT_SET
DELAY_RULE	PROPAGATION_DELAY
MATCHED DELAY	RELATIVE PROPAGATION DELAY

–	–	–
TOPOLOGY_TEMPLATE	ELECTRICAL_CONSTRAINT_SET	

The TOPOLOGY_TEMPLATE_REVISION property is obsolete and therefore ignored. If you have a legacy PCB Editor design that uses DELAY_RULE, for example, you can change the line in the configuration file from:

PROPAGATION_DELAY=YES
to
DELAY_RULE = PROPAGATION_DELAY

With an alias, you can have two different names for equivalent properties, one for Capture and one for PCB Editor. In the above example, DELAY_RULE properties get passed into the netlist as PROPAGATION_DELAY properties. You would include one of the above lines or the other in your configuration file. Alternatively, you could modify your design to use PROPAGATION_DELAY rather than DELAY_RULE.

Generating PCB Editor Netlist

The steps to forward annotate information from Capture to PCB Editor are as follows.

1. In the Capture project manager, select the design for which PCB Editor netlist is to be created.
2. From the *Tools* menu, choose the *Create Netlist* command.
3. In the Create Netlist dialog box, select the PCB Editor tab.
4. Ensure that the Create PCB Editor Netlist check box is selected.
5. Select the Setup button.
    a. In the Setup dialog box, ensure that the correct configuration file is specified in the Configuration File text box.
    By default, <*install_hier*>*/tools/capture/allegro.cfg* file is used. However, if you want to use a custom configuration file, specify its the path in the text box.

    b. To ignore the electrical constraints during netlisting, select the Ignore Electrical constraints check box.
    Following constraints are ignored.
        - PROPAGATION_DELAY
        - RATSNEST_SCHEDULE
        - RELATIVE_PROPAGATION_DELAY
        - DIFFERENTIAL_PAIR
        - NET_SPACING_TYPE
        - NET_PHYSICAL_TYPE
        - ELECTRICAL_CONSTRAINT_SET

- RATSNEST_SCHEDULE
- VOLTAGE
- MIN_LINE_WIDTH
- MIN_NECK_WIDTH
- MATCHED_DELAY

c. To suppress the netlisting warnings during the netlisting process, enter the warning (for example ALG0051) you want to suppress in the Suppress Warnings text box and click Add to add the warning to list of warnings to be suppressed.
d. Specify the value of other fields as required as click OK.

6. In the Netlist Files directory, enter the folder name where the generated netlist is to be saved.
7. Select the View Output check box to open the generated netlist files in Capture GUI.
8. Click OK.

The warnings and errors generated during the netlist process are listed in the session log.

✅	An error will be generated during netlisting if a part has an invisible power pin and the device property for the part has Power Pins Visible checked.

As shown in the following figure, Capture netlister generates the three PCB Editor-compatible netlist files.

The netlist files generate are:

- PSTCHIP.DAT This file contains a description for each different type of part used in the design. The netlister extracts this information from properties on occurrences.
- PSTXNET.DAT This connectivity file, also referred to as the flat list or expanded net list, contains each net, its properties, its attached nodes, and node properties. The list is ordered by physical net name.
- PSTXPRT.DAT This file, also referred to as the expanded parts list, contains a list of physical parts and lists each reference designator and the sections assigned to it, ordered by reference designator and section number.

⚠	For sample files and details on the file formats, click on the file links above.

Generating Initial Board File

While netlisting a Capture schematic, if required, you can also generate initial board file by selecting the Netrev option in the PCB Editor tab of the Create Netlist dialog box.

In order to generate PCB Editor board file, perform the steps listed in the Generating PCB Editor Netlist section to launch the Create Netlist dialog box and specify netlisting options.

1. Select the *Create or Update PCB Editor Board (Netrev)* check box.
   Different netrev options are enabled.
2. In the Input Board File text box, enter a name for an existing layout file to be used as template for generating the initial board file for the current design.

   > This field is optional if you are creating a new board.

3. In the Output Board File box, enter a name for the output file to be generated.

   > If you want to update an existing board, this field should have the same value as specified in the Input Board File text box.

4. Select the other options as required.
5. To open the output file immediately after the design is netlisted, from the Board Launching Option, select an appropriate layout application.

   > The extension of the input and output board files specified in step 2 and step 3, respectively, is related to the layout application selected.
   >
   > The supported options are:
   >
   > - Open Board in Allegro PCB Editor - generates the physical layout file with .brd extension
   > - Open Board in APD (Allegro Package Designer) - generates the physical layout file with .mcm extension
   > - Open Board in Cadence SiP - generates the physical layout file with .sip extension
   > - Open Board in OrCAD PCB Editor- generates the physical layout file with .brd extension
   > - Do not open board file - generates the board file in the path specified in step 3, but does no application is launched

6. Click OK to close the Create Netlist dialog box and create the .brd file.

On successful netlisting, blank board file is opened in PCB Editor. You can now place the parts and route your ratsnest.

⚠	Yellow triangles in the ratsnest indicate unrouted, zero-length connections (connections that lead directly from a pad on the top layer to a pad on the bottom layer). These connections need to be routed using a via.

Cross Probing for PCB Editor

After creating the board file, you place and route the board. This includes placing the parts in PCB Editor/Allegro SI/PCB Editor, APD and routing the nets. Sometimes, you may al
so require to swap pins or sections/functions to make routing easier. You can select the components from the Select elements for placement list in the Placement dialog box and then place them directly on the board. You can also place the components directly from the Capture schematic design. This feature is called cross probing. Between Capture and PCB Editor, there are two cross probing functions: cross highlighting and cross selection.

⚠	The cross probe function works only in the Interactive Place mode of PCB Editor.

Cross probing between PCB Editor and Capture uses Intertool Communication (ITC). To enable ITC in Capture:

1. Choose Option - Preferences.
2. In the Preferences dialog box, select the Miscellaneous tab.
3. Select the Enable Intertool Communication check box.
4. Click OK.

When you export the netlist for your design into PCB Editor, cross-probing will be enabled.

⛔	Both the backslash (\ ) and underscore (_) characters in net names interfere with cross probing. Also, the design name must not contain period (.).

Cross selecting between Capture and Allegro PCB Editor

If you are placing parts in PCB Editor using *Place - Manually* command, then select one or more parts in Capture and the corresponding parts will be selected in the Placement dialog box in PCB Editor. This option is only available when PCB Editor is active (running) and Intertool Communication (ITC) is enabled in Capture.

- The cross probe function works only in the Interactive Place mode of PCB Editor.
- If you select a part/pin/signal in PCB Editor that has been deleted from Capture design, a warning message will be printed in the Session log of Capture.

Cross highlighting between Capture and PCB Editor

Cross highlighting applies to three different types of objects: parts, nets and pins. Here are the general rules of cross probing:

- If PCB Editor is in highlight mode, you can select an object in PCB Editor, and the corresponding logical element in Capture is highlighted.
- If PCB Editor is in dehighlight mode, when you dehighlight a physical object, the corresponding logical element is dehighlighted in Capture. Deselecting an element in Capture dehighlights the corresponding element in PCB Editor.
- In Capture, when you select a component, its corresponding physical part is only highlighted in PCB Editor if you are in PCB Editor highlight mode. Otherwise, selection in Capture has no effect in PCB Editor, unless you are using cross selection.

The following tables show how highlighting and dehighlighting work between the two tools.

Selecting in Capture	Result in PCB Editor
Select a part	Highlights the corresponding component
Select a wire	Highlights all trace segments in the net
Select a pin	Highlights the corresponding pa

Deselecting in Capture	Result in PCB Editor
Deselect a part	Dehighlights the corresponding component

Deselect a wire	Dehighlights all trace segments in the net
Deselect a pin	Dehighlights the corresponding pa

Highlighting in PCB Editor	Result in Capture
Highlight a component	Highlights all parts in the corresponding package
Highlight a net	Highlights the entire corresponding flat net
Highlight a pad	Highlights the corresponding pi

Dehighlighting in PCB Editor	Result in Capture
Dehighlight a component	Dehighlights all parts in the corresponding package
Dehighlight a net	Dehighlights the entire corresponding flat net
Dehighlight a pad	Highlights the corresponding pi

Locking Components during Cross-Probing

When you cross probe between Capture and PCB Editor, you need to keep selecting components in your design to place them on your board.

In many cases, you create elaborate design with a large number of components and intricate connectivity. So when you keep selecting the components and nets on your design, you might inadvertently shift a component. This shift, in some cases, might even cause issues of connectivity.

To avoid shifting a component during the cross-probe operation, you can temporarily lock the component. This ensures that the component is locked to the canvas and cannot be moved.

However, this is a temporary locking operation and the lock status of the object is lost as soon as you click anywhere on the page or on another object. To create a persistent lock on a component you need to use the Graphical Operation (GOp) Locking feature in Capture.

Pin Swapping In Capture-PCB Editor Flow

PCB Editor allows function/section swapping depending on the logical pin list on the

function/section. This plays a key role when you create asymmetrical parts. If you have used invisible pins in a Capture design, these pins are not used by PCB Editor while deciding swappable sections. That is, body sections do not play any role in section swapping. PCB Editor calls each section as a function. A function is made with only logical pin list and any two functions that have same logical pin list are swappable in PCB Editor.

In the Capture-PCB Editor flow (only) you can perform pin swapping between parts in a heterogeneous package by using the SWAP_INFO or the SPLIT_INST property. This is useful when you are working with parts that have large pin counts (such as BGA parts). For information about the SWAP_INFO and the SPLIT_INST property, see Using the SWAP_INFO property and Using the SPLIT_INST property. You can assign the SWAP_INFO or the SPLIT_INST property to a part at the library level (recommended) or at the instance level.

When you assign the SPLIT_INST property to a part at the library or instance level, all lower level occurrences of that part (on the schematic) inherit the property. This is the only occasion in which a property at a higher level overrides properties at lower levels.

If you want to do pin swaps in PCB Editor and then back annotate those changes, you must set up the pin properties in Capture first. Pin swap specifications will be produced only if the Swap Id properties are set correctly on pin-swappable parts.

To do this, you can open the part in the library or select the part of interest in your design. Then, from the Edit menu, choose the Part option. From the View menu, choose Package, then from the Edit menu, choose Properties.

In the Properties spreadsheet set the PinGroup value to 1 for each swappable (input) pin of the part. If you have a multi-section part you only have to set the PinGroup value for one section; the part editor adds the same value for all the other sections automatically. For example, on a 7400, set the PinGroup to 1 for pins 1 and 2. Leave the other PinGroup values blank and they are filled in automatically when you click the update all button. (PinGroup = 1 for pins 1, 2, 4, 5, 8, 9, 11, and 12.)

When you select a pin and edit its properties, the value shows as the Swap Id property. The default value is -1, meaning the pin is not swappable. Therefore you must add the PinGroup property to enable pin swapping for your part if you want to be able to swap pins. Swap Id value of 0 and greater than 0 in the property editor mean that the pin is swappable. If you made a pin swappable, and later decide to remove the swappability of the part, then you just delete the PinGroup property in the property editor. You can see which pins are swappable in the PSTCHIP.DAT file by looking for the PIN_GROUP line under the pin name.

Using the SWAP_INFO property

To enable pin swaps across sections of heterogeneous split parts in PCB Editor, a SWAP_INFO property has been introduced in PCB Editor.

⚠	In Capture 9.2.3, the SPLIT property provided a method for swapping pins between parts of a heterogeneous package. For all later releases of the product, it is recommended that the SWAP_INFO property should be used, as it provides the same functionality with better features.

⛔	In existing designs, do not replace the SPLIT property with the SWAP_INFO property. The SPLIT property converts parts as flat parts whereas the SWAP_INFO property leaves the part as a split part. On an existing design, replacing the SPLIT property on a component package with the SWAP_INFO property will change its definition from a flat part to a split part. This will cause component rip-offs in the board file.

⛔	Design created in old Capture releases, might have the SPLIT property assigned on components to enable swapping pins between parts of a heterogeneous package. For such designs, it is recommended that the SPLIT property must not be replaced with the SWAP_INFO property. This is because, the SPLIT property converts parts as flat parts whereas the SWAP_INFO property leaves the part as a split part. On an existing design, replacing the SPLIT property on a component package with the SWAP_INFO property will change its definition from a flat part to a split part. This will cause component rip-offs in the board file.

The SWAP_INFO property defines a logical group of parts of a package in a heterogeneous (split) part. This allows two pins - having the same PIN_GROUP property - to be swapped within a logical group, regardless of the physical availability of the pin in the given part of the package.

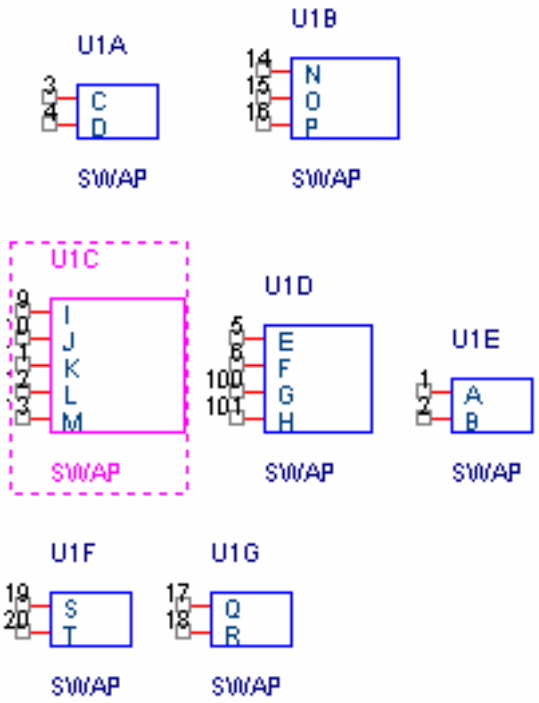
The SWAP_INFO property is defined under the [ComponentDefinitionProps] section in the allegro.cfg file. The swap information is written into body section of the primitive of the pstchip.dat file. PCB Editor reads the SWAP_INFO property and accordingly allows pin swaps across sections.

If you have a customized PCB Editor configuration file (allegro.cfg) for your Capture PCB Editor netlister, you need to add the following entry under the [ComponentDefinitionProps] section in the Allegro.cfg file:

```
[ComponentDefinitionProps]
SWAP_INFO=YES
```

Example

The usage of the SWAP_INFO property is described using a split part.



The above figure shows the example of a heterogeneous (split) part having seven parts-per-package. Let us assume that parts with the reference designators:

- U1A and U1B, form one logical section
- U1C, U1D and U1E, form the second logical section
- U1F and U1G, form the third logical section

In this scenario, the value of the SWAP_INFO property on all the parts of the package will be defined as:

```
(S1+S2),(S3+S4+S5),(S6+S7)
```

The logical section (S1+S2) indicates that a pin in section S1 is swappable with a pin in section S2, provided they have the same PIN_GROUP value. Therefore, pin 3 of S1 can be swapped with pin 14 of S2 if they have the same PIN_GROUP value.

In a SWAP_INFO syntax, the logical and physical sections are separated by a sign.

Note the following when you use the SWAP_INFO property:

- You must assign the SWAP_INFO property to all parts in the heterogeneous package for which you want to perform pin swapping.
- There must not be any duplicate pins on any of the parts in the heterogeneous package if they are to receive the SWAP_INFO property.
- Ensure that you name the sections as (S1+S2+ S3+…S*n*). If you use any other notation for naming the sections, the SWAP_INFO property will not work.
- Ensure that you use `(` and `)' brackets for defining single sections also. For example, If you have a split part having 6 sections (S1 to S6) and wish to add the SWAP_INFO property such that swapping happens only between sections S5 and S6. In that case you need to add the SWAP_INFO value as (S1), (S2), (S3), (S4), (S5+S6).
- The SWAP_INFO property does not function for homogeneous parts. No DRC errors will be reported if the SWAP_INFO property is assigned to a homogeneous part.
- DRC errors will not be reported if there are syntax or semantic errors in the value of the SWAP_INFO property.

Using the SPLIT_INST property

The SPLIT_INST property provides a method for swapping pins between parts of a heterogeneous package. You can assign it to a part in a library, or to an instance in a design.

	When you assign the SPLIT_INST property to a part at the library or instance level, all lower level occurrences of that part (on the schematic) inherit the property.

The SPLIT_INST property has two possible values: TRUE and FALSE (default). Assigning a value of TRUE to the SPLIT_INST property on the parts of a heterogeneous package indicates that the netlister treats the package as a flat part, thus allowing pin swapping between the parts of that

package.
When you use the SPLIT_INST property, be aware of the following points:

- You must assign the SPLIT_INST property to each part in the heterogeneous package for which you want to perform pin swapping.
- The SPLIT_INST property does not function for homogeneous parts (nor will the Design Rules Check tool detect an error if SPLIT_INST is assigned to a part that is part of a homogeneous package).
- There must not be any duplicate pins on any of the parts in the heterogeneous package if they are to receive the SPLIT_INST property.
- Cross-probing will not function on pins of parts that have the SPLIT_INST property assigned to them. You can work around this by cross-probing the connected nets rather than the pins.

SinceCapture 10.5 release, the SPLIT property has been renamed to SPLIT_INST.

Some of the other pin properties used in Capture for pin and gate swapping are:

NO_SWAP_COMP property

The NO_SWAP_COMP property defined on a component instance ensures that while swapping the component, the symbol associated with the component does not get swapped. This property takes a boolean value. Make sure that you set the value of this property to TRUE.

NO_SWAP_GATE property

The NO_SWAP_GATE property defined on a reference designator or a function designator (gate) specifies the functions within the component that cannot be swapped. The function remains fixed in its current slot in the component. This property takes a boolean value. Make sure that you set the value of this property to TRUE. For more information on this property, see the *Allegro PCB and Package User's Guide*.

NO_SWAP_GATE_EXT property

The NO_SWAP_GATE_EXT property defined on a function designator ensures that the function is not swapped with a function from another component. However, the function can be swapped among slots within its current component. This property takes a boolean value. Make sure that you set the value of this property to TRUE.

NO_SWAP_PIN property

The NO_SWAP_PIN property defined on a reference designator, function designator (gate), or a pin ensures that the pins on the component or function are not swapped either interactively or automatically. This property takes a boolean value. Make sure that you set the value of this property to TRUE.

When you add this property at the instance level in Capture, you need to add the following entry under the [ComponentInstanceProps] section in the Allegro.cfg file:

```
[ComponentInstanceProps]
NO_SWAP_PIN=YES
```

The above entry contains the properties that you add to the components in Capture.

- **Part name.** The part name found between single quotation marks in the PSTCHIP.DAT file is just the value of the DEVICE property present. If there is no a DEVICE property on the part, then the part name is a made by combining the values of the Source Package, PCB Footprint and other properties that may be found in the [ComponentDefinitionProps] section of the configuration file. The part name string is a concatenation of these properties, with each value separated by an underscore character. By changing the order of component definitions properties in Allegro.cfg you can change the concatenation order.
- **Pin level property transfer between Capture and PCB Editor.** You can also pass pin level properties between Capture and PCB Editor using a configuration file. This ensures seamless transfer of pin level constraints between Capture and PCB Editor. In the earlier versions of Capture, there was no way pin level properties could be transferred back and forth from Capture and PCB Editor and had to be manually specified at both ends. For example, now you can specify a property say, NO_SHAPE_CONNECT to a schematic pin. PCB Editor, on finding this property, will ensure that no connection is created between the pin (that passes through a shape with the same net) and a shape.
- **Packaging of multi-section parts.** During netlisting, multi-section, heterogeneous parts are treated as single-section parts. For multi-section parts, all sections must have the same values for the properties listed under [ComponentInstanceProps] in Pin Swapping In Capture-PCB Editor Flow. For example, ROOM is a component instance property, so if you add {ROOM} to the combined property string when you annotate, then sections with differing ROOM properties will not be packaged together.

  You can separate or combine component instances in a multi-section parts just by specifying distinguishing properties in the combined property string. Check the configuration file to identify the component instance properties currently available.
- If you are planning to back-annotate your design from Capture, do not modify the schematic in Capture while working on the design in PCB Editor.

Back Annotation from PCB Editor

The Back Annotate dialog box appears when you choose Back Annotate from the Tools menu after selecting the design folder of a Capture project. The back annotation process generates a Capture-compatible swap file, which is based on the differences between the logical view

(PST*.DAT netlist files) and the physical view (*VIEW.DAT files of board changes).
You use back annotation to synchronize the design file with the changes done in the board file. Changes in the PCB Editor board need to be back annotated to the Capture schematic to ensure the physical board design is consistent with the logical schematic design.
The Back annotation process includes the following steps:

1. Generating feedback files (*VIEW.DAT files) - A utility called genfeedformat generates board file information in four files named compview.dat, pinview.dat, netview.dat, and funcview.dat. These four files are also called *VIEW.DAT files.
2. Generating PCB Editor netlist files (PST* files) - Capture- PCB Editor netlister generates netlist files (PST* files) again. This step is necessary to check if any changes are made in the Capture design after board file creation.
3. Generating the swap file (.swp file) - Capture-PCB Editor netlister runs in the Feedback mode and generates the swap files by comparing netlist files with feedback files.
4. Updating the design with swap information - Capture updates the design based on the information in the swap file.

While generating *VIEW.DAT files, the PCB Editor Export Logic utility (genfeedformat.exe) uses the pxlBA.txt file to decide which properties need to be written into the *VIEW.DAT files. The pxlBA.txt sets up the properties that are back annotated from the PCB Editor board file.
When you create an PCB Editor netlist (forward mode), the pxlBA.txt file is generated and is stored in the same location as the PST*.DAT files. When you back annotate a design (backward mode), the pxlBA.txt file is generated again and is stored in the same location as the .BRD file (board file).
If PCB Editor is not installed on the same system as Capture, you can use the Export Logic command of PCB Editor on the system where PCB Editor is installed. By default, PCB Editor picks the pxlBA.txt file from the location where the board file resides. If the pxlBA.txt file does not exist at the board file location, PCB Editor picks it from the standard PCB Editor installation path, which is *<install_dir>*/share/pcb/text/views. However, this pxlBA.txt file may not have all the properties that you want to back annotate to Capture and some of the properties may get annotated as deleted or with a null value. To avoid this problem, you must copy the pxlBA.txt file generated by Capture to the board file location, before running the Export Logic command from PCB Editor.
PCB Editor back annotation includes property changes, additions and deletions; changes to part reference designators; and gate (function) and pin swaps. Here are some details:

Table 2-3 Modifications in PCB Editor back-annotated to Capture

Pin swaps	Interchanges two pin numbers. For example, pin 6 could become pin 9. Pin 9 would become pin 6 in the process. On the board, the net is just routed to a different pin, since the order of the pins on the physical IC cannot be changed. On the schematic, the pin numbers will visually switch places.
Gate swaps	Switches or interchanges two gates, or functions. For example, a 74LS00 has four NAND gates: U1A, U1B, U1C, and U1D. You can swap U1A with U1B or any other

	of the NAND gates in the package.
Reference changes	You can change reference designators, U1 to a value of ST1, for example. If the part is a multi-package, then U1A through U1D, would become ST1A through ST1D.
Property changes	Properties defined or changed in PCB Editor are back annotated to Capture, provided the properties are listed in the configuration file. Just as occurrence values are always used in the PCB Editor netlist, these values are also the ones replaced or updated in the back annotation process. Instance values are neither netlisted nor back annotated unless the instance value is the same as the occurrence value. If you double-click on the part to invoke the part editor on the schematic page the occurrence values are the ones in the yellow rows below the instance values (white rows). Back annotation from PCB Editor only uses CHANGEREF and PINSWAP format lines in the .SWP file for pin and gate swaps and reference designator changes. The properties are back annotated in a separate section. If a net name is renamed in the physical design (on the PCB Editor board) and net properties are added or edited, the net name does not back annotate to Capture, even though the properties do. To get around this naming discrepancy between the physical layout and schematic designs, you should rename the net in Capture, then netlist the design to PCB Editor. The net names then correspond and properties may be passed without a problem.

Setup button	_Click this button to open Setup dialog box, where you can set up, edit and view information about the configuration file used for netlisting and back annotating property information between Capture and PCB Editor. You can also specify the number of backup files to keep in your design directory.
Generate Feedback Files	Select this option to generate the *VIEW.DAT back annotation files from the specified PCB Editor Board File. These files are listed under the project manager. Selecting this option is equivalent to using the Export Logic command in PCB Editor. This option is only available if you have PCB Editor installed. If this option is unselected, then make sure the * VIEW.DAT files are saved in the same directory as PST*.DAT netlist files for your design.
PCB Editor Board File	Accept the path and file listed or navigate to the PCB Editor board (.BRD) file that contains previously-imported netlist information and the design changes you want to back annotate. This is the same board file used to create feedback files (*VIEW.DAT files) need for generating the .SWP file during back annotation. By default, the name of your design (with a .BRD extension) in the allegro

	subfolder is used, unless you have run a previous back annotation. In this case, the field contains the file previously entered. If the file in this field is not valid, back annotation cannot proceed and Capture issues an error message. Back annotation from PCB Editor only uses CHANGEREF and PINSWAP format lines in the .SWP file for pin and gate swaps and reference designator changes. The properties are back annotated in a separate section.
Netlist Directory	Browse to the directory where you have your PST*.DAT files. This is also the location where the * VIEW.DAT files will be placed after being extracted from the board. The default directory is the allegro subfolder for your design. If you have run a previous back annotation on the current .DSN design, the netlist directory for that back annotation is the default. A netlist directory must be specified for back annotation to proceed. It is critical that the original design not be modified before attempting to back annotate. Otherwise, errors can result when comparing the netlist files with the *VIEW.DAT board files.
Output File	Specifies the path and file name for the .SWP file that is saved after back annotation. By default the file name is DESIGN_NAME.SWP unless you have previously run a back annotation on the current design. In this case the default output file is the name given to the previous output file.
Back Annotation	**Update Schematic.** Select this option if you want the Capture schematic design to be updated with back annotation information from the .SWP file. Selecting this check box lets you review the back annotation details. This option is selected by default. If you don't select this check box, you can still use the Layout tab later to back annotate the generated .SWP file to Capture. You might choose this option, for example, if you wanted to view the SWP file before actually back annotating. In this case, you can also select the check box later in the Allegro tab when you rerun the back annotation. **View Output (.SWP) File.** Select this option if you want the .SWP file to be automatically opened and available for viewing and editing in a Capture text window after the .SWP file is generated. You can also close the file and re-open it from the project manager. This check box is not selected by default

PCB Editor back annotation allows you to do the following:

- Perform more than one back annotation in a row without netlisting in-between, once you have made an initial netlisting.
- Netlist occurrence values for user-added properties.
- Back annotate parts with added connections or properties that are not used, including unwired parts and those that could be used in the future.
- Back annotate properties and their values to components, pins, and nets using a configuration file.
- Back annotate numeric and alphabetic reference designators for multi-section parts.
- Check the Capture session log for errors.

Best practices for smooth back annotation

- Do not change design name, hierarchical block names, or reference designators in Capture after board files creation.
- Do not edit a part from schematic in Capture after board file creation.
- Do not replace cache as it changes the Source library name and part name, in capture.
- Do not change the values of component definition properties in capture after board files creation.
- Capture does not support electrical constraint sets (ECSets). ECSets will not be back annotated to your Capture design.
- Do not change Design file/root schematic/hierarchical block names in Capture after board file creation.
- Do not add or delete components to or from the schematic design immediately after the board file creation. Add or delete components after finishing the back annotation process.
- Do not add any additional components in PCB Editor. Instead, add components in Capture and take them to PCB Editor.
- Do not add, rename, or delete a net in PCB Editor.
- Do not change the format for reference designators for parts in PCB Editor as <Alphabet(s)><Numeric><Alphabet(s)> or <Alphabet(s)>-<Alphabet(s)>.
- Run PCB Editor Dbdoctor before running Back annotation by selecting the Database Check command from the Tools menu in PCB Editor.
- Make backups of the original design before updating the design with the swap information in Capture.
- Back annotate the design immediately after making the board file. Though not a mandatory

step, back annotating the design before placing components helps avoid problems in back-annotation at a later stage.

- During back annotation, if you encounter Error *[ALG0037] Unable to read physical netlist data.* The probable reasons for this error are:
  - Netlist files not found.
    or
  - Unable to read the netlist file because either the path name is long or has spelling errors.

  If back annotation at this stage generates an empty swap file, you can proceed with placing and routing the board file. In case any problems are detected, you must correct them in the design file and generate the board file again until an empty swap file is generated.
- In PCB Editor, if you modify properties on a net, which does not have a corresponding physical object (also called invisible nets) in Capture, the modified properties will not be imported during back annotation. The error messages are displayed in the sessions log.

Physical Layout of a Simulation Design

This section is valid only if you have used PSpice to run simulations for functional verification of your design, and after verifying the schematic, you want to create the physical layout for same schematic.
For creating the physical layout of designs that have been simulated in PSpice, perform the following tasks.

- Assign appropriate footprints to components.
  For details, see the section on Assigning Footprint Property.
- Add Board only Components.
- Mark Simulation Only components.
  Simulation only components need to be marked so that they are ignored by the netlister during the physical netlist generation process.
- For this, add the PSPICE_ONLY on the simulation only components and set the property value to TRUE.

