



OrCAD X Capture User Guide

Product Version 23.1
September 2023

© 2023 Cadence Design Systems, Inc.
Printed in the United States of America.

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

Open SystemC, Open SystemC Initiative, OSCI, SystemC, and SystemC Initiative are trademarks or registered trademarks of Open SystemC Initiative, Inc. in the United States and other countries and are used with permission.

Trademarks: Trademarks and service marks of Cadence Design Systems, Inc. (Cadence) contained in this document are attributed to Cadence with the appropriate symbol. For queries regarding Cadence's trademarks, contact the corporate legal department at the address shown above or call 800.862.4522.

All other trademarks are the property of their respective holders.

Restricted Permission: This publication is protected by copyright law and international treaties and contains trade secrets and proprietary information owned by Cadence. Unauthorized reproduction or distribution of this publication, or any portion of it, may result in civil and criminal penalties. Except as specified in this permission statement, this publication may not be copied, reproduced, modified, published, uploaded, posted, transmitted, or distributed in any way, without prior written permission from Cadence. Unless otherwise agreed to by Cadence in writing, this statement grants Cadence customers permission to print one (1) hard copy of this publication subject to the following conditions:

1. The publication may be used only in accordance with a written agreement between Cadence and its customer.
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.

Disclaimer: Information in this publication is subject to change without notice and does not represent a commitment on the part of Cadence. Except as may be explicitly set forth in such agreement, Cadence does not make, and expressly disclaims, any representations or warranties as to the completeness, accuracy or usefulness of the information contained in this document. Cadence does not warrant that use of such information will not infringe any third party rights, nor does Cadence assume any liability for damages or costs of any kind that may result from use of such information. Cadence is committed to using respectful language in our code and communications. We are also active in the removal and replacement of inappropriate language from existing content. This product documentation may however contain material that is no longer considered appropriate but still reflects long-standing industry terminology. Such content will be addressed at a time when the related software can be updated without end-user impact.

Restricted Rights: Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor.

Contents

1		17
OrCAD X Capture		17
Launching OrCAD X Capture		18
Using Windows Start Menu		18
Using Command Line Arguments and Switches		21
Capture Custom Launch		22
OrCAD X Capture Start Page		23
2		27
OrCAD X Capture Interface		27
Project Manager		27
Session Log		30
Command Window		31
Property Spreadsheet Editors		31
The Property Editor Window		31
Changing the Appearance of Property Editor		34
		34
Browse Spreadsheet Editor		35
Package Properties Editor		37
Component Explorer		37
File Manager		39
User Interface Customizations		41
Capture Configuration		46
User Interface Configuration Settings		46
Capture.ini File		47
Customizing Menus and Toolbars		47
Capture.ini Variables		48
SearchToolBarSetting		49
Search Toolbar Options		49
Docking		49
Print Settings		50
Preferences		50
Footprint Viewer Type		50

Allegro Footprints	50
Part Management	50
Frame View Options in CIS Explorer Window	51
ConvertDialog	51
OrCAD X Capture Toolbars	51
3	54
Working with Projects	54
Creating a Project	54
Creating a New Project	55
Setting up New Project for Simulation	57
Setting up an Existing Project for Simulation	57
Setting Project Preferences	58
Setting up the Design Template	72
Defining Fonts for New Designs	72
Setting up a New Page	75
Defining Title Block	78
Defining Grid References	83
Setting up Default Hierarchy	85
Setting SDT Compatibility	86
Moving Objects by Drag and Drop	92
Capture Directory Map	93
Opening a Project	96
Saving a project, design, or library	97
Closing a Project	100
4	101
Managing Workspaces	101
Configuring Workspaces	102
Creating a Workspace	102
Setting up a Default Workspace	105
Sharing Workspaces	109
Managing Workspace Files	113
Adding a Local Project to a Workspace	114
Viewing Status of Project Files in a Workspace	114
Refreshing File Status	116
Viewing Projects in File Browser	117
Sharing Projects	121

Editing Shared Projects	126
Publishing Shared Projects	128
Restoring a Revision	130
5	133
Working with Designs	133
Creating a Design	133
Opening a Design	136
Creating a new VHDL or Verilog file	137
Creating a Text File	138
Using Non-Electrical Objects	138
Creating Graphics	139
Drawing Arcs	140
Drawing Bezier Curves	141
Drawing Ellipses and Circles	142
Drawing Elliptical Arcs	143
Drawing Lines	144
Drawing Polylines	145
Drawing Rectangles and Squares	146
Placing OLE Objects	147
Placing Pictures	149
Placing Text	149
Placing Text in Schematic Page Editor	149
Placing Text in Part Editor	151
Placing IEEE Symbols	152
Flat and Hierarchical Designs	153
Flat designs	153
Hierarchical Designs	154
Editing Hierarchical block look and feel	156
Browsing in Capture	157
Searching in Capture	161
Locating an Object in a Project	161
Searching a Design Hierarchy	163
Find Pane	165
Find Results Window	165
Renaming a Design	167
Saving and Closing a Design	167

Compare Designs	168
Exporting and Importing a Design from XML	176
6	178
Working with Schematic Folders	178
Creating a Schematic Folder	178
Deleting a Schematic Folder	179
Renaming a Schematic Folder	179
Moving Schematic Folders	180
Attaching a Schematic Folder	181
7	184
Working with Schematic Pages	184
Creating a Schematic Page	184
Defining Schematic Page Characteristics	185
Schematic Page Editor and Part Editor	186
Moving Around in the Editors	188
Working with Objects	188
Scrolling	189
Panning	189
Moving to a Location	190
Using Zoom	193
Setting a Bookmark	197
Magnifying Objects Using Fisheye	198
Selecting and Deselecting Objects	201
Copying Objects	205
Mirroring Objects	207
Moving Objects	207
Rotating Objects	210
Working with Label States	211
Working with VHDL and Verilog files	212
Creating a VHDL Model from a Hierarchical Block	213
Creating a Verilog Model from a Hierarchical Block	215
Checking the syntax of VHDL or Verilog files	216
Moving Schematic Pages	218
Renaming a Schematic Page	220
Deleting a Schematic Page	221
Closing and Saving a Schematic Page	221

8		222
Working with Title Blocks		222
Setting up the default title block		225
Creating a custom title block		227
Placing Multiple Title Blocks		228
9		230
Working with Properties		230
Push Occurrence Properties into Instance Utility		231
Defining Properties		235
Adding a User-Defined Property		235
Creating a New Property in the Browse Spreadsheet Editor		236
Filtering Properties		237
Related Topics		240
Editing Properties		240
Importing Part and Pin Properties		245
Exporting Part and Pin Properties		246
10		253
Managing Projects		253
Archiving a Project		253
Auto Recovery		258
Adding and Deleting Project files		260
Undoing and Repeating commands		261
Graphical Operation (GOp) Locking		264
11		271
Establishing Connectivity		271
Wire connectivity		271
Placing and naming wires		273
Placing junctions		276
Junction Dot Formation		277
Auto-Wiring in Capture		281
Auto-Wire two points		281
Auto-Wire Multiple Points		282
Auto-Wire to Bus		283
Auto-Wire to NetGroup		284
Bus connectivity		286

Naming Conventions	291
Modifying Wires and Buses	291
Labeling wires and buses	292
Editing wire and bus look and feel	293
Moving Connectivity Objects	295
Deleting wires and buses	299
Shorting Part Pins	299
Placing power, ground, and no connect symbols	299
No connect symbols	300
Power and ground symbols	301
Working with Power Pins	303
Browsing for Power Pins	304
Making power pins visible	306
Placing off-page connectors	308
Adding hierarchical ports	310
Establishing connectivity in schematic pages	313
Using intersheet references	319
Creating Intersheet references	320
Guidelines for Creating Intersheet References	321
Intersheet references in a flat design	324
Intersheet references in a hierarchical design	325
Reporting Intersheet References	327
Signal Navigation in Capture	328
Working with nets	329
Assigning net aliases	329
Net operations	331
Tracing a net	333
Using NetGroups	336
Introducing NetGroups	336
Named NetGroup	337
Unnamed NetGroup	346
Components of a NetGroup Block	351
Netlising NetGroup Designs	352
NetGroup Connectivity	353
NetGroup and Bus Member Net Generation	358
Net Generation Scenarios	359

12		369
Working with Capture TCL		369
Using the Capture Command Window		369
Registering and Logging TCL Commands		370
Running a TCL Command		371
Creating a Capture TCL Script		372
TCL Script for all Capture Commands		372
TCL Script for Specific Capture Commands		374
Replaying a Capture TCL Script		374
Running TCL Script in Capture		375
Running TCL Script in Windows		375
13		377
Design Annotation		377
Customizing Part References in a Design		378
Backannotating		384
Selecting Annotation Sequence		386
Swap File		390
Designating Pins, Gates, or Packages for Swapping		393
Update File		394
Creating a Combined Swap and Update File		397
Advanced Annotation		399
Reference Range Assignment		399
Selective Annotation		400
Property Block Based Annotation		403
14		406
Generating Reports		406
Creating an Include File		406
Creating a Bill of Materials		407
Creating a Cross Reference Report		410
Creating a Placement Report		411
Creating a Find Results Report		412
Check and Save		413
15		414
Printing and Plotting		414
Printing or Plotting a Part		414

Setting Print Options for Schematic Page Objects	415
Printing Documents	416
Printing or Plotting a Schematic Page	417
Previewing Print Output	418
Printer or Plotter Setup	419
Scaling a Print or Plot	420
16	421
Netlisting a Design	421
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.	422
Net name resolution	422
Create Netlist dialog box tabs	424
Creating a Netlist	425
Creating a Flat Netlist	426
Generating a Layout Editor Netlist	427
	429
Working with Hierarchical Netlists	429
Creating a Hierarchical Netlist	429
Creating Subcircuit Netlists	431
Using SUBPARAM	432
Specifying an Alternate Netlist Template	436
17	437
Simulating Capture Designs Using PSpice	437
Overview of Simulation Using PSpice	437
Creating Design for Simulation	438
Adding PSpice and Parametrized Libraries and Parts	439
Special simulation-only parts	440
Vendor-supplied parts	440
Passive parts	443
Breakout parts	443
Behavioral parts	444
Defining Part Properties Needed for Simulation	445
Editing simulation properties	446
PSPICETEMPLATE property	446
IO_LEVEL property	450
MNTYMXDLY property	451

PSPICEDEFAULTNET property	451
Guidelines and Best Practices for Specifying Values for Part Properties	452
Placing PSpice Ground 0 Symbols for PSpice Simulations	452
Using the FLOAT Property for Unconnected Pins	453
Using Global Parameters and Expressions	454
Global parameters	454
Expressions	456
Associating PSpice Model to Capture Parts	457
Viewing PSpice Models	462
Viewing Information on Missing PSpice Models in Online DRC	463
Importing PSpice Schematic Projects in Capture	464
Defining Stimuli	465
Analog stimuli	465
Digital stimuli	468
Using Partial Design Simulation	468
Working with a Test Bench	471
Comparing and Updating Master Design	474
Running a Simulation and Viewing Results Using PSpice	475
Creating a new simulation profile	475
Creating a simulation netlist	476
Viewing a simulation netlist	477
Running a simulation	477
Viewing the results as the simulation progresses	478
Viewing the most recent simulation results	478
Viewing the output file	478
Editing simulation settings	478
Placing markers	479
Showing, hiding, and deleting markers	479
Simulating and viewing the results of multiple profiles	480
Making a simulation profile active	481
Imported Parts in Schematic Design	481
Imported Model of a Restricted Model Type	482
Simulation Scenarios with Pre-Installed or Imported Models	482
Utilizing Unused Markers for Data Collection	486
Model Parameters for Specific Devices	487
Files Needed for Simulation	487
Files that Design Entry Programs generate	488

Netlist file	488
Other files that you can configure for simulation	488
Files that PSpice generates	490
Probe data file	490
PSpice output file	490
Running PSpice in Batch Mode	491
Interactive Mode	491
Non-Interactive Mode	491
SPICE netlist format	492
18	495
Physical Layout in a Layout Editor	495
Preparing the Schematic for Layout	496
Best Practices for Schematic-Layout Flow	496
Assigning Physical Properties to a Schematic Design	498
Working with Footprints	499
Custom Footprints in Design Flow	500
Assigning Footprint Properties	501
Assigning Footprints to Components	502
Specifying Alternative Footprints	503
Viewing Footprints	504
Instance-Level Properties for Physical Design	508
ROOM Property	508
Defining Properties on Nets	509
Signal Property Flow	510
Limitations of the Signal Property Flow	512
PROPAGATION_DELAY	512
RATSNEST_SCHEDULE	514
RELATIVE_PROPAGATION_DELAY	515
DIFFERENTIAL_PAIR	518
The Voltage Property	525
Properties on Power Pins	525
POWER_PINS property	525
Assigning Power Nets to Invisible Power Pins	526
Using the POWER_GROUP property	528
Assigning No Connect Pins in Capture-PCB Editor Flow	531
Assigning No Connect Pins	531

Assigning No Connect Power Pins	533
Property Flow from Schematic to Layout Editor	534
Layout Editor Configuration File	534
User-Defined Properties in Capture - PCB Editor Flow	537
Creating PCB Editor Board	539
Setting up Advanced Options for PCB Flow	542
Processing Changes after Board Creation (ECO)	544
Cross Probing for PCB Editor	546
Pin Swapping In Capture-PCB Editor Flow	549
Back Annotation from PCB Editor	557
Best practices for smooth back annotation	561
Physical Layout of a Simulation Design	562
Design Reuse for a Layout Editor	563
Creating a Reuse Design	566
Reusing a Design	569
Running Design Rules Check - Physical Rules	571
Validating Design for Physical Layout	571
Browsing DRC Markers	572
Launching PCB Viewer from Capture	573
19	576
Using Capture with PCB SI	576
Setting up a Library	576
Assigning Models	578
Managing Electrical Csets	580
Concurrent Mode	581
Distributed Mode	582
Validating Electrical Csets	583
20	584
Managing Toolbars	584
Displaying a Toolbar	584
Moving a Toolbar	584
Align Toolbar	585
Capture Toolbar	587
Draw Electrical Toolbar	590
Draw Graphical Toolbar	592
Part Manager Toolbar	592

PCB Toolbar	593
PSpice Toolbar	594
SI Analysis Toolbar	596
Customizing Toolbars	598
Docking Toolbars	600
21	602
Shortcut Keys	602
All Capture windows	602
Text editor	603
Schematic page editor	603
Part editor	605
Property editor	605
Schematic page and part editors	606
Session log	608
Text boxes	609
Browse spreadsheet editor	610
Fisheye View	610
22	611
PCB Editor Netlist Files	611
PSTCHIP.DAT	611
PSTCHIP File format	612
PSTCHIP File Elements	612
PSTCHIP Sample file	614
PSTXNET.DAT	615
PSTXNET File Elements	616
PSTXNET File format	617
PSTXNET Sample file	617
PSTXPRT.DAT	621
PSTXPRT File Elements	621
PSTXPRT File format	622
PSTXPRT Sample file	623
23	627
SDT Configuration Files	627
Path and libraries	627
Page dimensions and units	628

Part fields	628
24	629
Netlist Examples	629
Accel netlist format	630
Algorex netlist format	631
Altera ADF netlist format	632
AppliconBRAVO netlist format	636
AppliconLEAP netlist format	637
Cadnetix netlist format	638
Calay90 netlist format	639
Calay netlist format	639
Case netlist format	640
CBDS netlist format	641
Computervision netlist format	642
DUMP netlist format	643
EDIF 2 0 0 netlist format	643
EEDesigner netlist format	651
Futurenet netlist format	652
HiLo netlist format	656
Intel ADF netlist format	658
Intergraph netlist format	660
Mentor netlist format	661
Multiwire netlist format	662
OHDL netlist format	663
PADS 2000 netlist format	665
PADS PCB netlist format	666
PCAD netlist format	668
PCADnlt netlist format	669
PCBII and PCBIIIL netlist formats	671
PDUMP netlist format	671
PLD netlist format	671
Protel2 netlist format	672
RecalRedac netlist format	674
RINF netlist format	675
Scicards netlist format	676
Tango netlist format	677

Telesis netlist format	678
Vectron netlist format	679
Verilog netlist format	680
VHDL netlist format	687
VST Model netlist format	693
WinBoard netlist format	694
WireList netlist format	695
25	698
Glossary	698
A	698
B	699
C	700
D	702
E	703
F	704
G	706
H	706
I	707
J	707
L	708
M	708
N	708
O	709
P	709
R	713
S	714
T	716
U	717
V	717
W	717
X	717
Y	717
Z	718

OrCAD X Capture

OrCAD® X Capture is a schematic design toolset for the Windows environment. With OrCAD X Capture or Capture, you can draft schematics and produce connectivity and simulation information for printed circuit boards and programmable logic designs. OrCAD X Capture is fully integrated with a number of tool suites, including PSpice and PCB board layout toolset including OrCAD X Presto.

With OrCAD X Capture, you can:

- Create schematics and produce connectivity and simulation information for printed circuit boards and programmable logic designs.
- Create designs for other EDA applications by choosing to set up a PSpice project at the start of creating a new project.
- Set user preferences for the appearance of all the designs on your system.
- Set up design options for each particular project or design you create.
- Use standard libraries to design schematics.
- Place parts in the OrCAD X Capture schematic editor.
- Create your own parts and part packages.
- Create your own library.
- Connect the parts with buses, wires, off-page connectors, and more.

OrCAD X Capture also includes verification and reporting, printing, and netlisting features for a schematic page, a folder, or an entire design.

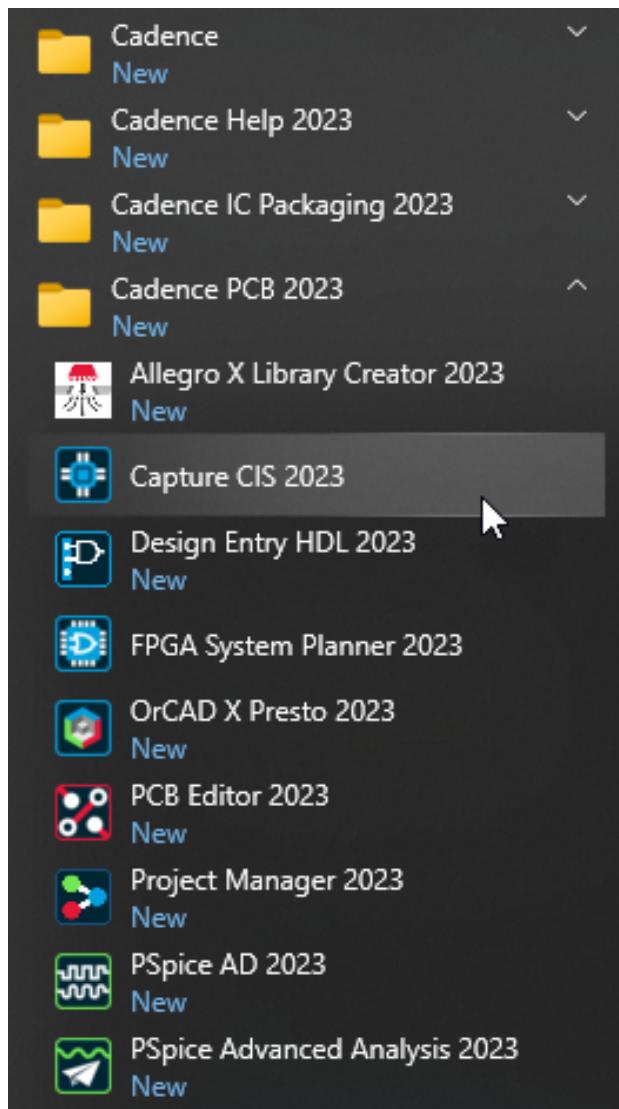
Launching OrCAD X Capture

This topic explains how to launch OrCAD X Capture. You can launch Capture using the Windows Start menu or [command line](#).

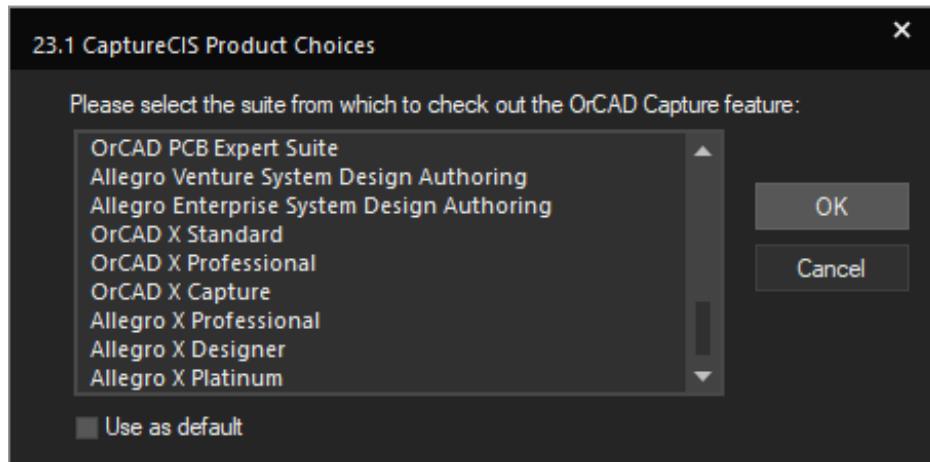
Using Windows Start Menu

To start Capture from the Windows Start menu, do the following:

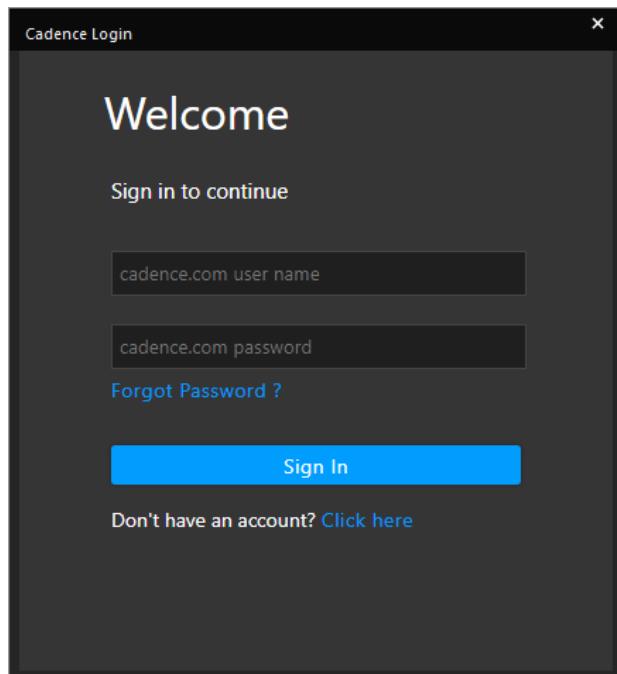
1. Choose *Start – All Programs – Cadence PCB <release> – Capture CIS<release>*.



The *23.1 CaptureCIS Product Choices* dialog box is displayed.



2. Choose a product license, such as *OrCAD X Standard* or *OrCAD X Professional* and click *OK*.
3. Choose *Use As Default* to avoid having to choose a license at each launch.
If you choose *OrCAD X Standard* or *OrCAD X Professional*, you are prompted to specify your login credentials.



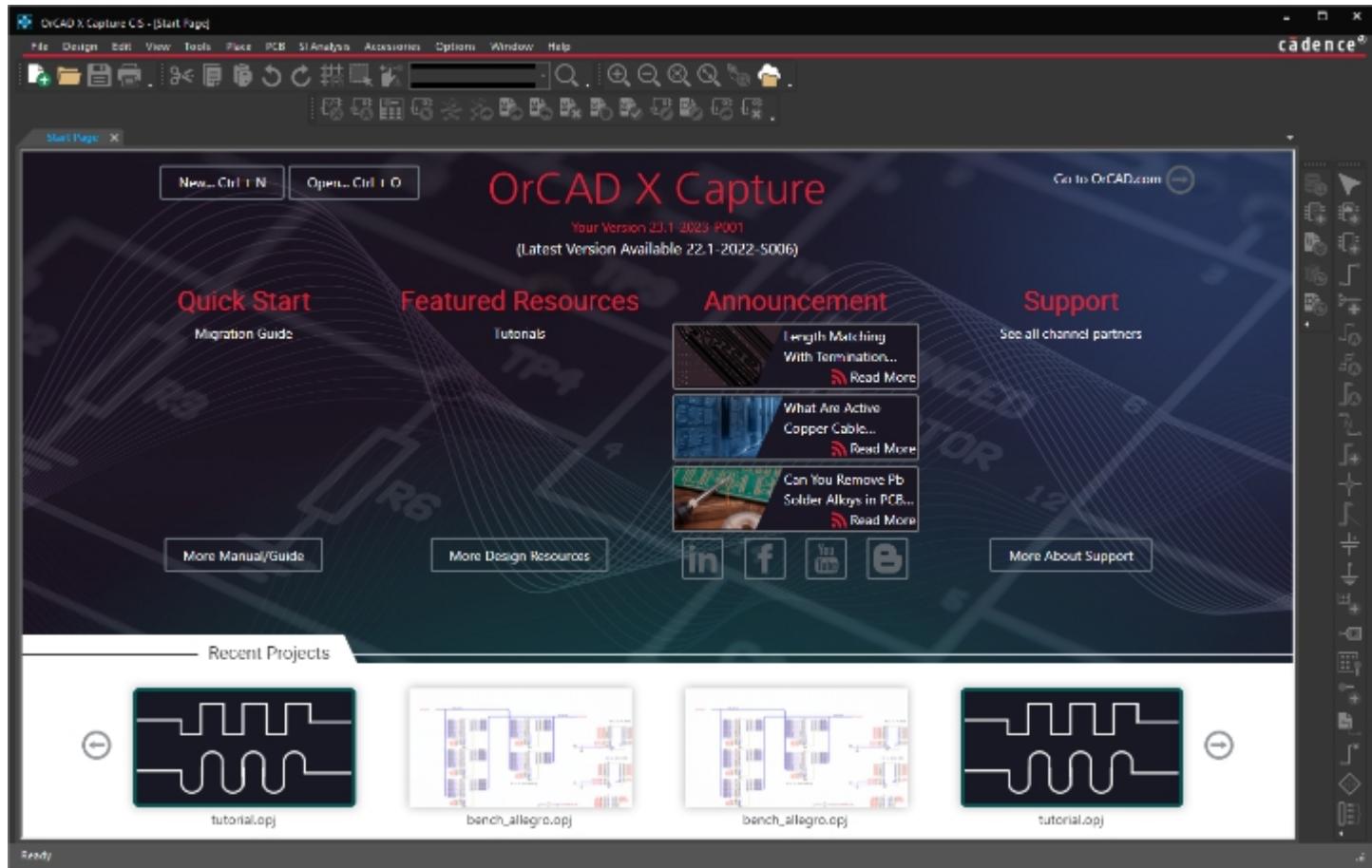
4. Specify user name and password provided by Cadence and click *Sign In*.

With the OrCAD X Professional (POX200 Pro) and OrCAD X Standard (POX100 Standard) licenses, you can seamlessly access your OrCAD X Capture designs, libraries, components, and other design elements across devices based on your OrCAD X Cloud login id. For example, you can start creating a Capture project on your laptop using the desktop application and then continue working on the same design files from a different device. By default, the designs are created at the following location in the user's %HOME% folder:

```
%HOME%\cdssetup\workspace
```

For the same login credentials, design data is automatically synchronized with OrCAD X Cloud across devices to maintain consistency. Using [File Manager](#) you can manage all the project and library files stored on the Cloud server.

Capture is launched and the *Start Page* is displayed.



After the login credentials are authenticated for the OrCAD X license, you can access all the OrCAD X-enabled features.

Using Command Line Arguments and Switches

You can launch OrCAD X Capture directly from the command line with specific arguments and switches.

- To launch Capture from command line, type `capture` along with the required arguments in the command window and press `Enter`.

Syntax:

```
capture [-version|-viewer|-product=<license product display name>] [-i <capture.ini location>]
```

Switch / Argument	Description
<code>-i <capture.ini location></code>	Invokes an instance of Capture with respect to the Capture.ini file.
<code>-viewer</code>	Invokes an instance of Capture Viewer.
<code>-product=<license product display name></code>	Invokes an instance of Capture with respect to license product display name given.
<code>-version</code>	Displays Capture's version number and version release date.

The following combinations are valid for different arguments or switches:

- `capture -version`
- `capture -viewer -i <capture.ini location>`
- `capture -product=<license product display name> -i <capture.ini location>`

Related Topics

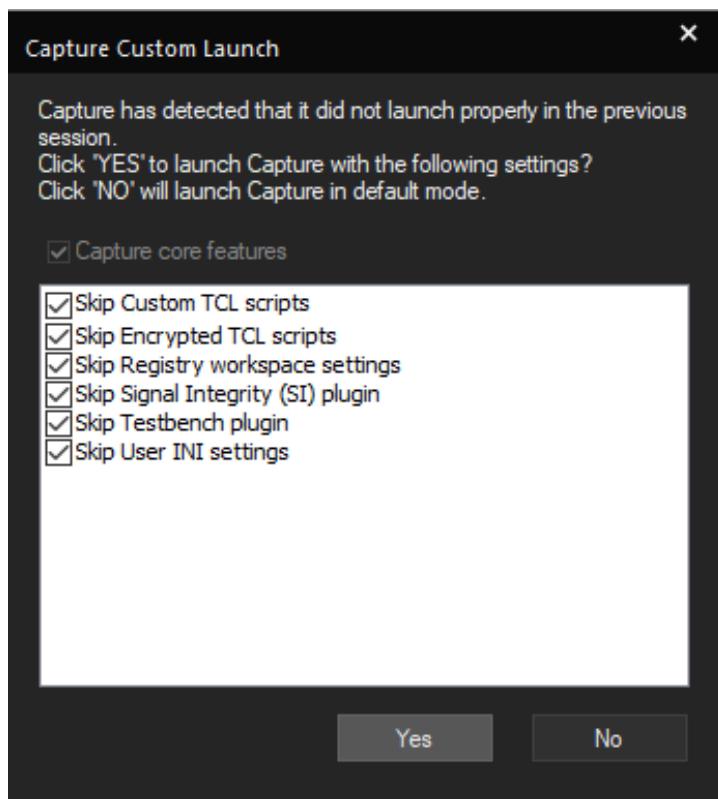
- [OrCAD X Capture Start Page](#)
- [Creating a Project](#)

Capture Custom Launch

You can relaunch OrCAD X Capture in a custom launch mode if Capture fails to launch for the first time. You can select different options to skip reading and initializing various settings and registry options in the next Capture launch for safety purpose.

You can skip any of the following options for the custom launch:

- Launching Capture with custom TCL scripts that are initialized at the start
- Launching Capture with encrypted TCL scripts that are initialized at the start
- Reading the user INI settings
- Reading the registry workspace settings
- Reading the Signal Integrity plugin
- Reading Testbench plugin



- Click *Yes* to enable custom launch. If you click *No*, Capture is launched with all the specified options.

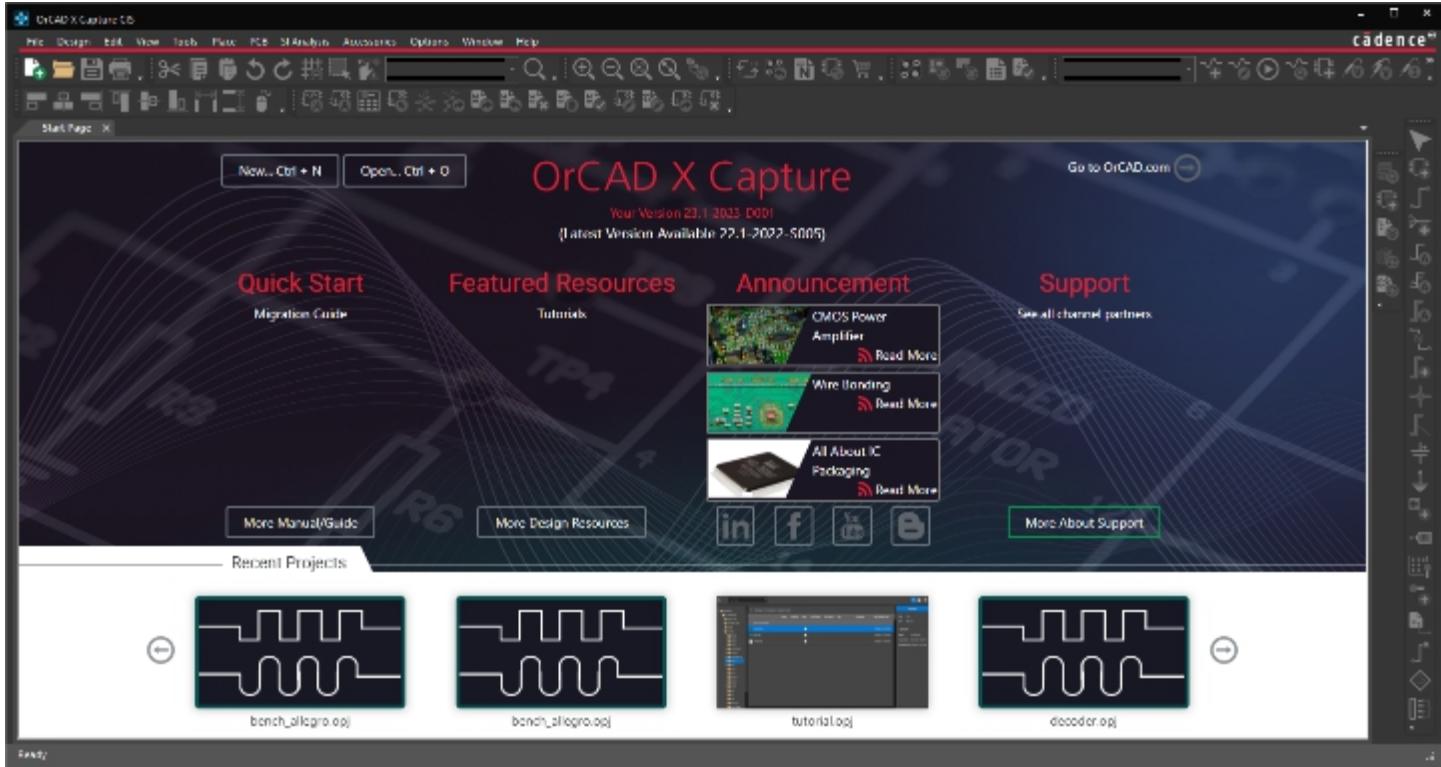
⚠ The Capture Custom Launch window is displayed only after Capture fails at launching.

OrCAD X Capture Start Page

When you open OrCAD X Capture, you see a content-rich, well-organized start page. From the start page, you can access a variety of information and projects. From this page, you can read about OrCAD X Capture, go through brief descriptions of the available features, and access quick start guides and video walkthroughs. You can also access help content, product announcements, and industry news. The page provides contact information for connecting to your local channel partners or Cadence customer support.

You can also view the version of OrCAD X Capture installed on your system and available newer versions. To download and install the latest version, launch Cadence® Download Manager or visit

the [Downloads](#) site.



Sections in Start Page

Section	Description
Quick Start	Includes link to Migration Guide.
Featured Resources	Includes links to product Tutorials.

Announcement	Includes links related to the EDA industry and new trends in the OrCAD X space.
Support	Provides information related to the Cadence contact or regional Cadence Channel Partners.
Recent Projects	Displays the projects that you recently opened in OrCAD X Capture. Click the arrow buttons in this section to browse through the projects. To view the thumbnail view of the design in this section, the design needs to be saved.
Reopening Start Page	You can open the start page from <i>Help – Start Page</i> .

Enabling or Disabling Start Page

You can disable the Start Page from appearing when the tool launches. To do so, use one of the following methods. However, after every hotfix, the Start Page will appear once before these settings apply.

1. Add the following statement in the *Preferences* section of the `Capture.ini` file:

```
EnableStartPage=False
```

2. Enter the required command in *Command Window* of Capture.

The associated TCL command to disable the Start Page is:

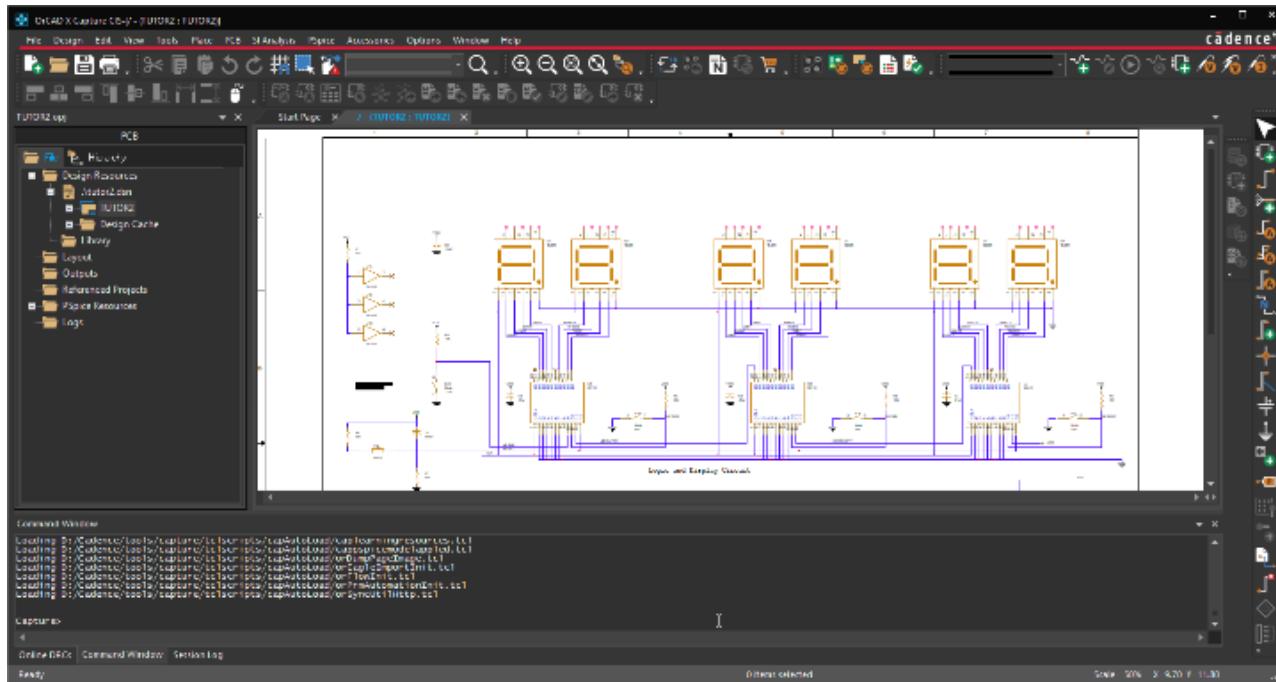
```
SetOptionBool EnableStartPage FALSE
```

The associated TCL command to enable the Start Page is:

```
SetOptionBool EnableStartPage TRUE
```


OrCAD X Capture Interface

OrCAD X Capture is a graphical user interface based application used for schematic design. This application provides you with a large set of user-friendly tools and features to easily create your schematic design. It also has a well-defined work area that ensures you can work on your schematic design in the most optimized manner.

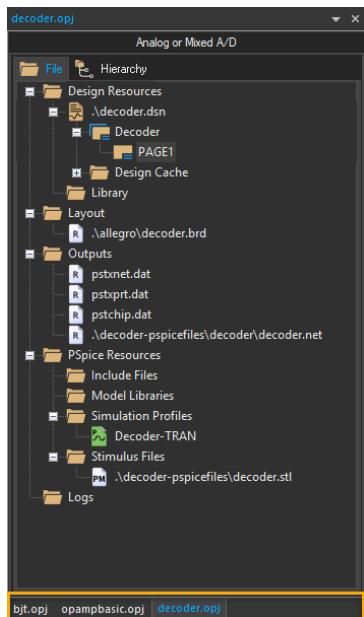


- The project manager provides a hierarchical (Windows) File Explorer type access to the resources in an OrCAD X Capture design.
- The schematic page and part editors provide a consistent and easy way to create and modify pages and parts that you require to build your design.
- The Session Log window logs the application messages.
- The Online DRCs window checks and lists the violations of design rules as you create or update the schematic design. When you double-click the violation message, 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 or Verilog models.
- The user interface also includes a set of menus that contain menu commands that you use to perform various operations in Capture. In addition to the menus, Capture also provides a set of toolbars that provide shortcuts to many of the commonly used commands.

Project Manager

The project manager window appears as a floating, dockable, or tabbed panel on the left hand side of the OrCAD X Capture application window when you open or create a project. The project manager window displays and organizes all the resources used in a project throughout the design flow.

These resources include schematic design files, part libraries, netlists, VHDL or Verilog models, simulation models, timing files, stimulus files, layout files, and any other related information. You can access all the files from within the project manager window.



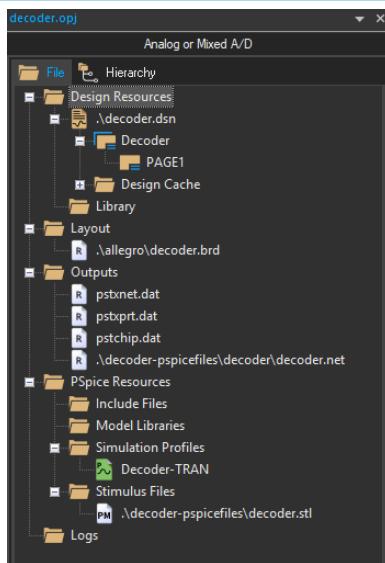
Project Manager Tabs

The project manager provides the following tabs to view a project:

- File
- Hierarchy

File

The *File* tab is organized into folders and displays all the files included in the project including the design file, the libraries, VHDL models, netlists, schematic pages, simulation models, stimulus files, or any other files that contain information related to the project.



The files in the *File* tab of the project manager are organized in related **folders** including *Design Resources*, *Layout*, *Outputs*, *PSpice Resources*, and so on.

The shortcut menu on each of these folders provides the commands to perform the following functions:

- Add a related file to the project. For example, use the *Add File* shortcut menu on *Design Resources – Library* to add a library file to the project.
- [Save the project](#) with a different name and settings. This can also be accessed from *File – Save Project As*.
- Open the Part Manager window that summarizes the status of all the parts in a design.

i Design Resources

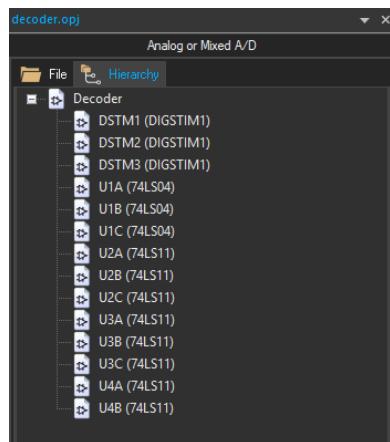
The *Design Resources* folder contains the design (.dsn) and all the design related resources including the design cache and libraries:

- The *Design Cache* folder holds all the parts and symbols used in the project.
- The *Library* folder contains the libraries added to the project. All the parts in the library are available for placement on the schematic pages.

The shortcut menu for the *Design Resources* folder provides an additional command, [Remove PSpice Resources](#), to remove the simulation profile associated with the current project in case of a non-PSpice project.

Hierarchy

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* tab as part of a hierarchical tree. The hierarchical view of the design is derived from the files that exist in the *Design Resources* folder in the *File* view.

Project Manager Behavior

- Within the project manager window, you can expand or collapse the structure you are viewing by clicking the plus 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 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 a schematic page, a VHDL entity, or Verilog model, that object is opened in an associated editor. For example, double-clicking 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 tab within the project manager window. If you close the project manager window, the active project tab is closed.

One Design for One Project

Each project can contain only 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.

Session Log

The OrCAD X Capture Session Log window or output window contains a record of events that occur during the current session of the tool.

The Session Log window also includes results and messages from Capture utilities available from the *Tools* menu.



The following table lists the operations you can perform related to the Session Log window:

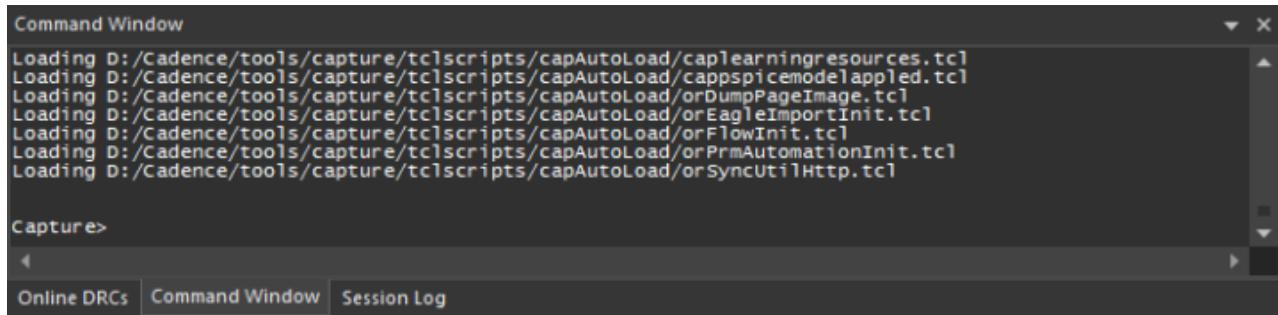
Operations related to Session Log	Description
Show	To show the Session Log window, choose <i>View – Session Log Window</i> . Alternatively, choose <i>Window – 1 Session Log - Session Log</i> .
Save	To save the contents of Session Log to any location on your system, click in the Session Log window and choose <i>File – Save As</i> .
Hide	To hide the Session Log window, choose <i>Hide</i> from the shortcut menu in the Session Log window.
Clear	<ul style="list-style-type: none"> Choose <i>Clear Session log</i> from the shortcut menu in the Session Log window. Or Choose <i>Edit – Clear Session Log</i> or press CTRL + Delete.
View as	To view the Session Log window as <i>Floating</i> or as a <i>Tabbed document</i> , choose the corresponding command from the shortcut menu in the Session Log window.

i **capture_crash_session.log**

If Capture exits with a crash, the information in Session Log is saved as `capture_crash_session.log` in the `%TEMP%` folder, if defined. Else, it is saved in the current working directory.

Command Window

OrCAD X Capture includes Tcl commands that are internally used when you perform tasks in the user interface. When you perform an operation (function) in Capture, the associated command is registered with the Tcl interpreter and the command is logged in Command Window.



To display Command Window, do one of the following:

- Select *View – Command Window*.
- Right-click any menu bar or toolbar and select *Command Window*.
The *Command Window* menu command acts as a toggle to hide or display Command Window.

To clear the contents of Command Window, at the Command prompt type *c/s* and press *Enter*.

Property Spreadsheet Editors

OrCAD X Capture includes three spreadsheet editors to view, edit, and add properties to the objects in a project:

- [The Property Editor Window](#)
- [Browse Spreadsheet Editor](#)
- [Package Properties Editor](#)

Related Topics

[Changing the Appearance of Property Editor](#)

The Property Editor Window

The Property Editor window lets you view, edit, and add properties to the objects in a project. This window displays all the library definitions, instance properties, and occurrence properties for an object.

Using Property Editor, you can edit properties for the [instances or occurrences](#) of:

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

The properties that appear in Property Editor depend on the object selected in the schematic page.

In the Property Editor window:

- Each column represents 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 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, the library definition of the property is visible for the instance.
- If a yellow cell contains hash marks, the corresponding property does not have an occurrence value, the instance value is visible for the occurrence.
- The properties that appear depend on the object selected in the schematic page.
- The properties depend on the tab selection at the bottom of the Property Editor window. For example, if the *Parts* tab is active, the properties for the selected parts are displayed.

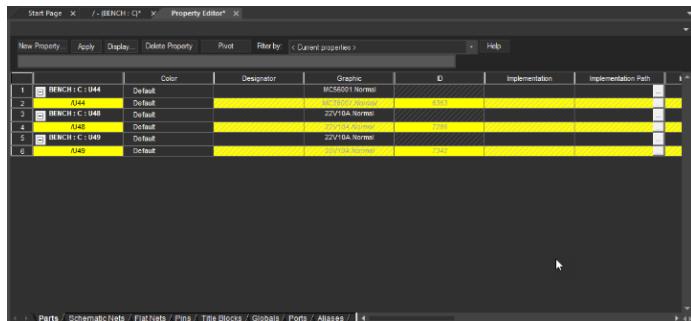
To open the Property Editor window, do the following:

1. Select one or more parts, nets, pins, title blocks, aliases, or globals in the schematic page editor.

2. Choose the [Edit – Properties](#) menu command.

Alternatively, you can use one of the following ways to open the Property Editor window:

- Right-click an object on the canvas and choose *Edit Properties* from the shortcut menu.
- Double-click an object on the canvas.



When editing properties in Property Editor:

- Property values that are applied to instances are visible on all the occurrences of those instances, unless a specific value is explicitly set for the occurrence, 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 be visible on the occurrences.
- Deleting a property value from an occurrence causes the instance property value to be visible on that occurrence.
- Library definitions are visible on the instance and occurrence of the object only if the instance or occurrence value is unedited.

i To browse and edit properties for an entire design, see [Browse spreadsheet editor](#).

The following table provides a description of the fields and buttons in the Property Editor window:

Field	Description
-------	-------------

OrCAD X Capture User Guide
OrCAD X Capture Interface--Command Window

New Property	<p>Displays the Add New Property dialog box to add a new property column or row. The new property is added to a single object if a single cell is selected or to multiple objects if multiple objects are selected on the schematic.</p> <ul style="list-style-type: none"> To add a property to multiple objects, shift-select the cells or select an entire row or column by clicking the property name. To add the property to an object, specify a property value for a given object. If no object is selected on the sheet, the value field is not activated in the <i>Add New Property</i> dialog box. <p>You cannot add a new property unless you manually specify values for the new fields in the property spreadsheet.</p>
Apply	<p>Applies the changes in Property Editor to the schematic page. The <i>Apply</i> button does not close the Property Editor window. You can also apply the changes to the schematic page by explicitly closing the Property Editor window.</p>
Display	<p>Displays the Display Properties dialog box to set the display option of the selected property and its value. Using the <i>Display Properties</i> dialog box, you can:</p> <ul style="list-style-type: none"> Define the display property format Display property font, size, and color Rotate display property at given angles Justify display property text as default, left, center, or right <p>You cannot display properties of an occurrence property from the <i>Display Properties</i> dialog box.</p>
Delete Property	<p>Deletes an editable property from the selected object or objects. Properties that are not editable appear in italics. If you select all the cells for a property and click the <i>Delete Property</i> button, the property is removed from the selected objects, but is retained in the filter. This is indicated by the hash marks that appear in the cell.</p>
Filter by	<p>Specifies a filter by which to view the objects. Use the Property Editor filters to constrain the available properties.</p> <p>For example:</p> <ul style="list-style-type: none"> The Capture filter displays common schematic capture properties available to most parts, The Capture-PCB Editor filter displays the properties needed to send a design to the layout editor. <p>You can view all the properties available on the objects in the Property Editor window by selecting the <Current properties> filter from the drop-down list.</p> <p>Another example of constraining properties is using the <i>Allegro_Signal_Flow_Routing</i> filter. This filter setting lets you view signal flow properties, such as <code>PROPAGATION_DELAY</code>, <code>RELATIVE_PROPAGATION_DELAY</code>, <code>RATSNEST_SCHEDULE</code>, and <code>DIFFERENTIAL_PAIR</code> in the <i>Flat Nets</i> tab.</p>
Column Value Editor	<p>Displays and lets you change the value in the 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.</p>
Tabs in Property Editor	<p>The tabs along the bottom of Property Editor let you edit properties specific to the objects in the selected tab. All the Property Editor tabs provide one row of property information per instance or occurrence.</p>
Parts	<p>Displays the part properties of the selected objects. The <i>Parts</i> tab includes hierarchical blocks. You can use the <i>Parts</i> tab in Property Editor to add and delete property instances and occurrences and to change their values.</p> <p>The <i>Graphic</i> 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.</p>
Schematic Nets	<p>Displays the schematic net properties of the selected objects. This tab includes constituent nets within buses.</p>
Pins	<p>Displays the pins of the selected objects. This tab includes hierarchical pins in hierarchical blocks.</p>
Title Blocks	<p>Displays the properties of the title block used on the schematic page. In the <i>Title Blocks</i> tab, you can add a property to the title block instance on a schematic page to display the full hierarchical path to the schematic.</p>

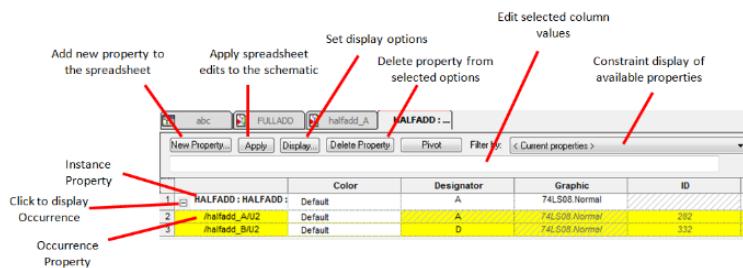
OrCAD X Capture User Guide

OrCAD X Capture Interface--Command Window

Globals	Displays the properties of the selected ground or power symbols for simultaneous editing of multiple symbols.
Ports	Displays source symbol, source library, and type of port properties. Facilitates the simultaneous editing of the 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	<p>In 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 or alongside their associated instance row depending on the orientation of the spreadsheet. 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 Property Editor cells show the property values for each instance or occurrence. A cell with hash marks indicates that the property does not exist on the object that the cell represents.</p> <p>You can add a value by clicking inside the cell, typing the value, and pressing ENTER or clicking the <i>Apply</i> button. A property value in italics is a read-only property and, therefore, cannot be edited.</p>

Moving Around in Property Editor

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

You can change the appearance of the Property Editor by showing or hiding columns, changing their widths, or even changing the orientation of the spreadsheet. Any changes you make to the appearance of the spreadsheet are saved to the **PREFPROP.TXT** file when you close the Property Editor window. The next time you open to that particular tab in that particular filter, you see the saved settings. This does not apply to any changes you make to the spreadsheet appearance while using the **<Current properties>** filter.

Changing the Orientation of the Spreadsheet

The default orientation of the spreadsheet is landscape as it shows properties in columns and instances and occurrences in rows. If you pivot the spreadsheet, the instances and occurrences appear in columns across the top, and properties appear in rows. This may be advantageous when the selected object or objects have several properties.

To change the orientation of the spreadsheet from the landscape (horizontal) to portrait (vertical), do the following:

1. Right-click the empty cell in the top-leftmost position of the spreadsheet.

2. Choose *Pivot* from the shortcut menu.

Alternatively, Double-click the empty cell in the top-leftmost position of the spreadsheet.

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

Showing or Hiding Occurrence Properties

- To display all the occurrence properties with a single keystroke, press and hold the `CTRL` key while clicking on one of the plus (+) symbols in the leftmost column.
- To hide the occurrence properties, press and hold the `CTRL` key while clicking on a dash (-) symbol in the left column.

Moving Columns

To move columns in the Property Editor window, do the following:

1. Click the title cell of the column to be moved.
2. Click the title cell of the column again and drag the column to the new location.

Sorting Columns

To sort columns in the Property Editor window, do one of the following:

- If the spreadsheet is not pivoted, double-click on the column heading to toggle the sort order between ascending and descending.
- If the spreadsheet is pivoted, that is if the orientation is portrait, right-click the column heading and choose *Sort Ascending* or *Sort Descending*.

Resizing Columns

To change the width of a column, do the following:

1. Move the cursor to the right edge of the title cell of the column to be resized.
2. When the cursor (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 the columns to the same width. When the spreadsheet is pivoted, these changes are saved on the tab and the filter in use.

Browse Spreadsheet Editor

You use the [Browse Spreadsheet editor](#) to create a new property, copy values across properties, or remove a user-defined property. You can display the *Browse Spreadsheet* editor from project manager, schematic page editor, the part editor, or the Find Results window.

Opening Browse Spreadsheet Editor

To open the *Browse Spreadsheet* editor, do the following:

1. Select the schematic design or the schematic page from the project manager.

⚠ You can also edit pin properties in the Browse Spreadsheet editor when opened from the [part editor](#) and part properties from the [Find Results](#) window.

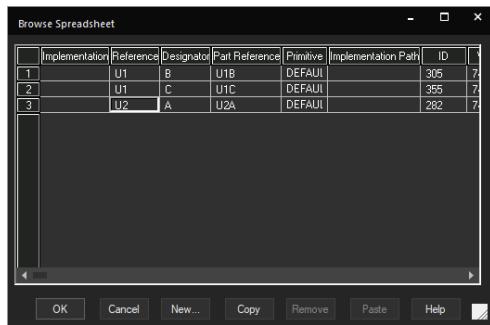
2. Choose *Edit – Browse*.
3. Select the required object type from the list.
4. In the *Browse Properties* dialog box, choose the preferred mode and click *OK*.

When opened from the project manager, you can change the properties of the following objects:

- Parts (including hierarchical blocks)
- Nets (including constituent nets within a bus) occurrences
- Flat Netlist
- Hierarchical ports
- Off-page connectors
- Title blocks
- Bookmarks
- DRC markers

The entire design is browsed for the object type you select and a list of the objects is displayed.

5. Select one or more objects from the resultant list.
6. Choose *Edit – Properties*.
The *Browse Spreadsheet* editor opens.
7. Add a new property or edit existing properties as required and click *OK*.

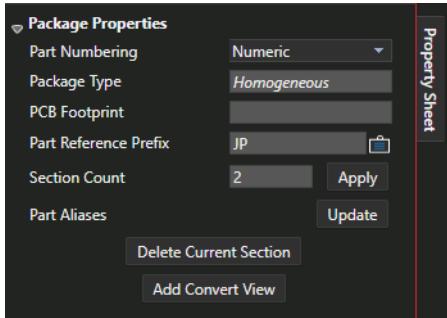


Related Topics

- [Defining Properties](#)
- [Editing Properties](#)
- [Browse Spreadsheet Editor](#)

Package Properties Editor

You can edit package properties using the Package Properties editor. The Package Properties editor is available along with the part editor when you open a part.



The spreadsheet displays all the package information on pins. The Package Properties editor gives you the options described in the following table:

Field	Description
Part Numbering	Specify the numbering mode. The valid values are <i>Numeric</i> and <i>Alphanumeric</i> . <i>Numeric</i> is the default value.
Package Type	Specify the type of project.
PCB Footprint	Specify the PCB footprint of the package.
Part Reference Prefix	Specify the part reference prefix. U is the default value.
Section Count	Enter the number of sections. The default value is 1.
Part Aliases	Specify any part aliases.
Display Property (icon)	Specify the visibility for the property name and value. You can select from <i>Do Not Display</i> , <i>Value only</i> , <i>Name and Value</i> , <i>Name Only</i> , <i>Both if Value Exists</i> , <i>Value if Value Exists</i> .
Apply	Click to apply the section count.
Update	Click to update part alias values.
Delete Current Section	Click to delete the selected section. This button is disabled for a part with a single section and for a heterogeneous part with only two sections.
Add Convert View	Click to add a convert view.

Component Explorer

Component Explorer is a component management system that provides an intuitive user interface to access components from various sources including Cadence-supplied libraries and content providers, without any additional overhead of creating a preferred part database and setting up an ODBC data source.

The Component Explorer interface provides a complete part authoring solution with the following functions:

- [Search](#) for components from a variety of sources and add them to schematic designs
- Create libraries by adding [components](#) from content providers
- [Create new categories](#) or templates
- [Create new components](#) in your local workspace
- [Link](#) a manufacturer part number with an existing library component
- [Create shared workspaces](#) and assign roles

OrCAD X Capture User Guide
OrCAD X Capture Interface--Component Explorer

- **Collaborate** with other team members on design projects by publishing components to shared workspaces

⚠ The workspace creation flow is only supported in the OrCAD X Professional (POX200 Pro) license. With this license you can collaborate with your team members by sharing workspaces and providing access rights on the shared workspaces. With the OrCAD X Standard (POX100 Standard) license, you can only access workspaces.

Supported Functions

In the Component Explorer interface, you can view, search, and place components from the following sources:

- Cadence-supplied PSpice libraries
 - Workspaces:
 - Local – Represented by the *My Workspace* node
 - Shared – A named workspace shared with other users so that they can view and use the shared components
 - Cadence-supported content search providers – SamacSys, Ultra Librarian, SnapEDA

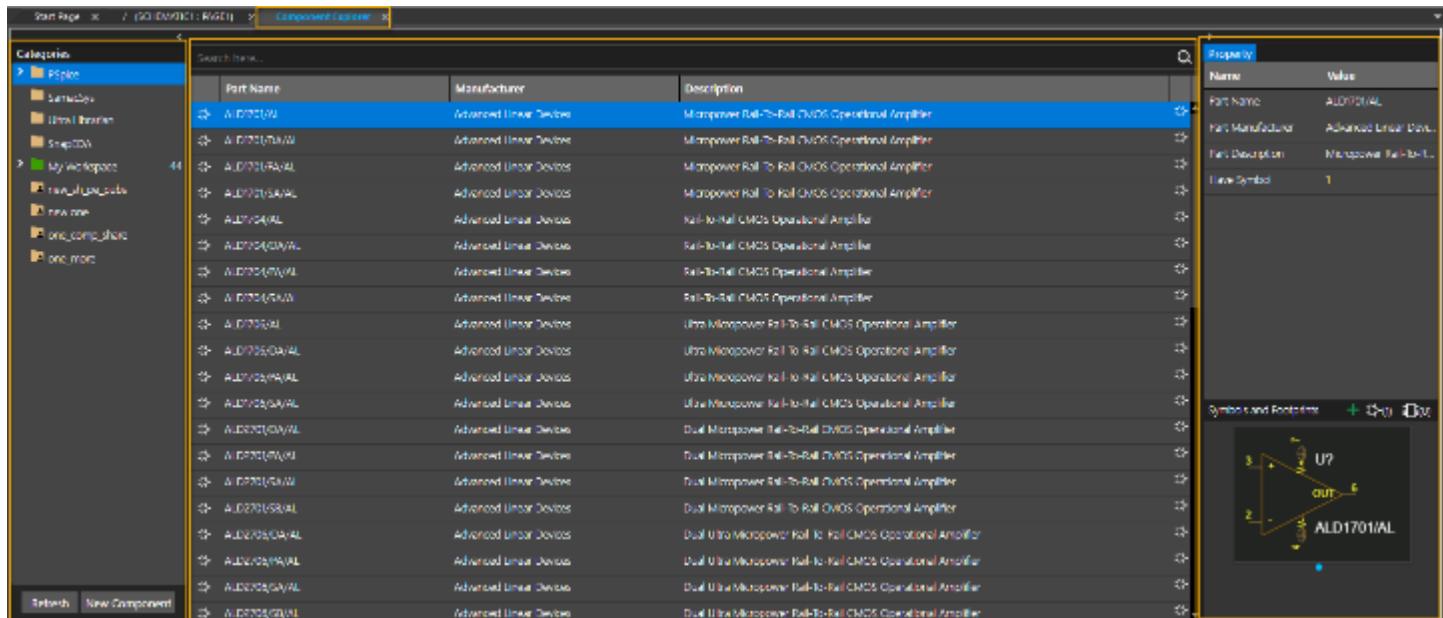
You can directly search for components matching a specific search string. When you are connected to any of the sources in Component Explorer, you can use the following functions:

- View components and their properties
 - View symbols and footprint
 - View Manufacturer Part Number (MPN) details for parts in the workspace
 - Place components in the design

Component Explorer Interface

- To open Component Explorer, choose *Place – Component* from the main menu of OrCAD X Capture.

The Component Explorer interface opens in a new tab.



The Component Explorer interface is divided into three distinguishable panes:

Pane	Description
------	-------------

<i>Left Pane</i>	The left pane or the Categories browser displays a list of sources from where you can place components in the design. The Categories tree includes the following nodes: <ul style="list-style-type: none">• <i>PSpice</i>: Includes thousands of components shipped with the OrCAD X installation.• <i>SamacSys, Ultra Librarian, and SnapEDA</i>: Cadence-supported content search providers with a database of components available from various component manufacturers.• <i>My Workspace</i>: Includes the new components you create from scratch, add from the search providers, and some default components. For any component, you can create and assign categories or sub-categories, attach default symbols and footprints, and add Manufacturer Part Number (MPN) details. You can share your local workspace components with others using shared workspaces.
Middle pane	The middle pane or the part browser displays the components from the node selected in the left pane. This pane also includes a <i>Search</i> box at the top where you can search for a component within a selected category or subcategory by specifying keywords associated with the component. You can also use search queries to perform a more specific search on the selected source in the left pane. Additionally, for the components in My Workspace, a <i>Manufacturer Part Number</i> table shows MPN details in the bottom of the middle pane, if they are linked to the component.
Right pane	The right pane includes the <i>Property</i> browser that displays the properties of the component selected in the part browser. The <i>Symbols and Footprints</i> section in this pane displays the symbol and footprint information of the selected component in a carousel view. If a part has multiple symbols, the symbol that appears in the Symbols and Footprints view at the time of selection, is placed.

Related Topics

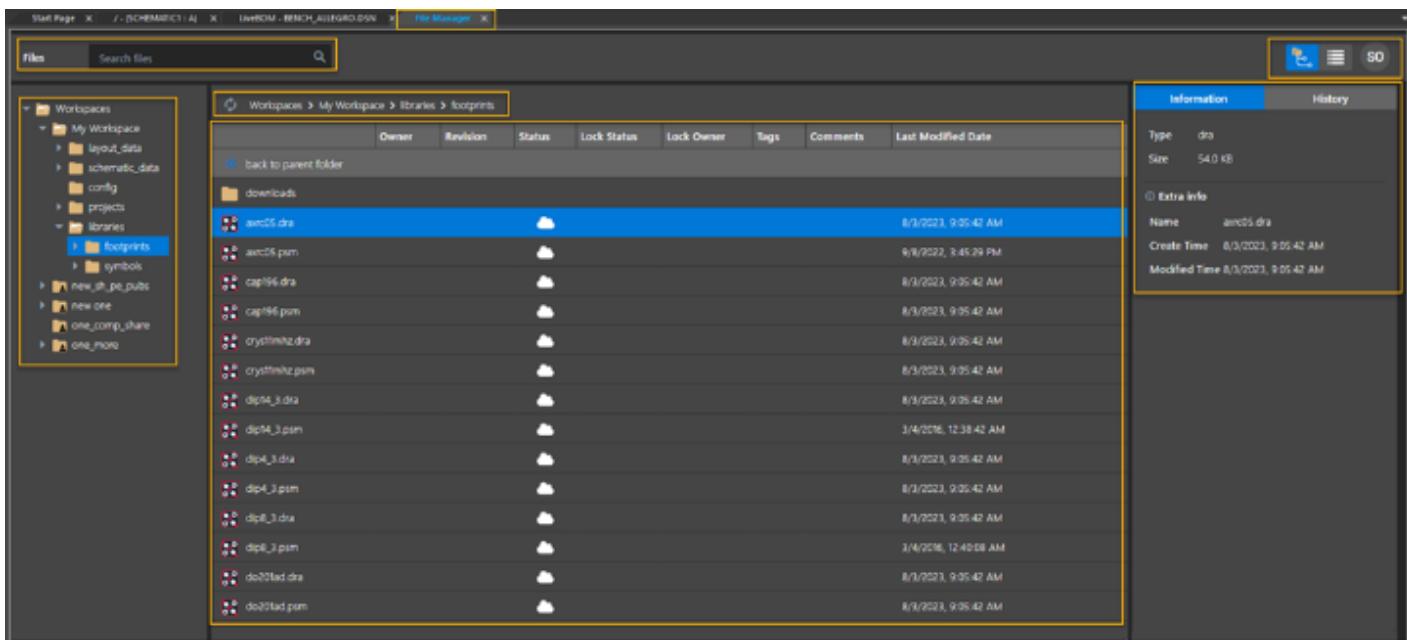
- [Managing Libraries in OrCAD X Cloud](#)
- [Creating Components](#)
- [Sharing Components](#)

File Manager

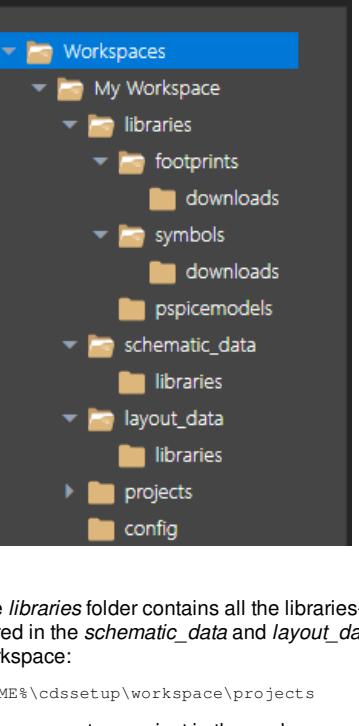
With an OrCAD X license, Capture design and library data in a workspace is read from the Cloud and the sync status is synchronized with the client. You can access workspaces using the File Manager user interface. File Manager displays the local workspace and all the workspaces shared by or with you. You can select each file and check its metadata including name, creation and modification dates, size, sync status, lock status, and so on. The complete folder structure of the projects and the designs in the workspaces are displayed just the way they appear for a project on a local disk in the project manager.

- Access File Manager from the *View – Workspace – File Manager* menu.

OrCAD X Capture User Guide
OrCAD X Capture Interface--File Manager



The File Manager user interface is divided into three panes:

Pane	Description
Left Pane	<p>Displays a tree view of <i>My Workspace</i> and all the shared workspaces. All the libraries, projects, and design files in the workspace are listed here.</p>  <p>The screenshot shows a dark-themed file explorer window. At the top, there's a blue header bar with a back arrow icon. Below it, the main area shows a tree structure under a folder named "Workspaces". The "My Workspace" folder is expanded, revealing sub-folders: "libraries", "schematic_data", "layout_data", "projects", and "config". Each of these sub-folders has its own "downloads" and "pspicemodels" sub-folders. The "libraries" folder under "schematic_data" and "layout_data" also contains a single "libraries" folder.</p> <p>The <i>libraries</i> folder contains all the libraries-related data, such as PCB footprints and schematic symbols. All the schematic and layout files are stored in the <i>schematic_data</i> and <i>layout_data</i> folders, respectively. The <i>projects</i> folder contains all the projects created or stored in the Cloud workspace:</p> <pre>%HOMEPATH%\cdssetup\workspace\projects</pre> <p>You can create a project in the workspace or add an existing project to the workspace.</p> <p>A user with the Librarian role has complete access to the contents of the <i>libraries</i> folder. Similarly, a user with the Designer role has complete access to the <i>projects</i> folder containing all the design data.</p>

Middle Pane	Displays the files and folders contained in the node selected in the left pane. The metadata of the selected files is also displayed. The details listed depend on the selection in the left pane: <ul style="list-style-type: none"> • Owner: Owner of the shared workspace. • Revision: Major or minor revision (version) of the file in use. • Status: The sync status of a file. The values can be <i>In Sync</i>, <i>Pending</i>, <i>Sync error</i>, and <i>Sync in progress</i>. • Lock Status: Indicates whether the file is locked by a user. • Lock Owner: The id of the user who has locked the file. • Tags: A user-defined value that can be used for filtering data. • Comments: Any comment or note. • Last Modified Date: The date of the most-recent file modification.
Right Pane	Displays file information and history of revisions of a file. You can view the revision history of a file, and roll back to a specific revision.

Additionally, File Manager includes the following screen options:

UI Element	Description
View modes	Provides the option to view the files in the middle pane in either the <i>Hierarchical View</i> or <i>Flat View</i> mode.
Username	Displays the initial two letters of the username. Moving the mouse pointer over the initials reveals the username (email id).
Files Search bar	Searches for a specified search string in the node selected in the left pane.
Refresh button	Reloads the selected folder in the left pane.
Breadcrumb Navigation	Indicates the current location of the selected file. Also helps you navigate to the parent folder.

Related Topics

- [Managing Workspace Files](#)
- [Managing Libraries on OrCAD X Cloud](#)
- [Sharing Workspaces](#)

User Interface Customizations

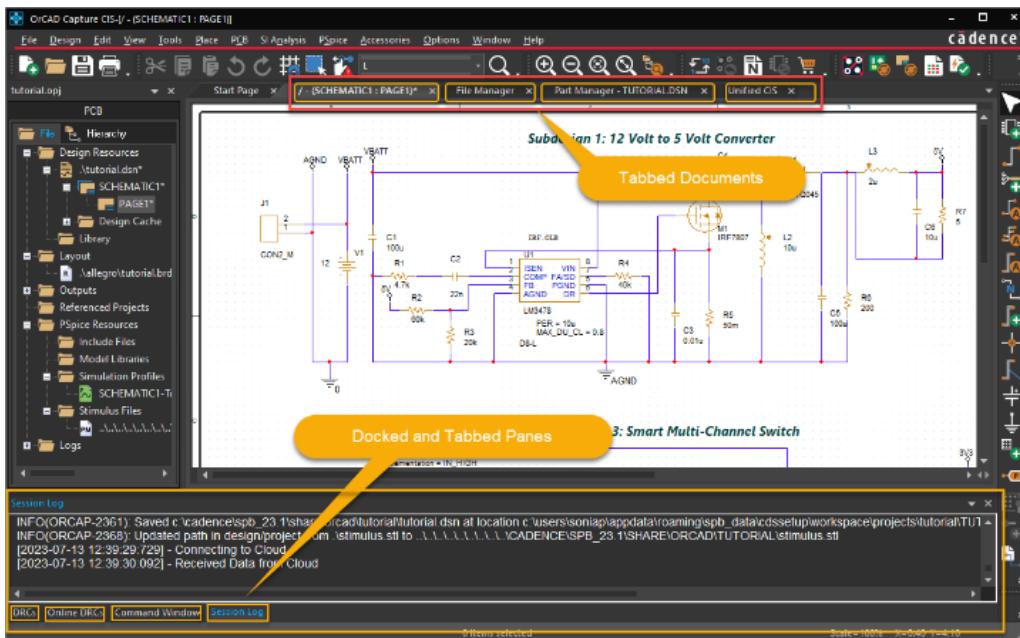
In OrCAD X Capture, schematic pages, parts, projects, instances of part manager, open in separate panes. You can open as many panes as supported by your computer resources. For example, if you wish to work with three schematic pages or three parts, you can open each in its own pane. All resources opened from a project are displayed as horizontal tabbed documents in the canvas area. By default, all panes displaying any kind of output are at the bottom of the application. If multiple panes are open in the output window, they are displayed as docked and tabbed panes.

UI Panes

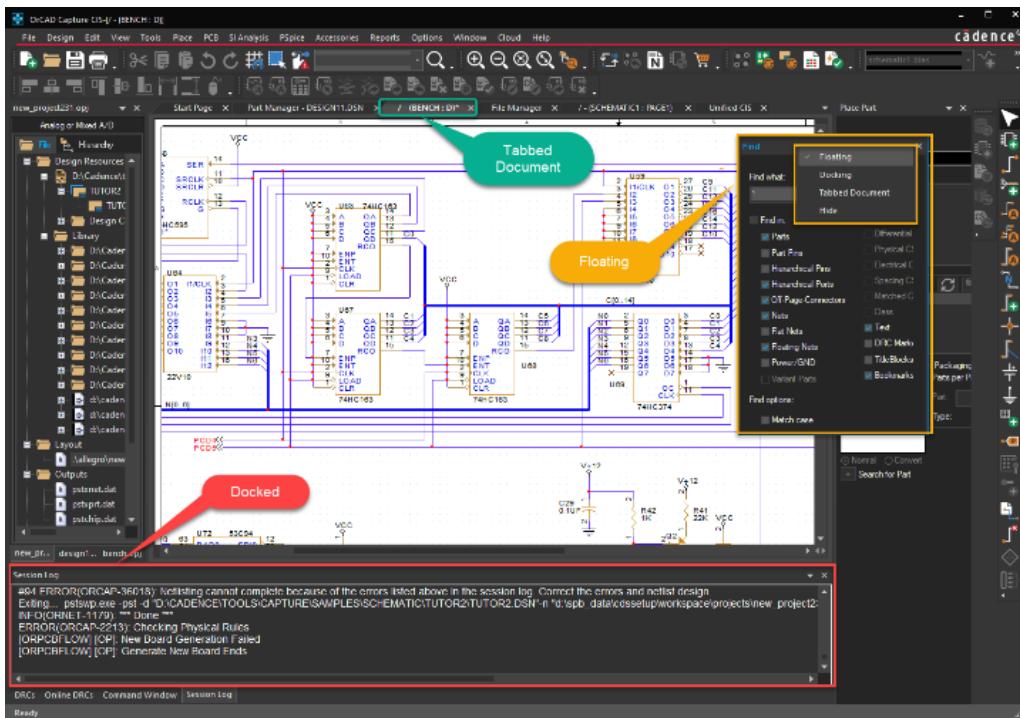
UI panes in the canvas area are displayed as horizontal tabbed documents or panes. These panes, which have labels, are not reduced in size or resolution when the number of open panes increases. They can also be moved and arranged as desired.

OrCAD X Capture User Guide

OrCAD X Capture Interface--User Interface Customizations



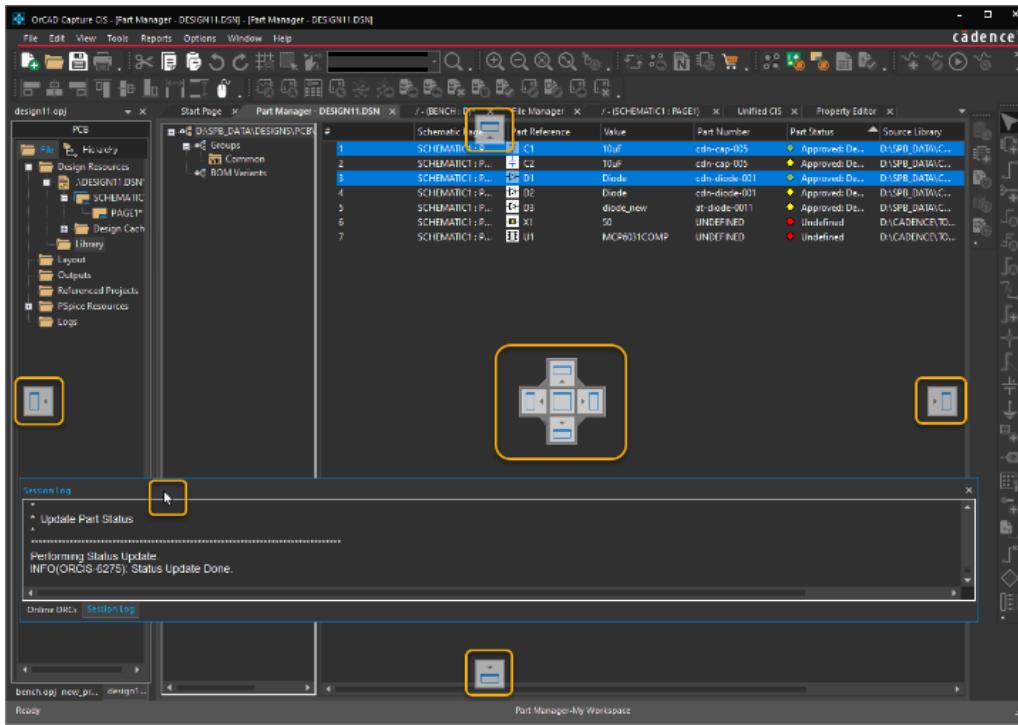
- Use the right-click shortcut menu commands for each pane to display the planes in the desired way, *Floating, Docking, or Tabbed Document*.