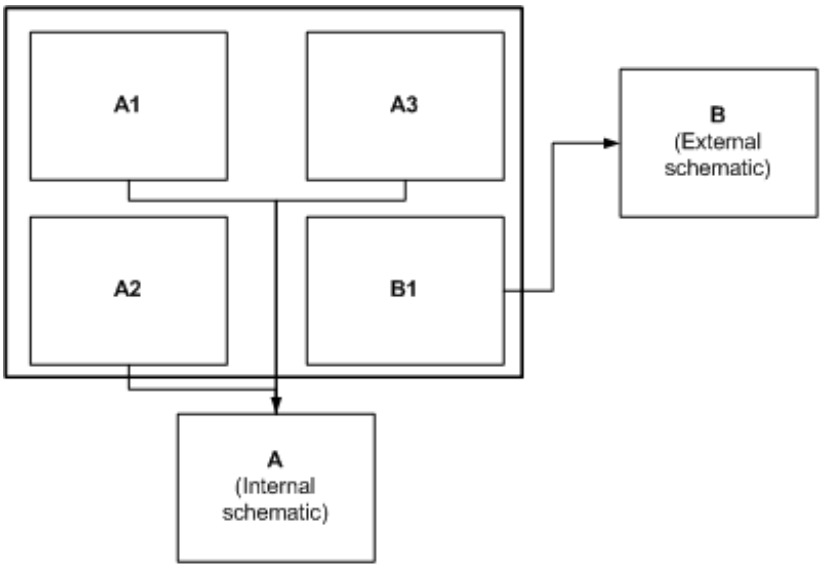
- The PSPICE_ONLY = TRUE property is added by default on all components from libraries such as, STIMULUS.OLB, and ANALOG.OLB.
- Voltage Sources in Cadence supplied Source library already have this property assigned.

- During circuit simulation, if you have added series parasitic elements to the schematic, to include effect of track resistance, track inductance, or capacitance, these need to be ignored while generating PCB Editor netlist.
  For these components, besides PSPICE_ONLY= TRUE, you also need to add PACK_SHORT property with its value equal to the logical pin numbers of the component. For example, PACK_SHORT = (1,2).

Design Reuse for PCB Editor

Design reuse is the capability of defining and using design modules, both in the Capture and PCB Editor layout environments, that are compatible between both of these PCB design tools. You can create modules for reuse in your design in either the Capture schematic or in PCB Editor for physical layout part of the design process.
Design reuse enables you to reuse packaging and annotation information from your schematic when you route your PCB (in the form of a PCB Editor .BRD file).
You can use the annotation information for particular schematic, whether internal to the design, or as an external reference, multiple times in your PCB Editor design, and all the part references and packaging information will be duplicated correctly with each occurrence of the reuse design in the .BRD file.

Consider this example:



Suppose a design consists of multiple occurrences of two schematics:

- Schematic A is an internal schematic (that is, a schematic that is part of the design, and appears in the project manager as a folder)
- Schematic B is an external design (a schematic that is referenced by the design but that is not part of the design)

Schematic A has three occurrences in the design: A1, A2, and A3. Schematic B has one occurrence, B1. If you specify each of these schematics as "reuse schematics," when Capture annotates the design, packaging information is assigned to parts in the design such that part packaging is contained within each occurrence of the reuse schematic.
So, in this example, when you annotate the design in Capture, no parts in A1 would share packaging with parts from A2 or A3, etcetera.
In PCB Editor, the source of a design reuse module is the Capture netlist. This netlist contains a set of Reuse properties that are used to identify and group each of the packages within the module. After placing and routing the board in PCB Editor, you can save the design as a reuse module by creating an MDD file

Multi-level reuse designs

When creating a multi-level schematic design in which you embed successive levels of reuse modules, you must take care to ensure you preserve the reuse design. In general, modifying properties from the root level of a design is not a problem unless the property causes a component change at a lower level.

For example, say you have a three-level design called HIGH.DSN. HIGH.DSN references MID.DSN which is a reuse module, and MID.DSN references LOW.DSN which is also a reuse module.

If you want to make design modifications that will result in changes in connectivity then you must edit lower-level designs first before you can see the changes reflected at higher levels. Occurrence properties however, may be changed at any level in a referenced schematic without modifying the source design. Consider the following two design scenarios. The first is a component change but the second only changes occurrence properties.

Scenario A--Modifying LOW.DSN by adding a series terminating resistor

1. First, open the LOW.DSN design and make the change. Afterward, netlist the design to PCB Editor and update the PCB Editor board and reuse module from within PCB Editor.
2. Open MID.DSN and, again, netlist the revised design with connectivity changes to PCB Editor. In PCB Editor, read in the revised LOW physical module and update the MID board and physical module.
3. Open HIGH.DSN and generate the netlist for PCB Editor. Once the design is in PCB Editor, refresh the instances of the MID reuse module.

Scenario B--Changing a Reference Designator in LOW.DSN from U1_1 to U444

In this case, all you have to do is modify the higher-level occurrence tree for the component that is located in LOW.DSN. The next time you take the netlist from HIGH.DSN into PCB Editor

the individual component will get its Reference Designator updated.

- Except for occurrence properties, the schematics of externally-referenced libraries and designs should not be edited. You should view them as read-only designs. Trying to edit and save these designs from within your schematic, can introduce errors such as duplicate reference designators and other problems.
- When saving schematics with externally-referenced libraries or designs, occurrence properties are saved but altered instance values are not. If you want to change externally-referenced libraries or designs you should first close the referencing design. Then, open the referenced library or design, make the necessary changes, and save and close the referenced library or design. At this point, you can reopen the original design and reference the modified design.

Creating a Reuse Design

To create a reuse design

1. In Capture, create the schematic that will become your reuse design.
   You can use the project wizard to set up your schematic and build the design from scratch or use library components.
2. Perform a design rules check (DRC) by selecting the .DSN file in the project manager, then choosing the Design Rules Check command from the Tools menu. This action checks for disconnected nets, no connects, off-grid objects, packaging problems, duplicate part references and other types of errors.
3. Annotate the design by choosing the *Annotate* command from the Tools menu. Select the Packaging tab of the Annotate dialog box and check appropriate options.
   If you do not plan to make any design changes in PCB Editor that would affect the netlist, skip to step 7. Otherwise, proceed to the next step.

   In step 7, there are two options you can check in the PCB Editor reuse tab, that are, Generate Reuse Modules and Renumber Design for Using Reuse Modules. By checking both of these options, you can generate a reuse design and annotate it at the same time, saving yourself extra steps.

4. Generate an PCB Editor netlist by choosing the *Create Netlist* command from the Tools menu, select the PCB Editor tab in the Create Netlist dialog box. Complete the appropriate options in the dialog box. Either enable the Create or Update PCB Editor Board (Netrev)

option to open a .BRD design in PCB Editor during netlisting or import the netlist into PCB Editor by choosing the Import Logic command in PCB Editor.

5. In PCB Editor, place and route the physical design, then export the design logic to Capture.
6. In Capture, back annotate design changes from PCB Editor by selecting the .DSN file and choosing the *Back Annotate* command from the Tools menu. The Backannotate dialog appears. Select the appropriate options in the PCB Editor tab of the Backannotate dialog box.
7. From the Tools menu, choose the *Annotate* command. In the PCB Editor reuse tab of the Annotate dialog enable the *Generate Reuse Module* option to create a reuse module from the design.
   This step adds a unique REUSE_ID property for each package. You can view these properties in the Property Editor window.
8. Netlist the design to PCB Editor, and create an .MDD reuse module in PCB Editor from the schematic reuse design.
   You complete the module creation in PCB Editor by selecting the Create Module command from the Tools menu. You are prompted to select the extents of the module and pick an origin. By doing so, you have designated an PCB Editor Module Definition File (.MDD).

	See the PCB Editor online Help documentation for how to create a physical design reuse module in PCB Editor from a placed-and-routed board.

9. Repeat steps 1 through 8, as necessary, for multiple levels of design in a design reuse hierarchy.
10. Use the reuse module in a Capture design, either as a library part or as a hierarchical block.

	• When creating design reuse modules, it is a good idea to avoid making multiple-page schematics (with off-page connectors). When trying to descend the hierarchy of a referenced design, such as a reuse module, you cannot choose which page gets opened**.**

Reuse properties

Capture assigns reuse properties to identify and distinguish the reuse parts used in simple hierarchical and complex hierarchical designs. These properties ensure that part packaging is preserved and references are renumbered in such a way that they do not conflict with each other.

Capture Assigned Reuse Properties	Description
REUSE_ID	This property is added to every part in a reuse design. Within a reuse design there are as many values of this property as there are packages so that each package has a unique REUSE_ID. All parts in a package have the same REUSE_ID value. Capture assigns these property values when you enable the Generate Reuse Module in the PCB Editor reuse tab of the Annotate dialog box.
REUSE_PID	If a reuse module contains another reuse module as part of its external design, then the netlister assigns a REUSE_PID value to every component in each package of the external design. The value of the REUSE_PID is the same as the value of the component's previous REUSE_ID. A new REUSE_ID value is then assigned to each module. This way, occurrences of the same module will have different REUSE_IDs on them, but the same REUSE_PIDs for corresponding components. Using REUSE_IDs, makes it possible for Capture to propagate changes to lower levels of a reuse module.
REUSE_NAME	The default value of the REUSE_NAME property, assigned by the netlister, is a concatenation of the design name and the schematic name coupled by an underscore character. Here is an example: testmodule_schematic1 The REUSE_NAME property is propagated down throughout the design hierarchy to all parts below.
REUSE_INSTANCE	This value is computed by the netlister and added to the netlist. The value of this property is unique for each usage, or instance, of a reuse module. A design may have one REUSE_NAME value but many REUSE_INSTANCE values. The REUSE_INSTANCE property is obtained from the name of the referencing hierarchical part. If a REUSE_INSTANCE property is not present, it is created as follows: < REUSE_NAME >_<document ID of the referencing hierarchical part> Like the REUSE_NAME property, the REUSE_INSTANCE property is propagated down throughout the design hierarchy to all parts below.
REUSE_MODULE	PCB Editor assigns this property as a unique name to identify a physical reuse module. The property corresponds to placed and routed board in PCB Editor (.BRD file) which has been saved as an .MDD file. If this property is user-defined in Capture, it specifies the reuse module to use in PCB Editor.

Using a Reuse Design

Once you have generated a reuse module in Capture, you can use it in one of two ways, either by:

- Placing the module as an external design schematic (.DSN file) using the Place Hierarchical Block dialog box;

  or

- Placing the module as a library part (.OLB file) using the Place part dialog box if the part was created with the Generate Part command in Capture.

You can use a design reuse module either as an .OLB part from a library or as an external .DSN design, placed as a hierarchical block.

Reuse Module as a Library Part

The steps to place a reuse design that was created as library part, in your design, are:

1. From the Place menu, choose the Part command. The Place Part dialog box appears.
2. Locate and select the .OLB part previously saved as a reuse design. For more information, see Searching for a part in the libraries.
3. Click OK. An image of the part is attached to the mouse pointer.
4. Move the part image and click the left mouse button to place the part.
5. For each instance property of the part you want to place, repeat step 4.
6. Press the ESC key or select another tool to dismiss the part attached to the mouse pointer.

Placement Guidelines

- When you place a part off-grid, it remains off-grid through any cut-and-paste and drag-and-drop operations.
- If you place parts so that two pins meet end to end, the pins are connected. OrCAD recommends that you connect the pins of the parts using a wire, and avoid placing parts so that two pins meet end to end. This is because, parts with direct pin to pin connections produce a system generated net name to establish the connection. Using system-generated net names is not recommended as:
  - Capture does not support overriding system generated net names with user-specified net names.
  - Searching for the system generated net name can be difficult if you are not aware of the pin to pin connection.
  - If you move the parts after creating the netlist, the system generated net name might change. This may cause net name conflicts when you run backannotation.
- OrCAD recommends that you do not connect a power symbol directly to a power pin.

Connect the power symbol to the power pin using a wire.

- You can place a part in the middle of a wire segment without redrawing the wire by placing the part over the wire such that two pins on the part connect with the wire segment. Then click the left mouse button over the part with the TAB key pressed until just the overlapping wire segment is selected. Finally, delete the wire segment.

Reuse Module as Hierarchical Block

To place a reuse module as a hierarchical block:

1. In Capture, place a hierarchical block on a schematic page by choosing the *Hierarchical Block* command from the Place menu. Reference the reuse module as an external design.
2. Annotate the design by choosing the *Annotate* command from the Tools menu.
3. Select the PCB Editor Reuse tab and enable the Renumber Design For Reuse Modules option. Complete the other options in the PCB Editor Reuse tab, then click OK.

⚠	After placing the reuse module, if you want to synchronize your Capture schematic with its corresponding PCB Editor .MDD equivalent, you must regenerate an PCB Editor netlist from Capture. In PCB Editor, you can either update the PCB Editor board or re-import the netlist onto the PCB Editor board. This action ensures the reuse properties are incorporated into the physical reuse module. The module can now be used in any schematic or physical design as a reuse module.

Running Design Rules Check - Physical Rules

Before generating a physical netlist for exporting to PCB Editor, it is a good design practice to verify your design by running Design Rules Check for validating physical rules.

Validating design for Physical Layout

The Design Rules Check tool scans schematic folders to verify that a design conforms to design rules; it generates a report of error and warning messages and places markers on the schematic page to help you locate problems.

1. In project manager, select the schematic design (*.dsn).
2. Choose *Tools - Design Rules Check*.
   The Design Rules Check dialog box appears with the Design Rules Check tab displayed.
3. To check physical rules, ensure that following options are selected.
   a. Check design rules option - Must for running DRC
   b. Run physical rules
4. Select the Physical Rules tab.
   In this tab, select the options to run different physical rule check that on your design. If

required, you can use the options available in the Custom DRC section, to configure and run a custom DRCs on your design.
It also provides options for the reports that you can view.
For detailed description of each option on the tab, see Design Rules Check dialog box.

5. Click OK.

	If you run a Design Rules Check on a single schematic page, Capture checks all pages in the entire schematic folder, which ensures that all nets on the schematic page are valid.

Shortcut

Toolbar: 

Browsing DRC markers

When you run the Design Rules Check tool, errors are marked on your schematic pages. Warnings are also marked, provided, the *Create DRC markers for warnings* check box was selected before running the design rules check.

When you run the Design Rules Check tool, Capture creates a report (.DRC) of warning and error messages. You can view the report in a text editor. These messages also appear in the session log.

In addition to the report, the Design Rules Check tool places error and warning markers on the schematic pages as well.

To browse DRC markers

To locate a DRC marker in the schematic, perform the following steps.

1. Choose *Edit - Browse - DRC Markers*.
   DRC markers are listed in the DRC Markers page.

2. Double-click on the DRC Marker.
   The schematic page editor opens with the marker displayed and highlighted.

See Also

Netlisting a Design

Working with Old Designs

Moving from Capture-Layout Flow To Capture-PCB Editor Flow

If you have an old design where the Capture schematic(.dsn) was in sync with the physical design in Layout (.max), you can migrate this to a Capture-PCB Editor flow.

1. Select the schematic design in Project Manager.
2. Choose *Accessories - Layout to PCB Editor Translator - Layout to PCB Editor.*
3. Specify the location of the .max file and the directory in which the translated board (.brd) file is to be saved.
4. It is recommended that you select the *Update dsn with brd* check box.

Selecting this check box ensures that changes made to the board are back-annotated on the schematic and schematic is in sync with the PCB Editor board. For example, running this translator changes the value of the footprint property. Unless new footprint value is back-annotated on schematic, further ECOs from schematic to layout will cause component rip-offs in PCB Editor board.
To view the setup and migration steps, see *Layout to PCB Editor Migration Guide* on Online Documentation or on Cadence Support website.

Using the IGNORE_PROP property to ignore the DEVICE property on your design

In cases where your design library includes the DEVICE property (an anachronism from previous releases), you can avoid having to remove the property from each part in your library by employing the IGNORE_PROP property. To ignore the DEVICE property on a complete design, define IGNORE_PROP as an environmental/system variable and assign it a value of "DEVICE."

As with all environmental variables, IGNORE_PROP is specific to a system login. You must have administrative privileges to define IGNORE_PROP as a system variable for your system. Also, you must restart Capture in order to read the new environmental variable settings.

16

Using Capture with PCB SI

The OrCAD Capture Signal Integrity (SI) analysis feature enables you to perform SI analysis early in the design cycle to avoid iterations at a later stage. You can launch Signal Explorer on a flat net from OrCad Capture to perform SI analyses and associate the Electrical Constraint set (Electrical Cset) to the flat net back to Capture from Signal Explorer. The complete topology file is also embedded into the DSN. OrCAD Capture also supports a distributed design environment for SI analysis by allowing you to export the net connectivity as topology files that can be updated using Signal Explorer and then imported to OrCAD Capture.

In Capture, you can set up SI libraries, assign SI models and then explore the signals in Signal Explorer. You can also export and import Electrical Csets in Capture. In addition, you can audit Electrical Csets and model assignments.

You can validate the Electrical Csets in Capture or import the topology files to Allegro Constraint Manager and perform audit on the files.

> All Signal Integrity tasks are available under the SI Analysis menu in OrCAD Capture.

The following sections provide details about the various tasks you need to perform in SI analysis flow.

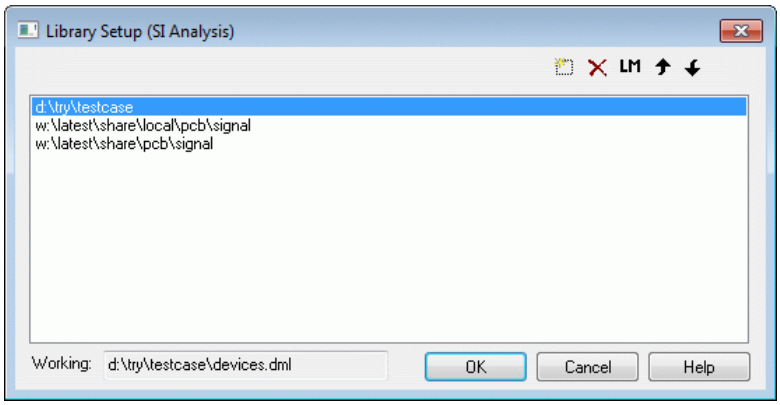
- Setting up a Library
- Assigning Models
- Managing Electrical Csets
- Validating Electrical Csets

Setting up a Library

You need to specify the libraries containing the DML or IBIS models to be able to assign these models to the parts in your design. You also need to specify a working library. Capture writes the assigned models to a device DML file in the working library. You can use the Library Setup dialog box to add libraries, change priority of added libraries by changing the sequential order, and remove libraries. You can also launch the Library Management utility to specify working libraries and to ignore libraries. You will not be able to assign models from the ignored libraries.

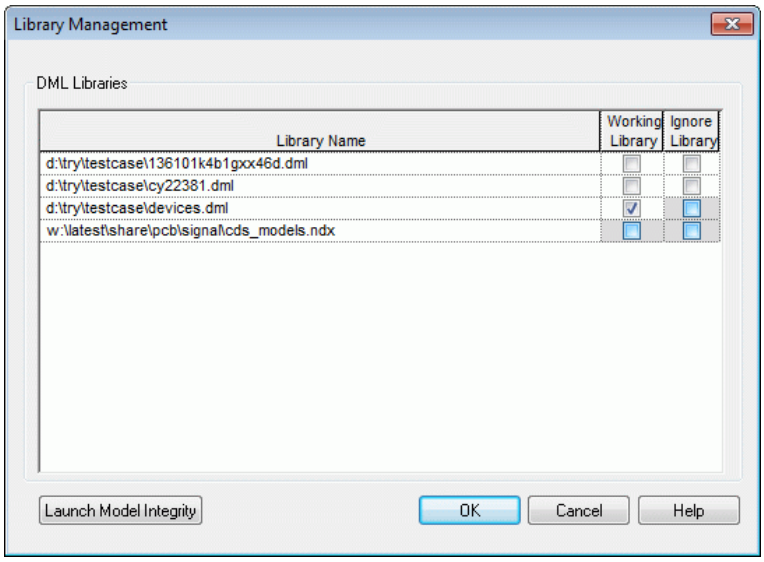
You can launch the Model Integrity tool that helps you ensure the integrity of the model data required for high-speed circuit simulations and allows you to create, manipulate, and validate models quickly in an easy-to-use editing environment. For more information on Model Integrity, refer *Model Integrity User Guide*.

To set up library, choose *SI Analysis –SI Library Setup* to launch the Library Setup (SI Analysis) dialog Box.



In the Library Setup (SI Analysis) dialog box, click the *Add a new library* ( ) button to add a library to list. You can also move the libraries up ( ) or down ( ) in the list, or delete ( ) a library.

You can click the *Launch Library Management* button ( ) to launch the Library Management dialog box.



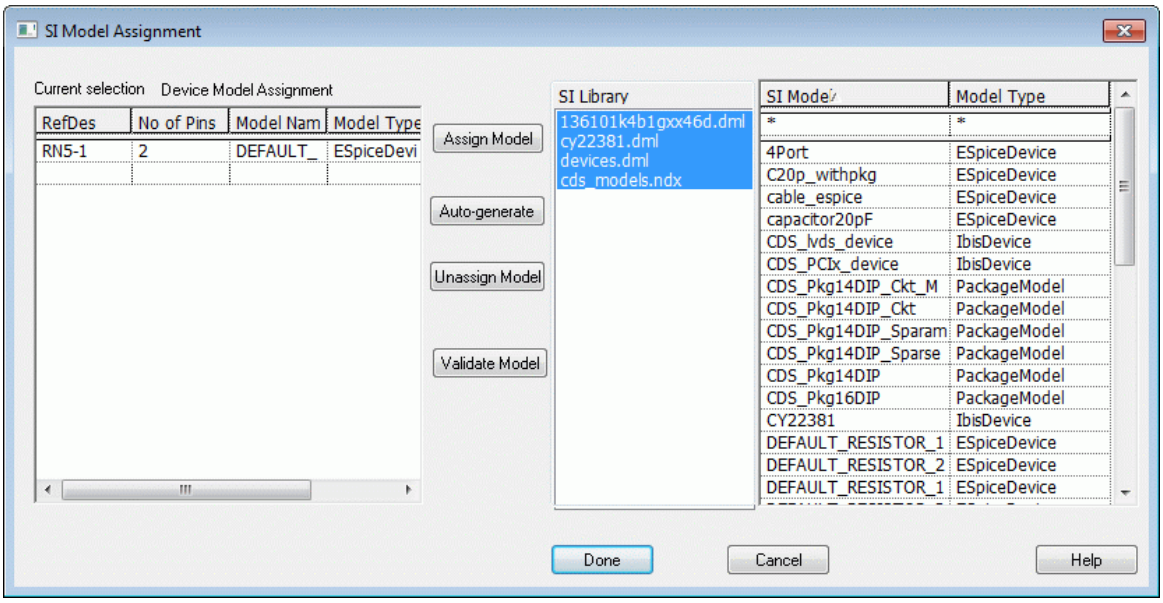
In the Library Management dialog box, you can select the *Working Library* field for any one of the libraries to specify it as your working library. Similarly, you can select the *Ignore Library* field for one or more listed libraries to ignore them. You can also launch Model Integrity by clicking *Launch Model Integrity*.

> You can also launch Model Integrity by choosing *SI Analysis – SI Model Integrity*.

Assigning Models

You can assign DML signal models on instances and pins by using one of the following options:

- Assign on individual instances and pins: Select the instances on the design and choose *SI Analysis – Assign SI Model* or choose *Signal Integrity – Assign SI Model* from the right-click menu. This launches the SI Model Assignment window.



You can either assign a model from an existing library or automatically generate a model. You can also use this dialog box to validate models and to remove model assignments.

You can choose *View – Selection Filter* and then choose *Assign SI Model* from the pop-up menu for any part or pin.

- Auto-assign for discrete models: Choose *SI Analysis – Auto Assign Discrete SI Models*. This generates and adds default models on all the discrete devices on the design.

The `SIGNAL_MODEL` property is added on occurrences instead of instances if there are differences in properties or if there are multiple occurrences of an instance.

> You cannot assign models to a part that is not in the occurrence hierarchy.

> In this flow `M` is the symbol for Meter. If you want to assign value in the Mega range such as MegaOHM to a part, use `Meg` instead of `M`, If you use M while specifying value for a part where Meter does not make sense, the M will be ignored while defining the `SIGNAL_MODEL` property.

You need to ensure that voltage is defined for the DC nets for proper extraction. If power and ground pins do not have voltage property, the performance might be affected. For example, if a large number of bypass capacitors are present and the ground pins do not have voltage property, the performance will be affected because each net will be explored. Choose *SI Analysis - Identify DC Nets* to modify or add voltages on the DC nets in the design.

Managing Electrical Csets

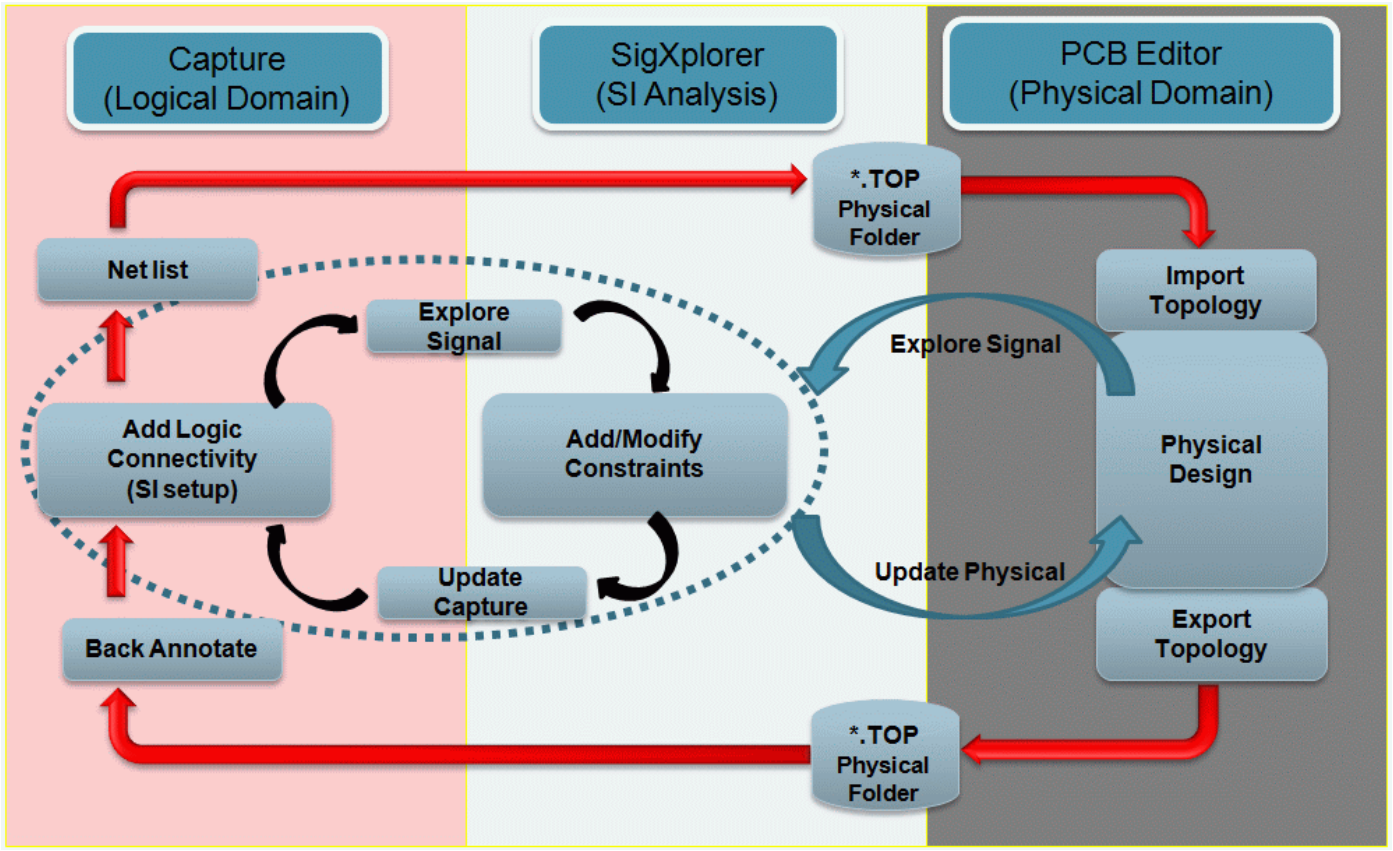
You can add Electrical Cset to your design either in the concurrent mode by exploring the signals in Signal Explorer or in the distributed mode by importing Electrical Csets. After adding the Electrical Cset to a design, you can assign or associate the Electrical Csets to the nets in your design. You can also export topology files that can be opened in the distributed mode in Signal Explorer.

> If you instantiate an external block in your top-level design and the block has Electrical Cset defined, you need to import the associated topology file into the top-level design.

In both the distributed and concurrent modes, you can import the Electrical Csets in PCB Editor and launch Constraint Manager for the Electrical Csets. You can backannotate any changes to OrCAD Capture from PCB Editor.

> If PROPAGATION_DELAY or RELATIVE_PROPAGATION_DELAY are present in the constraint set, an error message might be displayed when you perform Update Capture, Associate Electrical Cset, or Import Electrical Cset. You can ignore this error. For example, if PROPAGATION_DELAY is present in the constraint set, the following error message might be displayed:
>
> ERROR: unable to create property PROPAGATION_DELAY on object Xnet aaa TX+

Concurrent Mode



In concurrent mode, when Signal Explorer and Capture are in the same system, you can select any net on the design and choose *SI Analysis – Explore Signal* or choose *SI Analysis – Explore Signal* from the pop-up menu to generate XNET definition and launch Signal Explorer. A Topology file (`.top`) is generated and displayed by Signal Explorer.