- Drag and drop panes to move them around using the docking markers as shown in the following figure:

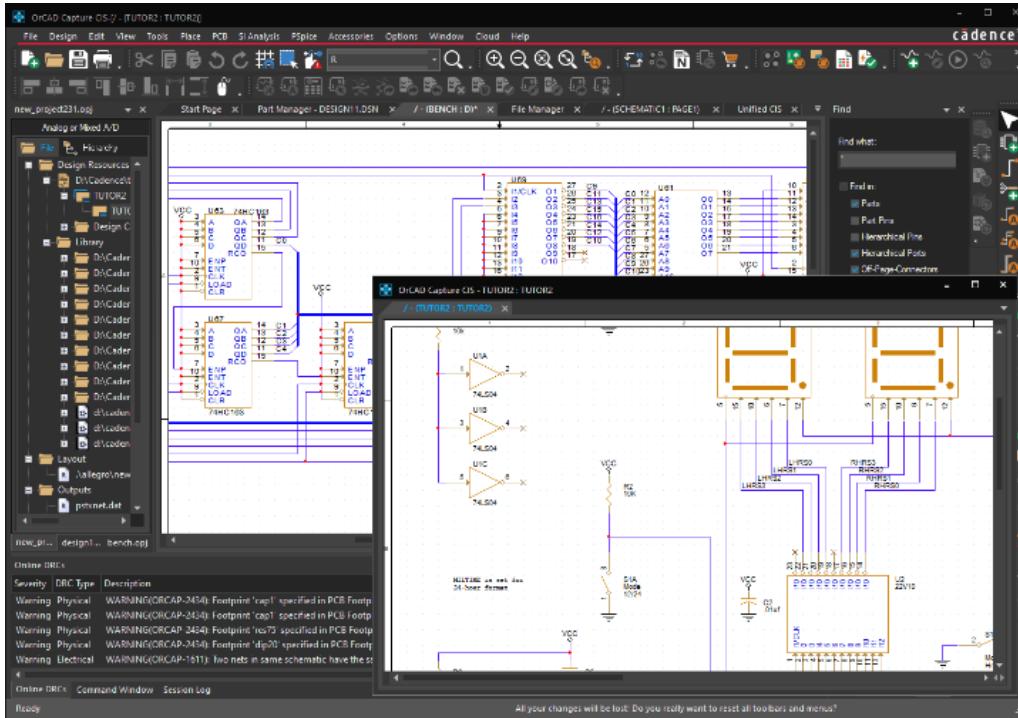
OrCAD X Capture User Guide

OrCAD X Capture Interface--User Interface Customizations



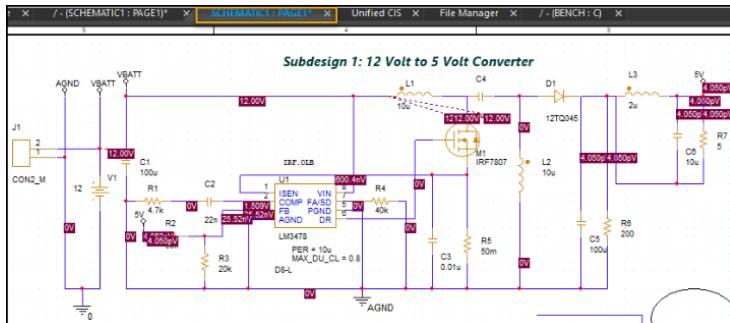
Multiple Monitor Support

You can drag a tabbed document and display it on another monitor as shown in the following figure. In this example, the active project is dragged and becomes a separate window, which can be displayed on a different monitor:

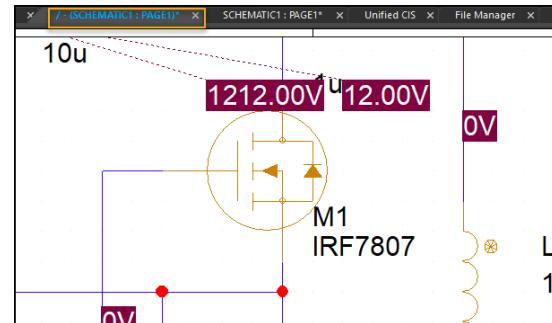


Schematic Pages

You can open a new instance of the currently active schematic page window. You can then scroll the two windows to different positions in the schematic and view the data.



Schematic: Page1 - Window 1



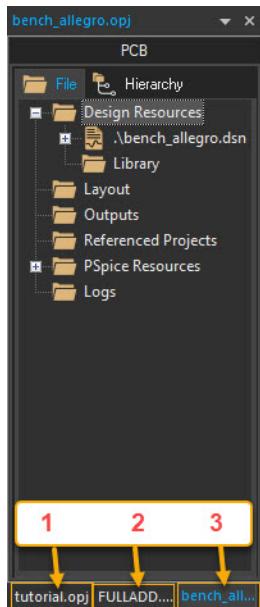
Schematic: Page1 - Window 2

Both the windows provide a simultaneous view of the schematic. Any change you make to the schematic in one window is reflected in the other window as well.

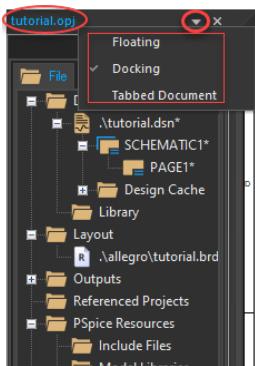
- To open a new window on the active schematic page, choose *Window – New Window*.
- To open a recently opened schematic page, from the *Window* menu choose the desired schematic page.
- To close all the open schematic pages, choose one of the following commands from the *Window* menu:
 - *Close all Tabs of Active Project*
 - *Close all Tabs of Active Project Except Current*
- To close all the windows, choose *Window – Close All Windows*.

Project Manager

When you work simultaneously on several projects, each opens in its own project manager window.



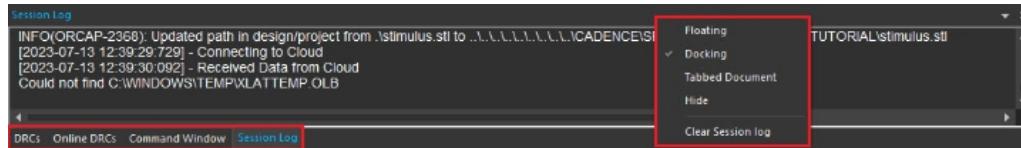
You can right-click the title bar of project manager and set it as *Docked*, *Floating*, or *Tabbed*.



Output Display

The common area at the bottom of the application area where you can view any kind of output: commands, results, messages, errors, or warnings. This output is displayed in the following panes:

- Command Window
- Session Log
- Online DRCs
- DRCs



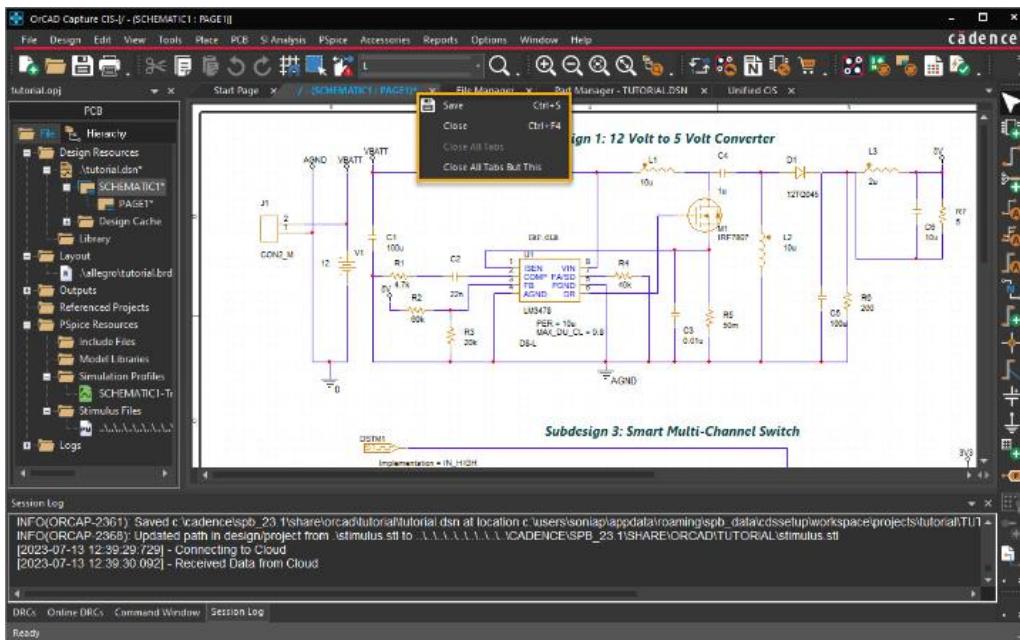
You can right-click the title bar or the tab name of an output pane and choose the option to set it as floating, docked, or tabbed (document) panes.

Navigating in Tabbed Panes

All the open documents are tabbed panes. You can right-click the tabs to save or close the panes.

OrCAD X Capture User Guide

OrCAD X Capture Interface--Capture Configuration



- To navigate between the open panes, click the tab for the pane or press **CTRL+TAB**.
- To close the current pane, right-click the tab and choose *Close*.
- To close all the open panes, right-click any open tab and choose *Close All Tabs*. This closes all the panes open for the project.

i The *Close All Tabs* option is not available for the project manager tab.

- To close all the open panes except the currently selected one, right-click the current tab and choose *Close All Tabs But This*. This closes all the panes for the project except the current tab.

Capture Configuration

In this section:

- [User Interface Configuration Settings](#)
- [Capture.ini File](#)
- [Customizing Menus and Toolbars](#)
- [Capture.ini Variables](#)

User Interface Configuration Settings

Capture provides configuration settings to customize the user interface for an enhanced user experience. Various commands and dialog boxes available from the *Options* menu help you configure the user interface in the following ways:

- Customize the working environment specific to your system by setting preferences in the *Preferences* dialog box.
- Create default settings for new designs with the *Design Template* command. These settings are honored for 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, or individual schematic pages with the *Schematic Page Properties* command.
- Create default settings for new parts by setting part properties in the *Part Properties* dialog box.

- Override default properties on individual parts by setting package properties in the [Package Properties](#) dialog box.

Regardless of which pane you are in Capture, the [Preferences](#) and [Design Template](#) commands are always active in the [Options](#) menu. In addition, the [Options](#) menu contains commands specific to the currently active window. For example, when the project manager is the active window, the [Options](#) menu displays the [Design Properties](#) command. Similarly, the [Options](#) menu displays the [Schematic Page Properties](#) command when the schematic page editor is currently active.

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 receive a design from another user, the preferences settings are not inherited. This implies that the colors, grid display, pan and zoom, and any other customizations you do on your system are honored even for the designs created on some other system with different settings.

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 is a good idea to check the design template settings before you create a new design.

After you begin working on a design, you can customize its 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](#) and the [Package Properties](#) commands on the [Options](#) menu for part manager to set default part properties and customize the settings for individual parts.

Capture.ini File

At the startup, Capture uses a pre-defined set of default values for the application settings. These default values are defined in the Capture configuration file `Capture.ini`.

If you run Capture for the first time on a computer, it uses a pre-defined set of configurations to create the `INI` file. Subsequently, each 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 an `INI` file exists in the 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 a different location, you need to specify the location as a command line argument to the `Capture.exe`.

Capture.ini Variables

The `Capture.ini` file contains a large set of initialization variables used by Capture. As 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 recommended in [Capture.ini Variables](#).

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` file, the CIS configuration settings will also be deleted.

However, Capture creates a backup of the CIS settings in a separate file: `BackupCaptureCIS.ini` at the same location as the `Capture.ini` file. This backup is maintained each time you close Capture. So the `BackupCaptureCIS.ini` file contains the most recent CIS configuration settings.

If you delete the `Capture.ini` file and restart Capture, you are prompted with a message to choose if you want to use the backed up CIS configuration settings. If you select yes, the CIS settings are copied from the `BackupCaptureCIS.ini` file to the newly created `Capture.ini`.

Customizing Menus and Toolbars

You can customize the menus and toolbars in OrCAD X Capture to run any TCL methods from the menus. You can also specify your own icons for the menus or toolbars items.

 Right-click shortcut menus cannot be customized.

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

`<Cadence_installation>\share\orResources`

To add a new menu item, you need to specify the following syntax:

```
<menuItem name="<menu name tag>">  
<type>popup/action</type>
```

```
<label>Menu Label</label>
<enabled>true/false</enabled>

<children>
  <menuItem name="
```

You can also add menus dynamically using TCL commands.

Capture.ini Variables

The `Capture.ini` file that contains default application settings is placed in the `HOME` directory, `%HOME%\cdssetup\OrCAD_Capture\<release_version\>`, in your system. However, you can specify another location for this file by using the `-i` switch when invoking Capture. The `Capture.ini` file contains a large set of initialization variables used by Capture. This section defines a set of these variables that alter the behavior of Capture at startup.

ⓘ Important

- The `Capture.ini` file is used by Capture at the startup. Any incorrect or corrupt information in this file can cause unexpected behavior in Capture. Before making any changes to this file, you are advised to first make a backup of your installed `Capture.ini`. This topic covers only a selected list of `ini` sections and only selected variables within these sections. It is advised that you work with only these variables.
- The `Capture.ini` file is a text-based file and it can be edited in any Text editor.

This topic covers initialization variables in the following sections of the `Capture.ini` file:

- [SearchToolBarSetting](#)
- [Search Toolbar Options](#)
- [Docking](#)
 - `session_docked`
 - `session_show`
- [Print Settings](#)
 - `InstanceMode`
- [Preferences](#)
 - `PSpiceSymLibPath`
 - `EnableITC`
- [Footprint Viewer Type](#)
 - `type`
- [Allegro Footprints](#)

- Example:
- Part Management
 - Configuration File
- Frame View Options in CIS Explorer Window
 - Visible
- ConvertDialog
 - Choice
 - Do Not Ask

SearchToolBarSetting

This ini section includes the variables that control the selection of the Search toolbar options. You can set a Search toolbar option as selected or un-selected by setting the variable values in this section.

- Use 1 to set an option as selected.
- Use 0 (zero) to set an option as un-selected.

Search Toolbar Options

- Parts
- OffPage
- BookMark
- CommentText
- Nets
- FlatNets
- HierPorts
- DRC
- TitleBlock
- PowerSymbol
- HierPins
- PartsPins
- VariantParts
- MatchCase
- Highlight

Docking

The variables in this section enable you to set the docked window status and visibility of Capture CIS windows.

session_docked

- Set the value to 1 to dock the Session Log window.
- Set the value to 0 (zero) to specify Session Log as an MDI window.

session_show

Set the value to 1 to ensure the Session Log window is visible at startup.

Set the value to 0 (zero) to ensure the Session Log window is hidden at startup. To display the Session Log if it is hidden, choose the *Session Log* option in the Capture CIS *Window* menu.

Print Settings

Use the variables in this section to define the print settings in Capture CIS.

InstanceMode

Set the value to `1` ensure that the default selected Print option in the *Print* dialog box is set to *Inst. Mode* (Instance Mode).

Set the value to `0` (zero) to ensure that the default selected Print option in the *Print* dialog box is set to *Occ. Mode* (Occurrence Mode).

Preferences

The Preferences section in the ini contains variables that provide access to the settings in the Capture CIS Preference dialog box.

PSpiceSymLibPath

Specify the path to libraries for PSpice simulation.

EnableITC

- Set the value to `TRUE` to Enable Intertool Communication in Capture CIS.
- Set the value to `FALSE` to disable Intertool Communication in Capture CIS.

Footprint Viewer Type

The variables in this section describe the type of footprints that will be displayed in the CIS Explorer footprint window. The section is also linked to the *Allegro Footprints* section that is then used to define the directory location for the footprint (.psm) files.

type

This variable defines the type of footprints to display.

- Set the value to *Allegro* to use Allegro files for footprints.
- Set the value to *Layout* to use Layout files for footprints.

Use the `dir0` variable in the Layout Footprints section to define the directory location containing the footprint files to use if the footprint type is defined as *Layout*. Or use the `dir0` variable in the Allegro Footprints section to define the directory location containing the footprint files to use if the footprint type is defined as *Allegro*.

Allegro Footprints

The variable (or variables) in this section define the directory location (or locations) for the psm and pad files used by the PCB Editor 3D Footprint Viewer to display a footprint. A `.psm` file is used by the 3D Footprint Viewer to display the 3D footprint. A pad file is used by the Viewer to display the pin information in the viewer.

You can define any number of paths that contain the `.psm` and `.pad` files. If you define multiple paths, Capture CIS searches the directories, for the footprint files, in the order in which they are defined in this section.

If the footprint information (corresponding `.psm` file) is not found in any of the paths defined by the variables here, Capture CIS displays an error message in the Session Log window.

Example:

```
[Allegro Footprints]
Dir1=C:\Cadence\SPB_23.1\share\pcb\pcb_lib\symbols
```

Part Management

The variables in this section define paths to the database configuration file used in the Capture CIS flow. The configuration file identifies the ODBC data source to use, Part properties that are transferred to your design and the relationship across the tables in your preferred parts database.

If this section is missing or the configuration file paths are incorrect, you will experience errors in any CIS database operations.

Configuration File

Use this variable to define the path for the database configuration file if you are using the Licensed version of Capture CIS.

Example:

```
Configuration File=C:\Cadence\SPB_23.1\tools\capture\samples\BENCHACC.DBC
```

Frame View Options in CIS Explorer Window

The following sections let you toggle the visibility of the corresponding frames in the Capture CIS Explorer window.

- Visibility Frame
This section defines the variables of the visibility frame in CIS Explorer.
- Schematic Part Frame
This section defines the variables of the schematic part frame in CIS Explorer.
- FootPrint Frame
This section defines the variables of the footprint frame in CIS Explorer.
- Relational Table Frame
This section defines the variables of the relational table frame in CIS Explorer.

Each of the above sections has two variables that define the frame visibility in the Capture CIS Explorer window.

Visible

- Set the value to `1` to ensure the corresponding frame is visible in the *Local Part Database* tab.
- Set the value to `0` (zero) to ensure the corresponding frame is hidden in the *Local Part Database* tab.

ConvertDialog

The variables in this section define the default behavior of the *Save Converted Libraries and Designs* dialog box. This dialog box appears when you exit Capture and you have included a configured library or a referenced design from a version of Capture prior to 16.3 in design that is based on a 16.3 or later version of Capture.

Choice

- Set this variable to `Save All` to ensure that all the displayed libraries or designs are saved to the upgraded version.
- Set this variable to `No All` to ensure that none of the displayed libraries or designs are saved to the upgraded version.

Do Not Ask

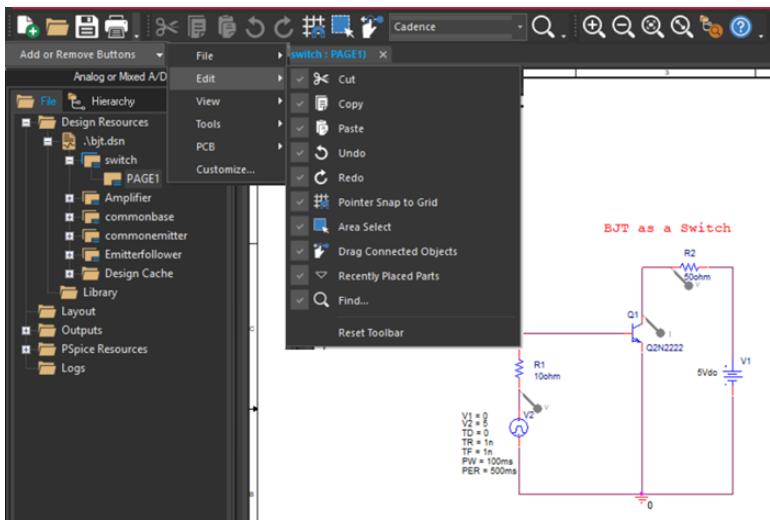
Set this variable to `1` to ensure that the dialog box is not displayed each time you close Capture in the upgrade library and design mode. Capture uses the option that is set as the *Choice* in this section.

This option applies to referred libraries only. These are libraries that are included in a project through the *Add Library* command in the *Place Part* dialog box. In addition, when you close an older version library directly opened in Capture or when you exit Capture after opening an older version library, the application always prompts you to upgrade the library.

Set this variable to `0` to ensure that the dialog box appears each time you close Capture in the upgrade library and design mode.

OrCAD X Capture Toolbars

Toolbars are organized according to function, and the icons in these toolbars are arranged based on their menus. You can toggle individual icons on or off in the toolbar.



The OrCAD X Capture user interface includes the following toolbars that provide shortcuts to the commonly used commands.

Capture Toolbar

The Capture toolbar categorizes commonly used commands under the following toolbars:

- File



- Edit



- View



- Tools



PSpice Toolbar

The PSpice toolbar provides shortcuts to common PSpice commands. This toolbar appears only if you have a PSpice license and open a project that uses PSpice.



Draw Electrical Toolbar

The Draw toolbar provides shortcuts to commands for placing electrical components, pins, wires, and buses.



Draw Graphical Toolbar

The Draw toolbar provides shortcuts to commands for placing drawing objects, such as arcs, polylines, ellipses, and text.



Align Toolbar

The Align toolbar offers a quick and easy way to align selected objects vertically (top, middle, and bottom) or horizontally (left, center, and right) with reference to other objects. It also provides an option to equally distribute selected objects horizontally or vertically within a virtual selection bounding box. You can also align selected objects with reference to a mouse click using Mouse Mode or equally distribute the selected objects within the area defined by mouse clicks using Mouse Mode.



CIS Explorer Toolbar

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



SI Analysis Toolbar

The SI Analysis toolbar provides shortcuts to the commands to set up a design for use with SigXplorer.



PCB Toolbar

The PCB toolbar provides shortcuts to open the following:

- Layout file
- *Update Layout* dialog box
- *Updated Schematic* dialog box
- Constraint Manager
- Design Rules Check dialog box



Related Topics

[Managing Toolbars](#)

Working with Projects

A project in OrCAD X Capture refers to the collection of design file, part libraries, report files, and other associated content within the Capture environment.

This section consists of the following topics:

- [Creating a Project](#)
- [Setting Project Preferences](#)
- [Setting up the Design Template](#)
- [Moving Objects by Drag and Drop](#)
- [Capture Directory Map](#)
- [Opening a Project](#)
- [Saving a project, design, or library](#)
- [Closing a Project](#)

Creating a Project

A project file (`.OPJ`) is a container for the design file (`.DSN`). There is 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 dialog boxes accessible from the *Tools* menu.

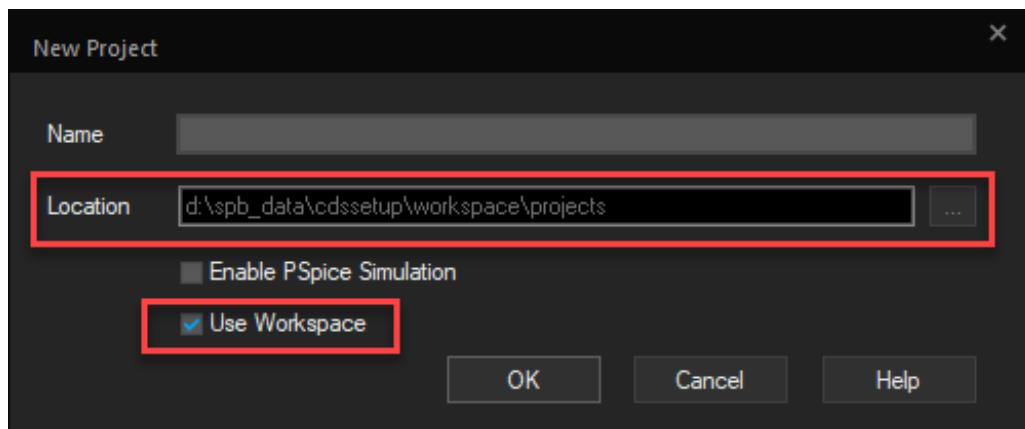
When the project is first created, the project manager creates a design file with the same name as the project. It also creates a schematic folder within the design file, and a schematic page within the folder. You can create a new design to replace the design created by the project manager.

Creating a New Project

To create a project, do the following:

1. Choose *File – New*.

The *New Project* dialog box appears.



2. In the *Name* text box, specify a name for your new project.

(i) 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. In the *Location* field, specify the path where you want the new project files to be saved, or use the *Browse* button to locate the directory.

With the *Use Workspace* check box selected, the *Location* box shows the pre-seeded workspace path, which cannot be edited. This option is selected by default and it is used to save only the [workspace-based](#) projects.

! You cannot manually select the workspace path to store a project.

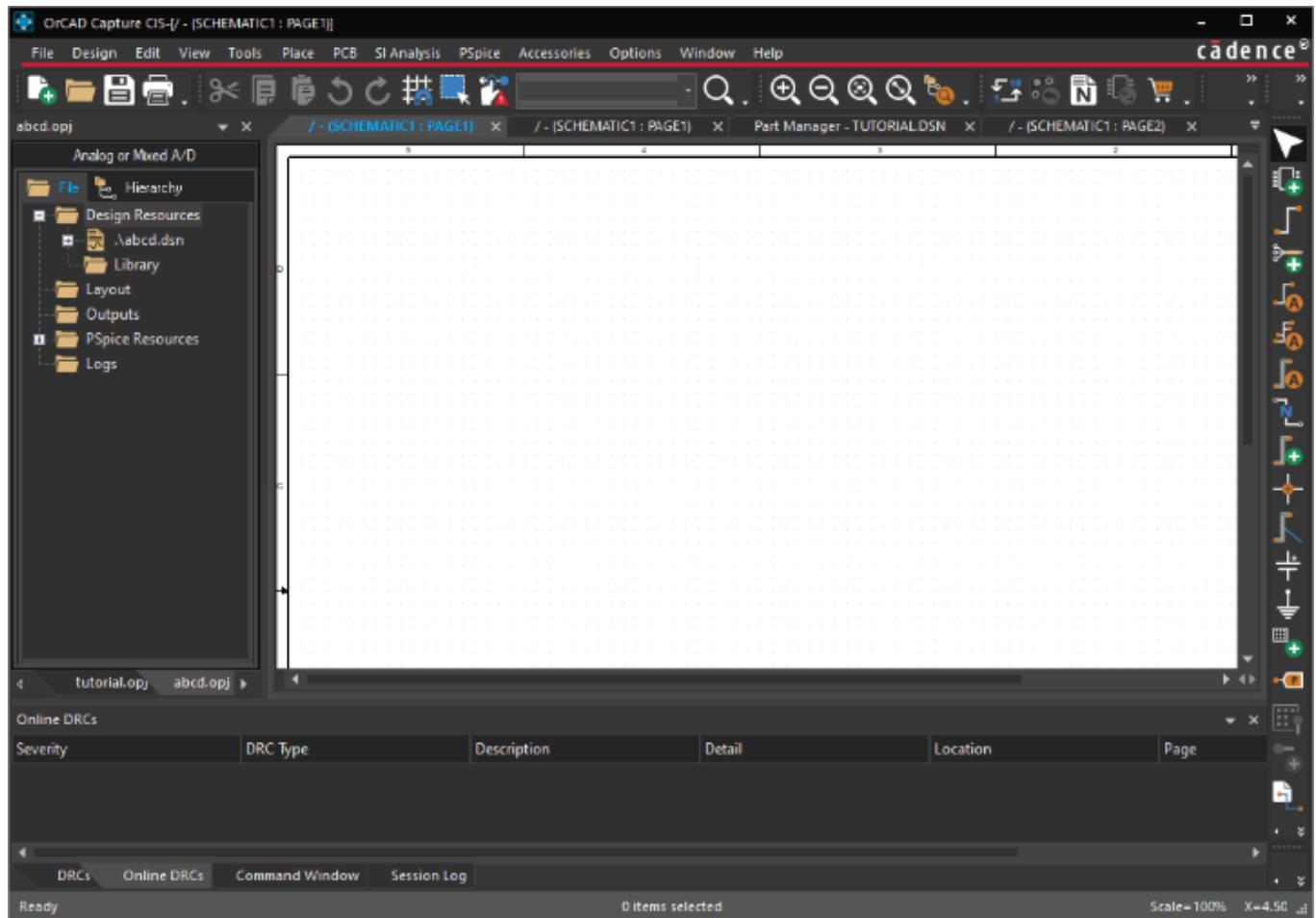
4. Choose another location to create a non workspace-based project,

! To move an existing local project to the workspace, use the *File – Add to Workspace* menu command.

5. Select the *Enable PSpice Simulation* check box if you intend to include simulation capabilities in your PCB design.

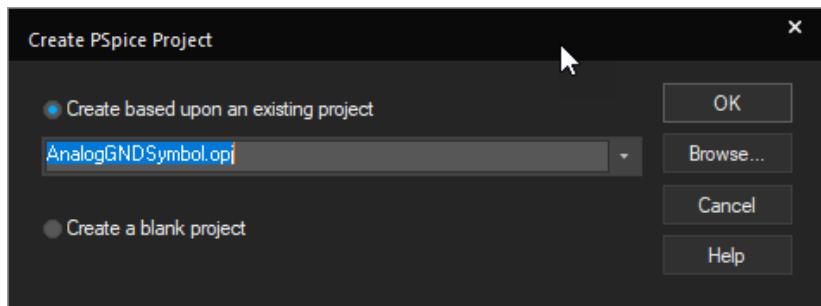
6. Click *OK*.

A new project is created and a blank schematic page is displayed.



Setting up New Project for Simulation

If you select the *Enable PSpice Simulation* check box, in the *New Project* dialog box, the *Create PSpice Project* dialog box appears.



1. Select one of the two options:

- *Create based upon an existing project*

If you select this option, choose a project file (*.OPJ*) from the drop-down list or the *Browse* button. PSpice provides a set of design templates covering basic electronics circuits and SMPS topologies. These design templates cover a 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 at: <installation_directory>\tools\capture\templates\pspice

- *Create a blank project*

By selecting this option, you create a new project that can be simulated in PSpice AD.

2. Click *OK* to create the new project directory and open the schematic page editor.

Setting up an Existing Project for Simulation

A design created without selecting the *Enable PSpice Simulation* option in OrCAD X Capture can still be simulated by PSpice. The basic process involves setting up a simulation profile and preparing the parts for simulation as described in [Creating Design for Simulation](#).

Related Topics

[Managing Workspace Files](#)

Setting Project Preferences

You set project preferences in the *Preferences* dialog box. The settings in the *Preferences* dialog box determine how Capture works on your system. These settings 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, the preference settings are not inherited. You can set themes, colors, grid display options, pan and zoom options, and so on to suit your requirements. The settings are retained even if you work on a project that was created on another system.

After you begin working on a project, you can customize its specific characteristics using the following commands from the main menu:

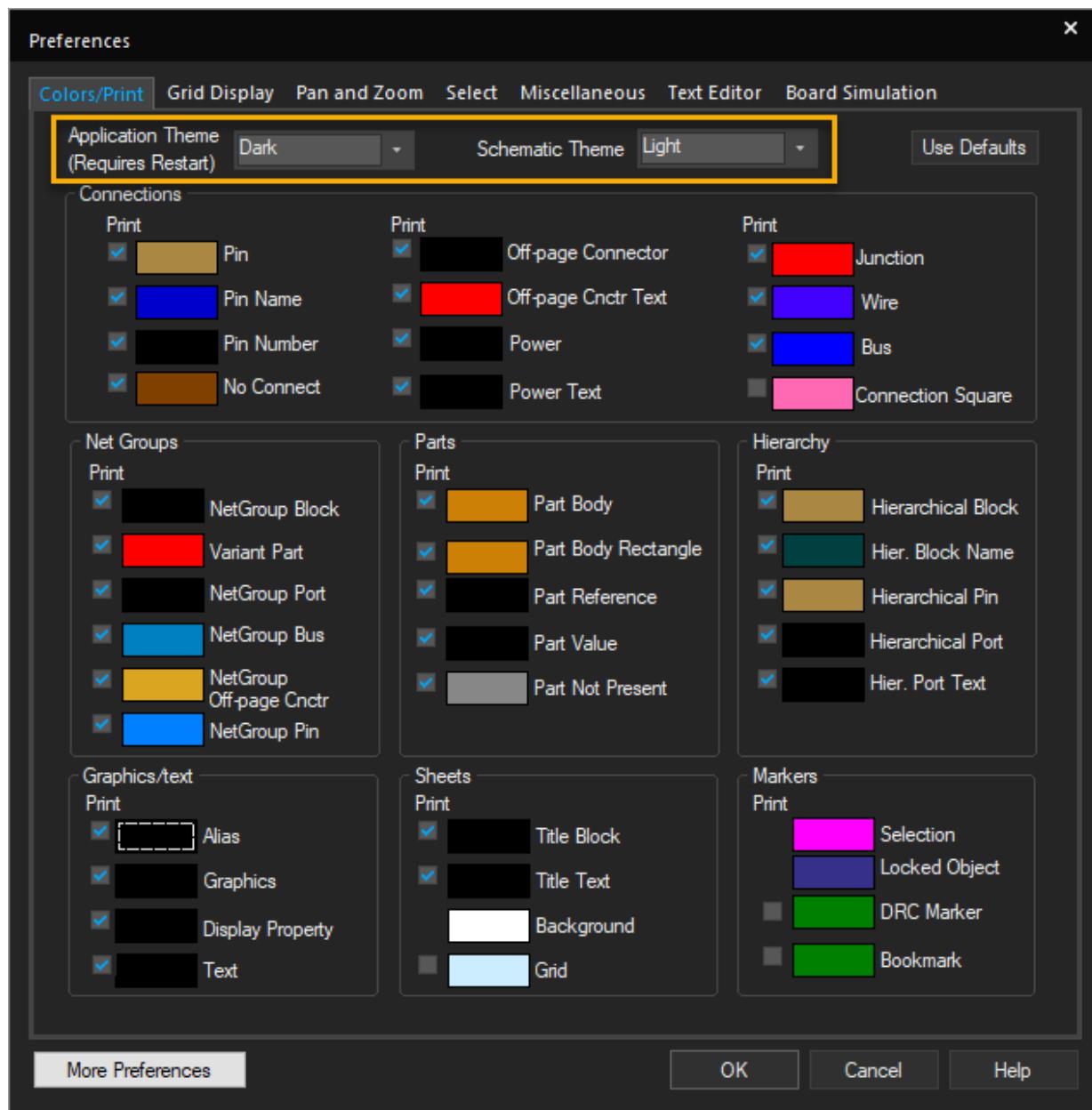
- *Options – Design Properties* when you are in the project manager
- *Options – Schematic Page Properties* when you are in the schematic page editor

Setting Theme for Application and Schematic

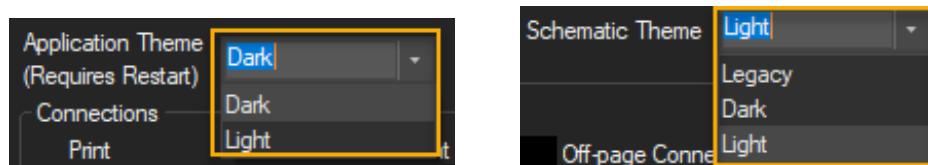
OrCAD X Capture opens in a dark theme by default. You can set the theme from the *Preferences* dialog box for both the application and the canvas (schematic page).

To change the application theme, do the following:

1. Choose *Options – Preferences*.
The *Preferences* dialog box is displayed.



- From the *Colors/Print* tab, select the theme for the application and the schematic from the respective drop-down lists.



- Click *OK*.

 The standard Windows operating system dialog boxes do not follow the application theme when launched from Capture.

Setting Colors for Objects on the Schematic Page

You can 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 from the *Colors/Print* tab in the *Preferences* dialog box.

1. Choose *Options – Preferences*.
2. Select the *Colors/Print* tab.
3. Click the color for an item.
The color palette window opens.
4. Select a new color.
5. Click *OK*.

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

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

Controlling the Grid Display

You can control the grid display by selecting either dots or lines for the grid and specifying whether to display or print the grid. You can also choose to have the pointer snap to grid as you place objects. These options can be set 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 is an integer with a value of 1, 2, 5, or 10

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.

You can choose between displaying a grid independently in the schematic page editor or the part editor, and displaying the grid with dots or lines. You can also specify whether the pointer snaps to grid in each editor. Additionally, the drawing objects, like Line, Polyline, Text, Rectangle, Ellipse, Arc, and Picture can also be placed on fine grid.

For the schematic page editor and the part editor, you can specify:

- Whether to display the grid.
- Whether the grid uses dots or lines.
- The grid spacing—the space between each point on the grid.
- Whether the pointer snaps to grid as you place objects.

To control the grid display, do the following:

1. Choose *Options– Preferences*.
2. In the *Preferences* dialog box, select the *Grid Display* tab.
3. Select the grid style and spacing, and click the *Displayed* or *Printed* option to change the visibility.
or
Choose *View – Grid* to set grid visibility only.
The visibility of the grid can be toggled.
4. To change snap to grid, in the *Grid Display* tab, select or clear the *Pointer snap to grid* option.

 By default, the pointer snaps to the grid for connectivity and drawing objects.

5. Click *OK*.

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

Customizing Placement and Movement of Objects

You can customize the placement and movement of connectivity objects, such as parts and symbols, and drawing objects, such as Line, Polyline, Text, Rectangle, Ellipse, Arc, and Picture, on coarse and fine grid in the schematic editor. You can use the options provided in the *Grid Display* tab of the *Preferences* dialog box to complete this task.

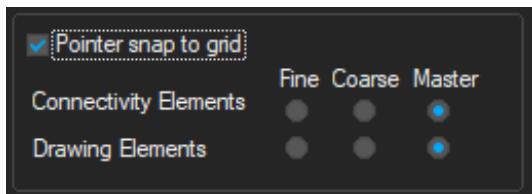


- ⚠
- These settings apply only to the schematic page grid; not to the part and symbol grid.
 - The settings are saved in the `CAPTURE.INI` file and it is used whenever you start the next Capture session.

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

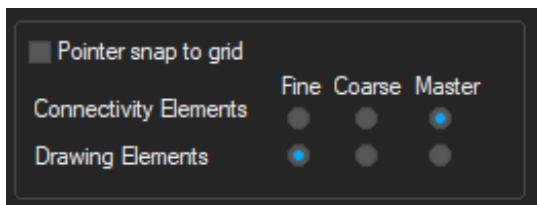
If...	Then...
<p>The <i>Master</i> option is selected for both Connectivity and Drawing Elements and the <i>Pointer snap to grid</i> check box is not selected:</p> <p>The screenshot shows the 'Grid Display' tab of the Preferences dialog. The 'Pointer snap to grid' checkbox is unchecked. Under 'Connectivity Elements' and 'Drawing Elements', the 'Master' radio button is selected.</p>	<p>The connectivity and drawing objects can be placed and moved only on the fine grid.</p>

The *Master* option is selected for both Connectivity and Drawing Elements and the *Pointer snap to grid* check box is selected:



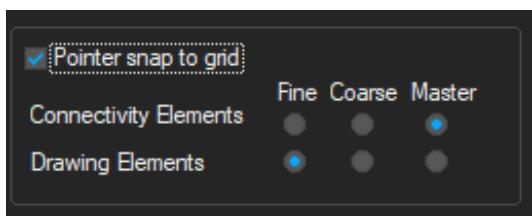
The connectivity and drawing objects can be placed and moved only on the coarse grid.

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:



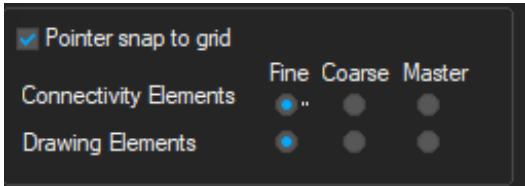
The connectivity and drawing objects can be placed and moved only on the fine grid.

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:



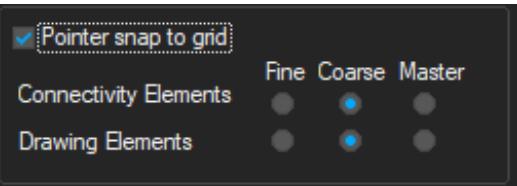
The connectivity objects can be placed and moved on the coarse grid and the drawing objects on the fine grid.

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:

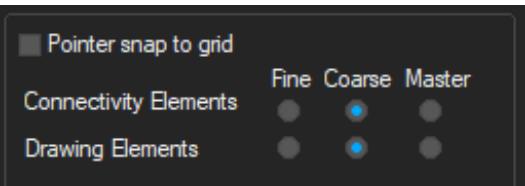


The connectivity and drawing objects can be placed and moved only on the fine grid.

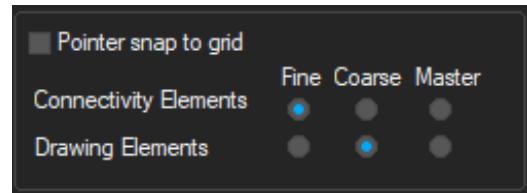
The *Coarse* option is selected for both Connectivity and Drawing Elements and the *Pointer snap to grid* check box is either selected or not selected:



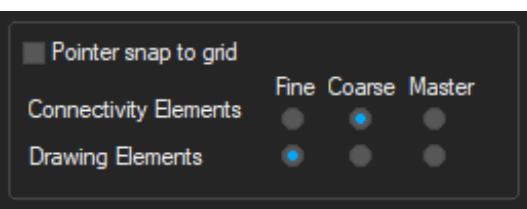
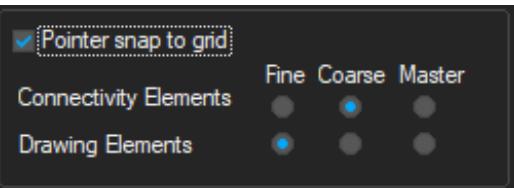
The connectivity and drawing objects can be placed and moved only on the coarse grid.



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:



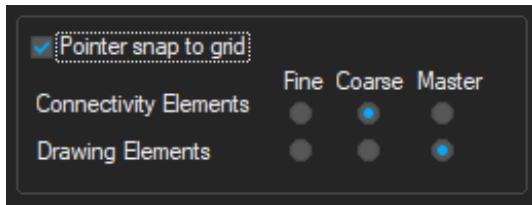
The *Coarse* option is selected for Connectivity Elements and the *Fine* option is selected for Drawing Elements, and the *Pointer snap to grid* check box is either selected or not selected:



The connectivity objects can be placed and moved on the fine grid and the drawing objects on the coarse grid.

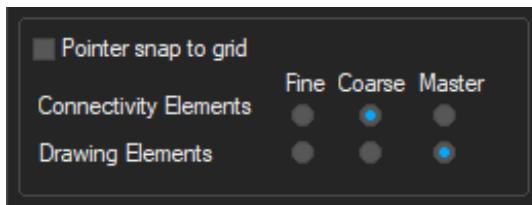
The connectivity objects can be placed and moved on the coarse grid and the drawing objects on the Fine grid.

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:



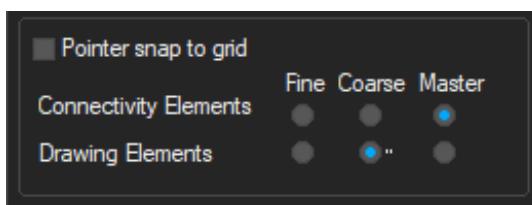
The connectivity and drawing objects can be placed and moved only on the coarse grid.

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.



- If the selection contains both connectivity and drawing objects, the precedence is given to the option selected 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.

Setting Pan and Zoom

You can 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. 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, do the following:

1. Choose *Options – Preferences*.
2. Select the *Pan and Zoom* tab.
3. For the schematic page editor and the part editor, set these options:
 - a. *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.
 - b. *Auto Scroll Percent*: Enter the percent of the window's horizontal or vertical dimension by which the display scrolls when the pointer approaches the edge of the window with an object attached.
4. Click *OK*.

Defining Selection Options

You can define the following selection options independently for Schematic Page Editor and Part and Symbol Editor:

- Select objects when they are completely enclosed in the selection area
- Select objects when the selection border intersects them
- The maximum number of objects to display at high resolution while dragging

To define selection options, do the following:

1. Choose *Options– Preferences*.
2. Select the *Select* tab.
3. For the schematic page editor and the part editor, set these options:
 - *Area Select*: Select either *Intersecting* or *Fully Enclosed*.

⚠ If the *Fully Enclosed* option is selected and you select an object on a schematic page, ensure that you select the object along with its name and number. Else, the object is not selected.

- *Maximum number of objects to display at high resolution while dragging*: If you drag more objects than you specify here, you see rectangular placeholders for the objects as you drag them.
4. Click *OK*.

Setting Miscellaneous Options

You can set the following options in the *Miscellaneous* tab of the *Preferences* dialog box:

- 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 X tools, such as PSpice and the PCB layout tools such as OrCAD X Presto.

The following table provides descriptions of the various options you can specify in the *Miscellaneous* tab:

Options	Descriptions
<i>Fill Style</i>	Specifies a fill pattern to be used when drawing rectangles, ellipses, and closed shapes drawn with the poly-line tool.
<i>Line Style and Width</i>	Specifies both line style and line width for lines, poly-lines, rectangles, ellipses, and arcs.
<i>Color</i>	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 overrides the default color, though changing this setting does not change the color of the objects already placed in the schematic page editor. Polylines and arcs use the default color of objects set in the <i>Colors</i> tab. You can change the fill style, line and width style, and color on individual objects using the Properties command on the Edit menu.
<i>Session Log</i>	Specifies the font for the project manager and Session Log. If you click the <i>Font</i> button, the <i>Project Manager and Session Log Font</i> dialog box appears displaying the standard Windows Font user interface for font selection. This option is neither a schematic page nor a part editor option.

Text Rendering	Determines how text on a schematic page appears on the screen, and how it is printed or plotted. The <i>Render TrueType fonts with strokes</i> option displays text as a series of lines, connected to resemble the outlines of the corresponding TrueType letters or numbers they represent. Enabling the <i>Fill text</i> option causes the text outlines to be filled in.
Auto Recovery	<p>Enables auto recovery for the project and determines the time 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 the project that is not saved, but modified since the last save, is saved as a temporary file with an <code>.ASP</code> extension in the <code>WINDOWS>TEMP>AUTOSAVE</code> directory.</p> <p>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, the <code>AUTOSAVE</code> directory and temporary files, are automatically deleted when you exit Capture normally. However, the temporary files are saved in case of a power outages or an unexpected system shutdown. When you restart Capture, it loads the auto-recovered files, showing <i>Restored</i> in the title bars of the recovered files. Use the <i>File – Save As</i> command and provide a file name to have an auto recovered file overwrite the original file.</p>
Auto Reference	Enables automatic annotating of reference designators when parts are placed.
Inter-tool Communication	Enables inter-tool communication (ITC), so that you can test and display design information using other OrCAD X tools, such as OrCAD X Presto and PSpice, in conjunction with Capture. Capture processes its designs faster when inter-tool communication is not selected.

To set miscellaneous options, do the following:

1. Choose *Options – Preferences* from the main menu.
2. Select the *Miscellaneous* tab.
3. For the schematic page editor and the part editor, set the *Fill Style* and *Line Style and Width* options.
4. For the schematic page editor, set the *Color* option.
5. Set the following options as required:
 - *Session Log*
 - *Text Rendering*

- Auto Recovery
 - Auto Reference
 - Inter-tool Communication
6. Click *OK*.

Setting Text Editor Options

You can 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.

To specify the text editor preferences, do the following:

1. Choose *Options – Preferences* from the main menu.
2. Select the *Text Editor* tab.
3. Set the following options as required:
 - 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 Setting: Set the tab spacing for the text editor.
4. Select the *Highlight Keywords, Comments, and Quoted Strings* option to have the VHDL items highlighted in the text editor. The colors used to highlight these items are the ones set in the *Syntax Highlighting* group box.
This option must be enabled for Capture to use the syntax highlighting options.
5. Click the *Reset* button to reset the text editor options to the Capture default values.
6. Click *OK*.

Setting up the Design Template

You set the design template using the [Design Template or 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 is 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 the schematic pages you add to an existing project.

Using the Design Template/Design Properties dialog box, you can do the following tasks:

- [Defining Fonts for New Designs](#)
- [Defining Title Block](#)
- [Setting up a New Page](#)
- [Defining Grid References](#)
- [Setting up Default Hierarchy](#)
- [Setting SDT Compatibility](#)

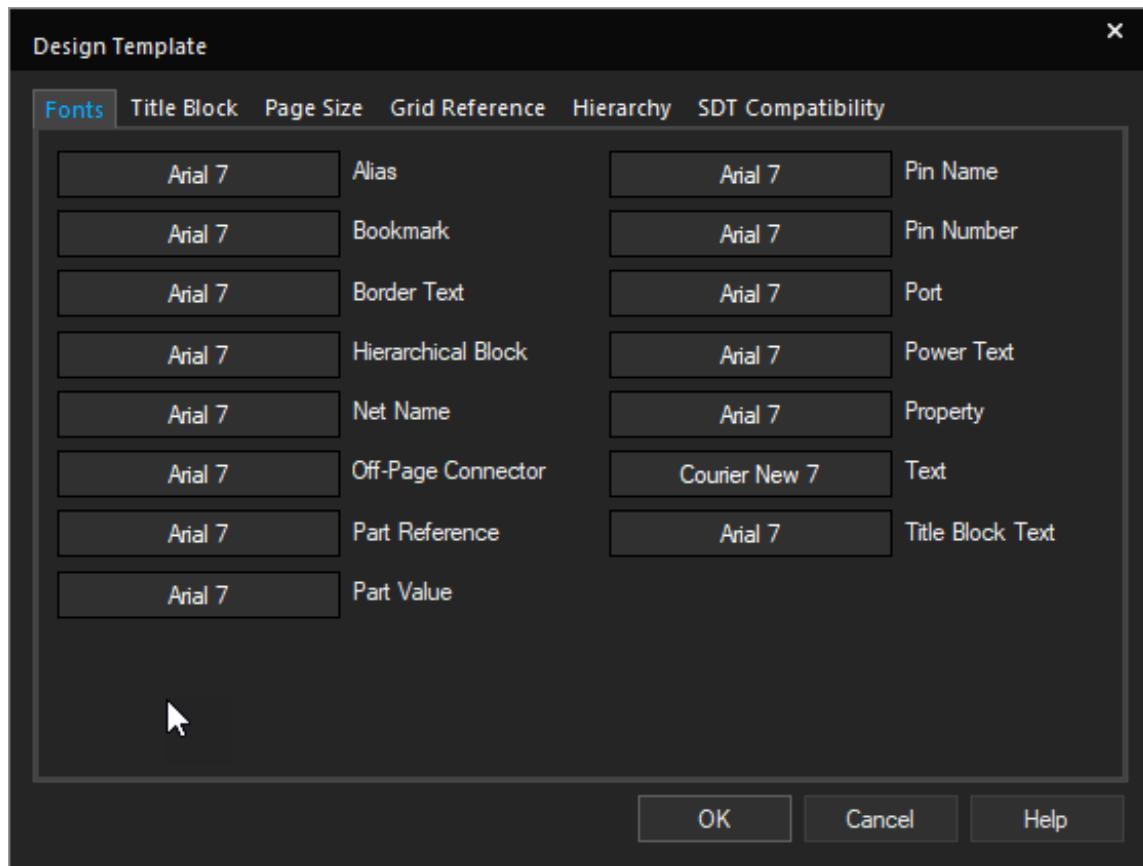
Defining Fonts for New Designs

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.

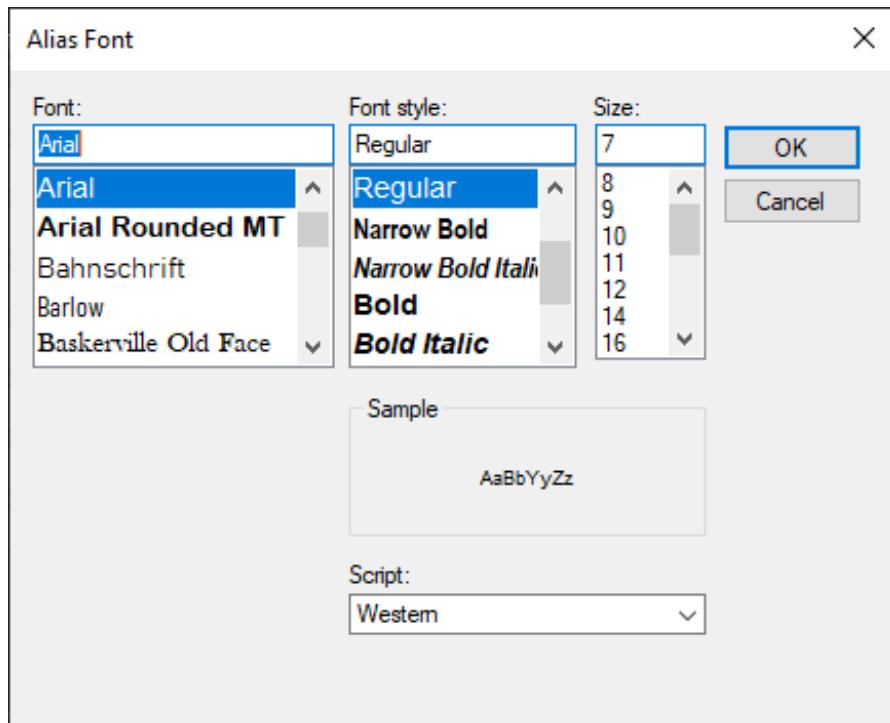
To assign fonts for new designs, do the following

1. Choose *Options – Design Template* from the main menu.

The Design Template dialog box is displayed.



2. In the *Fonts* tab, click the font of an item. For example, click the font for *Alias*.
The standard Windows font dialog box appears.



3. Select a font, font style, and size.
4. Click *OK* to close the Font dialog box.
5. Click *OK*.

⚠ The default fonts provide optimal compatibility with SDT. Changing these fonts may result in less than optimal text sizes for translated projects.

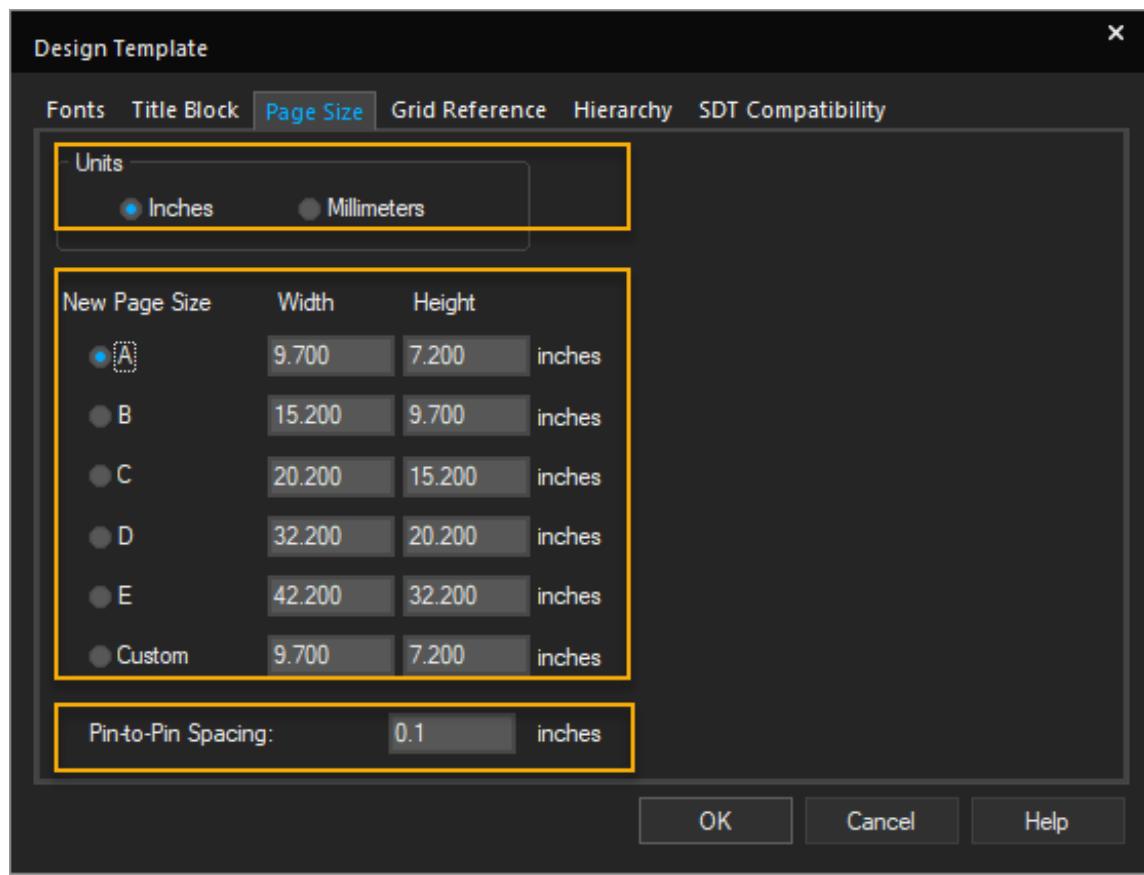
Setting up a New Page

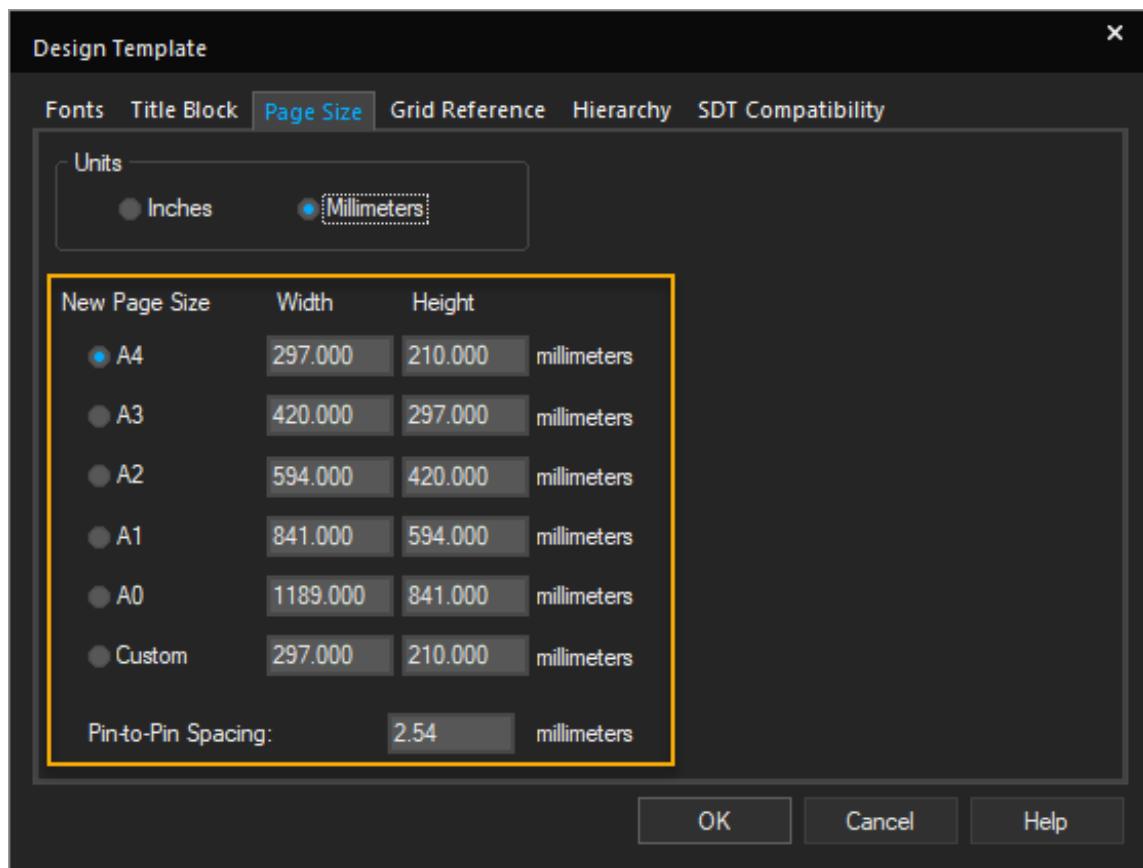
You can specify either *inches* or *millimeters* as the unit of measure for the width and height of a schematic page, and the spacing between pins in the *Page Size* tab. For new projects, you can specify the default unit of measure, the default width and height of schematic pages, and the spacing between 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, the space between grid dots or grid lines.

To set up the schematic page size, do the following:

1. Choose *Options – Design Template* from the main menu.
2. Select the *Page Size* tab.
3. 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.

 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.





4. Select the default schematic page size for new projects from the following page sizes:
 - *A, B, C, D, E*, and *Custom*, if the unit of measure is Inches.
 - *A4, A3, A2, A1, A0*, and *Custom*, if the unit of measure is Millimeters.
5. Specify the width and the height.

The values that you enter in the *Width* and *Height* text boxes translate to the dimensions for each page size. You cannot change these dimensions for individual schematic pages, although you can select a different page size, or define a custom size.
6. In the *Pin-to-Pin Spacing* text box, specify the default spacing between pins, grid dots, or grid lines.

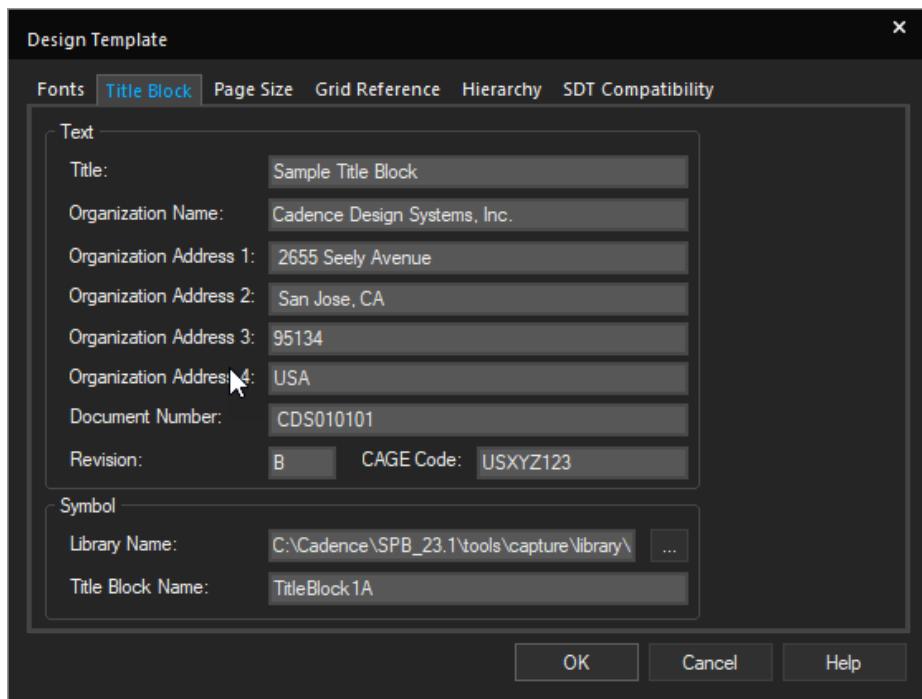
You cannot change this value for existing projects or individual schematic pages.

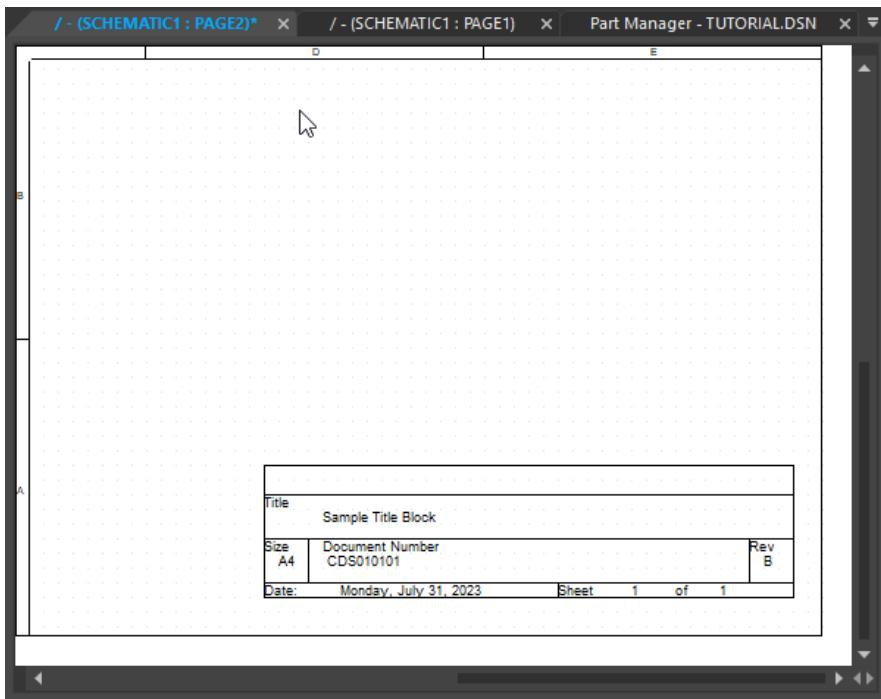
 Part size will vary when copying and pasting parts between pages with different pin-to-pin spacings

7. Click *OK*.

Defining Title Block

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](#) or [Design Properties](#) dialog box in the lower-right corner of each new schematic page. You 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.





Design Template Dialog Box – Title Block Tab Schematic Page with Title Block

⚠ You may place any number of optional title blocks anywhere on the schematic page, using the *Place – Title Block* command from the main 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 on 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 visibility of the default title block 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` does not 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* dialog box.

The following table lists the default title block properties that you specify in the *Title Block* tab of the *Design Template* dialog box:

Field	Description
<i>Text</i>	
Title	Specifies the title of the schematic page that appears in the title block placed on the page.
Organization Name	Specifies the name of the organization.
Organization Address 1	Specifies the first line of the organization address.
Organization Address 2	Specifies the second line of the organization address.
Organization Address 3	Specifies the third line of the organization address.
Organization Address 4	Specifies the fourth line of the organization address.
Revision	Specifies the revision.
Document Number	Specifies the document number.
CAGE Code	Specifies the <i>CAGE Code</i> . CAGE is an acronym for <i>Commercial and Government Entity</i> . It is a numeric code provided by the federal government to its suppliers. This code is used in the title block of a schematic page.
<i>Symbol</i>	

Library Name	Specifies the name and path of the source library that contains the title block symbols.
Title Block Name	Specifies the name of the symbol for the title block to be used from the source library.

⚠ You can create custom title blocks and store them in a library using the *Design – New Symbol* command from the project manager. If you specify the name of the custom library and title block in the *Title Block* tab, the custom title block appears in the lower right corner of each new schematic page. In addition, you can edit title block information in the schematic page editor

To choose a title block and define its contents, do the following:

1. Choose *Options – Design Template*.
2. In the *Design Template* dialog box, choose the *Title Block* tab.
3. In the Text group box, specify the required information to display in the title block on every page.
4. In the Symbol group box, specify 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.
5. Enter the exact name of the title block in the *Title Block Name* text box.
Symbol names are case sensitive and space sensitive.
6. Click *OK*.
The information added here automatically populates the title blocks on any new page.

Editing Title Block

To edit title block information, do the following

1. Select the information string on the title block that needs to change.
2. Choose *Edit – Properties*.
Alternatively double-click the string in the title block.

The *Display Properties* dialog box appears.

3. Update the values as required and click *OK*.

The information is updated 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 a new value.

6. Click *Apply*, and close the property editor.

The schematic page editor appears with the updated information in the title block.

To edit title block information on multiple pages, do the following:

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.

6. Edit the property values for one or more schematic pages at a time.

7. Click *OK*.

8. Click *Yes* to confirm.

To change the display of title block information, do the following:

1. Select the information string on the title block that needs to change.

2. Choose *Edit – Properties*.

3. In the *Display Properties* dialog box, select the *Change* button in the Font group box.

The *FONTs* dialog box appears.

4. Change the display properties as required and click *OK* to close the *Font* dialog box.

5. Click *OK*.

Defining Grid References

You define border grid references in the *Grid Reference* tab of the *Design Template* dialog box.

For horizontal and vertical border grid references, you can specify:

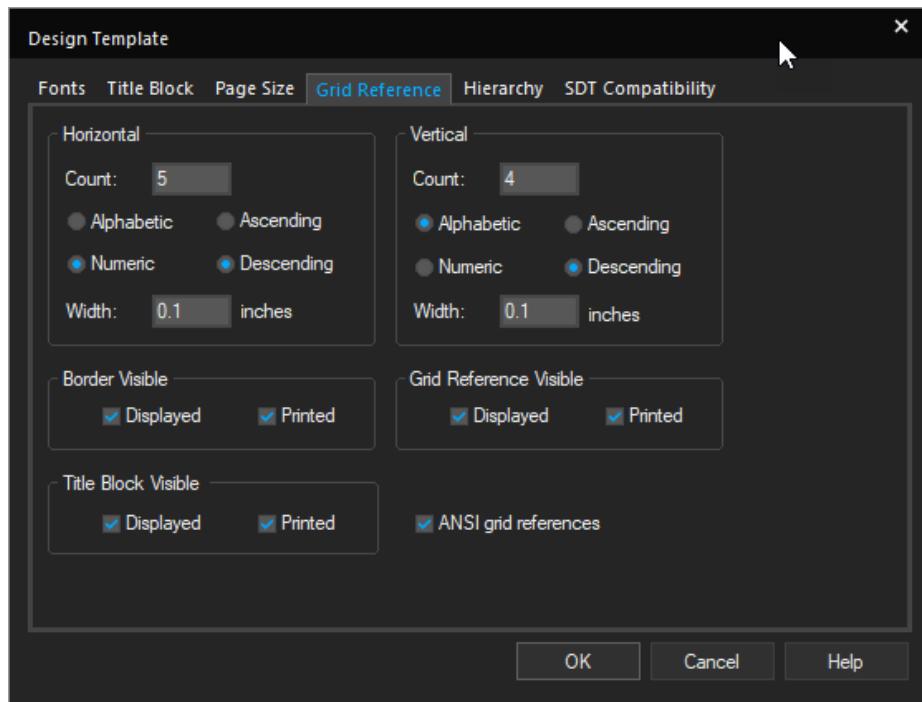
- The number of border grid references to be displayed.
- The use of either alphabetic or numeric characters for grid reference.
- The order in which to sort grid references across the schematic page—ascending or descending.
- The width of the grid reference cells.

You can also control the visibility of the border, grid references, and title block on the screen and on the schematic pages you print. The settings are applied on new projects.

 You can change these settings for existing schematic pages. Choose *Schematic Page Properties* from the *Options* menu in schematic page editor and select the *Grid Reference* tab in the *Schematic Page Properties* dialog box.

To define the grid reference, do the following:

1. Choose *Options – Design Template*.
2. Select the *Grid Reference* tab.



3. Specify the required [grid reference](#) values in the Horizontal and Vertical group boxes.

⚠ The size of the Grid Reference font is tied to the width.

4. Change the visibility of the border, title block, and grid reference, by selecting *Displayed* to display them on the screen or *Printed* to print on the schematic pages you print.
5. Select *ANSI grid references* to display the grid references in accordance with the ANSI standards.
6. Click *OK*.

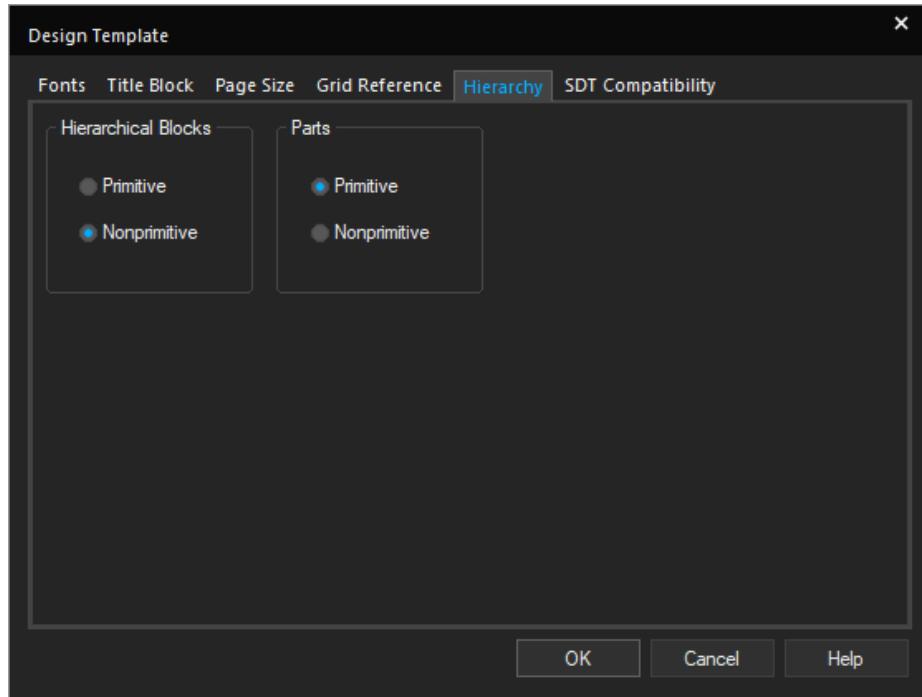
Setting up Default Hierarchy

For hierarchical blocks and part instances with the *Primitive* property set to *Default*, you can specify if you want Capture to treat each block or part as primitive (cannot descend into attached schematic folders) or nonprimitive (can descend into attached schematic folders). The *Primitive* and *Nonprimitive* options only affect new projects. 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.

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

To define the default hierarchy options, do the following:

1. Choose *Options– Design Template*.
2. Select the *Hierarchy* tab.



3. 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.

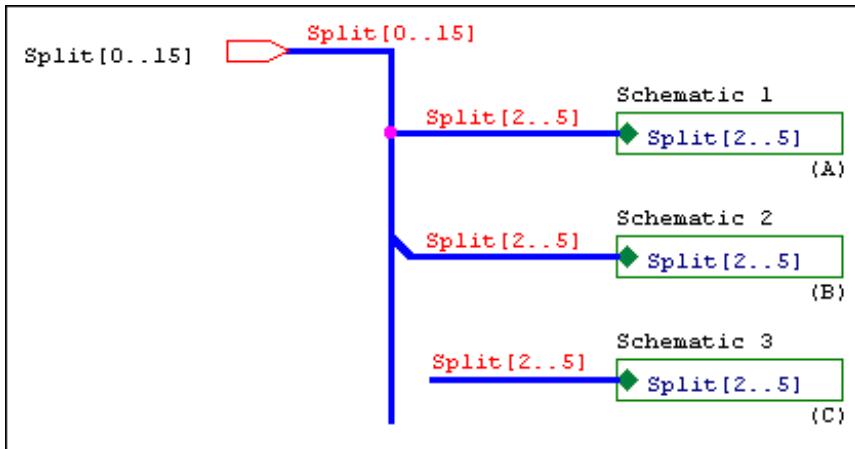
4. Click **OK**.

Setting SDT Compatibility

You can map Capture properties to the corresponding OrCAD X Schematic Design Tools (SDT) part fields when saving a project in the SDT format. Capture uses the SDT compatibility options in the *Design Template and 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 different set of connectivity rules than SDT. The following cases explain these 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	Yes

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

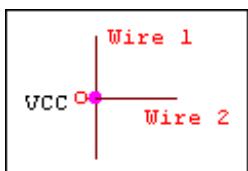
Case 2



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

SDT	Yes
Capture	No. Wires in Capture are for single signals only.

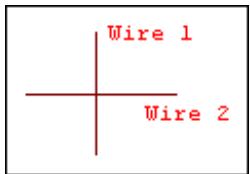
Case 3



The wire connects to the power symbol.

SDT	No
Capture	Yes

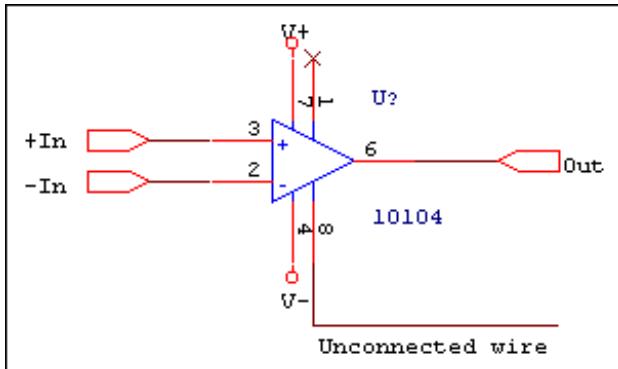
Case 4



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.

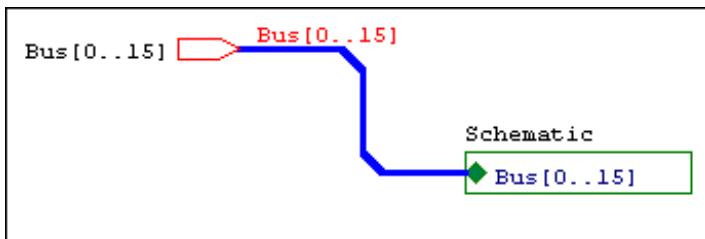
Case 5



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

SDT	No
Capture	Yes

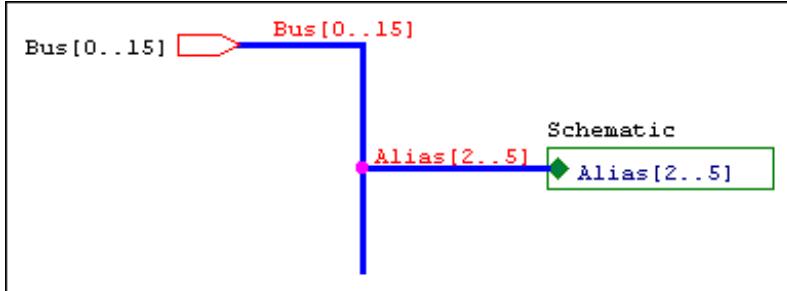
Case 6



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

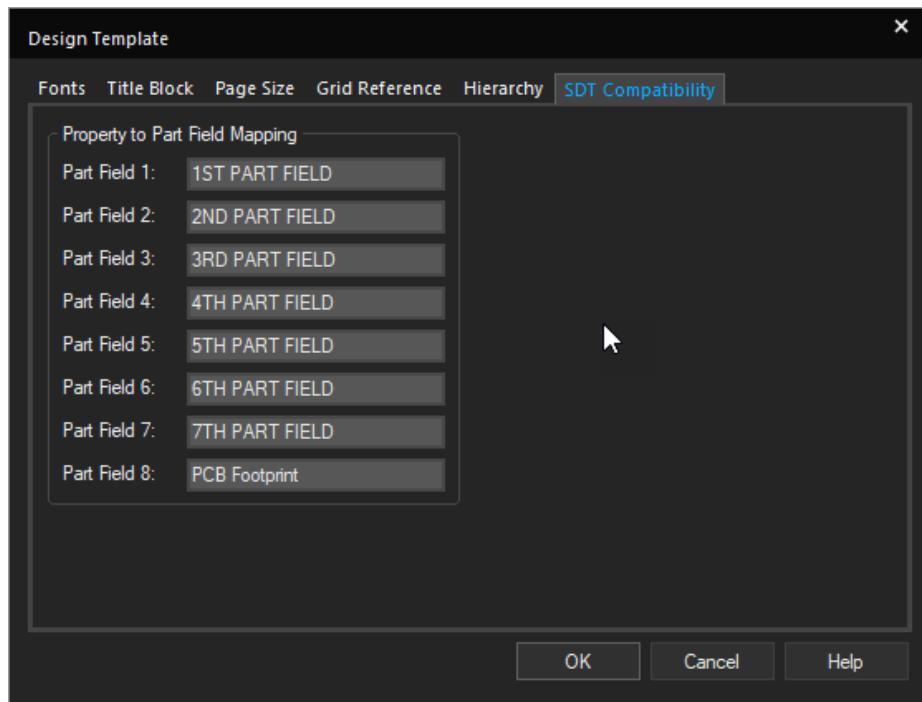
You can specify the properties that Capture stores in the eight SDT part fields when saving a project in the SDT format. You can also use the part fields for mapping netlists that use part field information.

⚠ 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 X Schematic Design Tools, do the following:

When you create a new design, the SDT compatibility options are inherited from the design template. To set up the design template for SDT compatibility, do the following:

1. Choose *Options – Design Template* from the main menu.
2. Select the *SDT Compatibility* tab.

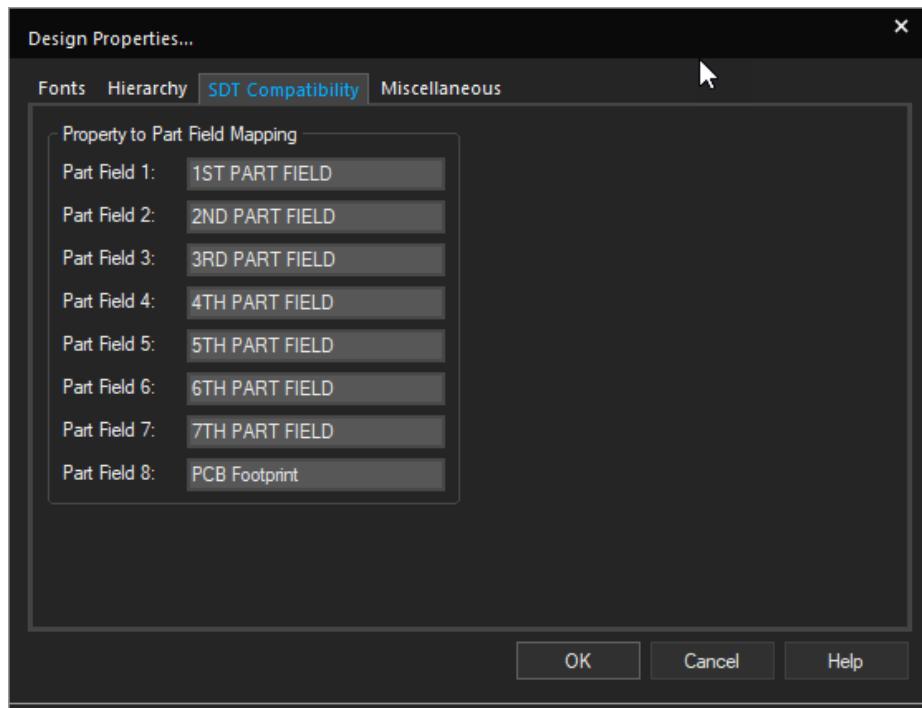


3. For each Capture property to be mapped to an SDT part field, specify the part field to contain the property value in the *Properties to Part Field Mapping* group box.
4. Click *OK*.

Changing SDT Compatibility Options for a Single Design

When you save a design in the SDT format, Capture uses the SDT compatibility options in the *Design Properties* dialog box. To change the SDT compatibility options for a design, do the following:

1. With the project manager active, select the design folder.
2. Choose *Options – Design Properties*.
3. Select the *SDT Compatibility* tab.



4. Specify the properties to be mapped to the SDT part fields for the active design in the *Properties to Part Field Mapping* group box.
5. Click *OK*.

Translating Part Fields from SDT to Capture Properties

Capture translates SDT part fields into properties. To change the user property names before translation, do the following:

1. Open the `SDT.CFG` file for the design in any text editor.
2. Locate the lines that specify the part field names, and change them to suit your requirement.
3. Save the 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*.
2. Select the *SDT Compatibility* tab.
3. Specify the properties to be mapped to the SDT part fields.

Moving Objects by Drag and Drop

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 X Capture install directory (<Installation Directory>\tools\capture) structure. The section also provides a brief description of the files types associated with OrCAD X Capture.

Capture directory contents

CAPTURE.EXE	The Capture executable. It appears in the OrCAD X Desktop program group as Capture.
CAPTURE.INI	<p>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:</p> <pre>CAPTURE -I directory\\ CAPTURE /I directory\\</pre> <p>where path is the directory where the .INI is located. For example:</p> <pre>CAPTURE -I C:\\CAPTURE</pre>
*.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 X 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:
 - a. In the *Look in* drop-down list, select a new drive or directory.
 - b. In the File name text box, enter a portion of the file name—you can use the standard "*" and "?" wildcard characters.
 - c. 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



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.

Managing Workspaces

With the OrCAD X Professional (POX200 Pro) license, OrCAD X Capture provides a collaborative development environment where you can create shared workspaces containing work-in-progress components, designs, and library and project files. With this license you can share workspaces and provide users access rights on the shared workspaces. A workspace is a cloud-based project storage location that is used to manage all the design data, which includes categories (templates) and components, and design, library, project, and board files.

The OrCAD X Professional (POX200 Pro) license supports the following features:

- Multiple workspaces can be added
- Members can be added to or removed from workspaces
- Privileges can be assigned to subscribed members
- Users can have subscription to multiple workspaces
- Users can have different access privileges for different workspaces

The OrCAD X Standard (POX100 Standard) license supports the following features:

- Users can save versions of an object, project, or library
- Shared workspace objects can be edited and published based on the assigned privileges

This section covers the following topics:

- [Configuring Workspaces](#)
- [Sharing Workspaces](#)
- [Managing Workspace Files](#)
- [Sharing Projects](#)

Configuring Workspaces

With the OrCAD X Professional license, OrCAD X Capture provides a comprehensive part development environment where you can create shared workspaces containing work-in-progress components, designs, libraries, board files, and all the other project related files. You can manage the files in workspaces from [File Manager](#) and components from [Component Explorer](#).

Creating a Workspace

To create a [workspace](#), do the following:

1. Choose *View – Workspace – Configuration* from the main menu.

Alternatively, Right-click *My Workspace* in [Component Explorer](#) and choose *Workspace Configuration*.

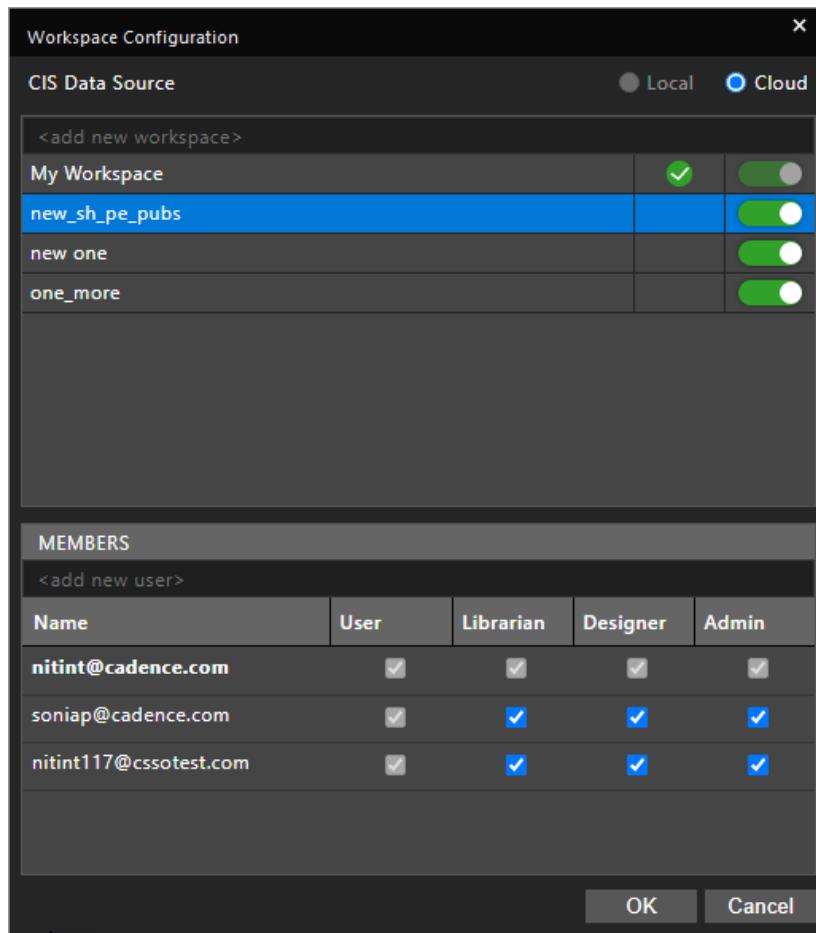
The *Workspace Configuration* dialog box opens. You add new workspaces and add or remove members here.

2. Select the CIS data source.

For workspace configuration, *Cloud* is selected as the default CIS data source.

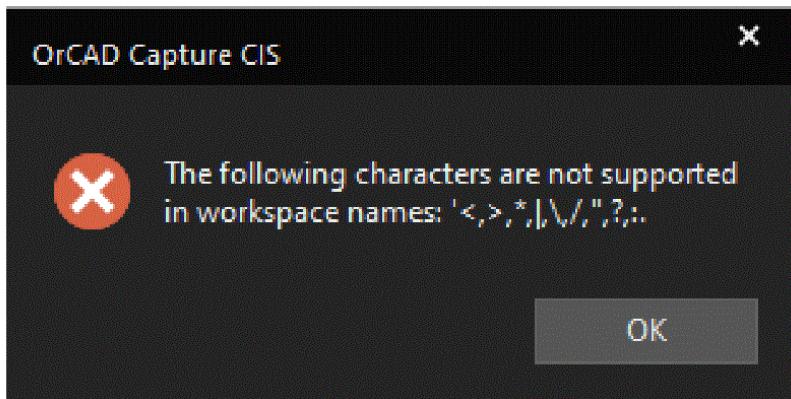
CIS Data Source	When to Use...
Local	The <i>Local ODBC</i> option lets you continue with a known good <i>CIS Data Source</i> . You can continue working seamlessly with the existing database and start using Cloud data management capabilities for the designs. If there is no CIS configuration found for the project, this option button remains disabled.
Cloud	The <i>Cloud</i> option enables you to leverage the part authoring and integrated Cloud CIS data management capabilities.

You provide the members access on the workspaces by assigning them roles with pre-defined access permissions to the workspaces.



3. Specify a name for the workspace in the <add new workspace> text box and press Enter.

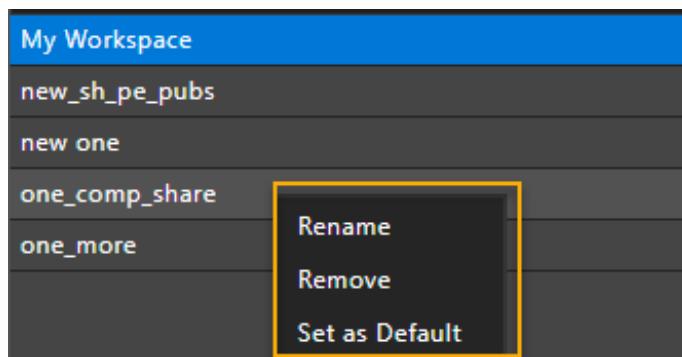
⚠ Not all special character are supported in a workspace name. A message pops up if you specify any unsupported character.



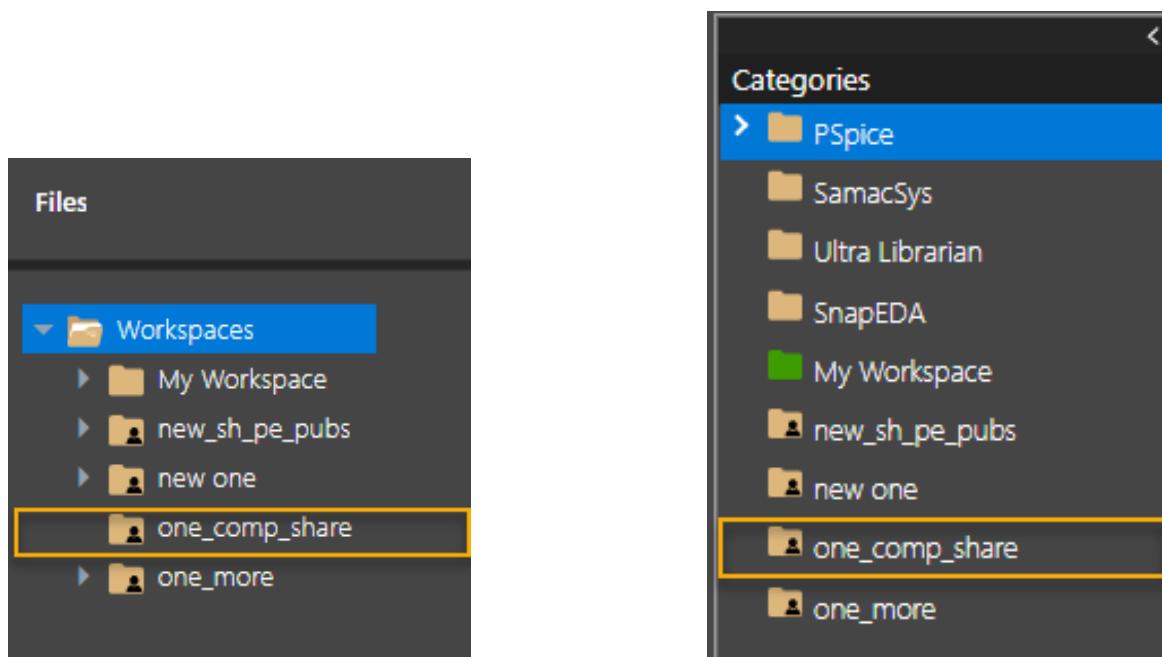
If valid characters are specified for the workspace name, a workspace with that name is created.

<add new workspace>			
My Workspace	✓	<input checked="" type="checkbox"/>	<input type="checkbox"/>
one_comp_share		<input type="checkbox"/>	<input checked="" type="checkbox"/>
new_sh_pe_pubs		<input type="checkbox"/>	<input checked="" type="checkbox"/>
new one		<input type="checkbox"/>	<input checked="" type="checkbox"/>
one_more		<input type="checkbox"/>	<input checked="" type="checkbox"/>

4. Use the toggle button to show or hide the workspace in [Component Explorer](#) or [File Manager](#). You can rename or remove a workspace by using the corresponding commands on the right-click shortcut menu on the workspace name.



5. Click **OK**.
The shared workspace appears in the File Manager and Component Explorer interfaces.



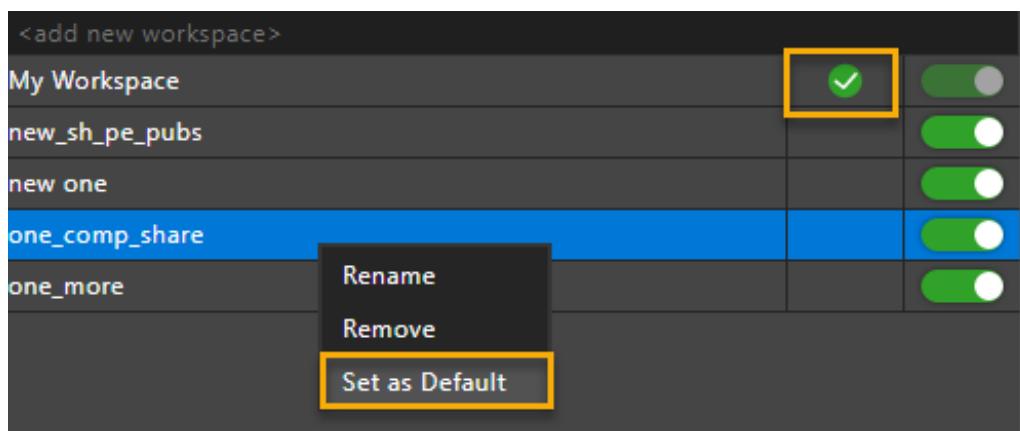
ⓘ You can move across shared workspaces in the same session of Capture CIS.

Setting up a Default Workspace

My Workspace is the default workspace for all operations. However, you can set any workspace as the default workspace. The default Cloud workspace database is referred by [Part Manager](#) to link and update database components.

To set a workspace as a default, do the following:

1. Right-click the workspace and choose *Set as Default*.

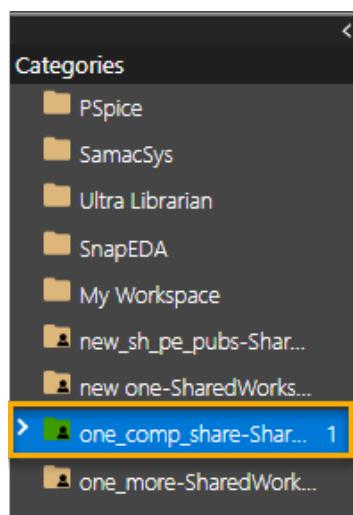


The green tick icon moves next to the new default workspace.

My Workspace	<input type="checkbox"/>
new_sh_pe_pubs	<input checked="" type="checkbox"/>
new one	<input checked="" type="checkbox"/>
one_comp_share	<input checked="" type="checkbox"/>
one_more	<input checked="" type="checkbox"/>

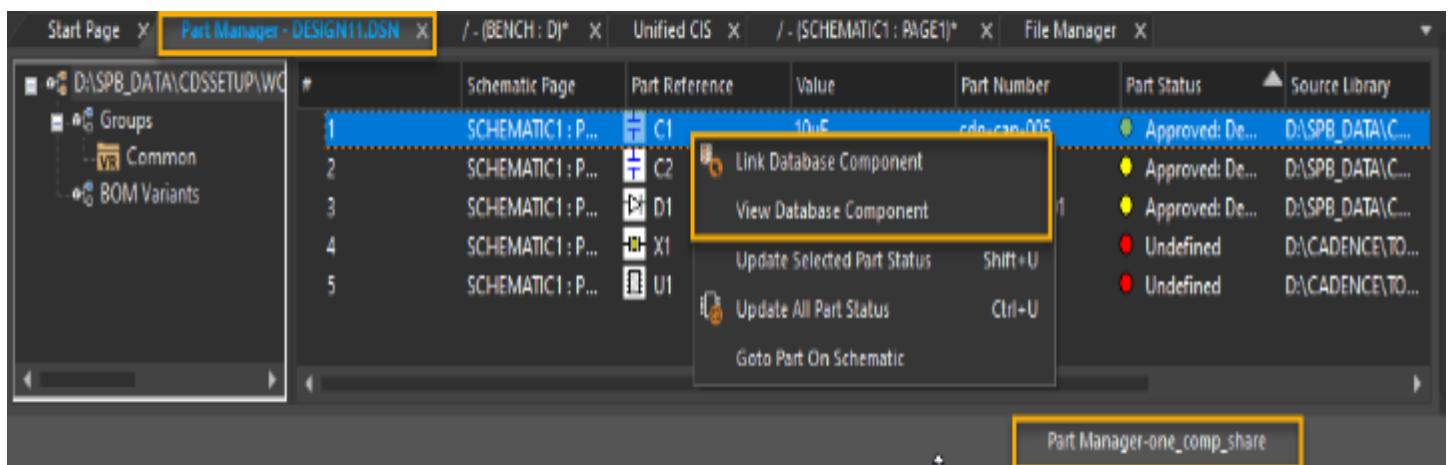
2. Click *OK* to confirm.

The workspace icon changes to green indicating that now this is the default workspace.



This workspace is the default source database for design authoring and for use in part manager.

 The Part Manager status bar displays the name of the default workspace as the data source.



When you run the [Link Database Component](#) or [View Database Component](#) commands on a part from Part Manager or the schematic page, the component is queried from the default workspace.

View Database Component-one_comp_share		
Name	Value In Database	Value On Schematic
PART NUMBER	cdn-diode-001	cdn-diode-001
VR	100V	100V
VF	0.7V	0.7V
IO	1A	1A
IR	1uA	1uA
POWER	0.25W	0.25W
PACKAGE	DO27	DO27
MAX_TEMPERATURE	+150C	+150C
MIN_TEMPERATURE	-40C	-40C
CLASS	DISCRETE	IC
PCB FOOTPRINT	do27a	do27a
MANUFACTURER PART NUMBER	cdn-diode-001	
PART DESCRIPTION	Switching Diode	
MANUFACTURER NAME	None	
HAVE SYMBOL	1	
		<button>Update</button> <button>Close</button>

Related Topics

- [File Manager](#)
- [Component Explorer](#)

Sharing Workspaces

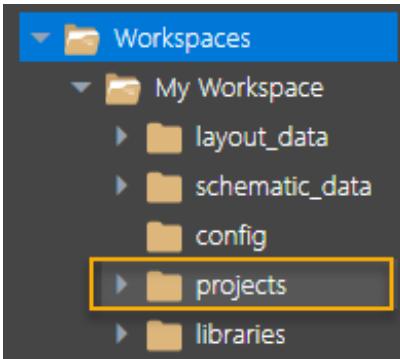
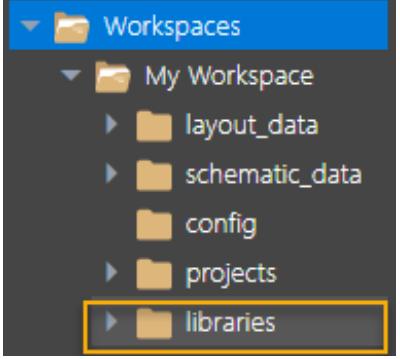
A workspace owner or admin can share a workspace by assigning access permissions to users. An Admin adds members and assigns specific roles to them based on their work profile. Each role has associated privileges that are transferred to the member who is assigned the role. You grant access rights to your team members on a workspace in the MEMBERS section of the *Workspace Configuration* dialog box.

The following image lists the roles supported in a shared workspace :

User	Librarian	Designer	Admin
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

The following table provides a description of all the supported roles:

Role	Description
Admin	Provides complete access on all the objects in a shared workspace. With the <i>Admin</i> role, you can perform all activities possible on the shared workspace, such as modifying user access, editing objects, overriding the lock on objects. By default, the creator or owner of the workspace is assigned the Admin role on the workspace that cannot be modified.

Designer	<p>Provides the capability to update and modify a design and also check in (publish) or check out (edit) designs in a shared workspace. A user with the <i>Designer</i> role has full access on all file types in the <i>projects</i> folder of File Manager. You can perform all Create, Read, Update, and Delete (CRUD) operations on all the files in the folder.</p> 
Librarian	<p>Provides part authoring, editing, and publishing capabilities in a shared workspace. A user with the <i>Librarian</i> role has full access on all file types in the <i>libraries</i> folder of File Manager. You can perform all Create, Read, Update, and Delete (CRUD) operations on all the files in the folder.</p> 
User	<p>Provides a read-only access to design and library objects in a shared workspace. With the <i>User</i> role, you can only view the design and library files.</p>

When a workspace is created, the user creating the workspace is designated as the default owner of the workspace with *Admin* rights, and the username appears in bold in the MEMBERS section.

To share a workspace, do the following:

1. In the *MEMBERS* section of the *Workspace Configuration* dialog box, specify the user id of the team members to share the workspace with in the <add new user> text box and press Enter.

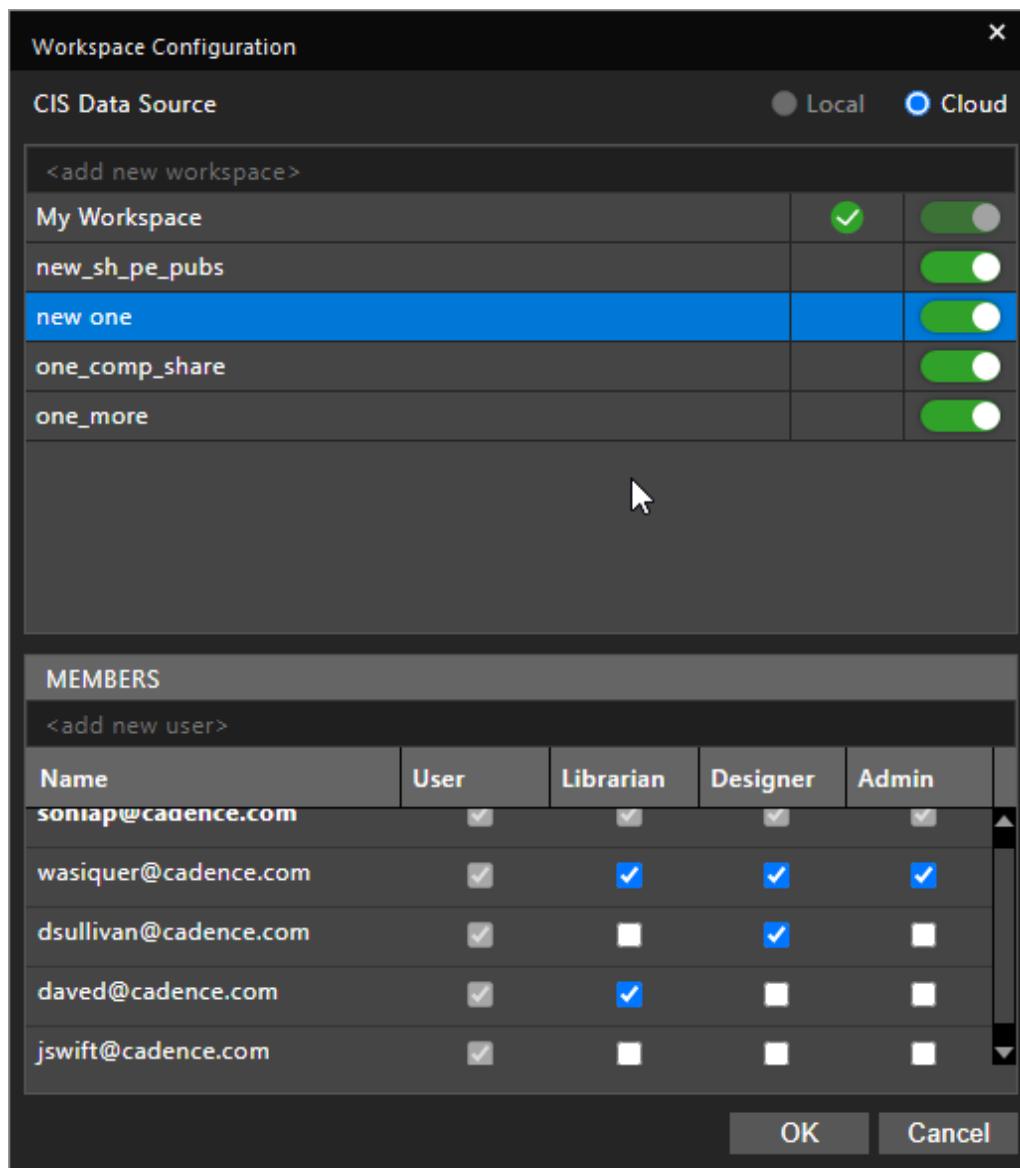
 In the current version, you can share a workspace with up to ten team members.

If the user id is present in the list, a red border appears around the text box and you cannot proceed without changing the user id.

MEMBERS				
Name	User	Librarian	Designer	Admin
dsullivan@cadence.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
soniap@cadence.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
wasiquer@cadence.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
dsullivan@cadence.com	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

By default, a new member is assigned the most basic role of a *User*. Also note that the *Admin* role has the highest level of access.

2. Select the required check box to assign the corresponding access rights to the member.



If you modify a role, any subset role check box is updated automatically. For example, if you deselect either the *Designer* or *Librarian* check box for a member, the *Admin* check box is automatically deselected.

- ⓘ Any member with a non-Admin role can view all the role details in the *Workspace Configuration* dialog box, but cannot edit the roles as the check boxes are grayed out.
Also, the owner of the workspace always appears grayed out and no member with any role, including the *Admin* role, can change the rights or role or delete the owner from the workspace.

3. Click *OK*.

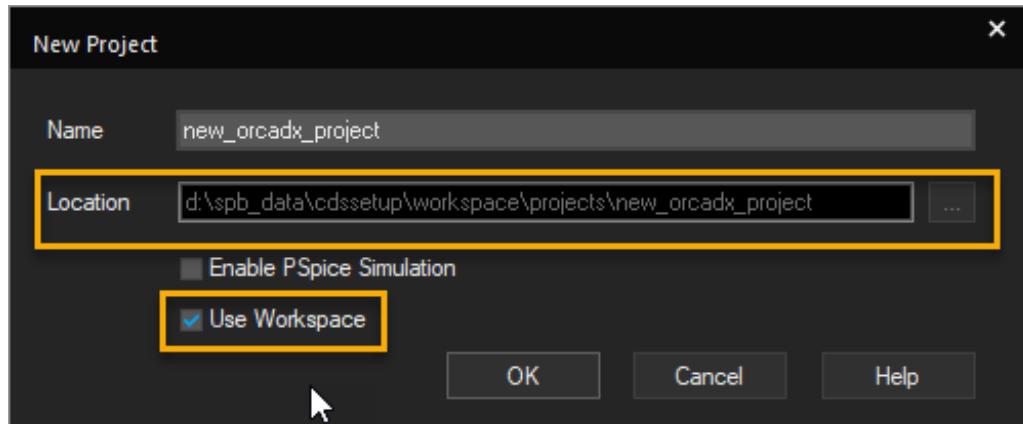
The members can now view this workspace in the [File Manager](#) and [Component Explorer](#) interfaces.

Managing Workspace Files

With the OrCAD X Professional (POX200 Pro) and OrCAD X Standard (POX100 Standard) licenses, you can seamlessly access your OrCAD X Capture designs, libraries, components, and other design elements across devices.

At the time of creating a new project, you can decide if you want to store the project in the OrCAD X Cloud workspace. If you select the *Use Workspace* option in the *New Project* dialog box, the location is pre-seeded and cannot be changed:

```
%HOME%\cdssetup\workspace\
```

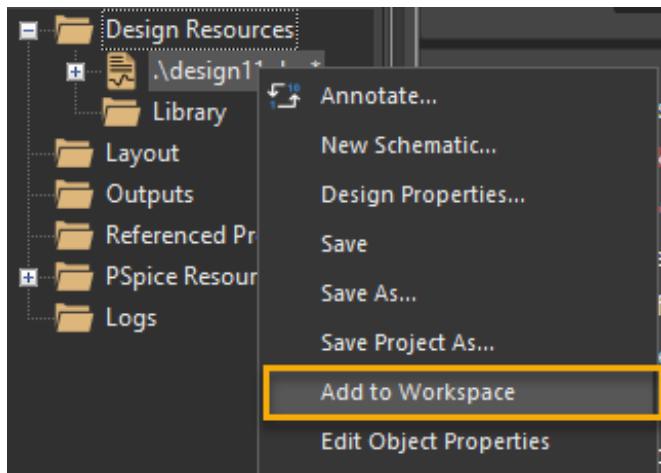


This project can be accessed from the [File Manager](#) window, which provides an interface to manage all the design files stored on the Cloud. For the same OrCAD X Cloud login credentials, the design data is automatically synchronized with OrCAD X Cloud across devices to maintain consistency.

Adding a Local Project to a Workspace

You can also add a project from your local disk to the Cloud workspace and then access it from any device.

- To add a local project to the Cloud Workspace, choose *File – Add to Workspace* from the main menu. Alternatively, right-click the design folder in project manager and choose *Add to Workspace*.

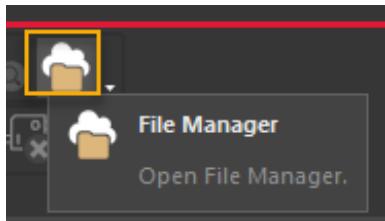


All the files related to the project are added to the workspace. When you open File Manager, you see all the files added to the workspace folder.

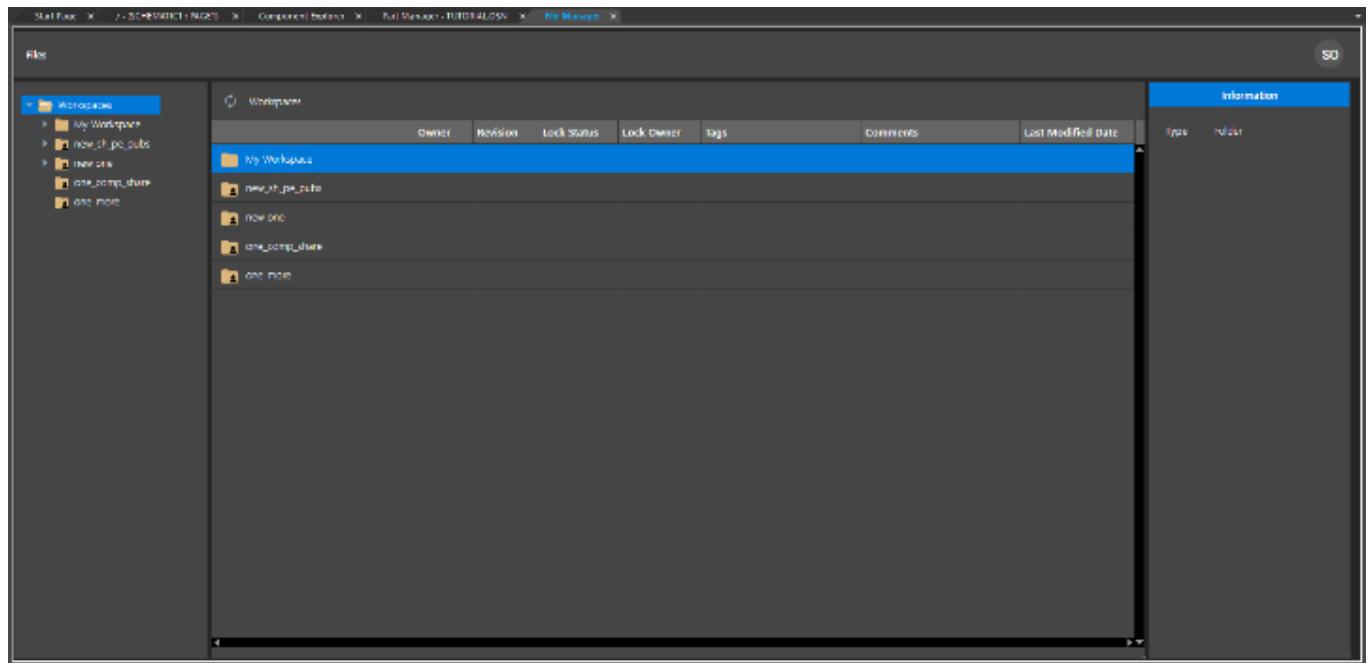
Viewing Status of Project Files in a Workspace

The [File Manager](#) window offers you a unified location to access and manage all the workspaces and all the files and folders related to projects in the workspaces ensuring that they are in sync with the Cloud data.

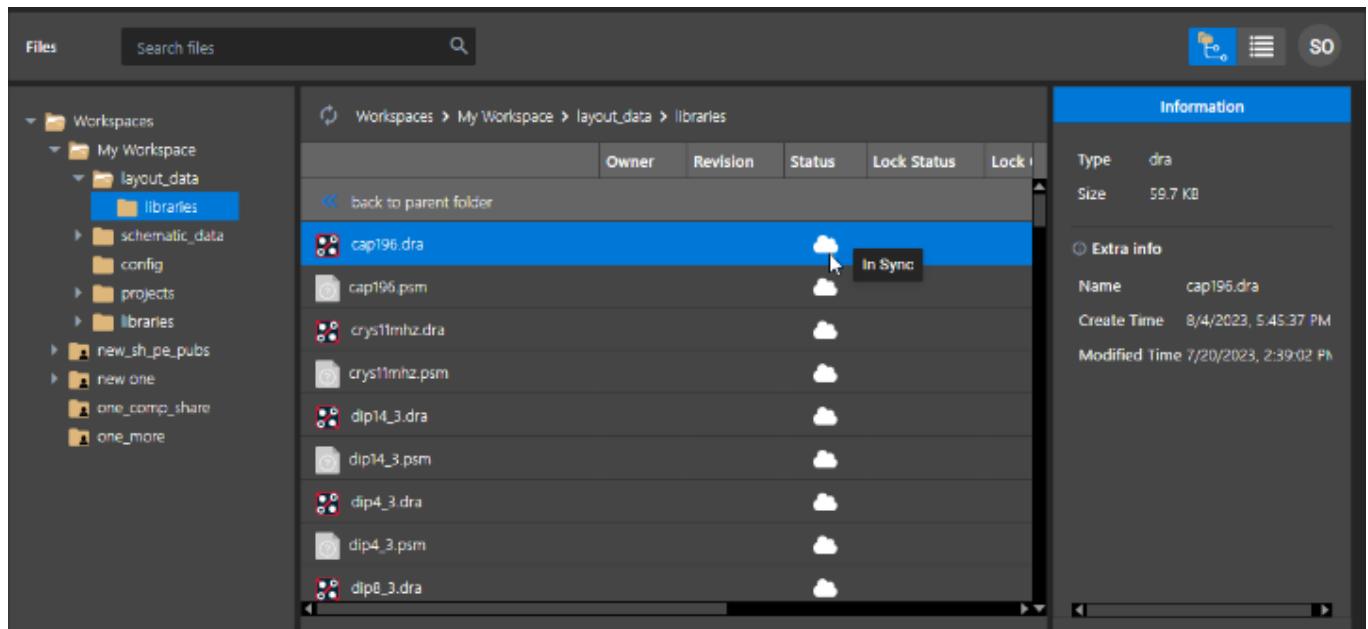
1. To open File Manager, choose *View – Workspace – File Manager* from the main menu.
Alternatively, click the *File Manager* icon:



All the workspaces available to you are displayed in the left pane.



2. Descend into the folder structure to view the files and folders.



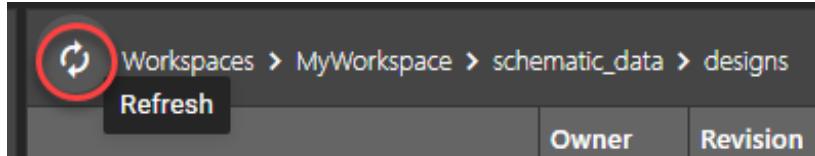
On selecting a folder in the left pane, the files and subfolders in the folder are displayed in the middle pane. The white cloud icon in the *Status* column indicates that the files in local workspace and the Cloud are *In Sync*. If you hover the mouse pointer over the image in the *Status* column, the datatip displays the current sync status of the file.

⚠ If the `node.exe` process is manually ended from the Task Manager while the client and Cloud server are being synchronized, the sync log reports the status of a previously-synced project as *Processing Started*. Before you manually end `node.exe`, ensure that the sync process is complete and the sync status for all the files in File Manager is *In Sync*.

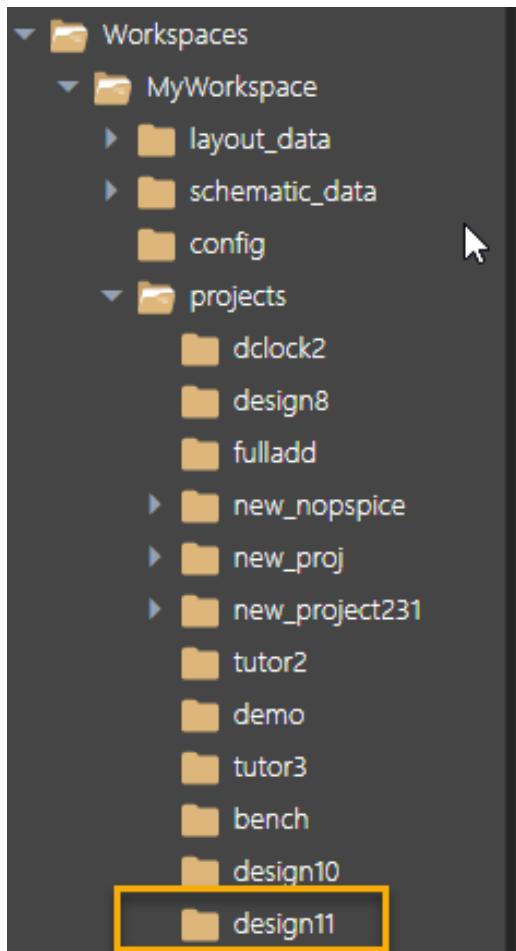
Refreshing File Status

When any file is published, updated, locked, or unlocked in a workspace, the file status is automatically refreshed. You can also manually refresh File Manager to sync the changes.

- Click the *Refresh* icon to update the File Manager view with the latest files.



For example, if you add a new project to the workspace, and click the *Refresh* icon in the File Manager, the newly added project is updated in the *My Workspace* node on the left pane of File Manager:

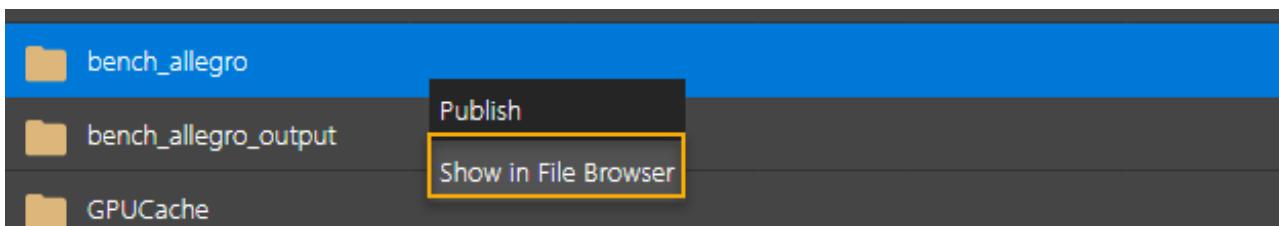


⚠ If there is a discrepancy in the sync time between the client and the Cloud server, the sync utility automatically runs to synchronize the client and the server data.

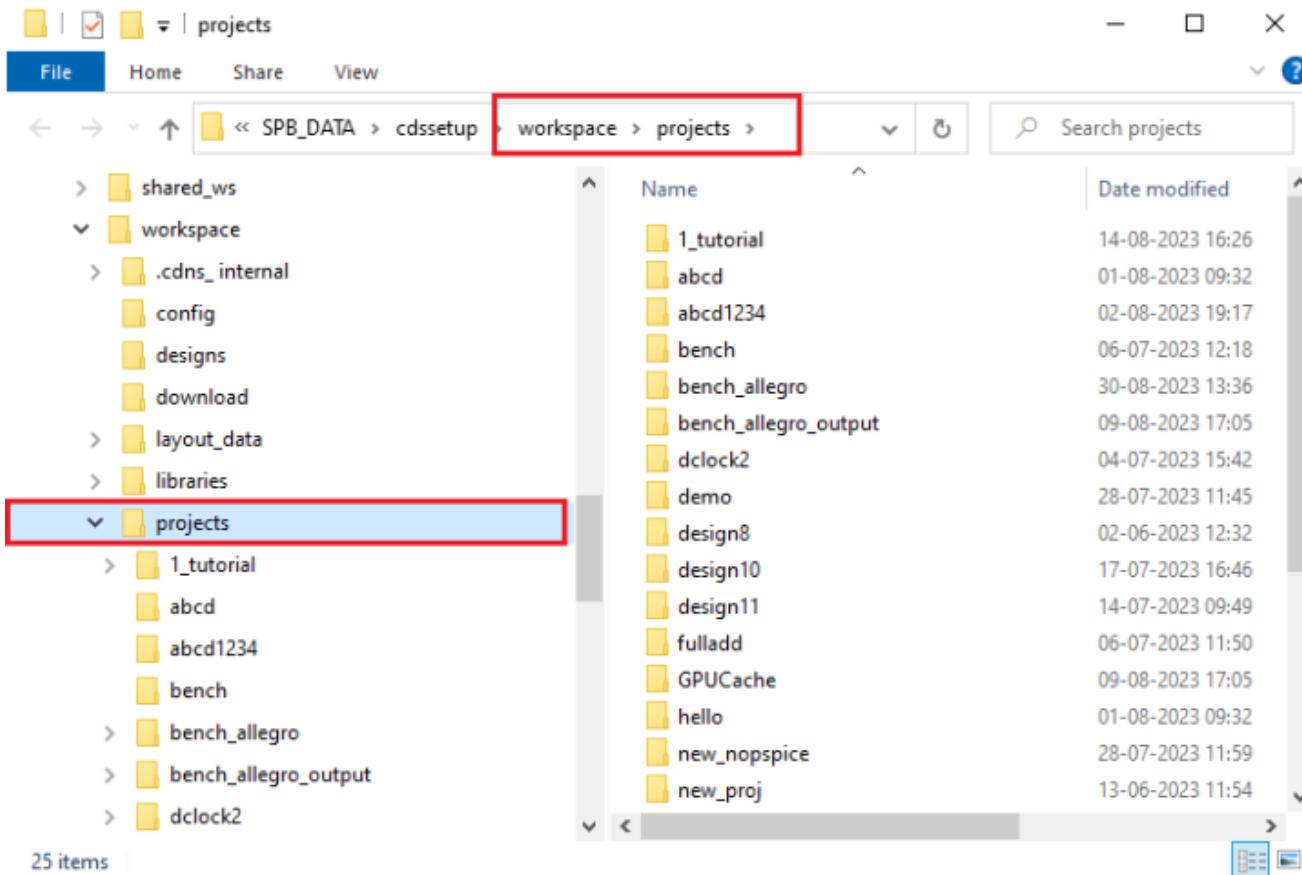
Viewing Projects in File Browser

In addition to File Manager, you can also open and view the *projects* folder and its contents in (Windows) File Browser.

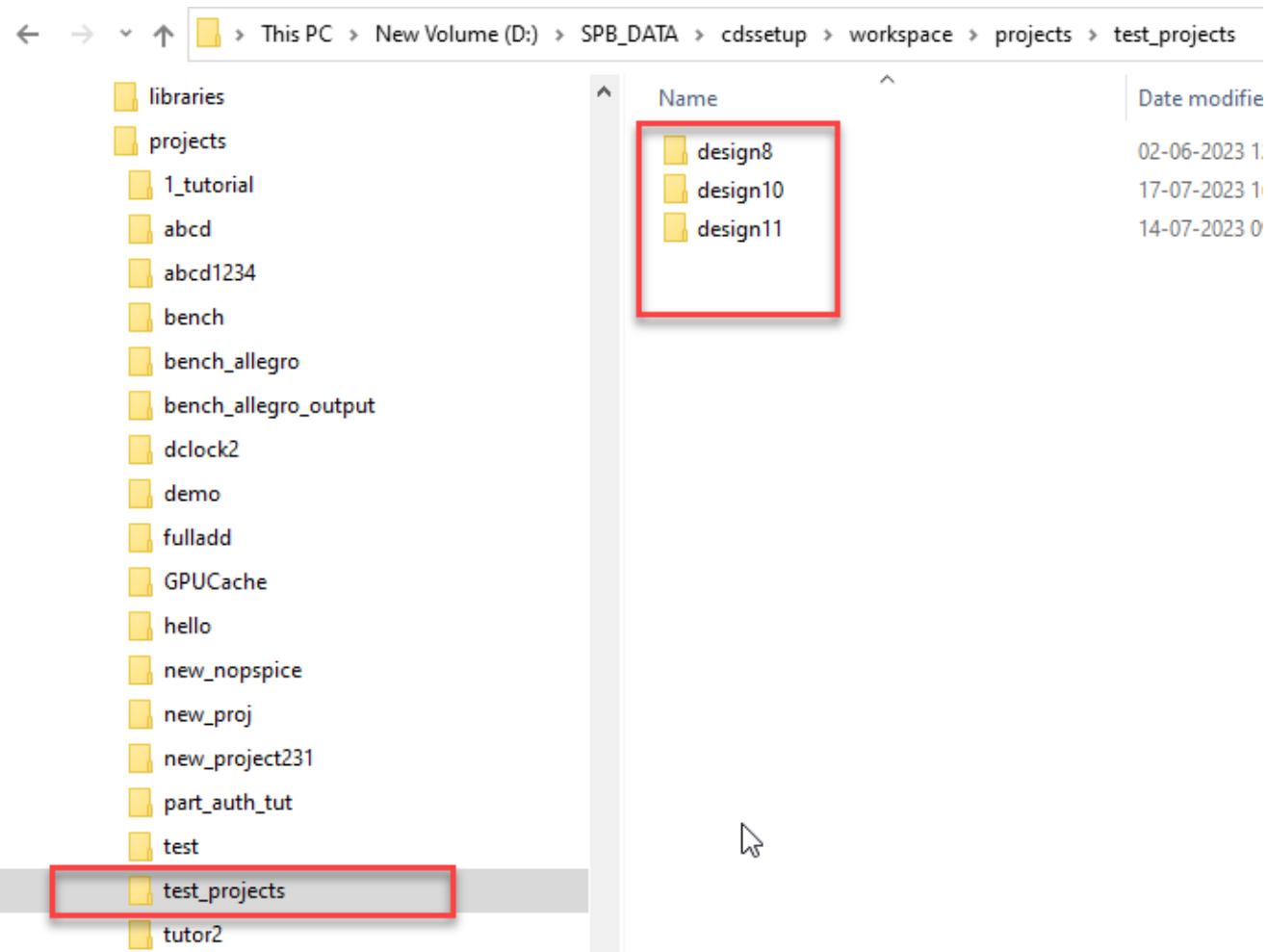
- To view a project in File Browser, right-click a project folder or a file in File Manager and choose *Show in File Browser*.



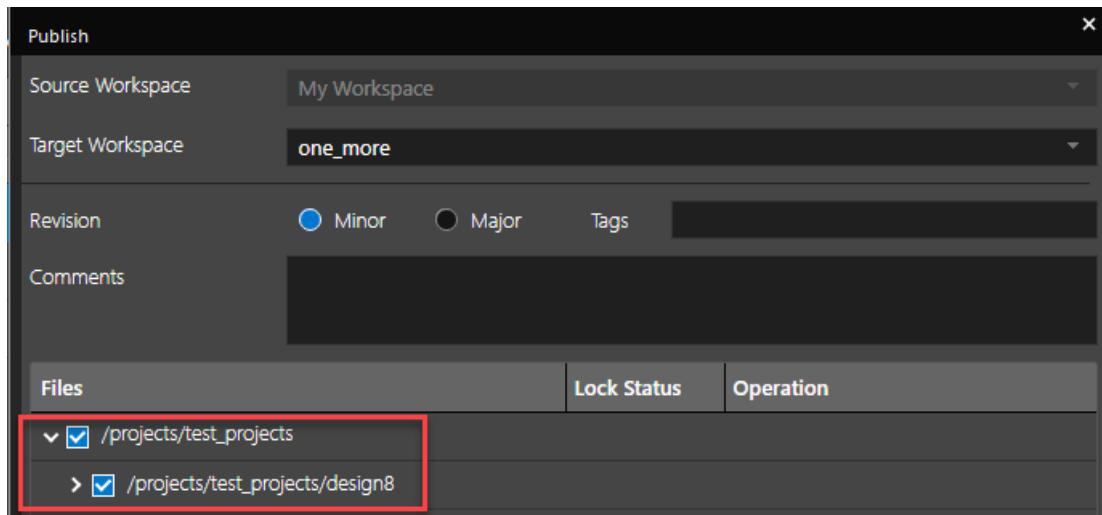
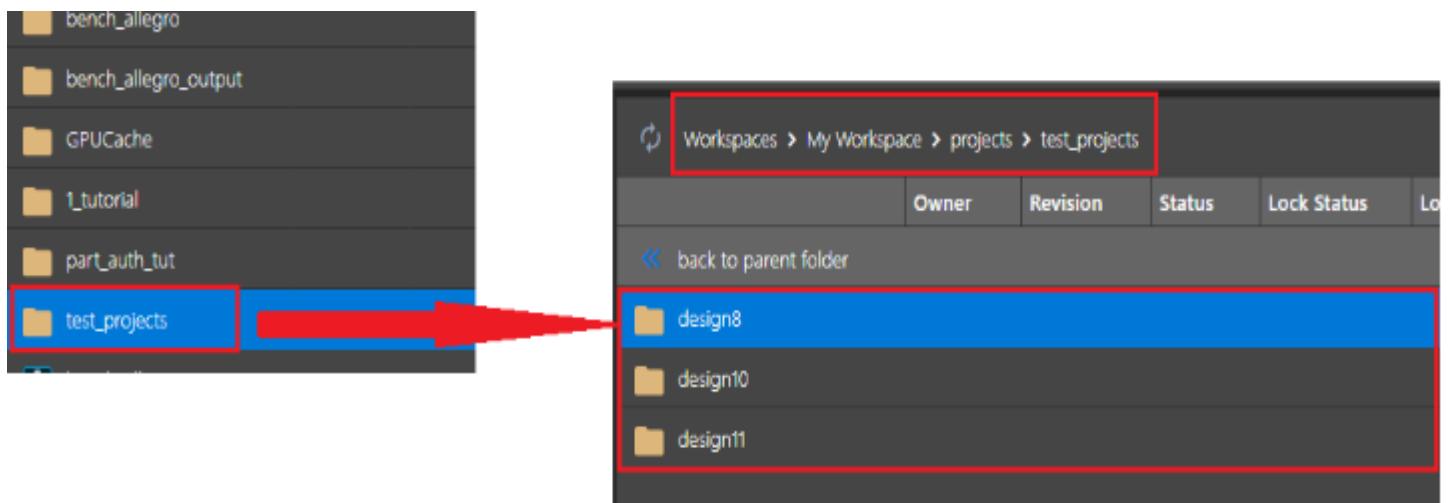
File Browser opens the location of the *projects* folder on the local disk: %HOME%\cdssetup\workspace\projects.



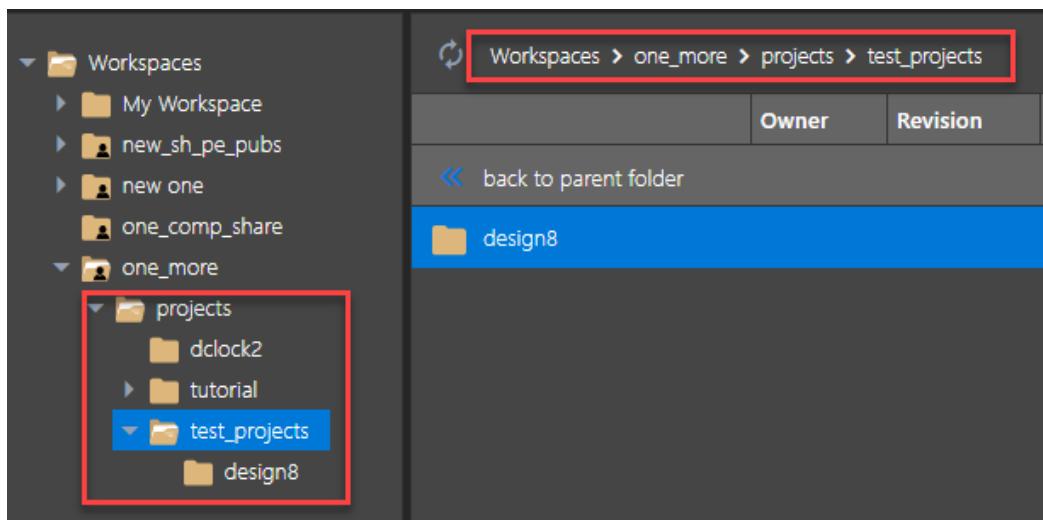
In the Cloud workspace, all the projects are created and stored in a flat structure under the *projects* folder. However, you can create a nested folder structure in File Browser by creating a new folder and moving project folders under it.



In File Manager, when you refresh the view, the nested structure is retained, and the new folder appears under the *projects* folder.



In the target (shared) workspace also the project structure is maintained.



i Important

You can publish the projects in the nested folder structure, though it is recommended to keep projects directly under the *projects* folder.

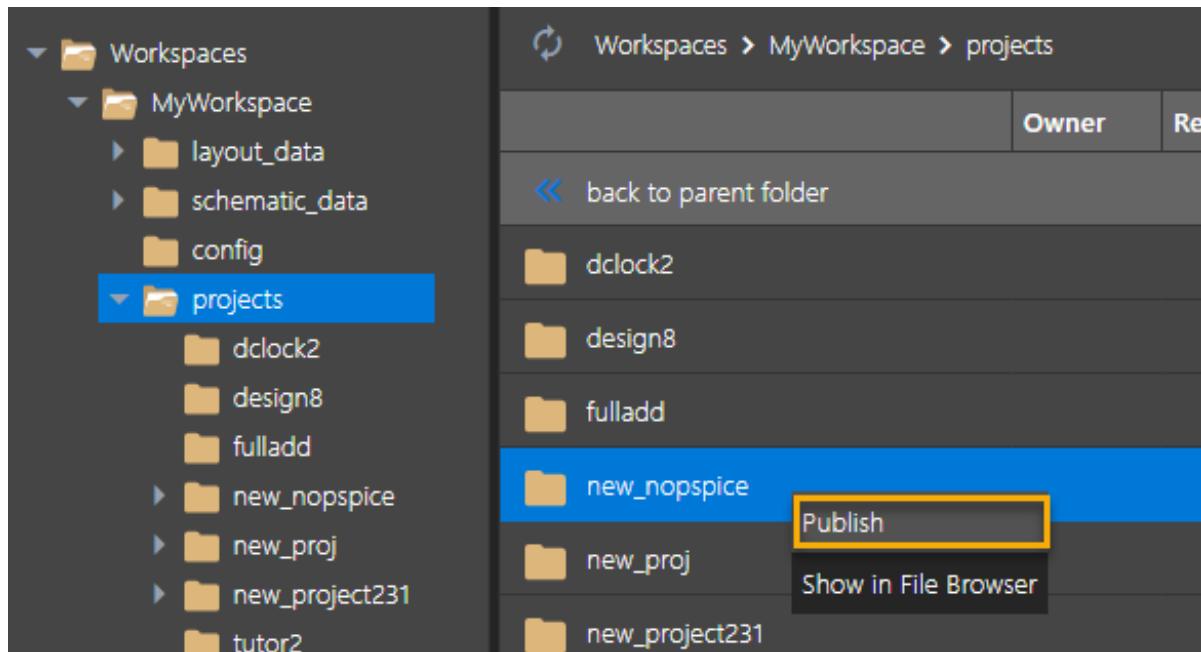
Sharing Projects

To collaborate with other team members on a project, you need to publish the project to a [shared workspace](#) with appropriate [roles](#) assigned to the team members.

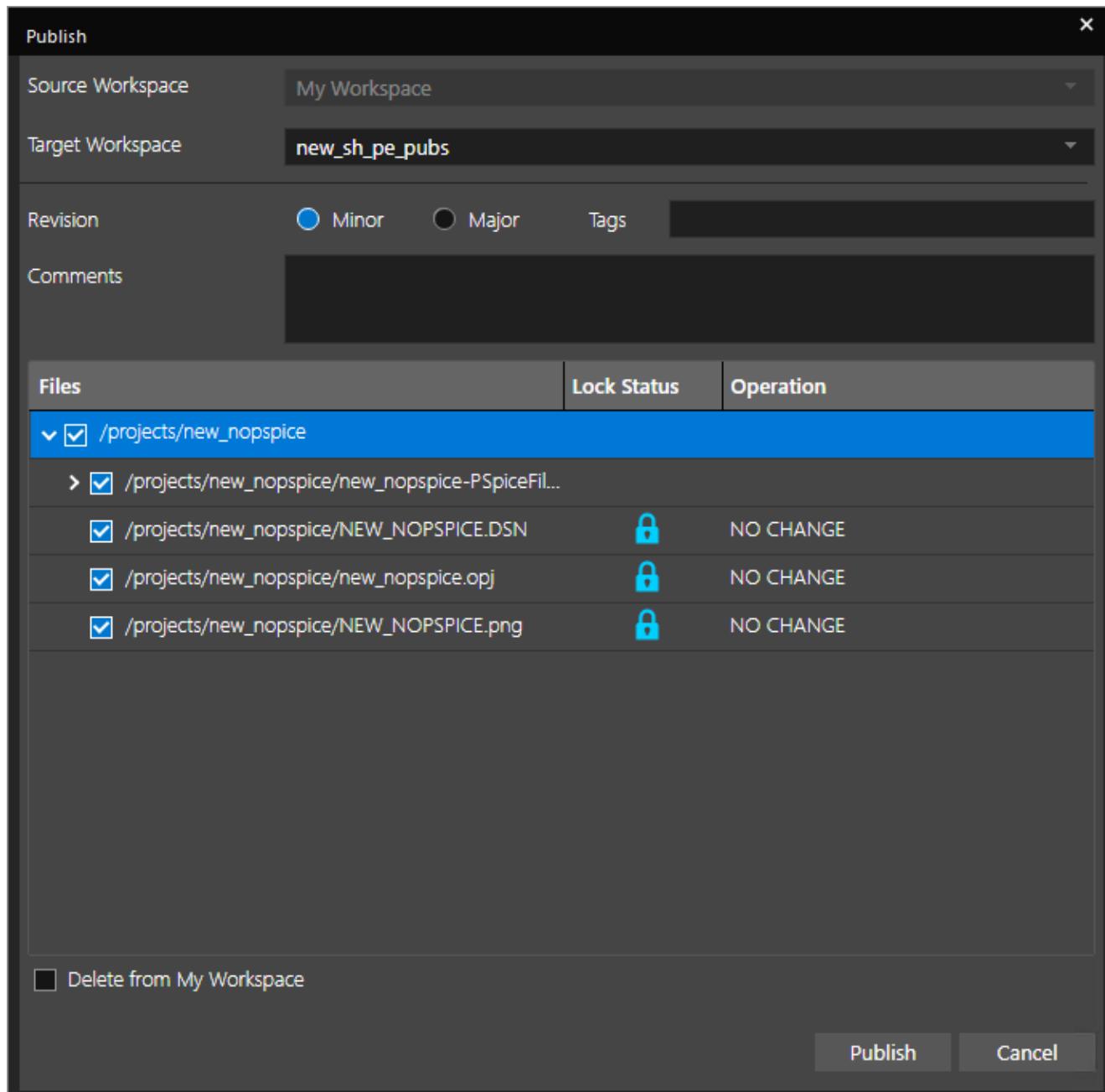
To publish a project from the local workspace to a shared workspace, do the following:

1. Create a [shared workspace](#) and assign [roles](#) to the team members who need to work on the shared files.
2. In [File Manager](#), select the project folder from *My Workspace*.

3. Right-click and choose **Publish**.



The ***Publish*** dialog box is displayed.



The Publish Dialog Box

Field	Description
-------	-------------

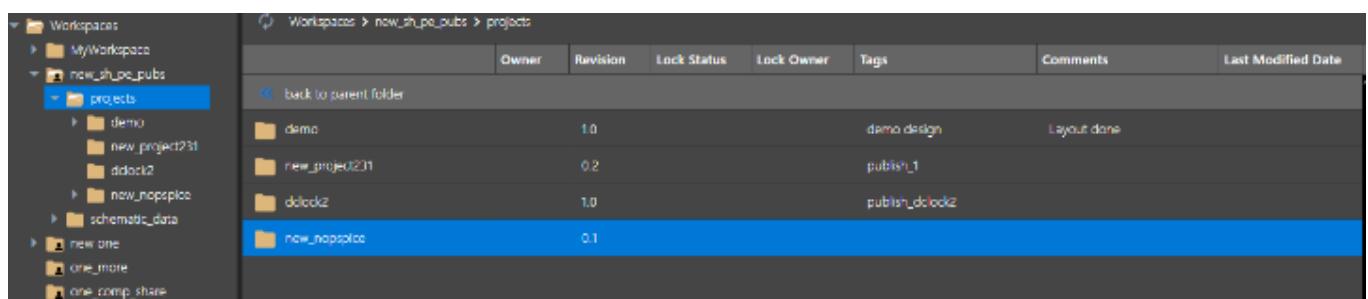
Source Workspace	The workspace containing the project or the files to be shared or published. The name of the workspace cannot be changed in the <i>Publish</i> dialog box.
Target Workspace	The workspace where the project or the files are to be published. Select a shared workspace from the drop-down list.
Revision	Select the revision, either <i>Minor</i> or <i>Major</i> depending on how you want to save the version of the file or the project. <ul style="list-style-type: none"> ◦ A <i>Major</i> version is identified by a whole number. For example, 1 . 0, 3 . 0, 5 . 0, and so on. ◦ A <i>Minor</i> version is identified by a decimal number. For example, 1 . 2, 2 . 4, 5 . 1, and so on. By default, the published project or file is saved with a <i>Minor</i> version.
Tags	Any user-defined string to tag the file or the project being published. Tags help you organize your files and filter the list of files.
Comments	Add a text note regarding the file being published.
Files	Descend into the folder structure to view the files being published with the project. By default, all the files and subfolder are selected. You can deselect a file that you do not want to share.
Lock Status	The lock status of the file.
Operation	Indicates the operation, such as addition or deletion done on the file since the last time it was published. <i>ADDED</i> indicates the file is being published for the first time. <i>NO CHANGE</i> indicates that the files are the same.
Delete from My Workspace	Removes the selected file from the local workspace after it is published to a shared workspace.

Similarly, you can publish the *libraries* folder or the selected footprints and symbol files to a shared workspace. A user with the *Librarian* role will have full access on the entire *libraries* folder.:

Files	Lock Status	Operation
✓ /libraries		
✓ /libraries/footprints		
> ✓ /libraries/footprints/downloads		
✓ /libraries/footprints/axrc05.dra	🔓	ADDED
✓ /libraries/footprints/axrc05.psm	🔓	ADDED
✓ /libraries/footprints/cap196.dra	🔓	ADDED
✓ /libraries/footprints/cap196.psm	🔓	ADDED
✓ /libraries/footprints/crys11mhz.dra	🔓	ADDED
✓ /libraries/footprints/crys11mhz.psm	🔓	ADDED
✓ /libraries/footprints/dip14_3.dra	🔓	ADDED
✓ /libraries/footprints/dip14_3.psm	🔓	ADDED

4. In the *Publish* dialog box, select the target workspace where the project or the file is to be published for sharing.
5. Select a revision, *Minor* or *Major*.
6. Add a tag or comment as required.
7. Deselect any file you do not want to share.
8. Click *Publish*.
The selected project or files are published to the target shared workspace.
9. Select the shared workspace in *File Manager*.

The published project appears in the shared workspace. Based on the assigned roles and the privileges associated with them, users can view or update the project by checking it out to the local workspace.



The screenshot shows the OrCAD File Manager interface. On the left, there is a tree view of workspaces and shared folders. A folder named "new_sh_pe_pubs" is expanded, showing its contents: "projects", "new_nosplice", "schematic_data", "new one", "one more", and "one comp share". The "projects" folder is also expanded, showing sub-folders "demo", "new_project231", "ddock2", and "new_nosplice".

On the right, there is a table titled "Workspaces > new_sh_pe_pubs > projects". The table has columns: Owner, Revision, Lock Status, Lock Owner, Tags, Comments, and Last Modified Date. There are four rows in the table:

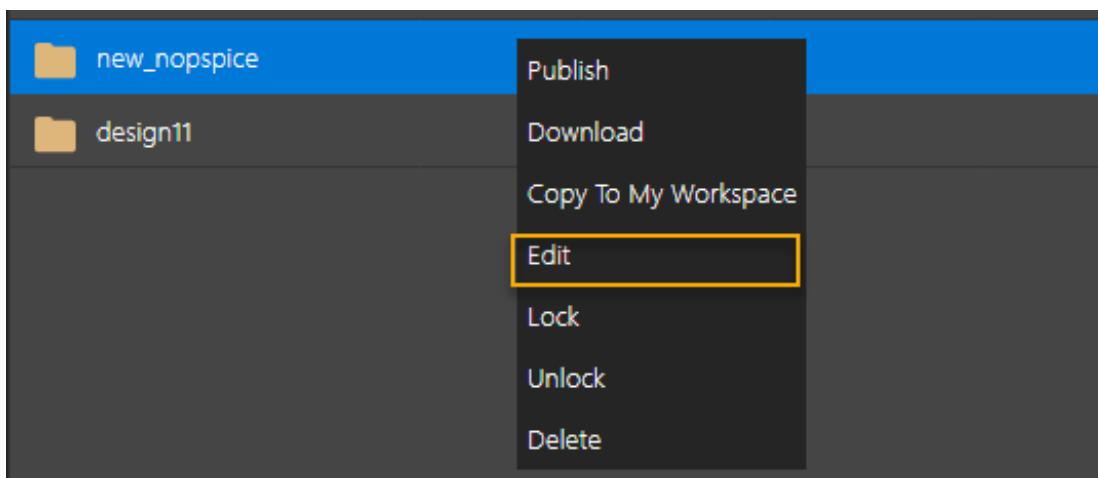
	Owner	Revision	Lock Status	Lock Owner	Tags	Comments	Last Modified Date
demo	1.0				demo design	Layout done	
new_project231	0.2				publish_1		
ddock2	1.0				publish_ddock2		
new_nosplice	0.1						

Editing Shared Projects

All the members with access to a shared workspace, can access and use the shared projects and libraries depending on the assigned [roles](#). To edit a project or a file, the users first need to move it to their local workspaces (*My Workspace*).

Also, before any changes are made to the file, it needs to be locked to prevent file updates by multiple users. To edit a project, do the following:

1. Right-click the project in the shared workspace and choose *Edit* to check it out and copy it to *My Workspace*.



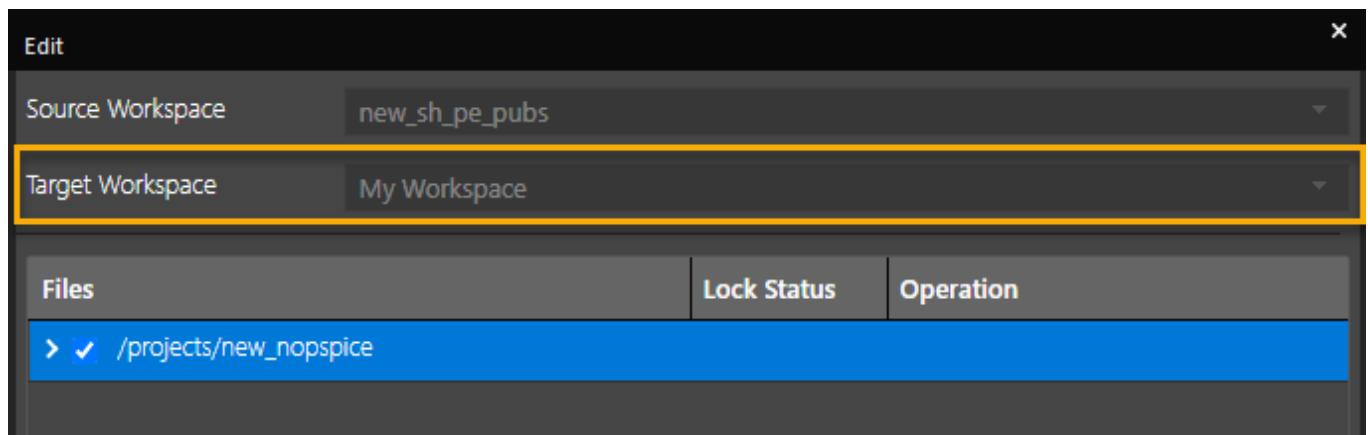
When you edit (check out) a project, the libraries folder, or a file, it is automatically [locked](#).

These commands are available on the project folder to the members with the *Designer* role and on the libraries folders to the members with the *Librarian* role. Members with the *Admin* role can access all the commands on all types of files and folders.

Members with the *Designer* role can perform operations, such as publish, edit, lock, unlock , or delete only on the files with the following extensions:

.aap.als.bom.brd.cim.cir.dat.dpf.dra.dsn.inc.lib.log.lst.mcm.mdd.mrk.net.olb.opj.out.png.prb.prp.psm.sch.sim.sip.stl.sww.top

2. In the *Publish Part* dialog box, *My Workspace* is now the target workspace.



3. Click *OK*.

The project folder is copied to the local workspace where you can edit it.

File Locking

When a file, folder, or project is checked out, it is locked by the current user (lock owner). The *Lock Status* and *Lock Owner* columns are updated to reflect the status. This file will be available for editing by other users only after the lock owner checks it back into the shared workspace by publishing it or unlocks the file explicitly using the *Unlock* command. Any user trying to edit a file locked by another user is notified of the file status.

When publishing a file from a source shared workspace to a target shared workspace, it is checked whether the file is locked in the destination workspace. The check is not done on lock status of the file in the source workspace. This is done because the files in the two shared workspaces are independent of each other. A file can be published to a shared workspaces and then copied to *My workspace* for editing. From *My Workspace*, the file can be published back into the shared workspace.

As the files in the two shared workspaces are independent of each other, they can have the same revision, but different content.

 A user with the *Admin* role can unlock a file checked out (locked) by any other user.

new_nopspice-PSpiceFiles				
new_nopspice.opj	soniap@cadence.com	0.1		7/4/2023, 10:03:43 AM
NEW_NOPSPICE.png	soniap@cadence.com	0.1		7/4/2023, 10:03:43 AM
NEW_NOPSPICE.DSN	soniap@cadence.com	0.1	soniap@cadence.com	7/4/2023, 10:03:43 AM

You can also explicitly lock a file without checking it out using the *Lock* command to prevent other users from modifying it. The lock icon acts as a visual indicator to the users that the file cannot be modified. To unlock this file, use the *Unlock* command.

Downloading a File

You can copy or download a file in *My Workspace* using the *Copy To Workspace* command. However, this command does not check out the file. Only a read-only version of the file is copied and there is no change in the lock status of the file as well.

You can also download a file or folder using the *Download* command. When downloading a file or folder from a shared workspace, a folder is created with a unique identifier for the workspace, *Workspace ID* and not the workspace name. This is because if a workspace with an identical name exists, it will be overwritten. The complete path to downloaded file or the folder is displayed in the session log.

The screenshot shows the 'Session Log' window with the following text:
Session Log
Download started for file NEW_NOPSPICE.DSN.
File NEW_NOPSPICE.DSN downloaded successfully at following path:
D:/SPB_DATA/cdssetup/OrCAD_Capture/23.1.0/downloaded_files/new_sh_pe_pubs_1687945085494/projects/new_nopspice

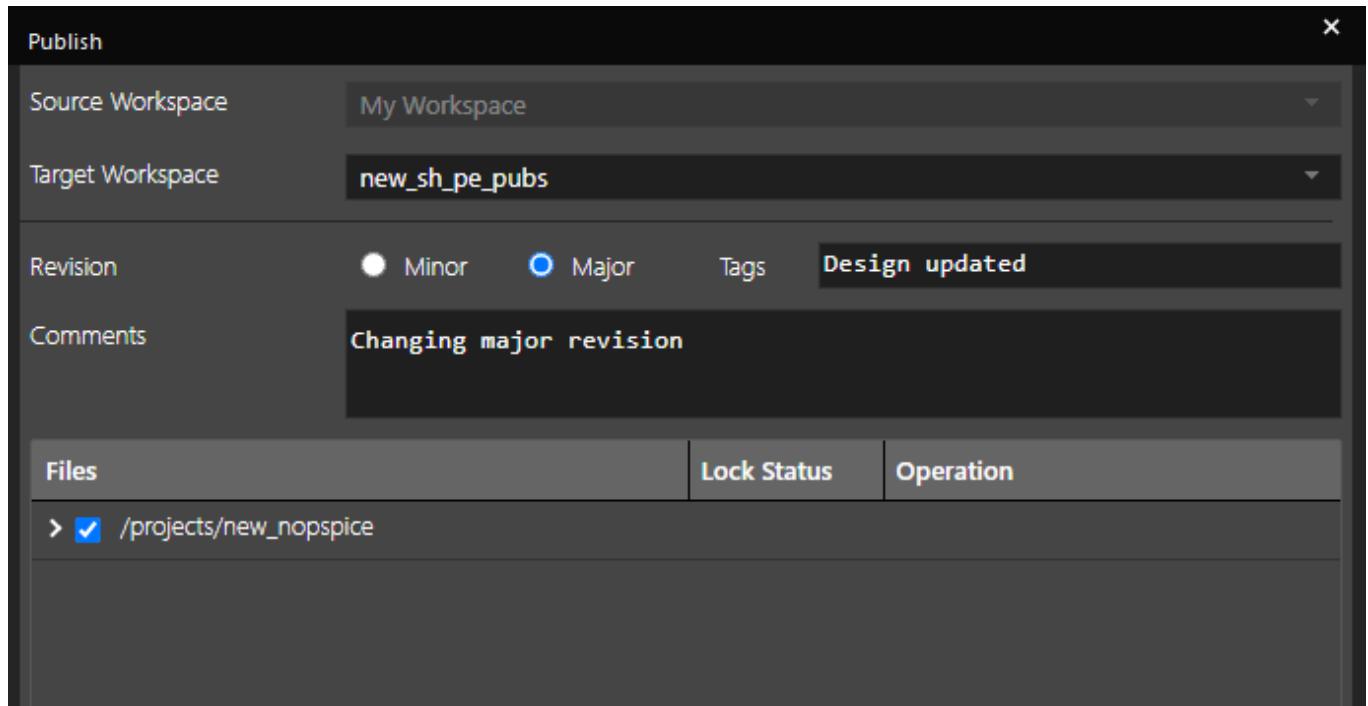
Publishing Shared Projects

After editing the shared files, you need to publish (check in) the project or the files to the shared workspace for other users to view or use them.