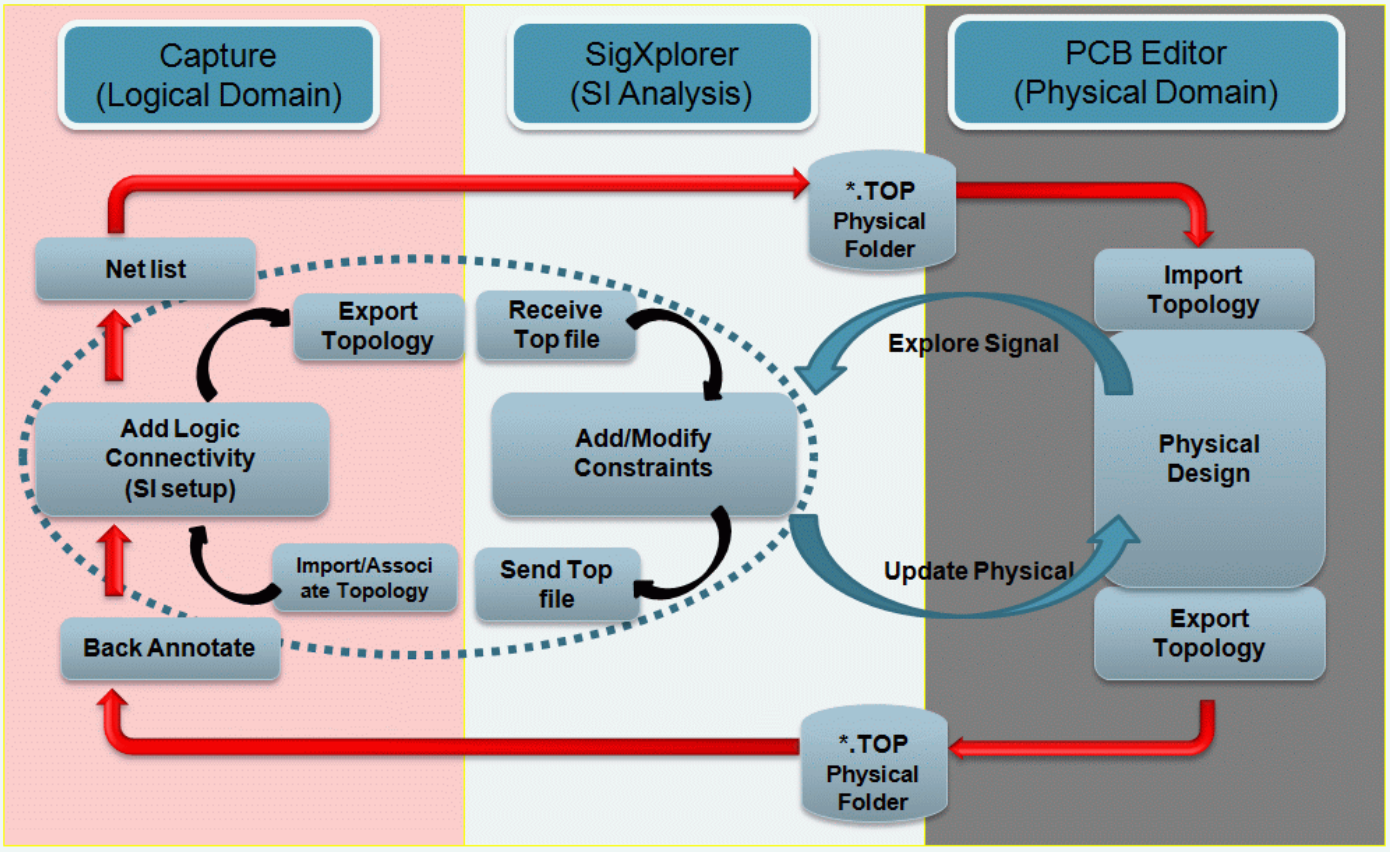
The displayed topology includes any discrete components with `SIGNAL_MODEL` property which are part of the XNET and any T-points that are part of the topology.

You can choose to change parameter values for discrete components in Signal Explorer and save the topology file. In Signal Explorer, choose *File – Save* to save the topology and then choose *File – Update Capture*.

> You might need to backannotate your design if you changed any properties for a component in Signal Explorer.

When you exit Signal Explorer, the topology file is added as the value of the `ELECTRICAL_CONSTRAINT_SET` property of the net in Capture.

Distributed Mode



In the distributed mode, where Signal Explorer is located in another system, you can use *Export Electrical Cset* to export the Cset file from OrCAD Capture. The exported Cset can then be sent for SI Analysis. After importing the Csets into SigXplorer and updating them, the Csets can be sent back to be imported and associated in OrCAD Capture using the *Import Electrical Cset* menu item.

> You must load and save the Electrical Csets in SigXplorer before importing and applying them in OrCAD Capture.

Validating Electrical Csets

You can validate Electrical Csets in Capture or in Constraint Manager.

To validate the Electrical Csets in Capture, choose *SI Analysis - Validate Electrical Cset Assignments*.

To validate Electrical Csets using Constraint Manager, netlist the design for PCB Editor and launch Constraint Manager. Import the topology file and then choose *Audit – Topology* to perform audit of the topology templates. You can also open and make changes to the topology file in Constraint Manager.

17

Designing for FPGA

In this section:

Overview of FPGA Projects

Capture provides a method for developing FPGA/CPLD devices. A schematic folder can represent the internal logic of a programmable device such as an FPGA or CPLD. The schematic folder can contain silicon-vendor provided primitive and macrofunction symbols, as well as user-created macrofunction symbols.

FPGA and CPLD vendors (Actel, Lattice, Xilinx, and so forth) provide a CAE Interface Package containing symbols that provide you with the building blocks to create a structural design description of an FPGA or CPLD, and simulation models so a functional and timing-based simulation can be performed on the design.

You can merge design modules created in a hardware description language like VHDL or OHDL with the design description with some vendors. External View or Properties can record the file name of the module.

Capture's netlist tool creates reports of the project and any HDL modules in a standard format such as EDIF or OrCAD INF. FPGA place-and-route tools or CPLD device-fitters read the design netlist and implement the logic into the physical constraints of the device.

You can also use the FPGA Export Dialog Box to specify settings for FPGA parts that you want to generate. You can then export a FPGA part using the FPGA Export Dialog Box. FPGA flow is supported in the PCB Editor flow from Capture. When you swap pins in PCB Editor for a FPGA component, the swapped pins can be backannotated to the Capture design. You can then export the design with the swapped pins. You can also specify reserved pins (Input, Bidirectional and

Output) in the FPGA component using the FPGA Export Dialog Box. This dialog box is invoked from the FPGA Export dialog box.

When you select Output Reserve Pins, the reserve pin assignment made in the Reserve Pin dialog will appear in the output tcl file. Not selecting this option does not remove the assignments, they do not appear in the tcl file.

When you choose Place – Swap – Pins in PCB Editor and click on a pin of a FPGA component, all the swappable pins are highlighted.

Making reserve pin assignments is currently supported for Altera devices.

Creating a FPGA project

To create a new FPGA project:

1. In Capture, choose *File – New – Project*.
   The New Project dialog box appears.
2. In the Name text box, enter the name for the new project.
3. Select *Programmable Logic Wizard*.
4. In the Location text box, enter the path where you want the new project files to be stored, or use the Browse button to locate the directory.
5. Click *OK*.
6. In the Programmable Logic Project Wizard page expand the vendor and then select the target family.
7. Specify the compiled library path. (See Compiling vendor simulation libraries)
8. Click *Finish*.

Compiling vendor simulation libraries

In order to simulate your design with NC VHDL, you must compile the simulation library that corresponds to the target technology. For example, if the target technology for your design is Xilinx XC9500, in order to simulate the design, you must compile the Xilinx UNISIM libraries.

You need only compile your simulation libraries once. If you have compiled a particular simulation library previously, you can skip this procedure. However, if you update your vendor libraries, or if you obtain a new version of NC VHDL, you must recompile the simulation libraries.

To compile the simulation library for the target vendor

1. From the *Tools* menu, choose *Compile vendor libraries*. Capture displays the NC VHDL Library Compilation dialog box.
2. In the list box, select the set of libraries that corresponds to the target technology of your design.

   > Legacy programmable logic projects (that is, projects created with a version of Capture previous to version 14.2) that include schematic components from the Xilinx XC4000E library, must use the OrCAD XC4KE VHDL libraries for simulation. However, Xilinx XC4000E schematics created with Capture version 14.2 (or any later versions), or schematics that have been modified to include the new Xilinx XC4000E part symbols, must use the UNISIM VHDL libraries for simulation.

3. If desired, select the *Run display in session log* option to include the results of the compilation in Capture's session log.
4. In the CMD Path text box, type (or browse for) the path to, and name of, the command file that contains the compile options for the vendor library.
5. Click *Compile*.
   Capture then compiles the simulation libraries.

The target vendor may include options for the compilation. You can specify any such options in the command file associated with the simulation libraries. An example command file, for Xilinx XC4000E libraries, is illustrated below:

echo off

rem -- File name: C_x4ke.bat

rem --

rem -- DESCRIPTION

rem -- This is a batch file to compile Xilinx XC4000E prerouted simulation models

rem -- Two files will be compiled in Cadence NC VHDL Desktop simulator:

rem -- 1> .\FPGA_LIB\library\xilinx\x4ke\x4ke.vhd

rem -- 2> .\FPGA_LIB\library\xilinx\x4ke\x4ke_m.vhd

rem --

rem -- The compiled library will be named: x4kelib

rem --

rem -- Before running this file, please made sure that Cadence NC VHDL Desktop simulator

rem -- was installed.

rem --

rmdir /S /Q .\sim_lib\x4kelib

mkdir .\sim_lib\x4kelib

echo on

ncvhdl -v93 -work x4kelib -log ncvhdl.log -messages .\Xc4000e\x4ke.vhd .\Xc4000e\x4ke_m.vhd

Introducing NC VHDL simulations

NC VHDL is the Cadence VHDL simulation tool suite that provides the means by which you can simulate your Capture design for both functional simulation (sometimes called functional verification) and timing simulation.

> For the specifics of working with NC VHDL, please refer to your NC VHDL documentation.

Starting NC VHDL

To start NC VHDL:

1. In Capture, choose *PICFlow – Simulate*.
   The Select Simulation Configuration dialog box appears.
2. Select:
   - Preroute: To perform functional simualtion
   - Postroute: To perform timing simulation

Setting the Simulation Mode

You can start NC VHDL, either in the batch mode or the interactive mode. When you select the simulation mode, choose the method that best provides for your needs. That is, if you want to experiment with different simulation resolutions, or run the simulation for different run times, choose Interactive mode. If you have already developed a simulation session that covers your design, and have saved it in a batch file, choose Batch mode.

Running NC VHDL in interactive mode

1. Click *Setup*.

2. In the *Simulation* tab (NC VHDL) of NC VHDL, set the paths to, and names of, your HDL.VAR file and Log directory.
3. In the Flow options area, specify the particular options you want to run in the NC VHDL session. These include:

Compile:	Specify that NC VHDL compile your design for simulation. This is a prerequisite for actually performing the simulation. You can provide command line options for the compile in the Command Options text box, by entering them exactly as you would from the command line. ⚠ You must also have compiled the simulation models that correspond to the target vendor library before you can simulate your design. See Compiling vendor simulation libraries for more information.
Elaborate:	Specify that NC VHDL perform elaboration (the process of mapping your design to machine code that NC VHDL can interpret) on your design. This is a prerequisite for actually performing the simulation. You can provide command line options for the elaboration in the Command Options text box, by entering them exactly as you would from the command line.
Simulate:	Run the simulation in interactive mode. You can provide command line options for the simulation in the Command Options text box, by entering them exactly as you would from the command line. ⚠ You need only compile and elaborate your design once. After that, when you start NC VHDL, you can skip the Compile and Elaborate steps, provided that the design has not changed since your last simulation run.

4. In the Sim Vision launch area, specify the particular options you want to run in the NC VHDL session. These include:

Start Sim Vision:	Causes NC VHDL to display its user interface in a new window on the screen. If you do not choose this option, NC VHDL runs in the background using the command options you have specified for Compile, Elaborate, and Simulate. In effect, this is similar to running NC VHDL in batch mode, but rather than using a batch file, NC VHDL uses the options specified in the NC VHDL dialog box.
Enter Interactive Mode:	Causes NC VHDL to set simulation time to 0 and await your inputs in order to run the simulation. This option is only available if you have selected the Start Sim Vision option.

5. Choose the *Testbench* tab (NC Verilog) tab, and specify whether you want to generate a

new testbench for your design, include an existing testbench, or not include a testbench in the current NC VHDL session.

> ⚠ If you choose None (that is, if you specify that no testbench should be included), you cannot simulate your design, since testbenches provide the stimulus. In general, you should only choose None when you want to compile and/or elaborate your design without simulating it.

6. Click *OK*. Capture returns to the NC VHDL Preroute Simulation dialog box.
7. Click *Run* to start the simulation session.

Performing functional simulation

In the typical FPGA flow, functional simulation provides a method for verifying the logic of your design without regard for timing constraints. That is, the simulation determines that the design produces outputs that coincide with applied inputs irrespective of gate or propagation delays, and that, therefore, the logic of the design is correct.

Functional simulation occurs before the actual implementation of the design (before synthesis and place-and-route).

If you determine that the design logic does not produce expected results, you can return to the design creation stage to correct any logic problems before committing the design to timing-specific gates.

The Cadence FPGA flow uses NC VHDL as the simulation tool for functional simulation.

For information specific to using the Capture tool suite with your particular vendor, please refer to the technical documents located on the Cadence web site: http://www.cadence.com/orcad.

Selecting the configuration for functional simulation

When you select the Preroute configuration for simulation, Capture creates a VHDL netlist of your design as it currently exists in the project manager and stores it in the Preroute folder. This netlist does not include timing information, and uses unit/delay for all components in the design. The purpose of functional simulation, as the name implies, is to verify logical functionality of the design without accounting for the timing of specific components or net delays.

To select the configuration for functional simulation:

1. From the *PICFlow* menu, choose the *Simulate* command. Capture displays the *Select Simulation Configuration* dialog box.
2. Select *Preroute* in the selection window. This indicates that you want to simulate your design functionality, without timing considerations.
3. Click *OK*. Capture generates a VHDL netlist of your design and stores it in the Preroute

folder of the project manager, then displays the NC VHDL Preroute Simulation dialog box
4. Specify either interactive or batch mode for the NC VHDL session.
5. Click *Setup*. Capture displays the NC VHDL Preroute Simulation Setup dialog box.
6. Complete the dialog box as described in Starting NC VHDL.

Performing Synthesis and Place-and-Route

After you have checked your design for behavioral compliance (using functional simulation), the next step is to implement it using timing-specific components from your vendor library. This is accomplished in two major steps:

- Synthesis (including optimization), where any HDL modules in the design are implemented using vendor components and then optimized using logic optimization algorithms;
- Place-and-route, where the design is mapped to a particular device from the target vendor.

Capture uses Synplicity's Synplify tool to perform synthesis and optimization.

For information specific to using the Capture tool suite with your particular vendor, please refer to the technical documents located on the Cadence web site: http://www.cadence.com/orcad.

Synthesis with Synplify

When you perform synthesis on your design, Synplify creates a structural (gate level) representation of all the schematic and VHDL components of your design. This structural representation is in the form of an EDIF netlist that you can then use an input for the vendor place-and-route tool.

Synplify also optimizes your design as part of the synthesis process. The optimization removes extraneous logic while maintaining any hierarchical boundaries for the design.

To perform synthesis and optimization on your design with Synplify:

1. From the *PICFlow* menu, choose the *Synthesize* command. Capture saves a VHDL netlist of the design and stores it in the Synthesis folder of the project manager, then displays the Synthesis Option dialog box.
2. Choose *Interactive Mode* if you want to run Synplify interactively.

or

1. Choose *Batch Mode*, if you want to run Synplify in batch mode and have a batch file that specifies the parameters for the synthesis session.
2. Select the *Create Synplify Project* option if you want to create a new Synplify project for this simulation run. Alternately, you can leave this option unselected and specify an existing Synplify project in the text box. However, a Synplify project must exist in order to run Synplify. Any Synplify project is created and stored in the Synthesis directory (of the project

manager).
3. Click *Run*. Capture invokes Synplify according to the parameters you set.

Place-and-route using the vendor tool set

The tool suite that Capture invokes for the place-and-route process depends on the target technology you have chosen for your programmable logic project. Of course, you must have installed the tool suite appropriate to the target technology in order for Capture to start the tool.

Each vendor has a unique tool suite that performs place-and-route for designs that use its technology. The following table provides an (incomplete) list of possible tool suites that correspond to some vendor technologies. This is not a complete list. Contact your vendor for information on the exact tool suite your require.

Vendor	Tool Suite
Actel	Actel Designer
Altera	MAX+PLUS II
Xilinx	Design Manager

To initiate place-and-route for your design:

1. From the *PICFlow* menu, choose the *P&R* command. Capture displays the Enter EDIF Netlist Location dialog box.
2. Enter the path to, and name of, the EDIF netlist that resulted from the synthesis run.
3. Click *Run*. Capture starts the vendor tool appropriate to the target technology of your design.

Performing Timing Simulation

Timing simulation provides a method for you to check your design to be sure that the timing properties of the components in the target vendor do not change the functionality of your design. That is, timing simulation allows you to be sure that propagation delays and other factors (setup times, capacitance, hold times, and so on) are within the tolerance levels of your design such that they do not affect the design's logic.

Timing data is generally included in your design during the place-and-route process and takes the form of a standard delay file (SDF) or an annotated VHDL netlist. These files are generated by your vendor's place-and-route tool.

The Cadence FPGA flow uses NC VHDL as the simulation tool for timing simulation.

For information specific to using the Capture tool suite with your particular vendor, refer to the technical documents located on the Cadence web site: http://www.cadence.com/orcad.

Selecting the configuration for timing simulation

When you select the Postroute configuration for simulation, Capture creates a VHDL netlist of your design as it currently exists in the project manager and stores it in the Postroute folder. Capture includes any SDF file associated with the design in the Postroute folder, as well.

To select the configuration for timing simulation:

1. From the *PICFlow* menu, choose the *Simulate* command. Capture displays the Select Simulation Configuration dialog box.
2. Select *Postroute* in the selection window. This indicates that you want to simulate the timing of your design using post-place-and-route, vendor-specific delays.
3. Click *OK*. Capture generates a VHDL netlist of your design and stores it in the Postroute folder of the project manager, then displays the Place and Route Settings dialog box dialog box.
4. Specify the paths for the place-and-route netlist and standard delay file created by your vendor place-and-route tool, then click *OK*. Capture displays the NC VHDL Simulation dialog box.
5. Specify either interactive or batch mode for the NC VHDL session.
6. Click the *Setup* button. Capture displays the NC VHDL Simulation Setup dialog box.
7. Complete the dialog box as described in Starting NC VHDL.

Generating a part for FPGA projects

Once you have created the final programmable logic netlist, you can generate a part for your project. This part represents the FPGA/CPLD component that is the result of the programmable logic design flow.

- Overview of Part Generation for FPGA
- Generating a part for your FPGA
- Supported Vendor Pad and Pin file formats

Overview of Part Generation for FPGA

Capture reads a variety of PLD vendor pin reports to create library parts for the Capture schematic system. Most PLD vendor pin reports (Actel, Vantis, Lattice, Lucent, OrCAD SPLD, and Xilinx formats) describe the pin number, signal name, and direction (or mode) of a package pin programmed by the place-and-route process. When you create a new part with the Generate Part

command, Capture creates a new schematic library (OLB) and part based on the pins defined in the report file. You can also define multiple sections for the part based on I/O Bank information present in the .pin or .pad files by selecting the Create multi-section part option in the FPGA Options dialog box. If you want to have separate power/GND symbols, you can use the Separate Symbols for Power/NC pins options to achieve that goal. The Power/NC pin limit will create the Power/GND symbols based on the set limit and will generate a new symbol if the limit is exceeded. You can now also assign auto-pingroup values to the pins in a given section. Pins can be grouped together according to I/O Bank or pins with compatible I/O Standards within an I/O Bank. Pins are sorted alphabetically by name, with input type pins located on the left and output or bidirectional pins on the right.

Generate part can create new parts or update the pin numbers of an existing library part with the Update pins on existing part in library option, which allows for engineering change orders (ECOs) from a programmable logic project to update the part of the system schematic.

The most common Implementation type used with the parts created from PLD vendor pin reports is type <none> or Project to create a hierarchy of projects for system simulation.

Capture can read an EDIF netlist or VHDL or Verilog model created by 3rd-party EDA tool or intellectual property (IP) provider to create library parts for a programmable logic project. Most 3rd-party synthesis and module generators describe the port name and mode of the signals that interface to the EDIF netlist, VHDL model, or Verilog model.

> To create a pin on a symbol using the Generate Part utility, the pin must have a pin to port mapping in the pin file.

Generating a part for your FPGA

1. From the *Tools* menu in the project manager, choose the *Generate Part* command. Capture displays the Generate Part dialog box.
2. Specify the final, fitted netlist in the Netlist File Name text box. Use the Browse button to locate the netlist file, if necessary. When you select a netlist with the Browse button, Capture places default values in the Part Name and Part Library Name text boxes, and automatically selects a Vendor File Type according to the file extension of the netlist file.
3. If necessary, enter a name and library for the part in the appropriate text boxes.
4. Specify that the part is a new part or that you want to update pins on an existing part in the library with the appropriate radio button.
5. Specify ascending or descending order for the part pins.
6. If you want to specify additional pins for the part, select the appropriate button and enter the number of pins in the Number of pins text box.
   By default, the part generator creates a part with a number of pins equal to the number of input and output ports in the netlist file. However, if you are using a specific device for your FPGA/CPLD component, you may want to specify the number of pins on that device, if it

differs from the number of netlist ports. This is especially true if you plan to use the part on a PCB schematic page.

7. If necessary, select the *Vendor File Type* for the netlist file from the drop-down list box.
8. If you specified a “raw” pin-out file in step 1, specify an implementation type, name, and file for the part.

   The most common implementation type used with the parts created from PLD vendor pin reports is either <none> or Project (which creates a hierarchy of projects for system simulation). Implementation types signify the following:
   <none> Primitive library part.
   EDIF Non-primitive library part. Contents defined by an EDIF netlist generated by a third-party EDA tool.
   Project Primitive library part. Associated with the Simulation Resources of a programmable logic project for system-level simulation.
   Schematic View Non-primitive library part. Contents defined by a schematic folder/page.
   VHDL Non-primitive library part. Contents defined by a VHDL model.
   Verilog Non-primitive library part. Contents defined by a Verilog model.
   The table below illustrates typical combinations of vendor file type and implementation types.

Vendor file type	Implementation type
Actel Pin	[<none>|Project]
Altera Pin	[<none>|Project
AMD/MINC JEDEC	[<none>|Project]
APD BGA/Die-Text File	[<none>|Project]
EDIF Netlist	[EDIF|<none>|Project]
Lattice Pin	[<none>|Project]
Lucent Pad	[<none>|Project]
OrCAD SPLD Pad	[<none>|Project]
Verilog Netlist	[<none>|Project|Verilog]
VHDL Netlist	[<none>|Project|VHDL]
Xilinx M1 Pad	[<none>|Project]
Xilinx Pin	[<none>|Project]
XNF Netlist	[<none>|Project]

You can also use the Actel Pin File format to create a part for your Atmel project by modifying the Atmel pin file.

9. Click *FPGA Setup* to open the FPGA Options dialog box.

   1. In the General tab, specify the settings for the symbol, pins, and pin swap.
   2. In the Pin Defaults tab, specify the default settings for pin direction and shape. Additionally, you can modify the pin-shape for low-asserted pins.
   3. Click OK to close the FPGA Options dialog box.

10. Click *OK*. Capture generates a library with the file partname.OLB and references it in the project manager Outputs folder.

    > Selecting Create multi-section part in the general tab of FPGA Setup dialog box will open Split Part Section Input Spreadsheet when you close the Generate Part dialog box. You can use this spreadsheet to modify the part.

    You can use the part you generate with the Generate Part command to represent the actual component (FPGA or CPLD) in schematics for other projects (including PCB designs). When you use the Generate Part command, Capture creates a part file (with a .OLB extension) and references it in the Outputs folder of the project manager.

Supported Vendor Pad and Pin file formats

Capture CIS supports the pad and pin file formats from the following vendors:

- Actel Pin File.

  For the following series:

  - 3200DX
  - ACT1
  - ACT2

- Altera Pin File.

  For the following series:

  - Quartus II
  - Maxplus II

- Lattice Pin File
- Lucent ORCA Pad File
- Xilinx Pad and Pin Files.

  For the following series:

  - Virtex
  - Spartan

Simulating the Board

Board-level simulation provides a method for you to simulate your entire PCB design (including any programmable logic devices created as FPGA projects in Capture) before migrating it to your board layout tool. You can create a test bench to provide stimulus and a design netlist for board-level simulation (only) with either VHDL or Verilog, by specifying one or the other in the Board Simulation tab of the Preferences dialog box.

Choosing an HDL for board simulation

You can use either VHDL or Verilog as the hardware development language for board simulation. When you simulate your PCB design, Capture creates a netlist, in the HDL that you specify, and uses it as the design source for the simulation.

> Be sure that, when you choose an HDL for board simulation, you choose an HDL that does not conflict with the elements of your design. That is, if your design includes Verilog components, you must choose Verilog as the board simulation HDL. Mixed Verilog/VHDL designs are not supported in this release of Capture.

To choose an HDL for board simulation:

1. Choose the Preferences command from the Options menu. Capture displays the Preferences dialog box.
2. Select the Board Simulation tab.
3. Select either Verilog or VHDL as the hardware development language to be used for board simulation.
4. Click OK.

Starting NC VHDL or NC Verilog for board-level simulation

NC VHDL and NC Verilog, collectively, comprise the NC Desktop simulation tool set that OrCAD provides. This tool set provides the means by which you can simulate your Capture design at the board level. For the specifics of working with the NC Desktop tools, please refer to your NC Desktop documentation.

To start NC VHDL or NC Verilog for board-level simulation:

1. Choose the Board simulation command from the Tools menu. Capture displays the NC Verilog Simulation dialog box or the NC VHDL Simulation dialog box, depending on the option you specified in the Board Simulation tab of the Preferences dialog box.
2. Choose the Use Interactive option.
   or
   Choose the Specify batch file option, and enter the path to, and name of the batch file in the

text box.

3. Click Setup. Capture displays the NC VHDL Simulation Setup dialog box or the NC Verilog Simulation Setup dialog box, depending on the option you specified in the Board Simulation tab of the Preferences dialog box.

NC VHDL in Batch Mode

When you create a batch file for your NC VHDL simulation, it is best (and easiest) if you have already performed an interactive simulation run on your design, and you are confident that the simulation provides adequate coverage for your design. NC VHDL automatically creates log files for each step in the simulation process; specifically compilation, elaboration, and simulation.
The compilation log file, NCVHDL.LOG, contains information about the compilation of your design. An example compilation log file follows:

Example NCVHDL.LOG file

```
ncvhdl: v3.20.(p1): (c) Copyright 1995 - 2000 Cadence Design Systems,
ncvhdl: *W,DLNOHV: Unable to find an 'hdl.var' file to load in.
ncvhdl: v3.20.(p1): Started on Aug 10, 2001 at 11:16:27
ncvhdl
-work worklib
-cdslib E:/cad/junk4/cds.lib
-logfile ncvhdl.log
-errormax 15
-update
-v93
-messages
-status
E:/cad/junk4/PREP4.VHD
E:/cad/junk4/PREP4TB.VHD
...
```

In this example, the text in **bold** represents the compiler invocation and the associated options. To create a batch file that replicates the compile run, copy the compiler invocation and the associated options into a text file. You can name the file anything you want, as long as it has a .BAT extension. So, for example, you might name the file SIMULATE.BAT.
It is best to include all the options on a single line, thus:

```
ncvhdl -work worklib -cdslib ./cds.lib -logfile ncvhdl.log -errormax 1
```

> Note that you should also change any "hard" paths in the file to dynamic paths, to facilitate using the batch file on different systems. For example, using the illustration above, the paths for the two .VHD files are changed from "hard" to dynamic when they are included in the batch file.

Similarly, you can include appropriate text from the elaboration and simulation log files in your batch file. Continuing with the example, an elaboration log file, NCELAB.LOG, might contain the following data:

Example NCELAB.LOG file

```
ncelab: v3.20.(p1): (c) Copyright 1995 - 2000 Cadence Design Systems,
ncelab: *W,DLNOHV: Unable to find an 'hdl.var' file to load in.
ncelab: v3.20.(p1): Started on Aug 10, 2001 at 11:16:37
ncelab
-work worklib
-cdslib E:/cad/junk4/cds.lib
-logfile ncelab.log
-errormax 15
-messages
-status
worklib.prep4tb:test_bench
...
```

A simulation log file, NCSIM.LOG, might contain the following data:

Example NCSIM.LOG file

```
ncsim: v3.20.(p1): (c) Copyright 1995 - 2000 Cadence Design Systems, I
ncsim: *W,DLNOHV: Unable to find an 'hdl.var' file to load in.
ncsim: v3.20.(p1): Started on Aug 10, 2001 at 11:16:47
ncsim.dll
+cuzco_gui
-gui
-cdslib E:/cad/junk4/cds.lib
-logfile ncsim.log
-errormax 15
-messages
-status
worklib.prep4tb:test_bench
...
```

With simulation log files, before including the relevant text into your batch file, you should remove the .DLL extension from the invocation, as well as the +CUSCO_GUI option, thusly:

```
ncsim -gui -cdslib ./cds.lib -logfile ncsim.log -errormax 15 -messages
./worklib.testbench_file:test_bench
```

To complete the example, the illustration below shows a batch file, SIMULATE.BAT, that includes invocations (with associated options) for the compilation, elaboration, and simulation of a design:

Example batch file for NC VHDL

```
ncvhdl -work worklib -cdslib ./cds.lib -logfile ncvhdl.log -errormax 1
ncelab -work worklib -cdslib ./cds.lib -logfile ncelab.log -errormax 1
ncsim -gui -cdslib ./cds.lib -logfile ncsim.log -errormax 15 -messages
```

For the specifics of working with NC VHDL, please refer to your NC VHDL documentation.

VHDL and Verilog reserved words

In this section:

VHDL reserved words

The 1076-93 VHDL standard lists the following reserved words. For complete information, see the specification and associated documents.

abs	access	after	alias	all
and	architecture	array	assert	attribute
begin	block	body	buffer	bus
case	component	configuration	constant	disconnect
downto	else	elsif	end	entity

exit	file	for	function	generate
guarded	if	in	inout	is
label	library	linkage	loop	map
mod	nand	new	next	nor
not	null	of	on	open
or	others	out	package	port
procedure	process	range	record	register
rem	report	return	select	severity
signal	subtype	then	to	transport
type	units	until	use	variable
wait	when	while	with	xor

Verilog reserved words

always	and	assign	begin	buf
bufif0	bufif1	case	casex	casez
cmos	deassign	default	defparam	disable
edge	else	end	endattribute	endcase
endmodule	endfunction	endprimitive	endspecify	endtable
endtask	event	for	force	forever
fork	function	highz0	highz1	if
initial	inout	input	integer	join

large	macromodule	medium	module	nand
negedge	negedge	nmos	nor	not
notif0	notif1	or	output	parameter
pmos	posedge	primitive	pull0	pull1
pullup	pulldown	rcmos	real	realtime
reg	release	repeat	rnmos	rpmos
rtran	rtranif0	rtranif1	scalared	signed
small	specify	specparam	strength	strong0
strong1	supply0	supply1	table	task
time	tran	tranif0	tranif1	tri
tri0	tri1	triand	trior	trireg
unsigned	vectored	wait	wand	weak0
weak1	while	wor	xnor	

18

PCB Editor netlist files

In this section:

PSTCHIP.DAT

The PSTCHIP.DAT file contains a description for each physical part used in a Capture design. The Capture netlister extracts this physical description from properties on all occurrences rather than just instances.

In this section:

PSTCHIP File format

Here is the file format for the PSTCHIP.DAT file:

```
FILE_TYPE=LIBRARY_PARTS;
primitive 'Part Name';
pin
'Name':
PIN_NUMBER='(Number,Number...)';
INPUT_LOAD='(*)';
OUTPUT_LOAD='(*)';
OUTPUT_TYPE='(Type)';
PIN_GROUP='PinGroup';
.
.
.
end_pin;
body
POWER_PINS='(Power:Number;Ground:Number)';
```

```
PART_NAME='Source Package';
JEDEC_TYPE='PCB Footprint';
VALUE='Value';
NC_PINS='(Number,Number)';
Package (Component Definition) Property='occurrence_value;
.
.
.
end_body;
end_primitive;
END.
```

PSTCHIP File Elements

PSTCHIP.DAT section	Description
Header	This line begins the PSTCHIP.DAT file by declaring the file type. A PSTCHIP.DAT file always starts with the FILE_TYPE=LIBRARY_PARTS statement.
primitive	A primitive is the description of the physical part.
Part Name	Concatenation of Source Package, PCB Footprint and other properties found in the [ComponentDefinitionProps] section of the configuration file used for netlisting.
pin	Starts the pin section.
Name	Pin name. There is a section for every pin name.
Number, Number…	The pin number for that pin name. If you have a multi-section part, then the pin numbers containing that pin name are separated by commas.
INPUT_LOAD	The netlister assigns this property to an input pin. The input local current is measured in milliamperes. If there is an output load on an output pin you get an OUTPUT_LOAD property.
OUTPUT_TYPE	Netlister assigns this property to define an output pin as open collector, open-emitter, or tri-state (3 state). This data is used to make sure all outputs on a net have the same output type. The OUTPUT_TYPE property also specifies the logic function created by tying the outputs together.

Type	Value of the output pin type when open collector, open emitter, 3 state.
PinGroup	This is taken from the PinGroup column in the part editor package property spreadsheet. To see the spreadsheet, from the Package menu choose View, then from the Properties menu choose Edit. This property only shows up in PSTCHIP.DAT if you have a positive value for PinGroup meaning that pin is swappable with the other input pins for that section in the multi-section part.
Power:Numbers; Ground:Numbers	This POWER_PINS line defines the default power and ground requirements for the physical part. Power or ground pins that need to be connected together on the board will share the same name. This syntax for this line is: name of your power pins (VCC for example):the numbers of the power pins; the name of the ground pins:the numbers of the ground pins
NC_PINS	Describes the pins not connected to the logic, but which are present in the physical package. Currently the netlister gets this value from an NC property added to the part and this property has a value of the pins you want to be no connects; these values are separated by commas. You can also cause this line to be generated if you put a no-connect symbol on a pin in the schematic. This property doesn't show up if you don't have any NC pins in your design.
Package (Component Definition) Property	Any property found in the property editor on a part that is specified as a property to use in the [ComponentDefinitionProps] section of the configuration file. Properties are separated by commas. The last property is followed by a semicolon. There can by any number of properties. PART_NAME, JEDEC_TYPE, and VALUE are always given in this section regardless of what is in the configuration file.
occurrence_value	The occurrence value of the property is given between the single quotation marks.

PSTCHIP Sample file

The PSTCHIP.DAT file contains one or more primitives, organized into a pin section and a body section. Here is an example:

```
FILE_TYPE=LIBRARY_PARTS;
{ Using PSTWRITER 14.0-p002 Oct-09-2000 at 10:32:05}
primitive 'OR14';
```

```
pin
'I0':
PIN_NUMBER='(1,4,9,12)';
INPUT_LOAD='(*)';
PIN_GROUP='1';
'I1':
PIN_NUMBER='(2,5,10,13)';
INPUT_LOAD='(*)';
PIN_GROUP='1';
'O':
PIN_NUMBER='(3,6,8,11)';
OUTPUT_LOAD='(*)';
end_pin;
body
POWER_PINS='(VCC:14)';
POWER_PINS='(GND:7)';
PART_NAME='74LS32_0';
CLASS='IC';
JEDEC_TYPE='dip14_3';
VALUE='74LS32';
end_body;
end_primitive;
primitive 'AND14';
pin
'I0':
PIN_NUMBER='(1,4,9,12)';
INPUT_LOAD='(*)';
PIN_GROUP='1';
'I1':
PIN_NUMBER='(2,5,10,13)';
INPUT_LOAD='(*)';
PIN_GROUP='1';
'O':
PIN_NUMBER='(3,6,8,11)';
OUTPUT_LOAD='(*)';
end_pin;
body
POWER_PINS='(VCC:14)';
POWER_PINS='(GND:7)';
PART_NAME='74LS08_0';
CLASS='IC';
JEDEC_TYPE='dip14_3';
VALUE='74LS08';
end_body;
end_primitive;
primitive '74LS04_IC_DIP14_3_74LS04';
```

```
pin
'I':
PIN_NUMBER='(1,3,5,9,11,13)';
INPUT_LOAD='(*)';
'O':
PIN_NUMBER='(2,4,6,8,10,12)';
OUTPUT_LOAD='(*)';
end_pin;
body
POWER_PINS='(VCC:14)';
POWER_PINS='(GND:7)';
PART_NAME='74LS04';
CLASS='IC';
JEDEC_TYPE='dip14_3';
VALUE='74LS04';
end_body;
end_primitive;
END.
```

PSTXNET.DAT

The PSTXNET.DAT file is the connectivity file. This file lists each net, its properties, its attached nodes, and node properties. The list is ordered by physical net name and contains all net properties and the logic-to-physical binding of nets and nodes.

In this section:

PSTXNET File Elements

PSTXNET.DAT section	Description
NET_NAME	Marks the beginning of a net entry. The net name entry always ends with a semicolon after the net property list.
'Name'	Name of the net or the value of the Name property. If you have a net alias, it is used as the name. This is the flat net name, so if a child schematic has a different name or alias than the same net on the root, the name or alias on the root is the one that gets used.

Canonical Path	The first canonical path uniquely identifies each net in your schematics. It contains your design name, schematic folders, name, and other identifiers.
NODE_NAME	Marks the beginning of a node entry. The node name entry always ends with a semicolon after the node property list.
Reference	The reference of the physical part.
Number	The pin number on the part attached to the net.
Canonical Path	The second canonical path uniquely identifies each part the net is attached to in your schematic pages. It contains your design name, schematic folders, ID, and other identifiers.
Type	Pin type of the pin attached to the net: input (I), output (O), bidirectional (IO), and so on.
Net Property	Any property found in the property editor that is specified as a property to use in the configuration file, [netprops] section. Properties are separated by commas. The last property in the list is followed by a semicolon. There can by any number of properties. An example would be ECL='TRUE';
occurrence_value	The occurrence value of the property is given between the single quotation marks.

PSTXNET File format

```
FILE_TYPE=EXPANDEDNETLIST;
NET_NAME
'Name'
'Canonical Path';
NODE_NAME Reference Number
'Canonical Path':
'Type':;
Net Property='occurrence_value',
.
.
. ;
END.
```

PSTXNET Sample file

```
FILE_TYPE = EXPANDEDNETLIST;
{ Using PSTWRITER 14.0-p002 Oct-09-2000 at 10:32:05 }
NET_NAME
'N00011'
'@FULLADD.FULLADD(SCH_1):N00011':
C_SIGNAL='@fulladd.fulladd(sch_1):n00011',
ECL='TRUE';
NODE_NAME U3 3
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679612@T
'I':;
NODE_NAME U2 4
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679613@F
'I0':;
NODE_NAME U2 9
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679614@F
'I0':;
NODE_NAME U1 8
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679615@F
'O':;
NET_NAME
'SUM'
'@FULLADD.FULLADD(SCH_1):SUM':
C_SIGNAL='@fulladd.fulladd(sch_1):sum';
NODE_NAME U1 6
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679615@F
 'O':;
NET_NAME
'X'
'@FULLADD.FULLADD(SCH_1):X':
C_SIGNAL='@fulladd.fulladd(sch_1):x';
NODE_NAME U3 5
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679610@T
'I':;
NODE_NAME U2 12
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679611@
'I0':;
NODE_NAME U4 5
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679614@
'I1':;
NET_NAME
'Y'
'@FULLADD.FULLADD(SCH_1):Y':
C_SIGNAL='@fulladd.fulladd(sch_1):y';
NODE_NAME U3 9
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679612@T
```

```
'I':;
NODE_NAME U4 1
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679613@F
'I0':;
NODE_NAME U4 4
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679614@F
'I0':;
NET_NAME
'CARRY_IN'
'@FULLADD.FULLADD(SCH_1):CARRY_IN':
C_SIGNAL='@fulladd.fulladd(sch_1):carry_in';
NODE_NAME U3 1
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679610@T
'I':;
NODE_NAME U2 1
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679611@F
'I0':;
NODE_NAME U2 10
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679614@F
'I1':;
NET_NAME
'N00013'
'@FULLADD.FULLADD(SCH_1):N00013':
C_SIGNAL='@fulladd.fulladd(sch_1):n00013';
NODE_NAME U1 1
'@FULLADD.FULLADD(SCH_1):I505679590@FULLADD.74LS32_0.NORMAL(CHIPS)':
'I0':;
NODE_NAME U2 8
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679614@F
'O':;
NET_NAME
'N00023'
'@FULLADD.FULLADD(SCH_1):N00023':
C_SIGNAL='@fulladd.fulladd(sch_1):n00023';
NODE_NAME&#9;U1 2
'@FULLADD.FULLADD(SCH_1):I505679590@FULLADD.74LS32_0.NORMAL(CHIPS)':
'I1':;
NODE_NAME U4 6
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679614@F
'O':;
NET_NAME
'CARRY_OUT'
'@FULLADD.FULLADD(SCH_1):CARRY_OUT':
C_SIGNAL='@fulladd.fulladd(sch_1):carry_out';
NODE_NAME&#9;U1 3
'@FULLADD.FULLADD(SCH_1):I505679590@FULLADD.74LS32_0.NORMAL(CHIPS)':
```

```
'O':;
NET_NAME
'X_BAR'
'@FULLADD.FULLADD(SCH_1):X_BAR':
C_SIGNAL='@fulladd.fulladd(sch_1):x_bar';
NODE_NAME U3 2
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679610@T
'O':;
NODE_NAME U2 5
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679613@F
'I1':;
NET_NAME
'N5056796111'
'@FULLADD.FULLADD(SCH_1):N5056796111':
C_SIGNAL='@fulladd.fulladd(sch_1):n5056796111';
NODE_NAME U2 2
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679611@F
'I1':;
NODE_NAME U3 4
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679612@T
'O':;
NET_NAME
'N00032'
'@FULLADD.FULLADD(SCH_1):N00032':
C_SIGNAL='@fulladd.fulladd(sch_1):n00032';
NODE_NAME U2 3
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679611@F
'O':;
NODE_NAME U1 4
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679615@F
'I0':;
NET_NAME
'N00034'
'@FULLADD.FULLADD(SCH_1):N00034':
C_SIGNAL='@fulladd.fulladd(sch_1):n00034';
NODE_NAME U1 5
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679615@F
'I1':;
NODE_NAME U2 6
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679613@F
'O':;
NET_NAME
'X_BAR_74'
'@FULLADD.FULLADD(SCH_1):X_BAR_74':
C_SIGNAL='@fulladd.fulladd(sch_1):x_bar_74';
```

```
NODE_NAME U3 6
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679610@T
'O':;
NODE_NAME U4 2
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679613@F
 'I1':;
NET_NAME
'N5056796111_76'
'@FULLADD.FULLADD(SCH_1):N5056796111_76':
C_SIGNAL='@fulladd.fulladd(sch_1):n5056796111_76';
NODE_NAME U2 13
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679611@F
'I1':;
NODE_NAME U3 8
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679612@T
'O':;
NET_NAME
'N00032_77'
'@FULLADD.FULLADD(SCH_1):N00032_77':
C_SIGNAL='@fulladd.fulladd(sch_1):n00032_77';
NODE_NAME U2 11
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679611@F
'O':;
NODE_NAME U1 9
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679615@F
'I0':;
NET_NAME
'N00034_79'
'@FULLADD.FULLADD(SCH_1):N00034_79':
C_SIGNAL='@fulladd.fulladd(sch_1):n00034_79';
NODE_NAME U1 10
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679615@F
'I1':;
NODE_NAME U4 3
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679613@F
'O':;
END.
```

PSTXPRT.DAT

The PSTXPRT.DAT file (the expanded part list) lists each reference designator and the sections assigned to it. The PSTXPRT.DAT file is ordered by reference designator and section number.

In this section:

PSTXPRT File Elements

PSTXPRT.DAT section	Description
DIRECTIVES	Marks the beginning of the directives section. Directives always end with a semicolon.
PST_VERSION	Version of PCB Editor interface.
ROOT_DRAWING	Root schematic folder of design in Capture.
POST_TIME	Date and time of netlist.
SOURCE_TOOL	Tool used is Capture Writer or Design Entry HDL Writer.
END_DIRECTIVES	Marks the end of the Directives section.
Part Reference	The reference designator name of the physical part.
ComponentInstanceProperty	Properties and values of any component instance (package) properties found on the part and listed in the configuration file under the [ComponentInstanceProperty] section.
PART NAME	Concatenation of Source Package, PCB Footprint, and other properties found in the [ComponentDefinitionProps] section of the configuration file used for netlisting.
SECTION_NUMBER #	Marks the beginning of a physical section number. Each section of the package used gets its own section number. Single section parts have only one section number.
Canonical path	The canonical path uniquely identifies each part in your schematic pages. It contains your design name, schematic folders, part ID, source part, implementation type, and other identifiers.
Physical path	The physical path uniquely identifies each part in a design. The physical path contains the design name, schematic folder, page

	number, part ID, source part, implementation type, and other identifiers specific to the selected part in the design.
Part (function) property	Any property found in the property editor that is specified as a property to use in the configuration file, [functionprops] section. Properties are separated by commas. The last property in the list is followed by a semicolon. There can by any number of properties.
occurrence_value	The occurrence value of the property is given between the single quotation marks.
PRIM_FILE	Location of the where package properties are listed. This is the PSTCHIP.DAT file which is closely linked to the PSTXPRT.DAT file.
designator	The designator is now stored so that we know if designators are numeric or alphabetic.

PSTXPRT File format

```
FILE_TYPE = EXPANDEDPARTLIST;
DIRECTIVES
PST_VERSION='PST_HDL_CENTRIC_VERSION_0';
ROOT_DRAWING = 'Root schematic folder of design' ;
POST_TIME = 'Date and Time of Netlist' ;
SOURCE_TOOL='Capture_Writer';
END_DIRECTIVES ;
PART_NAME
Part Reference ’PART NAME’;
ComponentInstanceProperty='occurence_value';
.
.
SECTION_NUMBER #
'Canonical path',
'Physical path',
Part(function) property='occurrence_value',
.
.
.
PRIM_FILE='.\pstchip.dat',
SECTION='designator';
END.
```

PSTXPRT Sample file

```
FILE_TYPE = EXPANDEDPARTLIST;
{ Using PSTWRITER 16.5.0 p001Apr-05-2011 at 10:09:06 }
DIRECTIVES
 PST_VERSION='PST_HDL_CENTRIC_VERSION_0';
 ROOT_DRAWING='FULLADD';
 POST_TIME='Mar 29 2011 00:05:38';
 SOURCE_TOOL='CAPTURE_WRITER';
END_DIRECTIVES;

PART_NAME
 U1 'ORGATE':;

SECTION_NUMBER 1
 '@FULLADD.FULLADD(SCH_1):I505679590@FULLADD.74LS32.NORMAL(CHIPS)':
 C_PATH='@fulladd.fulladd(sch_1):i505679590@fulladd.\74ls32.normal\(ch
 P_PATH='@fulladd.fulladd(sch_1):page1_i505679590@fulladd.\74ls32.norm
 PRIM_FILE='.\pstchip.dat',
 SECTION='A';

SECTION_NUMBER 2
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679615@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='B';

SECTION_NUMBER 3
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679615@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='C';

PART_NAME
 U2 'ANDGATE':;

SECTION_NUMBER 1
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679611@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='A';
```

```
SECTION_NUMBER 2
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679613@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='B';

SECTION_NUMBER 3
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679614@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='C';

SECTION_NUMBER 4
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679611@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='D';

PART_NAME
 U3 'NOTGATE':;

SECTION_NUMBER 1
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679610@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='A';

SECTION_NUMBER 2
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679612@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='B';

SECTION_NUMBER 3
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679610@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='C';
```

```
SECTION_NUMBER 4
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679612@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='D';

PART_NAME
 U4 'ANDGATE':;

SECTION_NUMBER 1
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679613@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='A';

SECTION_NUMBER 2
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679614@
 C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1
 PRIM_FILE='.\pstchip.dat',
 SECTION='B';

END.
```

19

SDT configuration files

Capture requires either an SDT.CFG file or an SDT.BCF file when translating SDT designs into Capture. If you don't have one of these files, you can create an SDT.CFG in a text editor using the sample file provided here.
You may need to edit the following items in your SDT.CFG file to insure correct importation of your schematic folders:

- Path and libraries
- Page dimensions and units
- Part fields

  After you finish editing the SDT.CFG file, save it in the directory where you keep your SDT schematic folders. Capture will use the SDT.CFG file to locate the libraries and parts needed to translate your schematic folders.

In this section:

Path and libraries

The PLIB= line in the SDT.CFG file specifies the exact path to the SDT libraries (.LIB files). Each library must be specified by a LIB= line. Make only one 'PLIB=' line in your SDT.CFG file. Make one 'LIB=' line for each library your SDT schematic folders use. If you keep your SDT libraries in the same directory as your SDT schematic folder, use the following lines in the SDT.CFG (where pathname specifies the directory path where the .LIB files are located, and filename specifies the name of one .LIB file):

```
PLIB = 'pathname\*.LIB'
LIB = '.\filename.LIB'
```

Page dimensions and units

If design is in metric units, replace the corresponding lines above with those found below:

```
HOR = 264000 374300 548300 795300 1143300
```

```
VRT = 177000 264000 374300 548300 795300
P2P = 2540 2540 2540 2540 2540
UNTS = 'METRIC'
```

Part fields

Use part field lines in the SDT.CFG file to map properties and values of parts from SDT to Capture. The eighth part field is typically reserved for the PCB footprint.

Sample SDT.CFG file

The following is a sample SDT.CFG file:

```
PLIB = 'C:\ORCADESP\SDT\LIBRARY\*.LIB'
LIB = 'TTL.LIB'                          Design library filename
FN1 = '1ST PART FIELD'
FN2 = '2ND PART FIELD'
FN3 = '3RD PART FIELD'
FN4 = '4TH PART FIELD'
FF5 = '5TH PART FIELD'
FN6 = '6TH PART FIELD'
FN7 = '7TH PART FIELD'
FN8 = 'PCB Footprint'
HOR = 9700 17000 20200 32200 42200
VRT = 7200 11000 15200 20200 32200
P2P = 100 100 100 100 100
SIZ = 'B'
UNTS = 'ENGLISH'
ATB = 0
```

20

Netlist examples

This chapter provides a brief overview of some of the netlist formats available from Capture.

In this section:

Accel netlist format

The Accel PCB format netlists from ACCEL Technologies have these characteristics:

- All ASCII characters are legal.
  For more information, see the ACCEL Technologies website or the Protel website.

Example

```
(compinst "Y1"
(patternName "10MHz")
(compvalue "10MHz "))
(compinst "Y2"
(patternName "DIP.100/14/W.300/L.800")
(compvalue "24.576MHz"))
(compinst "Y3"
(patternName "4.9152MHz")
(compvalue "4.9152MHz"))
(compinst "Y4"
(patternName "3.6864MHz")
(compvalue "3.6864MHz"))
(net "N03627"
(node "JP5" "26")
(node "R43" "2"))
(net "N08082"
(node "L1" "2")
(node "C8" "1")
(node "L2" "1"))
(net "N03663"
(node "U72" "1")
(node "R53" "2"))
(net "N08139"
(node "L6" "2")
(node "C19" "1")
```

```
(node "R20" "1")
(node "C15" "2"))
```

Algorex netlist format

The Algorex format has these characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to six digits following the "N" prefix.
- Pin names are not used.
  All ASCII characters are legal.

Example

```
GND
U1 (14DIP300)-7,
U2 (14DIP300)-7
VCC
U1 (14DIP300)-14,
U2 (14DIP300)-14
CLOCK
U1 (14DIP300)-10
Q
U1 (14DIP300)-6,
U2 (14DIP300)-2,
U1 (14DIP300)-9
OUT
U2 (14DIP300)-3
B
U1 (14DIP300)-4
N00019
U1 (14DIP300)-3,
U2 (14DIP300)-1
N00013
U1 (14DIP300)-5,
U1 (14DIP300)-8
A
U1 (14DIP300)-1,
U1 (14DIP300)-2
```

Altera ADF netlist format

The AlteraADF format has these characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- All ASCII characters are legal.

Altera netlist constraints

When you create an AlteraADF netlist, you must include the OrCAD-supplied ALTERA_P.OLB and ALTERA_M.OLB libraries in your project. You can use only the parts in these two libraries to create the schematic design.
Inputs and outputs are handled differently in Capture and the Altera software. Capture defines inputs and outputs with hierarchical ports and library objects. Altera defines inputs and outputs with a library object, which is then tagged with the appropriate pin number. In the example schematic page, the CLOCK signal is an input and the STROBE signal is an output.
Additionally, library objects with unused pins default to predefined levels in the Altera software. Because Capture does not default unconnected pins to any particular level, you must tie all unused pins to the appropriate level.

Altera pipe commands

You can place equations in your schematic folder to be included in the netlist. To place these equations on the schematic page, choose the Text command from the Place menu.
Each equation must start with the pipe character (|). The first line must be:

```
|EQUATIONS
```

This tells Capture that some AlteraADF equations need to be included in the netlist. The equations can contain any information you want to include in the netlist.

Altera title block information

Title block information is placed in the first 10 lines of the netlist. The following table shows an example netlist header and the title block information from which the header was extracted. Header information in bold is text entered in the schematic page's title block.

i

Line	Example header	Title block field
1	ADF Example	Title of schematic page
1	May 15, 2002	Date
2	OrCAD-02	Document number

2	A	Revision code
3	OrCAD	Organization name
4	9300 SW Nimbus Avenue	1st Address Line
6	Turbo = ON	3rd Address Line
7	5C031	4th Address Line

Example

```
ADF Example Revised: Friday, November 13, 1998
OrCAD-02 Revision: A
OrCAD
9300 S.W. Nimbus Ave.

TURBO = ON
5C031

OPTIONS:TURBO = ON
PART:5C031

INPUTS:
CLOCK
ENABLE
COINDROP
CUPFULL
RESET

OUTPUTS:
STROBE
POURDRNK
DROPCUP

NETWORK:
J=INP(ENABLE) % SYM 1 %
N=INP(CUPFULL) % SYM 2 %
O=OR(P,Q) % SYM 3 %
POURDRNK,E=RORF(O,D,H,I,J) % SYM 4 %
Q=AND(F,R) % SYM 5 %
R=NOT(E) % SYM 6 %
B=XOR(E,F) % SYM 7 %
A=AND(B,C) % SYM 8 %
```

```
STROBE=CONF(A,VCC) % SYM 9 %
C=NOT(D) % SYM 10 %
D=INP(CLOCK) % SYM 11 %
H=AND(F,E) % SYM 12 %
I=INP(RESET) % SYM 13 %
G=AND(K,L,M) % SYM 14 %
DROPCUP,F=RORF(G,D,H,I,J) % SYM 15 %
M=INP(COINDROP) % SYM 16 %
K=NOT(F) % SYM 17 %
P=AND(K,E,L) % SYM 18 %
L=NOT(N) % SYM 19 %

EQUATIONS:
G = (K & L & M);
H = (F & E);
O = (P # Q);
END$
```

AppliconBRAVO netlist format

AppliconBRAVO netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.

Example

```
*** Desig 14DIP300
U1
*** Desig 14DIP300
U2
** NET GND
U1 7
U2 7
*** NET VCC
U1 14
U2 14
*** NET CLOCK
U1 10
*** NET Q
U1 6
```

```
U2 2
U1 9
*** NET OUT
U2 3
** NET B
U1 4
*** NET N00019
U1 3
U2 1
*** NET N00013
U1 5
U1 8
*** NET A
U1 1
U1 2
```

AppliconLEAP netlist format

AppliconLEAP netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.

  See the AppliconLEAP netlist format example for more information.

Example

```
*** NET GND
U1 7 14DIP300
U2 7 14DIP300
*** NET VCC
U1 14 14DIP300
U2 14 14DIP300
*** NET CLOCK
U1 10 14DIP300
*** NET Q
U1 6 14DIP300
U2 2 14DIP300
U1 9 14DIP300
*** NET OUT
U2 3 14DIP300
*** NET B
```

```
U1 4 14DIP300
*** NET N00019
U1 3 14DIP300
U2 1 14DIP300
** NET N00013
1 5 14DIP300
1 8 14DIP300
*** NET A
U1 1 14DIP300
U1 2 14DIP300
```

Cadnetix netlist format

Cadnetix netlists have the following characteristics:

- Part names can contain up to 17 characters.
- Module names can contain up to 15 characters.
- Reference strings plus pin numbers can contain up to 12 characters.
- Node names can contain up to 16 characters.
- Pin numbers can contain up to three digits.
- Pin names are not used.
- Node numbers are not checked for length.
- All ASCII characters are legal.

Example

```
PARTS LIST
74LS00 14DIP300 U1
74LS32 14DIP300 U2
EOS
NET LIST
NODENAME GND $
U1 7 U2 7
NODENAME VCC $
U1 14 U2 14
NODENAME CLOCK $
U1 10
NODENAME Q $
U1 6 U2 2 U1 9
NODENAME OUT $
U2 3
NODENAME B $
U1 4
NODENAME N00019 $
```

```
U1 3 U2 1
NODENAME N00013 $
U1 5 U1 8
NODENAME A $
U1 1 U1 2
EOS
```

Calay90 netlist format

The Calay 90 format creates two files: the netlist file and a component file. You must enter the component filename in the appropriate text box in the Create Netlist dialog box. Calay 90 netlists have the following characteristics:

- Part names, module names, and reference strings can each contain up to 19 characters.
- Node names can contain up to eight characters. Legal characters for node names are: + - 0..9 A..Z a..z
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- Pin numbers are not checked for length.
- All ASCII characters are legal except as noted for node names.

Example

```
GND U1('7) U2('7);
VCC U1('14) U2('14);
CLOCK U1('10);
Q U1('6) U2('2) U1('9);
OUT U2('3);
B U1('4);
N00019 U1('3) U2('1);
N00013 U1('5) U1('8);
A U1('1) U1('2);
```

Calay netlist format

This is the older of two Calay netlists formats. The newer Calay format is Calay 90.
Calay netlists have the following characteristics:

- Part names, module names, and reference strings can each contain up to 19 characters.
- Node names can contain up to eight characters. Legal characters for node names are: + - 0..9 A..Z a..z
- Node numbers are limited to five digits following the "N" prefix.

- Pin names are not used.
- Pin numbers are not checked for length.
- All ASCII characters are legal except as noted for node names.

Example

Calay netlists normally have a .NET file extension.

```
/GND U1(7) U2(7);
/VCC U1(14) U2(14);
/CLOCK U1(10);
/Q U1(6) U2(2) U1(9);
/OUT U2(3);
/B U1(4);
/N00019 U1(3) U2(1);
/N00013 U1(5) U1(8);
/A U1(1) U1(2);
```

Case netlist format

Sophia Systems & Technologies CASE netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal except as noted for node names.