To publish or check in a project, do the following:

1. Right-click the file or the project in *My Workspace* and choose *Publish*.
2. Select the *Minor* or *Major* options as required and add any tags or comments as required.

3. Click *Publish*.



The file is updated in the shared workspace. The version number of the file is modified as specified and the file is unlocked.

Workspaces > new_sh_pe_pubs > projects > new_nopsice							
	Owner	Revision	Lock Status	Lock Owner	Tags	Comments	Last Modified Date
back to parent folder							
new_nopsice-PSpiceFiles							
new_nopsice.op	sonlap@cadence.com	0.1				Design updated	7/4/2023, 10:03:43 AM
NEW_NOPSPICE.png	sonlap@cadence.com	0.1				Changing major rev	7/4/2023, 10:03:43 AM
NEW_NOPSPICE.DSN	sonlap@cadence.com	1.0					7/4/2023, 10:03:43 AM

Restoring a Revision

The *Information* tab in the right pane reflects the updated information about a file after it is published to the shared workspace. The information includes file details, such as the name, revision, creator, and creation time of the file.

It also includes information, such as when was the file last modified and by whom.

Information		History
Type	DSN	
Size	108.0 KB	
① Extra info		
Name	DEMO.DSN	
Revision	3.0	
Creator	nitint@cadence.com	
Create Time	6/28/2023, 3:10:47 PM	
Modified Time	7/4/2023, 3:29:59 PM	
Modified By	soniap@cadence.com	
Tag	major version update	

The *History* tab lists all the earlier versions of the file. Each time this file is modified and checked in, a new row is added for the previous version.

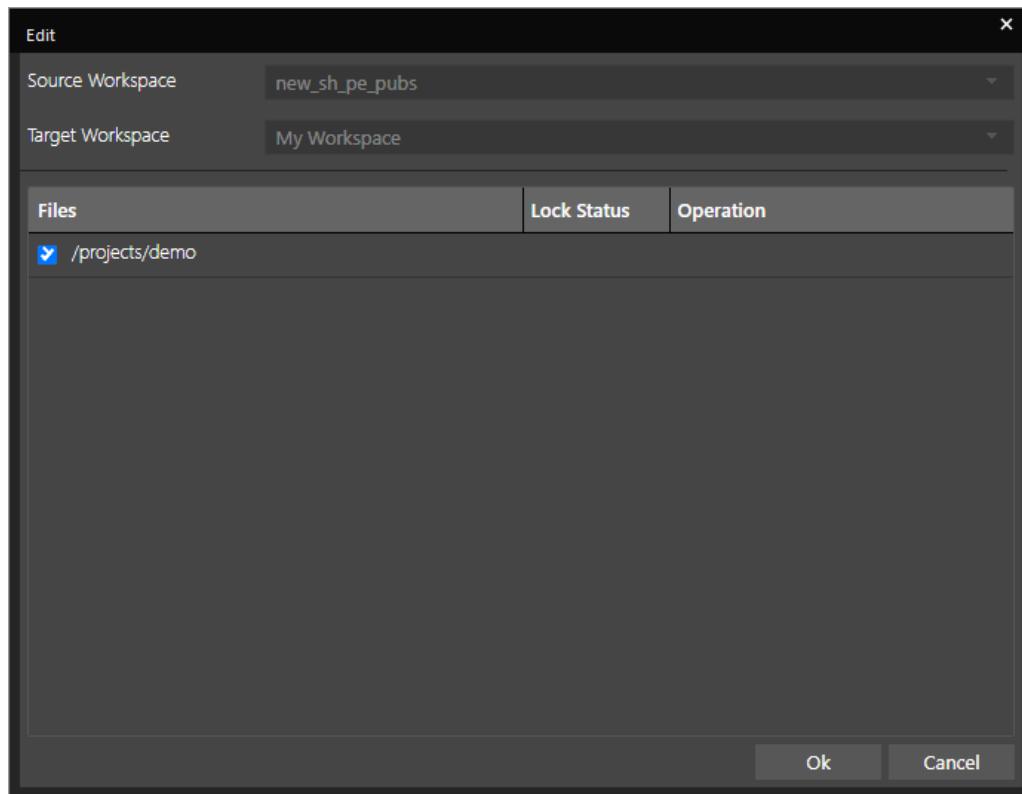
Information	History
Revision	Modified Date
0.1	7/9/2023, 11:37:49 AM
1.0	7/10/2023, 5:57:17 PM
1.1	7/10/2023, 5:58:51 PM
2.0	7/28/2023, 10:31:08 AM

To roll back to a previous version (restore) of the file, do the following:

1. Right-click the version of the file to be restored.
2. Choose the *Edit* command from the shortcut menu.

Information	History
Revision	Modified Date
0.1	7/9/2023, 11:37:49 AM
1.0	7/10/2023, 5:57:17 PM
1.1	7/10/2023, 5:58:51 PM
2.0	7/28/2023, 10:31:08 AM

The *Edit* dialog box opens, displaying the names of the target and source workspaces.



3. Click *OK*.

The selected version of the file is restored in the workspace.

- You can also copy a read-only version of the file to the local workspace. Any existing file with an identical name in the workspace is overwritten.
- You can also delete a version of the file from the shared workspace.

Related Topics

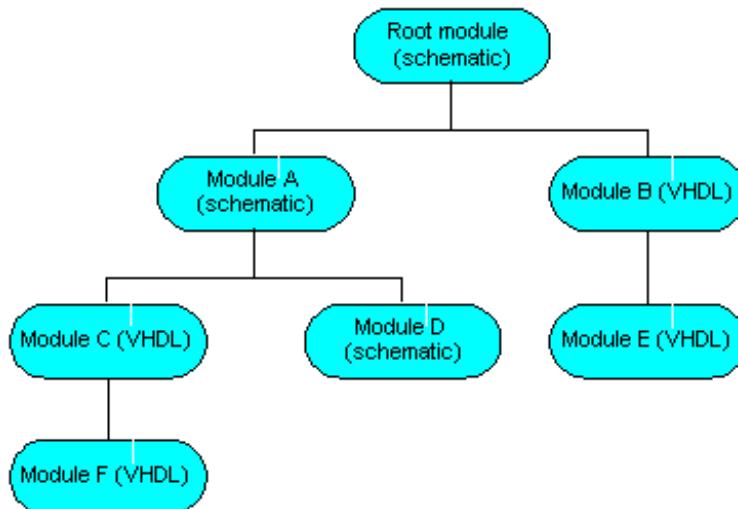
[Sharing Components](#)

Working with Designs

You create schematic designs in Capture. There can be only one design file in a project. If do not specify a root for your design, no reports are generated for the design. Also, when folders are copied to a new design, the ROOT designation is lost and must be reestablished in the design.

Schematic designs can include either VHDL or Verilog models, but not both, as lower-level hierarchical modules. However, 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 or Verilog models as instantiated components. However, VHDL or Verilog design modules are limited to other modules of the same type as instantiated components. Therefore, if the root module of a design is a VHDL model, all the lower-level modules must also be VHDL models.

Creating a Design

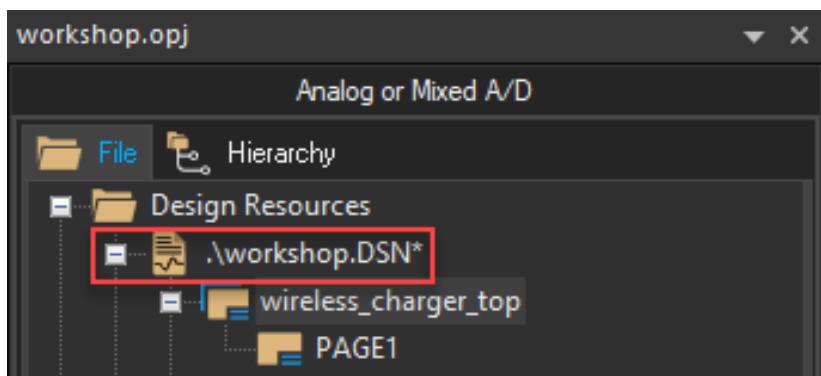
When you create a project in Capture, a design with the same name as the project is immediately created for you. However, you also have the option of creating a design without first creating the project.



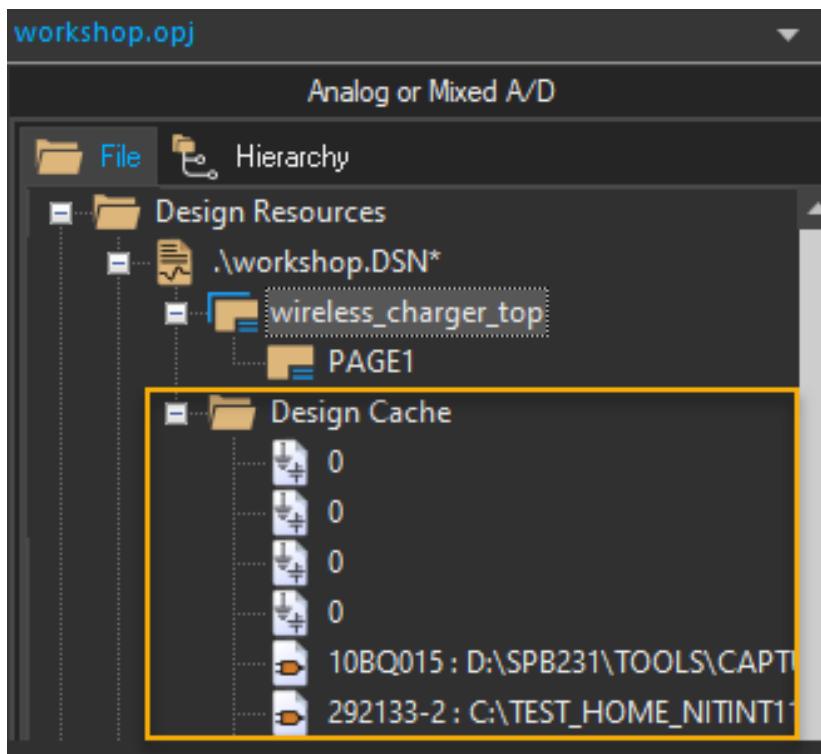
- A new design inherits characteristics from the settings in the *Design Template* dialog box, you need to check those settings before you create a design.
- There can be only one design file in a project. If you create a new design file, or move or copy a different one into the project, the project manager prompts you to replace the existing design file.

Creating a Design During Project Creation

When a project is first created, a design file is added in the project manager with the same name as the project. A schematic folder within the design file is also created with 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. You can view the hierarchy in the *File* tab of the project manager.



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 in the schematic pages.

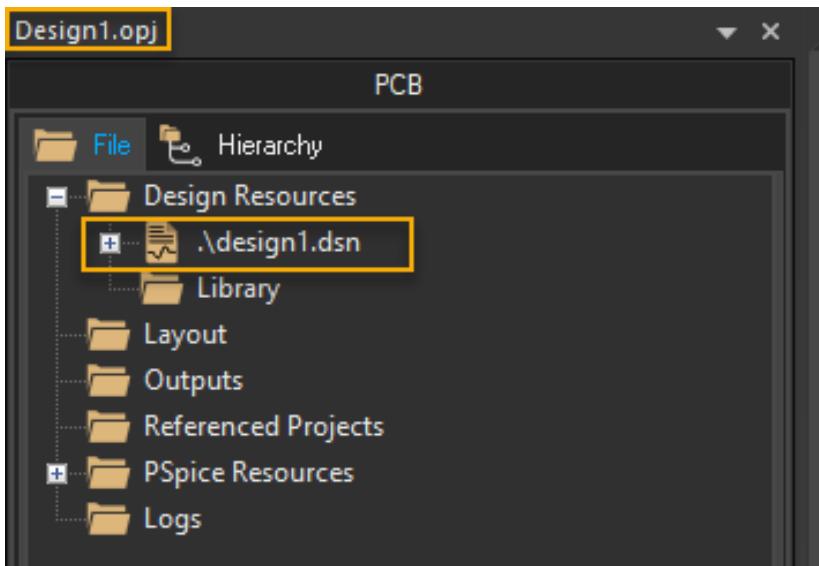


Creating a Design Outside a Project

You can also create a design without first creating a project.

- To create a design, choose the *File – New – Design* menu command from the main menu.

The project manager opens a new panel for the new project structure. By default, the new project is named `Design<n>.opj` and a new design with the name `Design<n>.dsn` is created. The design is created in the `Windows\temp` directory. Capture checks if a file with an identical name exists in the `temp` directory. If it exists, the number, `<n>`, in the design name is incremented by one. A design created this way cannot be renamed.



Related Topics

- [Creating a Project](#)
- [Renaming a Design](#)

Opening a Design

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

To open an existing design, do the following:

1. Choose the *File – Open* menu 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 the project manager window.

Shortcut



Toolbar:

To open a recently used project, choose File menu, choose the project either by name or by number.

The project opens in a project manager window.

Important

The *Select Project Type* dialog box opens if you open a design in Capture that does not have an associated project. You then select the simulation option and Capture creates the corresponding project file (.opj).

Creating a new VHDL or Verilog file

You can create VHDL or Verilog files as part of the functionality description of a design, 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, do the following:

Use one of the following ways to create a new VHDL or Verilog file in Capture:

- Choose *File – New – VHDL File* or *File – New – Verilog File* from the main menu as required.

A file of the appropriate type opens as a new tab.

OR

1. With the project manager active, choose *Design – New VHDL File* or *Design – New Verilog File*.

The file opens in the appropriate editor and a dialog box appears, asking if you want to add

the file to the project.

2. Select the Yes button to add the file to the current project.

The *Save As* dialog box appears.

If you select the *No* button, the file is not added to the project. You need to save the project later.

3. Select a directory for the file and supply a filename.

By default, a VHDL file is named `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. Click the *Save* button.

The file is saved and appears in the *Design Resources* folder in the project manager.

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.

Using Non-Electrical Objects

Text and graphics in schematics are considered as non-electrical components used for documentation purposes. You can use the text and drawing objects provided in Capture to provide additional information on the schematic and create graphical objects on the schematic, respectively. These objects provide a way for you to document your schematic without affecting its connectivity. They do not have any effect on the netlist generated from the schematic.

Topics covered in this section:

- [Creating Graphics](#)
- [Drawing Arcs](#)
- [Drawing Bezier Curves](#)
- [Drawing Ellipses and Circles](#)

- Drawing Elliptical Arcs
- Drawing Lines
- Drawing Polylines
- Drawing Rectangles and Squares
- Placing OLE Objects
- Placing Pictures
- Placing Text
- Placing IEEE Symbols

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 the *Line Style* and *Line 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 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

In schematic page editor:

1. Select the required 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*.

In part editor:

1. Select the required object.
2. Use the *Basic Attributes* section in the *Property Sheet* pane to modify the line style or fill style.

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 an 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 create a full circle, use the ellipse tool.

To draw an arc

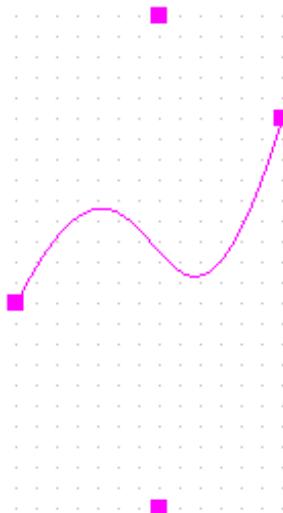
1. From the *Place* menu, choose the [Arc command](#).
2. Move the pointer to the location where you want to start the arc.
The arc is drawn counterclockwise from this start point.
3. Drag the mouse to define the shape of the arc.
4. Release the left mouse button to mark the end of the arc.
The arc appears in the selection color.
5. Choose the selection tool or press `ESC` to dismiss the arc tool.

Shortcut

Tool palette: 

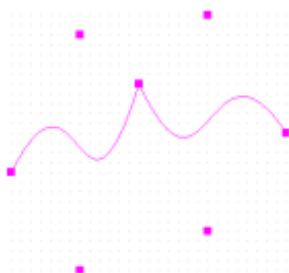
Drawing Bezier Curves

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

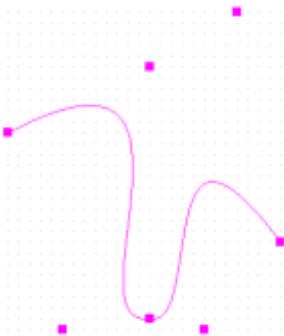
1. From the *Place* menu, choose the `Bezier` command.
2. Click the canvas to mark the start point of the curve.
3. Click to mark the first control point.
4. Click again to mark the second control point.
5. Click to mark the end point of the curve.;
6. Select the selection tool or press `Esc` to dismiss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

Shortcuts

- Tool palette: 
- Keyboard: SHIFT+Q

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](#).

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.

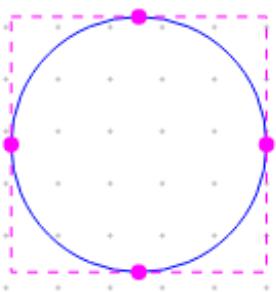
4. Choose the selection tool or press Esc to dismiss the ellipse tool.
5. Click an area where there are no parts or objects to deselect the ellipse.
Or
 1. From the Place menu, choose the [Ellipse command](#).
 2. Move the pointer to an edge of the intended ellipse and click the left mouse button.
 3. 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 an area where there are no parts or objects to deselect the ellipse.

Shortcuts

- Tool palette: 
- Keyboard: SHIFT+F

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 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 location where you want to start the arc.
The arc is drawn counterclockwise from this start point.
3. Drag the mouse to define the shape of the arc.
4. Release 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.

Shortcuts



- Tool palette: 
- Keyboard: [SHIFT+T](#)

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 want to draw a line with multiple contiguous segments, use the [polyline tool](#).

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.

Shortcuts



- Tool palette:
- Keyboard: `SHIFT+L`

Drawing Polylines

When you want to draw a line with multiple contiguous segments, use the polyline tool. 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 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

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

Shortcuts



- Tool palette:
- Keyboard: `Y`

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 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 an area where there are no parts or objects to deselect the rectangle.

Shortcuts



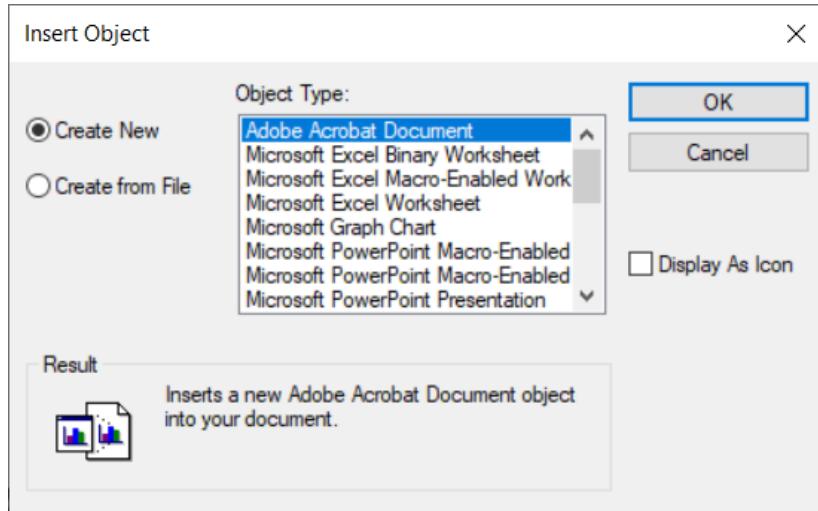
- Tool palette:
- Keyboard: `SHIFT+R`

Placing OLE Objects

You can place objects of other applications on your schematic page using the OLE Object command. This allows you to 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 *OLEObject*.
The Insert Object dialog box opens.



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 the 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 *OLEObject*.

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 an image

1. From the Place menu, choose the [Picture command](#).
The [Open dialog box](#) appears.
2. Select the image file. If the required file is not listed:
 - In the *Look in* text box, select a new drive, directory, or both.
 - In the *Files of type* text box, select the type of file you want to open.
3. Click *Open*.
4. Create a rectangle to place the image and release the left mouse button.

 Capture supports many different image file formats, including the following: **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.

Placing Text in Schematic Page Editor

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.
3. Use the following options to modify 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 group box to specify the orientation of the text on the page.
 - c. Use the Change button in the Font group box to change the font of the text.
 - d. Use the Text Justification drop-down list to justify text as Default, Right, Center, or Left.
4. Click *OK*.
The text is immediately attached to the cursor.
5. Click the schematic page where you want to place the text.
6. Select the selection tool or press *ESC* to complete placing the text.

Netlist Comments

When you start a comment with @PSpice:, it is netlisted to PSpice during the netlist creation process. This is done only when such a comment is placed in the root schematic. The concatenation of comments with @PSpice:, from subcircuits is not done because that can create invalid netlists.

Examples of single-line comment and multi-line comment starting with @PSpice: are:

- **Single-line comment**

```
@PSpice: R1 1 0 1k will be netlisted as R1 1 0 1k
```

- **Multi-line comment**

```
@PSpice: .autoconverge ITL1=1000 ITL2=1000 ITL4=1000 RELTOL=0.05 ABSTOL=1.0E-6  
VNTOL=.001 PIVTOL=1.0E-10 .TEMP 125  
will be netlisted as  
.autoconverge ITL1=1000 ITL2=1000 ITL4=1000 RELTOL=0.05 ABSTOL=1.0E-6 VNTOL=.001  
PIVTOL=1.0E-10 .TEMP 125
```

Using Options or Flags not in Simulations Settings Window

To use options or flags that are not available in the Simulations Settings dialog box:

1. Open the design in Capture.
2. Choose *Place – Text*.
The Edit Text window appears.
3. Specify the option using the following syntax:

@PSPICE:

```
.options <option_name>=<value>
```

OR

@PSPICE:

```
.options <option_name> <value>
```

4. Specify the flag using the following syntax:

```
.options <flag_name>
```

 This signifies that the options or flags that follow @PSPICE: are considered PSpice directives.

5. Click *OK*.
6. Run the simulation.

The simulation will run using the options or flags set and you can view them in the .out file generated after the simulation.

Placing Text in Part Editor

To place text in the part editor:

1. From the *Place* menu, choose the *Text* command.
The Edit Comment Text dialog box appears.
2. Specify the text to place on the part page.
3. Click *OK*.
The text is immediately attached to the cursor.
4. Click the page where you want to place the comment text.
5. Select the selection tool or press *ESC* to complete placing the text.

Shortcuts

- Tool palette: 
- Keyboard: T

In-place Editing of Text Objects

You can edit text objects, such as pin name, pin number, and property name, in the part editor canvas. Double-click any text object and modify the values in the input box that appears.

Shortcuts

Shortcut Key	Function
Esc	Use this key to exit the in-place editing mode without saving changes.
Enter	Use this key to exit in-place editing mode with saving the changes.
CTRL+Enter	Use this key combination to provide a newline entry.

Placing IEEE Symbols

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

To place an IEEE symbol in the part editor

1. From the Place menu, choose the *IEEE Symbol* command. The Place IEEE Symbol dialog box appears.
2. From the *Select IEEE Symbols* list, select a symbol. The symbol appears in the preview box.
3. Select the required symbol and 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.

Shortcuts

- Tool palette: 
- Keyboard: SHIFT+X

Flat and Hierarchical Designs

In this section:

- [Flat designs](#)
- [Hierarchical Designs](#)
- [Searching in Capture](#)

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 through 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

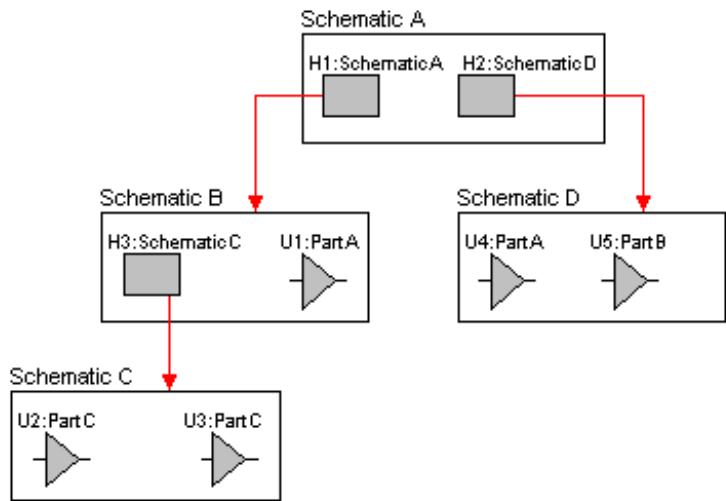
You can manage both schematic and VHDL design resources in a hierarchical manner. That is, you can create schematic pages containing hierarchical blocks or parts with schematic or VHDL implementations. The hierarchical block symbol (or part with an attached schematic page or model) in the schematic page 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

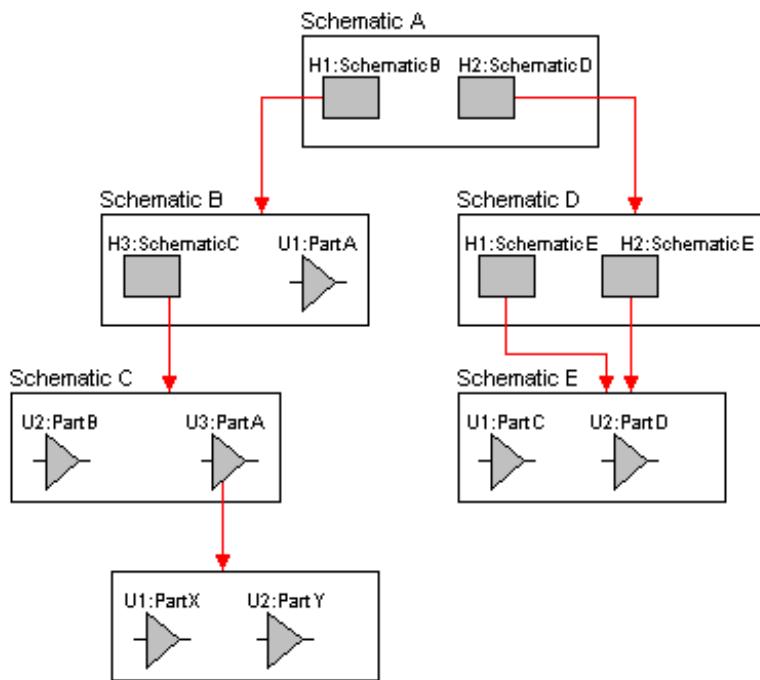
A one-to-one correspondence between hierarchical blocks or parts and the schematic, EDIF or VHDL implementations they reference is called a simple hierarchy. The picture below is an example of a simple hierarchy typical of most PCB designs in Capture.



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](#)
- [Browsing in Capture](#)

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).

Browsing in Capture

Using the [project manager](#), you can list objects and also select and edit them. 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
- Nets
- Flat netlist (nets as they appear in a netlist)
- Hierarchical ports

- Off-page connectors
- Title blocks
- Bookmarks
- DRC Markers
- Power Pins
- 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 pane, see the *Searching for variant information on a schematic page* section of the *OrCAD X 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

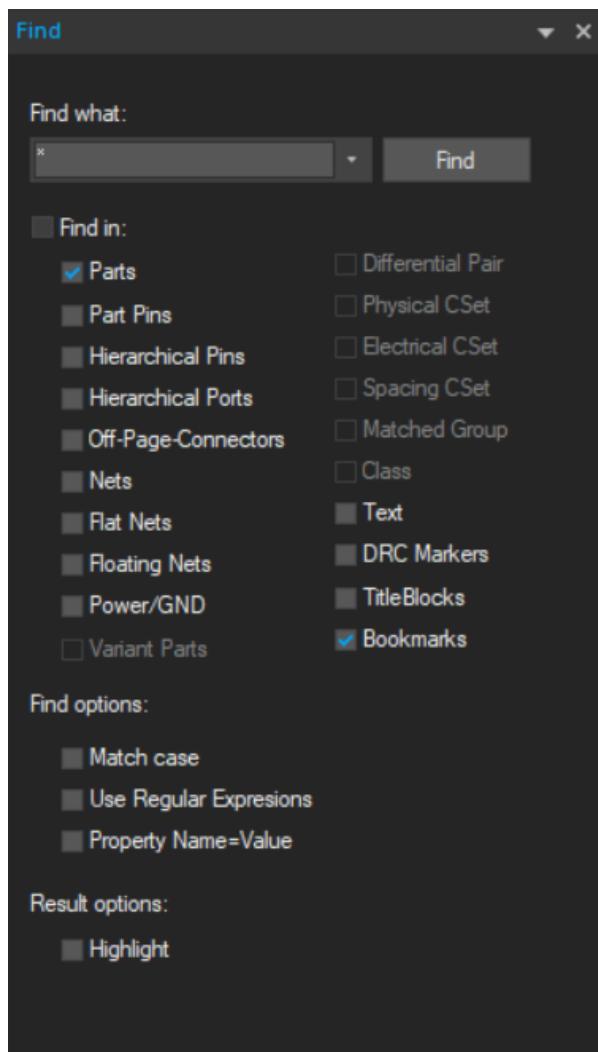
- In the project manager, double-click the Design Cache icon.

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 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 objects you want to search. To search the entire design, select all schematic folders.
2. From the Edit menu, choose the Find command, or press [`CTRL+F`](#).
The Find pane opens.



3. In the 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 Find pane contains search options that allow you to further refine your search.

4. From *Find options*, verify that the *Match case* option is set as required.
5. From the search options drop-down list select the object type.
6. Press Enter or click the *Find* button.

The Find Results window displays a list of objects that meet the criteria you specified.

7. If you want to display an object, double-click the entry in the Find Results window. The schematic page editor opens and the object appears in the selection color.

OR

If you want 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 want to open the Browse Spreadsheet for a part in the Find Results window, right-click the part and choose *Edit Properties*. The Browse Spreadsheet dialog box opens.

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.

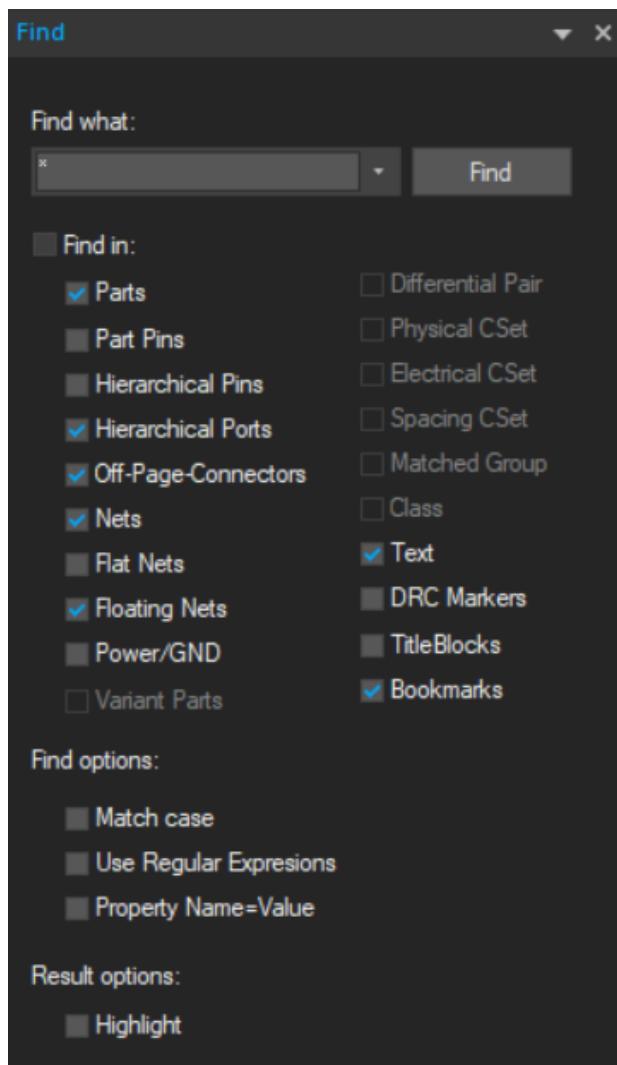
Locating an Object in a Project

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 pane, 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, do the following:

1. In the project manager, select the schematic folders or schematic pages you want to search.
2. Choose *Edit – Find*, click the search icon () , or press **CTRL+F**.

The Find pane appears.



3. In the search text box, enter the property value string for the part you search. You can use wildcard characters (standard "*" or "?") with a truncated search. For example, to search for resistors, enter "R*".

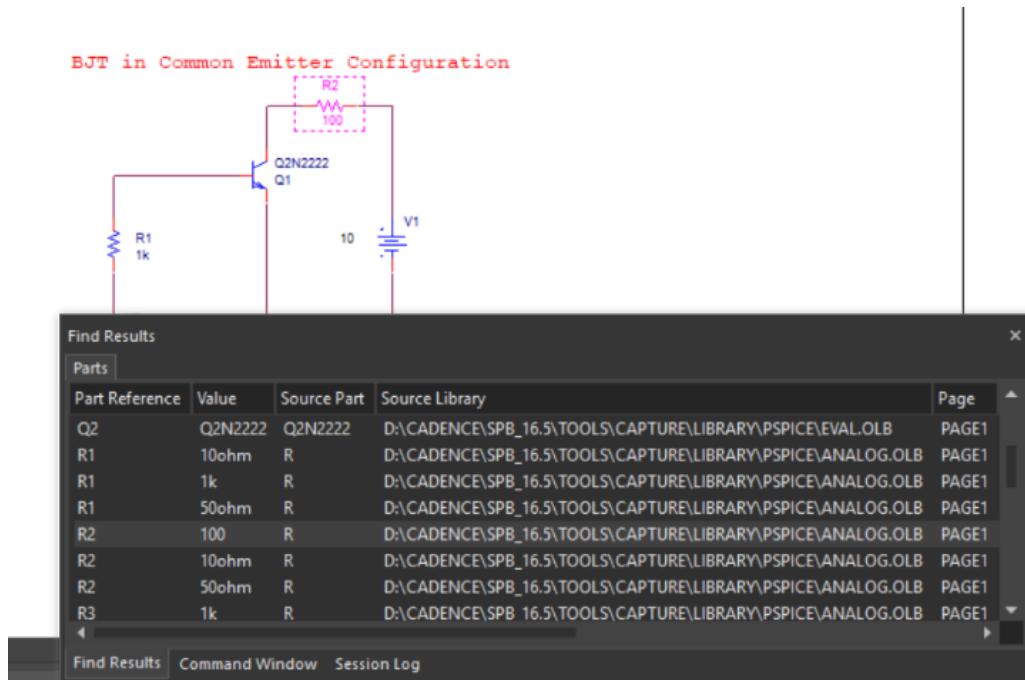
You can also search for a part by property or use regular expressions in your search string.

⚠ 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 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 *Find* button to start the search.
- Object that have a property value matching the property value string in step 3 are displayed in the Find Results window.
6. Double-click the part in the Find Results window list to open the schematic page editor with that part displayed and selected.



- i**
- To search for all parts with references containing R or C followed by any number between 2 and 9, use the search string *Part Reference=(C/R)[2-9]* with both Property Name=Value and Regular Expressions option selected.
 - To search for whole word of parts containing R or C followed by any number between 2 and 9, use search string *\m(C/R)[2-9]\M* or *^(C/R)[2-9]\$* with Regular Expressions option 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 a design and choose *Find*.
2. In the *Find what* text box, type the search string and press `Enter`.

The search results displayed in the Find Results window include all objects found within the entire design.

Folder Level

1. In the project manager, right-click a folder and choose *Find*.
2. In the *Find what* text box, type the search string and press `Enter`.

The search results displayed in the Find Results window include all objects found within the selected folder.

Page Level

1. In the project manager, right-click a schematic page and choose *Find*.

OR

Open the schematic page to search and choose *Find* from the *Edit* menu.

2. In the *Find what* text box, type the search string and press `Enter`.

The search results displayed in the Find Results window include all objects found on the selected page.

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 the selection and choose *Find*.

3. In the *Find what* text box, type the search string and press `Enter`.

The search results displayed in the Find Results window include all objects found within the selected items in the design hierarchy.

Find Pane

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

Find what 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 button	Run the search command.
Find in	This is a multiple selection 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 Results Window

After the search is complete and if it returned at least one result, the result is displayed in the Find Resultst window. This is a tabbed dockable window.

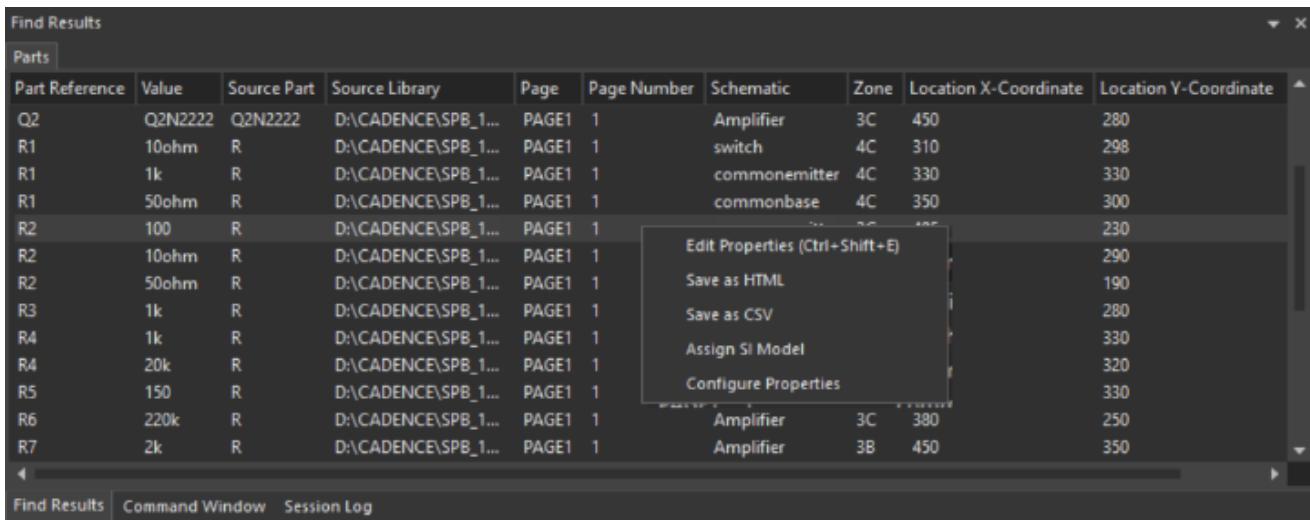
Each result of the search will display as one line item in the window. A result line item contains other information besides the search object reference. This includes the page and schematic and properties specific to object types.

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 the title bar. In the docked mode, use the down-arrow button to change the window to *Floating* or *Tabbed*, or to hide it.

The pop-up menu on the Find Results window contains options as shown in the following figure:



The screenshot shows the 'Find Results' window in OrCAD. The 'Parts' tab is selected. A context menu is open over a row for a resistor R2 with a value of 100 ohms. The menu options are: Edit Properties (Ctrl+Shift+E), Save as HTML, Save as CSV, Assign SI Model, and Configure Properties.

Part Reference	Value	Source Part	Source Library	Page	Page Number	Schematic	Zone	Location X-Coordinate	Location Y-Coordinate
Q2	Q2N2222	Q2N2222	D:\CADENCE\SPB_1...	PAGE1	1	Amplifier	3C	450	280
R1	10ohm	R	D:\CADENCE\SPB_1...	PAGE1	1	switch	4C	310	298
R1	1k	R	D:\CADENCE\SPB_1...	PAGE1	1	commonemitter	4C	330	330
R1	50ohm	R	D:\CADENCE\SPB_1...	PAGE1	1	commonbase	4C	350	300
R2	100	R	D:\CADENCE\SPB_1...	PAGE1	1				230
R2	10ohm	R	D:\CADENCE\SPB_1...	PAGE1	1				290
R2	50ohm	R	D:\CADENCE\SPB_1...	PAGE1	1				190
R3	1k	R	D:\CADENCE\SPB_1...	PAGE1	1				280
R4	1k	R	D:\CADENCE\SPB_1...	PAGE1	1				330
R4	20k	R	D:\CADENCE\SPB_1...	PAGE1	1				320
R5	150	R	D:\CADENCE\SPB_1...	PAGE1	1				330
R6	220k	R	D:\CADENCE\SPB_1...	PAGE1	1	Amplifier	3C	380	250
R7	2k	R	D:\CADENCE\SPB_1...	PAGE1	1	Amplifier	3B	450	350

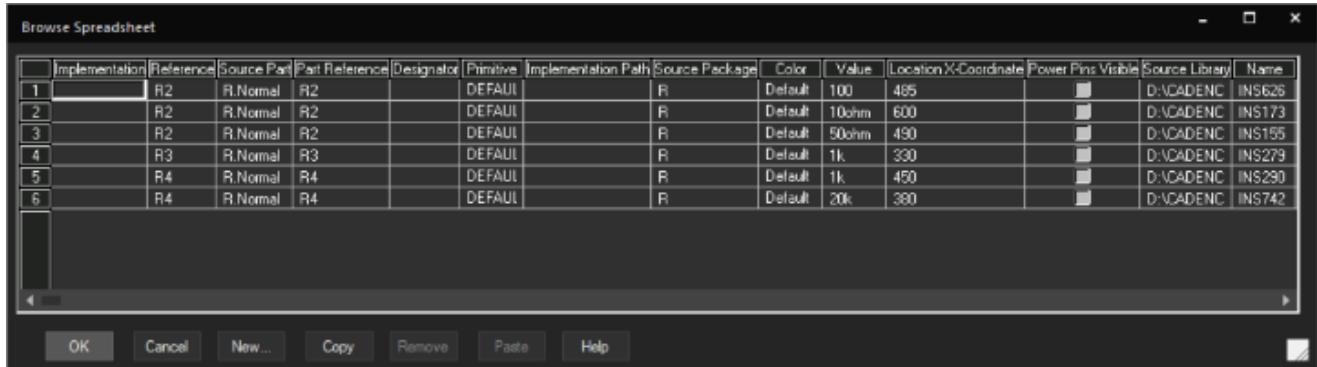
Edit Properties

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

To do so:

1. Right-click one or more results line items.
2. Choose *Edit Properties*.

The Browse Spreadsheet window displays the editable part properties.



The screenshot shows the 'Browse Spreadsheet' window. It displays a table of parts with various properties like Implementation Path, Source Package, Color, Value, Location, and Power Pins Visible. The 'Implementation Path' column contains formulas such as =R2, =R3, etc. The 'Source Package' column contains formulas like =R, =Default, etc. The 'Color' column contains values like Default, 100, 50ohm, 1k, etc. The 'Value' column contains values like 100, 10ohm, 50ohm, 1k, etc. The 'Location X-Coordinate' and 'Location Y-Coordinate' columns contain numerical values. The 'Power Pins Visible' column has checkboxes. The 'Source Library' and 'Name' columns show the source library and name for each part. At the bottom, there are buttons for OK, Cancel, New..., Copy, Remove, Paste, and Help.

Implementation Path	Reference	Source Part	Part Reference	Designator	Primitive	Implementation Path	Source Package	Color	Value	Location X-Coordinate	Power Pins Visible	Source Library	Name
=R2	R2	R.Normal	R2		DEFAULT	=R2	=R	Default	100	485	<input checked="" type="checkbox"/>	D:\CADENC	INS626
=R2	R2	R.Normal	R2		DEFAULT	=R2	=R	Default	10ohm	600	<input checked="" type="checkbox"/>	D:\CADENC	INS173
=R2	R2	R.Normal	R2		DEFAULT	=R2	=R	Default	50ohm	490	<input checked="" type="checkbox"/>	D:\CADENC	INS155
=R3	R3	R.Normal	R3		DEFAULT	=R3	=R	Default	1k	330	<input checked="" type="checkbox"/>	D:\CADENC	INS279
=R4	R4	R.Normal	R4		DEFAULT	=R4	=R	Default	1k	450	<input checked="" type="checkbox"/>	D:\CADENC	INS290
=R4	R4	R.Normal	R4		DEFAULT	=R4	=R	Default	20k	390	<input checked="" type="checkbox"/>	D:\CADENC	INST742

Save as HTML

You can also save your search results in the HTML format.

1. Right-click a search results 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 Results window in CSV format.

1. Right-click a search result line item.
2. Choose *Save as CSV*.

A message displays with the location and name of the exported CSV.

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

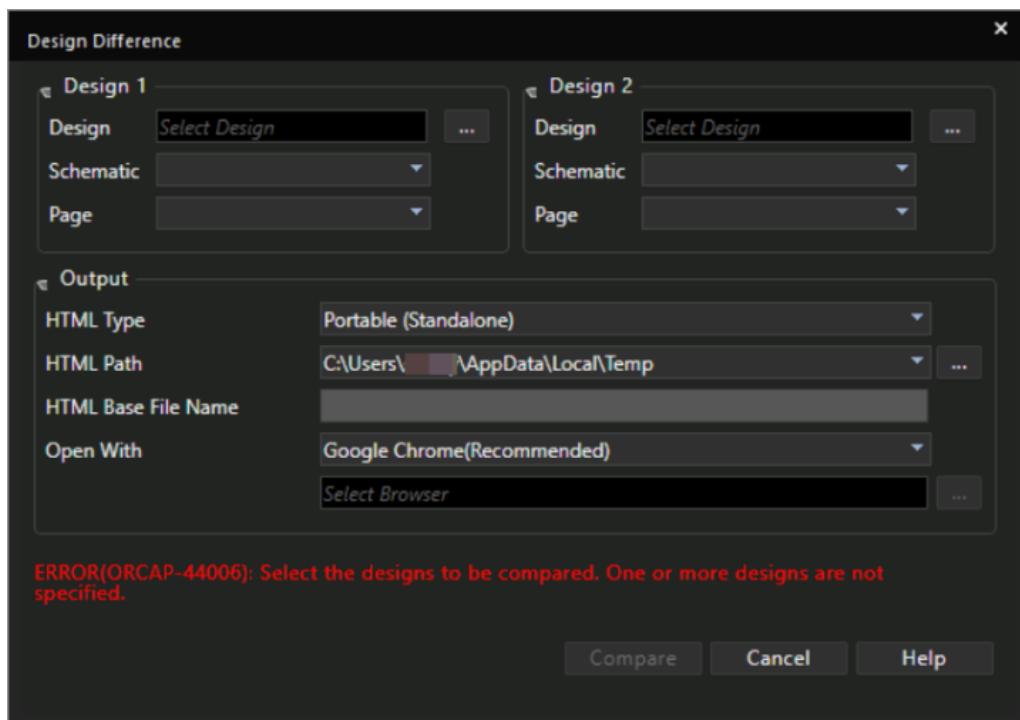
- From the File menu choose Close.

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

Compare Designs

You can compare designs, schematic folders, schematic pages in Capture using the Design Differences dialog box. To access the Design Differences dialog box in Capture, select *Tools - Compare Designs*.

You can compare two designs, two individual schematic folders or All schematic folders, two individual schematic pages or All schematic pages.



Using the Design Difference dialog box, you can generate either a lightweight HTML or a portable HTML to view the differences. The lightweight HTML requires the Cadence hierarchy on the machine. The portable HTML, which takes more space on the disk than the lightweight HTML, does not require the Cadence hierarchy and can be launched from any machine.

After you click *Compare*, you can view logical differences and all differences in the Design Difference Summary window. All differences consists of logical and graphical differences. To view details about differences, click the Show Details button in the Design Differences Summary window.

Design Difference Summary

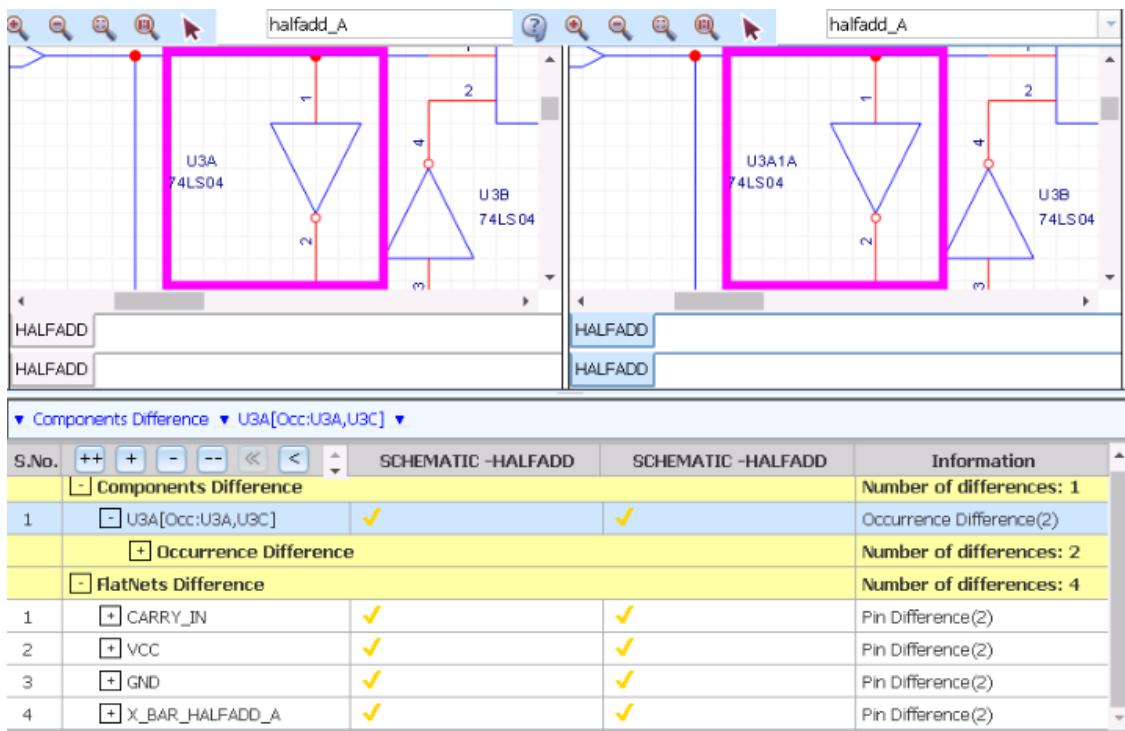
Differences generated using **Object ID**. Regenerate using [Object Name / Part Reference](#).

Logical Differences		All Differences	
		Show Details	
Object	Number of Differences	Object	Number of Differences
Components	1	Page	1
Pin Net Connectivity	0	Components(Logical Differences)	1
FlatNet	4	Components(All Differences On Matching Pages Only)	1
		Pin Net Connectivity	0
		FlatNet	4
		Wire	0
		OffPage	0
		Hierarchical Port	0
		TitleBlock	1

General Information		
Selected Options	Design1	Design2
Design	D:\Cadence1\SPB_17.2\tools\capture\samples\Schematic\FULLADD\FULLADD.DSN	D:\Cadence1\SPB_17.2\tools\capture\samples\Schematic\FULLADD\FULLADD_MOD.DSN
Schematic	HALFADD	HALFADD
Page	<All>	<All>
HTML Type	Lightweight (Cadence hierarchy dependent)	
HTML Base File Name	fulladd-halfadd_vs_fulladd_mod-halfadd	

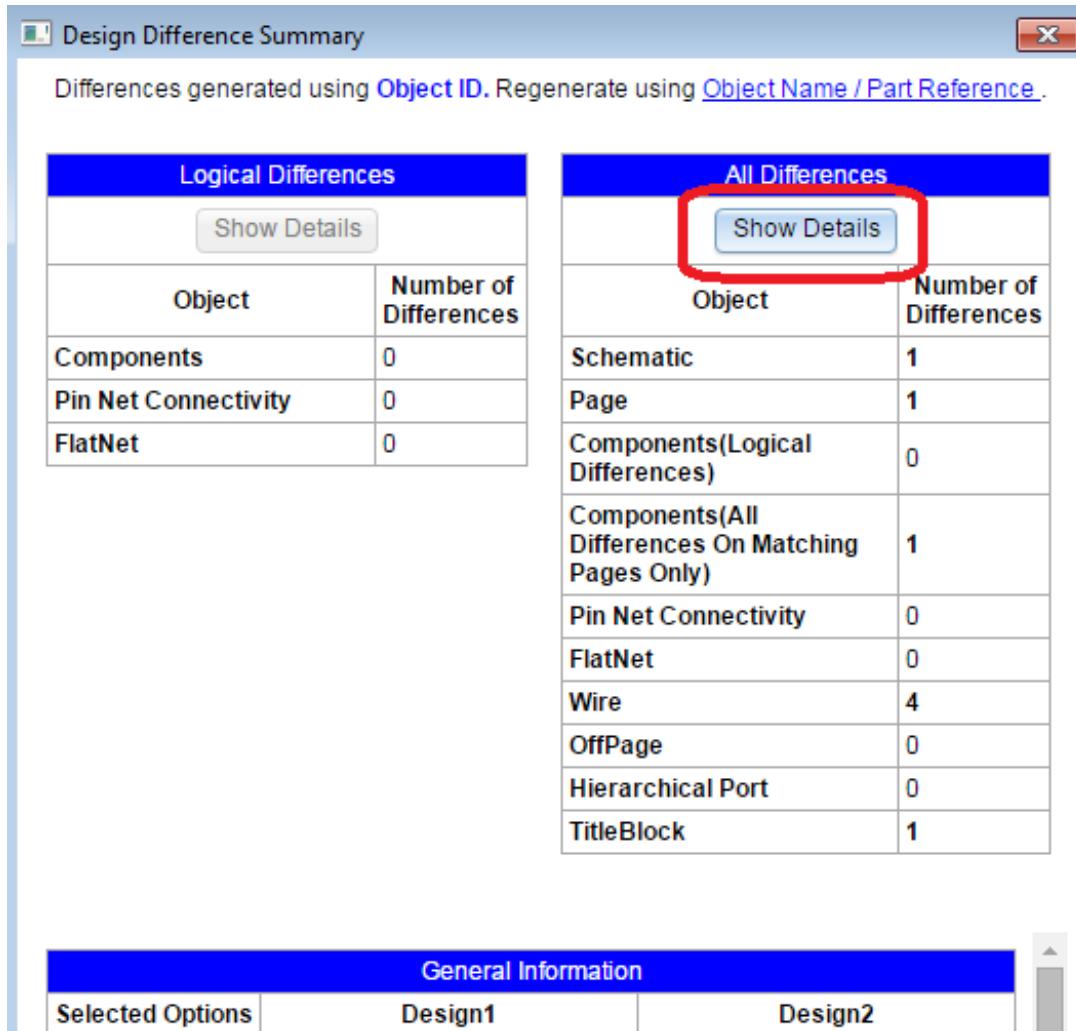
[Close](#)

The following is an example of logical differences displayed in the HTML browser. In this example, two different part references are compared.



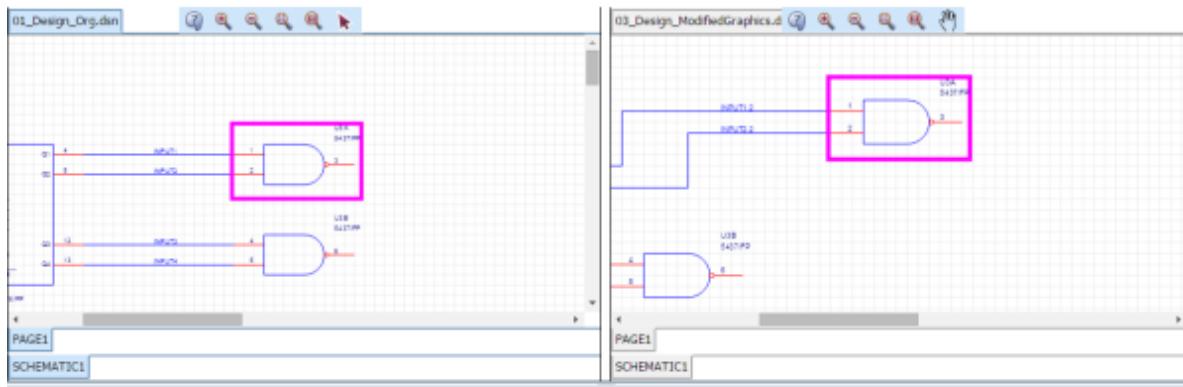
▼ Components Difference ▼ U3A[Occ:U3A,U3C] ▼						
S.No.	++	+	-	SCHEMATIC -HALFADD	SCHEMATIC -HALFADD	Information
1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> U3A[Occ:U3A,U3C]	<input checked="" type="checkbox"/>	Number of differences: 1 Occurrence Difference(2)
	<input checked="" type="checkbox"/> Occurrence Difference			Number of differences: 2		
	<input type="checkbox"/>			Number of differences: 4		
1	<input checked="" type="checkbox"/> CARRY_IN	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Pin Difference(2)
2	<input checked="" type="checkbox"/> VCC	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Pin Difference(2)
3	<input checked="" type="checkbox"/> GND	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Pin Difference(2)
4	<input checked="" type="checkbox"/> X_BAR_HALFADD_A	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Pin Difference(2)

To view details about all differences, click the *Show Details* button in the All Differences table of the Design Difference Summary window.



Following is an example of graphical differences displayed in the HTML browser. In this example, the part's location in the first design is different from the part's location in the second design.

OrCAD X Capture User Guide
Working with Designs--Compare Designs



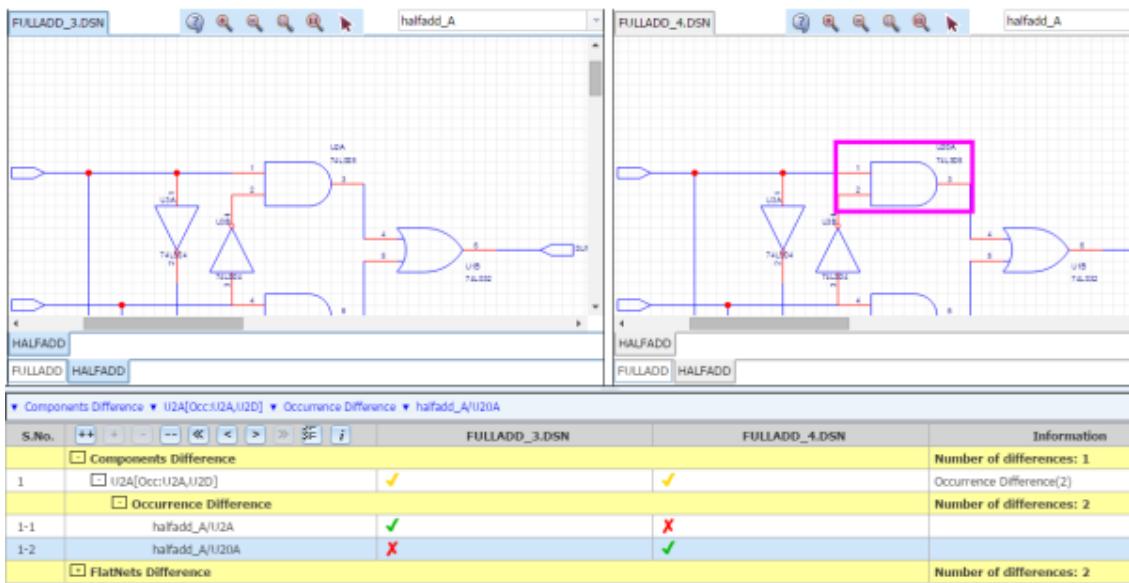
S.No.		01_DESIGN_ORG.DSN	03_DESIGN_MODIFIEDGRAPHICS.DSN	Information
1	<input type="checkbox"/> Schematic Difference	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Number of differences: 1
1-1	<input type="checkbox"/> SCHEMATIC1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Number of differences: 1
1-1-1	<input type="checkbox"/> Page Difference	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Number of differences: 1
1-1-1-1	<input type="checkbox"/> PAGE1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Number of differences: 1
1-1-1-2	<input type="checkbox"/> Components Difference	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Number of differences: 1
1-1-1-3	<input type="checkbox"/> U3A	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Property Difference(Z)
1-1-1-4	<input type="checkbox"/> Property Difference	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Number of differences: 2
1-1-1-5	Location X-Coordinate	470	670	
1-1-1-6	Location Y-Coordinate	110	60	

Using the Compare Design feature, you can verify if two designs are same part with different occurrence value.

Differences generated using Object ID. Regenerate using Object Name / Part Reference .	X		
Logical Differences		All Differences	
Object	Number of Differences	Object	Number of Differences
Components	1	Schematic	2
Pin Net Connectivity	0	Page	2
FlatNet	2	Components(Logical Differences)	1
		Components(All Differences On Matching Pages Only)	1
		Pin Net Connectivity	0
		FlatNet	2
		Wire	0
		OffPage	0
		Hierarchical Port	0
		TitleBlock	2

General Information		
Selected Options	Design1	Design2
D:\WorkData\Capture\17.2\OrCA	D:\WorkData\Capture\17.2\OrCA	

Following is an example of logical differences displayed in the HTML browser. These logical differences display different occurrence values of the same part in two different design.



Importing Component Differences to Microsoft Excel and HTML

To export component differences found in the Compare Designs window for logical and all differences to a Microsoft Excelesheet, click on  in the generated Compare Design html file.

Following is an example of an excel export that illustrates exporting of component differences to an .xls file:

S.No.	SCHEMATIC -HALFADD	SCHEMATIC -FULLADD	Information
Components Difference			Number of differences: 9
1	U3A[Occ:U3A,U3C]	✓	✗
2	U2A[Occ:U2A,U2D]	✓	✗
3	U3B[Occ:U3B,U3D]	✓	✗
4	U2B[Occ:U2B,U4A]	✓	✗
5	U2C[Occ:U2C,U4B]	✓	✗
6	U1B[Occ:U1B,U1C]	✓	✗
7	halfadd_A	✗	✓
8	halfadd_B	✗	✓
9	UIA[Occ:UIA]	✗	✓
FlatNets Difference			Number of differences: 9
1	X_BAR_HALFADD_A	✓	✗
2	N5056796111_HALFADD_A	✓	✗
3	N00032_HALFADD_A	✓	✗
4	N00034_HALFADD_A	✓	✗
5	X_BAR_HALFADD_B	✓	✗

Similarly, to export component differences found in the Compare Designs window for logical and all differences to a HTML file, click on  in the generated Compare Designs html file.

Following is an example of an HTML export that illustrates exporting of component differences to a HTML file:

S.No.		SCHEMATIC -HALFADD	SCHEMATIC -FULLADD	Information
	Components Difference			
1	U3A[Occ:U3A,U3C]	Present	Not Present	
2	U2A[Occ:U2A,U2D]	Present	Not Present	
3	U3B[Occ:U3B,U3D]	Present	Not Present	
4	U2B[Occ:U2B,U4A]	Present	Not Present	
5	U2C[Occ:U2C,U4B]	Present	Not Present	
6	U1B[Occ:U1B,U1C]	Present	Not Present	
7	halfadd_A	Not Present	Present	
8	halfadd_B	Not Present	Present	
9	U1A[Occ:U1A]	Not Present	Present	
	FlatNets Difference			
1	X_BAR_HALFADD_A	Present	Not Present	
2	N5056796111_HALFADD_A	Present	Not Present	
3	N00032_HALFADD_A	Present	Not Present	
4	N00034_HALFADD_A	Present	Not Present	
5	X_BAR_HALFADD_B	Present	Not Present	
6	N5056796111_HALFADD_B	Present	Not Present	
7	N00032_HALFADD_B	Present	Not Present	
8	N00034_HALFADD_B	Present	Not Present	
9	CARRY_OUT	Not Present	Present	

Exporting and Importing a Design from XML

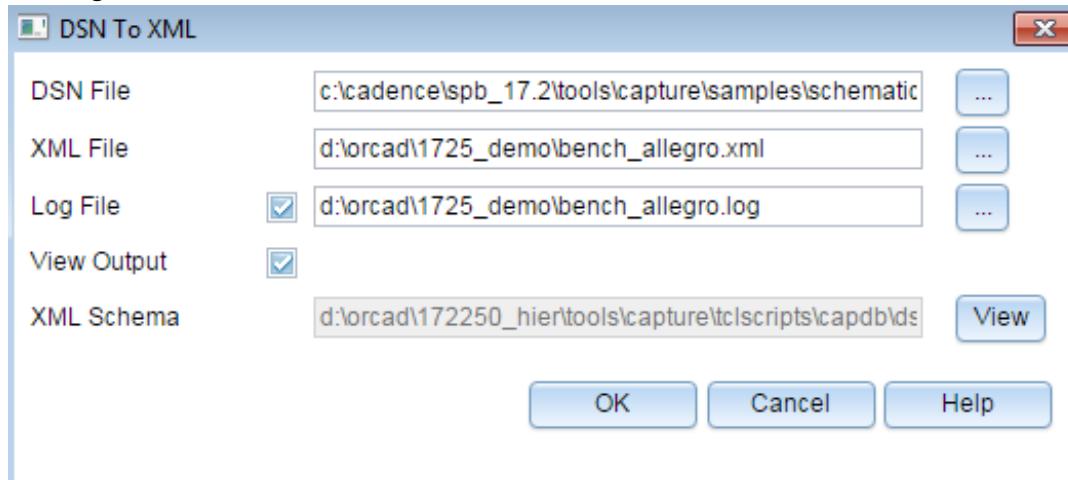
You can export Capture designs to XML format. To do so, perform the following tasks:

1. Open the required design in Capture.

2. Select *File – Export – Design XML*.

The DSN To XML dialog box opens with the design file path already updated in the *DSN File* field.

3. Change the XML file name and location, if needed.



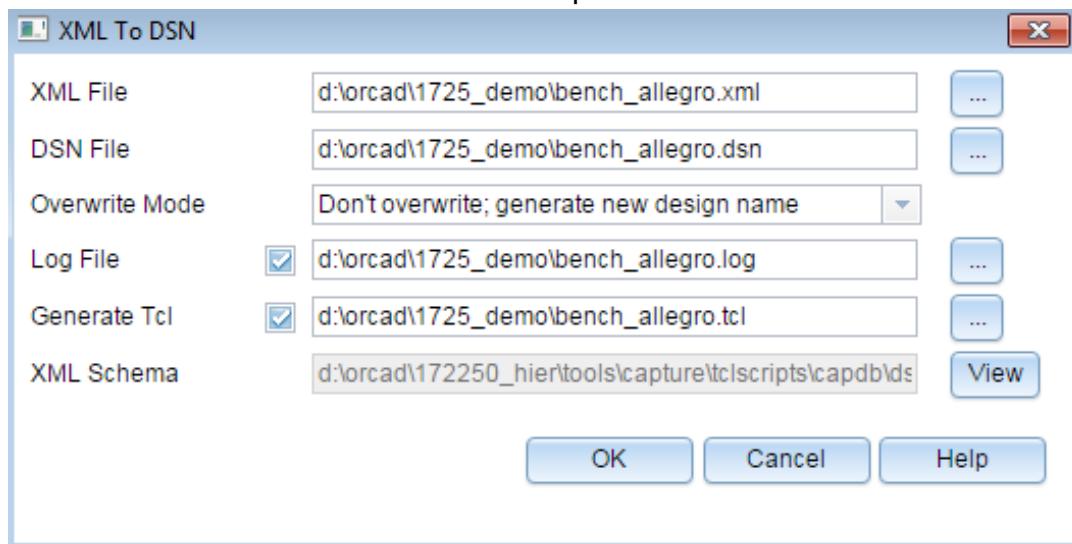
To convert the Capture design (.dsn) file to an XML file, Capture uses an XML schema that is located at the following

path: <installation_directory>\tools\capture\tclscripts\capdb\dsn.xsd

4. Click *OK*.

To import an XML to create a Capture design, do the following:

1. Launch Capture.
2. Select *File – Import – Design XML*.
The XML to DSN dialog box opens.
3. Specify the input XML file path.
The *DSN File* field gets updated.
4. Click the Browse button corresponding to the *DSN File* field if you want to save the generated design at a different location. This is optional.
5. Select the *Overwrite Mode* from the drop-down list.



6. Click *OK*.

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](#)
- [Deleting a Schematic Folder](#)
- [Renaming a Schematic Folder](#)
- [Moving Schematic Folders](#)
- [Attaching a Schematic Folder](#)

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 OK.

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.

Working with Schematic Pages

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.

In this section:

- [Creating a Schematic Page](#)
- [Defining Schematic Page Characteristics](#)
- [Working with Label States](#)
- [Moving Schematic Pages](#)
- [Renaming a Schematic Page](#)
- [Deleting a Schematic Page](#)
- [Closing and Saving a Schematic Page](#)

Creating a Schematic Page

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.

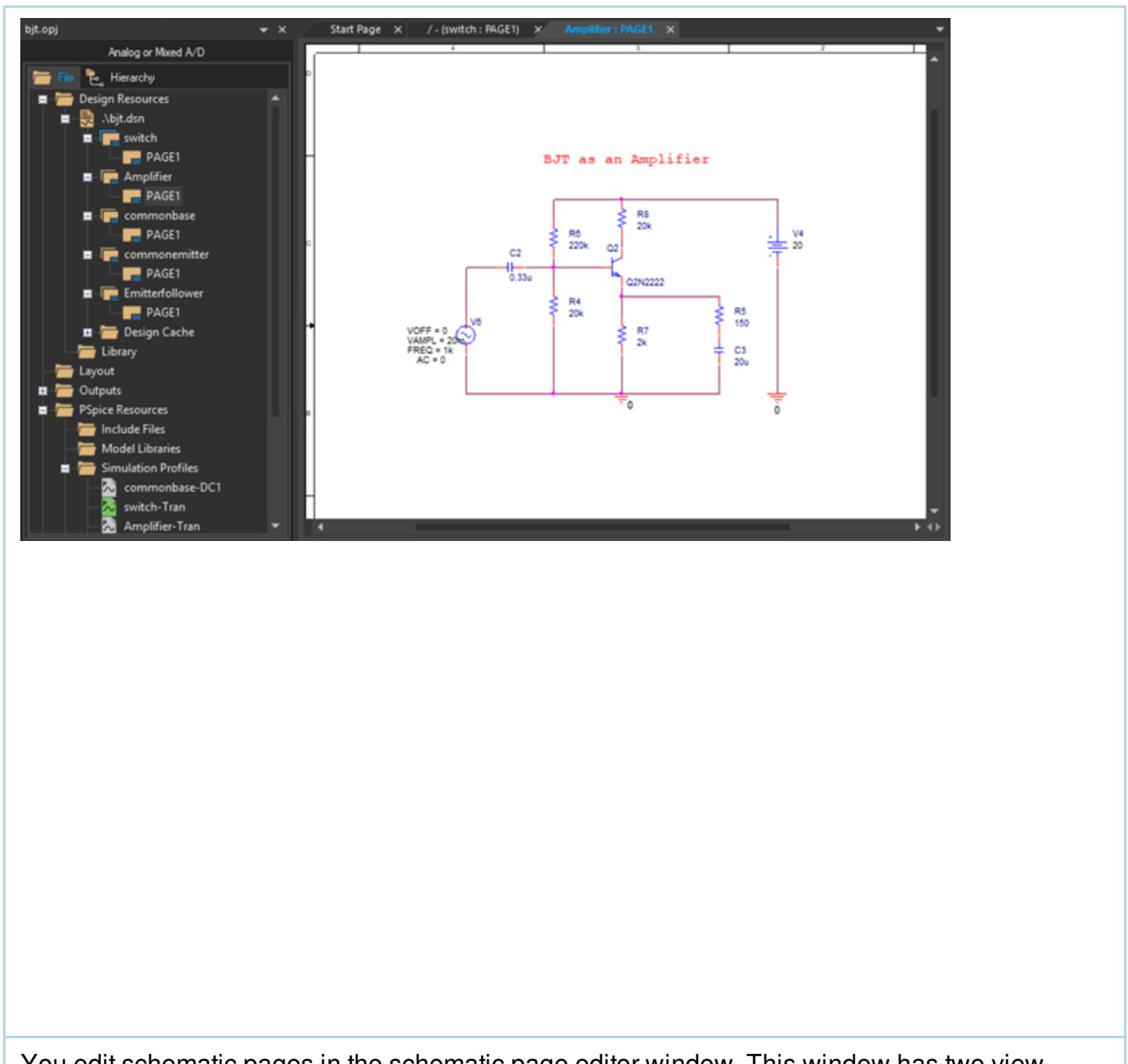
3. Select a page size.
 - If you choose inches, the choices are *A*, *B*, *C*, *D*, *E*, and *Custom*.
 - If you choose millimeters, the choices are *A4*, *A3*, *A2*, *A1*, *A0*, and *Custom*.
4. Specify a value for pin-to-pin spacing, and click *OK* to close the Design Template dialog box.

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

Schematic Page Editor and Part Editor

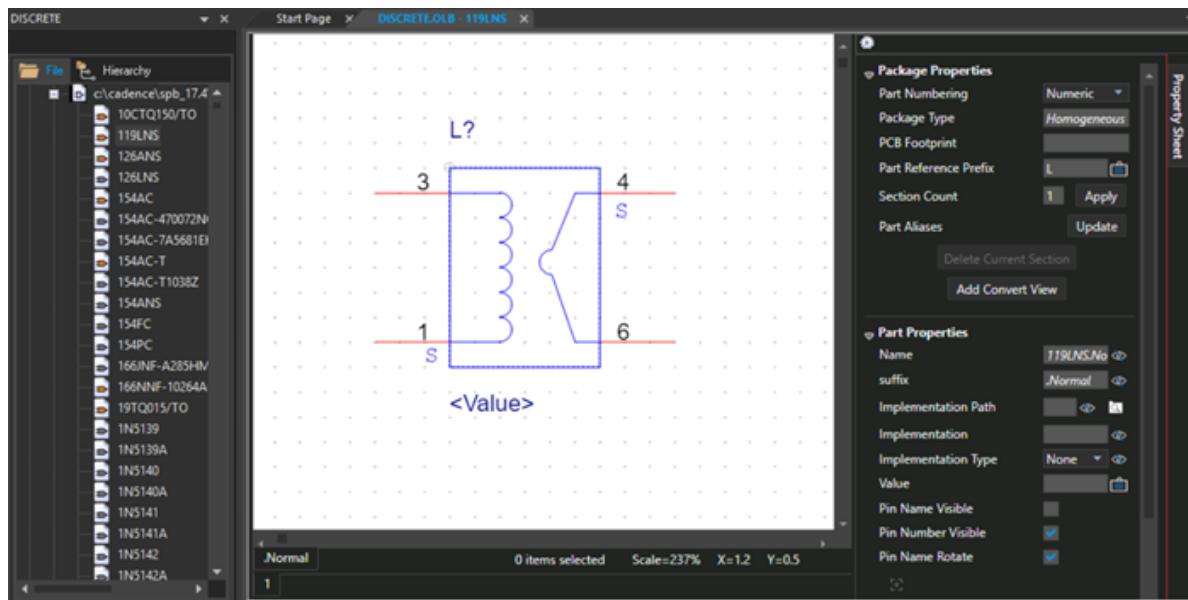
OrCAD X Capture includes a number of editors, including a text editor with features for creating VHDL models, a schematic editor, and a part editor. These editors mostly function in accordance with the general standards that one might expect in a Windows-based tool. However, there are certain unique 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.

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.

Moving Around in the Editors

When working on a schematic page, some operations help you move around the schematic easily, such as [scrolling](#), [panning](#), [zooming](#) in and out, [moving](#) to a specific location, reference or bookmark, or even using the [non-linear](#) (fisheye) mode to set focus to specific objects on the schematic.

Working with Objects

Capture provides standard functions to work with schematic objects—[copying](#), [moving](#), [mirroring](#), [rotating](#), and [selecting and deselecting objects](#).

Scrolling

Capture provides the Scrolling function to help you focus on different portions of the active window—scroll up, down, to the left, or to the right. Even though some objects on the *Place* menu are attached to your pointer while you are placing them, you can still scroll.

- Click either side of the scroll button to scroll the panning distance in the corresponding direction—up or down using the vertical scroll bar or right or left using the horizontal scroll bar.
- Click 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 in the schematic page editor.
- Press the left arrow key to scroll the panning distance to the left in the part editor.
- Press `Ctrl+Page Down` to scroll the panning distance to the right.
- Press the right arrow key to scroll the panning distance to the right in the part editor.
- Roll the mouse wheel up and down to scroll through vertically in the schematic page editor, part 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.
- In the schematic page 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, such as moving an object, drawing a selection area, and so on, 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.

To pan the display region while not performing any action, for example while viewing, you can use the Scroll mouse button:

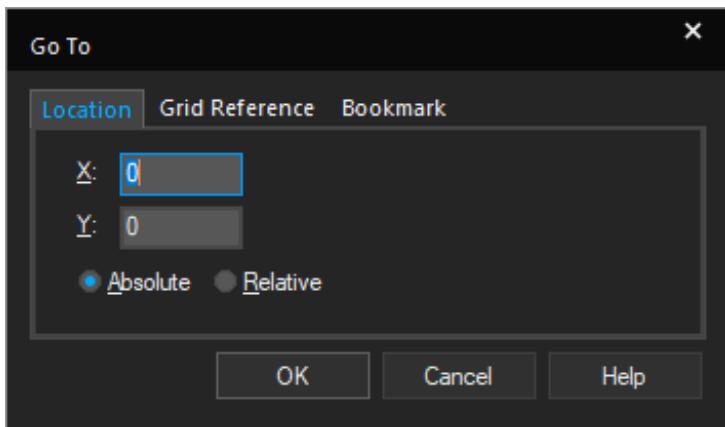
1. Click the scroll button and a Pan cursor appears.
 2. Now move the mouse anywhere across the page to pan the display area.
As you move close to the edge of the display area, the area out of display will move into display.
 3. To end panning, click any mouse button or press the `Esc` key.
-
- To change the display region 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, do the following:

1. Choose *Options – Preferences*
2. Select the *Pan and Zoom* tab.
3. In the *Auto Scroll Percent* text box, specify 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.
4. Click *OK*.

Moving to a Location

To move to specific location, such as grid coordinates, references, or bookmarks in an editor, use the *Go To* command. 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.



⚠ In the part editor, the *Grid Reference* and *Bookmark* tabs are not available in the *Go To* dialog box.

To move to a specific location, do the following:

1. Choose *View – Go To*.

2. In the *Location* tab:

a. Specify the X and Y values.

When moving to a specific location, note that the X and Y coordinates of the current location of the pointer appear:

- on the right-hand side of the status bar in the schematic page editor.
- on the bottom right corner of the part editor window.

b. Select the *Absolute* option.

3. 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](#) or [Design Properties](#) dialog box.

The pointer moves to the new coordinates.

⚠ In the part editor, only the view moves to the given location and not the mouse pointer.