Example

```
ASSERTIONS=OFF;VERSION=400;LOCATION=LOC;
[SIZE=1;TIMES=1;LOC=(U1);PLOC=U1;SHAPE=14DIP300]
1=A;
2=A;
3=N00019;
4=B;
5=N00013;
6=Q;
7=GND;
8=N00013;
9=Q;
10=CLOCK;
11=NC;
12=NC;
```

```
13=NC;
14=VCC;
;
[SIZE=1;TIMES=1;LOC=(U2);PLOC=U2;SHAPE=14DIP300]
1=N00019;
2=Q;
3=OUT;
4=NC;
5=NC;
6=NC;
7=GND;
8=NC;
9=NC;
10=NC;
11=NC;
12=NC;
13=NC;
14=VCC;
;
;
```

CBDS netlist format

BNR CBDS netlists have the following characteristics:

- Part names, module names, reference strings, and pin numbers are not checked for length.
- Node names can contain up to 20 characters. These characters are legal: / - 0..9 a..z A..Z
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal except as noted for node names.

Example

```
.SEARCH P,C
.DD U1 14DIP300
.DD U2 14DIP300
.S,GND,U1,7,U2,7
.S,VCC,U1,14,U2,14
.S,CLOCK,U1,10
.S,Q,U1,6,U2,2,U1,9
.S,OUT,U2,3
.S,B,U1,4
```

```
.S,N00019,U1,3,U2,1
.S,N00013,U1,5,U1,8
.S,A,U1,1,U1,2
```

Computervision netlist format

ComputerVision CADDS3 and CADDS4X netlists have the following characteristics:

- Part names, module names, reference strings, and pin numbers are not checked for length.
- Node names can contain up to 19 characters.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal except as noted for node names.

Example

```
0001 GND U1-7 U2-7
0002 VCC U1-14 U2-14
0003 CLOCK U1-10
0004 Q U1-6 U2-2 U1-9
0005 OUT U2-3
0006 B U1-4
0007 N00019 U1-3 U2-1
0008 N00013 U1-5 U1-8
0009 A U1-1 U1-2
```

DUMP netlist format

This format produces a flat netlist containing all the information on the schematic pages. No information is omitted or changed. You can use this netlist format when troubleshooting a design.

EDIF 2 0 0 netlist format

EEDesigner netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- Legal characters are:
  0..9 a..z A..Z _(underscore)
  Case is not significant. When Capture encounters an illegal character, it issues a warning

and makes the following changes:

- Changes "-" to "MINUS"
- Changes "+" to "PLUS"
- Changes "\" and "/" to "BAR"
- Changes all other illegal characters to "_"

EDIF 2 0 0 formats

Capture provides two EDIF netlist formats. The first format produces either hierarchical or flat netlist output, depending on your design structure and the active mode. It is accessible from the EDIF 2 0 0 tab in the Create Netlist dialog box. The second format produces only flat netlists, and is accessible through the Other tab in the Create Netlist dialog box.
Use the EDIF 2 0 0 tab if:

- You want to include net, pin, or part properties in the netlist.
- You want a hierarchical netlist.

  Use the Other tab if:
- You want a flat netlist for a simple hierarchical design.

Hierarchical designs in EDIF

Capture manages the hierarchy by defining pages in the schematic folder as CELLs in the main LIBRARY. These cells can then be referred to by INSTANCE where needed. Because EDIF requires a define-before-use philosophy, the hierarchy appears to be inverted in the netlist (the root schematic page is the last CELL in the main LIBRARY).

Note: Some of the options specific to the EDIF netlist format are included to support PC Board Layout Tools 386+. If you are creating a netlist for use with PCB 386+, be sure to select the Allow non-EDIF characters option.

Example flat netlist

```
(edif (rename &FIG_BMINUS01 "FIG_B-01")
(edifVersion 2 0 0)
(edifLevel 0)
(keywordMap (keywordLevel 0))
(status
(written
(timeStamp 0 0 0 0 0 0)
(program "EDIF.DLL")
(comment "Original data from OrCAD CAPTURE schematic"))
(comment "Generic Netlist Example")
(comment "Thursday, November 12, 1998")
(comment "OrCAD-01")
(comment "A")
(comment "OrCAD")
```

```
(comment "9300 S.W. Nimbus Ave.")
(comment "Beaverton, OR 97008")
(comment "(503) 671-9500 Corporate Offices")
(comment "(503) 671-9400 Technical Support"))
(external OrCAD_LIB
(edifLevel 0)
(technology
(numberDefinition
(scale 1 1 (unit distance))))
(cell &74LS00
(cellType generic)
(comment "From OrCAD library D:\ORCAD DEMO\CAPTURE\SDT\FIG_B-01.OLB")
(view NetlistView
(viewType netlist)
(interface
(port &1 (direction INPUT))
(port &2 (direction INPUT))
(port &3 (direction OUTPUT))
(port &4 (direction INPUT))
(port &5 (direction INPUT))
(port &6 (direction OUTPUT))
(port &7 (direction INPUT))
(port &8 (direction OUTPUT))
(port &9 (direction INPUT))
(port &10 (direction INPUT))
(port &11 (direction OUTPUT))
(port &12 (direction INPUT))
(port &13 (direction INPUT))
(port &14 (direction INPUT)))))
(cell &74LS32
(cellType generic)
(comment "From OrCAD library D:\ORCAD DEMO\CAPTURE\SDT\FIG_B-01.OLB")
(view NetlistView
(viewType netlist)
(interface
(port &1 (direction INPUT))
(port &2 (direction INPUT))
(port &3 (direction OUTPUT))
(port &4 (direction INPUT))
(port &5 (direction INPUT))
(port &6 (direction OUTPUT))
(port &7 (direction INPUT))
(port &8 (direction OUTPUT))
(port &9 (direction INPUT))
(port &10 (direction INPUT))
```

```
(port &11 (direction OUTPUT))
(port &12 (direction INPUT))
(port &13 (direction INPUT))
(port &14 (direction INPUT))))))
(library MAIN_LIB
(edifLevel 0)
(technology
(numberDefinition
(scale 1 1 (unit distance))))
(cell (rename &FIG_BMINUS01 "FIG_B-01")
(cellType generic)
(view NetlistView
(viewType netlist)
(interface
(port &CLOCK (direction INPUT))
(port &OUT (direction OUTPUT))
(port &B (direction INPUT))
(port &A (direction INPUT)))
(contents
(instance &U1
(viewRef NetlistView
(cellRef &74LS00
(libraryRef OrCAD_LIB)))
(property PartValue (string "74LS00"))
(property ModuleValue (string "14DIP300"))
(property TimeStampValue (string "6CB84CBA")))
(instance &U2
(viewRef NetlistView
(cellRef &74LS32
(libraryRef OrCAD_LIB)))
(property PartValue (string "74LS32"))
(property ModuleValue (string "14DIP300"))
(property TimeStampValue (string "6E46169D")))
(net &GND
(joined
(portRef &7 (instanceRef &U1))
(portRef &7 (instanceRef &U2))))
(net &VCC
(joined
(portRef &14 (instanceRef &U1))
(portRef &14 (instanceRef &U2))))
(net &CLOCK
(joined
(portRef &CLOCK)
(portRef &10 (instanceRef &U1))))
(net &Q
```

```
(joined
(portRef &6 (instanceRef &U1))
(portRef &2 (instanceRef &U2))
(portRef &9 (instanceRef &U1))))
(net &OUT
(joined
(portRef &OUT)
(portRef &3 (instanceRef &U2))))
(net &B
(joined
(portRef &B)
(portRef &4 (instanceRef &U1))))
(net &N00019
(joined
(portRef &3 (instanceRef &U1))
(portRef &1 (instanceRef &U2))))
(net &N00013
(joined
(portRef &5 (instanceRef &U1))
(portRef &8 (instanceRef &U1))))
(net &A
(joined
(portRef &A)
(portRef &1 (instanceRef &U1))
(portRef &2 (instanceRef &U1)))))))))
(design (rename &FIG_BMINUS01 "FIG_B-01")
(cellRef &FIG_BMINUS01
(libraryRef MAIN_LIB))))
```

Example hierarchical netlist

```
(edif FULLADD
(edifVersion 2 0 0)
(edifLevel 0)
(keywordMap (keywordLevel 0))
(status
(written
(timeStamp 1998 11 13 23 03 20)
(program "CAPTURE.EXE" (Version "9.00.1120"))
(comment "Original data from OrCAD/CAPTURE schematic"))
(comment "Hierarchy (Complex) Example")
(comment "Thursday, November 12, 1998")
(comment "OrCAD-06")
(comment "A")
(comment "OrCAD")
```

```
(comment "9300 S.W. Nimbus Ave.")
(comment "Beaverton, OR 97008")
(comment "(503)671-9500 Corporate Offices")
(comment "(503) 671-9400 Technical Support"))
(external OrCAD_LIB
(edifLevel 0)
(technology
(numberDefinition
(scale 1 1 (unit distance))))
(cell &74LS32
(cellType generic)
(comment "From OrCAD library FULLADD.OLB")
(view NetlistView
(viewType netlist)
(interface
(port &1 (direction INPUT))
(port &2 (direction INPUT))
(port &3 (direction OUTPUT))
(port &14 (direction INPUT))
(port &7 (direction INPUT))
(port &4 (direction INPUT))
(port &5 (direction INPUT))
(port &6 (direction OUTPUT))
(port &9 (direction INPUT))
(port &10 (direction INPUT))
(port &8 (direction OUTPUT))
(port &12 (direction INPUT))
(port &13 (direction INPUT))
(port &11 (direction OUTPUT)))))
(cell &74LS08
(cellType generic)
(comment "From OrCAD library FULLADD.OLB")
(view NetlistView
(viewType netlist)
(interface
(port &1 (direction INPUT))
(port &2 (direction INPUT))
(port &3 (direction OUTPUT))
(port &14 (direction INPUT))
(port &7 (direction INPUT))
(port &4 (direction INPUT))
(port &5 (direction INPUT))
(port &6 (direction OUTPUT))
(port &9 (direction INPUT))
(port &10 (direction INPUT))
(port &8 (direction OUTPUT))
```

```
(port &12 (direction INPUT))
(port &13 (direction INPUT))
(port &11 (direction OUTPUT)))))
(cell &74LS04
(cellType generic)
(comment "From OrCAD library FULLADD.OLB")
(view NetlistView
(viewType netlist)
(interface
(port &1 (direction INPUT))
(port &2 (direction OUTPUT))
(port &14 (direction INPUT))
(port &7 (direction INPUT))
(port &3 (direction INPUT))
(port &4 (direction OUTPUT))
(port &5 (direction INPUT))
(port &6 (direction OUTPUT))
(port &9 (direction INPUT))
(port &8 (direction OUTPUT))
(port &11 (direction INPUT))
(port &10 (direction OUTPUT))
(port &13 (direction INPUT))
(port &12 (direction OUTPUT))))))
(library MAIN_LIB
(edifLevel 0)
(technology
(numberDefinition
(scale 1 1 (unit distance))))
(cell EX6B
(cellType generic)
(view NetlistView
(viewType netlist)
(interface
(port X (direction INPUT))
(port Y (direction INPUT))
(port CARRY (direction OUTPUT))
(port SUM (direction OUTPUT)))
(contents
(instance U1
(viewRef NetlistView
(cellRef &74LS32
(libraryRef OrCAD_LIB))))
(instance U2
(viewRef NetlistView
(cellRef &74LS08
```

```
(libraryRef OrCAD_LIB))))
(instance U3
(viewRef NetlistView
(cellRef &74LS04
(libraryRef OrCAD_LIB))))
(net Y
(joined
(portRef &9 (instanceRef U2))
(portRef &3 (instanceRef U3))
(portRef &4 (instanceRef U2))
(portRef Y)))
(net CARRY
(joined
(portRef &8 (instanceRef U2))
(portRef CARRY)))
(net SUM
(joined
(portRef &6 (instanceRef U1))
(portRef SUM)))
(net X_BAR
(joined
(portRef &5 (instanceRef U2))
(portRef &2 (instanceRef U3))))
(net X
(joined
(portRef &10 (instanceRef U2))
(portRef &1 (instanceRef U3))
(portRef &1 (instanceRef U2))
(portRef X)))
(net N00037
(joined
(portRef &5 (instanceRef U1))
(portRef &6 (instanceRef U2))))
(net N00035
(joined
(portRef &3 (instanceRef U2))
(portRef &4 (instanceRef U1))))
(net GND
(joined
(portRef &7 (instanceRef U3))
(portRef &7 (instanceRef U2))
(portRef &7 (instanceRef U1))))
(net VCC
(joined
(portRef &14 (instanceRef U3))
(portRef &14 (instanceRef U2))
```

```
(portRef &14 (instanceRef U1))))
(net N5056796111
(joined
(portRef &4 (instanceRef U3))
(portRef &2 (instanceRef U2)))))))
(cell FULLADD
(cellType generic)
(view NetlistView
(viewType netlist)
(interface
(port SUM (direction OUTPUT))
(port X (direction INPUT))
(port Y (direction INPUT))
(port CARRY_OUT (direction OUTPUT))
(port CARRY_IN (direction INPUT)))
(contents
(instance U1
(viewRef NetlistView
(cellRef &74LS32
(libraryRef OrCAD_LIB))))
(instance halfadd_A
(viewRef NetlistView
(cellRef EX6B)))
(instance halfadd_B
(viewRef NetlistView
(cellRef EX6B)))
(net CARRY_IN
(joined
(portRef X (instanceRef halfadd_A))
(portRef CARRY_IN)))
(net SUM
(joined
(portRef SUM (instanceRef halfadd_A))
(portRef SUM)))
(net N00015
(joined
(portRef CARRY (instanceRef halfadd_A))
(portRef &1 (instanceRef U1))))
(net X
(joined
(portRef X (instanceRef halfadd_B))
(portRef X)))
(net N00013
(joined
(portRef Y (instanceRef halfadd_A))
```

```
(portRef SUM (instanceRef halfadd_B))))
(net Y
(joined
(portRef Y (instanceRef halfadd_B))
(portRef Y)))
(net N00025
(joined
(portRef CARRY (instanceRef halfadd_B))
(portRef &2 (instanceRef U1))))
(net CARRY_OUT
(joined
(portRef &3 (instanceRef U1))
(portRef CARRY_OUT)))
(net VCC
(joined
(portRef &14 (instanceRef U1))))
(net GND
(joined
(portRef &7 (instanceRef U1)))))))))
(design FULLADD
(cellRef FULLADD
(libraryRef MAIN_LIB))))
```

EEDesigner netlist format

EEDesigner netlists have the following characteristics:

- Part names, module names, and pin numbers are not checked for length.
- Reference strings are limited to eight characters.
- Node names are not supported.
- Node numbers are limited to three digits following the "UN" prefix.
- Pin names are not used.
- All ASCII characters are legal.

Example

```
(PATH,OrCAD()
(COMPONENTS
U1 ,14DIP300
U2 ,14DIP300
)
(NODES
(UN001
U1 , 7
```

```
U2 , 7
)
(UN002
U1 , 14
U2 , 14
)
(UN003
U1 , 10
)
(UN004
U1 , 6
U2 , 2
U1 , 9
)
(UN005
U2 , 3
)
(UN006
U1 , 4
)
(UN007
U1 , 3
U2 , 1
)
(UN008
U1 , 5
U1 , 8
)
(UN009
U1 , 1
U1 , 2
)
)
),OrCAD
```

Futurenet netlist format

FutureNet netlists have the following characteristics:

- Part names are limited to 16 characters.
- Module names, node names, and pin numbers are not checked for length.
- Reference strings are limited to six characters.
- Node numbers are limited to eight digits.

You can use Capture to generate FutureNet pinlists or netlists.

Pinlist example

```
PINLIST,2
(DRAWING,ORCAD.PIN,1-1
(SYM,1
DATA,2,U1
DATA,3,74LS00
DATA,4,14DIP300
PIN,,A,1-1,5,23,I0_A
PIN,,A,1-1,5,23,I1_A
PIN,,N00019,1-1,5,21,O_A
PIN,,B,1-1,5,23,I0_B
PIN,,N00013,1-1,5,23,I1_B
PIN,,Q,1-1,5,21,O_B
PIN,,GND,1-1,5,23,GND
PIN,,N00013,1-1,5,21,O_C
PIN,,Q,1-1,5,23,I0_C
PIN,,CLOCK,1-1,5,23,I1_C
PIN,,UN000001,1-1,5,21,O_D
PIN,,UN000002,1-1,5,23,I0_D
PIN,,UN000003,1-1,5,23,I1_D
PIN,,VCC,1-1,5,23,VCC
)
(SYM,2
DATA,2,U2
DATA,3,74LS32
DATA,4,14DIP300
PIN,,N00019,1-1,5,23,I0_A
PIN,,Q,1-1,5,23,I1_A
PIN,,OUT,1-1,5,21,O_A
PIN,,UN000004,1-1,5,23,I0_B
PIN,,UN000005,1-1,5,23,I1_B
PIN,,UN000006,1-1,5,21,O_B
PIN,,GND,1-1,5,23,GND
PIN,,UN000007,1-1,5,21,O_C
PIN,,UN000008,1-1,5,23,I0_C
PIN,,UN000009,1-1,5,23,I1_C
PIN,,UN000010,1-1,5,21,O_D
PIN,,UN000011,1-1,5,23,I0_D
PIN,,UN000012,1-1,5,23,I1_D
PIN,,VCC,1-1,5,23,VCC
)
SIG,GND,1-1,5,GND
SIG,VCC,1-1,5,VCC
SIG,CLOCK,1-1,5,CLOCK
```

```
SIG,Q,1-1,5,Q
SIG,OUT,1-1,5,OUT
SIG,B,1-1,5,B
SIG,N00019,1-1,5,N00019
SIG,N00013,1-1,5,N00013
SIG,A,1-1,5,A
)
```

Netlist example

```
NETLIST,2
(DRAWING,ORCAD.NET,1-1
DATA,50,Generic Netlist Example
DATA,51,OrCAD-01
DATA,52,A
DATA,54,Thursday, November 12, 1998
)
(SYM,1-1,1
DATA,2,U1
DATA,3,74LS00
DATA,4,14DIP300
DATA,23,I0_A
DATA,23,I1_A
DATA,21,O_A
DATA,23,I0_B
DATA,23,I1_B
DATA,21,O_B
DATA,23,GND
DATA,21,O_C
DATA,23,I0_C
DATA,23,I1_C
DATA,21,O_D
DATA,23,I0_D
DATA,23,I1_D
DATA,23,VCC
)
(SYM,1-1,2
DATA,2,U2
DATA,3,74LS32
DATA,4,14DIP300
DATA,23,I0_A
DATA,23,I1_A
DATA,21,O_A
DATA,23,I0_B
DATA,23,I1_B
```

```
DATA,21,O_B
DATA,23,GND
DATA,21,O_C
DATA,23,I0_C
DATA,23,I1_C
DATA,21,O_D
DATA,23,I0_D
DATA,23,I1_D
DATA,23,VCC
)
(SIG,,GND,1-1,5,GND
PIN,1-1,1,U1,23,GND
PIN,1-1,2,U2,23,GND
)
(SIG,,VCC,1-1,5,VCC
PIN,1-1,1,U1,23,VCC
PIN,1-1,2,U2,23,VCC
)
(SIG,,CLOCK,1-1,5,CLOCK
PIN,1-1,1,U1,23,I1_C
)
(SIG,,Q,1-1,5,Q
PIN,1-1,1,U1,21,O_B
PIN,1-1,2,U2,23,I1_A
PIN,1-1,1,U1,23,I0_C
)
(SIG,,OUT,1-1,5,OUT
PIN,1-1,2,U2,21,O_A
)
(SIG,,B,1-1,5,B
PIN,1-1,1,U1,23,I0_B
)
(SIG,,N00019,1-1,5,N00019
PIN,1-1,1,U1,21,O_A
PIN,1-1,2,U2,23,I0_A
)
(SIG,,N00013,1-1,5,N00013
PIN,1-1,1,U1,23,I1_B
PIN,1-1,1,U1,21,O_C
)
(SIG,,A,1-1,5,A
PIN,1-1,1,U1,23,I0_A
PIN,1-1,1,U1,23,I1_A
)
```

HiLo netlist format

HiLo netlists have the following characteristics:

- Part names, module names, reference strings, and pin numbers are not checked for length.
- Node names are limited to 14 characters.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.

Example

```
** Generic Netlist Example Revised: Thursday, November 12, 1998
** OrCAD-01 Revision: A
** OrCAD
** 9300 S.W. Nimbus Ave.
** Beaverton, OR 97008
** (503) 671-9500 Corporate Offices
** (503) 671-9400 Technical Support
CCT ORCAD (
** Please put your circuit interface definition here
);
14DIP300
U1 (
A,
A,
N00019,
B,
N00013,
Q,
GND,
N00013,
Q,
CLOCK,
,
,
,
VCC
);
14DIP300
U2 (
N00019,
Q,
```

```
OUT,
,
,
,
GND,
,
,
,
,
,
,
VCC
);
```

Intel ADF netlist format

Intel ADF netlists have the following characteristics:

- Part names, module names, reference strings, and pin numbers are not checked for length.
- Node numbers are not used.
- All ASCII characters are legal.

Intel ADF netlist constraints

When you create an Intel ADF netlist, you must include the OrCAD-supplied ALTERA_P.OLB and ALTERA_M.OLB libraries in your project. You can use only the parts in these two libraries to create the schematic folder.
Inputs and outputs are handled differently in Capture than in the Altera software. Capture defines inputs and outputs with hierarchical ports and library objects. Altera defines inputs and outputs with a library object, which is then tagged with the appropriate pin number. In the example schematic page, the CLOCK signal is an input and the STROBE signal is an output.
Also, library objects with unused pins default to predefined levels in the Altera software. Because Capture does not default unconnected pins to any particular level, you must tie all unused pins to the appropriate level.

Intel ADF pipe commands

You can place equations in your schematic folder to be included in the netlist. To place these equations on the schematic page, choose the Text command from the Place menu.
Each equation must start with the pipe character (|). The first line must be:

```
|EQUATIONS
```

This tells Capture that some Intel ADF equations need to be included in the netlist. The equations can contain any information you want to include in the netlist.

Intel ADF title block information

Title block information is placed in the first 10 lines of the netlist. The following table shows an example netlist header and the title block information from which the header was extracted. Header information in bold is text entered in the schematic page's title block

.

Line	Example header	Title block field
1	ADF Example	Title of schematic page
1	May 15, 2002	Date
2	OrCAD-03	Document number
2	D	Revision Code
3	Dade’s House of Boards	Organization Name
4	933 SW 52nd St.	1st Address Line
6	Turbo=ON	3rd Address Line
7	5C031	4th Address Line

Example

```
ADF Example Revised: Friday, November 13, 1998
OrCAD-02 Revision: A
OrCAD
9300 S.W. Nimbus Ave.
TURBO = ON
5C031
OPTIONS:TURBO = ON
PART:5C031
INPUTS:
CLOCK
ENABLE
COINDROP
CUPFULL
```

```
RESET
OUTPUTS:
STROBE
POURDRNK
DROPCUP
NETWORK:
J=INP(ENABLE) % SYM 1 %
N=INP(CUPFULL) % SYM 2 %
O=OR(P,Q) % SYM 3 %
POURDRNK,E=RORF(O,D,H,I,J) % SYM 4 %
Q=AND(F,R) % SYM 5 %
R=NOT(E) % SYM 6 %
B=XOR(E,F) % SYM 7 %
A=AND(B,C) % SYM 8 %
STROBE=CONF(A,VCC) % SYM 9 %
C=NOT(D) % SYM 10 %
D=INP(CLOCK) % SYM 11 %
H=AND(F,E) % SYM 12 %
I=INP(RESET) % SYM 13 %
G=AND(K,L,M) % SYM 14 %
DROPCUP,F=RORF(G,D,H,I,J) % SYM 15 %
M=INP(COINDROP) % SYM 16 %
K=NOT(F) % SYM 17 %
P=AND(K,E,L) % SYM 18 %
L=NOT(N) % SYM 19 %
EQUATIONS:
G = (K & L & M);
H = (F & E);
O = (P # Q);
END$
```

Intergraph netlist format

Intel ADF netlists have the following characteristics:

- Part names, module names, reference strings, and pin numbers are not checked for length.
- Node numbers can have up to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.

Example

```
%PART
```

```
14DIP300 U1
14DIP300 U2
%NET
GND U1-7 U2-7
VCC U1-14 U2-14
CLOCK U1-10
Q U1-6 U2-2 U1
OUT U2-3
B U1-4
N00019 U1-3 U2-1
N00013 U1-5 U1-8
A U1-1 U1-2
$
```

Mentor netlist format

Mentor Graphics BoardStation Version 7 netlists have the following characteristics:

- Part names, module names, and reference strings are limited to nineteen characters.
- Node names and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.

Note: Capture includes a netlister (available from the Accessories menu) developed specifically for Mentor netlist generation. There is also a document that discusses the use of the netlister. It is available in the Vendor directory of your Capture installation.

Example component file

```
# OrCAD Formatted Netlist for MENTOR Board Station V6
# Reference Value Field Module Field
U1 PART 74LS00 14DIP300
U2 PART 74LS32 14DIP300
```

Example netlist

```
NET 'GND' U1-7 U2-7
NET 'VCC' U1-14 U2-14
NET 'CLOCK' U1-10
NET 'Q' U1-6 U2-2 U1-9
NET 'OUT' U2-3
NET 'B' U1-4
NET 'N00019' U1-3 U2-1
```

```
NET 'N00013' U1-5 U1-8
NET 'A' U1-1 U1-2
```

Multiwire netlist format

MultiWire netlists have the following characteristics:

- Part names and module names are not checked for length.
- Reference strings and pin numbers together are limited to thirty-two characters.
- Node names are limited to sixteen characters.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.

Example

```
GND U1 7
GND U2 7
VCC U1 14
VCC U2 14
CLOCK U1 10
Q U1 6
Q U2 2
Q U1 9
OUT U2 3
B U1 4
N00019 U1 3
N00019 U2 1
N00013 U1 5
N00013 U1 8
A U1 1
A U1 2
-1
```

OHDL netlist format

OrCAD PLD 386+ netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.

- All ASCII characters are legal.
  For more information, see the OrCAD web site.

OHDL netlist constraints

The OHDL netlist format uses the OrCAD-supplied PLDGATES.OLB and TTL.OLB libraries. Be sure you include one of these libraries in your project.

Example

OHDL netlists normally have a .PLD file extension.

```
|| CNTMUX (Example for MACH 110) Revised: Friday, November 13, 1998
|| D:\ORCAD DEMO\CAPTURE\NETLIST UPDATES\DESIGN6\Revision: ?
|| OrCAD
||
||
||
||
||
| Type: "IFX780_132"
|
|
| Netlist:
| {
| PAD (I0,"IN") | PAD1
| PAD (I6,"IN") | PAD10
| PAD (O3,"OUT") | PAD11
| PAD (I7,"IN") | PAD12
| PAD (O4,"OUT") | PAD13
| PAD (I8,"IN") | PAD14
| PAD (O5,"OUT") | PAD15
| PAD (I9,"IN") | PAD16
| PAD (O6,"OUT") | PAD17
| PAD (I10,"IN") | PAD18
| PAD (O7,"OUT") | PAD19
| PAD (I1,"IN") | PAD2
| PAD (I11,"IN") | PAD20
| PAD (O8,"OUT") | PAD21
| PAD (I12,"IN") | PAD22
| PAD (O9,"OUT") | PAD23
| PAD (O10,"OUT") | PAD24
| PAD (I14,"IN") | PAD25
| PAD (O11,"OUT") | PAD26
| PAD (I15,"IN") | PAD27
| PAD (O12,"OUT") | PAD28
```

```
| PAD (O13,"OUT") | PAD29
| PAD (I2,"IN") | PAD3
| PAD (LOAD,"IN") | PAD30
| PAD (O14,"OUT") | PAD31
| PAD (O15,"OUT") | PAD32
| PAD (CLK,"IN") | PAD33
| PAD (UP,"IN") | PAD34
| PAD (COUNT,"IN") | PAD35
| PAD (SELECT,"IN") | PAD36
| PAD (I3,"IN") | PAD4
| PAD (O0,"OUT") | PAD5
| PAD (I4,"IN") | PAD6
| PAD (O1,"OUT") | PAD7
| PAD (I5,"IN") | PAD8
| PAD (O2,"OUT") | PAD9
| G169 (UP,CLK,I0,I1,I2,I3,GND,-,N00185,N00193,Q3,Q2,Q1,Q0,N00177) | U
| G257 (SELECT,Q12,I12,O12,Q13,I13,O13,-,O14,I14,Q14,O15,I15,Q15,GND)
| G257 (SELECT,Q0,I0,O0,Q1,I1,O1,-,O2,I2,Q2,O3,I3,Q3,GND) | U2
| G169 (UP,CLK,I4,I5,I6,I7,GND,-,N00185,N00177,Q7,Q6,Q5,Q4,-) | U3
| G04 (LOAD,N00185) | U4
| G257 (SELECT,Q4,I4,O4,Q5,I5,O5,-,O6,I6,Q6,O7,I7,Q7,GND) | U5
| G169 (UP,CLK,I8,I9,I10,I11,GND,-,N00185,N00193,Q11,Q10,Q9,Q8,N00355)
| G04 (COUNT,N00193) | U7
| G257 (SELECT,Q8,I8,O8,Q9,I9,O9,-,O10,I10,Q10,O11,I11,Q11,GND) | U8
| G169 (UP,CLK,I12,I13,I14,I15,GND,-,N00185,N00355,Q15,Q14,Q13,Q12,-)
| }
|
|Vectors:
|{ Display COUNT, LOAD, SELECT, CLK, \
| (I[15..8])d, (O[15..8])d, \
| (I[7..0])d, (O[7..0])d
|
| Test LOAD=1; CLK
| Test LOAD=0; COUNT=1; UP=1; CLK=25(0,1)
| Set I[15..8] = 10
| Set I[7..0] = 11
| Test SELECT=1,0
| Test LOAD=0; COUNT=1; UP=0; CLK=25(0,1)
| Test SELECT=1,0
| End }
```

PADS 2000 netlist format

PADS 2000 netlists have the following characteristics:

- Part names, module names, and pin numbers are not checked for length.
- Reference strings are limited to sixteen characters.
- Header information is included at the top of the netlist file:

```
!PADS-POWERPCB-V2
```

- Net and signal names are limited to forty-seven characters.
- Legal characters for reference strings and node names are limited to:
  - ~ ! # $ % _ - =
  - + | / . : ; < >
  - A..Z a..z 0..9
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal except as noted for reference strings and node names.

Note: You can add a property called "tracewidth" to nets in Capture. The value of tracewidth will carry through into the PADS 2000 netlist. Capitalization of this property is important, and the property won't appear in the netlist if any uppercase letters are used in the property name.

For more information see the PADS website.

Example

This netlist was created with no options selected. PADS 2000 netlists normally have a .ASC file extension.
The header information in this example was created without the Create Pads BGA netlist option selected. If you choose this option when you create the PADS netlist, the header information will appear differently. In effect, choosing the Create Pads BGA netlist option causes Capture to generate a Powerpcb v3.0 netlist.

```
*PADS 2000*
*PART*
U1 14DIP300
U2 14DIP300
*NET*
*SIGNAL* GND
U1.7 U2.7
*SIGNAL* VCC
U1.14 U2.14
*SIGNAL* CLOCK
U1.10
*SIGNAL* Q
```

```
U1.6 U2.2 U1.9
*SIGNAL* OUT
U2.3
*SIGNAL* B
U1.4
*SIGNAL* N00019
U1.3 U2.1
*SIGNAL* N00013
U1.5 U1.8
*SIGNAL* A
U1.1 U1.2
*END*
```

PADS PCB netlist format

PADS-Software PADS PowerPCB netlists have the following characteristics:

- Part names, module names, and pin numbers are not checked for length.
- Reference strings are limited to sixteen characters.
- Header information is included at the top of the netlist file:

```
!PADS-POWERPCB-V2
```

- Net and signal names are limited to twelve characters.
- Legal characters for reference strings and node names are limited to:
  - ~ ! # $ % _ - =
  - + | / . : ; < >
  - A..Z a..z 0..9
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal except as noted for reference strings and node names.

Example

This netlist was created with no options selected. PADS-PCB netlists normally have a .ASC file extension.
The header information in this example was created without the Create Pads BGA netlist option selected. If you choose this option when you create the PADS netlist, the header information will appear differently. In effect, choosing the Create Pads BGA netlist option causes Capture to generate a Powerpcb v3.0 netlist.

```
*PADS-PCB*
*PART*
```

```
U1 14DIP300
U2 14DIP300
*NET*
*SIGNAL* GND
U1.7 U2.7
*SIGNAL* VCC
U1.14 U2.14
*SIGNAL* CLOCK
U1.10
*SIGNAL* Q
U1.6 U2.2 U1.9
*SIGNAL* OUT
U2.3
*SIGNAL* B
U1.4
*SIGNAL* N00019
U1.3 U2.1
*SIGNAL* N00013
U1.5 U1.8
*SIGNAL* A
U1.1 U1.2
*END*
```

PCAD netlist format

PCAD PCB netlists from ACCEL Technologies have the following characteristics:

- Part names, module names, reference strings, and pin numbers are not checked for length.
- Node names are limited to eight characters.
- Node numbers are limited to five digits following the "NET" prefix.
- Pin names are not used.
- Characters are not checked for legality.

  For more information, see the ACCEL Technologies web site, the Protel web site, and the netlist example.

Example

This netlist was created with no options selected. PCAD netlists normally have a .NET file extension.

```
{COMPONENT ORCAD.PCB
 {ENVIRONMENT LAYS.PCB}
  {PDIFvrev 1.30}
```

```
  {DETAIL
  {SUBCOMP
{I 14DIP300.PRT U1
{CN
  1 A
  2 A
  3 N00019
  4 B
  5 N00013
  6 Q
  7 GND
  8 N00013
  9 Q
  10 CLOCK
  11 ?
  12 ?
  13 ?
  14 VCC
}
}
{I 14DIP300.PRT U2
{CN
  1 N00019
  2 Q
  3 OUT
  4 ?
  5 ?
  6 ?
  7 GND
  8 ?
  9 ?
  10 ?
  11 ?
  12 ?
  13 ?
  14 VCC
}
}
}
}
}
```

PCADnlt netlist format

PCADnlt netlists from ACCEL Technologies have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Legal characters for node names are limited to:

```
$ - + _ (underscore)
A..Z a..z 0..9
```

- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal except as noted for node names.
  For more information, see the ACCEL Technologies web site, the Protel web site.

Example

PCADnlt netlists normally have a .NET file extension.

```
% Generic Netlist Example Revised: Thursday, November 12, 1998
% OrCAD-01 Revision: A
% OrCAD
% 9300 S.W. Nimbus Ave.
% Beaverton, OR 97008
% (503) 671-9500 Corporate Offices
% (503) 671-9400 Technical Support
BOARD = ORCAD.PCB;

PARTS
14DIP300 = U1, % 74LS00
  U2; % 74LS32

NETS

GND = U1/7 U2/7 ;
VCC = U1/14 U2/14 ;
CLOCK = U1/10 ;
Q = U1/6 U2/2 U1/9 ;
OUT = U2/3 ;
B = U1/4 ;
N00019 = U1/3 U2/1 ;
N00013 = U1/5 U1/8 ;
A = U1/1 U1/2 ;
```

PCBII and PCBIIL netlist formats

PCB netlists are used with OrCAD's PCB II Layout Tools. See the PCB II User's Guide for details. The PCBII and PCBIIL netlist formats are identical with the following exception: the PCBIIL.DLL netlist format has no restrictions on netname length.
PCB netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- All ASCII characters are legal.
- Footprint names are limited to eight characters.
- PCBII (but not PCBIIL) net names are limited to eight characters.

For more information, see the OrCAD web site.

PDUMP netlist format

This format produces a parts list containing all the information on the schematic pages. No information is omitted or changed. You can use this netlist format when troubleshooting a project.

PLD netlist format

This file produces netlists that define logic for use with Programmable Logic Design Tools 386+. See the Programmable Logic Design Tools User's Guide and the Programmable Logic Design Tools Reference Guide for details.
PLD netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Pin names are not used.
- All ASCII characters are legal.

For more information, see the OrCAD web site.

PLD netlist constraints

When you create a PLD netlist, be sure to include the OrCAD-supplied PLDGATES.OLB library in your project. You can use only the parts in PLDGATES.OLB, DEVICE.OLB (VCC, POWER, GND), and TTL.OLB (most 74LSxx) in a schematic folder to be netlisted for PLD.

Example

This netlist was created with no options selected. PLD netlists normally have a .NET file extension.

```
|| PLD Netlist Example Revised: Friday, November 13, 1998
|| OrCAD-05 Revision: A
```

```
|| OrCAD
|| 9300 S.W. Nimbus Ave.
|| Beaverton, OR 97008
|| (503) 671-9500 Corporate Offices
|| (503) 671-9400 Technical Support
||
| Netlist: A0,A1,B1,B0
| ->
| Y0,Y3,Y1,Y2
| {
| G08 (B0,A0,Y0) | U1
| G32 (N00103,N00107,N00095) | U10
| G04 (B1,N00113) | U11
| G11 (N00113,B0,-,-,-,-,-,-,-,-,-,N00107,A1) | U12
| G11 (B1,A1,-,-,-,-,-,-,-,-,-,N00133,N00137) | U13
| G04 (A0,N00137) | U14
| G32 (N00133,N00145,Y2) | U15
| G04 (B0,N00151) | U16
| G11 (B1,N00151,-,-,-,-,-,-,-,-,-,N00145,A1) | U17
| G21 (A0,A1,-,B0,B1,Y3) | U18
| G04 (A0,N00063) | U2
| G11 (N00063,B0,-,-,-,-,-,-,-,-,-,N00069,A1) | U3
| G32 (N00069,N00083,N00087) | U4
| G04 (A1,N00081) | U5
| G11 (B1,N00081,-,-,-,-,-,-,-,-,-,N00083,A0) | U6
| G32 (N00087,N00095,Y1) | U7
| G04 (B0,N00101) | U8
| G11 (B1,N00101,-,-,-,-,-,-,-,-,-,N00103,A0) | U9
| }
```

Protel2 netlist format

Protel2 netlists have the following characteristics:

- Part names, module names, reference strings, and node names can be up to 16 characters in length.
- Node numbers are limited to 5 digits (plus the leading 'N'.
- Pin numbers are not checked for length.
- The Reference and ModuleName must be in uppercase only.
- All ASCII characters are legal except { '()[],-'}.
  For more information, see the Protel web site.

Example

Accel netlists normally have a .NET file extension.
This example was created with the combined property strings for Create Netlist set to their default values, as shown below:

```
]
[
DESIGNATOR
C10
FOOTPRINT
SM/C_0805
PARTTYPE
390PF
DESCRIPTION
]
[
DESIGNATOR
C100
FOOTPRINT
0.1UF
PARTTYPE
0.1UF
DESCRIPTION
]
[
DESIGNATOR
C101
FOOTPRINT
0.1UF
PARTTYPE
0.1UF
DESCRIPTION
```

RecalRedac netlist format

The newer version of the RacalRedac netlist format is RINF.
Zuken-Redac CADStar PCB netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.
  For more information, see the Zuken-Redac web site at http://www.zuken.com/.

Example

RacalRedac netlists normally have a .NET file extension.

```
.PCB
.REM Generic Netlist Example Revised: Thursday, November 12, 1998
.REM OrCAD-01 Revision: A
.REM OrCAD
.REM 9300 S.W. Nimbus Ave.
.REM Beaverton, OR 97008
.REM (503) 671-9500 Corporate Offices
.REM (503) 671-9400 Technical Support
.CON
.COD 2

.REM GND
U1 7 U2 7
.REM VCC
U1 14 U2 14
.REM CLOCK
U1 10
.REM Q
U1 6 U2 2 U1 9
.REM OUT
U2 3
.REM B
U1 4
.REM N00019
U1 3 U2 1
.REM N00013
U1 5 U1 8
.REM A
U1 1 U1 2
.EOD
```

RINF netlist format

Zuken-Redac Visual PCB netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.

Example

RINF netlists normally have a .NET file extension.

```
 .HEA
.APP "Cadstar RINF Output - Version 2.3"
.UNI INCH 1000.0 in
.TYP FULL
.JOB "FIG_B-01"

.ADD_COM U1 "14DIP300"
.ADD_COM U2 "14DIP300"

.ADD_TER U1 7 "GND"
.TER U2 7

.ADD_TER U1 14 "VCC"
.TER U2 14

.ADD_TER U1 10 "CLOCK"

.ADD_TER U1 6 "Q"
.TER U2 2
  U1 9

.ADD_TER U2 3 "OUT"

.ADD_TER U1 4 "B"

.ADD_TER U1 3 "N00019"
.TER U2 1

.ADD_TER U1 5 "N00013"
.TER U1 8

.ADD_TER U1 1 "A"
.TER U1 2

.END
```

Scicards netlist format

Harris EDA SciCards netlists have the following characteristics:

- Part names are limited to seventeen characters.
- Module names are limited to fifteen characters.
- Reference strings and pin numbers combined are limited to twelve characters.
- Pin numbers are limited to three characters.
- Node names are limited to eight characters.
- Node numbers are not checked for length.
- Pin names are not used.
- All ASCII characters are legal.

  For more information, see the netlist example.

Example

Scicards netlists normally have a .NET file extension.

```
PARTS LIST
74LS00 14DIP300 U1
74LS32 14DIP300 U2
EOS
NET LIST

NODENAME GND $
  U1 7 U2 7
NODENAME VCC $
  U1 14 U2 14
NODENAME CLOCK $
  U1 10
NODENAME Q $
  U1 6 U2 2 U1 9
NODENAME OUT $
  U2 3
NODENAME B $
  U1 4
NODENAME N00019 $
  U1 3 U2 1
NODENAME N00013 $
  U1 5 U1 8
NODENAME A $
  U1 1 U1 2
EOS
```

Tango netlist format

ACCEL Technologies Tango PCB and Tango PRO netlists have the following characteristics:

- Part names, module names, reference strings, and node names are limited to sixteen characters.
- Node numbers are limited to five digits following the "N" prefix.
- Pin numbers are not checked for length.
- Pin names are not used.
- Reference strings and module names must be uppercase characters.
- All ASCII characters are legal except:

```
( ) [ ] - (dash) , (comma)
```

and as noted for reference strings and module names.
For more information, see the ACCEL Technologies web site at http://www.techaccel.com and the Protel web site at http://www.protel.com, as well as the Tango netlist example.

Telesis netlist format

Cadence Telesis netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.
  For more information, see the OrCAD web site as well as the Telesis netlist example.

Example

Telesis netlists normally have a .NET file extension.

```
$PACKAGES
14DIP300! 74LS00; U1
14DIP300! 74LS32; U2
$NETS
GND; U1.7 U2.7
VCC; U1.14 U2.14
CLOCK; U1.10
Q; U1.6 U2.2 U1.9
OUT; U2.3
B; U1.4
N00019; U1.3 U2.1
N00013; U1.5 U1.8
A; U1.1 U1.2
$END
```

Vectron netlist format

Vectron netlists have the following characteristics:

- Part names, module names, and pin numbers are not checked for length.
- Reference strings are limited to eight characters.
- Node names are limited to twelve characters.
- Node numbers are limited to five digits following the "N" prefix.
- Pin names are not used.
- All ASCII characters are legal.

  In addition to the netlist file, Capture also creates a part list file when you select the Vectron netlist format. You must enter a second filename in the Destination 2 text box on the Netlist Format dialog box.

Example netlist

Vectron netlists normally have a .NET file extension.

```
*GND U1 7 U2 7
*VCC U1 14 U2 14
*CLOCK U1 10
*Q U1 6 U2 2 U1 9
*OUT U2 3
*B U1 4
*N00019 U1 3 U2 1
*N00013 U1 5 U1 8
*A U1 1 U1 2
```

Example part list

```
U1 14DIP300
U2 14DIP300
```

Verilog netlist format

Verilog netlists have the following characteristics:

- Part identifiers, module identifiers, reference strings, node identifiers, and pin numbers are not checked for length.
- Part identifiers, module identifiers, reference strings, node identifiers, and pin numbers must begin with a letter.

- Part identifiers, module identifiers, reference strings, node identifiers, and pin numbers must all be unique. That is, none of these can share a name.
- Legal characters for part identifiers, module identifiers, reference strings, node identifiers, and pin numbers are limited to:

```
A..Z a..z 0..9
```

If there are illegal characters in a part identifier, module identifier, reference string, node identifier, or pin number, the netlister converts them to legal Verilog names. This conversion uses the backslash character (\) to escape otherwise illegal characters. For spaces, the conversion uses the ASCII equivalent (#20).
Consider these examples:

This string....	converts to this...
$\overline{\text{RAS}}$	\R\A\S\
A B	\A#20B
AggB	\A#20#20B
A#B	\A#B
A#20B	\A#2320B
A#23B	\A#2323B

For more information, see the Verilog netlist example.

Note: In cases where a net name is different from a port name, Capture uses aliases to associate the two. That is, if a wire with one net name is connected to a port with a different name, Capture creates an alias to associate the two components to the same net.

The alias takes either of the following forms:

alias_bit alias_inst1(NetName, PortName)
alias_vector alias_inst1(NetName, PortName)

Where:

NetName is the name assigned to the net.
PortName is the name assigned to the port.

If these aliases are used, they will appear at the beginning of the netlist.

Assigning a Verilog parameter to a component instance

Identifiers for parts, modules, part references, nodes, pins, and nets must not conflict with any Verilog reserved word. See Verilog reserved words for a list of reserved words.
You can specify Verilog parameters on component instances as properties, using this method:

1. Assign the property Vlog_param to the component, using the following syntax:

```
Vlog_Param = Parameter_name:Parameter_type
```

Where:

Parameter_name is the name of the parameter to be specified in the netlist.
Parameter_type is the type of the parameter (for example, "integer," or "string").

Note: You can specify multiple Verilog parameters on a component instance, as well, by using the following format:

Vlog_ParamXX = Parameter_name:Parameter_type

Where:

XX is an integer that is unique to the parameter being defined.

1. Assign a value to the declared parameter:

Note: Parameter_name = Parameter_value

Where:

Parameter_name is the name of the parameter to be specified in the netlist.
Parameter_value is the value of the parameter.

The parameter will appear in the netlist as:

```
\7400 U7(
```

```
  .A_A( IN1 ) ,
  .B_A( IN2 ) ,
  .Y_A( OUT ) ,
  .VCC( VCC ) ,
  .GND( GND )
) ;
defparam U7.Parameter_name = Parameter_value;
```

If the parameter value is a string, the netlister encloses it in quotes ("") in the netlist. If a parameter does not have a value, the netlister will report an error.

Support of global signals and creation of global module

The Verilog netlister connects to global signals using the global module "glbl". This distinguishes global signals from local signals. For example:

```
//Verilog global signals module
module glbl() ;
wire global;
...
endmodule

module schematic1() ;
...
wire global; //local alias for the global signal "global"
assign global = glbl.global;
...
ls04 i1(
.a(global),
...);
...
endmodule
```

By default, only power signals are considered global. Capture inserts the global module at the top of the Verilog netlist.
If your design is a PSpice A/DV design, Capture places the connections to the global signals under `ifdef VAN. This is so that the normal Verilog simulation will not get affected. For example:

```
`ifdef VAN
module glbl;
wire global;
...
endmodule
`endif
module schematic1;
```

```
...
`ifdef VAN
wire global; //local alias for the global signal "global"
assign global = glbl.global;
`endif
...
ls04 i1(
`ifdef VAN
.a(glbl.global),
`else
.a(global),
`endif
...);
...
Endmodule
```

Verilog netlists normally have a .V file extension.

Example Verilog netlist with power pins included

```
`timescale 1ns/1ps
module alias_vector (a, a);
parameter size = 1;
inout [size-1:0] a;
endmodule
module alias_bit (a, a);
inout a;
endmodule
module glbl;
  wire VCC;
  wire GND;
endmodule
module HALFADD ( X, Y, CARRY, SUM);
input X;
input Y;
output CARRY;
output SUM;
// SIGNALS
wire VCC;
assign VCC = glbl.VCC;
wire GND;
assign GND = glbl.GND;
wire N00032;
wire X_BAR;
wire N00034;
```

```
wire N5056796111;
// GATE INSTANCES
\74LS32 U1(
  .I0_B( N00032 ) ,
  .I1_B( N00034 ) ,
  .O_B( SUM )
 ) ;
\74LS08 U2(
  .I0_A( X ) ,
  .I1_A( N5056796111 ) ,
  .O_A( N00032 ) ,
  .VCC( VCC ) ,
  .GND( GND ) ,
  .I0_B( Y ) ,
  .I1_B( X_BAR ) ,
  .O_B( N00034 ) ,
  .I0_C( Y ) ,
  .I1_C( X ) ,
  .O_C( CARRY )
) ;
\74LS04 U3(
  .I_A( X ) ,
  .O_A( X_BAR ) ,
  .VCC( VCC ) ,
  .GND( GND ) ,
  .I_B( Y ) ,
  .O_B( N5056796111 ) ) ;
endmodule
module FULLADD ( SUM, X, Y, CARRY_OUT, CARRY_IN);
output SUM;
input X;
input Y;
output CARRY_OUT;
input CARRY_IN;
// SIGNALS
wire VCC;
assign VCC = glbl.VCC;
wire GND;
assign GND = glbl.GND;
wire N00011;
wire N00013;
wire N00023;
// GATE INSTANCES
\74LS32 U1(
  .I0_A( N00013 ) ,
  .I1_A( N00023 ) ,
```

```
  .O_A( CARRY_OUT ) ,
  .VCC( VCC ) ,
  .GND( GND )
 ) ;
HALFADD HALFADD_A (
  .X( CARRY_IN ) ,
  .Y( N00011 ) ,
  .CARRY( N00013 ) ,
  .SUM( SUM )
 ) ;
HALFADD HALFADD_B (
  .X( X ) ,
  .Y( Y ) ,
  .CARRY( N00023 ) ,
  .SUM( N00011 )
 ) ;
endmodule
```

Example Verilog netlist without power pins included

```
`timescale 1ns/1ps
module alias_vector (a, a);
parameter size = 1;
inout [size-1:0] a;
endmodule
module alias_bit (a, a);
inout a;
endmodule
module glbl;
endmodule
module HALFADD ( X, Y, CARRY, SUM);
input X;
input Y;
output CARRY;
output SUM;
// SIGNALS
wire N00032;
wire X_BAR;
wire N00034;
wire N5056796111;
// GATE INSTANCES
\74LS32 U1(
  .I0_B( N00032 ) ,
  .I1_B( N00034 ) ,
  .O_B( SUM )
```

```
) ;
\74LS08 U2(
  .I0_A( X ) ,
  .I1_A( N5056796111 ) ,
  .O_A( N00032 ) ,
  .I0_B( Y ) ,
  .I1_B( X_BAR ) ,
  .O_B( N00034 ) ,
  .I0_C( Y ) ,
  .I1_C( X ) ,
  .O_C( CARRY )
 ) ;
\74LS04 U3(
  .I_A( X ) ,
  .O_A( X_BAR ) ,
  .I_B( Y ) ,
  .O_B( N5056796111 )
 ) ;
endmodule
module FULLADD ( SUM, X, Y, CARRY_OUT, CARRY_IN);
output SUM;
input X;
input Y;
output CARRY_OUT;
input CARRY_IN;
// SIGNALS
wire N00011;
wire N00013;
wire N00023;
// GATE INSTANCES
\74LS32 U1(
  .I0_A( N00013 ) ,
  .I1_A( N00023 ) ,
  .O_A( CARRY_OUT )
 ) ;
HALFADD HALFADD_A (
  .X( CARRY_IN ) ,
  .Y( N00011 ) ,
  .CARRY( N00013 ) ,
  .SUM( SUM )
 ) ;
HALFADD HALFADD_B (
  .X( X ) ,
  .Y( Y ) ,
  .CARRY( N00023 ) ,
  .SUM( N00011 )
```

```
 ) ;
endmodule
```

VHDL netlist format

VHDL netlists have the following characteristics:

- If the 1076-87 VHDL standard is selected, legal characters for node names are limited to:
  0..9 A..Z a..z _ (underscore)
  with the following limitations:
  - The first character is limited to: A..Z a..z
  - The last character restricted from: _ (underscore)
  - If the 1076-93 VHDL standard is selected, you can use special characters, VHDL reserved words, and names that begin with digits. To do so, delimit the name with backslashes (\) and precede any special characters—including "internal" backslashes (not the delimiters)—with a backslash. The following table contains some examples.

Object	Name	Problem	Solution
Node	signal	Reserved Word	\signal\
Node	SIGNAL	Case sensitivity	\SIGNAL\
Pin	Q\	Overbar	\Q\\\
Pin	R\E\S\E\T\	Overbar	\R\\E\\S\\E\\T\\\
Pin	12-GND	Leading digit, hyphen	\12\-GND\

For more information, see the 1076-93 VHDL standard, as well as the list of VHDL reserved words, and the VHDL netlist example.

Note: Do not name the data buses in your design in this format: *datain1 [11..0]*, *datain2 [11..0]*, and so on. Instead use this format for naming the data buses: *dataina [11..0]*, *datainb [11..0]*. Because the VHDL netlister expands the data bits in the port map section and writes it as *datain (111)*.

Schematic attributes in VHDL netlists

You can enter part or net attributes on your schematic for inclusion in the VHDL netlist in one of three ways:

- You can enter attributes (properties) with your own user-defined types.
  To do this, when you assign a property to a part or net in the schematic, you define it thusly:

```
attribute name: name
attribute value: vhdl_type is value
```

So, for example, to enter a property for the user-defined type part_version, you assign the name and value as follows:

```
attribute name: my_part
attribute value: part_version is XC1.2
```

The resulting VHDL netlist would include the attribute as follows:
ATTRIBUTE my_part:part_version
ATTRIBUTE my_part of PA3 : signal is XC1.2

Note: If you do not define the property's VHDL type in the value field, or if the VHDL type is not defined in the ATTRIBUTE.VHX file, Capture assigns the type "string" to that property.

- You can enter attributes (properties) without a user-defined type.
  To do this, when you assign a property to a part or net in the schematic, you define it thusly:

```
attribute name: name
attribute value: value
```

So, for example, to enter a property without a user-defined type, you assign the name and value as follows:

```
attribute name: blackbox
attribute value: no_touch
```

The resulting VHDL netlist would include the attribute as follows:

```
ATTRIBUTE blackbox:string
ATTRIBUTE no_touch of PA3 : signal is true
```

- You can enter attributes (properties) that are assigned types in the ATTRIBUTE.VHX file by matching the attribute name with an attribute type that is defined in that file.
  To do this, when you assign a property to a part or net in the schematic, you do not explicitly assign it a type and you use a name that is defined as a particular type in the ATTRIBUTES.VHX file. Thus:

```
attribute name: defined_name
attribute value: value
```

In this case, Capture checks the contents of the ATTRIBUTE.VHX file, locates the attribute name and associates the type with the attribute name.
So, for example, to enter a property and assign it a type as defined in the ATTRIBUTE.VHX file, you assign the name and value as follows:

```
attribute name: attribute_syn_preserve
attribute value: false
```

The property attribute_syn_preserve is defined as type “boolean” in the ATTRIBUTE.VHX file. Therefore, the resulting VHDL netlist would include the attribute as follows:

```
ATTRIBUTE attribute_syn_preserve:boolean
ATTRIBUTE attribute_syn_preserve of PA3 : signal is false
```

Note: The ATTRIBUTE.VHX file is a text file that you can edit to define your own attributes and associated types.

Example

This netlist was created with no options selected. VHDL netlists normally have a .VHD file extension.

```
LIBRARY IEEE;
USE IEEE.std_logic_1164.all;
ENTITY EX6B IS PORT (
X : IN std_logic;
Y : IN std_logic;
CARRY : OUT std_logic;
SUM : OUT std_logic
); END EX6B;
ARCHITECTURE STRUCTURE OF EX6B IS
-- COMPONENTS
COMPONENT \74LS32\
PORT (
I0_A : IN std_logic;
I1_A : IN std_logic;
O_A : OUT std_logic;
VCC : IN std_logic;
GND : IN std_logic;
I0_B : IN std_logic;
I1_B : IN std_logic;
```

```
O_B : OUT std_logic;
I0_C : IN std_logic;
I1_C : IN std_logic;
O_C : OUT std_logic;
I0_D : IN std_logic;
I1_D : IN std_logic;
O_D : OUT std_logic
); END COMPONENT;
COMPONENT \74LS08\
PORT (
I0_A : IN std_logic;
I1_A : IN std_logic;
O_A : OUT std_logic;
VCC : IN std_logic;
GND : IN std_logic;
I0_B : IN std_logic;
I1_B : IN std_logic;
O_B : OUT std_logic;
I0_C : IN std_logic;
I1_C : IN std_logic;
O_C : OUT std_logic;
I0_D : IN std_logic;
I1_D : IN std_logic;
O_D : OUT std_logic
); END COMPONENT;
COMPONENT \74LS04\
PORT (
I_A : IN std_logic;
O_A : OUT std_logic;
VCC : IN std_logic;
GND : IN std_logic;
I_B : IN std_logic;
O_B : OUT std_logic;
I_C : IN std_logic;
O_C : OUT std_logic;
I_D : IN std_logic;
O_D : OUT std_logic;
I_E : IN std_logic;
O_E : OUT std_logic;
I_F : IN std_logic;
O_F : OUT std_logic
); END COMPONENT;
-- SIGNALS
SIGNAL X_BAR : std_logic;
SIGNAL N00037 : std_logic;
```

```
SIGNAL N00035 : std_logic;
SIGNAL GND : std_logic;
SIGNAL VCC : std_logic;
SIGNAL N5056796111 : std_logic;
-- GATE INSTANCES
BEGIN
U1 : \74LS32\ PORT MAP
I0_A => 'Z',
I1_A => 'Z',
O_A => OPEN,
VCC => OPEN,
GND => OPEN,
I0_B => N00035,
I1_B => N00037,
O_B => SUM,
I0_C => 'Z',
I1_C => 'Z',
O_C => OPEN,
I0_D => 'Z',
I1_D => 'Z',
O_D => OPEN
);
U2 : \74LS08\ PORT MAP(
I0_A => X,
I1_A => N5056796111,
O_A => N00035,
VCC => VCC,
GND => GND,
I0_B => Y,
I1_B => X_BAR,
O_B => N00037,
I0_C => Y,
I1_C => X,
O_C => CARRY,
I0_D => 'Z',
I1_D => 'Z',
O_D => OPEN
);
U3 : \74LS04\ PORT MAP(
I_A => X,
O_A => X_BAR,
VCC => VCC,
GND => GND,
I_B => Y,
O_B => N5056796111,
I_C => 'Z',
```

```
O_C => OPEN,
I_D => 'Z',
O_D => OPEN,
I_E => 'Z',
O_E => OPEN,
I_F => 'Z',
O_F => OPEN
);
END STRUCTURE;
LIBRARY IEEE;
USE IEEE.std_logic_1164.all;
ENTITY FULLADD IS PORT (
SUM : OUT std_logic;
X : IN std_logic;
Y : IN std_logic;
CARRY_OUT : OUT std_logic;
CARRY_IN : IN std_logic
); END FULLADD;
ARCHITECTURE STRUCTURE OF FULLADD IS
-- COMPONENTS
COMPONENT \74LS32\
PORT (
I0_A : IN std_logic;
I1_A : IN std_logic;
O_A : OUT std_logic;
VCC : IN std_logic;
GND : IN std_logic;
I0_B : IN std_logic;
I1_B : IN std_logic;
O_B : OUT std_logic;
I0_C : IN std_logic;
I1_C : IN std_logic;
O_C : OUT std_logic;
I0_D : IN std_logic;
I1_D : IN std_logic;
O_D : OUT std_logic
); END COMPONENT;
COMPONENT EX6B PORT (
X : IN std_logic;
Y : IN std_logic;
CARRY : OUT std_logic;
SUM : OUT std_logic
); END COMPONENT;
-- SIGNALS
SIGNAL N00015 : std_logic;
```

```
SIGNAL N00013 : std_logic;
SIGNAL N00025 : std_logic;
SIGNAL VCC : std_logic;
SIGNAL GND : std_logic;
-- GATE INSTANCES
BEGIN
U1 : \74LS32\&#9;PORT MAP
I0_A => N00015,
I1_A => N00025,
O_A => CARRY_OUT,
VCC => VCC,
GND => GND,
I0_B => 'Z',
I1_B => 'Z',
O_B => OPEN,
I0_C => 'Z',
I1_C => 'Z',
O_C => OPEN,
I0_D => 'Z',
I1_D => 'Z',
O_D => OPEN
);
halfadd_A : EX6B PORT MAP(
X => CARRY_IN,
Y => N00013,
CARRY => N00015,
SUM => SUM
);
halfadd_B : EX6B PORT MAP(
X => X,
Y => Y,
CARRY => N00025,
SUM => N00013
);
END STRUCTURE;
```

VST Model netlist format

This format file produces netlists for modeling with OrCAD's Digital Simulation Tools 386+. See the Digital Simulation Tools User's Guide for details.
VST Model netlists have the following characteristics:

- Part names, module names, reference strings, node names, and pin numbers are not checked for length.