To move a specific distance, do the following:

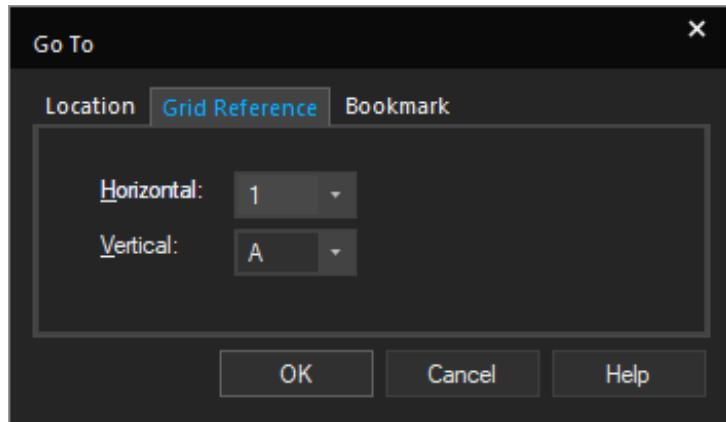
1. Choose *View – Go To*.
2. In the *Location* tab:
 - a. Specify the X and Y values that you want the pointer to move.
 - b. Select the *Relative* option.
3. Click *OK*.

The jump distance is measured in *inches* or metric units, depending on what you specified in the *Page Size* tab of the [Design Template](#) or [Design Properties](#) dialog box.

The pointer moves the specified distance.

To move to a specific grid reference, do the following:

1. Choose *View – Go To*.
2. Select the *Grid Reference* tab.

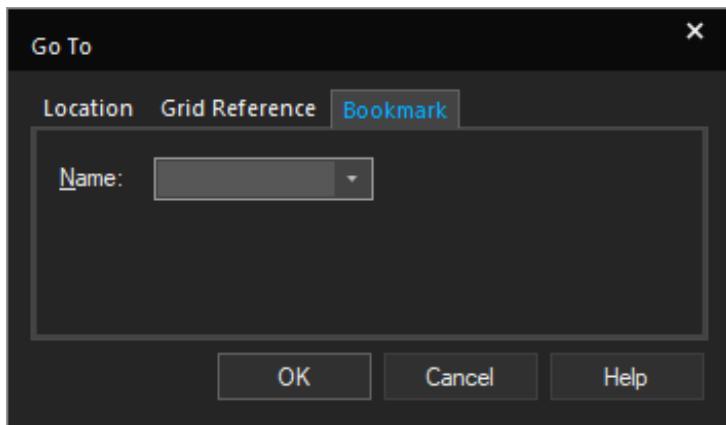


3. Enter the horizontal and vertical information corresponding to the grid reference and click *OK*.
The pointer moves the specified grid reference.

- i** In the schematic page editor, grid references appear on the left and upper edges of the schematic page.

To move to a specific bookmark, do the following:

1. Choose *View – Go To*.
2. Select the *Bookmark* tab.



3. Enter the name of the bookmark or select from the list.
4. Click *OK*.

The pointer moves the selected bookmark location.

Using Zoom

In the schematic page editor and the part editor, you can zoom in to look closely at a particular area. You can also use other zoom functions as required.

This section covers the following topics:

- [Zooming In](#)
- [Zooming Out](#)
- [Changing the Zoom Factor](#)
- [Zooming to a Specific Scale](#)

- [Viewing a Specific Area](#)
- [Viewing the Entire Schematic Page or Part](#)
- [Centering the View in Schematic Page Editor](#)
- [Refreshing the Display in Schematic Page Editor](#)

Zooming In

When you press **I** to zoom in on the schematic canvas, Capture centers your view on the current pointer position. If the pointer is outside the window or if you choose the [Zoom In](#) command or the toolbar button, Capture centers your view on the selected objects. Otherwise, Capture zooms in to the center of the active window.

You can also zoom into a specific object on the page by using the right mouse button. You can click 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 to an object, do the following, choose *View – Zoom* and then choose [In](#).

The current zoom scale is multiplied by the [zoom factor](#). For example, a zoom factor of 2 causes the image to appear twice as large and displays half the area of the previous view. Alternatively, hold down the **CTRL** key and roll up the mouse wheel.

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* and then choose [Out](#).

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.

Alternatively, hold down the **CTRL** key and roll down the mouse wheel.

⚠ 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.

Shortcut

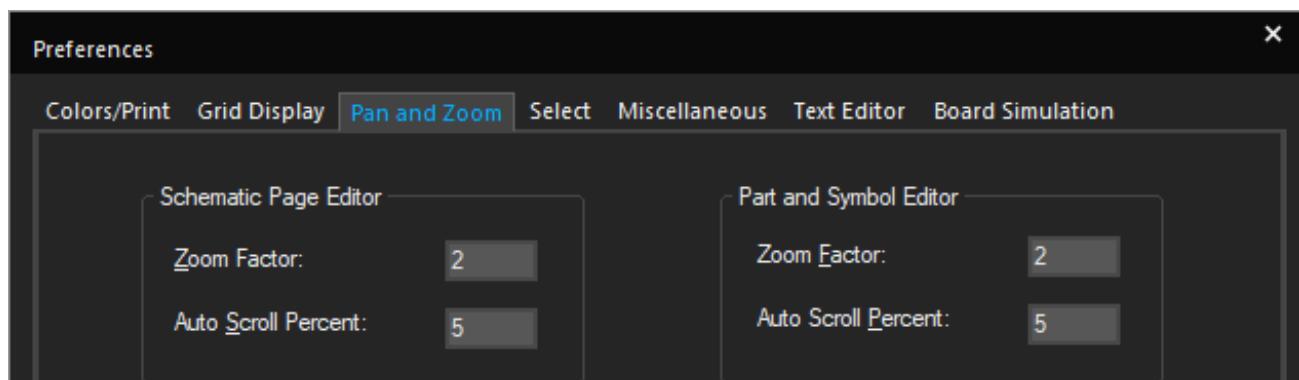
Toolbar: 
Keyboard: O

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, do the following:

1. Choose *Options – Preferences* and then choose the **Pan and Zoom** tab.



2. In the *Zoom Factor* text box, specify the new zoom factor.
The zoom factor can be any number between 1 and 10.
Note that you can specify separate values for the schematic page editor and the part editor.
3. Click *OK*.

Zooming to a Specific Scale

To view a part or schematic page at a specific scale, do the following:

1. Choose *View – Zoom* and then choose [Scale](#).
2. Select a preset scale or specify a custom scale.
3. Click *OK*.

 The *View – Zoom – Scale* command is not enabled for the part editor.

Viewing a Specific Area

To view a specific area of the schematic page editor or part editor, do the following:

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: 

Viewing the Entire Schematic Page or Part

You can view the entire schematic page at the same time. For a schematic page, Capture uses the dimensions set in the *Page Size* tab in the [Schematic Page Properties dialog box](#)

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

Shortcut

Toolbar: 

Centering the View in Schematic Page Editor

You can center the view on your pointer or focus the view on a specific object.

To center the view on a specific object, do the following:

1. Select one or more objects, or an area.
2. Choose *View – Zoom – Selection*.

The display scrolls so that the selected objects are at the center of the window. The zoom factor does not change.

To center the view on your pointer, do the following:

1. Move the pointer over the area to be centered.
2. Press `SHIFT+C` or just `C`.

Refreshing the Display in Schematic Page Editor

To refresh the display, choose [View – Zoom – Redraw](#).

Shortcut

Keyboard: `F5`

Setting a Bookmark

A bookmark comes in handy 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. When you set a bookmark, you need to 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.

To place a bookmark, do the following:

1. Choose [Place – Bookmark](#).
2. Enter the name of the bookmark.
3. Click *OK*.
4. Position the pointer where you want to place the bookmark and click the left mouse button.
The bookmark appears in the selection color.
5. Right-click and choose *End Mode*.
6. Click an area where there are no parts or objects to deselect the bookmark.

To rename a bookmark, do the following:

1. Select the bookmark.
2. Choose *Edit – Properties*.
The [Edit Bookmark dialog box](#) appears.
3. Enter a new name in the text box and click *OK*.

Magnifying Objects Using Fisheye

The *Fisheye* functionality provides a magnified view of the schematic to help you focus on a specific portion of the schematic. The two basic Fisheye features include [*Fisheye focus*](#) and the [*Dynamic Fisheye View*](#) mode.

Fisheye focus lets you set the focus to specific objects on your schematic. Setting the Fisheye focus to one or more objects on the schematic ensures that the selected objects are magnified.

Meanwhile, the other visible objects are demagnified. This ensures that you still have a view of the page, but the focus is on selective objects only.

 The *Fisheye* mode is page specific and not design specific. Also, moving in and out of the Fisheye mode retains 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.

 Use the Find functionality in conjunction with the Fisheye feature. Finding an object on the page will set the focus to the object. Pressing `Shift + F11` immediately sets the Fisheye focus on the selected object.

This section covers the following topics:

Fisheye Mode

To use the Fisheye feature, you need to switch into the Fisheye mode. You can use the Fisheye mode to zoom into only specific objects on your schematic.

To switch to the Fisheye mode, right-click the page and choose *Fisheye view*. Alternatively, choose [View – Fisheye – Fisheye view](#).

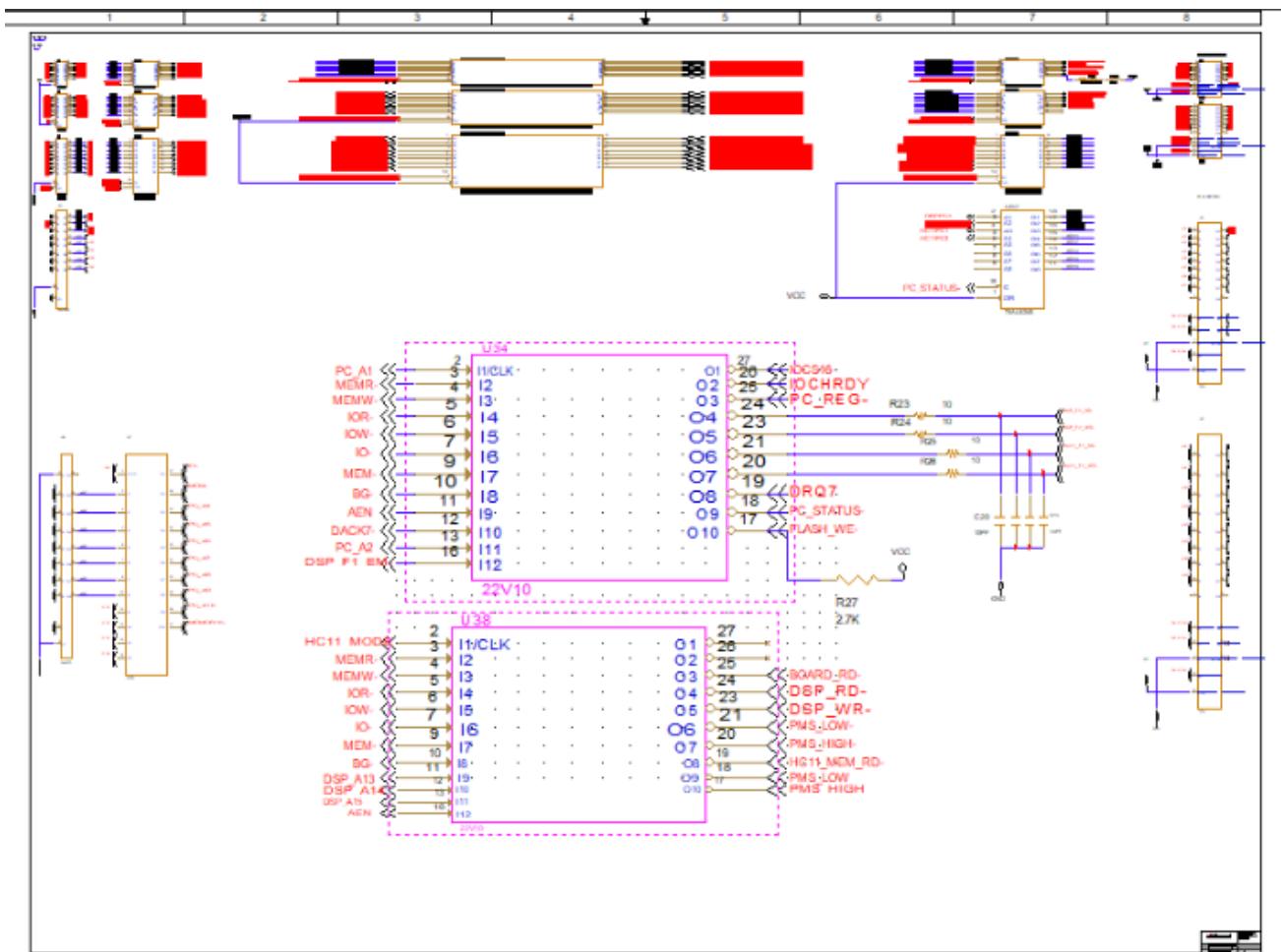
Setting Fisheye Focus

You set the Fisheye focus on a selected set of objects on your schematic. This results in only the selected objects to be zoomed in while the rest of the viewable area remains zoomed out.

To set the Fisheye focus, do the following

1. Select one or more objects on the page.
Use `Ctrl + click` to select multiple objects.
2. Right-click the page and choose [Set Fisheye Focus](#).

The selected items are magnified on the screen.



Shortcut

Keyboard: Shift + F11

To remove the Fisheye focus, right-click the page and choose *Reset Fisheye Focus*.

Shortcut

Keyboard: Shift + Ctrl + F11

Fisheye Dynamic Focus Mode

In the Dynamic Fisheye focus mode, 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 appears zoomed out.

To set the Fisheye Dynamic Focus Mode, right-click the page and choose *Fisheye Dynamic Focus Mode*. Alternatively, choose [View – Fisheye – Fisheye Dynamic Focus Mode](#).

Shortcut

Keyboard: 

Non-Linear Zoom

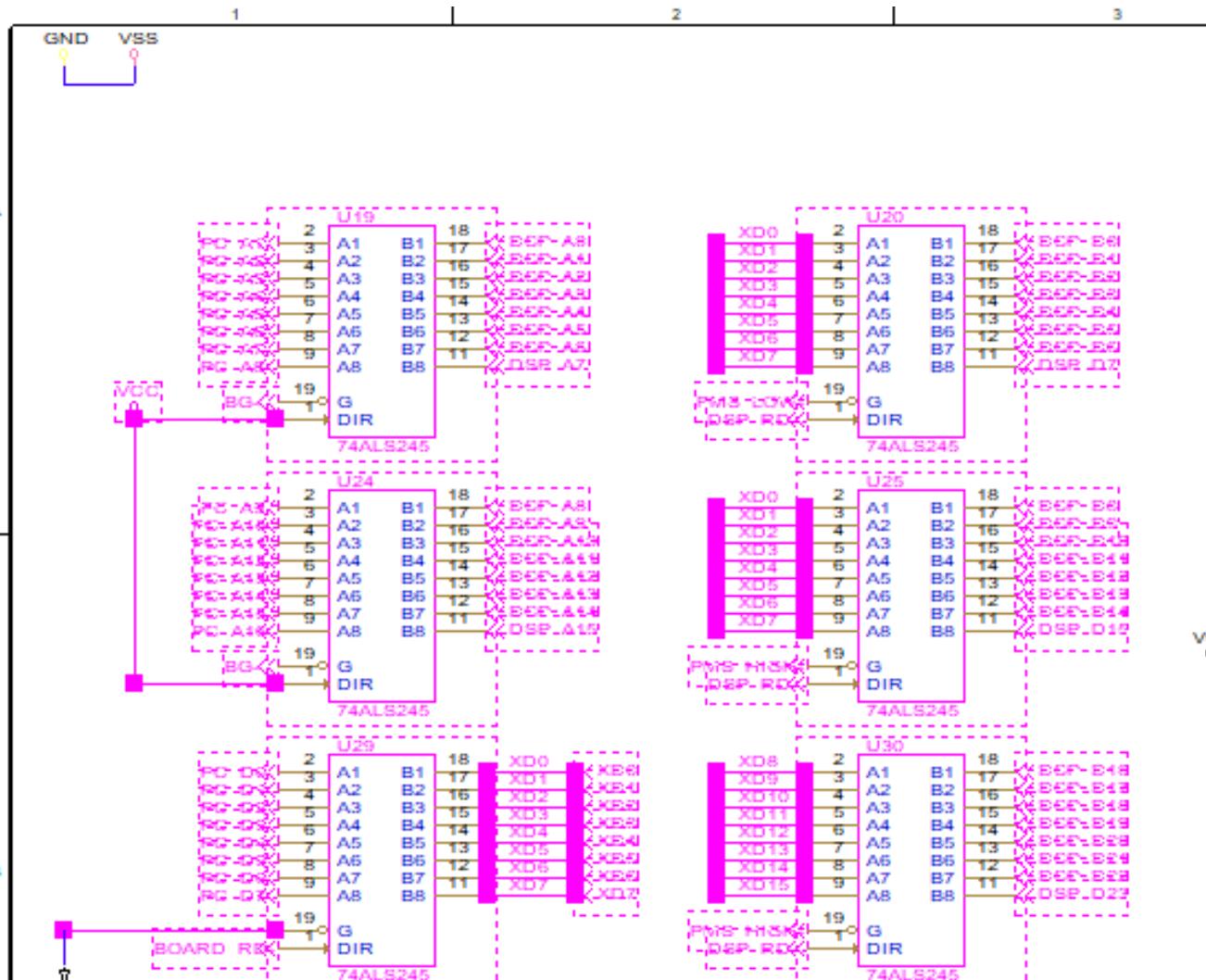
In the Dynamic Fisheye mode, you can further zoom into or zoom out of the view to get higher or lower zoom factor as you pan across the page. The magnification factor ranges from a minimum of 2 to a maximum of 10. This feature is used in conjunction with the Set Fisheye focus and the Dynamic Fisheye View modes to further zoom into or zoom out of the schematic in a non-linear manner.

- To *zoom in* in a non-linear manner, press `Ctrl + +` (Ctrl and plus key combination)
- To *zoom out* in a non-linear manner, press `Ctrl + -` (Ctrl and minus key combination)

Selecting and Deselecting Objects

You select objects to edit, move, or modify them in any way. You can simultaneously modify 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, click the mouse 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 resize 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 select all the objects on a schematic page editor or in the part editor, choose **Edit – Select All**. Alternatively, press **CTRL + A**.

- To add or remove an object from the selection set, position the pointer over the object and press **Ctrl** + click the mouse.

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.

Controlling the Selection of Objects

To control the selection of objects during a mouse-drag operation in the schematic page editor, do the following:

1. Choose [View – Selection Filter](#).

The [Selection Filter](#) dialog box appears.

2. Select the check box corresponding to the object 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.

Selecting all the Objects in an Area

To select all the objects in an area, do the following:

1. From the tool palette, choose the *Select* tool.
2. Move the pointer to one corner of the area.
3. Keeping the left mouse button pressed, drag the pointer to the opposite corner, and 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 the objects intersected by your selection rectangle or only the objects that are fully enclosed by the selection rectangle. From the *Options* menu, choose the [Preferences command](#), and then choose the [Select tab](#).

Selecting a Contiguous Polyline

To select an entire contiguous polyline, do the following:

1. From the tool palette, choose the *Select* tool.
2. Keeping the left mouse button pressed, drag the pointer to select an area that includes some portion of the line.

Selecting from Overlapping Objects

To select from among overlapping objects, do the following:

- In the schematic page editor, 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.
- In the part editor, press the `SHIFT` key and click the overlapped objects repeatedly to select each object one by one in rotation.

Selecting the Entire Net on a Schematic Page

To select all portions of a net on one schematic page, do the following:

1. Select one segment of the net.
The segment changes to the selection color.
2. Right-click and choose [Select Entire Net](#).

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

Deselecting Objects

To deselect objects, click an area where there are no objects, 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 you click the mouse or press `Esc` to end mode and press `Esc` again to deselect the object.

Changing the Selection Color

To change the selection color, do the following:

1. Choose [Options – Preferences](#).
2. Choose the *Colors* tab.
3. Click the Selection color.
The color palette window opens.
4. Select the new color and click *OK* to close the color palette.
5. Click *OK*.

Copying Objects

Capture supports the standard Cut, Copy, and Paste functions in all the editors. You can cut, copy, and paste information across schematic pages. You can copy text from other Windows applications and paste it into Capture text boxes. You can also copy a section of your schematic page to another Windows application.

 In the part editor, you can copy objects and paste them in the part editor only. Similarly, you can copy objects and paste them in the symbol editor only.

To copy and paste objects by dragging, do the following:

1. Select the object or objects.
2. Press and hold both the `ctrl` key and the left mouse button while you drag a copy of the object to another location.
3. Release the left mouse button to place the copied object.



- If you drag 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, do the following:

1. Select the required object.
2. Choose the [Edit – Copy](#) menu command.
The object is placed on the Clipboard.
3. If the object is to be copied to another window, open that window.
4. Choose the [Edit – Paste](#) menu 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. To place another copy of the object, repeat steps 4-6 above.

To copy text or graphics into other Windows applications, do the following:

1. Select the text or graphic.
2. Choose the [Edit – Copy](#) menu command.
The selected objects are copied to the clipboard
3. Open any Windows application and use that Paste command of the application to copy the contents of the clipboard.

Shortcuts



Toolbar:

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 required object.
2. Choose the **Edit – Mirror** menu command.
If the commands of the submenu are not available, the object cannot be mirrored.
3. Choose **Horizontally, Vertically, Both**.

The selected objects flip in the indicated direction.

Keyboard Shortcuts

- Mirror Horizontally: **H**
- Mirror Vertically: **V**

Moving Objects

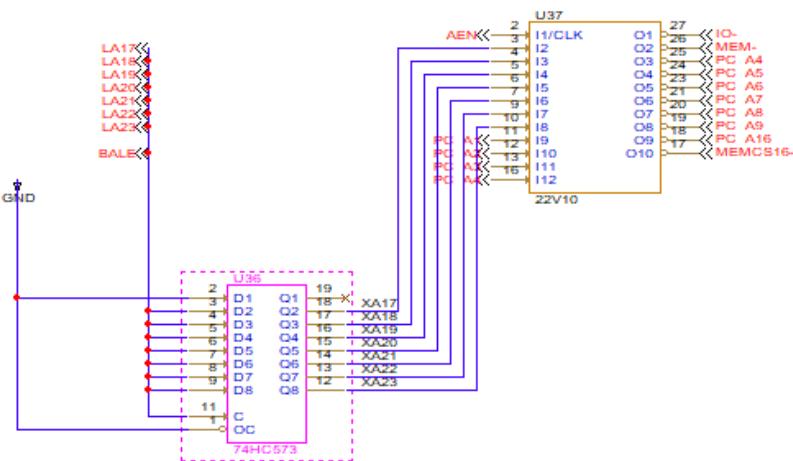
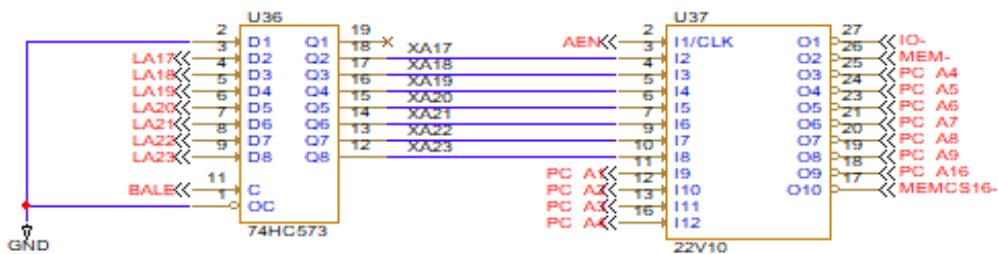
You can easily change the location of objects in the **schematic page editor** or the **part editor**. Immediately after you place an object, you need to select the selection tool or press **Esc** before you move any object.

⚠ The schematic page editor 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 to place the part on **Coarse** grid.

Moving Objects Maintaining the Connectivity

To drag any object, do the following:

1. Position the mouse pointer on the object.
2. Holding the left mouse button down, drag the object to the new location.
3. Release the mouse button.



**Before Moving the Object
Moving the Object (Connectivity Maintained)**

After

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 rubber-banded, that is stretched like rubber bands, to maintain the established net connectivity.

- ⚠ If you are dragging a part or wire to another location and that change affects the connectivity, the cursor changes and temporary markers are displayed on the schematic. Visible and off-screen connectivity changes will be saved in the session log if you complete the operation.

Moving Objects Without Maintaining Connectivity

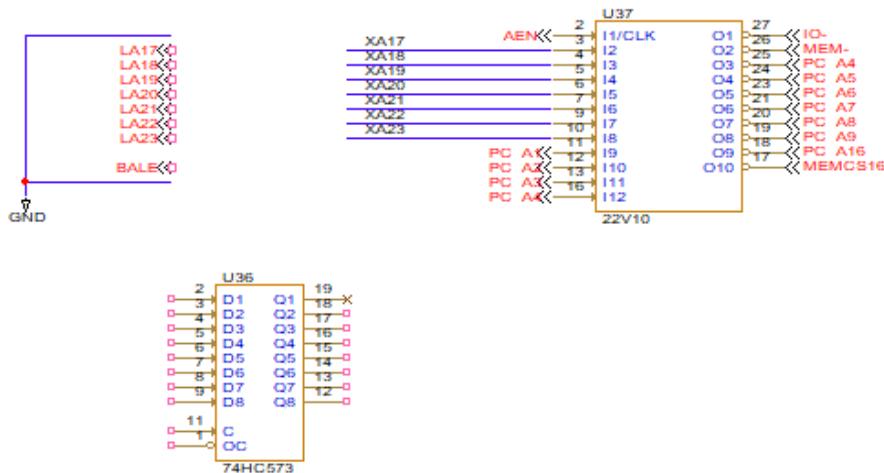
To move an object in the schematic editor without moving the connected nets, do the following:

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.



Loss of Connectivity after Moving the Object

Moving Objects Using Cut Command

To move objects using the Cut command, do the following:

1. Select the required object.
2. Choose **Edit – Cut**.
3. If the object is to be moved to another window, open that window.
4. Choose **Edit – Paste**.
The object is attached to the pointer.
5. Move the pointer to the desired location to place the object and click the left mouse button.
The object appears in the selection color.
6. Click an blank area on the canvas to deselect the object, or press **Esc**.



- In the schematic page editor, 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.
- In the part editor, you can cut or copy objects and paste them in the part editor only. Similarly, you can cut or copy objects and paste them in the symbol editor only.

Rotating Objects

Schematic objects can be rotated by increments of 90-degrees. Some objects, such as images, cannot be rotated.

To rotate an object, do the following:

1. Select the object.
2. Choose *Edit – Rotate*.

The selection set rotates 90 degrees counterclockwise.

If the `Rotate` command does not appear on the *Edit* menu, the objects cannot be rotated.

The pin names and pin numbers on the left and right side of a rotated part retain their default positions. For example, if a pin name is above the pin before a 180 degree rotation, pin name will be above the pin even after the 180 degree rotation as Capture retains the default position after rotation.

Shortcut

Keyboard: `R`

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.
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](#)
- [Creating a Verilog Model from a Hierarchical Block](#)
- [Checking the syntax of VHDL or Verilog files](#)

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 *Edit– Cut*.

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 *Edit– Paste*.
7. Choose *File– Save 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 *File – Save 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 *Rename*.

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.

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 Name	Specifies the name of 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).

⚠ Addition of Page Name property in Title Block will reflect the following changes in Capture from Quarterly Incremental Release (QIR) 3 onwards:

- **In Legacy designs that have no Page Name property in Title Blocks**

Page Name property is automatically added to Title Blocks, when legacy designs are opened.

- **In Legacy designs that have user added Page Name property in Title Blocks**

If user has added Page Name property to the Title Blocks, then the user added Page Name property is not updated on renaming the schematic page name. To solve this issue, delete the user added page name property from the title block.

In this section:

- [Setting up the default title block](#)
- [Creating a custom title block](#)
- [Placing Multiple Title Blocks](#)

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
Page Size	Specifies the size of the schematic page
Page Modify Date	Specifies the date when the schematic page is modified
Page Create Date	Specifies the date when the schematic page is created

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.
Or
Right-click the library and choose *New Symbol*.
The New Symbol Properties dialog box opens.
3. Specify a name and select *Title Block* as the *Symbol Type*.
4. Click *OK*.
The part editor opens with an empty part boundary box.
5. Use the graphics tools to create the title block. See [Creating graphics](#). The part boundary box will stretch to accommodate your graphic objects.
6. In the *Symbol Properties* section of the *Property Sheet* pane, click the add icon to specify a symbol property and its value.

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 Components](#) for further information.
4. Click a title block name in the list for a preview, or double-click on it to place the title block. Alternatively, select the title block name and click **OK** to place the title block.

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 do not 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 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 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. 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](#)
- [Filtering Properties](#)
- [Editing Properties](#)
- [Importing Part and Pin Properties](#)
- [Exporting Part and Pin Properties](#)

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.

Adding a User-Defined Property

To add a user-defined property in Property Editor, do the following:

1. Select an object on a schematic page.
2. Right-click the object and choose *Edit Properties* from the pop-up menu.
The Property Editor for the object opens.
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 the *New Property* button on the top left corner of the Property Editor window.
4. In the Add New Property dialog box, specify a name and value for property.

 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 box.
The new property is added in the Property Editor window.
6. To change the display properties of a property, right-click the property row or column and choose *Display*.
The Display Properties dialog box displays.
In this dialog box, 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 nameYou 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*.

Creating a New Property in the Browse Spreadsheet Editor

To create a new property in the Browse spreadsheet editor, do the following:

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.

Filtering Properties

You can also constrain the set of displayed properties by using the filters available in the drop-down list in the upper right corner of 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 the Actel Designer software. The <Current properties> filter causes Property Editor to display all the properties that currently exist for the selected item.

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 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.

Related Topics

- [Editing Properties](#)
- [Property Spreadsheet Editors](#)
- [The Property Editor Window](#)

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 [Using 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

In the schematic page editor:

1. 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. The *Pin Properties* section appears in the Property Sheet pane.
2.
 - a. Change the properties.
 - b. Save the part.
3. In the schematic page editor:
 - a. Select the pin
 - b. Right-click the pin and choose *Edit Properties*.
 - c. In the Property Editor window, edit the properties of the pin.
 - d. Click *Apply* 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

In the part editor:

1. In the *Section Pins* section of the Property Sheet pane, modify the properties of the required pins.
2. Click the *Apply Pin Changes* button.
3. Save the part.

To edit properties associated with comment text

In the part editor:

1. Double-click the text to edit.
The Edit Comment Text Properties dialog box appears.
2. Make the changes to the text.
3. Click *OK*.

In the schematic page editor:

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, its font, rotation, color, or visibility.
4. Click *OK*.

To edit the name of a property in schematic page editor

1. Double-click 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 in schematic editor

1. Select the object.
2. Right-click the object and choose Edit Properties.
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.

Create New Property 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 Property button. Capture displays the Add new property dialog box.
4. Enter a name for the new property and click *OK*. Capture adds a new property 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.

Copying Values Across Properties

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

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.

Removing User-Defined Properties

To remove a user-defined property in the Browse spreadsheet editor, do the following:

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.

Replacing Property Values

To replace property values, do the following:

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.

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 its 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.

- i** 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:
 - a. In the *Look in* drop-down list, select a new drive, a new directory, or both.
 - b. In the *Files of type* text box, select the type of file you want to open.
4. Click *OK* to apply the properties.

- ⚠** If you edit a library provided, 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 homogeneous part has 2 sections, it will output information corresponding to both the sections. These sections can be 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 does not 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 HEADER line and subsequent lines (delete a column).	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 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, 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, ensure 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" ...
:
```

Managing Projects

In this section:

- [Archiving a Project](#)
- [Auto Recovery](#)
- [Adding and Deleting Project files](#)
- [Undoing and Repeating commands](#)
- [Graphical Operation \(GOp\) Locking](#)

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](#)”.

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 files and directories to the archive:

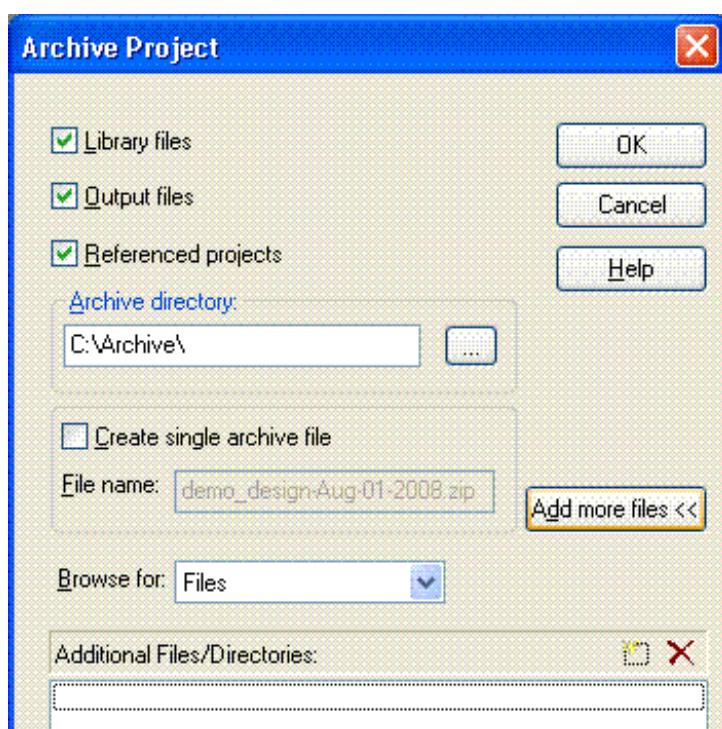
1. Make sure that the project you want to archive is active and the schematic pages for the

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 `%TEMP%\AUTOSAVE` 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 `%TEMP%\AUTOSAVE` directory.
4. If the project itself has been modified, or one of its files was auto-saved, the project itself is saved to the `%TEMP%\AUTOSAVE` 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 `%TEMP%\AUTOSAVE` 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 %TEMP%\AUTOSAVE directory with all paths pointing to the correct location: the %TEMP%\AUTOSAVE 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 FILES.ASL.
- If Capture finds the file, it reads it.
- If Capture closed unexpectedly 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 %TEMP%\AUTOSAVE 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 %TEMP%\AUTOSAVE directory.
- Any files that do are auto recovered to the %TEMP%\AUTOSAVE 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 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](#)
- [Features of a Locked Schematic Page object](#)
- [Features of a Locked Schematic Page](#)
- [Features of a Locked Schematic Folder](#)
- [Features of a Locked Design](#)
- [Cascading and Roll-up effects of Locking](#)

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 example, 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.

Establishing Connectivity

In this section:

- [Wire connectivity](#)
- [Bus connectivity](#)
- [Modifying Wires and Buses](#)
- [Shorting Part Pins](#)
- [Placing power, ground, and no connect symbols](#)
- [Placing off-page connectors](#)
- [Adding hierarchical ports](#)
- [Establishing connectivity in schematic pages](#)
- [Using intersheet references](#)
- [Working with nets](#)
- [Using NetGroups](#)
- [NetGroup and Bus Member Net Generation](#)

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.

It is recommended 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:

- [Placing and naming wires](#)

- [Placing junctions](#)
- [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](#)
- [To place a non-orthogonal wire](#)
- [To attach a wire to a net](#)

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 wire

- 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.

It is recommended 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.



- 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.
- 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](#)

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.



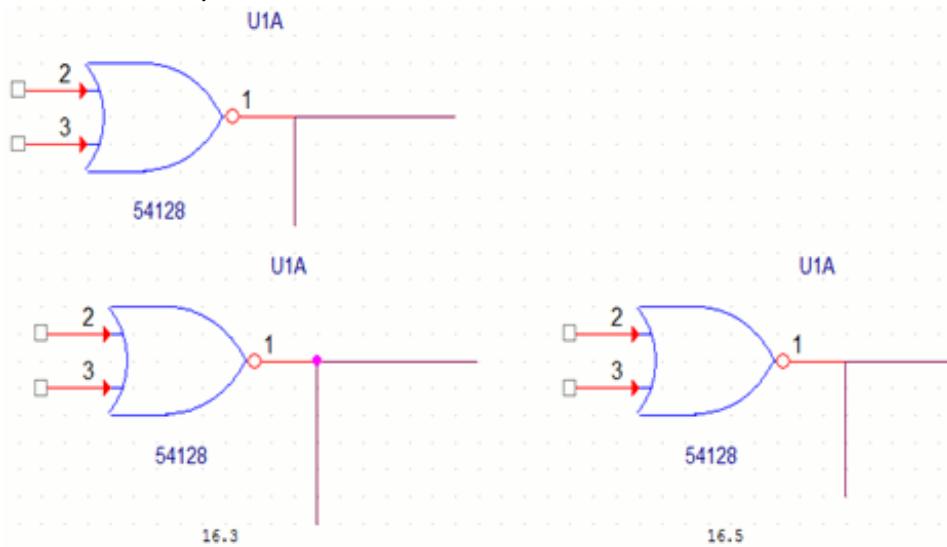
1. Since a junction is the attribute of a wire, it will appear only on a wire-end. If three other types of
For example, if a port, an off-page connector, and a ground are shorted together, a junction dot is not created at the point of connection. Connection objects are shorted together, no junction is 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

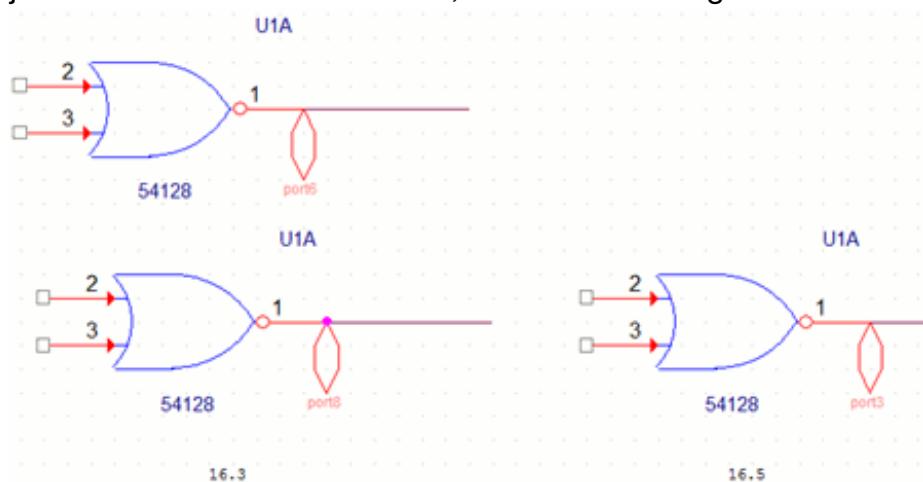
With the implementation of Junction on straight wire-break only, there are some differences in the creation of junctions in v16.5 and later releases of Capture. However, all designs created in releases prior to v16.5 will open with all the existing junctions intact. The enhanced junction creation logic 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](#)
- [Auto-Wire Multiple Points](#)
- [Auto-Wire to Bus](#)
- [Auto-Wire to NetGroup](#)

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 defined 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. Immediately 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 hierarchical 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 run the following rules.

i Important

These rules apply when two buses with different aliases are physically connected to each other or you attach multiple aliases on a single bus.

- 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 buses are with different aliases $A[x..z]$ and $B[p..t]$, the connection is done in bit-wise manner. $A[x]$ is connected to $B[p]$, $A[y]$ is connected to $B[q]$ and so on, where x , z , p , and t are whole numbers.
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. Select a wire or bus on the page.
2. Right-click the wire or bus and choose *Edit Wire Properties*.

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.

3. Make the required selections in the drop-down lists and click *OK*.

 To change the style of the wire to the OrCAD X 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 X 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.

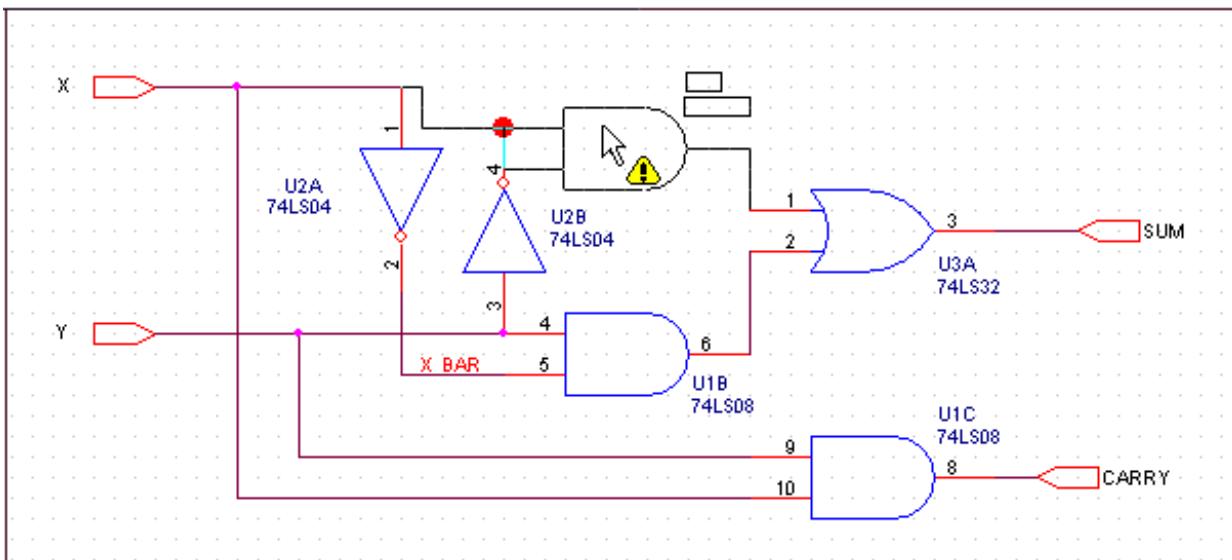
Moving Connectivity Objects

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.

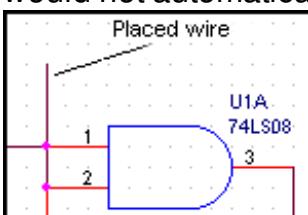


Because some connectivity changes may not be visible on screen, most connectivity changes for which you see an alert are documented in the session log.

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 key 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](#)

- Power and ground symbols

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](#)
- [Making power pins visible](#)

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

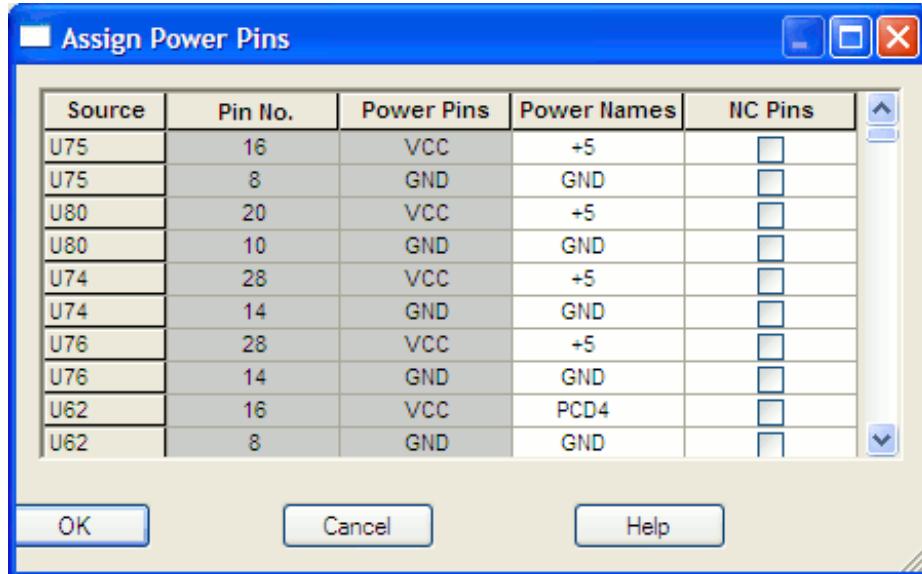
1. Select the design in the Project manager.
2. On the Edit menu choose Browse – Power Pins.
3. In the Browse Properties dialog box, you can choose the mode as occurrences or instances and click OK.

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 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.

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 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 or 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.

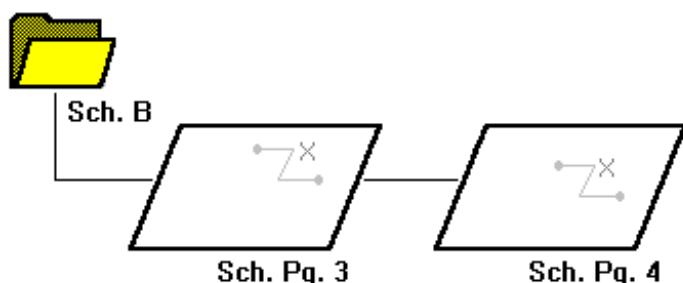
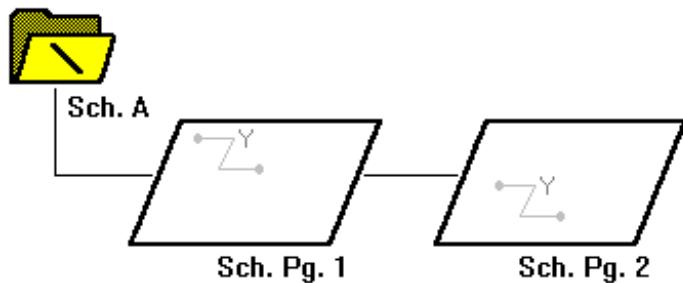
Attached files work much like their counterparts in email—they do not provide an alternative definition of the part (as do attached schematic folders).

Shortcuts

Tool palette:

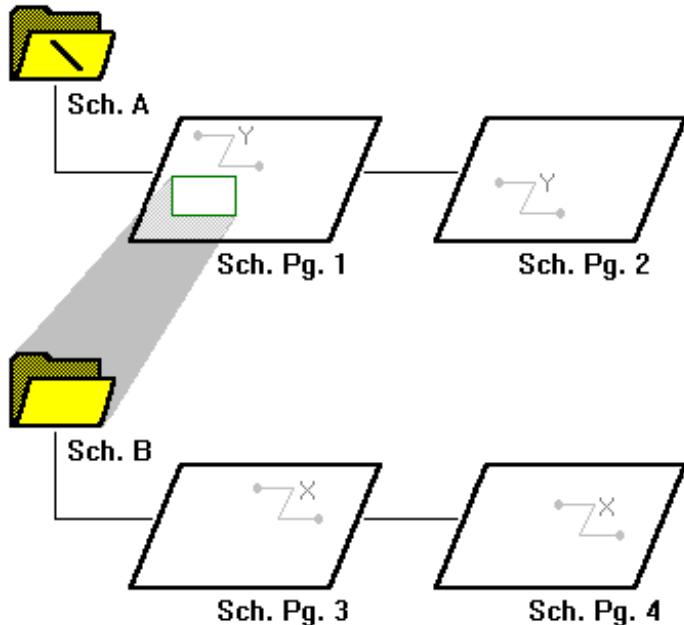


EXAMPLE



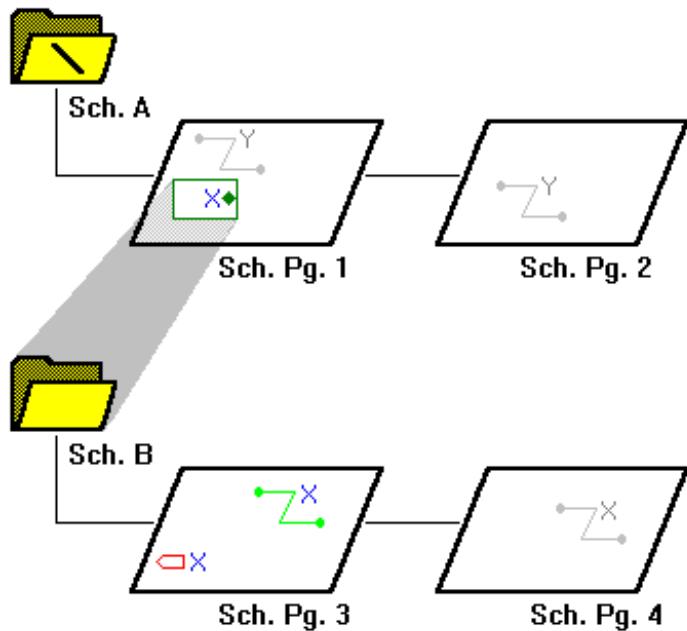
This figure shows two schematic folders, A and B, with two schematic pages each. The schematic folder marked with a backslash (\) is called the [root schematic folder](#). In this demonstration, you see

how to create a simple hierarchy.



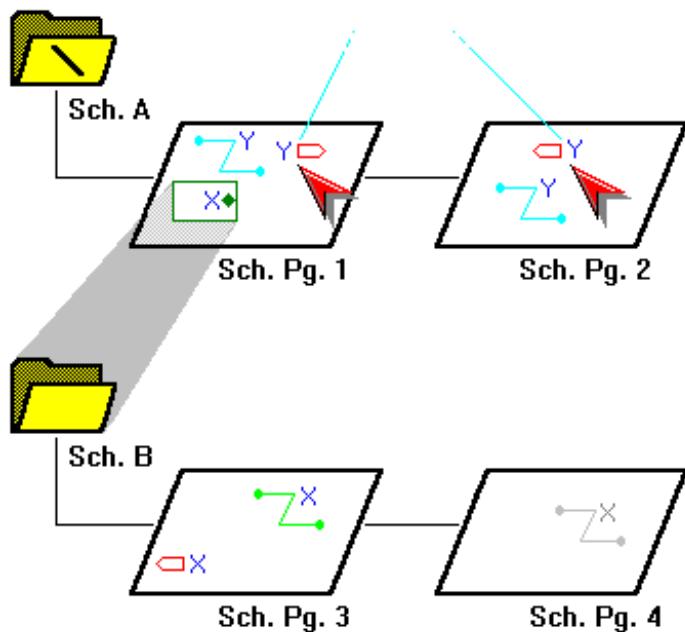
To establish the hierarchy with schematic folder A "above" schematic folder B:

1. Place a hierarchical block on schematic page 1.
2. Attach schematic folder B.

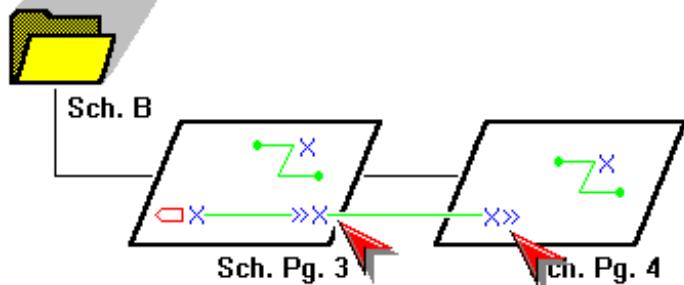
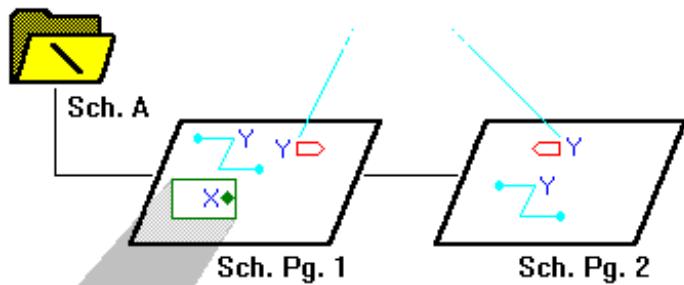


To carry a net between schematic folders A and B:

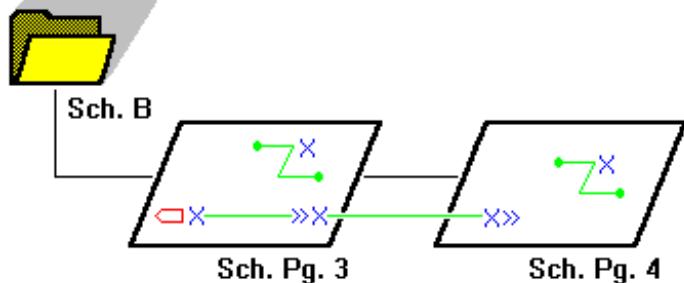
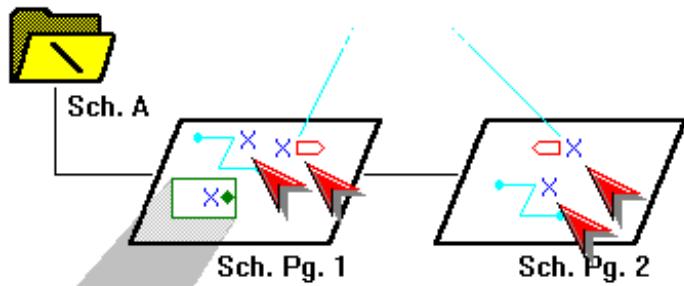
1. Select the hierarchical block on schematic page 1 and place a hierarchical pin named X inside it.
This hierarchical pin is a point of attachment for electrical connections between the hierarchical block and other objects on schematic page 1.
2. Place a hierarchical port named X on schematic page 3.
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.



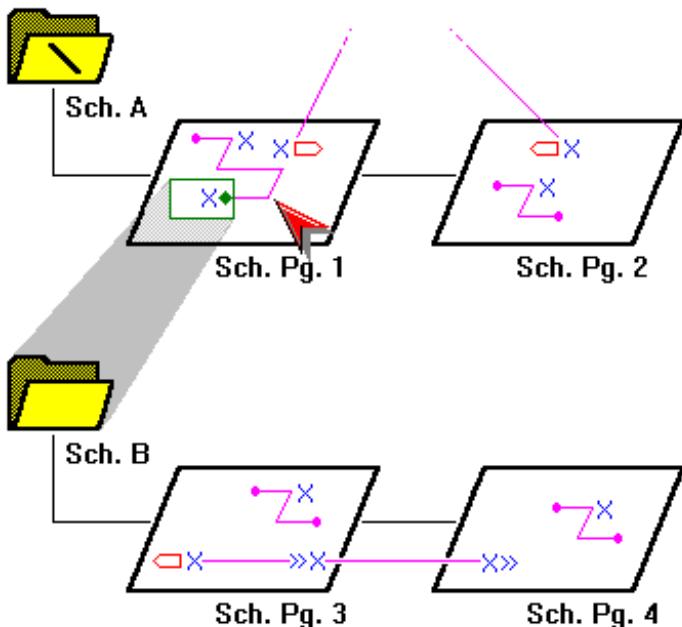
Hierarchical ports generally carry a net "up" through a hierarchy. In the root schematic folder, they usually represent external signals such as physical connectors on a PC board. Note that hierarchical ports in schematic folder A are electrically connected by name, so any like-named connectivity objects 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](#)
- [Guidelines for Creating Intersheet References](#)
- [Intersheet references in a flat design](#)
- [Intersheet references in a hierarchical design](#)

- [Reporting Intersheet References](#)
- [Signal Navigation in Capture](#)

Creating Intersheet references

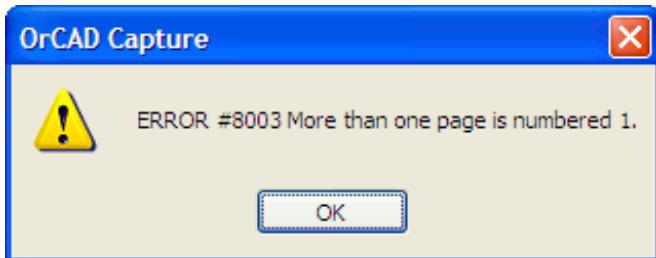
To add *intersheet references*

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 intersheet 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:

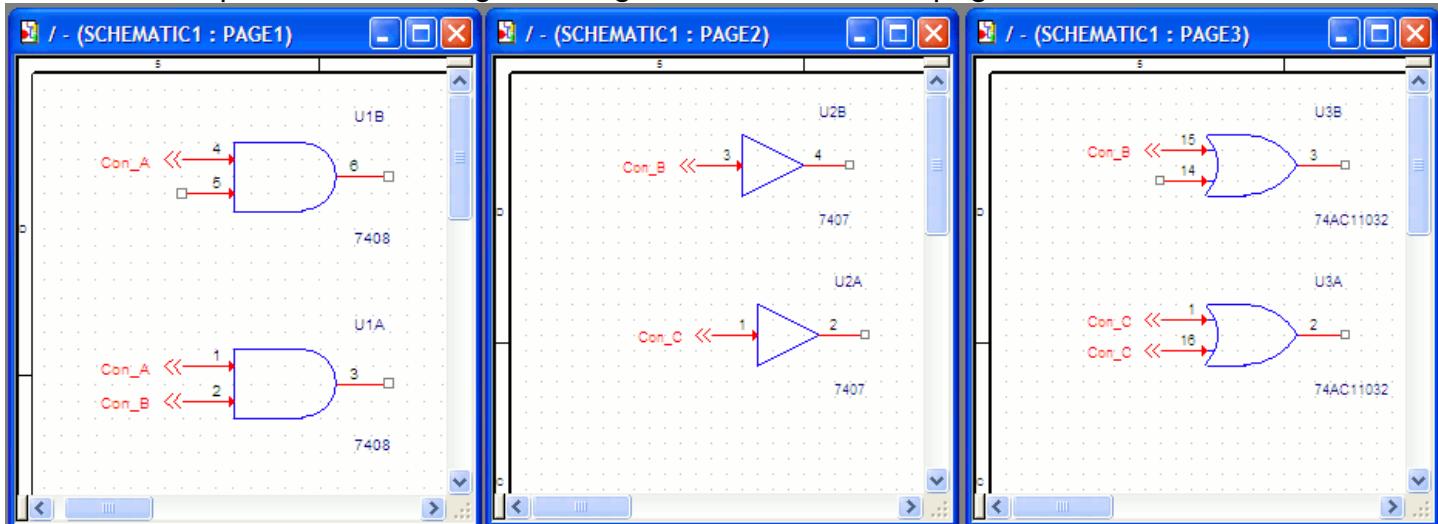
Scenario	IREF Generated
----------	----------------

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

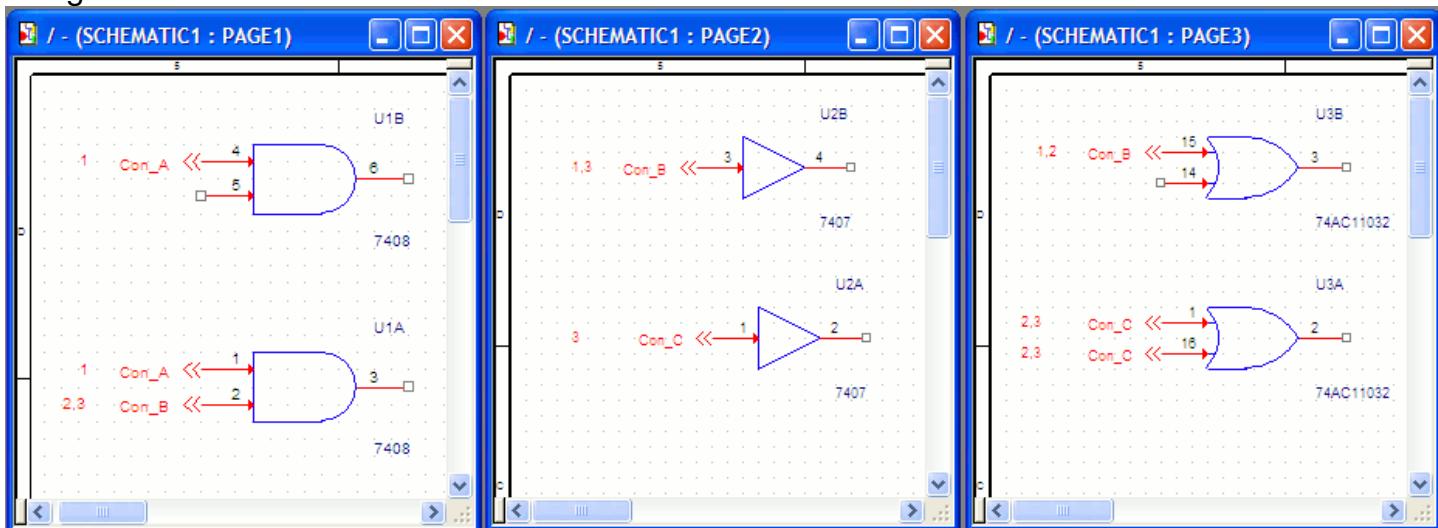
Intersheet references in a flat design

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

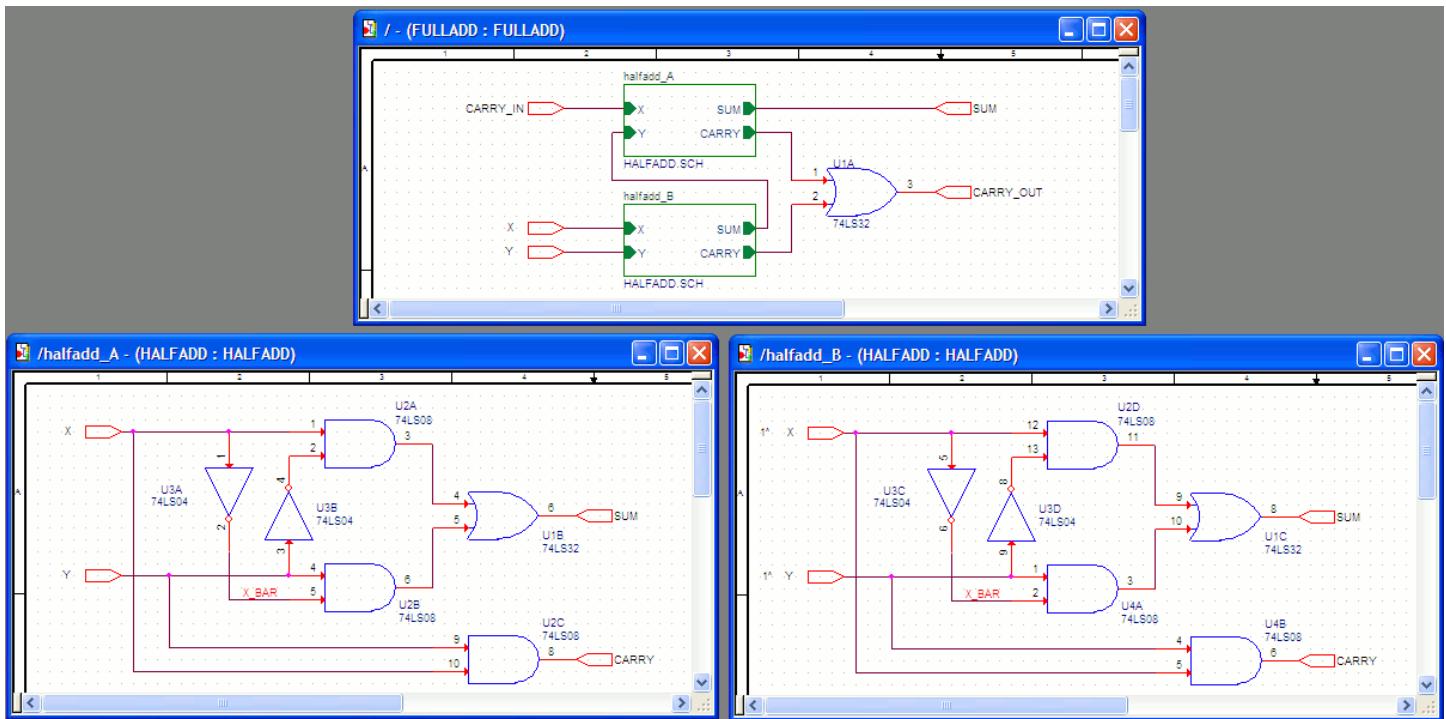
The page 2 reference for Con_B is 3 since the signals for this exist on page 3. Now the page 3 reference for both Con_C connectors on page 3 show as 2,3. This implies a signal exists on page 2. Also, another signal for this connector exists on page 3 itself.

⚠ 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.

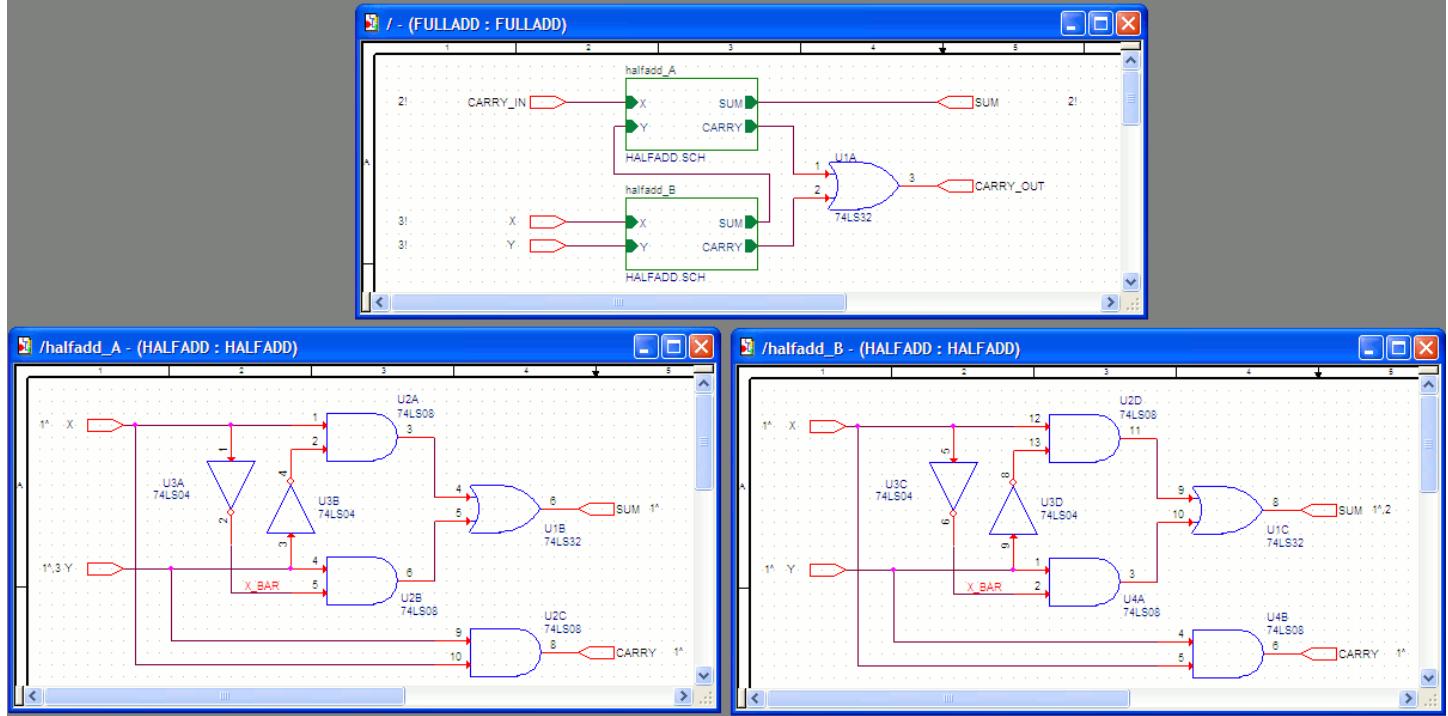
Intersheet references in a hierarchical design

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](#).

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 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.

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 the [Power 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 or 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](#)
- [Named NetGroup](#)
- [Unnamed NetGroup](#)
- [Components of a NetGroup Block](#)
- [Netlisting NetGroup Designs](#)
- [NetGroup Connectivity](#)

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 functionality. 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

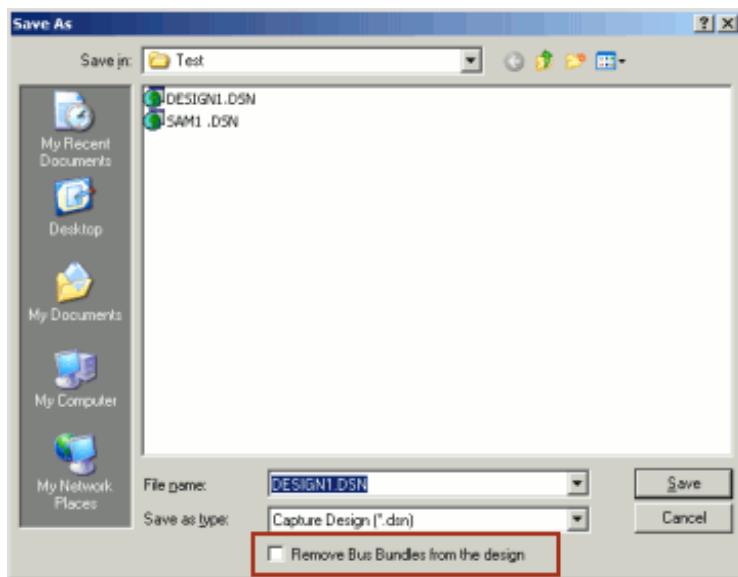
- Pin
- Global

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.

⚠ To synchronize the number of signals in the hierarchical pin and the NetGroup, choose *Tools – Sync NetGroups*.

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

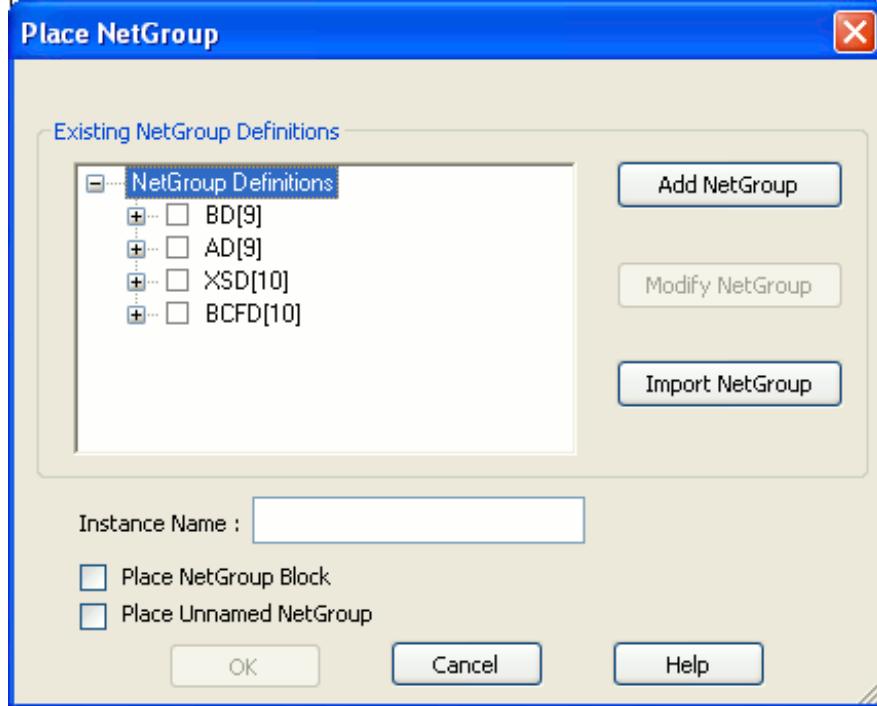
When you create a named NetGroup, you need to specify the associated NetGroup definition and then specify a name for the NetGroup. This type of NetGroup is persistent and can be instantiated across a design. The NetGroup can also be exported as a library and then instantiated and used in other designs. 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.

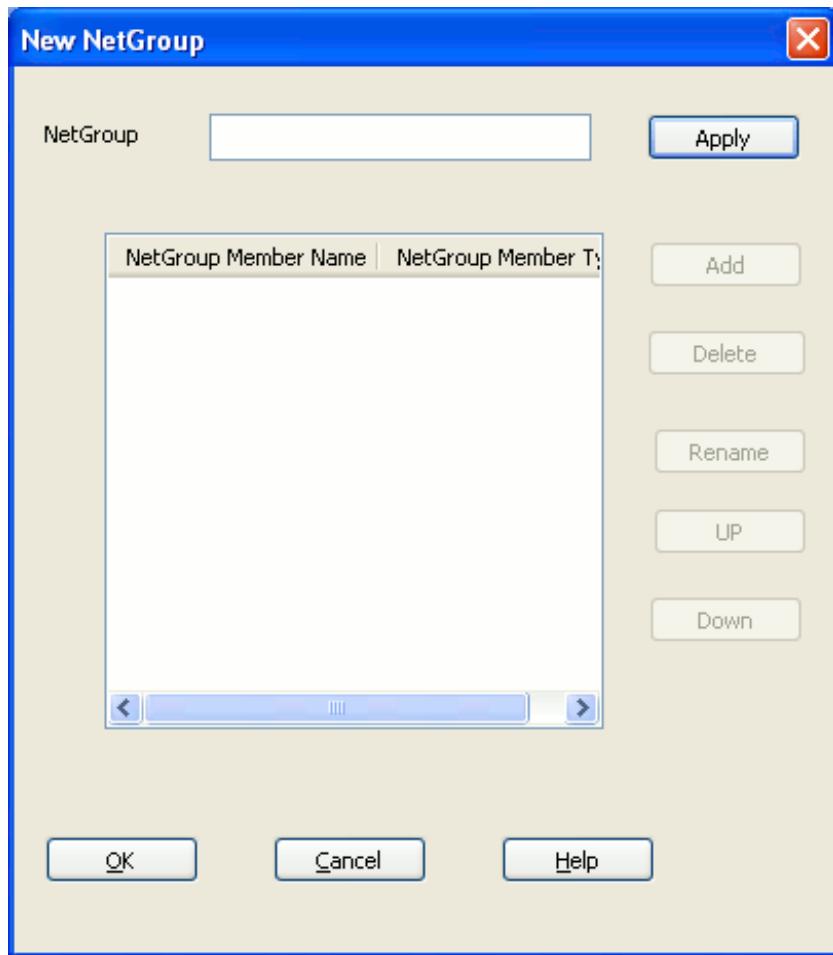
The Place / Create NetGroup dialog displays.

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[5-9]. 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 nomenclature [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 members. You can also **move** the existing members **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

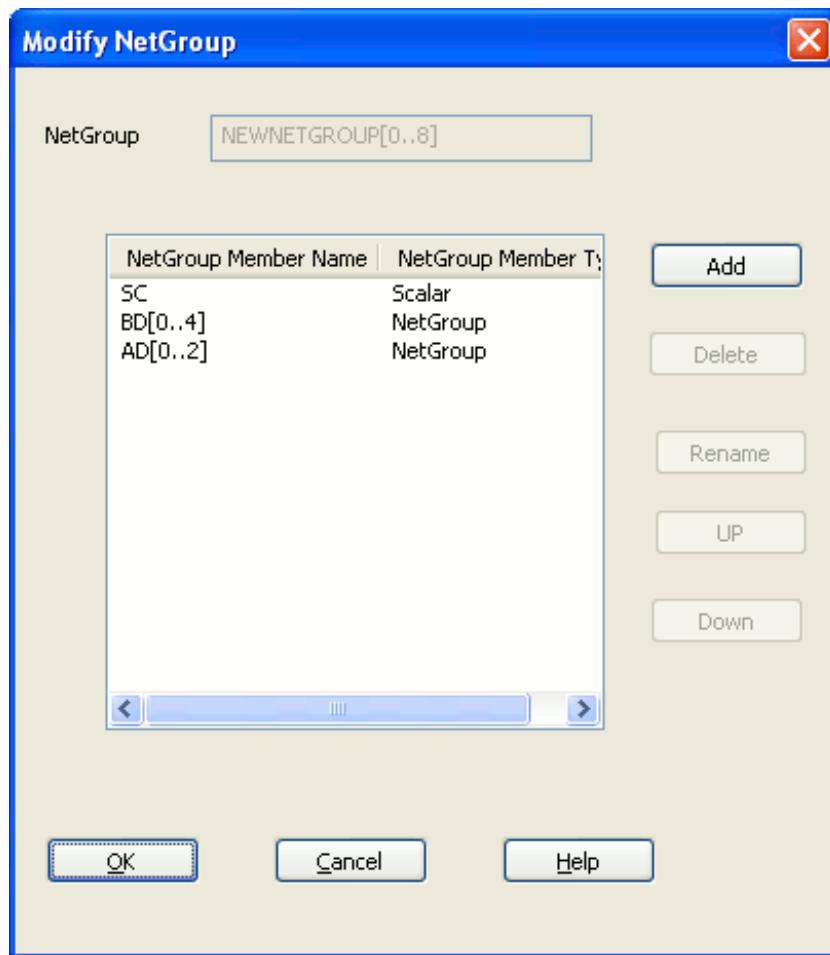
1. In the New NetGroup or Modify NetGroup dialog, select the NetGroup member to rename.
2. Click the Rename button.
The Rename NetGroup Member dialog displays.
3. Type a new name for the member and click OK.

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
- a hierarchical pin

 By default, only instance name is displayed for NetGroup Blocks, Hierarchical Ports, OffPage Connectors, Globals, and Hierarchical Pins. For Net Aliases, the NetGroup definition name is also displayed if there is a mismatch between the Instance name and the NetGroup definition name.

You can change the default settings in the NetGroup section of the Extended Preferences Setup window (*Options – Preferences – More Preferences*).

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.

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 box.
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 place a named NetGroup as a hierarchical pin

1. To place a named NetGroup as an hierarchical pin, you need go to the Place Hierarchical Pin dialog box.
To open this dialog box, choose *Place – Hierarchical Pin*.
2. Enter hierarchical pin's name and choose pin type from list of pin types.
3. Select the pin as bus, if hierarchical pin connects to a bus, or scalar, if the hierarchical pin connects to a wire.
4. Click User Properties button to edit pin's property.
5. To place the hierarchical pin as a NetGroup pin, choose the NetGroup Pin check box.
6. Select the required NetGroup from the drop-down list of NetGroups.
7. Click *OK* and place the NetGroup Pin in the hierarchical block.

 Sync up and down of NetGroup pins is supported in hierarchical blocks.

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.

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**.

 The NetGroup port displays the NetGroup instance name and the size of the NetGroup instance.

 The Netgroup entry pin type should be output type only.

NetGroup Wire

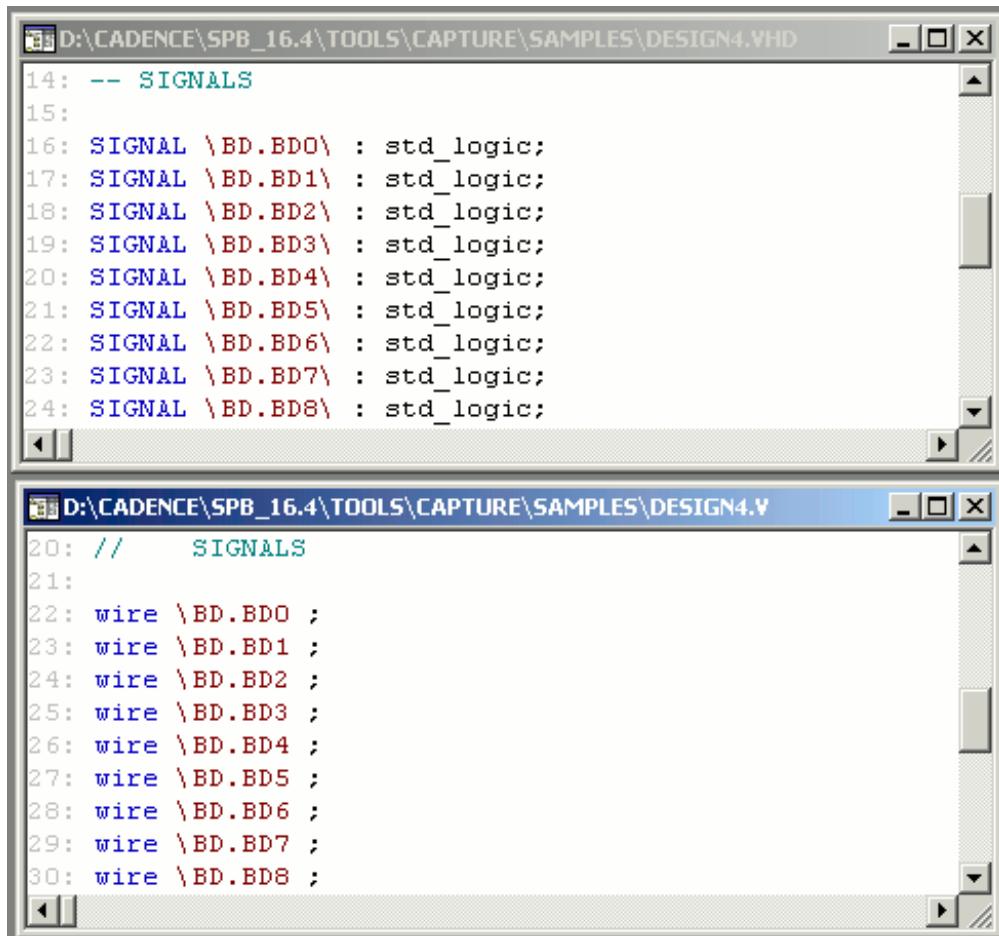
When you connect the entry / exit pins of the NetGroup to a wire, you are effectively connecting all the signals that constitute the NetGroup into one wire. This wire is known as the **NetGroup wire**.

Netlisting NetGroup Designs

When you netlist a design, the signals in the design are mapped on the output netlist. A NetGroup is a heterogeneous 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.



The image shows two side-by-side windows of the OrCAD Capture software interface. Both windows have a title bar indicating they are running on 'D:\CADENCE\SPB_16.4\TOOLS\CAPTURE\SAMPLES\DESIGN4.VHD'.

The left window displays VHDL code:

```
14: -- SIGNALS
15:
16: SIGNAL \BD.BD0\ : std_logic;
17: SIGNAL \BD.BD1\ : std_logic;
18: SIGNAL \BD.BD2\ : std_logic;
19: SIGNAL \BD.BD3\ : std_logic;
20: SIGNAL \BD.BD4\ : std_logic;
21: SIGNAL \BD.BD5\ : std_logic;
22: SIGNAL \BD.BD6\ : std_logic;
23: SIGNAL \BD.BD7\ : std_logic;
24: SIGNAL \BD.BD8\ : std_logic;
```

The right window displays Verilog code:

```
20: // SIGNALS
21:
22: wire \BD.BD0 ;
23: wire \BD.BD1 ;
24: wire \BD.BD2 ;
25: wire \BD.BD3 ;
26: wire \BD.BD4 ;
27: wire \BD.BD5 ;
28: wire \BD.BD6 ;
29: wire \BD.BD7 ;
30: wire \BD.BD8 ;
```

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 not 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)
orOhdInet.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 specify 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](#)

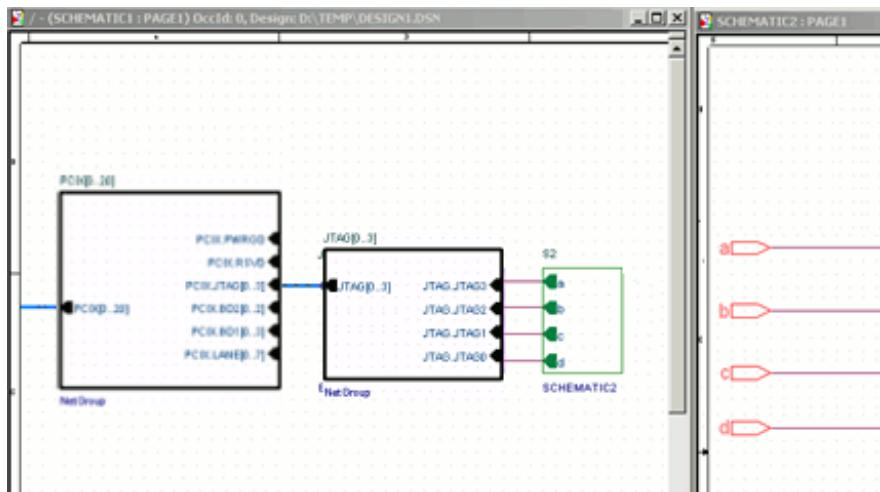
- Un-named NetGroup connectivity

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



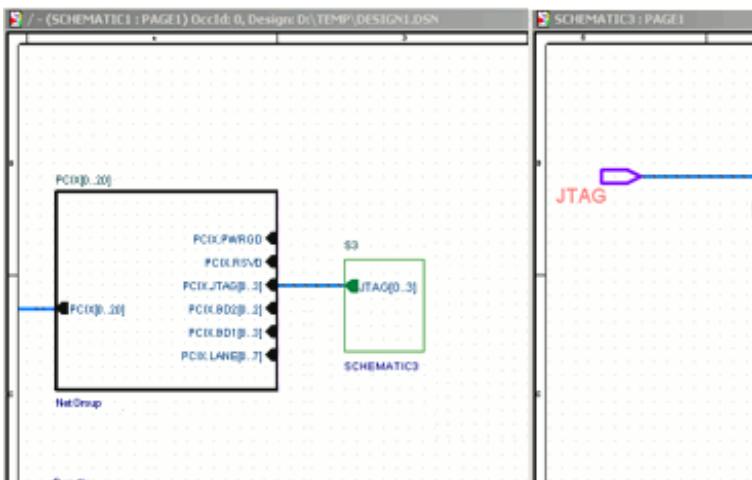
In this example, the signals at the top level (SCHEMATIC1:PAGE1) of the design need to be connected to signals on the page SCHEMATIC2:PAGE1 at lower levels of the hierarchy.

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

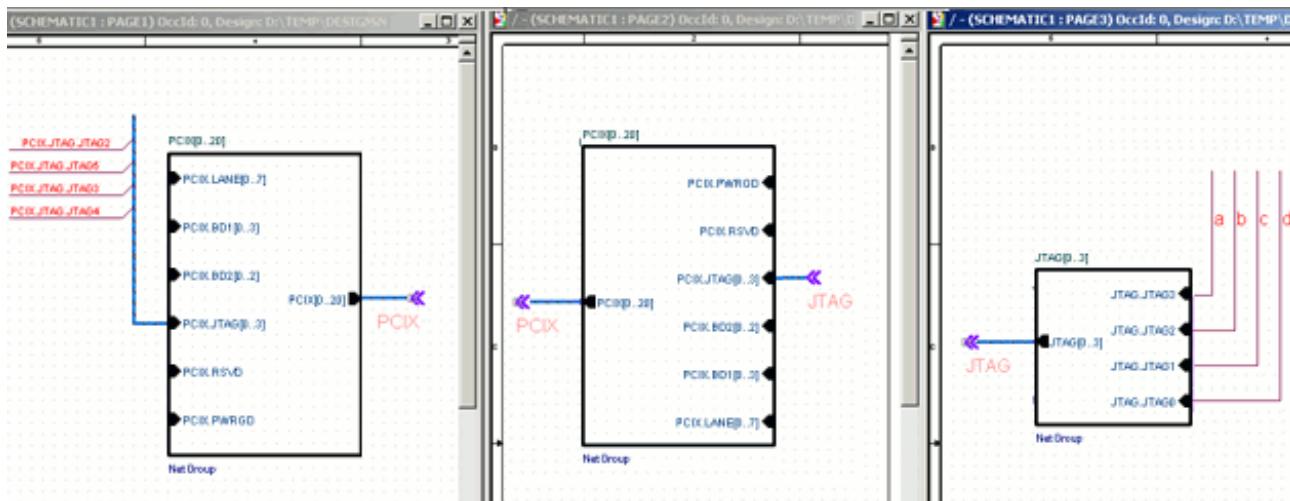
In this example, the four signals, a, b, c & d are connected to signals on SCHEMATIC3:PAGE1 using a NetGroup off-page connector.



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.



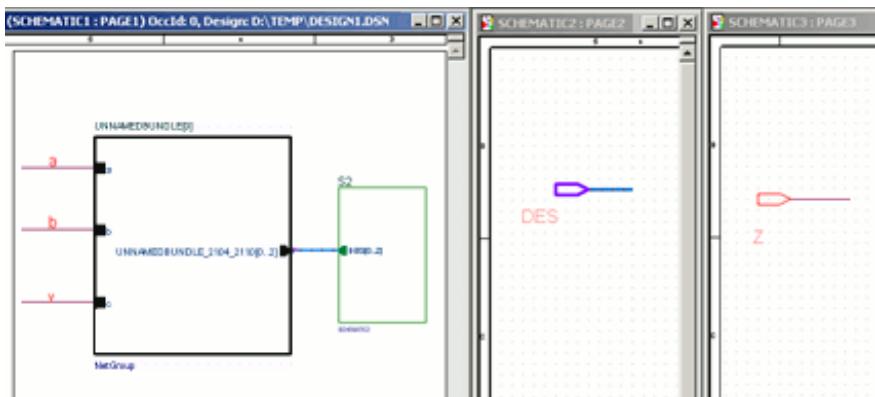
1. The signals on SCHEMATIC1:PAGE1 are first tapped into a large NetGroup, PCIX, containing 19 signals. This associated NetGroup definition contains all the signals of this page that need to be connected to pages across the design.
Four signals need to be connected from PAGE1 to PAGE3 on SCHEMATIC1.
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.

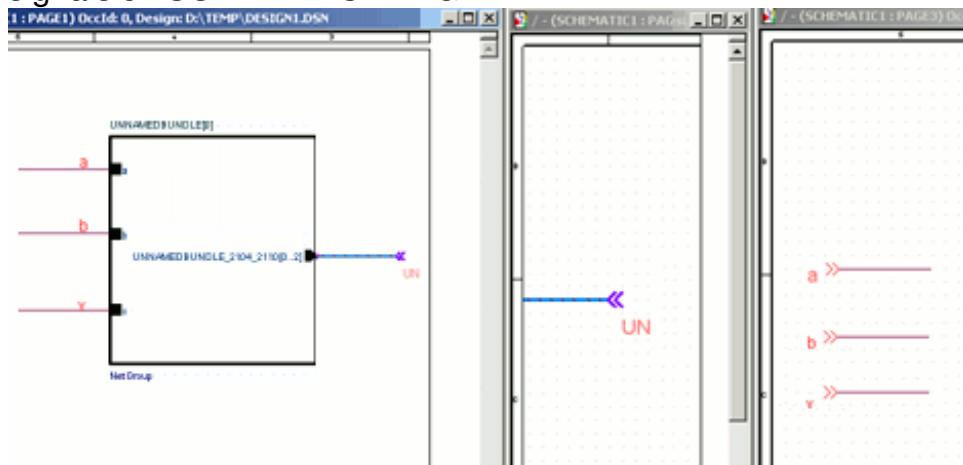


1. An unnamed NetGroup is first placed on SCHEMATIC1:PAGE1.
This is an empty associated NetGroup definition so we still need to add members to this NetGroup.
2. Specify three scalar members for the NetGroup. These will be used as entry points from the signals a, b, and y to the unnamed NetGroup. To this, 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](#)

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 (different width)	NetGroup	Higher width NetGroup	Winning Bus defines flat nets	associated NetGroup definition should match winning bus

EXAMPLE <p>NetGroup S[0..5] - Members: S.JTAG0, S.JTAG1 ,S.JTAG2 ,S.JTAG3, S.JTAG4, S.JTAG5</p> <p>NetGroup B[0:2] - Members: B.B0, B.B1, B.B2</p>				
	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: <ul style="list-style-type: none"> • 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 (different widths)	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 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

Working with Capture TCL

OrCAD X 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 logged command is registered with a TCL interpreter. When the command is played back, Capture uses the TCL interpreter to retrieve the command and run it in Capture. However, this process is completely abstracted from Capture. This makes logging and replaying of a set of commands an intuitive and simple task.

The user resources are better utilized to identify the type of script (set of steps) that is required for a specific task.

The following topics are covered in this section:

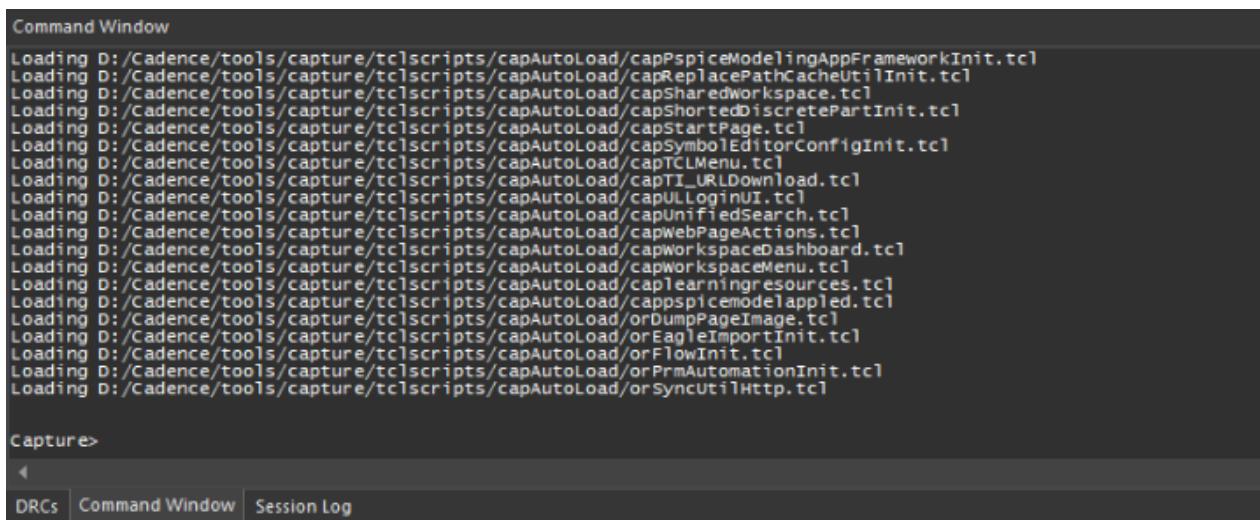
- [Using the Capture Command Window](#)
- [Creating a Capture TCL Script](#)
- [Replaying a Capture TCL Script](#)

Using the Capture Command Window

You use the Capture Command window to run a TCL command. When you perform any 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, right-click the Capture menu bar and choose *Command Window*. Alternatively, choose *View – Command Window* from the main menu.

The Command window displays with the capture> prompt.



The screenshot shows the 'Command Window' tab selected in the bottom navigation bar. The window displays a list of files being loaded, primarily from the path D:/Cadence/tools/capture/tclscripts/capAutoLoad. The files include various initialization scripts for modeling, replacement paths, shared workspace, discrete parts, start pages, symbol editors, and various user interface components like URLs, logins, unified search, and web page actions. The list ends with scripts for learning resources, model applied, dump page image, eagle import, flow init, prism automation init, and sync util http.

```
Command Window
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capPspiceModelingAppFrameworkInit.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capReplacePathCacheUtilInit.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capSharedworkspace.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capShortedDiscretePartInit.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capStartPage.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capSymbolEditorConfigInit.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capTCLMenu.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capTCLURLDownload.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capULLoginUI.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capUnifiedSearch.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capWebPageActions.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capworkspaceDashboard.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/capworkspaceMenu.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/caplearningresources.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/cappspicemodeled.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/orDumpPageImage.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/orEagleImportInit.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/orFlowInit.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/orPrmAutomationInit.tcl
Loading D:/Cadence/tools/capture/tclscripts/capAutoLoad/orSyncUtilHttp.tcl

Capture>

```

DRCs | Command Window | Session Log

⚠ Any operation you perform in Capture is registered with the TCL interpreter and logged by Capture even if the Command window is closed.

Registering and Logging TCL Commands

When you perform any 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 TCL 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 deselect all the objects on a schematic page, the associated TCL commands are:

SelectAll

UnSelectAll

Similarly, the following command places 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 commands for operations performed in the project manager. For example, the following command selects 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 is logged in the Capture Command window.

Running a TCL Command

To run a TCL Command in the Command window, type the command in the Command window and press **Enter**. If an error occurs while running the command, the error message is displayed in the Command window.

 All TCL commands are case-sensitive.

Besides the Capture TCL commands, you can also run 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 the command to change the working directory:

```
cd
```

Creating a Capture TCL Script

You can create a Capture TCL script manually in any text editor. Another way is to run a series of steps in Capture and record the associated commands. When you 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.

By default, the TCL logging feature is disabled in Capture. To log each command that you run in Capture to a TCL script file, you need to run the SetOptionBool command in the Capture Command window:

```
SetOptionBool Journaling TRUE
```

If you are in the TCL command logging mode, you can view each Capture command that is run 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, all the TCL commands for the corresponding Capture commands are displayed in the Session Log window. However, in some cases one Capture command might correspond 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 window. If a Capture command corresponds to a number of TCL commands, this may cause a delay in completing the Capture command. To avoid this, you can turn off the display functionality using the `SetOptionBool` command with the `FALSE` argument:

```
SetOptionBool DisplayCommands FALSE
```

By default, the display of TCL commands is disabled in the Command window.

Location of the TCL File

If you enable TCL logging in Capture, a TCL file is created in the `Temp\CAPTURELOG` directory for each Capture session.

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 all the support files required to run the script.

Capture Dialog settings in TCL

When you open a dialog box from a menu, such as *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 TCL file for the current session of Capture.

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_Oct_11_14_34_45_2023/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 TCL file for the current Capture session.

 The location of the `CAPTURELOG` directory in the above example is `C:\Temp`. However, this may change depending on the conditions described in [Location of the TCL File](#).

TCL Script for Specific Capture Commands

When you turn on TCL script logging, the TCL commands for all the Capture commands that you run 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, do the following:

1. Right-click in the Capture Command window and choose *Clear All*.
2. Perform the steps 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 box, 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 run these commands are not saved along with this script. For example, the dialog settings that are saved to an XML are not 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](#).

Replaying a Capture TCL Script

After creating a Capture TCL script (for details see [Creating a Capture TCL Script](#)), you can replay the script to run the set of commands defined in the script. You can replay the recorded TCL script from the Capture Command window or from the Windows Command line.

- [Running TCL Script in Capture](#)

- [Running TCL Script in Capture](#)
- Running TCL Script in Windows

Running TCL Script in Capture

You can run a TCL script directly from the Capture Command window using the source command.

```
source <Script Path and Name>
```

For example, to run 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 the 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 window. 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.

Running TCL Script in Windows

You can run a TCL script directly from the Windows command line, without explicitly launching Capture to play the script.

For this, you need to pass the script name and path as a command line argument to Capture. This ensures that Capture opens and all the steps defined in the script are done.

To run the script from the command line:

```
capture <Script Path and Name>
```

 After the script is run, it opens as a text file in the edit mode in Capture.

The path separator rules to run TCL script from the Windows command line are the same as the rules for running a script from the Capture Command window. For details see [Running 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.

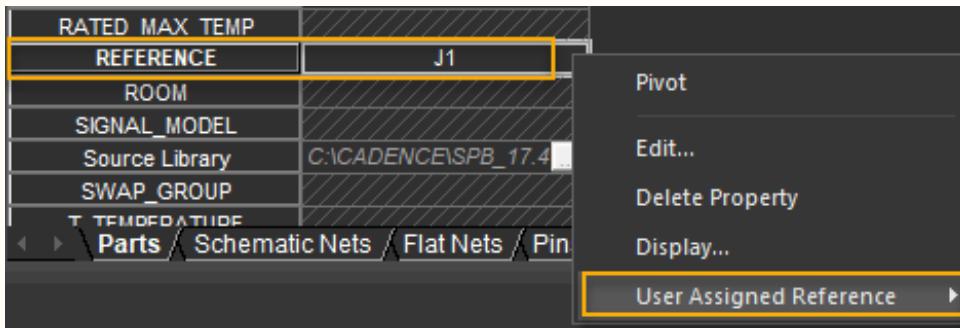
Design Annotation

Annotating a design involves assigning reference designators and net names to unnamed parts and electrical connections in your design. By annotating your design, you provide the means by which to pass it downstream to other layout design tools, such as PCB Editor and OrCAD X Presto, that take it beyond the schematic creation phase of the design. Some of the downstream applications include PCB layout tools, simulators, and logic optimizers.

Annotating a design includes the following tasks:

- [Customizing Part References in a Design](#)
- [Backannotating](#)
- [Selecting Annotation Sequence](#)
- [Swap File](#)
- [Designating Pins, Gates, or Packages for Swapping](#)
- [Update File](#)
- [Creating a Combined Swap and Update File](#)
- [Advanced Annotation](#)

- ⚠** If you want to preserve a reference designator during annotation, right-click a part on the schematic page and choose *User Assigned Flag – Set*. Alternatively, in the *Property Editor* window, right click the value for the *Reference* property and choose *User Assigned Flag – Set*. The User Assigned Flag is set by the tool if the reference designator is changed from Property Editor or Schematic Editor or through Backannotation.



If the instance and occurrence values are the same in a hierarchical design, this command cannot set the flag for an occurrence value. If you want to set the User Assigned Flag for the occurrence value, edit the Reference value and then set the flag.

- ⚠** If you place different sections of a homogenous part, such as U1A, U1B, U1C in a design, and run annotation with *Reset part references to ?* option selected and *Preserve designator* option unselected, the designator information is lost in the design. To preserve designator information of a design, select *Preserve Designator* option in the *Annotate* dialog box before running annotation.

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 uses to annotate a schematic page or a hierarchical block in your design. Use the *Annotate* dialog box to complete this task.

i This functionality works independently of the existing annotation behavior of Capture.

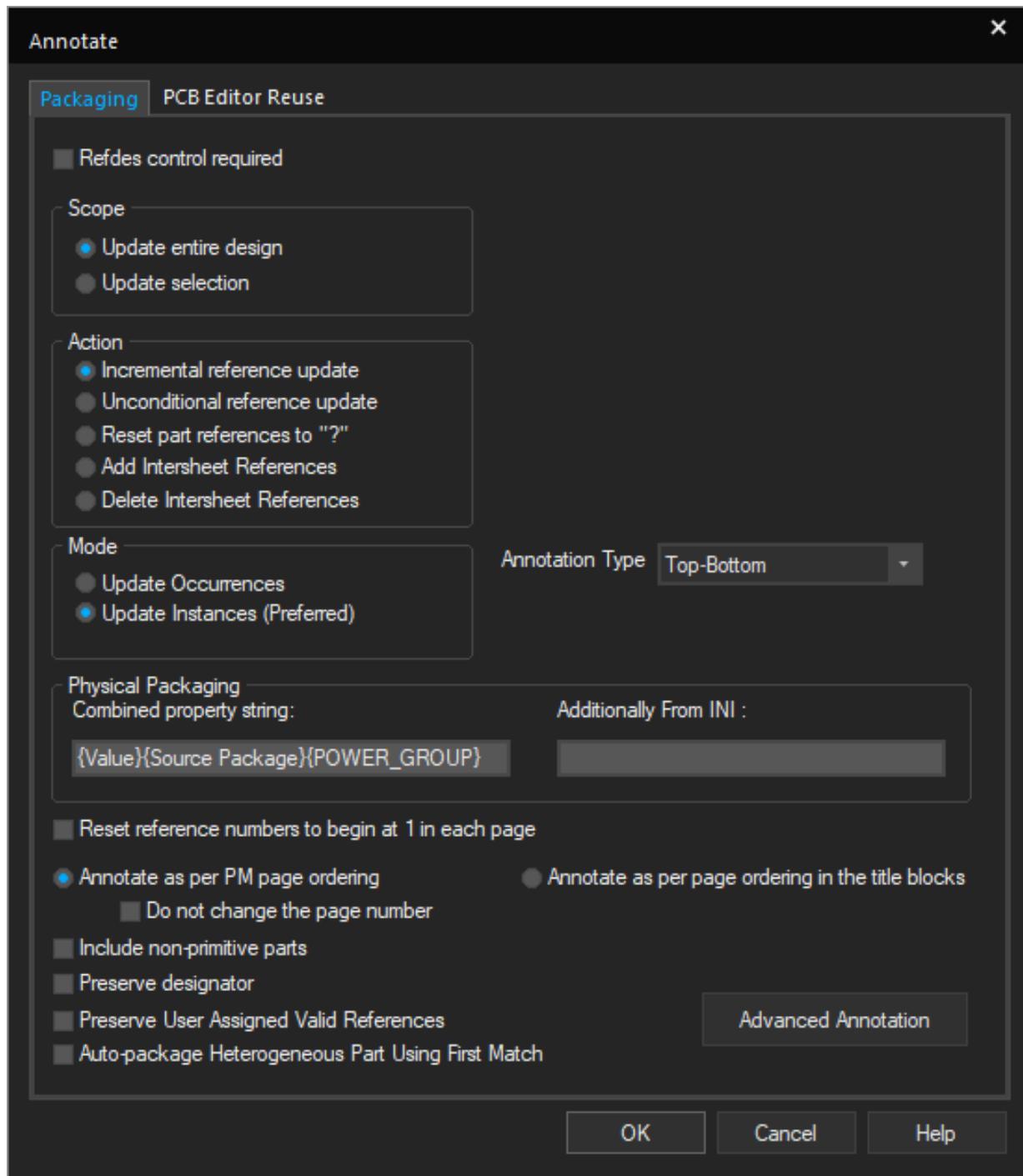
To customize part references in a design using either schematic page-wise or hierarchical block-wise annotation, do the following:

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 need to specify a part reference range for all the schematic pages or hierarchical blocks in your design.

2. Choose the *Tools – Annotate* menu command.

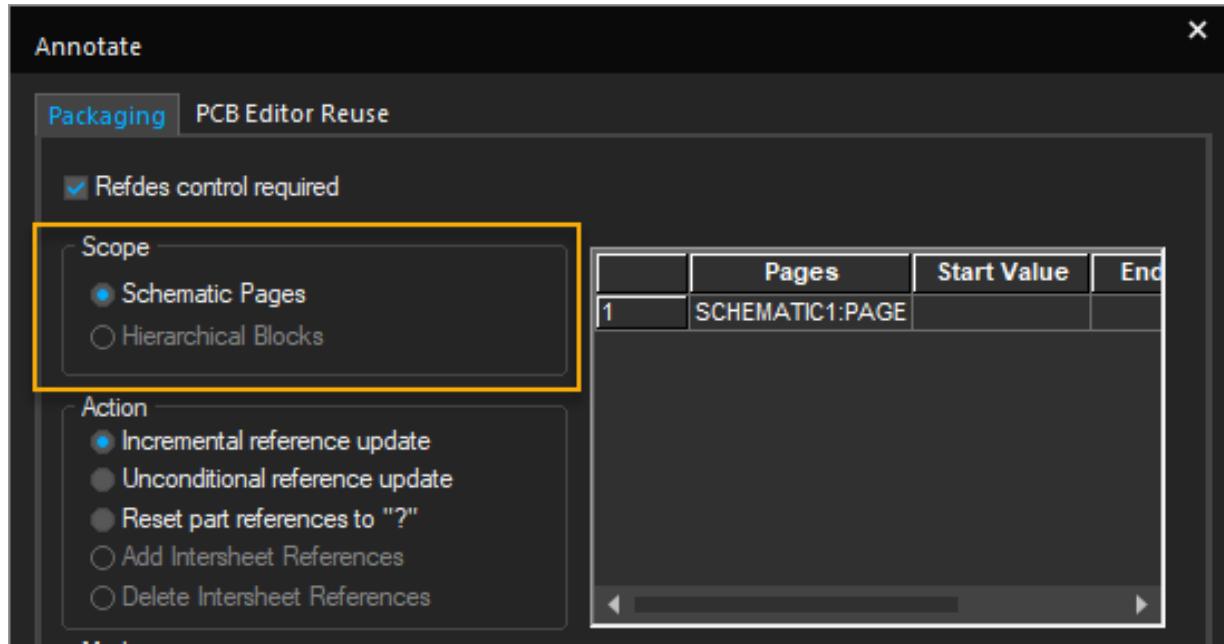
The *Annotate* dialog box appears.



 To perform regular annotation, see [Uniquely Identifying Parts](#).

3. Select the *Refdes control required* check box to specify a part reference range for each schematic page or a hierarchical block in your design.
The *Scope* options in the dialog box change 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.



Schematic Page-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*.

Hierarchical Block-Wise Annotation

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:

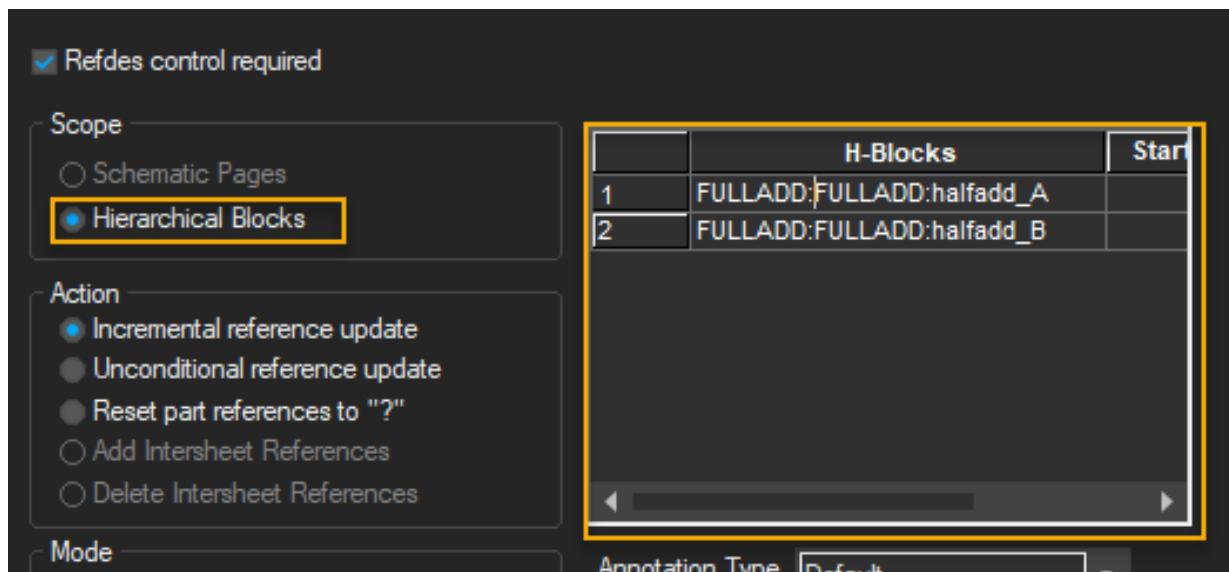
Schematic_Folder_Name:Hierarchical_Block_Name.

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 *Tutorial.dsn* sample design, which is a flat design, contains a schematic page defined at the root level of the design, *SCHEMATIC1*. When you select the Refdes control required check box, the grid displays the following entry in the Pages column: *SCHEMATIC1:PAGE1*, where *SCHEMATIC1* is the root schematic folder name and

PAGE1 is the schematic page name.

Consider the example of a hierarchical design, *FULLADD.DSN*, which contains two hierarchical blocks defined at the root level of the design. When you select the *Refdes control required* check box, 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.

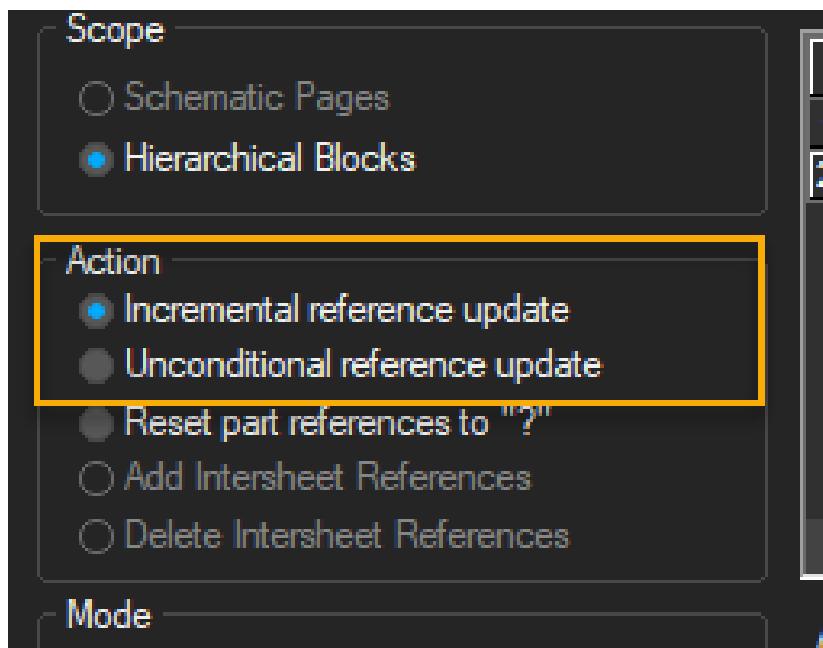
- Enter a numeric value greater than 0 in the Start Value and End Value columns corresponding to each schematic page name or the hierarchical block name.

- 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 *End Value* must be greater than the *Start Value*.

5. Specify all the other desired settings in the *Annotation* dialog box.

If you set the design for schematic page-wise or hierarchical block-wise annotation and also use the:

- *Incremental reference update* option, all the part references in your design are updated incrementally within the specified part reference range. However, this does not affect the existing annotated parts in your design.
- *Unconditional reference update* option, all the part references in your design are updated unconditionally from the start value specified in the part reference range.



Additionally, Capture flashes error and warning messages if it encounters any invalid operation while using this functionality. For example, the 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

6. Click *OK*.

Part references in each schematic page or hierarchical block are updated according to the range specified for them in the *Annotation* dialog box.



- Capture saves the annotation preferences for the project when you click *OK*. These settings are used whenever the project is opened next.
- If you are using the *Refdes control required* option for a project, the Auto reference placed part option in the *Miscellaneous* tab of the *Preferences* dialog box does not honor the range specified in the grid.
- If any changes are made to the root schematic of a design, all the part reference range values specified in the grid for the design are lost. However, this does not affect the existing annotated parts in the design.

Backannotating

You use backannotation to transfer packaging information to your schematic folder from other EDA tools. Using the *Tools – Back Annotate* command, you can import changes created by external tools, such as PCB layout editors. To backannotate properties, you use the *Update Properties* tool. 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 backannotation file, edit the file to match the format described in [Designating pins, gates, or packages for swapping](#).

Example

You might backannotate a design. After you complete your schematic design and while you are routing a printed circuit board, you realize that you can greatly reduce the via count, track length, or routing complexity by exchanging two of the gates or pins on a part. You 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 swapped places.

Backannotating board file information to your schematic design involves creating a report file and reading it back into Capture

To backannotate schematic information, do the following:

1. After making changes to the design in the layout tool, choose the *File – Reports* menu command.

The *Generate Reports* dialog box appears.

2. If you re-annotated the names of parts, or altered parts or nets, choose OrCAD backannotation file (.swp) to create a combined swap and update file.
3. Click *OK* to create the report.
4. In Capture, from the project manager, choose *Tools – Back Annotate*.
The *Backannotate* dialog box appears.
5. Use the *Browse* button to find the file (.swp) you created in step 2, and click *OK*.

Capture updates the schematic design.

To backannotate part packaging information, do the following:

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 to process only a portion of the design. To process the entire design, leave the schematic folders or schematic pages unselected.
3. Choose *Tools – Back Annotate*.
The [Back Annotate dialog box](#) displays.
4. In the *PCB Editor* tab, verify that the required options are set.
For example, check the name and location of the netlist directory and the swap file.
5. Click *OK*.

Shortcut



Related Topics

[Processing Changes after Board Creation \(ECO\)](#)

Selecting Annotation Sequence

You can select the sequence in which the components of your design are annotated.

To select the annotation sequence, do the following:

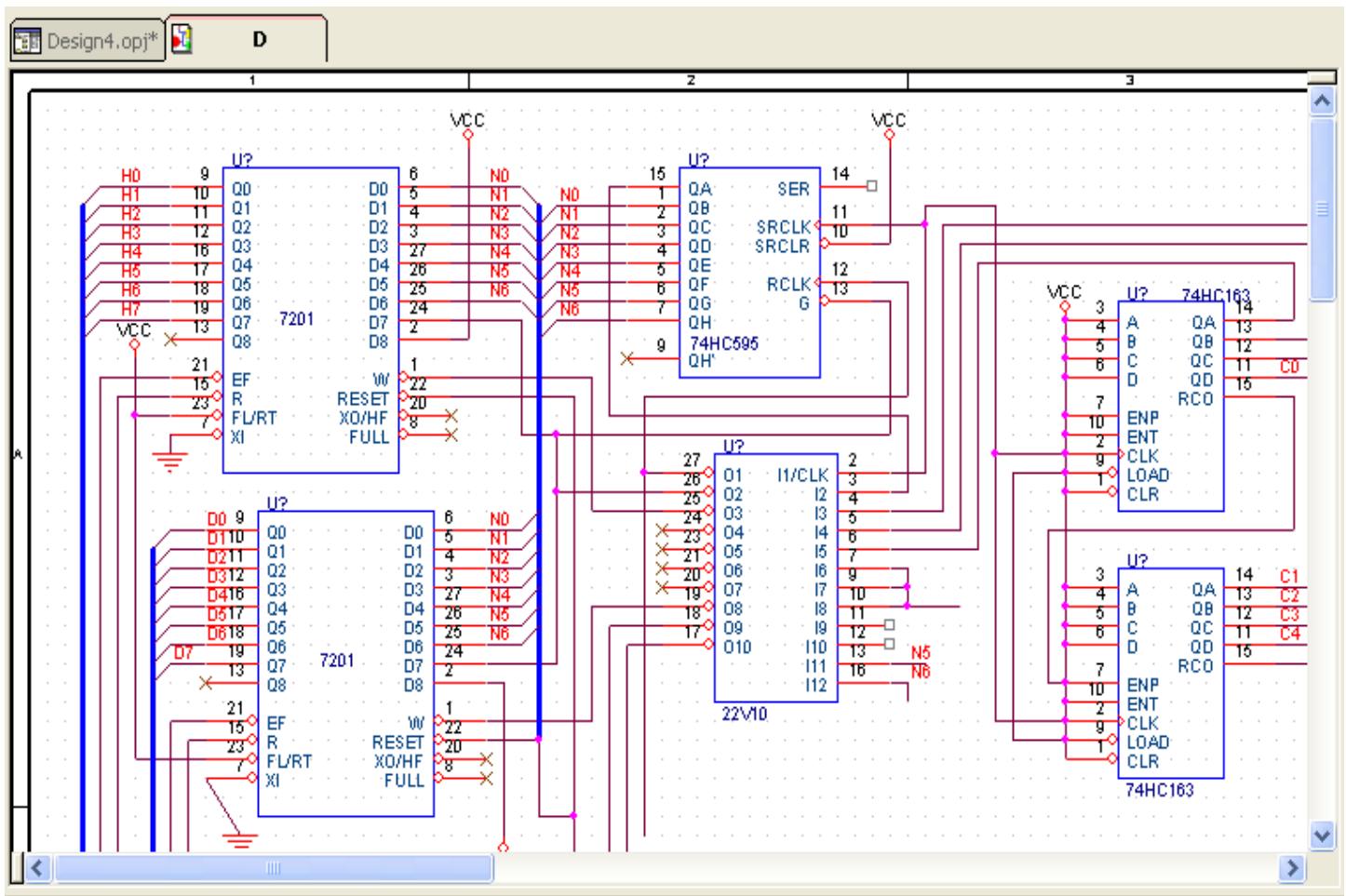
1. Choose *Tools – Annotate* from the main menu.
2. In the *Annotate* dialog box, select the *Annotation Type* drop-down button.
The *Annotation Type* list contains three options that you can use to decide the sequence in which the objects on your design are annotated:



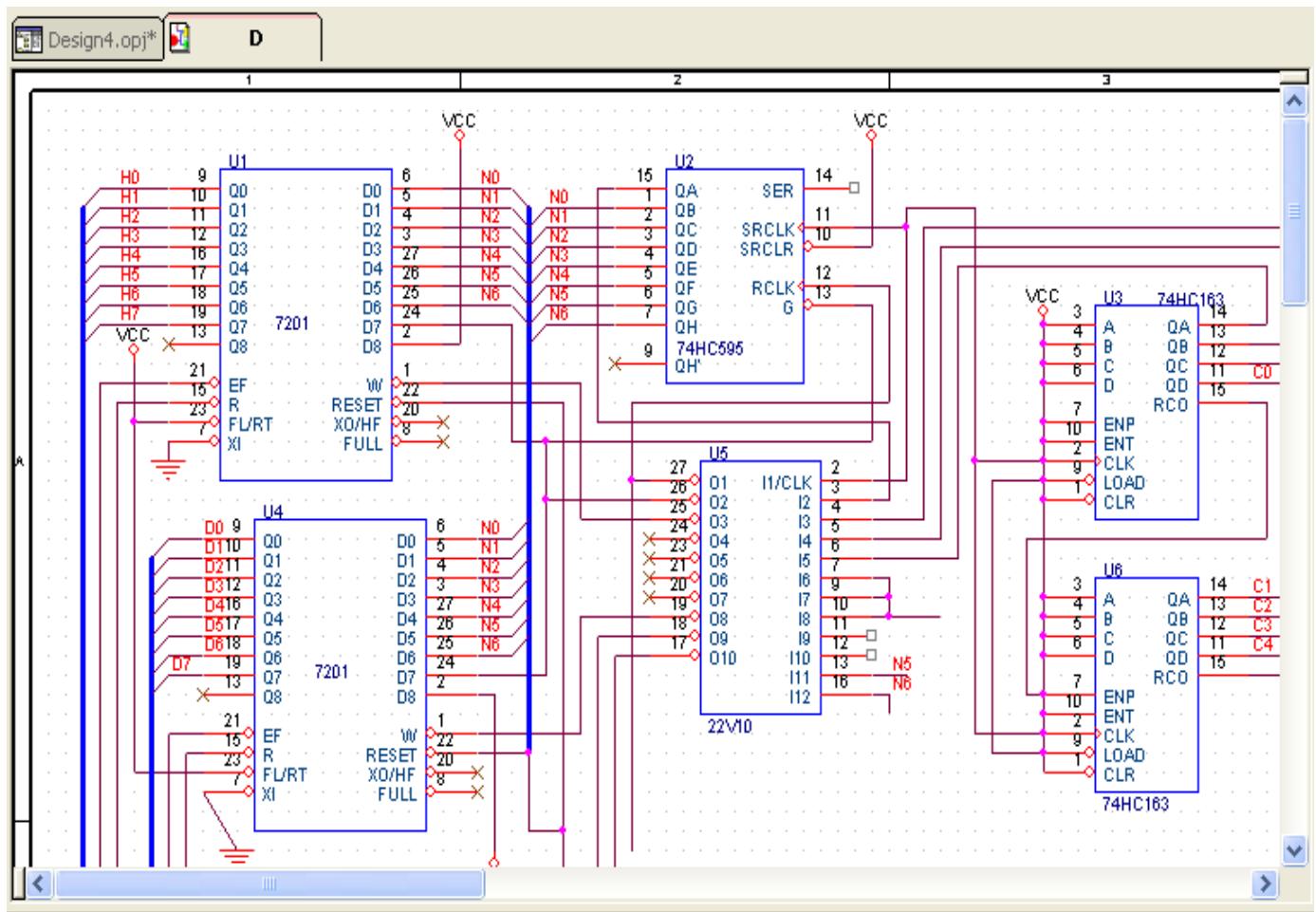
3. Select the desired option from the list.

Example

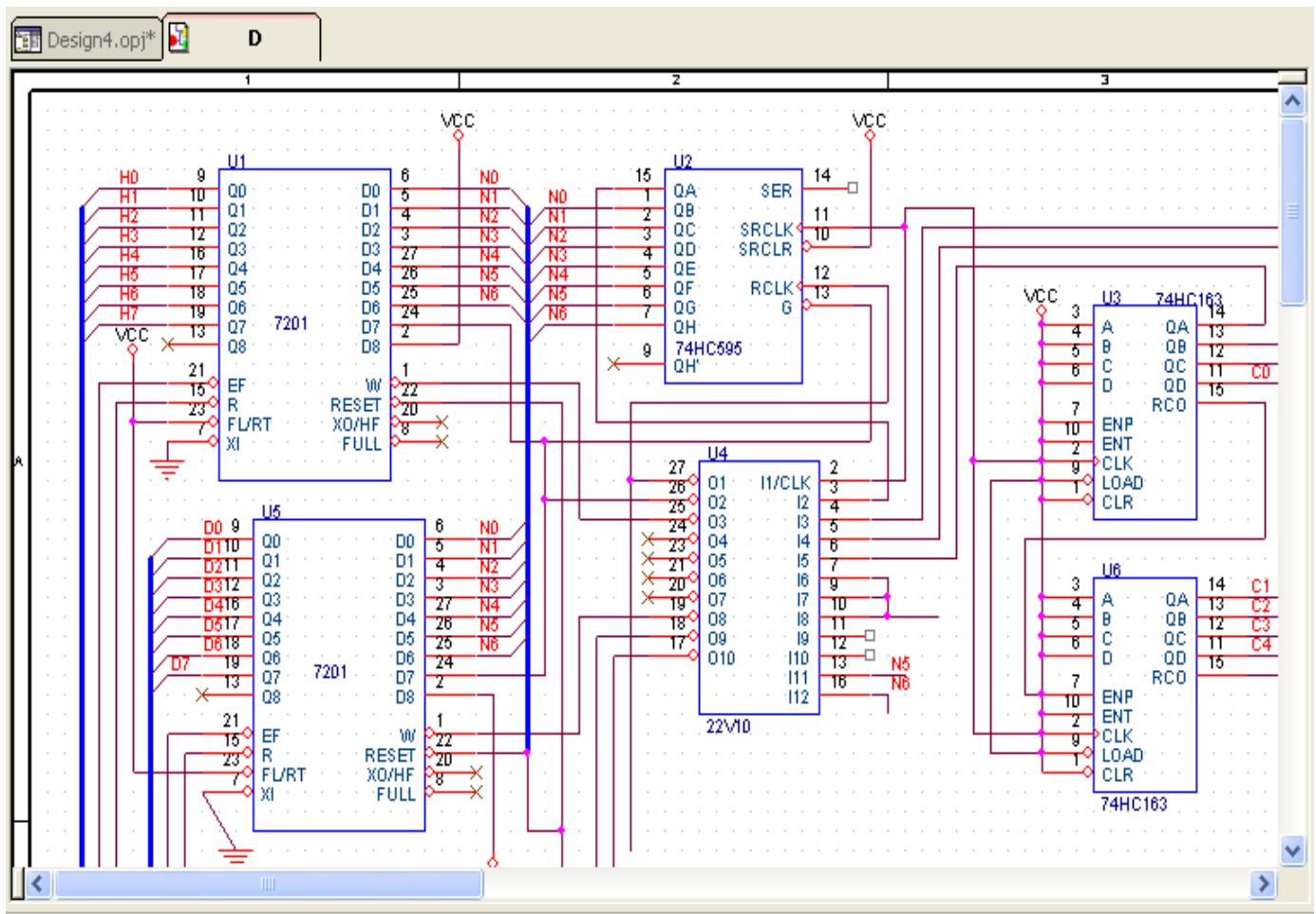
To differentiate between the way these sequences work on a design, see the following example of a dense design that contains a number of large components clustered together. Notice the sequence of annotation of the components on the design in each case.



- Annotating the design with the *Default Annotation Type* option selected leads to the following results:

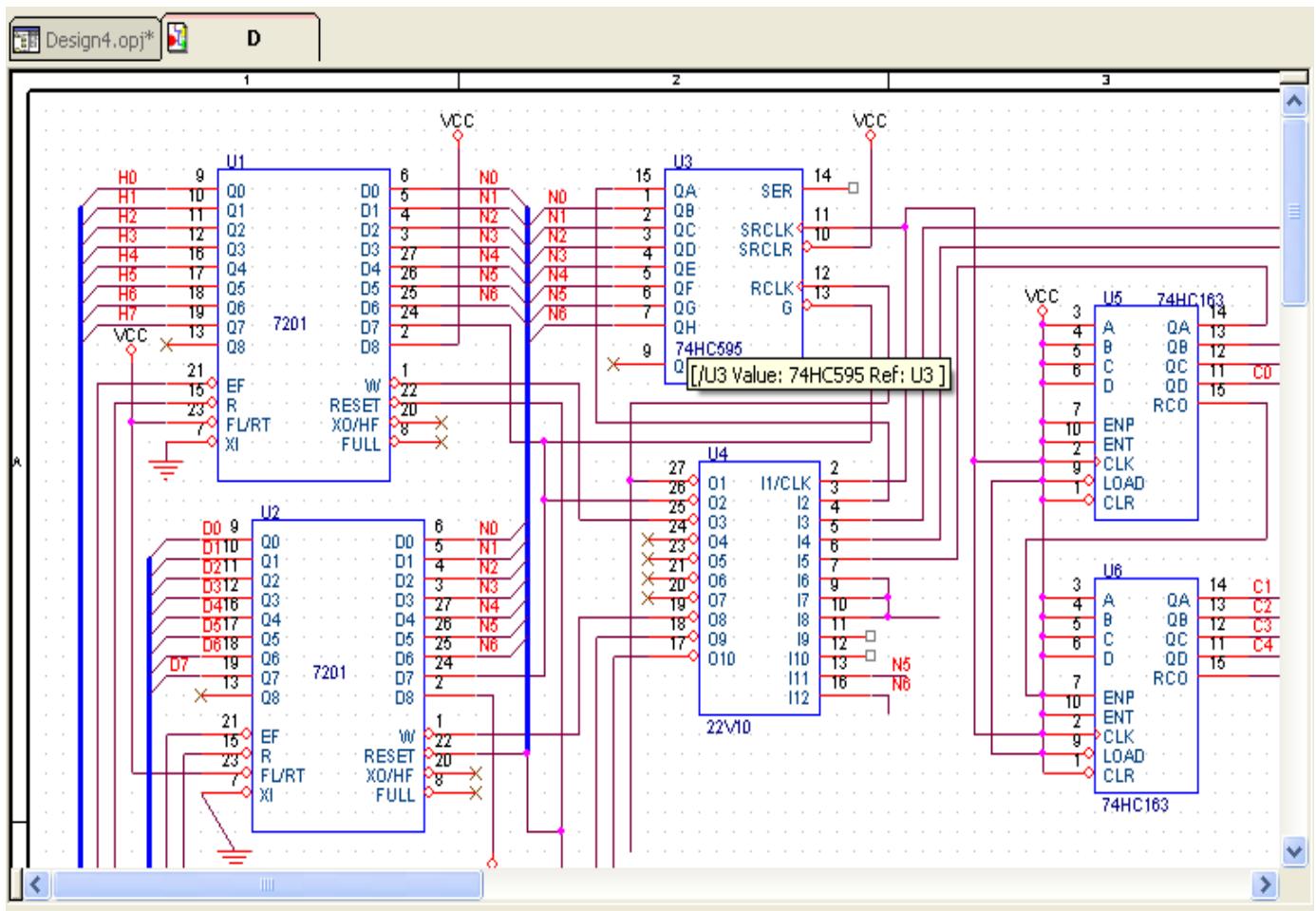


- Annotating the design with the *Left-Right* Annotation Type option selected leads to the following results:



The annotation procedure using this option scans the page grid from left to right one grid line at a time, and then moves downwards to the next grid line. In effect, the sequence is left to right and top to bottom.

- Annotating the design with the *Top-Bottom* Annotation Type option selected leads to the following results:



In this case, the procedure scans the pages from top to bottom one grid line at a time, and then moves towards the left of the screen.

Swap File

For PCB designs, a swap file (`.SWP`) 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 file using any text editor that saves files in the ASCII format, and save it with a `.SWP` extension.

Contents of a Swap File

A swap file contains the following elements:

- The file can include comments. During backannotation, any text to the right of a semicolon is ignored.

Example:

CHANGEREF U1 U2 ; Change part reference U1 to U2

- In a swap file, each line (unless preceded by a semicolon) causes one action.

Example:

GATESWAP U1 U2 ; Change part U1 to U2 and ;part U2 to U1

CHANGEREF U1A U1B ; Change part reference U1A to U1B

- The elements of each line may be separated with any number of space or tab characters.

Example:

PINSWAP U3 5 6 ; Swap pins 5 and 6 on U3

- In general, the first element of the line specifies the type of swap. If no swap type is specified, the default type *CHANGEREF* is assumed. The other swap types are *GATESWAP*, *PINSWAP*, and *CHANGEPIN*.

Example:

CHANGEREF U1A U1B ; Change part reference U1A to U1B

GATESWAP U1A U1B ; Swap gates A and B on U1

PINSWAP U5B "D0" "D1" ; Swap the pins named D0 and D1 on U5B

CHANGEPIN U5B "D0" "D1" ; Changes pin D0 to D1 on U5B

Considerations when Creating a Swap File

Include only the changes from the current state of the design to the desired state. 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, ensure that the gates being swapped are of the same type. Unmatched gate types might lead to incorrect results.
- For pin swaps, an additional element — the part reference — must be specified before the old

and new values. A pin swap is limited to the 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 reference of the specified part.

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, all the devices in U1 will change to U2, and conversely—U1A, U1B, and U1C change to U2A, U2B, and U2C, respectively. Similarly, 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
```

 Backannotation does not check part types before swapping. Swapping parts of different types, as illustrated in the following example, might lead to 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. A pin is identified by a name or a number. A pin name 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
```

 For a heterogeneous part, only the pins with the same name across sections can be swapped.

CHANGEPIN

Changes the first pin with the second pin. Only pins of the same type and shape on the same part can be swapped. A pin is identified by a name or a number. A pin name 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
```

Designating Pins, Gates, or Packages for Swapping

To improve board routing, you can designate pins, gates, or packages for swapping.

To specify that pins, gates, or packages in your PCB design are eligible for swapping, do the following:

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 to process only a portion of the design. To process the entire design, leave the schematic folders or schematic pages unselected.
3. Choose *Tools – Back Annotate*.
The [Back Annotate dialog box](#) displays.
4. Verify that the dialog box options are set as required. 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 of the swap file.

5. Click *OK*.

Shortcut



Toolbar:

ExampleDesignating

```
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
```

Update File

The update file is used by the *Update Properties* functionality to determine the objects to change, the impacted properties of the objects, and the values the properties receive on update.

You create an update (.UPD) file using any text editor that saves files in ASCII format.

Contents of Update File

The update file contains the following elements:

- **Header line** – 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 following 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.
- **Comments** – Any comment including any text to the right of a semicolon is ignored by the *Update Properties* tool.
- **Strings** – All strings in the update file except for comments are 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.
- The rest of the file contains lines for each match string to be compared and the values to be

recorded in the updated properties.

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.
```

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 following properties are updated with the corresponding values:

- Track Width
- Net Spacing
- Routing Priority

For instance, each object for which the `Net Name` property is set to `VCC` is updated as follows:

- Track Width property is set to 0.04
- Net Spacing property is set to 0.035
- Routing Priority property is set to 3

 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.

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.

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 need to 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	Definition
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
```

Advanced Annotation

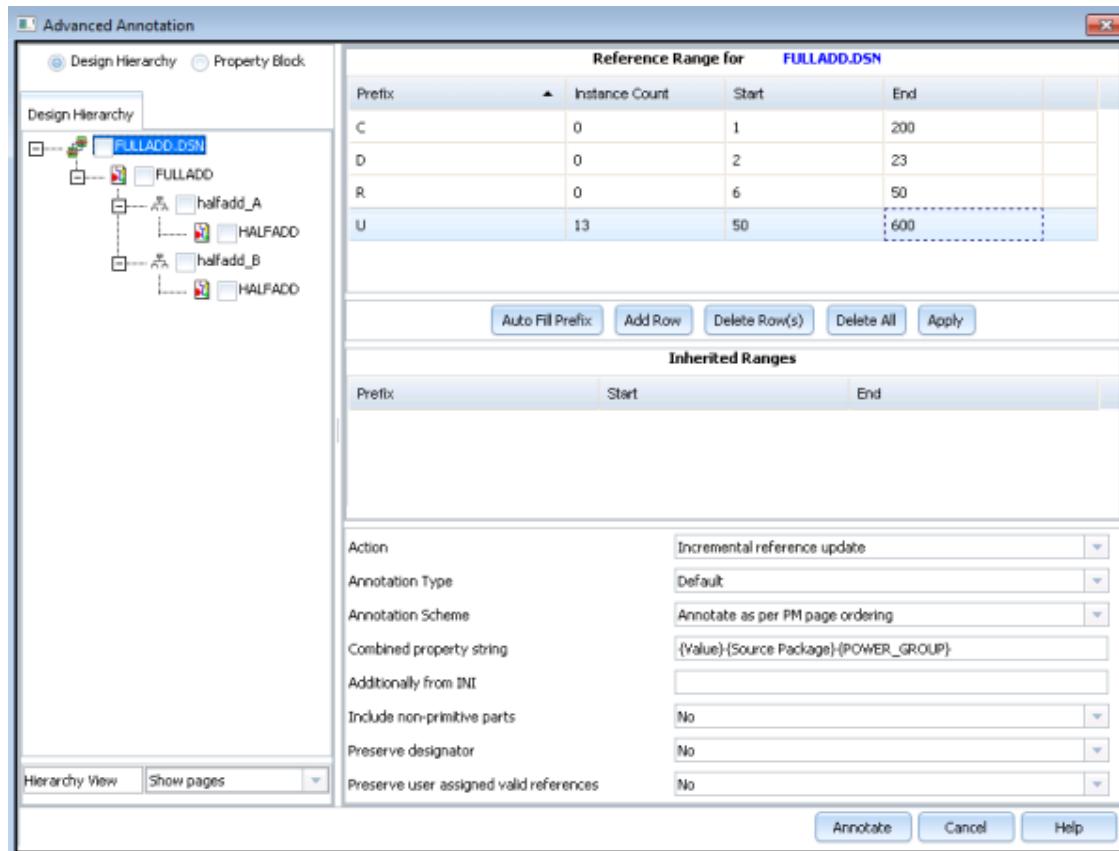
The Advanced Annotation feature provides you the capability to do multi-level annotation and property-block based annotation on a Capture Design. The multi-level annotation can be done either on a full design or selectively done on one or more schematic folders or schematic pages.

In this section, following topics are covered:

- Reference Range Assignment
- Selective Annotation
- Property Block based Annotation

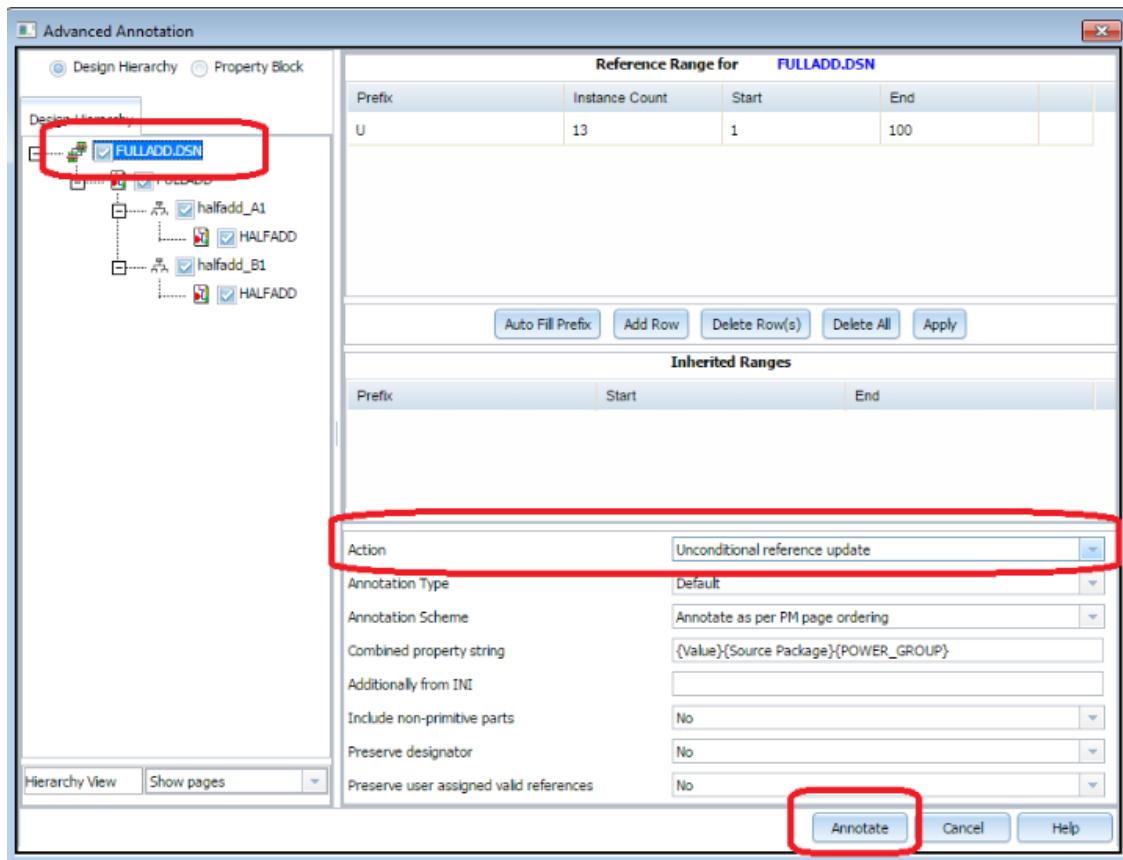
Reference Range Assignment

You can assign reference range in the Advanced Annotation dialog box for a complete design, an individual hierarchy block, an individual page in a hierarchy, or a property block.



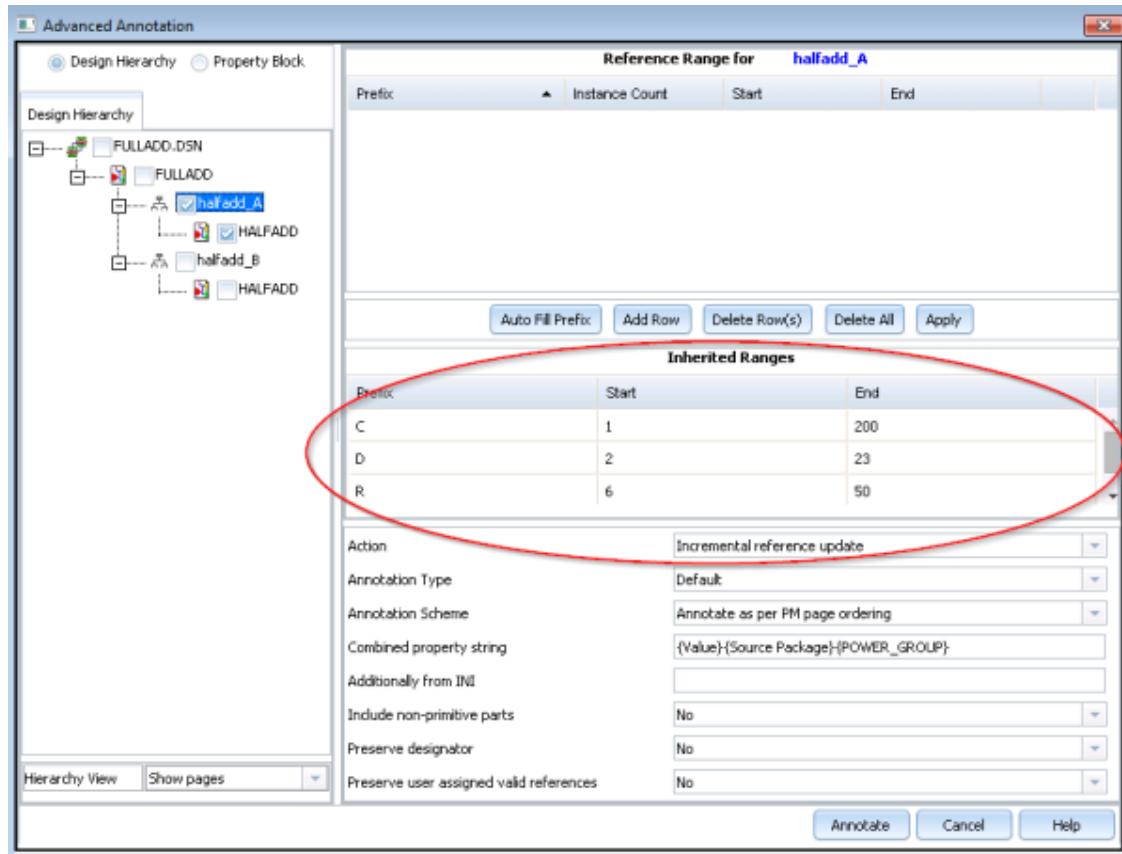
 Auto migration is done for old designs with assigned reference ranges.

For example, to annotate a complete design. Select the *FULLADD.DSN* check box in the design hierarchy tab. Once the *FULLADD.DSN* check box is selected, all the lower-level blocks are automatically selected.

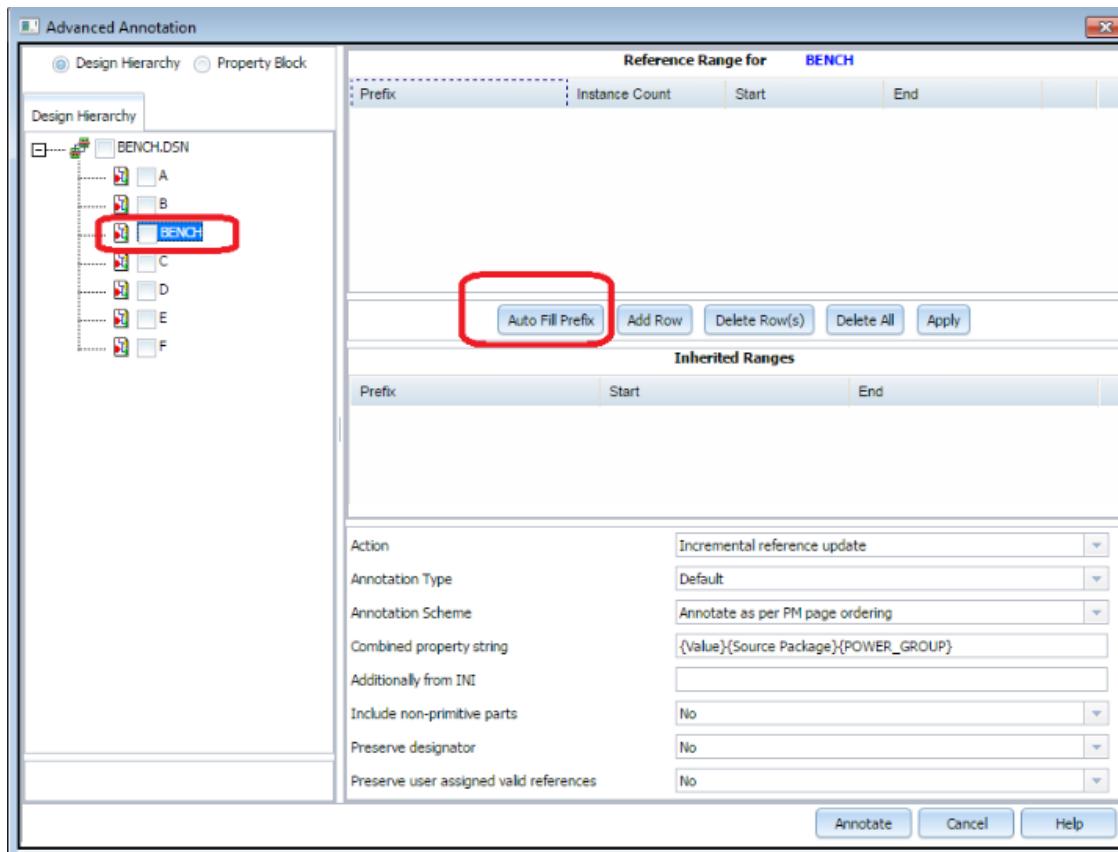


Selective Annotation

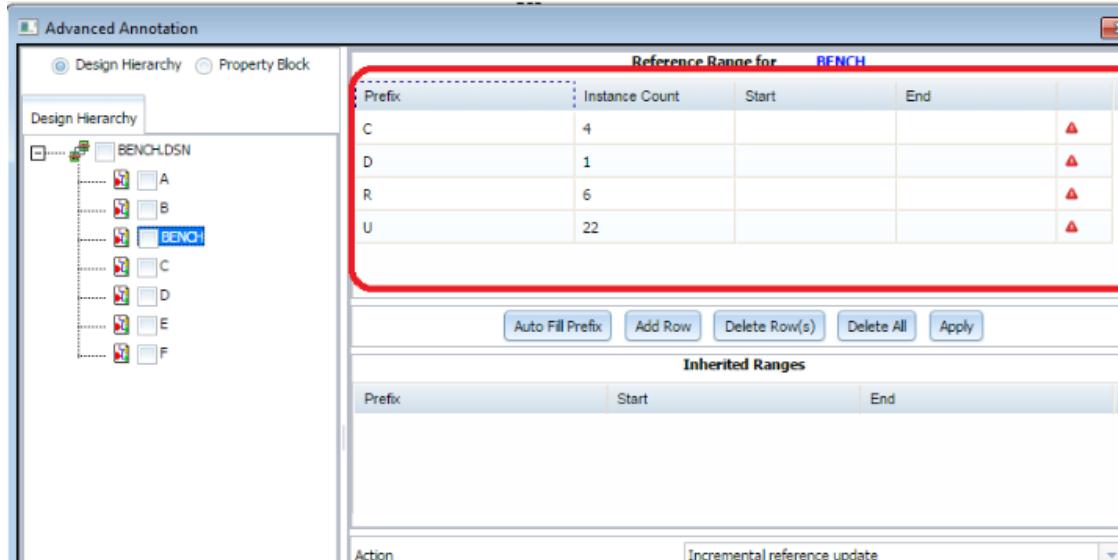
If you want to add reference range to a lower block or a child block, the inherited ranges can be seen in the Advanced Annotation dialog box as shown in the following screenshot.



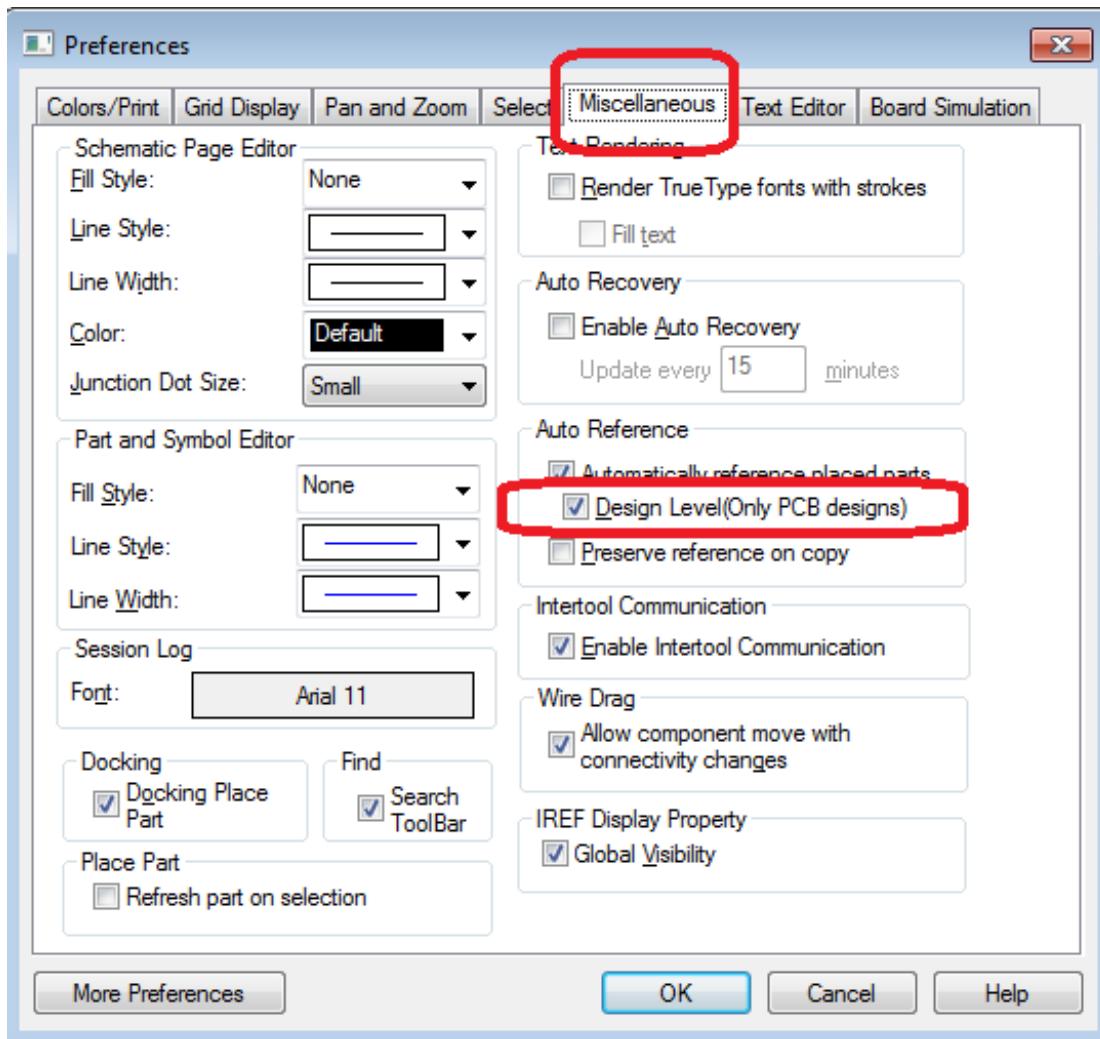
The *Auto Fill Prefix* button provides you an option to auto-scan the number of prefixes found in a design. Once you click the button, the prefixes' count is automatically filled-up in the Instance Count column of the Reference Range table.



Based on the instance count of each prefix, you can define the reference range.

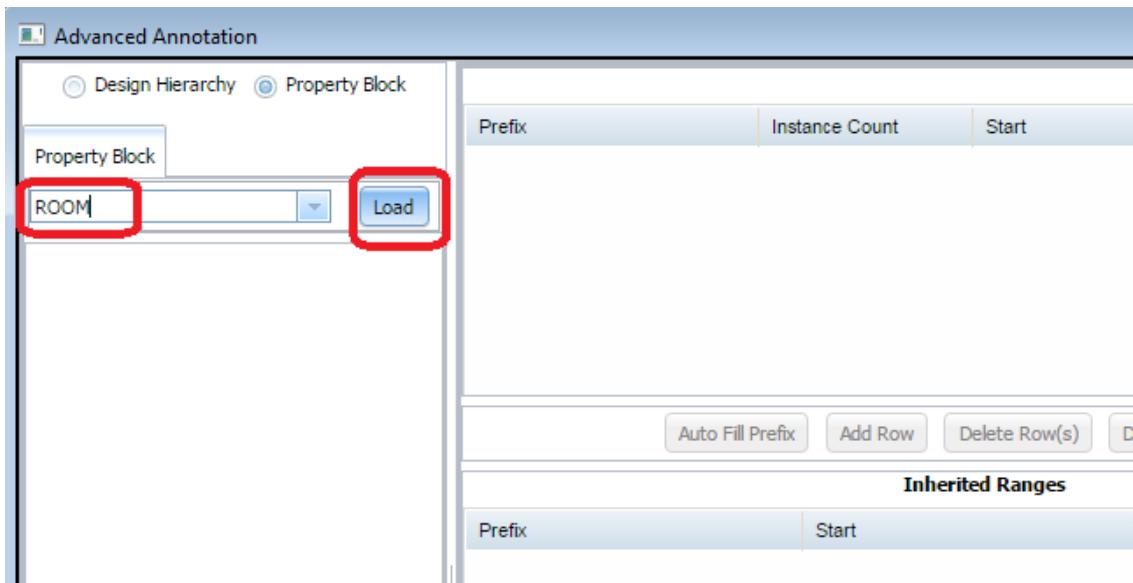


To do auto-referencing during placement of an assigned referenced part or parts, enable **Design Level Auto Referencing** from *Options – Preferences – Miscellaneous Tab*.