- Node numbers and pin names are not used.
- All ASCII characters are legal.

  For more information, see the VST netlist example.

VST netlist constraints

When you create a VST Model netlist, be sure you include the OrCAD-supplied VSTGATES.OLB, VSTRAM.OLB, VSTROM.OLB, and VSTOTHER.OLB part libraries in your project. You can use only the parts provided in these libraries to create the schematic folder.

VST pipe commands

Lines of text may be placed on your schematic page to be included in the VST Model netlist. Select the Text command from the Place menu to place the text on a schematic page.
Each line of text must start with the pipe character (|). The first line must be:

```
|VST_MODEL
```

This tells Capture to extract the information in the following lines of text when generating a VST Model netlist. The remaining lines can contain a header, comments, and directives compatible with OrCAD's Digital Simulation Tools 386+ Add Device Model device modeling language. For details on the Add Device Model Language, see the Digital Simulation Tools User's Guide.

WinBoard netlist format

Ivex WinBoard netlists have the following characteristics:

- Part names are not checked for length.
- Module names are not checked for length.
- Reference strings are not checked for length.
- Node names are limited to eight characters.
- Node numbers are limited to five digits following the "N" prefix.
- Pin numbers are not checked for length.

  For more information see the WinBoard netlist example.

Example

WinBoard netlists normally have an .NET file extension.

```
WINBOARD 1.01
`I "ORCAD CAPTURE 7.20";
`F "D:\ORCAD DEMO\CAPTURE\NETLIST UPDATES\DESIGN1\FIG_B-01.SCH";
`T "Generic Netlist Example";
`S "Thursday, November 12, 1998";
```

```
`C "OrCAD-01";
`R "A";
`C "OrCAD";
`C "9300 S.W. Nimbus Ave.";
`C "Beaverton, OR 97008";
`C "(503) 671-9500 Corporate Offices";
`C "(503) 671-9400 Technical Support";
`M 14DIP300,,74LS00,6CB84CBA,U1,C,GR1
(1 A IN A1)
(2 A IN A1)
(3 N00019 OU A1)
(4 B IN A1)
(5 N00013 IN A1)
(6 Q OU A1)
(7 GND PO A1)
(8 N00013 OU A1)
(9 Q IN A1)
(10 CLOCK IN A1)
(11 ?1 OU A1)
(12 ?2 IN A1)
(13 ?3 IN A1)
(14 VCC PO A1)
;
`M 14DIP300,,74LS32,6E46169D,U2,C,GR1
(1 N00019 IN A1)
(2 Q IN A1)
(3 OUT OU A1)
(4 ?4 IN A1)
(5 ?5 IN A1)
(6 ?6 OU A1)
(7 GND PO A1)
(8 ?7 OU A1)
(9 ?8 IN A1)
(10 ?9 IN A1)
(11 ?10 OU A1)
(12 ?11 IN A1)
(13 ?12 IN A1)
(14 VCC PO A1)
;
```

WireList netlist format

WireList netlists have the following characteristics:

- Part and node names are not checked for length.

- Module names are limited to twenty-nine characters.
- Reference strings are limited to nine characters.
- Node numbers are limited to five digits following the "N" prefix.
- Pin numbers are limited to seven characters.
- Pin names are limited to fifteen characters.
- Legal characters for node numbers are 0..9.
- Legal characters for pin numbers are 0..9, unless the option Do not output pin numbers for Grid Array parts is selected. If you select this option, Capture skips nonnumeric pin numbers, such as those on grid array parts, and any ASCII character is legal.
- All ASCII characters are legal except as noted for node numbers and pin numbers.

Note: WireList netlists generated by Capture use all uppercase letters for pin names, pin numbers, and net names.

For more information, see the WireList netlist example.

Example

WireList netlists normally have an .NET file extension.

```
Wire List
Generic Netlist Example Revised: Thursday, November 12, 1998
OrCAD-01 Revision: A
OrCAD
9300 S.W. Nimbus Ave.
Beaverton, OR 97008
(503) 671-9500 Corporate Offices
(503) 671-9400 Technical Support
<<< Component List >>>
74LS00 U1 14DIP300
74LS32 U2 14DIP300
<<< Wire List >>>
  NODE REFERENCE PIN # PIN NAME PIN TYPE PART VALUE
[00001] GND
  U1 7 GND Power 74LS00
  U2 7 GND Power 74LS32
[00002] VCC
  U1 14 VCC Power 74LS00
  U2 14 VCC Power 74LS32
[00003] CLOCK
  U1 10 I1_C Input 74LS00
[00004] Q
  U1 6 O_B Output 74LS00
  U2 2 I1_A Input 74LS32
```

```
  U1 9 I0_C Input 74LS00
[00005] OUT
  U2 3 O_A Output 74LS32
[00006] B
  U1 4 I0_B Input 74LS00
[00007] N00019
  U1 3 O_A Output 74LS00
  U2 1 I0_A Input 74LS32
[00008] N00013
  U1 5 I1_B Input 74LS00
  U1 8 O_C Output 74LS00
[00009] A
  U1 1 I0_A Input 74LS00
  U1 2 I1_A Input 74LS00
```

21

Database Migration

Capture CIS, v16.3, ships with a new design and library database schema version. This new data format improves application performance and ensures data integrity in your Capture designs and libraries. If you currently have designs and libraries developed in a previous version of Capture, you can automatically upgrade these using v16.3. This section provides a brief overview of the scenarios and application prompts you may encounter when upgrading your designs and libraries to v16.3.

> In this section, the Capture version v16.2 refers to Capture versions prior to and including v16.2.

> The data format used in Capture versions prior to and including v16.2 is v9.0.

In this section:

Upgrade Matrix

The following version matrix provides a quick glance and the possible Design versus Application versus Library version scenarios while upgrading your environment from v16.2 to v16.3.

Application Version	Design Version	External Referred Design Version	Library Version	Is Supported? (YES / NO)

v16.3	v16.3		v16.2	**YES** Recommendation to upgrade Library ***See*** Capture v16.2 Library in Capture v16.3
v16.3	v16.2		v16.3	**YES** Recommendation to upgrade Design ***See*** Capture v16.2 Design in Capture v16.3
v16.3	v16.3		v16.3	**YES** Recommended
v16.3	Any	v16.3	Any	**YES** Recommended
v16.3	Any	v16.2	Any	**YES** Recommendation to upgrade Referred Design ***See*** Capture v16.3 Design with Externally Referenced v16.2 Design
v16.2	v16.3		v16.3	**YES** ***See*** Downgrade v16.3 Design or Library
v16.2	v16.3		v16.2	**YES** ***See*** Downgrade v16.3 Design or Library
v16.2	v16.2		16.3	**YES** ***See*** Downgrade v16.3 Design or Library

Table H-1

Upgrade Batch Script

You can use a TCL batch script provided by Cadence to upgrade your v16.2 designs and libraries to v16.3. Please contact Cadence Customer Support for details.

Capture v16.2 Design in Capture v16.3

This scenario arises when you install Capture v16.3 and you currently have Capture designs in v16.2.

In this section:

Opening v16.2 design in Capture v16.3

As soon as you open a v16.2 design in Capture v16.3, the application prompts you to upgrade the design.

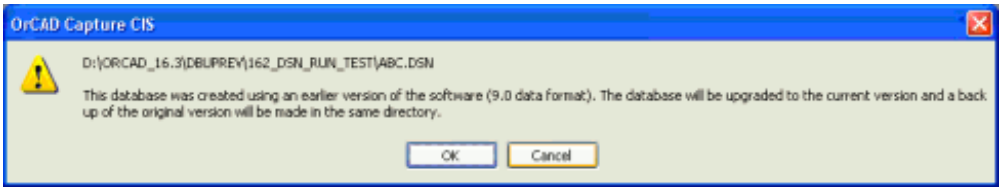


Figure H-1
Choose *OK* to upgrade the design.

> If you choose Cancel, the design will not open in v16.3.

Capture v16.2 Library in Capture v16.3

This scenario arises when you install Capture Release 16.3 and you currently have Capture libraries in v16.2.

In this section:

Saving v16.2 library in Capture v16.3

When you save a v16.2 library opened in v16.3, the application prompts you to upgrade the library.

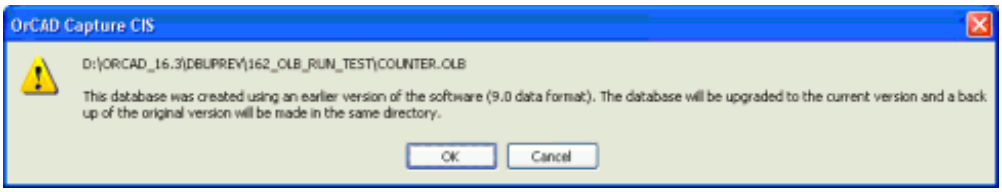


Figure H-2
Choose *OK* to upgrade the library.

> If you choose Cancel, the library will not be saved.

Closing (or exiting Capture) v16.2 library in Capture v16.3

When you close a v16.2 library opened in Capture v16.3 or when you exit Capture after opening a v16.2 library, the application prompts you to upgrade the library,

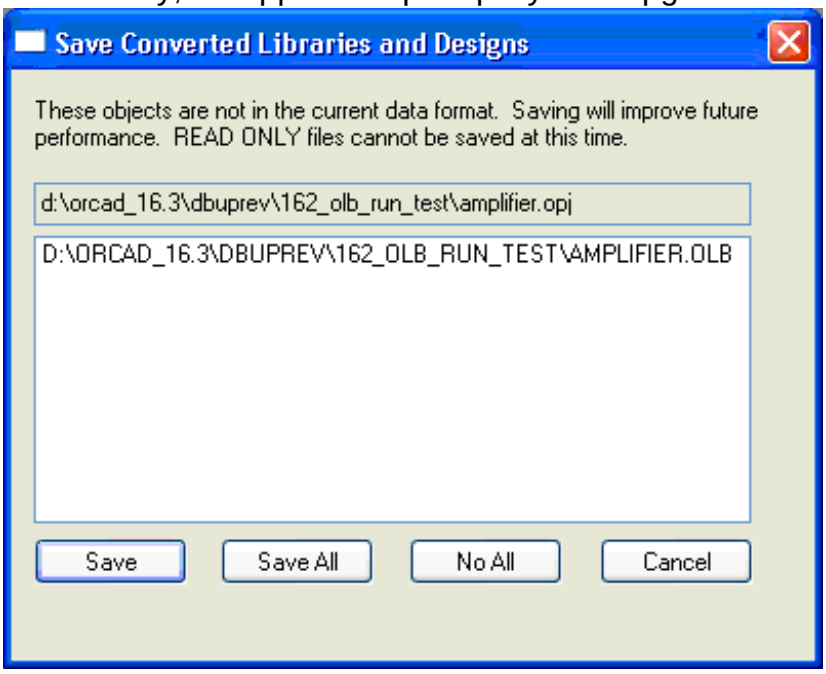


Figure H-3
The Save Converted Libraries and Designs dialog displays the list of the v16.2 libraries. You can:

- Choose *Save All* to upgrade all the listed libraries.
- Choose *No All* to not upgrade any of the listed libraries.

- The Cancel option is only available if you close the library and not when you exit the application.
- When you open a v16.2 design in v16.3, Capture prompts you to upgrade the design. However, this does not happen when you open a v16.2 library in v16.3. In this case, you will only be prompted to upgrade the library if you choose the Save command, close the project or close Capture.
- If the v16.2 library is in Read Only mode, you will not be able to save and hence upgrade the library.

Capture v16.3 with v16.2 Referenced Libraries

This scenario arises when v16.2 libraries are referenced through the Place Part dialog in Capture. This occurs either if v16.2 libraries are configured in the Capture.ini OR when you select v16.2 libraries through the Add Library command in the Place Part dialog.

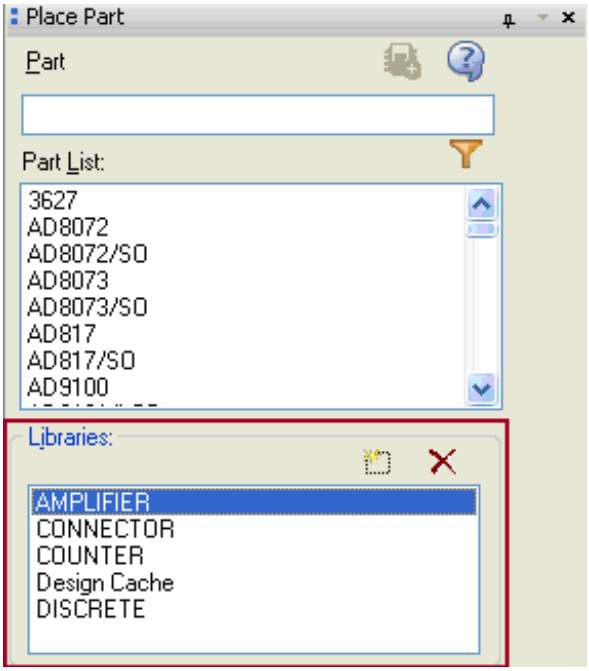


Figure H-4

In this section:

Exiting Capture v16.3 with v16.2 referenced libraries

When you refer v16.2 libraries in Capture v16.3, on exit the application prompts you to upgrade

these libraries.

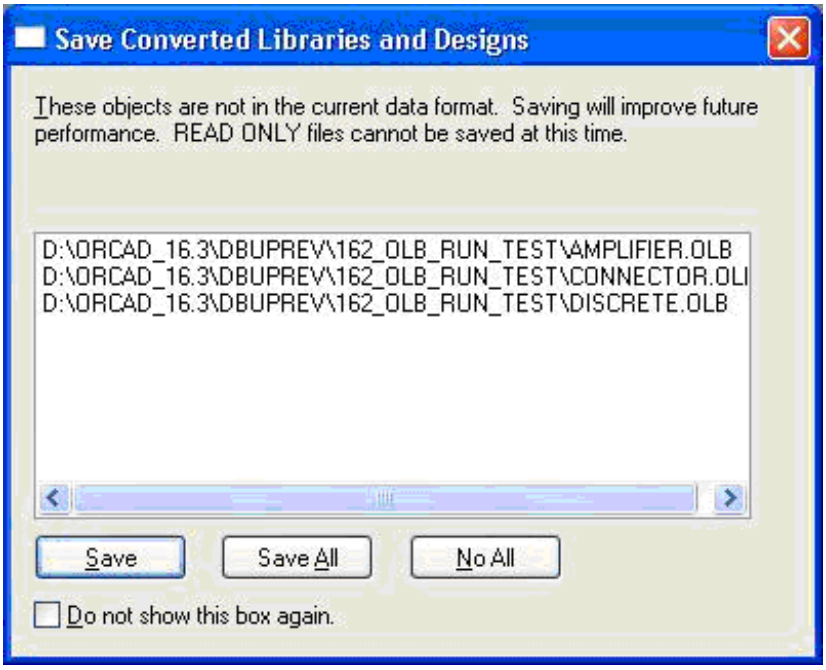


Figure H-5

The Save Converted Libraries and Designs dialog displays the list of all the v16.2 libraries. You can select specific libraries from the list and:

- Choose Save to upgrade the selected libraries.
- Choose *Save All* to upgrade all the listed libraries.
- Choose *No All* to not upgrade any of the listed libraries.
- Check the *Do not show this box again* option to retain your current selection.

For example, if you select this option and then choose Save All, then whenever this scenario arises, this dialog box will not be displayed and the libraries will be upgraded automatically on exit.

- If a v16.2 referenced library is in Read Only mode, you will not be able to save and hence upgrade the library.
- If you perform library-specific commands like Update Cache or Replace Cache in Capture v16.3, the libraries will not be upgraded.

Capture v16.3 Project with v16.2 Libraries included

This scenario arises when you have a v16.3 project with one or more v16.2 libraries added to the project.

You add a library in a Capture project by choosing Add File from the pop-up menu on the Library folder in Project Manager.

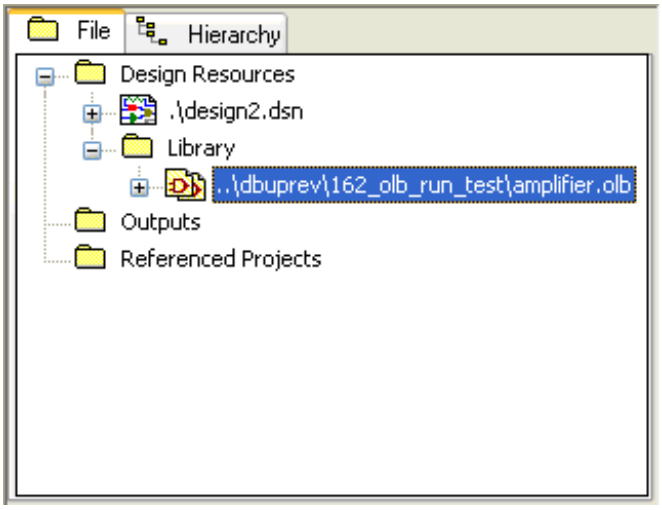


Figure H-6

In this section:

Saving v16.2 Library included in v16.3 Project

When you save a v16.2 library that is included in a v16.3 project, the application prompts you to upgrade the library.

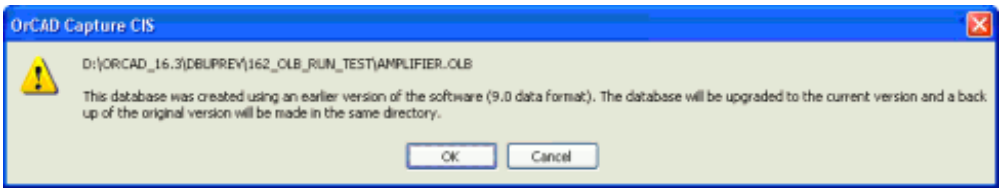


Figure H-7
Choose *OK* to upgrade the library.

> If you choose Cancel, the library will not be saved.

Closing (or exiting Capture) v16.3 project with added v16.2 library

When you close a v16.3 project to which you have added one or more v16.2 libraries, the application prompts you to upgrade the added libraries.

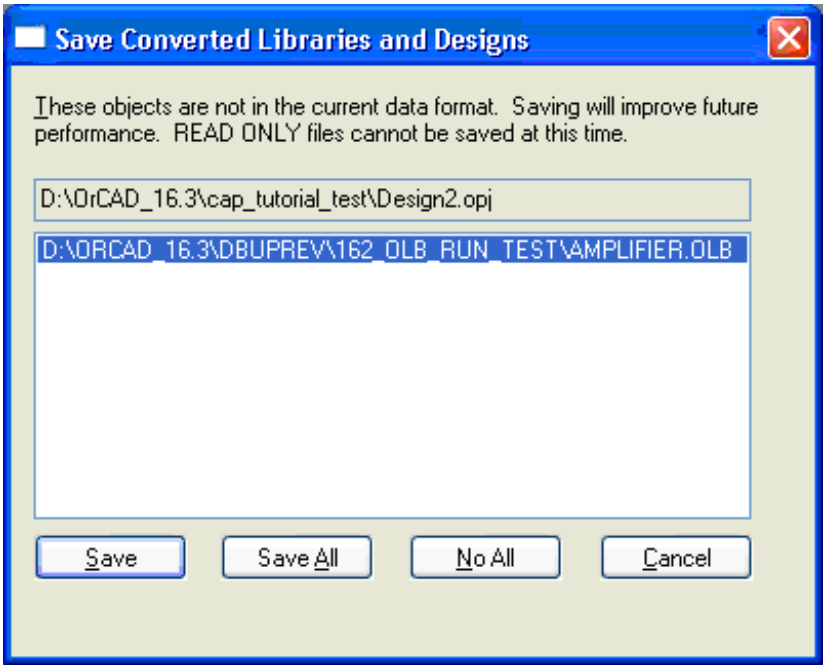


Figure H-8
The Save Converted Libraries and Designs dialog displays the list of all the v16.2 libraries. Select specific libraries from the list and choose:

- Save to upgrade the selected libraries.
- *Save All* to upgrade all the listed libraries.
- *No All* to not upgrade any of the listed libraries.

- The **Cancel** option is only available if you close the project and not when you exit the application.
- If a v16.2 configured library is in **Read Only** mode, you will not be able to save and hence upgrade the library.

Capture v16.3 Design with Externally Referenced v16.2 Design

This scenario arises when you have a v16.3 design that contains an external reference to a v16.2 design. For example, a v16.3 design with a hierarchical block that references a external v16.2 design or a v16.3 design with a hierarchical part that reference a v16.2 schematic.

In this section:

Exiting Capture v16.3 design with Externally Referenced v16.2

Design

If you have a v16.3 design that contains an external reference to a v16.2 design and you close Capture, the application prompts you to upgrade the externally referenced design.

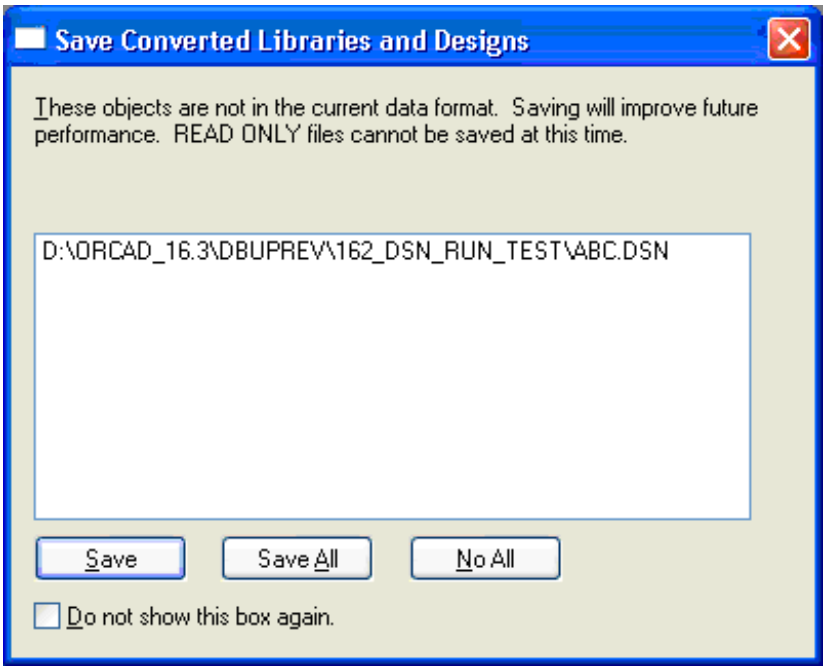


Figure H-9
The Save Converted Libraries and Designs dialog displays the list of the externally referenced designs.
Select specific design from the list and choose :

- *Save* to upgrade the selected designs.
- *Save All* to upgrade all the listed designs.
- *No All* to not upgrade any of the listed designs.

You can check the **Do not show this box again** option to retain your current selection.
For example, if you select this option and then choose Save All, then whenever this scenario arises, this dialog box will not be displayed and the Save All option will be executed.

If you open a v16.3 design with a v16.2 externally referenced design and you descend to the v16.2 design (by choosing the Descend Hierarchy command), the v16.2 design is upgraded to v16.3 and a backup of the v16.2 design is made in the same location as the original v16.2 design.

Opening a v16.3 Design or Library in Capture v16.2

This down-grade scenario arises when you need to open a v16.3 library or design in v16.2.

In this section:

Downgrade v16.3 Design or Library

To make a v16.3 design or library available for use in v16.2, you need to downgrade your design or library.
To downgrade a v16.3 design or library, choose the *File - Save As* menu option.
In the *Save As* dialog, choose the 16.2 Design / Library file type option.

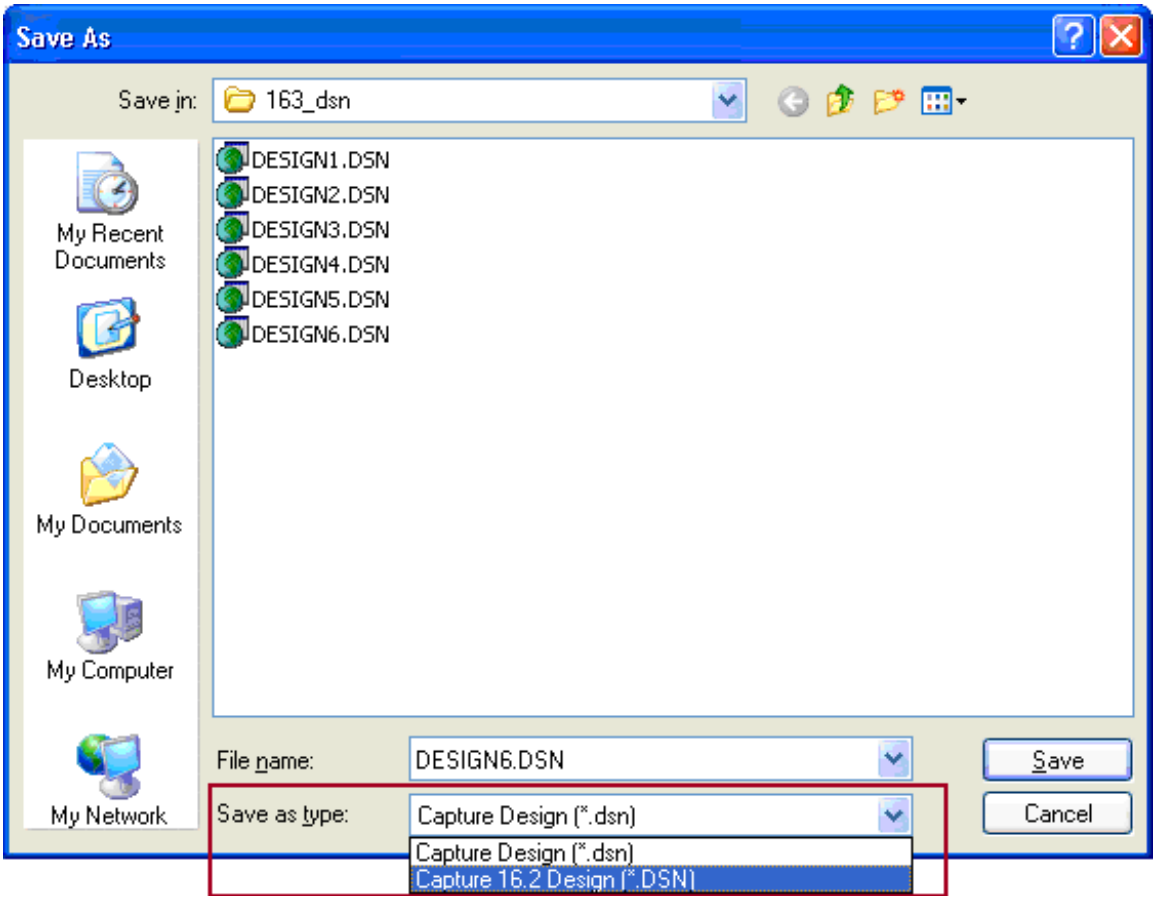


Figure H-10
You can now open the design or library in v16.2.

Capture v16.3 enhancements like user-defined pin shapes, color wire and color part, Bezier curves, OLE object support, will not be available in a design that is down-graded from v16.3 to v16.2. So, you are recommended to down-grade designs with caution as you may lose data in the down-grade.

Upgraded Design _ Library backup

In this section:

Design Backup

When a v16.2 design is upgraded, a backup of the design file (.DSN) is made in the same location using the nomenclature - <Design Name (excluding file extension)> _9_0_0.DBK. You can then rename this backup file with a .DSN extension to make it available for use in Capture v16.2.

Library Backup

When a v16.2 library is upgraded, a backup of the library file (.OLB) is made in the same location using the nomenclature - <Library Name (excluding file extension)> _9_0_0.OBK. You can then rename this backup file with a .OLB extension to make it available for use in Capture v16.2.