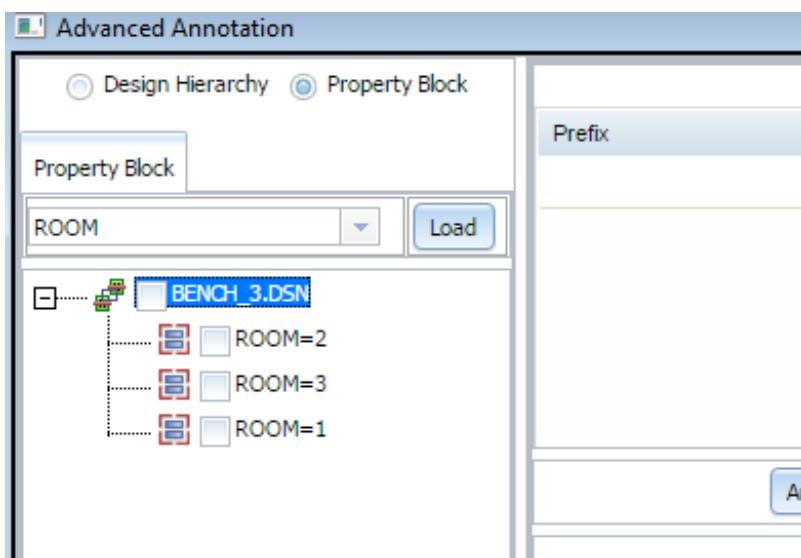


Property Block Based Annotation

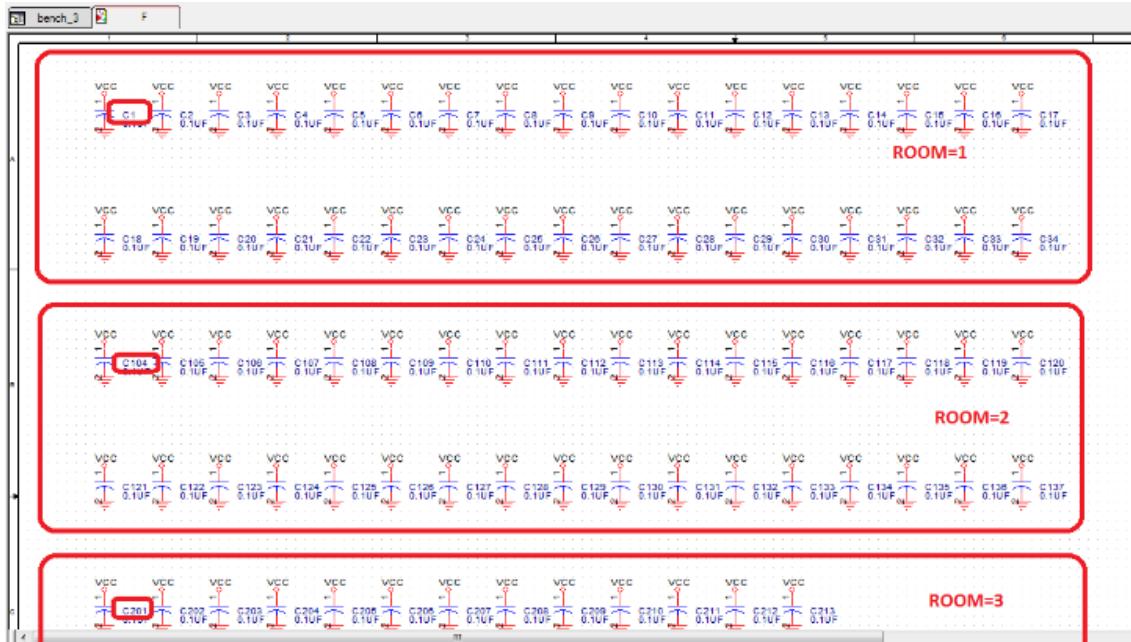
To do Property Block Annotation, you need to select Property Block radio button in the Advanced Annotation dialog box.



Specify the ROOM property and click the Load button.



Select each of the property and assign reference range 1 to 100 to the ROOM1 property, 101 to 200 to the ROOM2 property, and 201 to 300 to the ROOM3 property. After annotation is performed, the reference assignment is done according to the property block reference assignment.



- ⚠ To assign a reference range to an individual schematic folder or page, ensure that the selected schematic folder or page is highlighted and that the check box is unselected. However, when annotating one or more schematic folders or pages, make sure that the check box for each folder or page is selected.**

Generating Reports

To generate reports, you must specify a root for your design.

In this section:

- [Creating an Include File](#)
- [Creating a Bill of Materials](#)
- [Creating a Find Results Report](#)

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.

- ⚠** When you set BOM_IGNORE property to TRUE, the component instance does not appear in the Bill of Materials report.

To merge information from an external database

1. Create 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.

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.

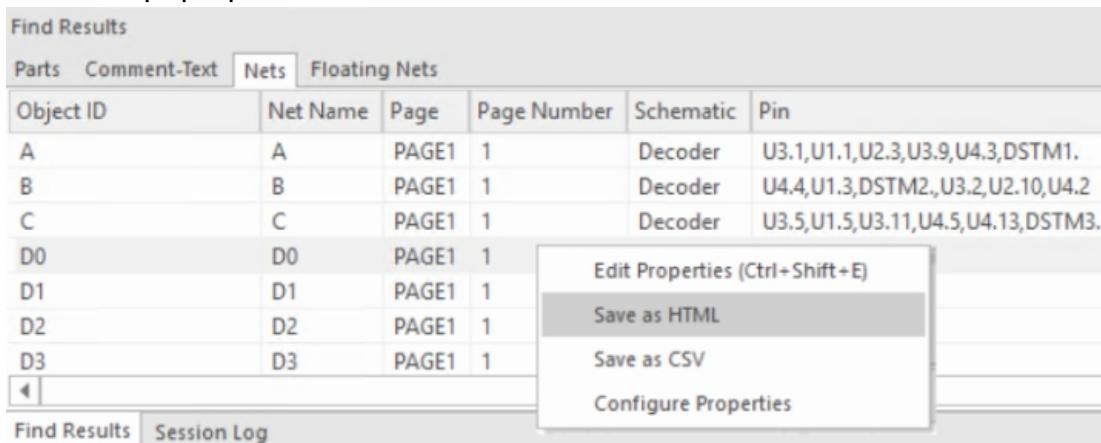
Creating a Find Results Report

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 various tabs of the Find Results window. In this case, you can export the data from each tab.

To create a Find Results report

1. Select *Edit – Find*, or press CTRL+F.
2. Right-click any search result row from the Find Results window.
3. From the pop-up menu, select, *Save as HTML* or *Save as CSV*.



A message box opens showing 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 electrical Design Rules Check on the current set of electrical design rules that you have already defined in the Design Rules Check dialog box.

To run the Check and Save command:

1. Select the design from the project manager pane.

2. Select *File – Check and Save*.

The DRCs window in the output area displays a list of electrical DRC errors and warnings generated from the running Design Rules Checks.

It also displays the DRC markers on the schematic design.

3. Double-click an 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

To print or plot a design, you work with the standard Windows dialog boxes, whether you want to send output to a printer, a plotter, or an encapsulated PostScript file. 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 the choices are 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, do the following:

1. Open the schematic page, part, or symbol to print.
2. From the *File* menu, choose the *Print* command.

In this section:

- [Printing or Plotting a Part](#)
- [Setting Print Options for Schematic Page Objects](#)
- [Printing Documents](#)
- [Printing or Plotting a Schematic Page](#)
- [Previewing Print Output](#)
- [Printer or Plotter Setup](#)
- [Scaling a Print or Plot](#)

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



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 Window
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



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.

 In the Print Preview window, if you select the *Print all colors in black* option, any color used in the design object will be printed in black.

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.

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.

Note:

- 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.

Netlisting a Design--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.

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.

Net name resolution

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.

Note: To transfer the bus to PCB Editor, ensure that you assign the `BUS_NAME` property to the bus or individual members of the bus.

- 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.

NETLIST_IGNORE Property

The `NETLIST_IGNORE`, a reserved property, is attached to component instances. When you set this boolean property to TRUE, the component instance does not appear in the Allegro netlist.

PACK_SHORT Property

Netlisting a Design--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.

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>]

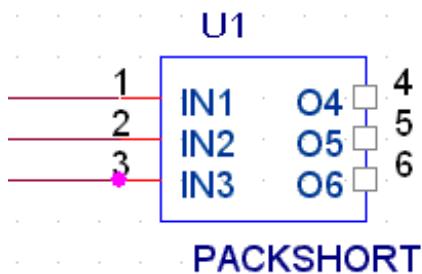
Where: <group> indicates (logicPin1, logicPin2 ... [logicPinN])

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.

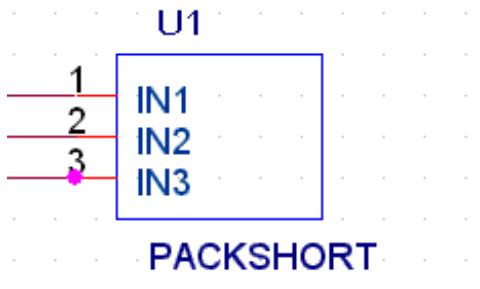
Method 1:



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 tab](#)
- [EDIF 2 0 0 tab](#)
- [INF 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.

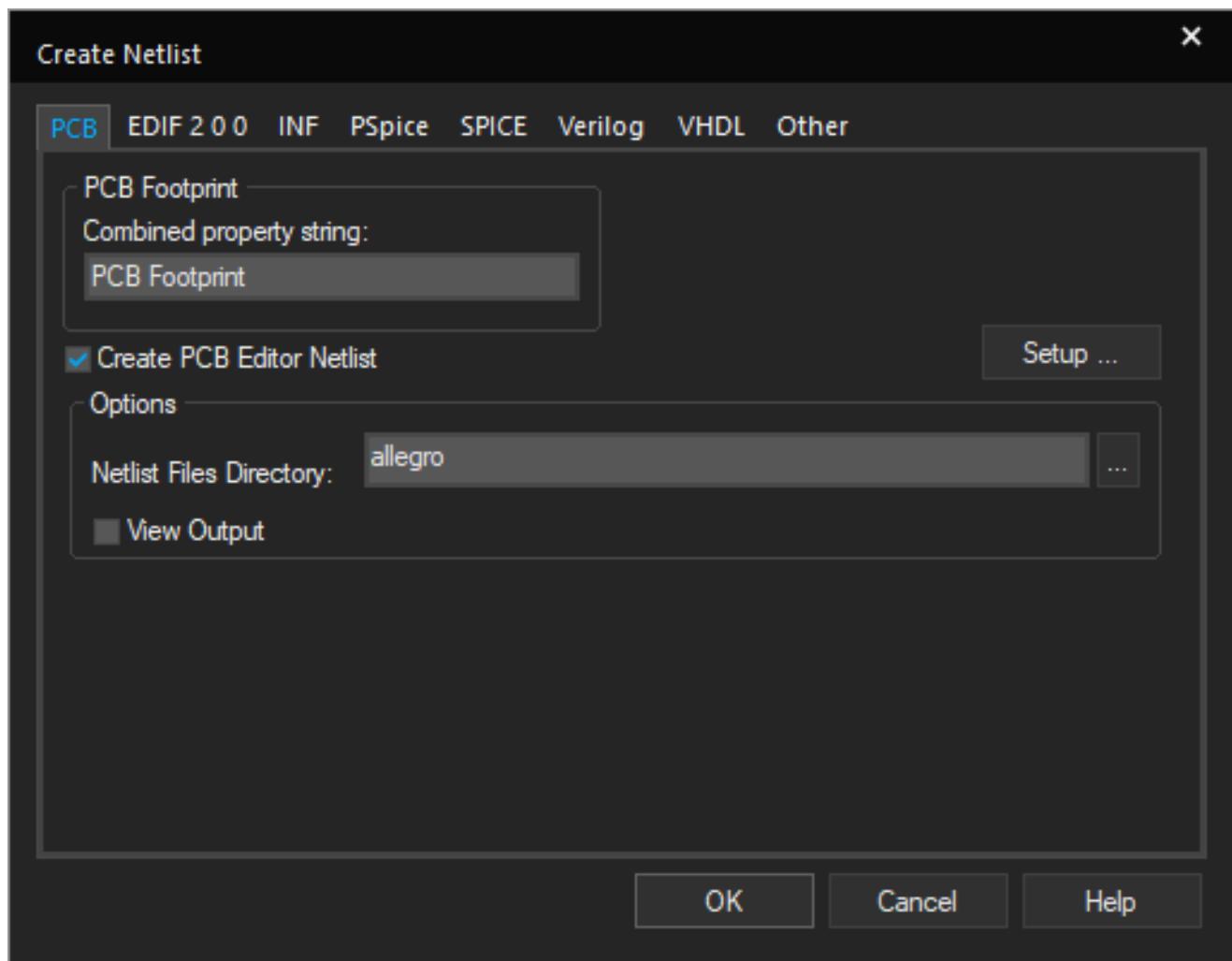
In this section:

- [Creating a Netlist](#)
- [Creating a Flat Netlist](#)
- [Generating a Layout Editor Netlist](#)
- [Working with Hierarchical Netlists](#)
- [Specifying an Alternate Netlist Template](#)

Creating a Netlist

To create a netlist, do the following:

1. In the project manager, select the design file for which you want to create a netlist.
2. Choose the *Tools – Create Netlist* menu to display the *Create Netlist* dialog box.
3. In the *Create Netlist* dialog box, 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, do the following:

1. In the project manager, select the design file (`.DSN`) to netlist.

2. From the *Tools* menu, choose *Create Netlist*.
3. Select 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*.

Generating a Layout Editor Netlist

To forward annotate information from Capture to a layout editor, such as PCB Editor or OrCAD X Presto, do the following:

1. In the Capture project manager, select the design for which a layout editor netlist is to be created.
2. Choose the *Tools – Create Netlist* menu.
3. In the *Create Netlist* dialog box, select the *PCB* 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, `<installation_directory>\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.
The 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 ORCAP-36006) you want to suppress in the *Suppress Warnings* text box and click *Add* to add the warning to the list of warnings to be suppressed.
- d. Specify the value of other fields as required and click *OK*.
6. In the *Netlist Files Directory* text box, 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* window.

-  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.

Capture netlister generates the following three netlist files compatible with PCB Editor and OrCAD X Presto.

- **PSTCHIP.DAT:** This file contains a description for each different type of part used in the design. The netlister extracts this information from the 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 the reference designator and the section number.

Working with Hierarchical Netlists

In this section:

- [Creating a Hierarchical Netlist](#)
- [Creating Subcircuit Netlists](#)
- [Using SUBPARAM](#)

Creating a Hierarchical Netlist

To create a hierarchical netlist, do the following:

1. In the project manager, select the design file (`.DSN`) you want to netlist.
2. Choose the *Tools – Create Netlist* menu command.
3. Select 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. After these settings are defined, they persist and 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. You can change the settings in the *Settings* dialog box.

Two groups of settings are saved: PSpice and *Layout versus Schematics* (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, do the following:

1. In the *PSpice* tab of the *Create Netlist* dialog box, click *Create Hierarchical Format Netlist* under the *Options* group box.
2. Click *Settings*, then enable or specify the 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 use this feature, use the `SUBPARAM` part in the `SPECIAL.OLB` library.

To learn more about setting up parameterized subcircuits for hierarchical netlists, see [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.

i **Warning Message**

During hierarchical netlist creation, a message might display indicating that occurrence properties are being ignored, '*Occurrence-specific properties on <part name>, ignoring.*'. No components are ignored in this case.

This message appears only during hierarchical netlist creation. There might be scenarios when the same hierarchical block is instantiated more than once on the schematic. The underlying components inside hierarchical block might have different values or properties (by virtue of occurrences). A hierarchical netlist creates just one subcircuit for both the replicated hierarchical blocks, and call that subcircuit multiple times in netlist based on the number of instances of the block . The underlying component values considered for the netlist are *instance* values and not *occurrence* values because only one subcircuit is being created by the netlister.

This message flags for all the components inside the hierarchical block, which has occurrence properties different from instance properties.

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:
 - o 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

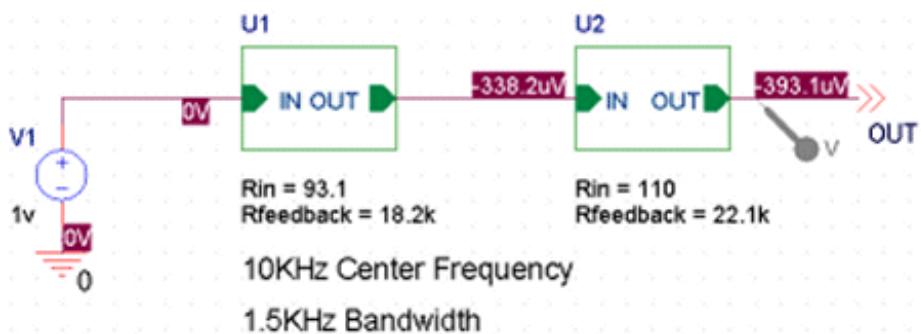
1. Make the subcircuit your active schematic page in the Capture schematic page editor.
2. From the *Place* menu, choose the *Part* command.
3. Select the part SUBPARAM from the PSpice library SPECIAL.OLB and place it on the subcircuit.
4. With the SUBPARAM part still selected, from the *Edit* menu, 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 $R_{in} = 110$ and $R_{feedback} = 22.1k$.

4th Order Chebyshev Filter



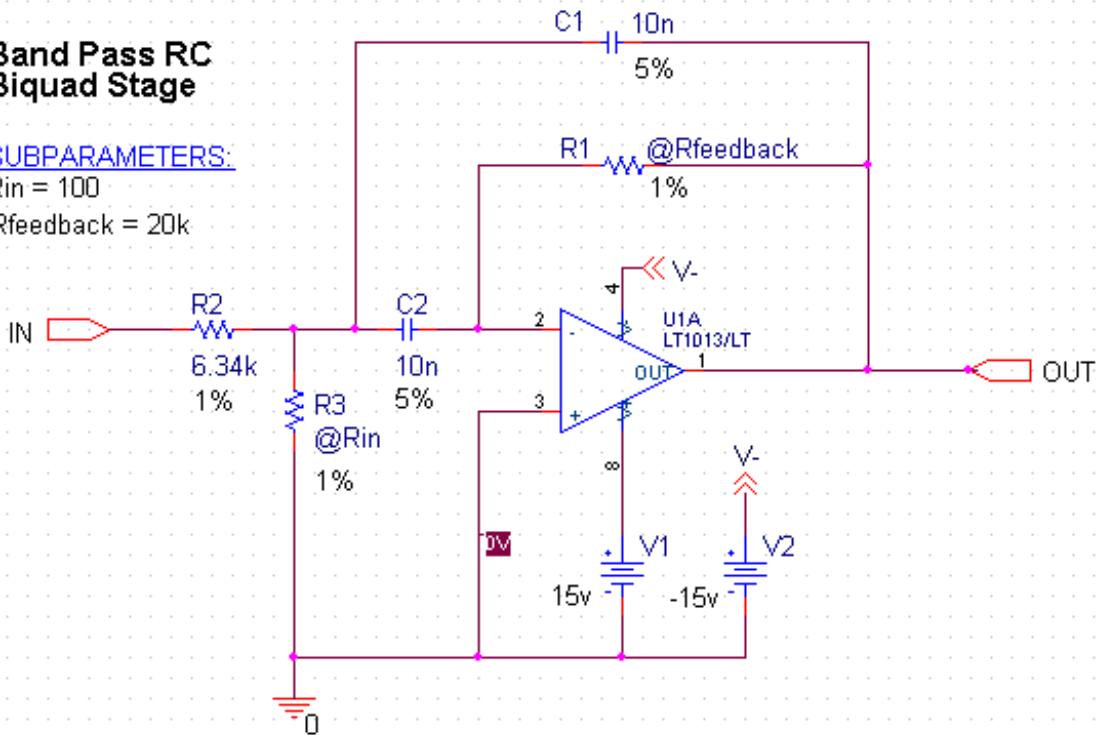
Descending into the hierarchy of the U2 subcircuit, you see the R_{in} and $R_{feedback}$ parameters listed with their corresponding default values of 100 and 20k, respectively, in the SUBPARAM part (under SUBPARAMETERS).

Band Pass RC Biquad Stage

SUBPARAMETERS:

Rin = 100

Rfeedback = 20k



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.

Simulating Capture Designs 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. PSpice supports analyzes that can simulate analog-only, mixed-signal, and digital-only circuits.

- [Overview of Simulation Using PSpice](#)
- [Creating Design for Simulation](#)
- [Using Partial Design Simulation](#)
- [Running a Simulation and Viewing Results Using PSpice](#)
- [Files Needed for Simulation](#)
- [Running PSpice in Batch Mode](#)
- [SPICE netlist format](#)

Overview of Simulation Using PSpice

To simulate your design in PSpice:

1. Ensure that you have PSpice license.

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.

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 analyses 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 Design for Simulation

A design that is targeted for simulation has:

- parts for which simulation models are 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 topics covered in this section:

- [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 Capture Parts](#)
- [Viewing PSpice Models](#)
- [Viewing Information on Missing PSpice Models in Online DRC](#)
- [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.

You can also add more PSpice Parts using PSpice Part Search (choose *Place - PSpice Component - Search*).

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.

i 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 parameterized 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 parameterized 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 PSpice User Guide.

Use parameterized 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 parameterized 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\advans\`
- 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.

To find parts using the part 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 PSpice Part Search:

You can also search for PSpice parts using PSpice Part Search. For more information, see the [Searching and Placing PSpice Parts](#) section.

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.

- i** 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.

- i** 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
<	I (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.
- You can use the European notation, where the decimal point is omitted and replaced by the unit symbol, for specifying values. For example, you can use 3K3, which is the European notation for 3.3K.
- 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?
- Do not add any space for the name of an hierarchical block.

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.
- You can place PSpice ground 0 symbol from PSpice Component quick pick, that is, Place - PSpice Component - PSpice Ground.

⚠ 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 Property 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

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 Guide)
- user-defined functions
- system variables (listed in the PSpice User 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.

Associating PSpice Model to Capture Parts

You can associate a PSpice model to an existing or new part created in Capture. Before associating a PSpice model to a part, ensure the following:

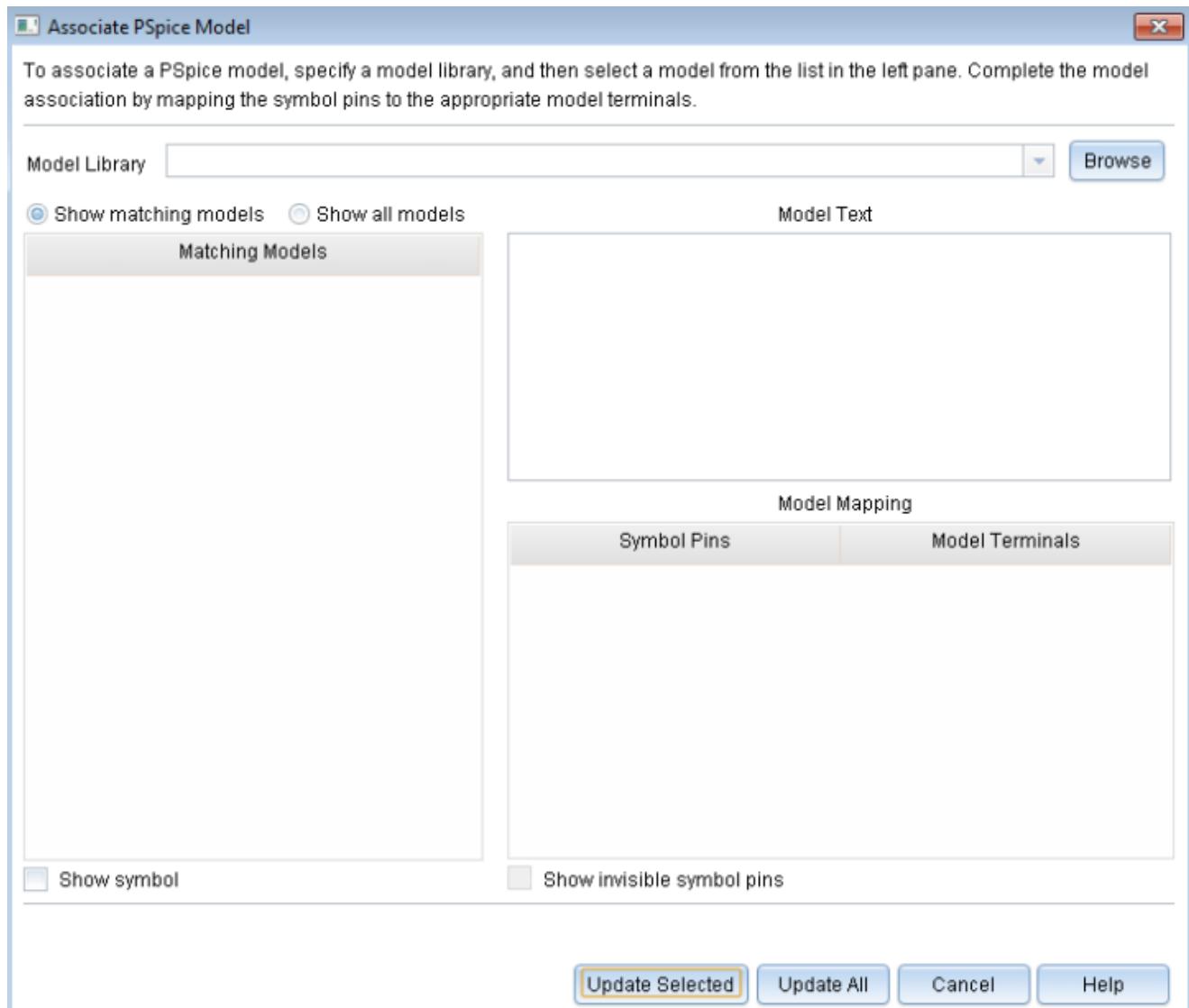
- Use this feature on a part placed on the root schematic. If the design is not a root schematic, open the project manager, right-click this schematic, and select *Make Root* to convert it into a root schematic design.

- Only a single part should be selected at a time.
- PSpice models can only be associated with homogeneous parts.
- If the PSpice model is to be associated to a part in an externally referenced design, you need to open this design separately. Next, associate a PSpice model to the required part, save the design and then reuse it in the root schematic.

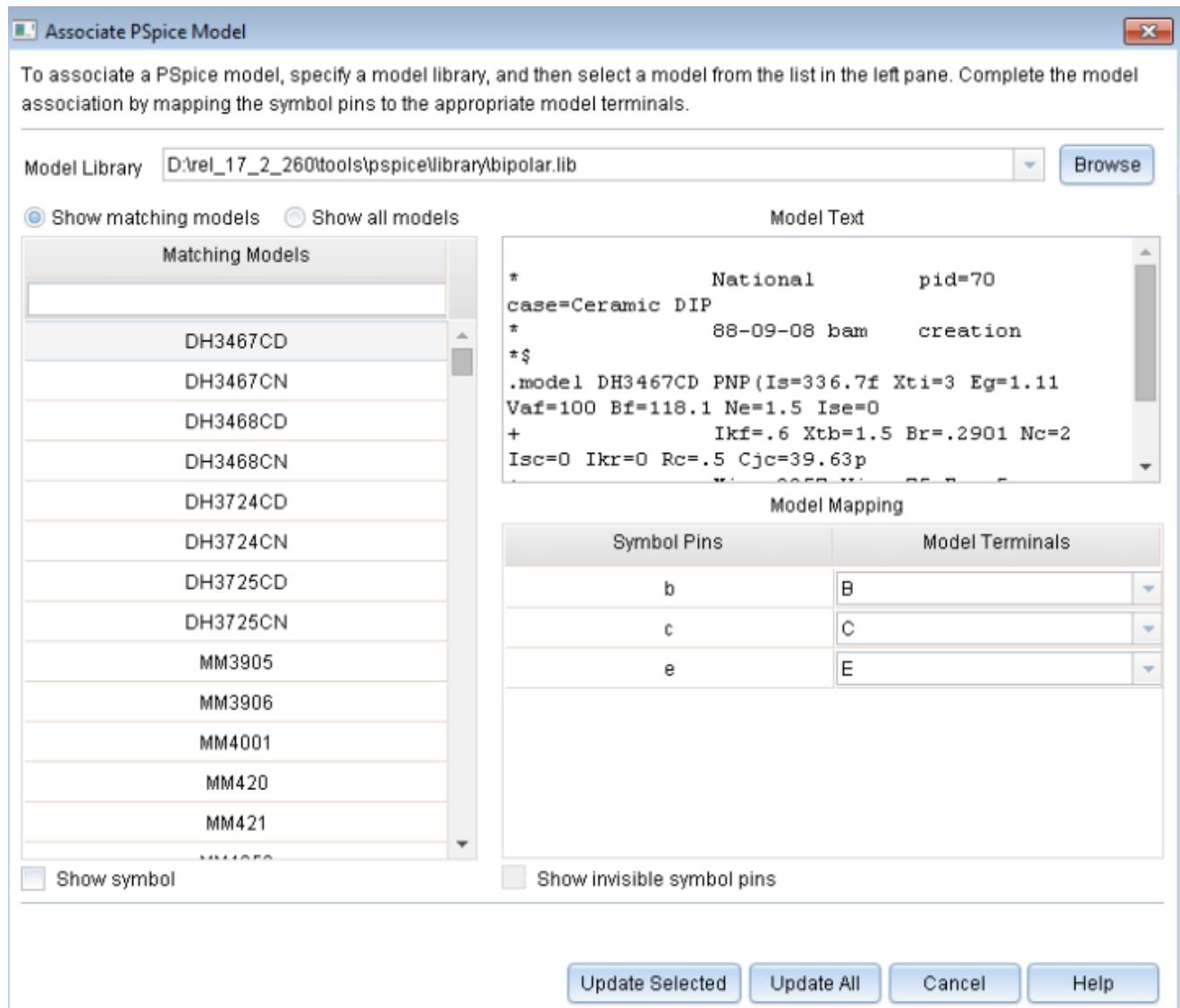
To associate a PSpice model to a Capture part in the schematic editor:

1. Select the Capture part to which you want to associate a PSpice model.
2. Select *Tools – Associate PSpice Model*.
or
From the pop-up menu, select *Associate PSpice Model*.
3. A message appears prompting you to save the design before model association. If the selected part already has an associated model, a message appears to confirm if you want to overwrite this implementation. Click *Yes*.

The Associate PSpice Model dialog box appears.



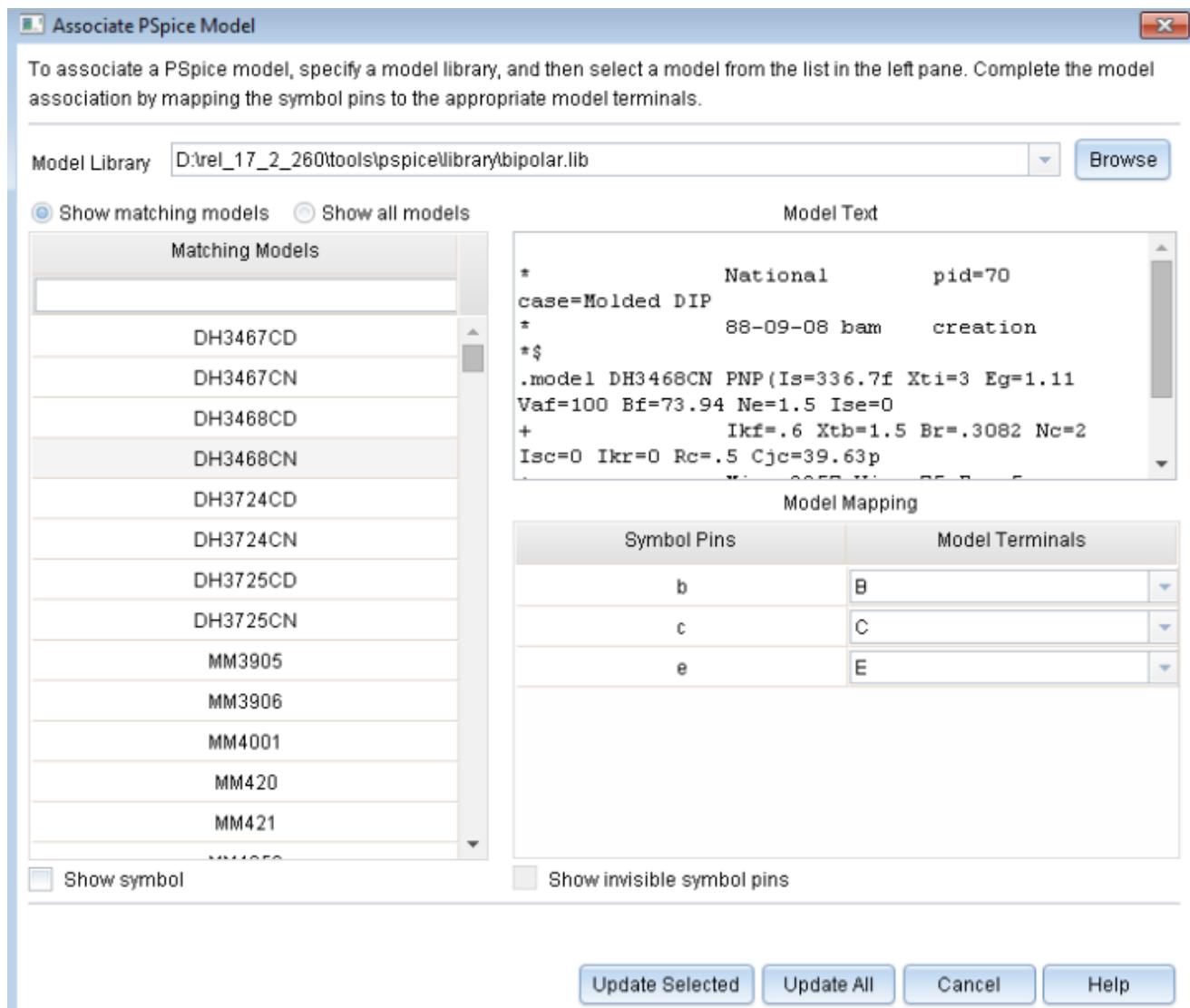
- ① If the symbol is open in Capture, the *Associate PSpice Model* option will appear disabled in the pop-up menu.
4. Click *Browse* to specify the name and location of the library containing the required PSpice model. You can either select the library from the *Model Library* drop-down list box or browse to the library location.
 5. Once you have selected the `.lib` file, the *Associate PSpice Model* window lists the matching models. By default, the *Show matching models* option is selected. To view all the models, select the *Show all models* option.



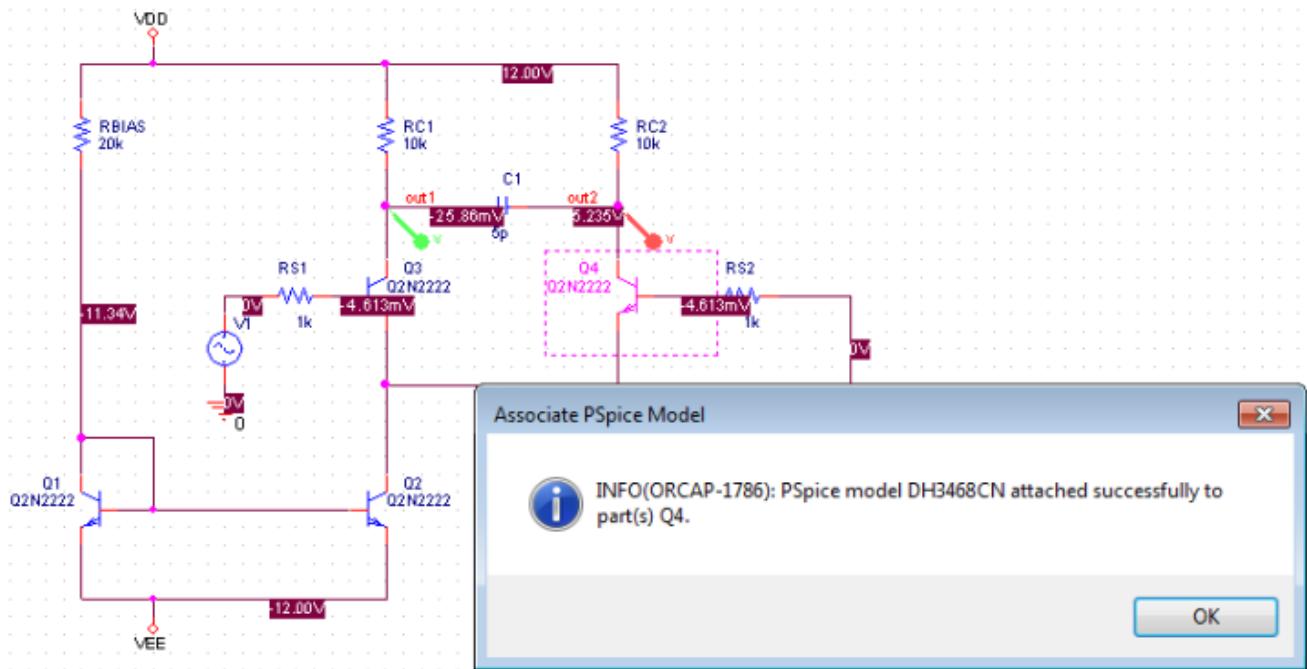
6. Select the required model from the *Matching Models* section.
7. Map each symbol pin to the correct model terminal in the *Model Mapping* section. The model terminals for the selected symbol pins appear in the *Model Terminals* drop-down list.

For mapping, you can view the model definition. When you select a model in the *Matching Models* section, its definition appears in the *Model Text* section.

Click the *Show symbol* check box to preview the symbol along with its sections. The symbol preview can only be seen when the library path specified in *Design Cache* exists.



8. Select the *Show invisible symbol pins* check box to view and map any invisible power pins in the symbol. This check box will appear disabled if there are no such pins.
9. Click *Update Selected* to complete attaching the selected PSpice model to the selected Capture part and to close the Associate PSpice Model window. A message appears to indicate that the selected model is now attached to the part selected on the schematic editor.



⚠️ Update All

If you click *Update All*, all part instances of the same source package are associated with the selected PSpice model. If you click *Update All*, and the selected part is also in an externally referenced design, the PSpice model will not be associated to such a part.

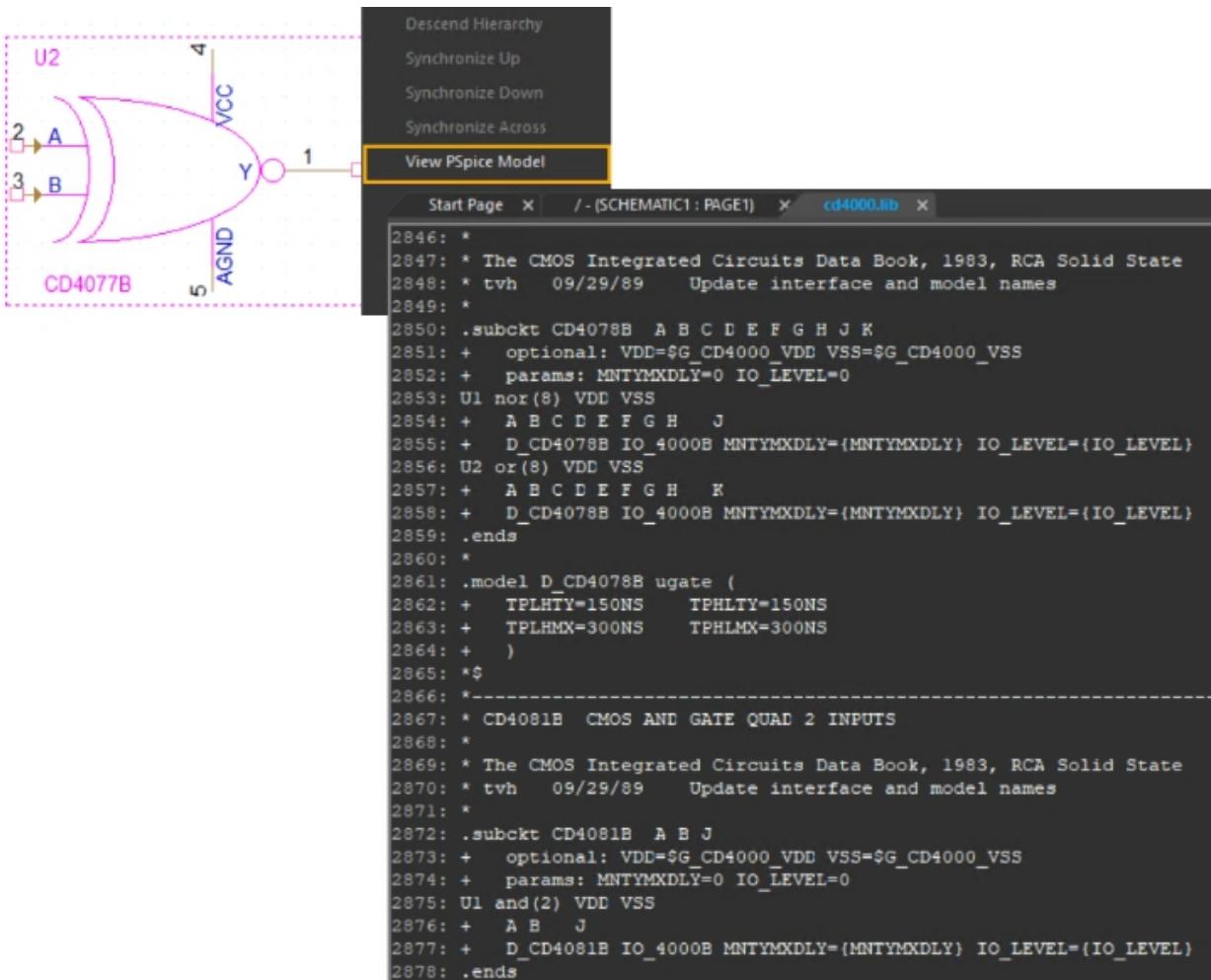
10. Click *OK*.

This automatically saves the design.

Viewing PSpice Models

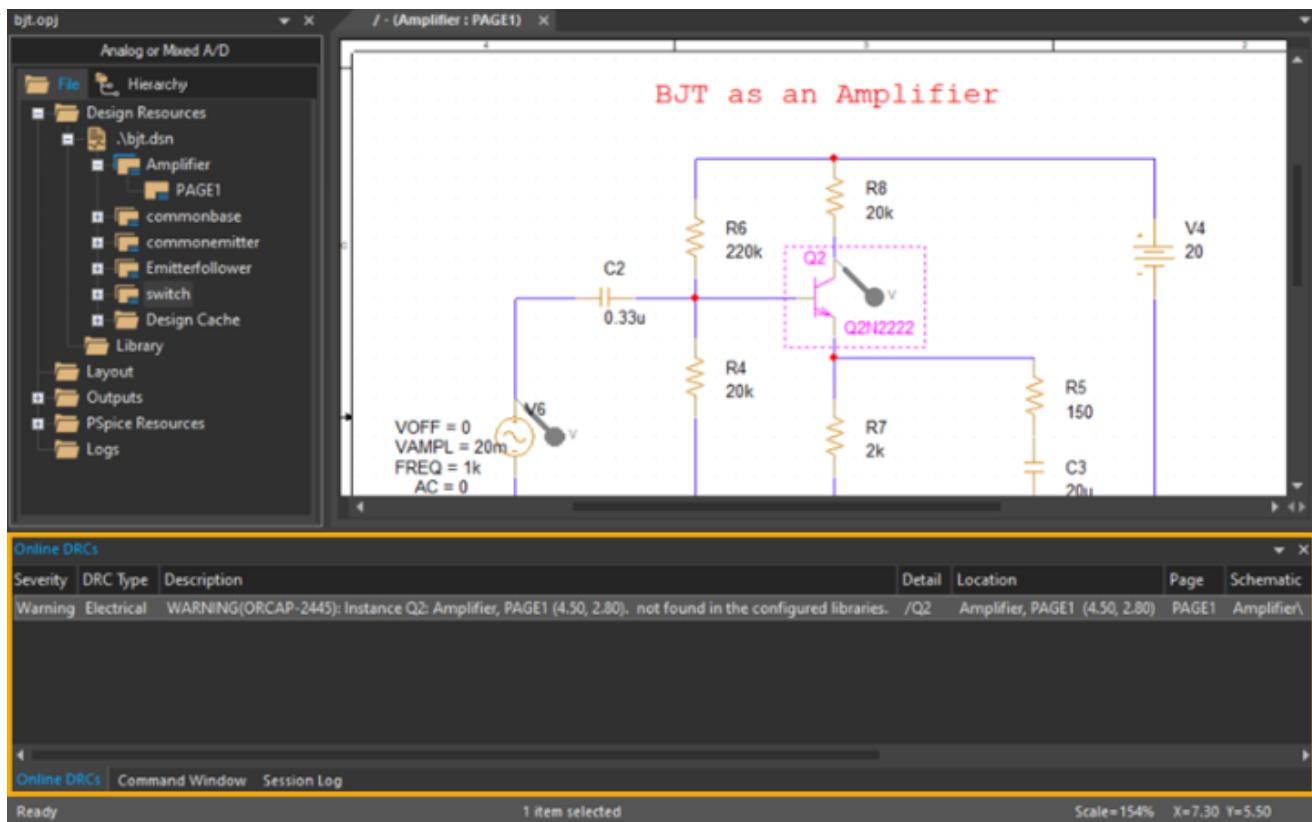
This feature allows you for a dedicated and within-Capture view of the PSpice model. For a part placed on the schematic, you can view the information of its associated PSpice model in a separate tab in the canvas area. To do so, right-click a part and select *View PSpice Model* from the pop-up menu.

This information is read-only. Some of the information may be non-readable if the associated model is encrypted.



Viewing Information on Missing PSpice Models in Online DRC

For a part placed on the schematic, if the PSpice model associated with the part is not found in the configured libraries, then information related to such a part instance appears as a message in the *Online DRC* tab of the Output pane.

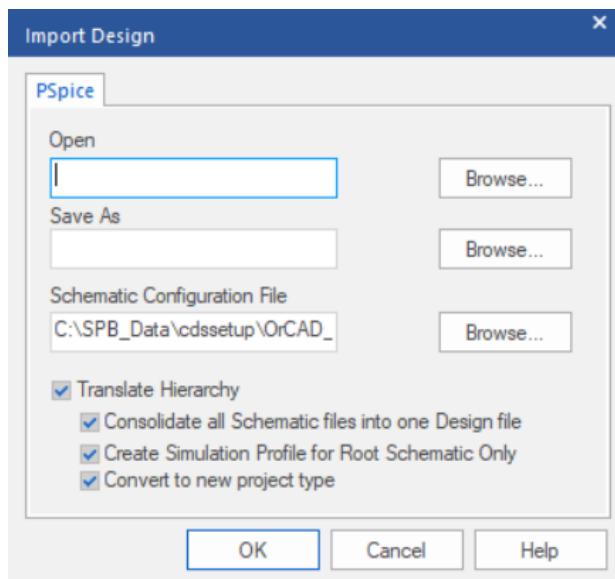


Importing PSpice Schematic Projects in Capture

You can import a MicroSim Schematics project to a Capture project.

To convert a Schematics project to Capture:

1. In Capture, choose *File – Import – PSpice*.
The Import Design dialog box opens.



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 detailed information about using the Schematics-to-Capture Translator, see the user guide, [Translating Designs from PSpice Schematics to Capture](#).

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 IEEXP, 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]

- i** 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

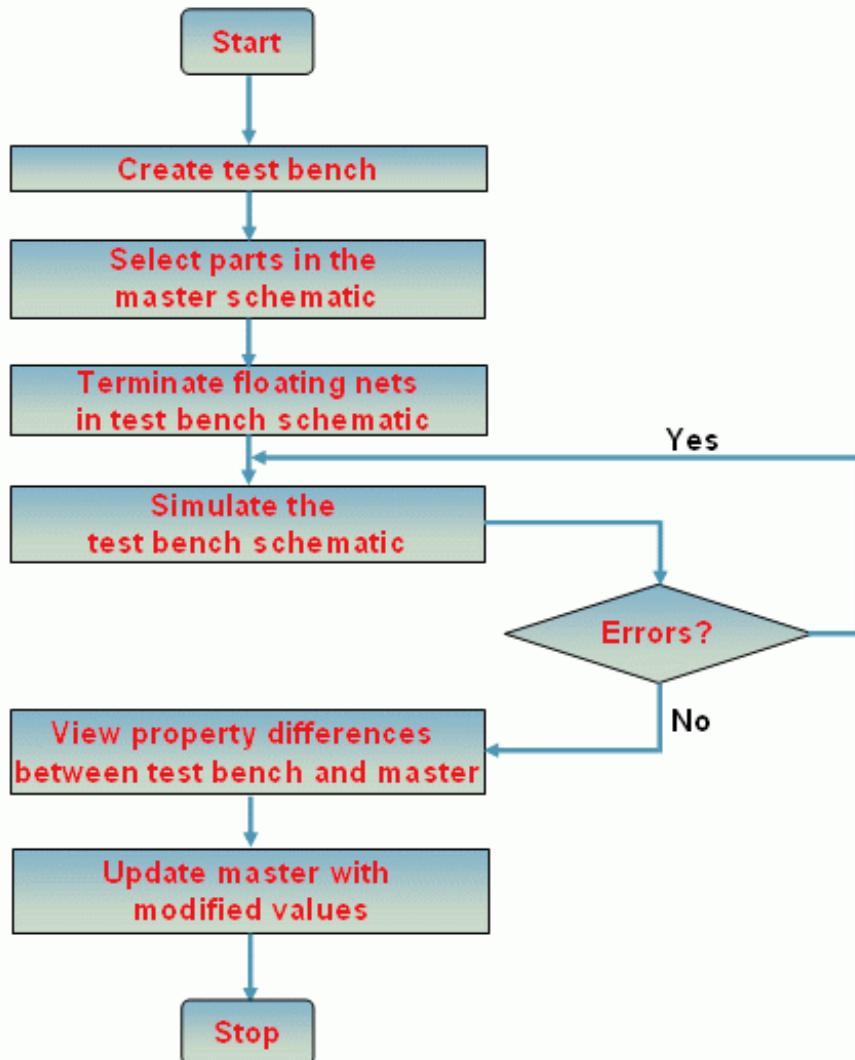
- Compare and merge portions of a design quickly

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 X 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 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
```

4. 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.



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, context-menu for selected parts in the test bench design, or from the hierarchy editor.

To activate components from the master design:

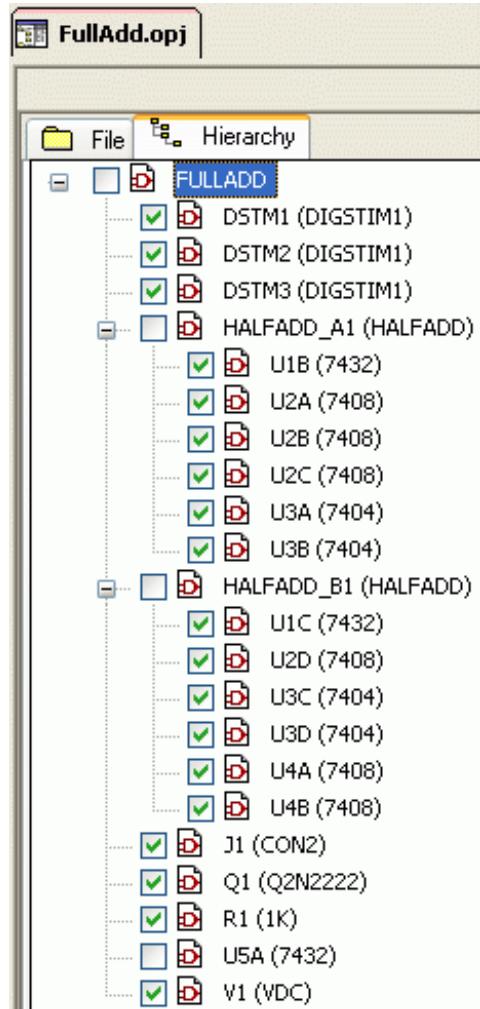
1. Select the components in the master design.
2. Right-click and choose *TestBench – Add Part(s) to Active TestBench*.

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.

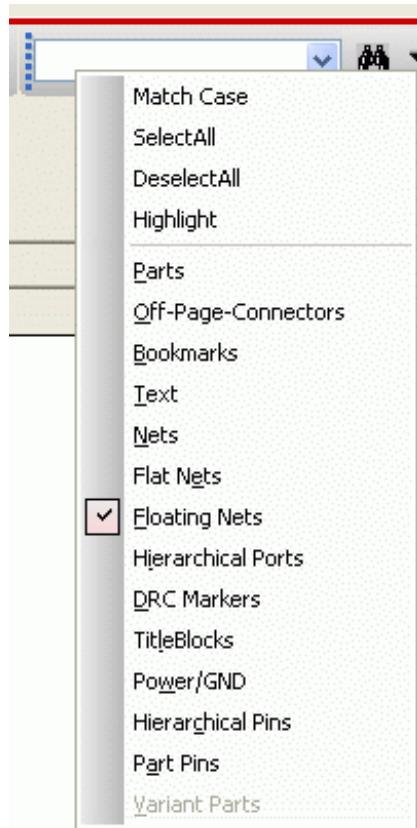


Similarly, you can deactivate a component from the test bench design from the context-menu

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.

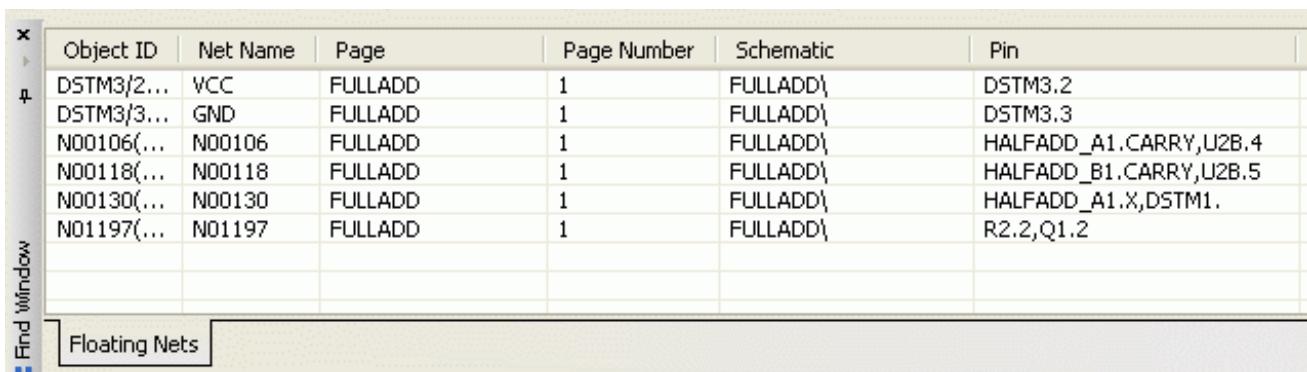
When you activate only a portion of the design, many nets might be floating because they are not terminated. You can easily resolve this problem by making a floating net search. To perform this search:

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

- 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 *Colortab*. You can also filter for different objects, if you do not want them to be listed.

To compare and propagate changes:

- Select the master DSN in Project Manager.
- Choose *Tools – Test Bench – Compare Test Bench*.

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, do the following:

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 select a profile from the current project, click the *Inherit From* drop-down list. This list shows all the simulation profiles in the current project.
 - To select a profile from another project, click the browse button to navigate to the desired simulation profile.
4. Click *Create* to create the profile and display the *Simulation Settings* dialog box.
The Simulation Settings dialog box appears.
5. In the *Analysis* tab, specify the analysis type.
Specify 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.

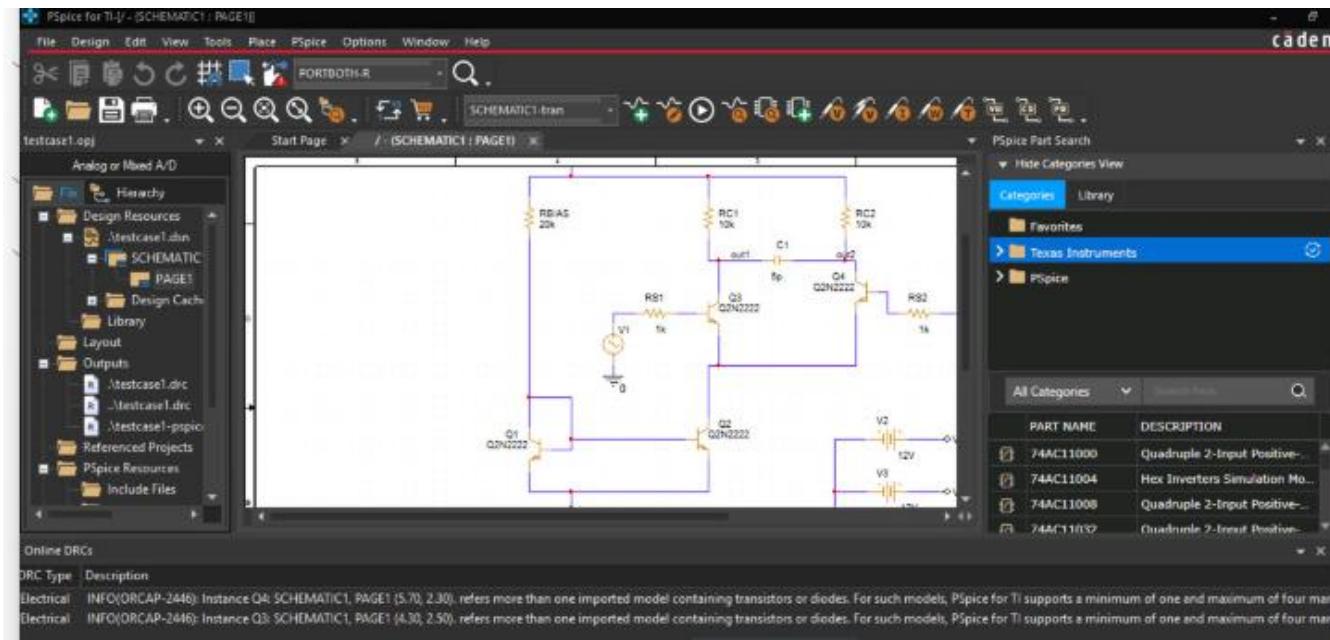
To activate a simulation profile:

1. In the project manager, choose the simulation profile you want to activate.
2. From the *PSpice* menu, choose *Make Active*.

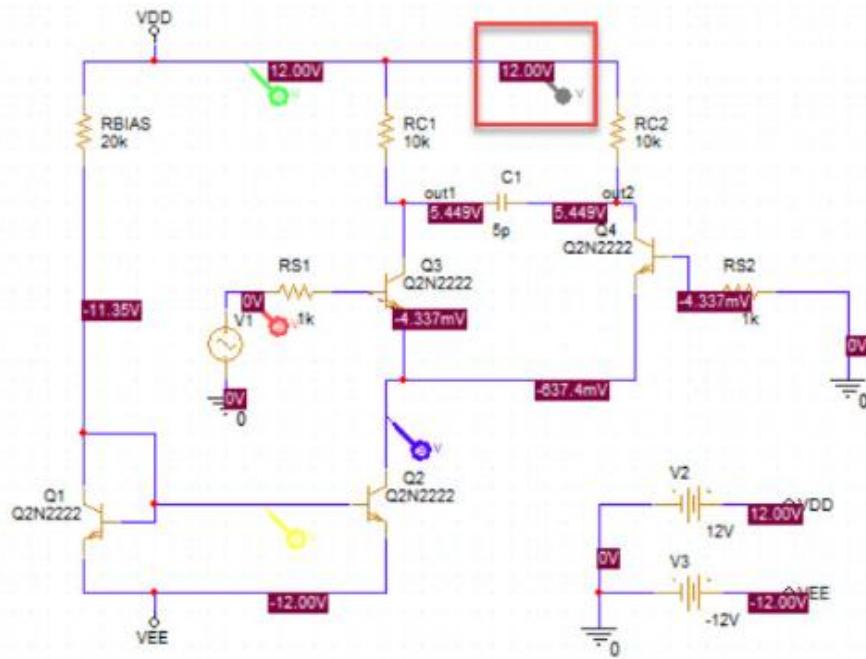
Imported Parts in Schematic Design

The *PSpice for TI* installation comes with a set of pre-installed libraries. A design can include parts or models from these pre-installed libraries as well as imported parts from external libraries.

When parts with imported models are used in a design, a corresponding message is displayed in the *Online DRCs* window.



When you run a PSpice simulation on a design containing imported models, you can place at least one and, at the most, four markers for data collection. Any marker placed after the fourth marker is not considered for data collection. After the simulation run, all such markers appear gray without the 's' symbol, identifying them as deactivated.



Imported Model of a Restricted Model Type

You can include at least one imported model with unlimited instantiations of a restricted model type, such as diodes and transistors.

- If a design contains only one imported model, there are no restrictions on the number of markers used for simulation.
- If a design contains more than one imported model, a minimum of one and a maximum of four markers are required for simulation.
- Multiple instances of an imported model are simulated with no restriction on the number of markers used.

Simulation Scenarios with Pre-Installed or Imported Models

The simulation scenarios are explained in the following table:

Case 1	Schematic contains all parts from pre-installed libraries. The simulation starts.
Case 2	Schematic includes parts containing only one imported model or multiple parts referring to only one imported model. The simulation starts.
Case 3	Schematic includes parts containing more than one imported model with no marker is placed on the design. The simulation does not start, a message indicates that at least one marker needs to be placed.
Case 4	Schematic includes parts containing more than one imported model and has markers ranging between one and four, for data collection. The simulation starts.

In addition to these scenarios, there might be a case where the schematic includes parts containing more than one imported model and has more than four markers for data collection. In such as case, the simulation does not start. You are prompted to reduce the number of markers to run the simulation.

To run the simulation, use one of the one of the following ways:

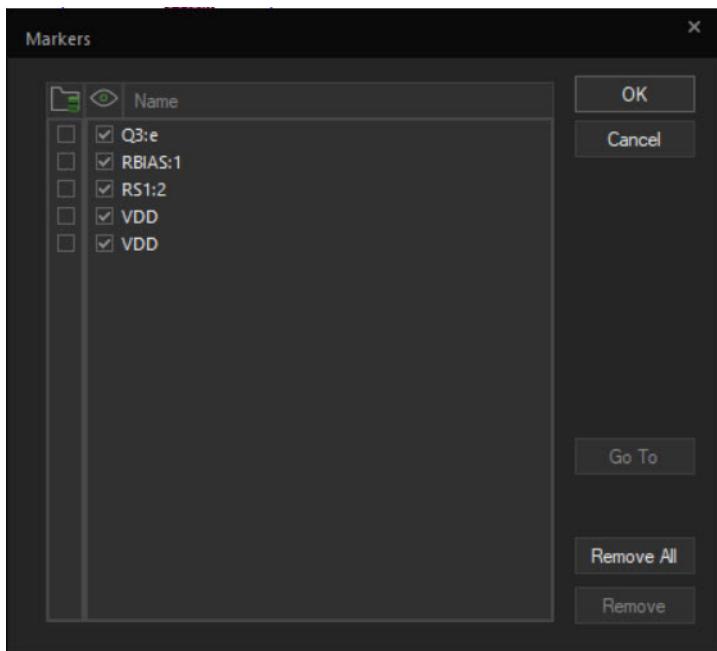
- [Remove Extra Markers](#)
- [Identify Marker as Special](#)

Remove Extra Markers

You can remove the additional markers in the *Markers* dialog box using the following steps:

1. Click *OK* in the message box.

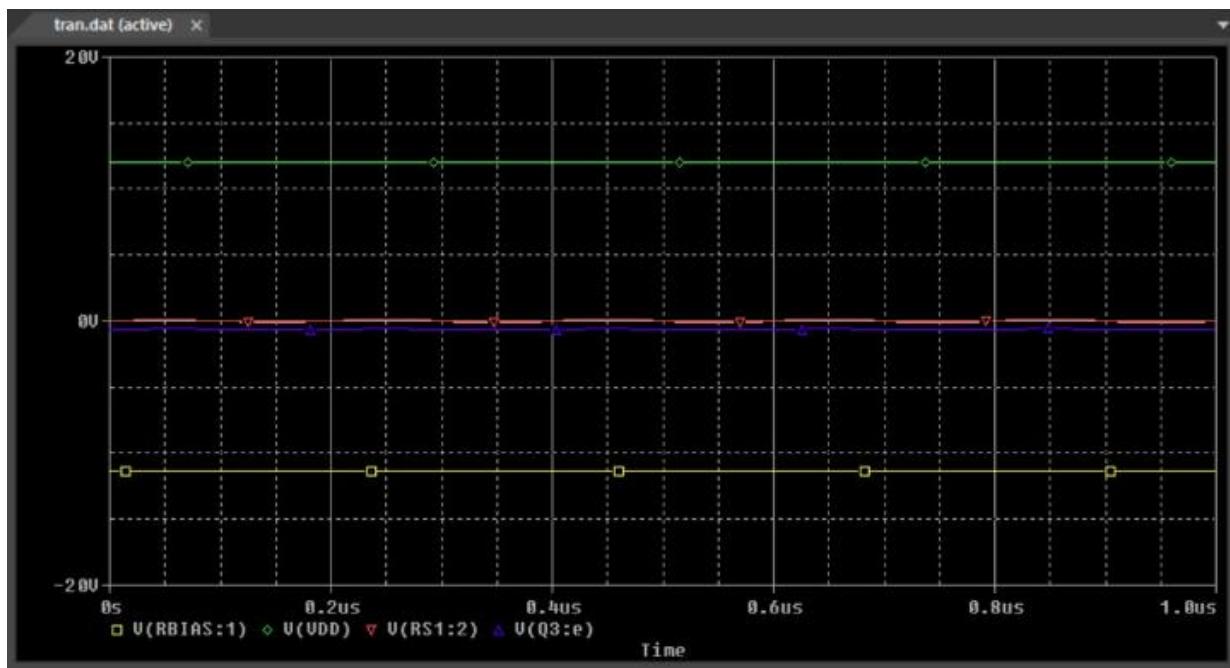
The *Markers* dialog box opens.



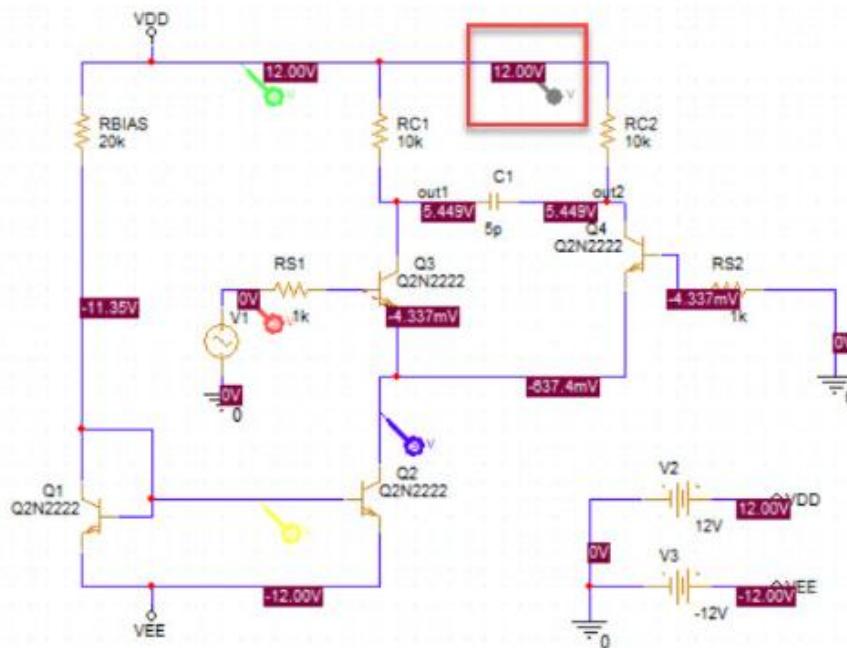
i You can also access the *Markers* dialog box by choosing *PSpice – Markers – List* from the Capture main menu.

2. Deselect the extra markers.
3. Click *OK*.
4. Run PSpice icon (⌚) to start the simulation.

The simulation starts, and you can see the waveform in PSpice.



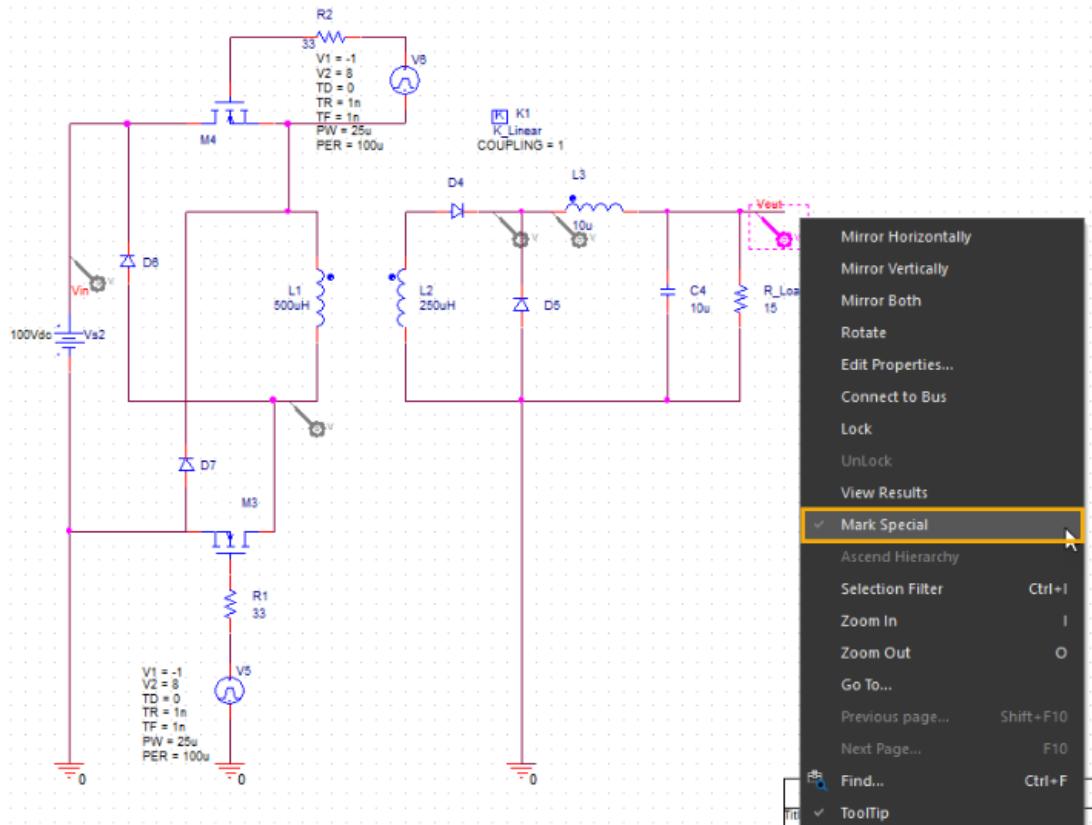
The deselected markers in the *Markers* dialog box appear in gray, as shown in the following figure:



Identify Marker as Special

You can remove any additional marker by assigning the marker as a special marker using the following steps:

1. Click *Cancel* in the message box.
2. On the schematic, select the extra markers.
3. Right-click and choose *Mark Special*.



4. Next, click the Run PSpice icon (play button) to start the simulation.

Utilizing Unused Markers for Data Collection

You can make use of the unused markers in a design, by doing the following steps:

1. Right-click any additional marker added that is marked as special.
2. Deselect *Mark Special*.
3. Repeat step 1-2 for any other unused markers.

4. At any point, ensure that the number of active markers for data collection is not more than four and then run the simulation.

Model Parameters for Specific Devices

You can edit model parameters for the following devices and plot all the waveforms for your simulation circuit.

- Passives: R, L, C, K
- Switches: S, W
- Transmission line: T
- Digital stimulus: U - STIM
- Independent sources: V, I
- Controlled sources: E, F, G, H
- Subcircuit: X

For other device types, after editing the models you can plot upto four markers.

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 , others come from libraries , 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 subcircuit 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, 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 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.

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>/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.

Physical Layout in a Layout 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 layout editors, such as OrCAD® X Presto or Allegro® X PCB Editor; the Cadence tools for designing physical layout of a PCB board.

Capture offers full integration with OrCAD X Presto and Allegro X PCB Editor tool suite, enabling you to use all of Capture schematic design capabilities to enter your PCB projects, then export the information to a layout editor for layout and routing. While the actual board design tasks are performed in the layout editor, there are a few tasks that must be performed in Capture to prepare the schematic for the layout.

This section lists the design tasks and the best practices that must be followed during the schematic capture stage to ensure that the process of exporting data to the layout editor is completed smoothly.

The tasks covered in this chapter are:

- [Preparing the Schematic for Layout](#)
- [Assigning Physical Properties to a Schematic Design](#)
- [Working with Footprints](#)
- [Instance-Level Properties for Physical Design](#)
- [Defining Properties on Nets](#)
- [Properties on Power Pins](#)
- [Assigning No Connect Pins in Capture-PCB Editor Flow](#)
- [Property Flow from Schematic to Layout Editor](#)
- [Creating PCB Editor Board](#)
- [Setting up Advanced Options for PCB Flow](#)

- Processing Changes after Board Creation (ECO)
- Cross Probing for PCB Editor
- Pin Swapping In Capture-PCB Editor Flow
- Back Annotation from PCB Editor
- Physical Layout of a Simulation Design
- Design Reuse for a Layout Editor
- Running Design Rules Check - Physical Rules
- Launching PCB Viewer from Capture

Preparing the Schematic for Layout

Before you design the physical layout of your schematic in a layout editor, you need to validate the design to ensure that the object names used in the schematic follow the object naming convention required in the layout editors. This section lists some of the recommendations or best practices to be followed in Capture to ensure that schematic is successfully exported to the layout editor.

Best Practices for Schematic-Layout Flow

- Avoid using parenthesis "()" in schematic names. If you have to use parenthesis in a schematic name, ensure that you map it with a valid character while generating a netlist and perform a reverse mapping while generating a `.swp` file.
- Naming nets, parts, or pins:
 - Keep the maximum length of a net name or alias up to 255 characters.
 - Limit part and pin names to 255 characters.If the limit of 255 characters cannot be maintained, ensure that before you generate the layout editor netlist, you modify the this limit in the *Setup* dialog box. This dialog box is invoked when you choose *Setup* in the *PCB* tab of the *Create Netlist* dialog box.
- Use only uppercase characters for part or symbol names, reference designators, and pin names. Do not use lowercase characters.

- Do not use special characters in net names, part names, reference designators, or pin names.
- Do not use 0 (zero) as a pin number.
- Do not use duplicate names for pins other than power pins.
- For multiple power pins with the same pin name, ensure that all pins are either visible or invisible.
- Avoid using the `Power Pins Visible` property at design-level.
- Use a 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.
- Ensure that there are no physical rules errors in the *Online DRC* window.
- Set path for layout editor footprint before running *Netrev*.
- Do not use `{GROUP}` as a property name in combined property strings. This may cause problems while annotating your design for a layout in the layout editor. The `GROUP` property is used in PCB Editor and OrCAD X Presto 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 a netlist for the design.
- Capture lets 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 the layout editors.

Unsupported Schematic-Layout Flow Field Values

Avoid using the following special characters when defining pin names, net names, or signal names in the the schematic-layout 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 cannot contain a period (.).

 Avoid using backslash (\) in net names and single-quote (') in pin names, as these are not considered legal characters in the Capture-PCB Editor and Capture-OrCAD X Presto flows and may fail the import logic in the layout editors.

To include backslash (\) in the legal character set, set the environment variable *legacy_character_set* as 1 in the command window.

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 of some of the properties are dumped in the netlist. This is then read by the layout editor for board creation.

 Only the properties that have values assigned to them and are listed in the configuration file are exported to the layout editor. To know more about property flow from Capture to PCB Editor or OrCAD X Presto, see [Property Flow from Schematic to Layout Editor](#).

This section lists the properties specific to the schematic-layout design flow and can be assigned in Capture.

There are several ways to make property assignments if the parts are already placed in a design:

Using Part Editor

To assign properties to the existing parts in a design from part editor, do the following:

1. In Capture, select the part to which properties are to be added.
2. Right-click and choose *Edit Part* to open the part editor.

3. Add or update part or package properties in the *Property Sheet* panel.

Using Property Editor

To assign properties from the Property Editor window, do the following:

1. Right-click a part on the schematic and choose *Edit Properties*. Alternatively, double-click the part.
2. Use the Capture-Allegro filter in Property Editor to view typical properties that may be assigned in Capture to be used in Allegro layout editors, such as PCB Editor and OrCAD X Presto.

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.

This list of properties comes from the `PREFPROP.TXT` file, placed in the `<Cadence_Installation_directory>\tools\capture` during installation. For information about the Allegro layout editor properties, see the [Allegro Platform Properties Reference](#) guide.

- ⓘ 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.

Working with Footprints

To map a logical component correctly 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](#).

Custom Footprints in Design Flow

Cadence ships a library of footprints and padstacks with the installation of layout editors, PCB Editor and ORCAD X Presto. These can be accessed from

`<installation_hierarchy>/share/pcb/pcb_lib/symbols`.

If you need to use custom footprints or padstacks, you can copy them at

`<installation_hierarchy>/share/local/pcb/symbols` or

`<installation_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 setup of the layout editors.

If you do not want to copy the footprints, you need to modify the layout editor setup to include the location of custom footprints and custom padstacks in `psmpath` and `padpath` variables, respectively.

Complete the following steps in PCB Editor:

1. Select *Setup – User Preferences* or use the `enved` command.
The *User Preferences* dialog box opens.
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.

Complete the following steps in OrCAD X Presto to add the libraries:

1. Choose *Edit – Preferences*.
The *Preferences* dialog box opens.
2. Select the *Directories & Paths* folder under *Libraries*.
3. Ensure the padstack library paths are set for both `.PSM` and `.PAD`.

 **Important**

The relative path specified in `psmpath` are in relation to the board file. This path is read by Capture if the board file is present in the Capture project.

In this section:

- [Assigning Footprint Properties](#)
- [Viewing Footprints](#)

Assigning Footprint Properties

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 the *Property Editor* window using one of the following methods:

- Double-click the part.
- Right-click the part and select *Edit Properties*.

The footprint information is displayed corresponding to 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, do the following:

1. Select *Tools – Design Rules Check*.
The *Design Rules Check* dialog box opens.
2. Click the *Physical Rules* tab.
3. Select the *Check missing/illegal PCB Footprint property* check box.
4. Click *OK*.

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, perform the following tasks:

1. In the project manager tab, select the design file.
2. Select *Tools – Bill of Materials*.
The *Bill of Materials* dialog box opens.
3. Modify the following two default entries in the *Line Item Definition* section.
 - In the *Header* text box, append `\tFootprints`.
 - In the *Combined property string* text box, append `\t{PCB_footprint}`.

4. Click *OK* to 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.

To assign footprint to one component using *Property Editor*, do the following steps:

1. Right-click the required part in the schematic and choose *Edit Properties* to launch *Property Editor*.
2. Specify the required value corresponding to the `PCB Footprint` property.

To assign the same footprint to multiple components, do the following:

1. Select the components.
2. Right-click and choose *Edit Properties* to launch *Property Editor*.
3. Right-click the `PCB Footprint` row (or column), and select *Edit*.
The *Edit Property Values* dialog box opens.
4. Specify the required value.
5. Click *OK*.

To create a board in PCB Editor or OrCAD X Presto, you need to assign a valid `PCB Footprint` property for each part in the design. This includes all the 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 the schematic-layout 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 Capture-PCB Editor and Capture-OrCAD X Presto flows. As a result, you can define different `PCB Footprint` properties for different PCB flows.

⚠ To include a zero pin mechanical part in the Capture-PCB Editor or OrCAD X Presto flow, add a property `CLASS` with its value as `MECHANICAL` to the part, that is `CLASS = MECHANICAL`, with an appropriate PCB footprint.

Specifying Alternative Footprints

You can specify alternative footprints using one of the following two methods:

- Using the `ALT_SYMBOLS` property
- Using the `FPLIST` property

Using ALT_SYMBOLS Property

While specifying component footprint, the `ALT_SYMBOLS` property can be used to specify a list of alternative footprint names that can be used to substitute the primary footprint during interactive placement in PCB Editor or OrCAD X Presto. 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:

```
ALT_SYMBOLS '(Subclass:Symbol,...;Subclass:Symbol,...)'
```

Where:

- `Subclass` can either be TOP (or T) for top layer, or BOTTOM (or B) for bottom layer
- `Symbol` is the standard footprint name

For example, you can assign the `ALT_SYMBOLS` value: `T:dip14_3; B:dip14_3` to the component, 7400.

To use the `ALT_SYMBOLS` property to specify a list of alternative footprints, do the following:

1. Add the `ALT_SYMBOLS` property to the component on which you want multiple footprint values.
2. Add the footprint information in the property value according to the [syntax](#).
3. Update the `allegro.cfg` file with the following information under the `section, [ComponentDefinitionProps]`:

```
ALT_SYMBOLS = YES
```

Using FPLIST Property

The `FPLIST` property can also be used to specify a list of alternative footprints in Capture. This list substitutes the primary footprint during interactive placement in the layout editor. The `FPLIST` property can be assigned to a package or to a component either at the library level or at the instance level.

To use the `FPLIST` property to specify a list of alternative footprints, do the following:

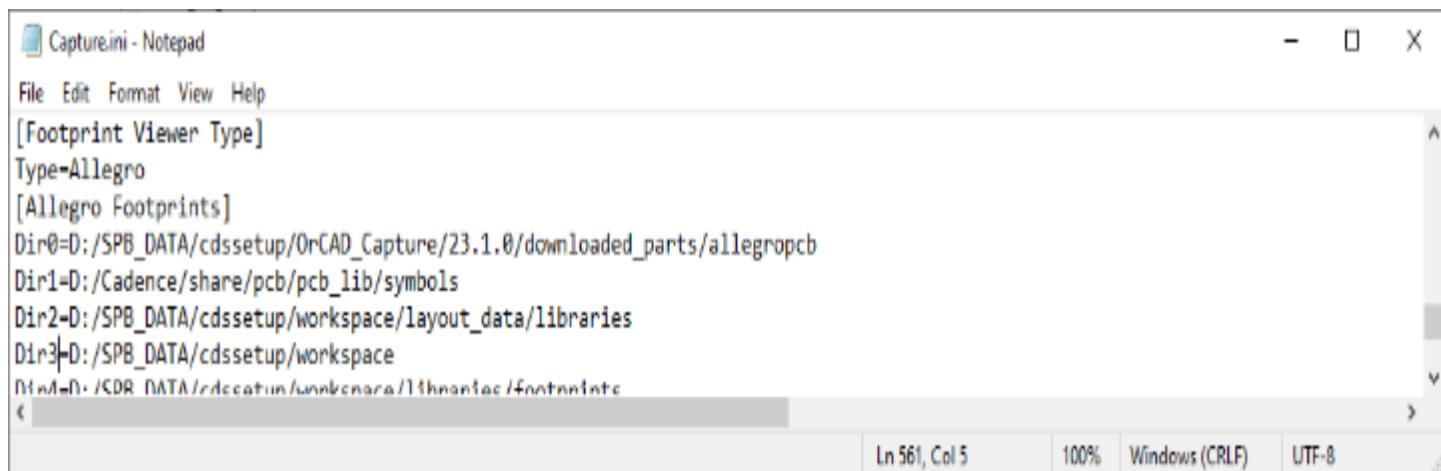
1. Add the `FPLIST` property to the component on which you want multiple footprint values.
2. Add the footprint information in the property value.
You can separate the footprints by commas or use the syntax explained in the previous procedure, *Using ALT_SYMBOLS Property*.
3. Update the `allegro.cfg` file with the following information under the section, `[ComponentDefinitionProps]` to map the `FPLIST` property to the `ALT_SYMBOLS` property.