22

Glossary

A | B | C | D | E | F | G | H | I | J | L | M | N | O | P | R | S | T | U | V | W | X | Y | Z

A

absolute simulation time
A time measured from the beginning of the simulation.
absolute stimulus
See force.
actual
The value of a parameter passed to a VHDL subprogram.
alias
See net alias, part alias.
ANSI
The acronym for American National Standards Institute.
ARCHITECTURE
A VHDL construct that describes the behavior of a design unit (ENTITY/ARCHITECTURE pair). The ARCHITECTURE can also serve to connect other VHDL design units.
arrow keys
On your computer keyboard, the keys you use to navigate around your screen. Each key is marked with an arrow and is named for the direction in which the arrow points. There is an up arrow, down arrow, left arrow, and right arrow key. Also known as direction keys.
ascend
In a hierarchical design, to move from a child schematic folder to its parent schematic folder. This is done in the schematic page editor using the Ascend Hierarchy command on the View menu. See also descend.
ASCII
The acronym for American Standard Code for Information Interchange. The ASCII character-coding set enables different applications to exchange information.
ASSERT
A VHDL keyword typically used in conjunction with the REPORT and SEVERITY keywords to detect a particular circuit state and announce the condition with an associated severity level.
AutoECO
The acronym for automatic engineering change order. Layout's AutoECO command translates schematic netlist information from Capture to Layout. See also forward annotate.

B

back annotate
To apply modifications to part properties in a schematic folder, such as updating part references

and pin numbers, swapping gates, or swapping pins. Properties are back annotated in the project manager, using the Back Annotate command or the Update Properties command on the Tools menu.

bitmap

Bitmaps are graphic images that are made up of pixels, which are the tiny dots on your computer screen. Each pixel in a bitmap is represented by a number between 0 and 255, inclusive, with 0 being the darkest (no luminance) and 255 being the lightest (full luminance). Bitmaps have a .BMP extension, and can be placed on a schematic page using the Picture command from the schematic page editor's Place menu.

BLIF

Berkeley Logic Interchange Format. This format, developed at the University of California, Berkeley, is used to convey Boolean logic between programs. BLIF files are termed PLAs.

bookmark

Just as you can place bookmarks in a book to mark a specific place, you can place bookmarks on a schematic page to indicate a location you would like to return to frequently. To place a bookmark, use the Bookmark command on the Place menu in the schematic page editor. To go to a bookmark when in the schematic page editor, use the Go To command on the View menu. To go to a bookmark when in the project manager, use the Browse command on the Edit menu to display bookmarks in the browse window, and then choose the bookmark.

BRD

An PCB Editor board file. The .BRD contains information about the board, component symbols, pins, nets, keep-ins and keep-outs, plus how the parts are placed and routed. The .BRD file also includes the properties and constraints that apply to parts and areas of the board.

breakpoint

A pause in the simulation triggered by a particular condition. You can set a breakpoint to occur when a certain state exists on a signal, or just before a particular line in a VHDL model is executed during a simulation.

browse window

This window displays the results of queries done using the Browse command from the Edit menu. You can double-click on an object in the browse window to go to that item on its schematic page.

bus

A group of scalar signals (wires) that are never connected to a net. A bus name defines the signals carried by the bus and connects those signals to the corresponding nets. For example, the bus name A[0:3] defines a four-signal bus and connects the four signals A[0], A[1], A[2], and A[3] with nets A0, A1, A2, and A3. See also bus pin, bus entry.

bus entry

A bus entry is used to tie a signal to a bus. The advantage of using bus entries instead of wires is that two bus entries can be connected at the same point on a bus without connecting the signals. If two wires are run directly to a bus at the same location, the signals are connected. See also bus, bus pin.

bus pin

A pin width that can carry multiple signals, as opposed to a scalar that carries only one signal. A bus pin represents all the pins for a bus, and it uses the same naming convention as buses. See also bus, bus entry.

C

CAGE code
Abbreviation for Commercial and Government Entity Code. A number—provided by the federal government to its suppliers—that can be present in the title block of a schematic page.

CELL
An EDIF keyword that defines the interface to a hierarchical block or part. An OrCAD Capture hierarchical block or part will generate a cell in the EDIF netlist. Simulate displays EDIF cells as contexts within the Simulate netlist.

child
In a hierarchical design, a schematic folder whose circuitry is represented by a hierarchical block in the parent schematic folder. To move from parent to child is to descend the hierarchy. This is done in the schematic page editor by selecting the hierarchical block representing the child, and then choosing the Descend Hierarchy command on the View menu. A child schematic folder contains circuitry referenced by its parent schematic folder. The child schematic folder may contain hierarchical ports that connect its signals to signals in the parent schematic folder or to signals on other pages of the child schematic folder. See also ascend.

Clipboard
A temporary storage location used to transfer data between files and between applications. You transfer data to the Clipboard by using the Copy or Cut command on the Edit menu, and you insert data from the Clipboard by using the commands on the Edit menu.

clock
A signal that has a simple repeating waveform pattern. Typically, clocks drive the synchronous devices in your design. Clock stimuli in Simulate can be overwritten by forces.

clock to output delay
The propagation time for a clocked device. That is, this delay is the length of time required for a data signal to propagate to the device output after being clocked.

command line window
Use the command line window to execute a subset of frequently-used Simulate commands.

complex hierarchy
A design in which two or more hierarchical blocks (or parts with attached schematic folders) reference the same schematic folder. See also hierarchical design, simple hierarchy.

context
The level of hierarchy at which logic macros, pins, and signals are found. A context in Simulate corresponds to an EDIF cell or a VHDL ENTITY/ARCHITECTURE pair. A hierarchical block on an OrCAD Capture schematic page appears as a context in Simulate.

convert
An alternate form—such as a DeMorgan equivalent—that can be stored with each part.

cross probing
When intertool communication is enabled in Capture, selecting objects in Capture causes the corresponding objects to be highlighted in PCB Editor. Also, selecting objects in PCB E ditor causes the corresponding objects to be highlighted in Capture. Both applications must be open. See also intertool Communication.

CurrentLocation
A value stored by Capture that determines the starting point for the next macro command. This value is set by the previous macro. You can also set this value by moving the pointer to the

desired location, and clicking the left mouse button.

D

DeMorgan equivalent
An electrically-equivalent part based on the DeMorgan rules of equivalence. These rules represent the duality of AND and OR in Boolean expressions: if all AND operations are changed to OR operations, all OR operations are changed to AND operations, and all variables and constants are negated, then the value of the expression remains unchanged. A DeMorgan equivalent can be stored in the convert of a part.

descend
In a hierarchical design, to open and view the child schematic folder represented by a hierarchical block in the parent schematic folder. To descend a hierarchical design, you select a hierarchical block in the schematic page editor, then choose the Descend Hierarchy command from the View menu. See also ascend.

design
The set of schematics and models that collectively define the behavior of your project.

design cache
A local library contained in each project that contains all the parts and symbols used in the design.

design entry
The process of expressing an electronic design. Typically, design entry involves describing a structure using schematic logic macros, behavioral description with a hardware description language (HDL), or some combination of both methods. The design expression is processed to produce a gate-level netlist that can be used for simulation or design implementation.

design implementation
The process of mapping, fitting, or routing your design to or within a specific device. Design implementation can yield timing values that allow you to perform timing analysis and ensure that your design meets your performance requirements.

device-fitter
A software tool to implement a logic design (usually recorded as an Open-PLA or gate-level netlist) into the physical resources of a CPLD.

DIFFERENTIAL_PAIR
Represents a pair of flat nets that will be routed in a way that the signals passing through them are opposite in sign with respect to the same reference. This ensures that any electromagnetic noise in the circuit is cancelled out.

document
A project, schematic page, library, part, or symbol. Each of these is part of a project or a library file. In addition, stimulus files, simulation result files, and simulation models are documents.

DRC
The abbreviation for Design Rules Check, a tool found on the Tools menu in the project manager. This tool checks a project (or a subset of the design) for conformance to a set of configurable design criteria, electrical rules and physical rules for creating netlists.

E

EDA
The acronym for Electronic Design Automation. Software and hardware tools used to ascertain the viability of an electronic design. These tools perform simulation, synthesis, verification, analysis, and testing of a design.
EDIF
The acronym for Electronic Design Interchange Format. A standard published by the EIA (Electronic Industries Association) that defines the semantics and syntax for an interchange format that communicates electronic designs. Simulate uses EDIF 2 0 0 standard netlists as a simulation resource.
ENTITY
A VHDL construct that defines the interface to a VHDL design unit (an ENTITY/ARCHITECTURE pair).
equivalent
See convert, DeMorgan equivalent.
ERC
The abbreviation for Electrical Rules Check, a subset of the Design Rules Check tool found on the Tools menu in the project manager. The ERC matrix is the decision matrix that tells the Design Rules Check tool the conditions to check for when evaluating connections between pins, hierarchical ports, and off-page connectors.
event
Any signal transition that occurs during simulation. An event appears as a transition in the wave window, and generates a new row in the list window. Simulate records the history of all events for any signal that are traced in either window.
external design
Any referenced design that is not included as part of the main design's schematic pages. An external design may be a library (.OLB) part which you can place with the

Part command

from the Place menu. Alternately, an external design may be a complete schematic (.DSN) design which you can include by using the

Hierarchical Block command

from the Place menu. Whenever you use external designs you set up a hierarchical structure. If you copy an external designs without taking into consideration the occurrence properties inherent in a hierarchical structure, you might get instance property but not occurrence values. Note: Except for occurrence properties, the schematics of externally-referenced libraries and designs should not be edited. You should view them as read-only designs. Trying to edit, then save, these designs from within your schematic can introduce errors such as duplicate reference designators and other problems. When saving schematics with externally-referenced libraries or designs, occurrence properties are saved but altered instance values are not. If you want to change externally-referenced libraries or designs you should first close the referencing design. Then, open the referenced library or design, make the necessary changes, and save and close the referenced library or design. At this point, you can reopen the original design and reference the modified design.

F

flat design
A schematic folder structure without hierarchy (no hierarchical blocks or ports; no parts with attached schematic folders). A flat design can include schematic pages in which output lines of one schematic page connect laterally to input lines of another schematic page through objects called off-page connectors. You place off-page connectors using the Off-Page Connector command on the Place menu in the schematic page editor. Flat designs are practical for small designs with few schematic pages. See also hierarchical design, complex hierarchy, simple hierarchy.

flat net
A type of net represented by a flattened (non hierarchical) netlist for a PCB, such as a layout netlist.

force
A scheduled state change that occurs at a specific simulation time. A force will override any other signals driving the node. That is, a force is equivalent to placing a probe on the node. When you place a force, it remains in effect until you replace it with another force, or until you remove the force from the stimulus file.

forward annotate
The process of sending netlist data in the form of a .BRD file from Capture to PCB Editor.

functional simulation
Simulation that verifies design logic and functionality without regard to timing (for example propagation or critical path).

fuse plot
An ASCII representation of a fuse map. You can use this file to visually review the fuse map that Express creates for your simple PLD. Fuse plots appear as shown in the following example:

FUSE MAP FOR P12H6

	0	2	4	6	8	10	12	14	16	18	20	22
0	--	x-	--	--	--	--	x-	--	--	--	--	--
24	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
48	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
72	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
96	x-	--	--	--	--	--	--	--	--	--	--	--
120	--	--	--	--	--	--	--	x-	--	--	--	--

144	--	--	--	-x	--	--	--	--	x-	--	--	--
168	--	--	--	x-	--	--	--	--	-x	--	--	--
192	--	--	-x	--	--	--	--	--	--	--	--	--
216	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
240	--	--	--	--	-x	--	--	--	--	--	--	--
264	--	--	--	--	--	--	--	--	--	--	-x	--
288	--	--	--	--	--	-x	--	--	--	-x	--	--
312	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
336	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
360	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx

	Legend:	x fuse intact

				- fuse open

	203 fuses open of 384 total.

G

globals
Power symbols and ground symbols. Could also be other objects that function the same as power or ground.
graphic object
An object drawn or placed on a schematic page or part -- such as an arc, line, rectangle, ellipse, polygon, image, or text -- that has no electrical connectivity.
grid references
The border around a schematic page that provides a visual reference to the grid. Grid references

can be used as a destination for the Go To command on the View menu. Grid references can be set to visible or hidden in both the Design Template and Schematic Page Properties commands on the Options menu.

H

heterogeneneous part
A package with multiple parts that are graphically different or contain different numbers of pins (for example, a relay). See also homogeneous part.
hierarchical block
A symbol that refers to a child schematic folder in a project. The connection points on a hierarchical block are called hierarchical pins and hierarchical ports. You place a hierarchical block using the Hierarchical Block command on the Place menu.
hierarchical design
A project structure in which schematic folders are interconnected vertically with hierarchical blocks. At least one schematic folder, the root schematic folder, contains symbols representing other schematic folders. See also complex hierarchy, simple hierarchy, flat design.
hierarchical pin
A symbol, placed within a hierarchical blocks, that represents a signal connected to a like-named hierarchical port on another schematic page. You place a hierarchical pin using the Hierarchical Pin command on the Place menu.
hierarchical port
A symbol that specifies that a signal on one schematic page connects to a hierarchical pin on another schematic page. A hierarchical port includes a name and a type (either scalar or bus). You place a hierarchical port using the Hierarchical Port command on the Place menu. See also hierarchical block.
homogeneous part
A package with multiple parts that are graphically identical. See also heterogeneneous part.
HPGL
Acronym for Hewlett-Packard Graphics Language, which is a plotter protocol.

I

IEEE
Acronym for Institute of Electrical and Electronics Engineers.
IEEE Std VDHL 1076
VHDL standard determined by the Institute of Electrical and Electronics Engineers.
implementation path
The path for an attached object, such as a referenced design folder or a library part.
inherent property
One of the set of properties required for a given object. Unlike user defined property, inherent properties cannot be removed.
instance
A part or a symbol that you have placed on a schematic page.
instance property

A user property applied to the placed instance of a part or symbol in the design. This includes PCB Footprint, Value, and Name properties of each placed part or symbol in a design. This is the same as the user properties displayed and editable from the Capture v7.2 Physical view.
An instance property will "shine through" to all occurrences of that instance unless it is overridden by occurrence properties that you have edited. A change using any of the tools, like Annotate, also may update the instance property.

intertool Communication
Abbreviated ITC. A capability that allows OrCAD EDA tools to share information for display and transfer.

ITC
Intertool Communication. A capability available with OrCAD for Windows tools that allows these tools to share information for display and transfer.

J

Junction
A junction, shown as a small dot, is placed at the connection point where two perpendicular wires or buses cross, to give visual confirmation that the items are electrically connected. If you draw a wire across another at a 90-degree angle, the wires are not electrically connected unless you create a junction by clicking the left mouse button on the existing wire as you draw the new wire across it.

L

library
A collection of often-used parts, graphics, schematic pages, and symbols.

library definition
A package property or user property associated with the part in the library.
A library definition will "shine through" to the instance and occurrences of that part property. Shine through is indicated by hash marks in the cell. You can assign a value to it creating an instance property. The instance property then will override the shine-through definition.

location
An X, Y coordinate on the schematic page or part. You can move to a location using the Go To command on the View menu.

M

macrofunction
A high-level building block made of two or more primitives. Muxes, counters, and adders are examples of macrofunctions.

MDD
PCB Editor Module Definition File (.MDD). This is the file type created in PCB Editor once you've designated the extents of a reuse module and specified a module origin. Each physical module is assigned a REUSE_MODULE property and contains placed and routed components.

mirror

To flip along the X (horizontal) or Y (vertical) axis, or both.

N

net

1. All of the wires, buses, parts, and symbols that are logically connected via net names, net aliases, off-page connectors, and hierarchical ports.
2. A general electronic term for a circuit node that ties a collection of component pins together. The EDIF 2 0 0 netlist format contains a netlist region that declares the net name and all component instances that are tied to it. You can trace EDIF nets in Simulate.

   net alias

   A name used to specify signal connections between unconnected wires or buses. For example, if you have wires in two remote locations in a schematic page, you can assign each wire an alias such as "ABC" to connect the signals without physically drawing a wire between them.

   netlist

   A file, usually ASCII, that lists the interconnections of a schematic folder by the names of the connected signals, parts, and pins.

   nonprimitive

   A part with an underlying hierarchy, such as an attached schematic folder.

O

occurrence

A user property applied to multiple occurrences of placed instances of parts or symbols in a design. This is the same as the user properties displayed and editable from the Capture v7.2 Physical view.

The spreadsheet will expand to display occurrence properties if values are different from the instance shine through value; otherwise, the rows are hidden from view. To quickly hide or display all the occurrence properties, press and hold the CTRL key while clicking on one of the plus (+) symbols in the property editor.

A change using any of the tools, like Annotate, also may update the instance property.

off-page connector

An object that conducts signals between schematic pages within a schematic folder. See also flat design, hierarchical port.

P

package

A physical part that contains more than one logical part. For example, a 2N3905 transistor, a fuse, and a 74LS00 are packages. Each part in a package has a unique part reference comprised of a prefix common to all the parts in the package, and a letter unique to each part. For example, a

74LS00 whose part reference prefix is U15 would have four parts whose part references are U15A, U15B, U15C, and U15D. See also homogeneous part, heterogeneneous part.

pan

To change the portion of the schematic page or part being viewed by dragging objects from one location to another. As you drag the object, the schematic page or part pans across the active window.

parent

A schematic folder that contains a hierarchical block that refers to another schematic folder (called a child schematic folder).

part

A part is a basic building block of a design. A part may represent one or more physical components, or it may represent a function, a simulation model, or a text description for use by an external application. A part's behavior is described by a SPICE model, an attached schematic folder, HDL statements, or other means. Parts usually correspond to physical objects—gates, connectors, and so on—that come in packages of one or more parts. Packages with more than one part are sometimes referred to as "multiple-part packages". See also package.

part alias

A duplicate copy of a part using a different name in a library. A part alias uses the same graphics, attached schematic folders, and properties as the original, with the exception of the part value.

part editor

The editor used to create and edit parts and symbols.

part instance

An instance property of a part.

part primitive

See primitive.

part property

A part property is a characteristic of a part that can be edited. A property consists of a name and a value. Examples of property names are part value and color. Their respective property values can be something such as capacitor and red.

part reference

When you place parts on a schematic page, all parts of the same type are assigned the same part reference. For example, C? is assigned to all capacitors. Regardless of the ultimate purpose of your design, each part needs a unique part reference. You can assign part references by editing individual parts in the part editor, or, for PCB designs, by creating a swap file to use with the Back Annotate tool.

If you want to incrementally update a design in which some of the schematic pages have already been updated, you can use the Annotate command to remove part references from those schematic pages.

pattern

A set of events that occur on a signal, relative to a specific simulation time. This pattern may or may not be repeating. Patterns may be overwritten by forces. Conflicts between patterns, or between a pattern and signal propagation are resolved using signal contention resolution.

PCB

Abbreviation for printed circuit board.

pending event

A simulation event that will occur in the future. As signals change state during simulation, a VHDL simulator must evaluate the input stimuli and all design units of the circuit, then anticipate or schedule all events that must be reported before the simulation time can advance. Simulate allows you to view pending events with the Pending Events command.

pin

A pin acts as a point of connectivity for the part it is attached to. In addition to input and output pins, there are also 3-state, bidirectional, open collector, open emitter, passive, and power pins. If a pin connects to a wire, it is a scalar pin; if it connects to a bus, it is a bus pin. See also hierarchical pin.

pin delay

The propagation delay for a pin to pin transition. That is, pin delay is the length of time required for the effects of a signal at an input pin to be reflected at the corresponding output pin(s).

pin swap

The exchange of identical pins in order to decrease route lengths.

pin to pin spacing

The physical spacing between pins on a device.

PLA

A file that uses the BLIF to express Boolean logic. Typically, PLA files are used as entry mechanisms for simulation models into Simulate.

place and route

1. A software tool to implement a logic design (usually recorded as a gate-level netlist) into the physical resources of an FPGA.
2. The process of determining a design layout in order to estimate routing delays and predict design performance.

PLD

Abbreviation for programmable logic device.

Preferred mode warning

Capture automatically sets the preferred mode based on the project type. FPGA and PSpice projects default to use instances, while PCB and Schematic projects default to occurrences.

p olygon

A graphic object made up of polylines (multiple contiguous segments) whose beginning and end are attached to form a closed shape that can be filled.

polyline

A line with multiple contiguous segments. You place a polyline using the Polyline command on the Place menu.

port

A VHDL term for an interface element of an ENTITY. A port serves as a communication channel between VHDL design units. A part pin on an OrCAD Capture schematic page generates a VHDL port. See hierarchical port.

primitive

A part or hierarchical block with no underlying hierarchy.

programmable logic device
A type of integrated circuit whose behavior can be determined by programming it. Abbreviated PLD.

project
An OrCAD project file (.OPJ) includes references to all of the resources you use throughout the design process. These resources including elements that define design structure (VHDL source files, schematic folders, etc.), as well as part libraries, test benches, stimulus files, simulation models, vendor files, and standard delay files. You can view these resources in the project manager.

project manager
The project manager is a tool that allows you to collect and organize all the resources you need for your project throughout the design flow. These resources include schematic pages, part libraries, and netlists, and may also include VHDL models, simulation models, timing files, stimulus files, and other related information.

PROPAGATION_DELAY
Defines the minimum and maximum propagation delay constraint between any pair of pins in a net. By assigning this property to nets, you can make the router restrict the length of interconnect to meet timing margin. This property often is best applied to a common clock sourced designed bus.

property
A characteristic of an object that can be edited. A property consists of a name and a value. Examples of property names are part value and color. Their respective property values can be something such as capacitor and red.

R

radix
The number base in which a signal value is displayed: binary, octal, signed or unsigned decimal, or hexadecimal.

RAM
Abbreviation for Random Access Memory. This is the memory that can be used by applications to perform necessary tasks while the computer is on. When you turn the computer off, all information in RAM is lost.

random access memory
The memory that can be used by applications to perform necessary tasks while the computer is on. When you turn the computer off, all information in random access memory is lost. Abbreviated RAM.

RATSNEST_SCHEDULE
Specifies the type of ratsnest calculation that Constraint Manager performs on the net. By using the RATSNEST_SCHEDULE property, you can meet a balance between time margin and noise margin. This property is useful for defining the placement of receiver or driver in multi-drop buses and asynchronous signals.

recursive design
A hierarchical design in which a schematic folder in the hierarchy is attached to a part instance or hierarchical block placed "higher" in the hierarchy. The simplest case of recursion is some schematic folder X containing a part instance or hierarchical block to which schematic folder X is attached.
reference designator
The designator, or identification code, for a component. A reference designator uniquely identifies a part in a design. For uniquely identifying parts, you can use the Annotate command on the Tools menu. For PCB designs, the Annotate tool assigns individual parts to a package and assigns unique pin numbers to each part in a multiple-part package. References are assigned in order from top to bottom and left to right; parts located at the top of the page have the lowest numerical designation. If two parts share a vertical coordinate, the part further to the left has the lower numerical designation.
The format for reference designators should never be changed as <Alphabet(s)><Numeric><Alphabet(s)> or <Alphabet(s)>-<Alphabet(s).
RELATIVE_PROPAGATION_DELAY
An electrical constraint attached to pin-pairs on a net. It specifies a group of pin-pairs that are required to have interconnect propagation delays matching a specified delta (offset) and tolerance with respect to the target pin pair. You can apply the RELATIVE_PROPAGATION_DELAY property to a source synchronous bus design, such as DDR interfaces.
root schematic folder
The schematic folder at the top of a flat design or hierarchical design. The root schematic folder contains a backslash (\ ) in its icon in the project manager. A project has only one root schematic folder.

S

scalar
A pin width that carries only one signal, as opposed to a bus pin that can carry multiple signals.
schematic folder
A collection of the schematic pages at the same level of hierarchy in a design is contained within a schematic folder, which is shown in the project manager. See also flat design, hierarchical design, schematic page, root schematic folder.
schematic page
A page within a schematic folder on which a design is drawn. Schematic pages display in a window called the schematic page editor, in which you can place parts and draw wires.
schematic page editor
The editor used to create and edit schematic page.
SDF
Standard Delay File. This is a file containing delay values that relate design performance after place-and-route. You can add this file to your simulation project in order to perform timing analysis.
session frame

1. The Capture application window in which the various components of Capture—such as the session log, project manager, schematic page editor, and part editor—run.

2. The Simulate application window in which the various components of Simulate—such as the session log, wave windows, list windows, watch window, and project window—run.

   session log

   A window that displays text messages generated by Capture, such as errors and informational messages. The session log starts empty with each new Capture session, but you can save its contents to a text file.

   setup time

   The length of time for which data must be stable at a pin before being clocked into the device.

   signal

3. An electrical impulse of a predetermined voltage, current, polarity, and pulse width.
4. The logical state that exists on a circuit node.
5. A VHDL term for a local circuit node that is not visible outside a VHDL design unit. A bus or wire on an OrCAD Capture schematic page that is not connected to a hierarchical port produces a signal.

   signal contention

   A condition that occurs when a circuit node is driven by multiple conflicting sources at the same time. In most circuit nodes, output-type ports fan out to drive multiple input-type ports. However, some networks are constructed such that it is possible for multiple drivers to drive a single node. Simulate uses MVL-9 signal contention resolution to resolve these conflicts.

   signal context

   The level of hierarchy at which a signal or port exists.

   simple hierarchy

   A project in which there is a one-to-one correspondence between hierarchical block (or parts with attached schematic folders) and the schematic pages they reference. Each hierarchical block (or part with attached schematic folder) represents a unique schematic page. See also hierarchical design, complex hierarchy.

   simulation model

   VHDL descriptions of the behavior of primitive components in your design. Typically, the simulation models for your design will exist in a single VHDL file, but they may also exist within the netlist file or in several different model files. Simulation models are necessary elements in an Simulate project.

   simulation project

   A simulation project is a collection of the resources you need to simulate your design. Generally, a simulation project requires the following elements: a netlist, a set of simulation models, and a set of stimuli. In addition, your simulation project may include timing annotation files after it has been through the design implementation process.

   simulation resolution

   The amount of time that represents one "step" in a simulation run. Simulate has two

resolution settings: nanoseconds (the default) and picoseconds.

source library
The path and filename of the part definition. A filename with an .OLB extension means that the part was placed as is from a library. A filename with a .DSN extension means that the part no longer match the original library definition and its current definition only resides in the design file where it was edited.

spreadsheet editor
A window used to edit the properties of multiple objects at once.

split part
A part that consists of a package in which pins are split across multiple sections.

static timing analysis
A process that inspects the layout of a PLD or FPGA design to estimate the timing characteristics of the manufactured device. Typically, static timing analysis generates a delay annotation file for a digital simulator.

stimuli
Signal states that are applied to nodes in an electronic design in order to view the effects of those states on circuit behavior. There are three types of stimuli in Simulate: forces, patterns, and clocks.

subprogram
A term used to refer, collectively, to VHDL functions and procedures.

symbol
The graphical object that represents a part on a schematic page.

T

tabbed dialog box
A dialog box that has different views you can display by clicking on tabs at the top of the dialog box.

test bench
A VHDL module that defines the interface to one or more designs under test, applies input vectors, and (optionally) generates reports about the output behavior of the design(s) under test. A test bench ENTITY does not provide communication ports; therefore, test benches are usually used exclusively by VHDL simulators.

timing analysis
Simulation that identifies timing problems in the design. Timing analysis is performed after design implementation.

timing annotation file
A file containing delay values associated with the implementation of a design. In general, timing annotation files are produced from place and route tools.

timing violation
A simulation condition indicating that the timing constraints for a device have been violated. Simulate detects timing violations via the error trapping of VITAL VHDL models.

tri state enable delay
The length of time required for a tri-state device to transition from a Z state to a 0 or 1 once an enable has been received.
True Type
A font (typeface) that appears in a printout exactly the way it appears on the screen. TrueType fonts are scalable to any font size, and several of these type of fonts are installed automatically when you install Windows.
twos complement
An alternate method for representing a binary value. Two's complement allows positive and negative values to be represented in the same format and thus enhances arithmetic operations. The most significant bit of a two's complement value is the sign bit: a "0" indicates a positive value; a "1" indicates a negative value. The two's complement of a value is derived by inverting each bit in that value, then adding 1 to it. Thus, the binary value 0111 (representing +7) becomes 1000+1, or 1001 (representing -7).

U

user defined property
A property[you add to an object. Unlike |Glossary#1026858]inherent properties, user-defined properties can be removed.

V

vertex
The point at which the sides of an angle meet. You create this by drawing a wire or line in one direction, then changing direction to create an L-shaped or V-shaped wire or line.
VITAL
VHDL Initiative Toward ASIC Libraries. An informal consortium formed to accelerate the development of ASIC macrocell simulation libraries modeled with VHDL.

W

waveform pattern
A set of events that occur on a signal, relative to a specific simulation time. A waveform pattern may or may not be repeating.
wildcard
A symbol, usually used in searches, that represents a missing or unknown character or sequence of characters. Valid wildcard characters are an asterisk (*) to match multiple characters and a question mark (?) to match individual characters.

X

X axis
The horizontal or left-to-right direction in a two-dimensional system of coordinates. The X axis is perpendicular to the Y axis.

XNF
Xilinx Netlist Format. This is a netlist format generated by Xilinx design implementation tools. You must convert XNF files to VHDL format before you can use them with Simulate.

Y

Y axis
The vertical or bottom-to-top direction in a two-dimensional system of coordinates. The Y axis is perpendicular to the X axis.

Z

zoom
To change the view of a window, making objects appear larger or smaller. When you zoom out, objects are smaller, and you see more of the schematic page, part, or waveform pattern. When you zoom in, objects are larger, but you only see a small portion of the schematic page, part, or waveform.
zoom factor
The amount by which the zoom scale is multiplied or divided when you choose Zoom In or Zoom Out on the View menu. The Zoom factor is normally 2, but you can change it using the Preferences command on the Options menu. For example, a zoom scale of two makes the image on the screen twice as large when you zoom in and half as large when you zoom out. You can also zoom in or out of a print preview.
zoom scale
The relative size of the image on the screen, as a percentage of the normal size. For example, a zoom scale of 250% means the image on the screen is two and one-half times as large as normal.