```
FPLIST = ALT_SYMBOLS
```

⚠ Before adding `FPLIST = ALT_SYMBOLS`, ensure that `ALT_SYMBOLS = YES` is specified in the `allegro.cfg` file.

Viewing Footprints

The footprint location in the installation is used to configure the `Capture.ini` file 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 `Capture.ini` is shown in the following example.



A screenshot of a Windows Notepad window titled "Capture.ini - Notepad". The window contains the following configuration file content:

```
File Edit Format View Help
[Footprint Viewer Type]
Type=Allegro
[Allegro Footprints]
Dir0=D:/SPB_DATA/cdssetup/OrCAD_Capture/23.1.0/downloaded_parts/allegropcb
Dir1=D:/Cadence/share pcb/pcb_lib/symbols
Dir2=D:/SPB_DATA/cdssetup/workspace/layout_data/libraries
Dir3=D:/SPB_DATA/cdssetup/workspace
Dir4=D:/CDR DATA/redcadun/unknwnsrc/1thnntac/footprntac
```

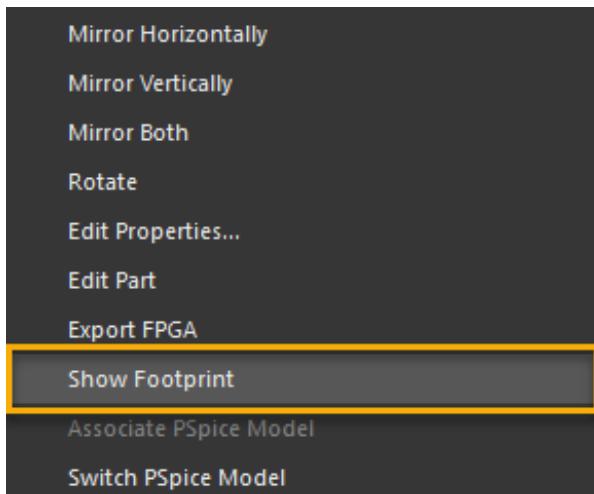
The window shows standard Notepad UI elements like a menu bar, scroll bars, and status bar at the bottom indicating "Ln 561, Col 5" and "Windows (CRLF) UTF-8".

If you are using the custom footprints, `Capture.ini` must be updated such that `Dir0`, `Dir1`, `Dir2`, ... variables point to the folder containing the footprints. This section of the `Capture.ini` file is read by Footprint Viewer to display component footprint in Capture.

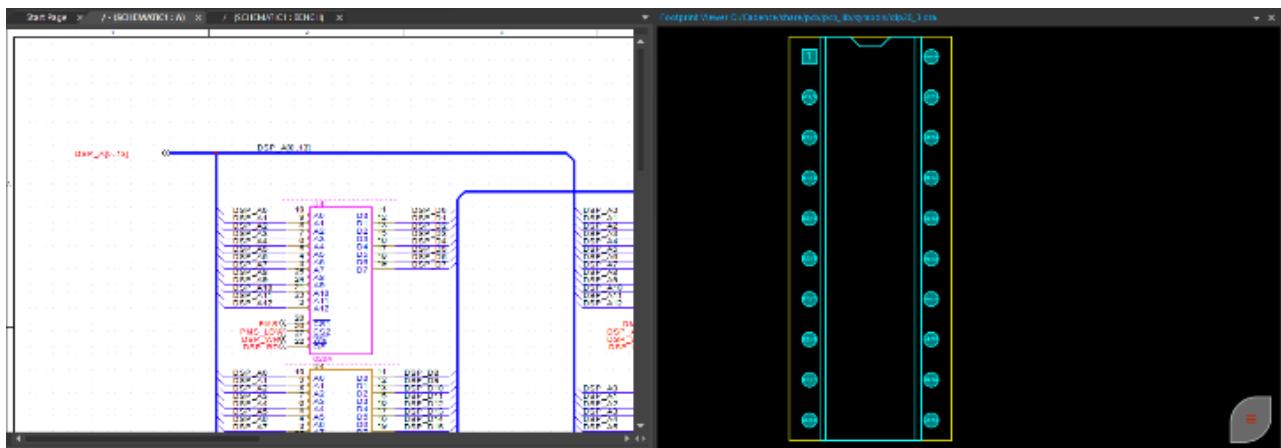
While working in Capture, you can view the footprints associated with the part in the Footprint

Viewer window. This window provides a two-dimensional view of the footprint symbol of a selected part on the schematic. Along with the footprint symbol, it also shows pin numbers and pin names. The Footprint Viewer window is available from within the schematic. To view the footprint:

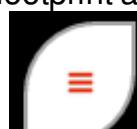
1. Right-click the part.
2. Choose *Show Footprint*.



The Footprint Viewer window opens with the two dimensional view of the footprint corresponding to the selected part.

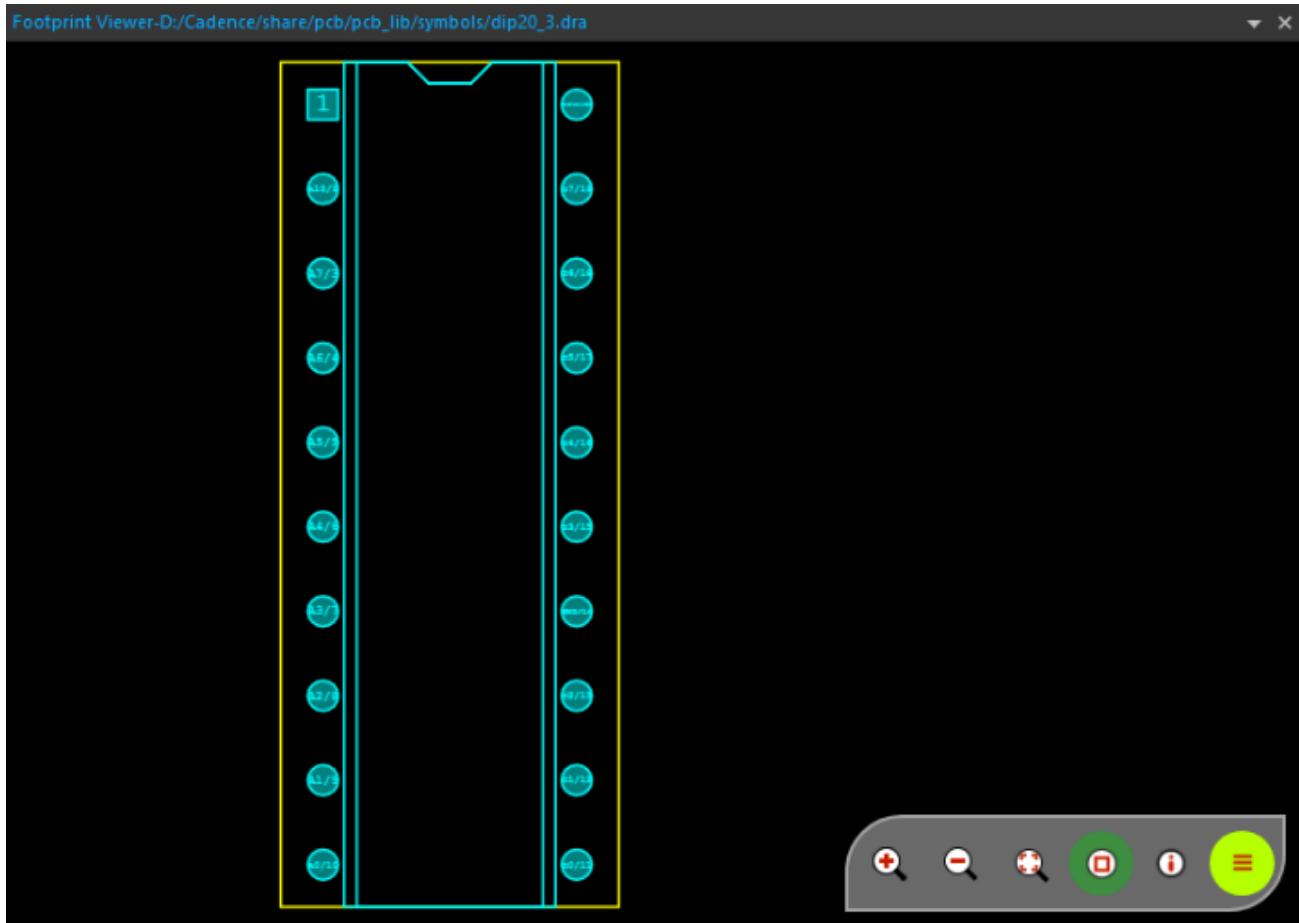


The Footprint Viewer window can be used to display footprint and pin information. To view the



details of the footprint, click the footprint viewer menu (). The menu expands and

shows various icons.



Each icon performs a specific function as explained in the following table:

Icons in the Footprint Viewer menu	Click this icon to...
	View or hide information about the footprint. The details are displayed at the top of the viewer.
	View or hide information about pins.
	The zoom scale is adjusted to fit the footprint in the footprint viewer window.
	The current zoom scale is divided by the zoom factor to reduce the size of the footprint view.



The current zoom scale is multiplied by the zoom factor to magnify the footprint view.

Mapping Symbol Pins with Footprint Pins

Footprint Viewer also supports cross-probing between the schematic design and the footprint view. When you select a symbol pin on the part, the corresponding pin along with pin details are highlighted in the footprint viewer.

Troubleshooting Footprint Viewing Errors

Problem: Footprints not displaying in the footprint 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 property defined in the package properties of the part.

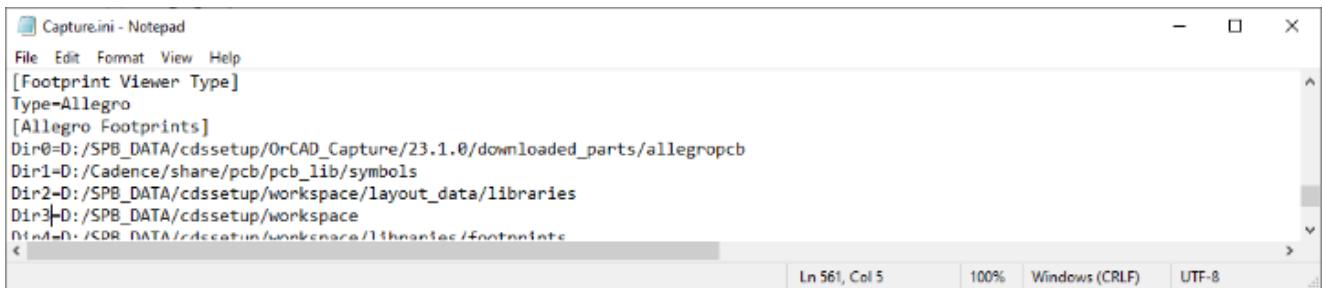
If none of the above properties or values are found, the viewer displays a blank canvas with an error. See the *Session Log* window for the list of errors.

- *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` file to ensure that it has a section named `[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` file is shown in the following figure:



For details on creating and editing the variables (including the Allegro Footprint variables) in `Capture.ini`, see [Capture.ini Variables](#).

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.

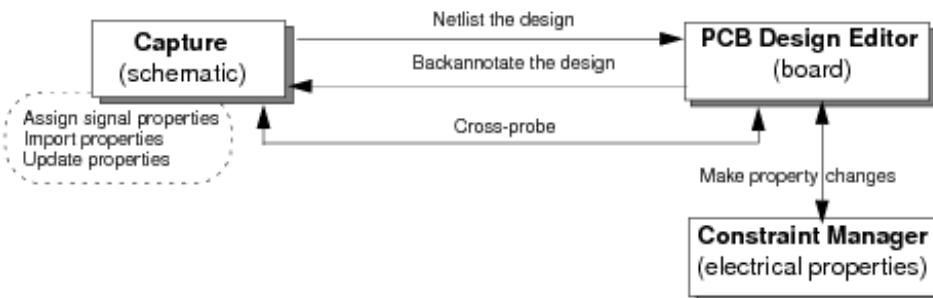
ROOM Property

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 the layout 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

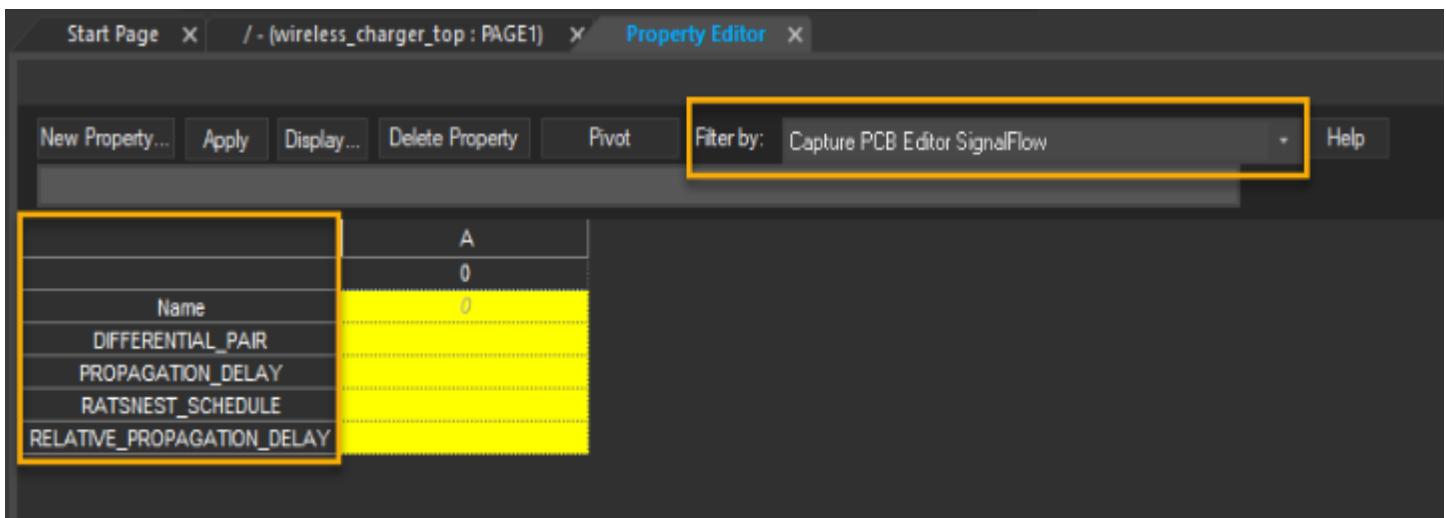
While designing a schematic in Capture, you can specify high-speed constraints as net properties and take them through a complete front-to-back flow. As net properties are passed to the physical netlist generated by Capture, these constraints are also passed to the layout editors. In the layout editor, you can modify these constraints by launching Constraint Manager. The following figure shows the flow of signal properties.



⚠ To enable cross-probing between OrCAD X Capture and Constraint Manager, enable highlighting in the layout editors, PCB Editor or OrCAD X Presto. When you highlight a component in a layout editor, it sends cross-probing messages to OrCAD X 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 box.

To view the list of high speed signal properties supported by Capture, open the *Property Editor* window, and from the *Filter by:* drop-down list, select *Capture PCB Editor SignalFlow*.

The properties are listed in the *Flat Nets* tab.



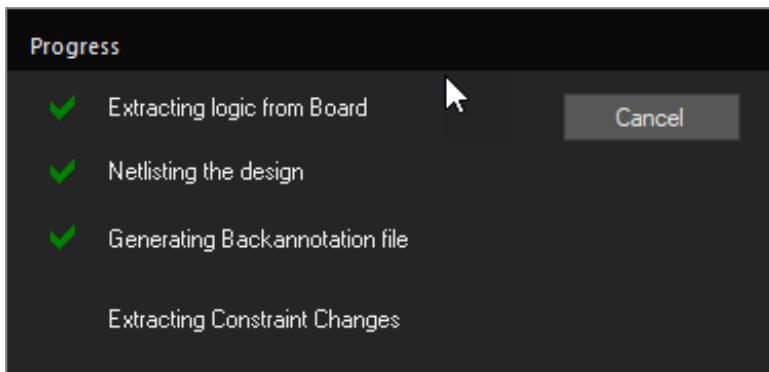
 Use the *Schematic Nets* tab to display the net properties on the schematic. Adding or modifying high-speed signal properties is supported using the *Flat nets* tab of the Property Editor.

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. Create 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.
 - DIFFERENTIAL_PAIR
 - PROPAGATION_DELAY
 - RATSNEST_SCHEDULE
 - RELATIVE_PROPAGATION_DELAY
4. Choose the *PCB – Design Rules Check* men command to perform design rule checks on the design.
5. Choose the *Tools – Create Netlist* menu command to netlist the design.

6. Right-click the board file name in the project manager window and choose *Launch PCB Editor*.
7. In the layout editor—PCB Editor or OrCAD X Presto—launch Constraint Manager.
 - a. Open the board file in the PCB Editor.
 - b. Click *Setup – Constraint* 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.
8. 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 *PCB Editor* tab, set different backannotation options and click *OK*.
The Progress window reports the details of backannotation.



- e. Open the *Property Editor* window 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 the *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 shortcut 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+ 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 choose 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 themSimilarly, 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* section.

- 
- 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.

i 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.

i 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* section.
 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, it 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.

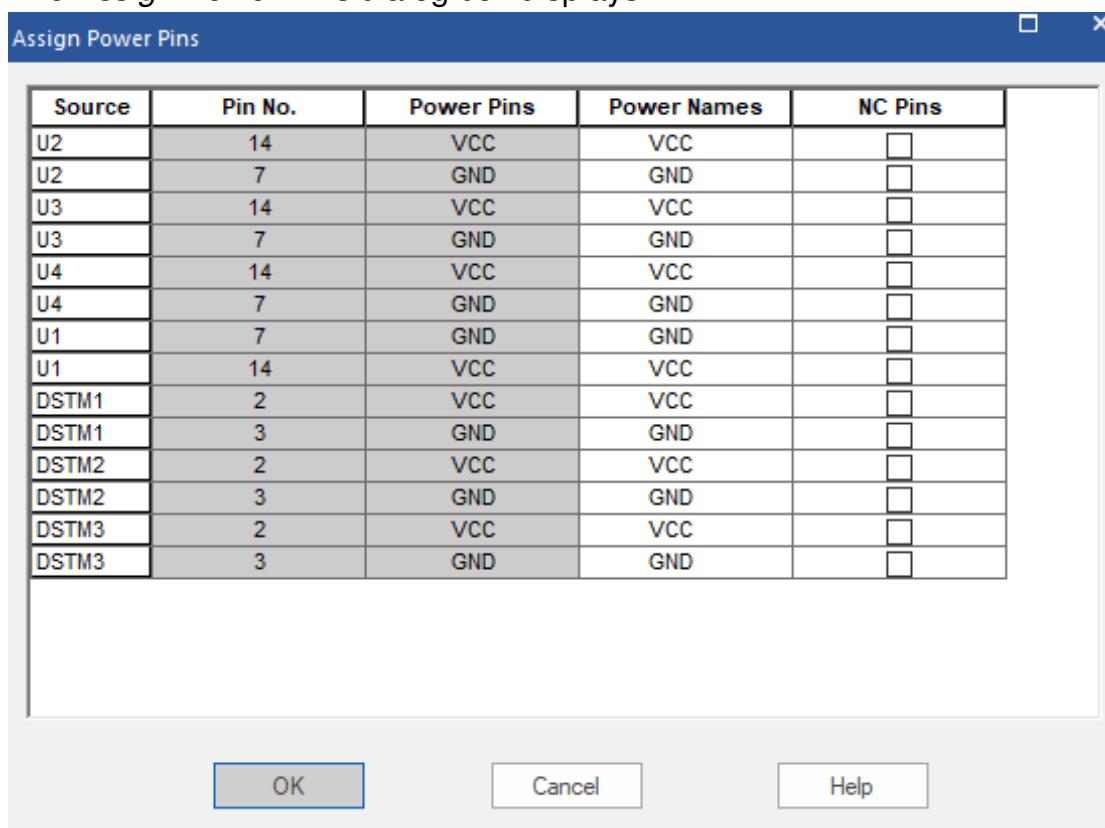
- Using Assign Power Pins command

- 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 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.

Source	Pin No.	Power Pins	Power Names	NC Pins
U2	14	VCC	VCC	<input type="checkbox"/>
U2	7	GND	GND	<input type="checkbox"/>
U3	14	VCC	VCC	<input type="checkbox"/>
U3	7	GND	GND	<input type="checkbox"/>
U4	14	VCC	VCC	<input type="checkbox"/>
U4	7	GND	GND <input checked="" type="checkbox"/>	<input type="checkbox"/>
U1	7	GND	GND	<input type="checkbox"/>
U1	14	VCC	VCC	<input type="checkbox"/>
DSTM1	2	VCC	VCC	<input type="checkbox"/>
DSTM1	3	GND	GND	<input type="checkbox"/>
DSTM2	2	VCC	VCC	<input type="checkbox"/>
DSTM2	3	GND	GND	<input type="checkbox"/>
DSTM3	2	VCC	VCC	<input type="checkbox"/>
DSTM3	3	GND	GND	<input type="checkbox"/>

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 =VPP Pin numbers 8,9=VSS Pin numbers 7,21 =GND	POWER_GROUP= VDD=VCC; VPP=VCC; VSS=VCC	POWER_PINS='(VCC:14,28,13,12,10,11,8,9)'; POWER_PINS='(GND:7,21)'

⚠ 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.

Assigning No Connect 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.

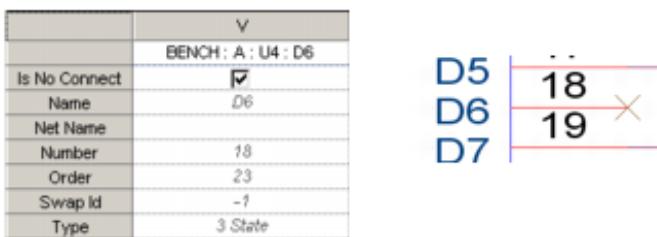
Assigning No Connect Pins

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, do one of the following to the part in your design:

1. Add the 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, an 8-pin footprint with the two through-holes being pins 7 and 8, results in a 6-pin part on the design with an NC property containing the value of 7,8.
2. Place a *No Connect* symbol on pins that you do not want to be connected to anything:
 - a. Launch Property Editor to display the component properties.
 - b. In the *Pins* tab, select the *Is No Connect* check box for the pin to be marked as NC pin.



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 the No Connect property on a part, use the *Update Cache* command to update the part in your design, if the part is missing pin numbers.

- Do not connect any nets to the non-electrical pins on the part. So, in the earlier example, there is an 8-pin part with nothing connected on pins 7 and 8.

 Combining any of these three options on the same part may lead to fatal netlisting errors.

 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` or `PSTXPRT.DAT`. Therefore, all pins connected to the `NC` net are unconnected on the PCB.

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 the `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, do the following:

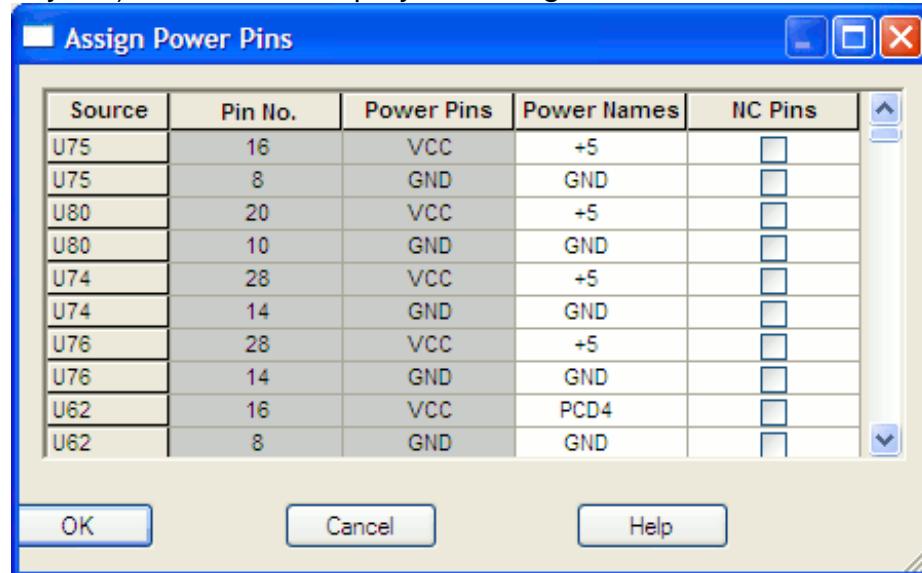
1. In the project manager, select the design, schematic folder, or schematic page that contains the power pins you need to set as NC.

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 Schematic to Layout Editor

When you netlist a Capture schematic, not all properties defined in Capture are transferred to the layout. For a property to flow from Capture to the layout 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 the layout editor or back annotated from the layout editor to Capture. If a Capture property is not included in the configuration file, it is not passed to the layout editor. Similarly, if a layout editor property is not listed in the file, it does not get back annotated to Capture.

Layout Editor Configuration File

The default configuration file installed with Capture is available at <installation_directory>\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** - The layout editor component definition properties, output in PSTCHIP.DAT file

- **ComponentInstanceProps** - The layout editor component instance properties, output in `PSTXPRT.DAT` file
- **netprops** - The layout editor net properties and all pin level properties, output in the `PSTXNET.DAT` file
- **functionprops** - The layout editor function properties, output in the `PSTXPRT.DAT` file
- **pinprops** - pin-level properties transferred between Capture and the layout editor

The following is a section of the default configuration file included in the Capture installation:

Section of allegro.cfg

```
[ComponentDefinitionProps]
ALT_SYMBOLS=YES
CLASS=YES
PART_NUMBER=YES
TOL=YES
VALUE=YES
POWER_GROUP=YES
SWAP_INFO=YES
CDS_FSP_FPGA_SYMBOL=YES
ASI_MODEL=YES

[ComponentInstanceProps]
GROUP=YES
ROOM=YES
VOLTAGE=YES
SIGNAL_MODEL=YES
NO_XNET_CONNECTION=YES

[netprops]
BUS_NAME=YES
CLOCK_NET=YES
DIFFERENTIAL_PAIR=YES
DIFFP_2ND_LENGTH=YES
DIFFP_LENGTH_TOL=YES
ECL=YES
ECL_TEMP=YES
ELECTRICAL_CONSTRAINT_SET=YES
EMC_CRITICAL_NET=YES
IMPEDANCE_RULE=YES
```

```
MATCHED_DELAY=YES
MAX_EXPOSED_LENGTH=YES
MAX_FINAL_SETTLE=YES
MAX_OVERSHOOT=YES
MAX_VIA_COUNT=YES
MIN_BOND_LENGTH=YES
...
...
VOLTAGE_LAYER=YES
CDS_FSP_NET=YES
CDS_FSP_UID=YES
CDS_FSP_BUS_INDEX=YES
NET_GROUP_GRP_NAME=YES

[functionprops]
GROUP=YES
HARD_LOCATION=YES
NO_SWAP_GATE=YES
NO_SWAP_GATE_EXT=YES
NO_SWAP_PIN=YES
ROOM=YES
CDS_FSP_IS_FPGA=YES
CDS_FSP_INSTANCE_ID=YES
CDS_FSP_TERM_NAME=YES
CDS_FSP_UID=YES
CDS_FSP_TERM_TYPE=YES
CDS_FSP_MAPPED_CELL=YES
CDS_FSP_TERM_INDEX=YES
CDS_FSP_FPGA_SYMBOL=YES

[pinprops]
NO_DRC=YES
NO_PIN_ESCAPE=YES
NO_SHAPE_CONNECT=YES
NO_SWAP_PIN=YES
PIN_ESCAPE=YES
PIN_SIGNAL_MODEL=YES
NET_SHORT=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

For a Capture design that has user-defined properties to be transferred to PCB Editor, you need to do the following:

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.
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.

The component definition properties `VALUE`, `ALT_SYMBOL`, `JEDEC_TYPE`, and `CLASS` are not back annotated as they cannot be changed in PCB Editor.

2. 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.

3. 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 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. The following table lists some PCB Editor properties where aliasing is required:

Examples of Property Aliasing

Old property name	New property name
<code>ASSIGN_TOPOLOGY</code>	<code>ELECTRICAL_CONSTRAINT_SET</code>
<code>DELAY_RULE</code>	<code>PROPAGATION_DELAY</code>

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
```

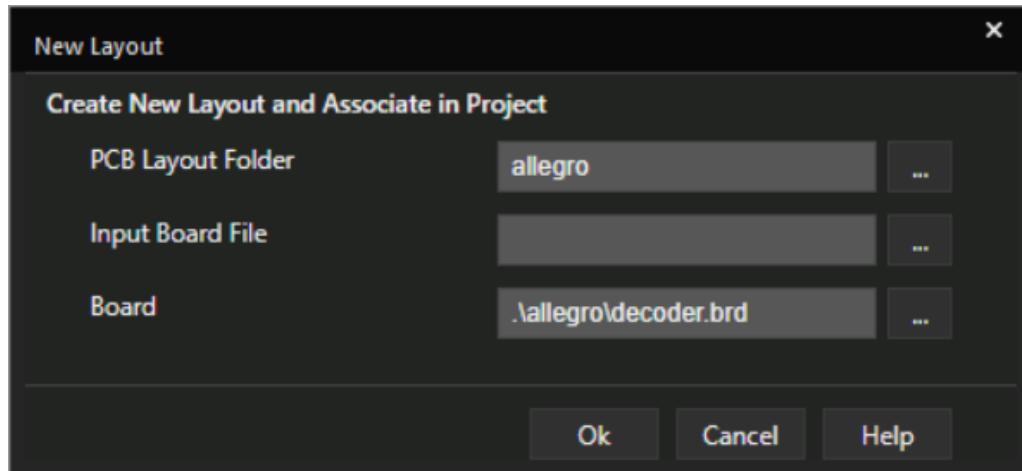
With an alias, you can have two different names for equivalent properties, one for Capture and one for PCB Editor. In the above example, `DELAY_RULE` properties get passed into the netlist as `PROPAGATION_DELAY` properties. You can include one of the above lines in the configuration file. Alternatively, you can modify your design to use `PROPAGATION_DELAY` rather than `DELAY_RULE`.

Creating PCB Editor Board

You can generate PCB Editor board by using the New Layout dialog box.

To generate PCB Editor board file, perform the following tasks:

1. Select *PCB – New Layout*.
 The New Layout dialog box opens.



2. The *PCB Layout Folder* text box shows the name of the folder in your design directory where the `*.BRD` and `PST*.DAT` files are to be saved. For a Constraint Manager-enabled design, the zip file, `pstdedb.cdsz` is created instead of PST files.

The default location is the directory named the last time this dialog box was invoked for the current design. If this is the first time the design is being netlisted, the default location will be an Allegro subfolder in your design directory. If the netlist files have been generated previously for the project, then the default is netlist directory of the board on which an operation was done the last time.

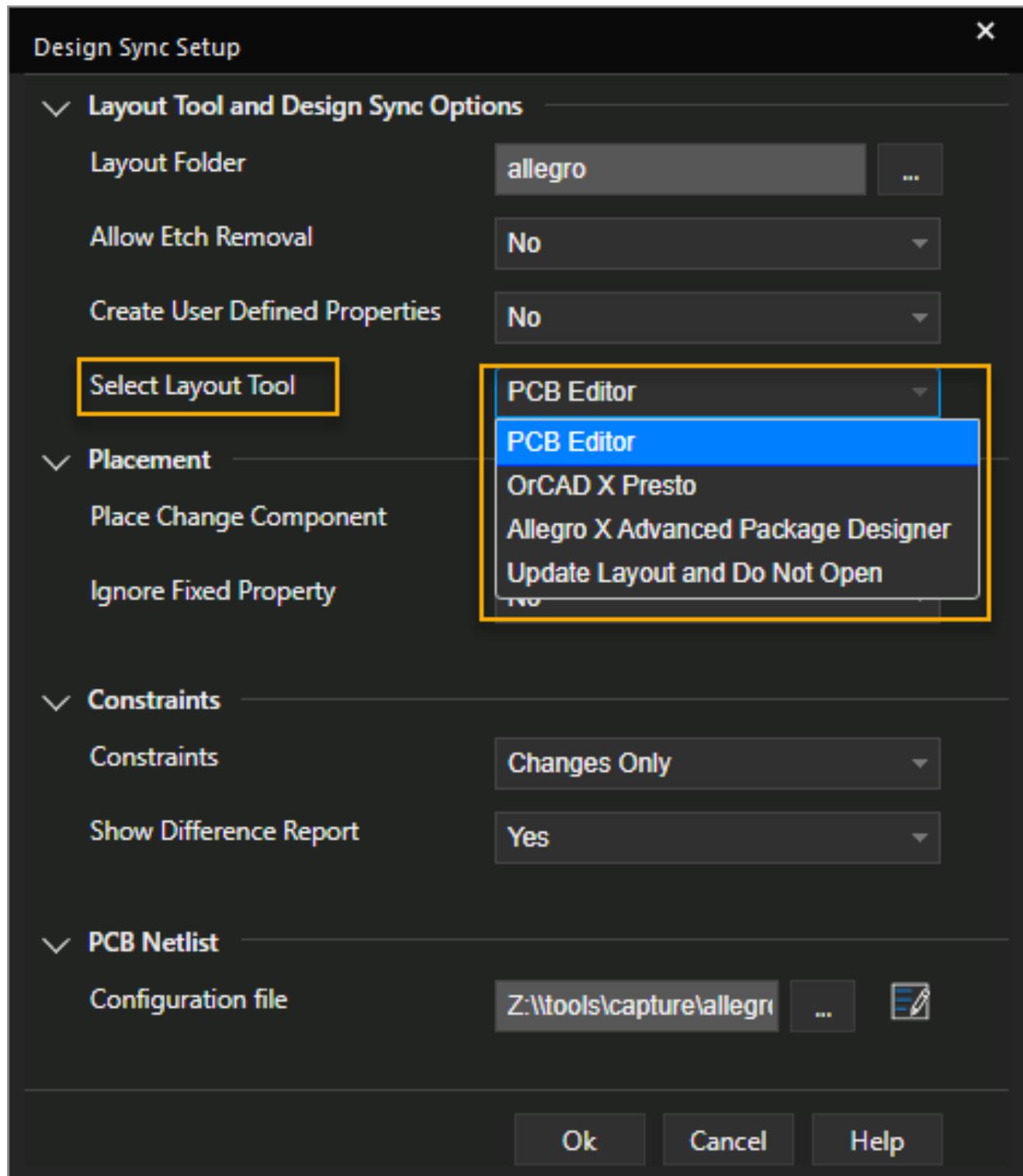
3. In the *Input Board File* text box, specify the name for an existing layout file to be used as template for generating the initial board file for the current design.

 This field is optional if you are creating a new board.

4. In the *Board* text box, specify a name for the output board 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.

5. To open the output board file immediately after the design is netlisted, ensure that the required layout tool is specified in the [Design Sync Setup](#) dialog box. The extension of the input and output board files specified in step 2 and step 3, respectively, is related to the layout tool selected in the *Design Sync Setup* dialog box.



The supported options are:

- *PCB Editor*: Generates the physical layout file with the .brd extension and opens the

board file in PCB Editor.

- *OrCAD X Presto*: Generates the physical layout file with the `.brd` extension and opens the board file in OrCAD X Presto.
- *Allegro X Advanced Package Designer*: Generates the physical layout file with the `.mcm` extension and opens it in *Allegro X Advanced Package Designer*.
- *Update Layout and Do Not Open*: Generates the board file in the path specified in step 3, but no application is launched.

6. Click *OK* to close the *New Layout* dialog box and create and open the `.brd` file.

On successful netlisting, blank board file is opened in the selected layout editor. You can now place the parts and route your ratsnest.

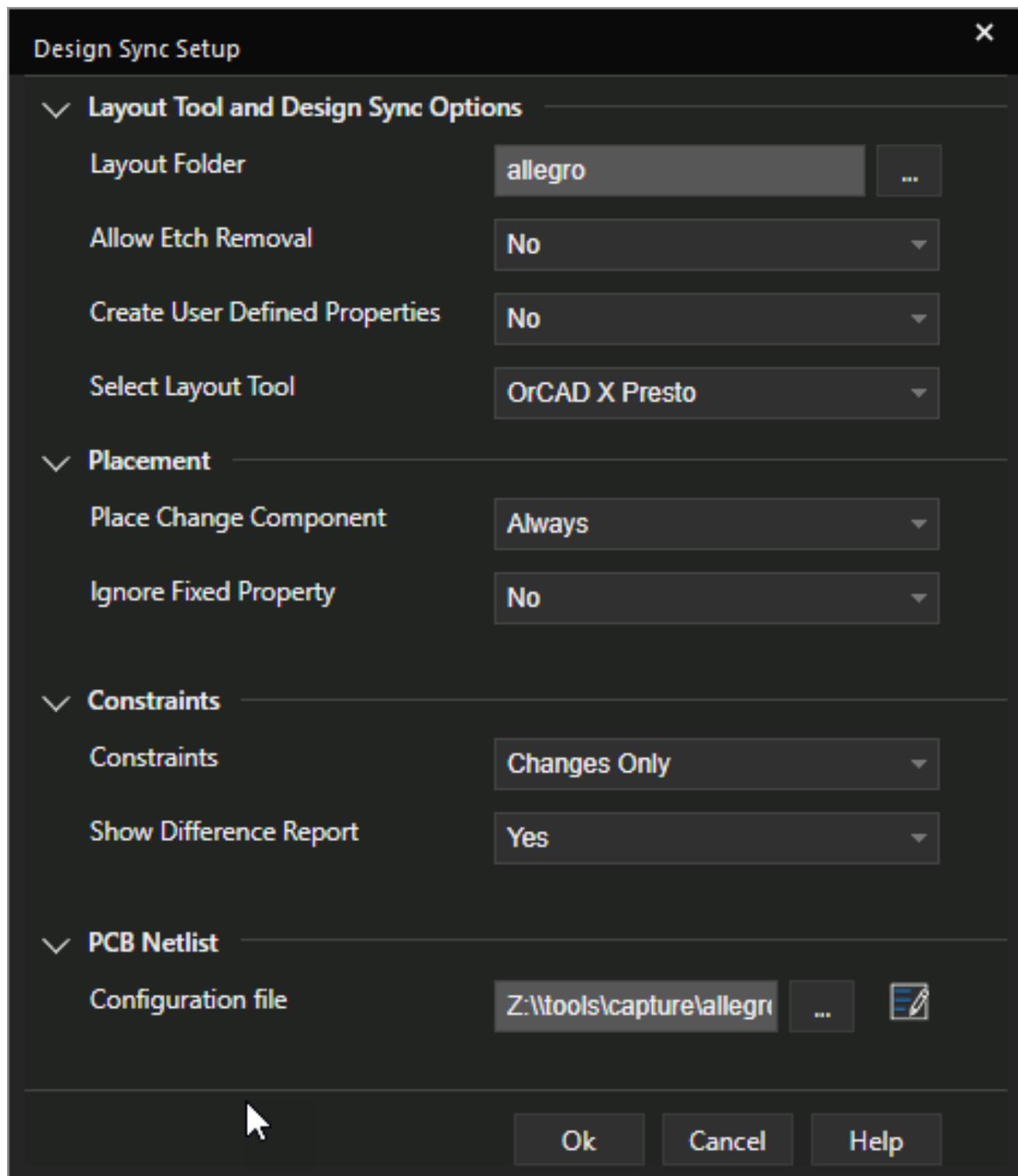
 Yellow triangles in the ratsnest indicate unrouted, zero-length connections (connections that lead directly from a pad on the top layer to a pad on the bottom layer). These connections need to be routed using a via.

Setting up Advanced Options for PCB Flow

Use the *Design Sync Setup* dialog box to set up options related to the layout tool, PCB netlist, and the *Update Layout* and *Update Schematic* dialog box.

To set up these options, do the following:

1. Choose the *PCB – Design Sync Setup* menu command. Alternatively, click  in the *Update Layout* or *Update Schematic* dialog box.
The *Design Sync Setup* dialog box opens.



- Specify the required preferences.
For details about each option, see [Design Sync Setup dialog box](#).

3. Click *Ok*.

Processing Changes after Board Creation (ECO)

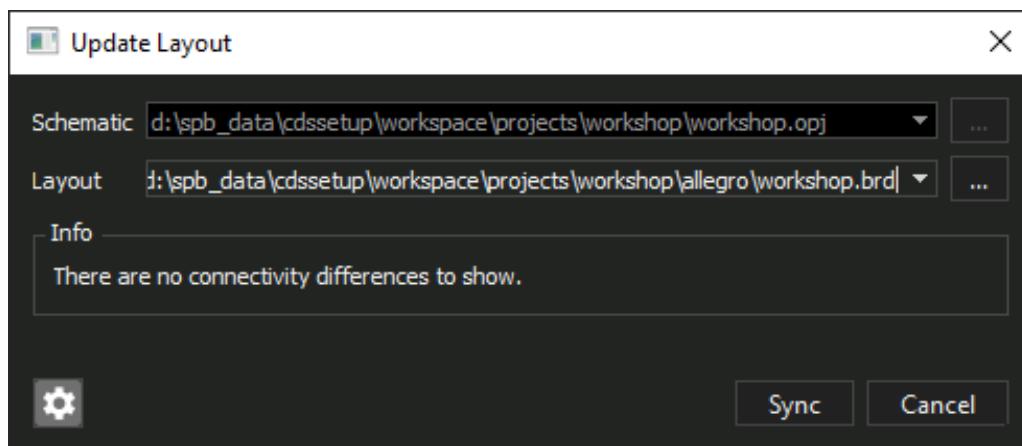
You can view the differences with respect to the selected board, and synchronizes the layout from schematic using the *Update Layout* dialog box. It shows the type of change, addition, modification, or removal of a design object.

To review design connectivity changes in real-time and synchronize the layout from schematic, do the following tasks:

1. Select *PCB – Update Layout* or click .

Alternatively, right-click the board file under the *Layout* folder of project manager, and select *Update Layout*.

The *Schematic* drop-down displays the location of the project file.



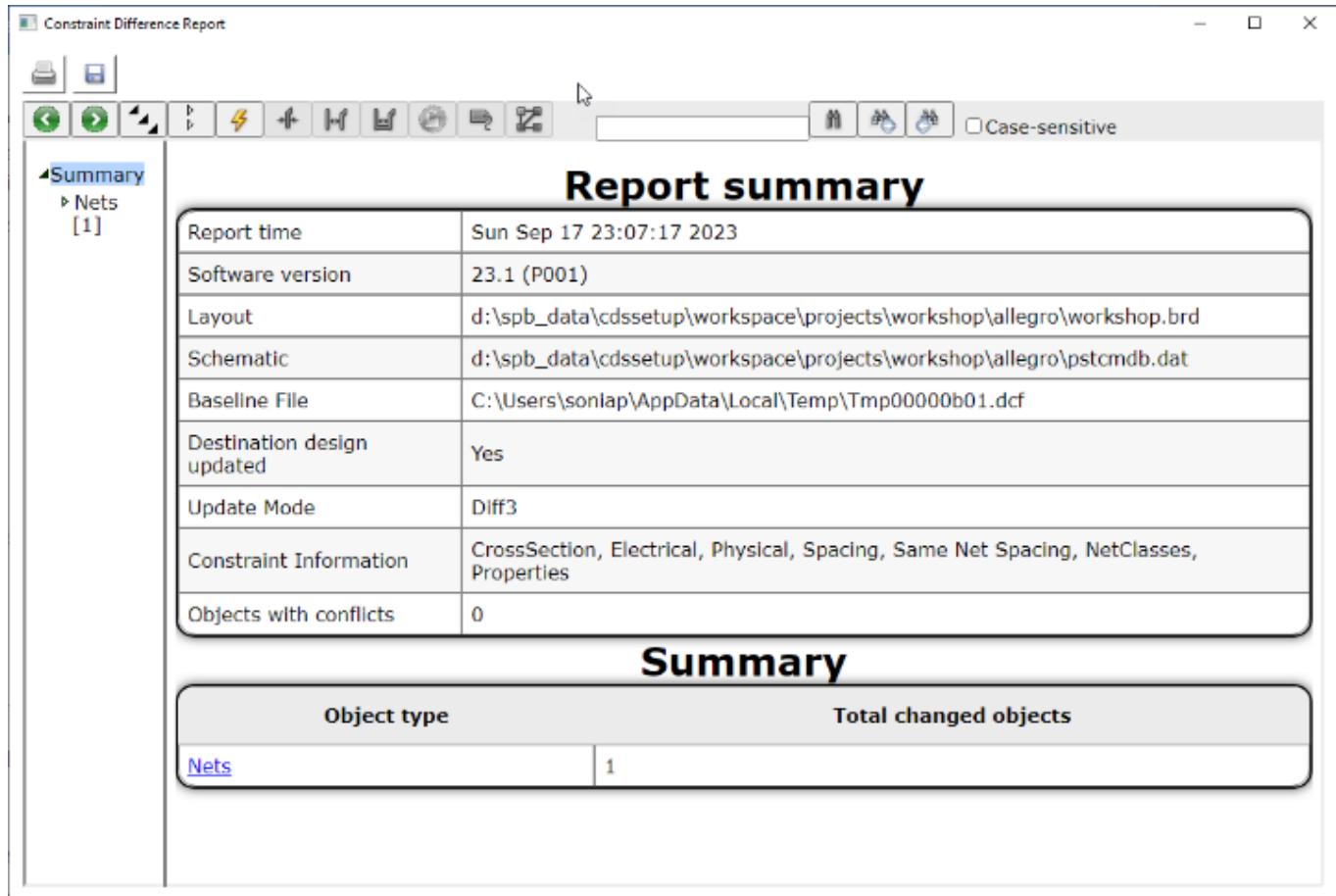
2. The *Layout* drop-down list displays the location of the board file.
If there are multiple board files, click the drop-down list to select the required board file.
3. The *Preview* section displays the all the differences from Schematic to Layout:
 - o Change Type
This can include Gate Swap, Ref Des, Pin Swap, Component Property, Net Property, Pin Property, Component, Net, and Connection.
 - o Object
 - o Action

- New Value

- Old Value

4. Click the *Sync* button.

When you click this button, forward annotation of changes takes place from Capture to the layout editor—PCB Editor or OrCAD X Presto. The Constraint Difference Report is published.



You can view the differences with respect to the selected schematic, and synchronizes the schematic from layout in the *The Update Schematic* dialog box. It shows the type of change, addition, modification or removal of a design object.

To review design connectivity changes in real-time and synchronize the schematic from layout, do the following tasks:

1. Select *PCB – Update Schematic* or click

Alternatively, right-click the board file under the *Layout* folder of project manager, and

select *Update Schematic*.

2. The *Schematic* drop-down displays the location of the project file.
3. The *Layout* drop-down list displays the location of the board file.
If there are multiple board files, click the drop-down list to select the required board file.
4. The *Preview* section displays the all the differences from layout to schematic:

- Change Type

This can include Gate Swap, Ref Des, Pin Swap, Component Property, Net Property, Pin Property, Component, Net, and Connection.

- Object

- Action

- New Value

- Old Value

5. Click the *Sync* button.

When you click this button, changes in the PCB Editor board are back annotated to the Capture schematic to ensure the physical board design is consistent with the logical schematic design.

Cross Probing for PCB Editor

After creating the board file, you place and route the board. This includes placing the parts in PCB Editor or OrCAD X Presto, Allegro X Advanced Package Designer and routing the nets.

Sometimes, you may also require to swap pins or sections or 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 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 *Options – 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 is 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, 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 the PCB Editor highlight mode is active. 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 pad

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 pad

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 pin

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 pin

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 the board.

In many cases, you create elaborate designs 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 cross-probing, 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. To unswap a pin, you can delete the PinGroup property in the Property Editor window. 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.

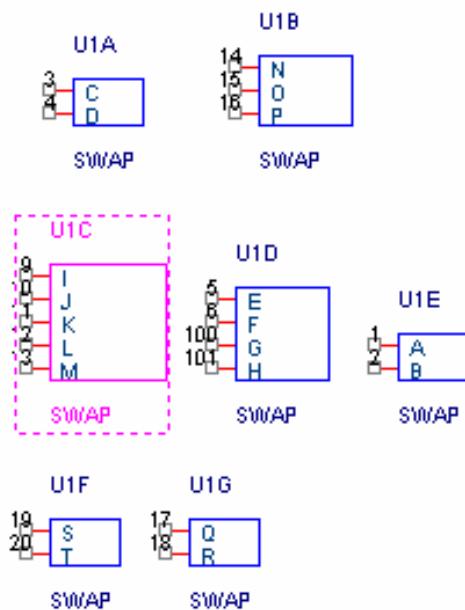
⚠ If you have a customized PCB Editor configuration file (allegro.cfg) for your Capture PCB Editor netlister, you need to add the following entry under the [ComponentDefinitionProps] section in the Allegro.cfg file:

[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+...Sn). 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.

 Since Capture 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 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 pxIBA.txt file to decide which properties need to be written into the *VIEW.DAT files. The pxIBA.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 pxIBA.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 pxIBA.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 pxIBA.txt file from the location where the board file resides. If the pxIBA.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 pxIBA.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 pxIBA.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	<p>Properties defined or changed in PCB Editor are back annotated to Capture, provided the properties are listed in the configuration file.</p> <p>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.</p> <p>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).</p> <p>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.</p> <p>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.</p> <p>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.</p>
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	<p>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.</p> <p>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.</p>
PCB Editor Board File	<p>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.</p> <p>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.</p> <p>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.</p>
Netlist Directory	<p>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.</p> <p>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.</p> <p>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.</p>
Output File	<p>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.</p>

Back Annotation	<p>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.</p> <p>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 PCB Editor tab when you rerun the back annotation.</p> <p>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</p>
------------------------	---

 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.

1. Assign appropriate footprints to components.
For details, see the section on [Working with Footprints](#).
2. Add Board only Components.
3. 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.

4. For this, add the PSPICE_ONLY on the simulation only components and set the property value to TRUE.

- a. The `PSPICE_ONLY = TRUE` property is added by default on all components from libraries such as, `STIMULUS.OLB`, and `ANALOG.OLB`.
 - b. Voltage Sources in Cadence supplied Source library already have this property assigned.
5. 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)`.

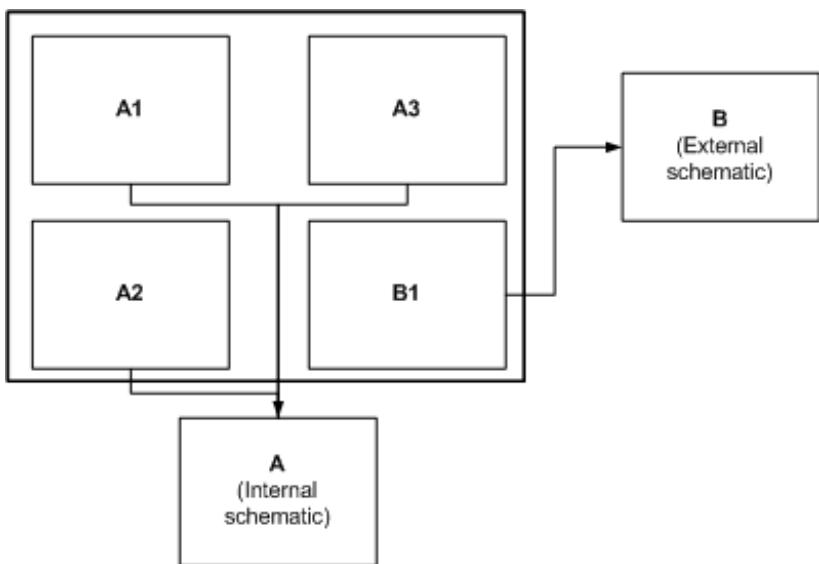
Design Reuse for a Layout Editor

Design reuse is the capability of defining and using design modules, both in the Capture and the layout editor environments, that are compatible between both of these PCB design tools. You can create modules for reuse in your design in either the schematic or in the layout 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 `.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 OrCAD X Presto or 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 OrCAD X Presto and 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

OrCAD X Presto or 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 OrCAD X Presto or PCB Editor and update the board and reuse module from within OrCAD X Presto or PCB Editor.
2. Open MID.DSN and, again, netlist the revised design with connectivity changes to the layout editor. In the layout 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 the layout editor. After the design is in OrCAD X Presto or 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 the layout 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 OrCAD X Presto or 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 OrCAD X Presto or 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 a netlist by choosing the *Create Netlist* command from the Tools menu, select the PCB 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 the layout 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	<p>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</p> <p>The REUSE_NAME property is propagated down throughout the design hierarchy to all parts below.</p>
REUSE_INSTANCE	<p>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.</p> <p>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></p> <p>Like the REUSE_NAME property, the REUSE_INSTANCE property is propagated down throughout the design hierarchy to all parts below.</p>
REUSE_MODULE	<p>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.</p>

Reusing a 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.
- 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 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 or OrCAD X Presto, 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.

To run design rule checks, do the following:

1. In the project manager, select the schematic design (*.dsn).
2. Choose *PCB – Design Rules Check*.
The Design Rules Check dialog box appears with the *Options* tab displayed.
3. In the *Options* tab, ensure that *Run on Design* is selected in the *DRC Action* field.

4. In the *Rules Setup* tab, select the *Physical Rules* check box.
5. Under the the *Physical Rules* section, select the various physical rules to be checked on your design.
6. If required, use the options available in the *Custom DRC* section, to select custom DRCs on your design.
7. In the *Report Setup* tab, under the *Physical DRC Reports* section, select the reports you want to view.
8. Click *Run*.

The settings are saved, selected DRC rules (batch and online) are executed, and the results are displayed in the *DRCs* tab.



- If you only click the *Apply* button, the list of DRC errors in the *Online DRC* window are refreshed, and the DRC options are saved.
- If you run 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` is selected before running the design rules check.
- Capture creates a report (.DRC) of warning and error messages under the *Outputs* folder of project manager. You can view the report in a text editor. These messages also appear in the *Session Log* tab.

In addition to the report, the Design Rules Check tool places error and warning markers on the schematic pages as well.

If `DRC Window` is selected in the *Show DRC Output* field, then all the design rule violations are listed in the DRCs window in the output area. You can browse the DRCs using this list displayed in this tab. This list is created and displayed whether or not `Create DRC Markers` is selected in the *Warning* field.

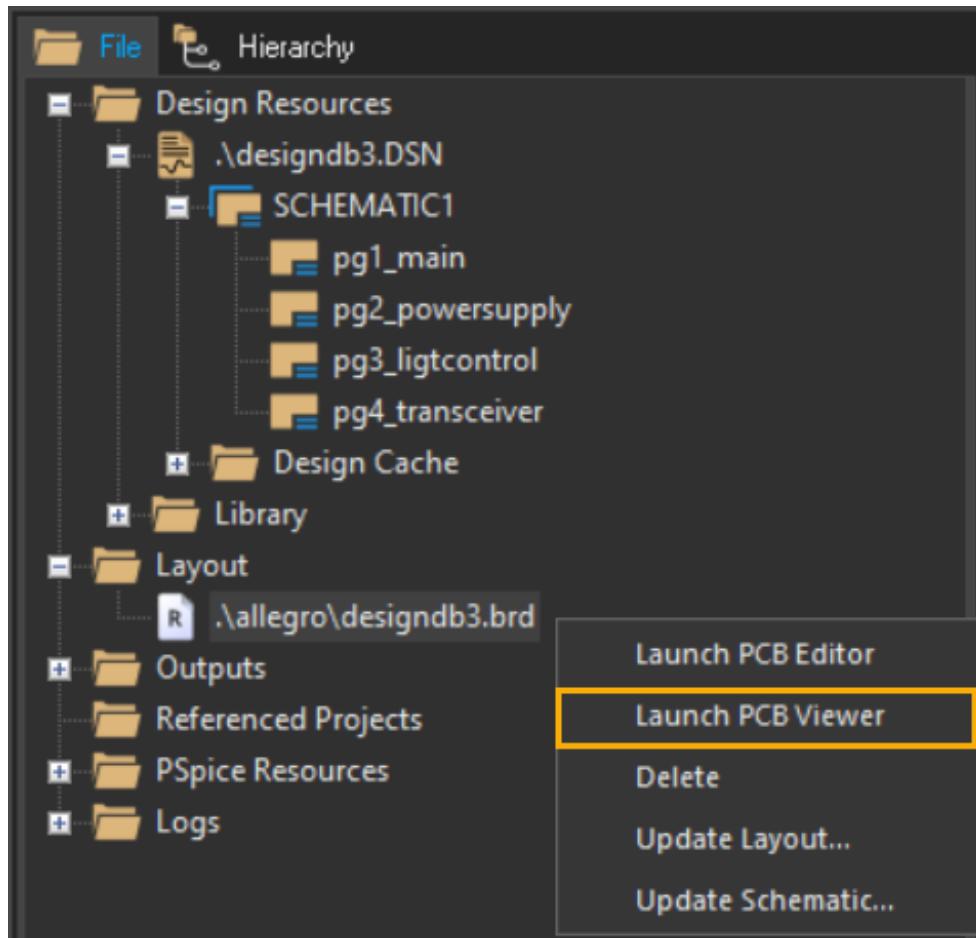
To browse DRC markers

To locate a DRC marker in the schematic, perform the following steps:

1. Choose *Edit – Browse – DRC Markers*.
DRC markers are listed in the *BROWSE DRC markers* tab.
2. Double-click the DRC marker.
The schematic page editor opens with the marker displayed and highlighted.

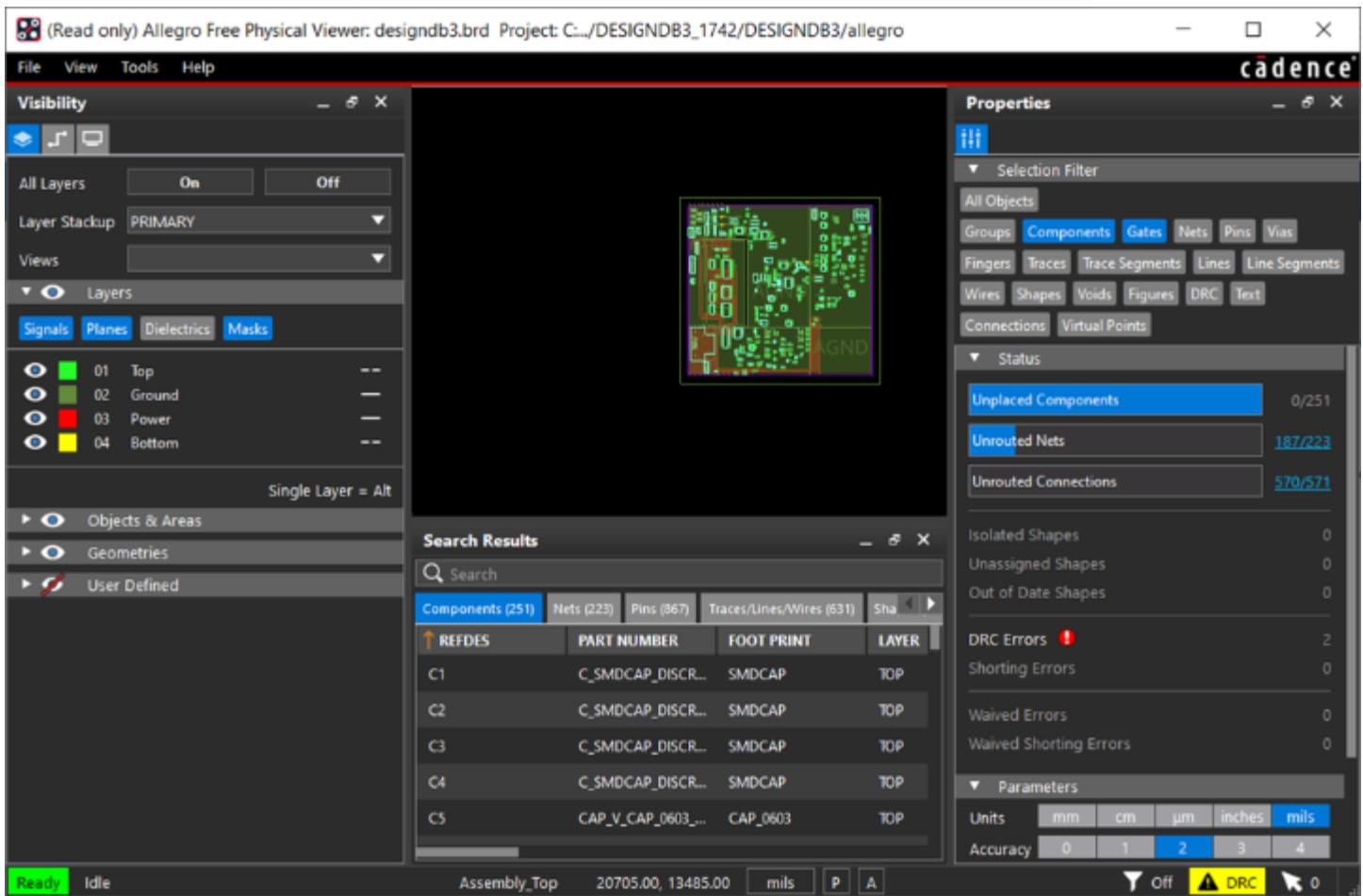
Launching PCB Viewer from Capture

You can directly open the Allegro Free Physical Viewer from within Capture. From the project manager, right-click the layout file and select Launch PCB Viewer.



The Allegro Free Physical Viewer opens showing the selected board file.

OrCAD X Capture User Guide
Physical Layout in a Layout Editor--Launching PCB Viewer from Capture



Using Capture with PCB SI

The OrCAD X 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 SigXplorer on a flat net from OrCAD X Capture to perform SI analysis and associate the Electrical Constraint set (Electrical Cset) to the flat net back to Capture from SigXplorer. The complete topology file is also embedded into the DSN. Capture also supports a distributed design environment for SI analysis by enabling you to export the net connectivity as topology files that can be updated using SigXplorer and then imported to Capture.

In Capture, you can set up SI libraries, assign SI models and then explore the signals in SigXplorer. 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 Constraint Manager and perform audit on the files.

 All Signal Integrity tasks are available under the *SI Analysis* menu in OrCAD X 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](#)

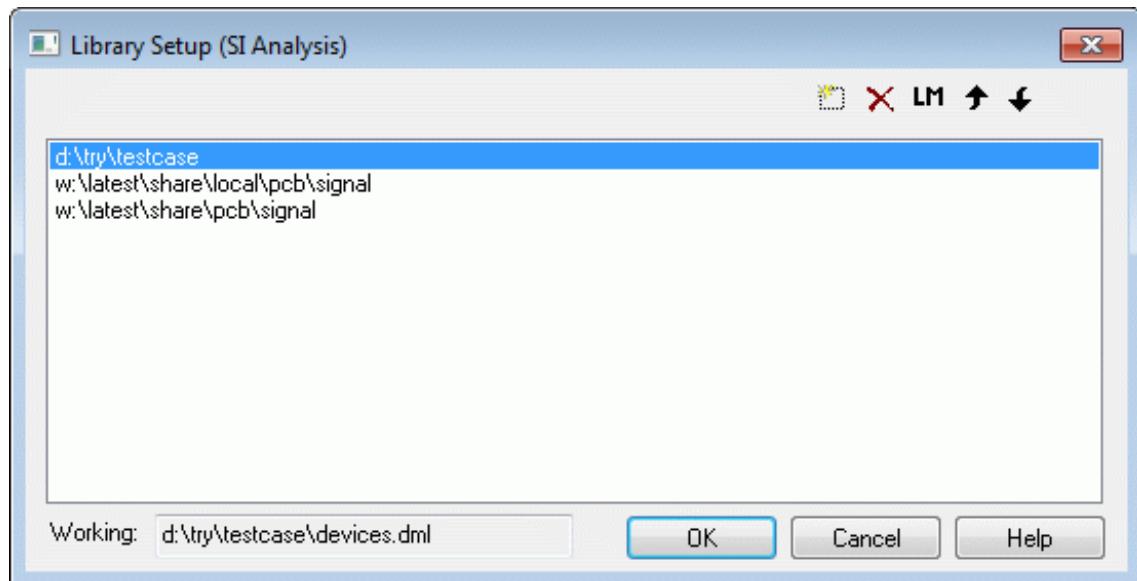
Setting up a Library

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.

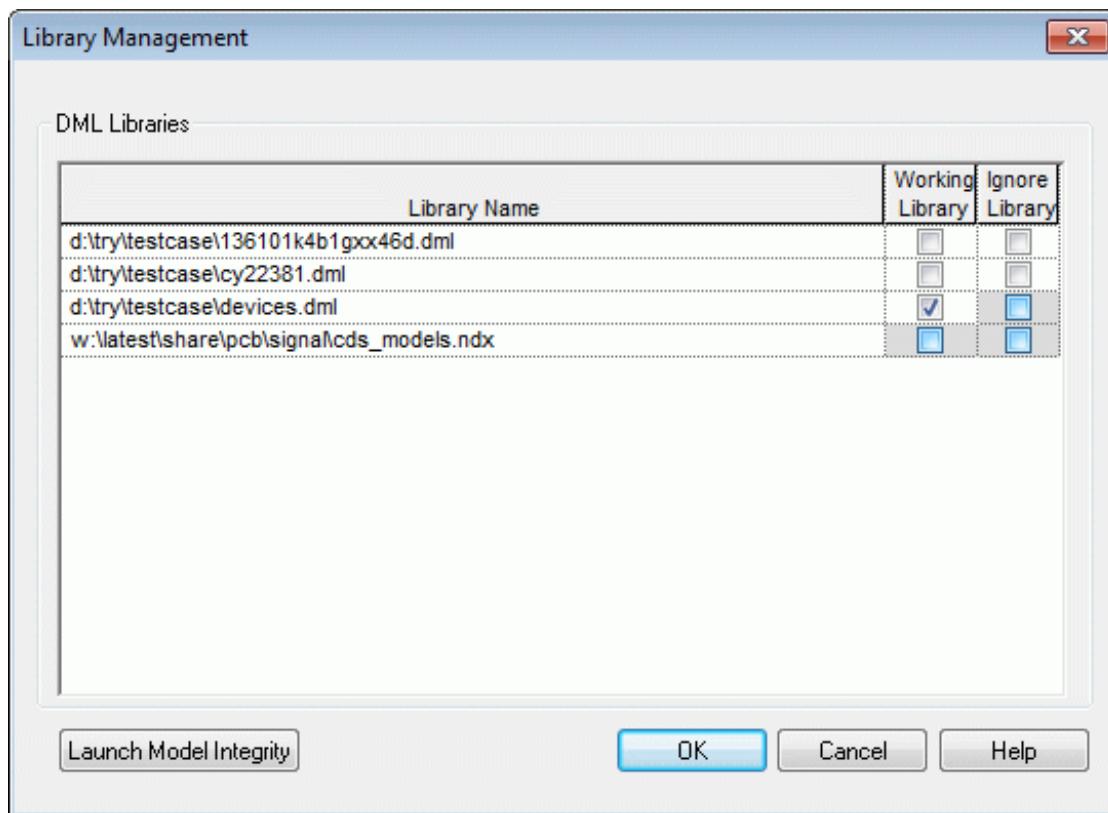
You can launch the Model Integrity tool that helps you ensure the integrity of the model data required for high-speed circuit simulations and lets you create, manipulate, and validate models quickly in an easy-to-use editing environment. For more information on Model Integrity, refer *Model Integrity User Guide*.

To set up library, do the following steps:

1. Choose *SI Analysis – SI Library Setup* to launch the Library Setup (SI Analysis) dialog box.



2. 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 (LM) to launch the Library Management dialog box.

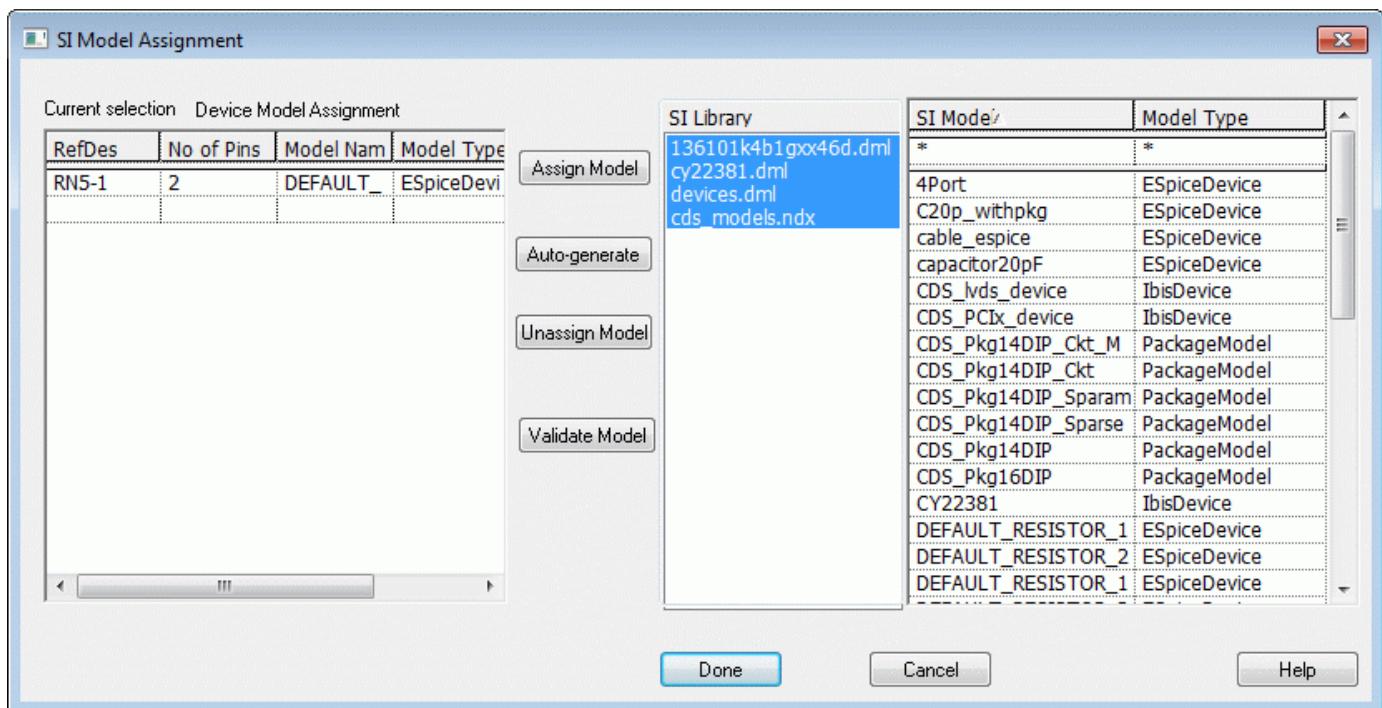
3. In the *Library Management* dialog box, select the *Working Library* field for any one of the libraries to specify it as your working library.
4. Similarly, select the *Ignore Library* field for one or more listed libraries to ignore them.
You can also launch Model Integrity by clicking *Launch Model Integrity*.

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 shortcut menu.

This launches the *SI Model Assignment* dialog box.



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 remove model assignments.

You can choose *View – Selection Filter* and then choose *Assign SI Model* from the shortcut 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.

 Auto-model assignment for a part will fail if any separator except the following is used in the `VALUE` property: comma (,), semi-colon (;), slash (/).

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 SigXplorer 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.

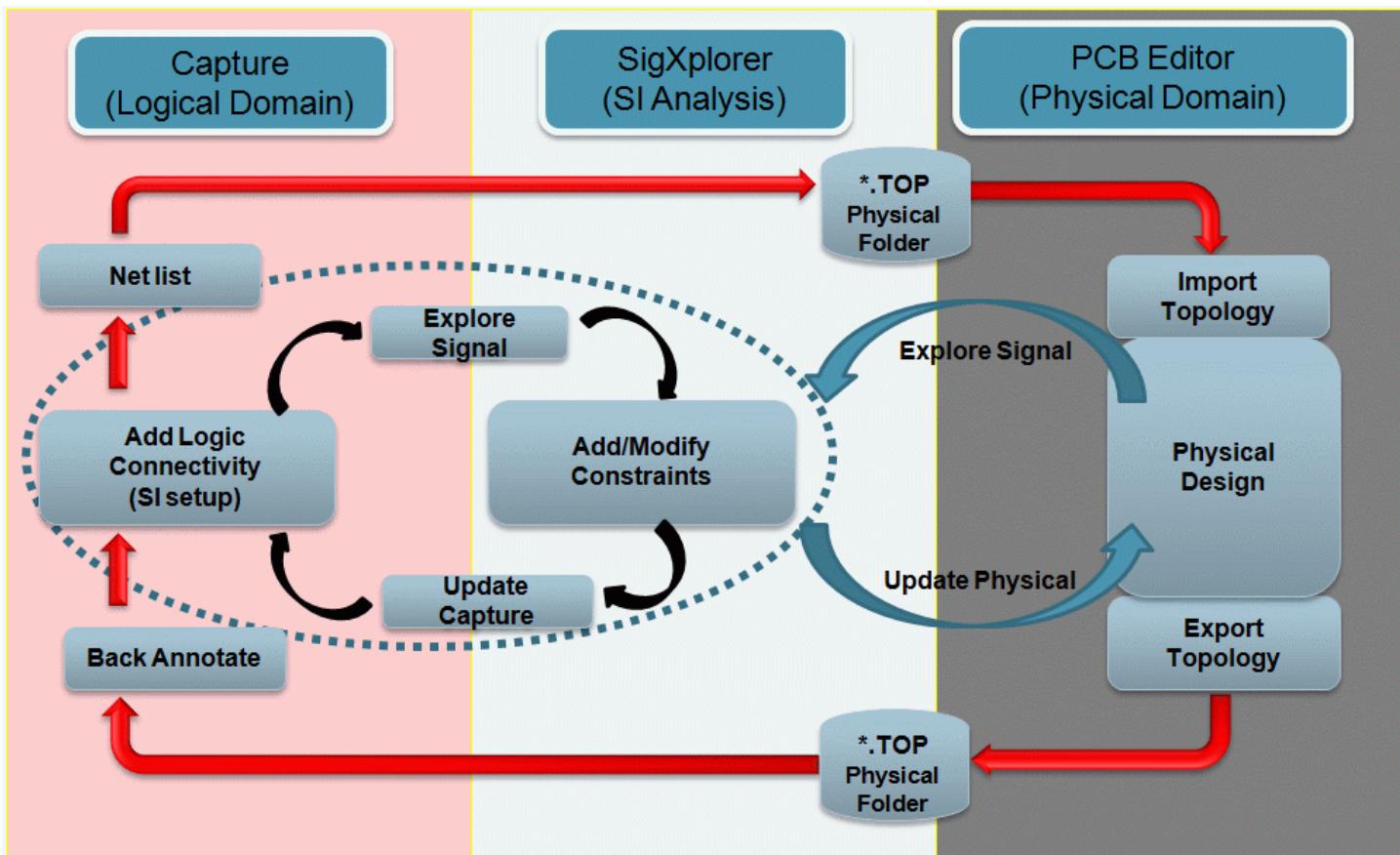
 If you instantiate an external block in your top-level design and the block has Electrical Cset defined, you need to import the associated topology file into the top-level design.

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 back annotate any changes to OrCAD X 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 SigXplorer.

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.

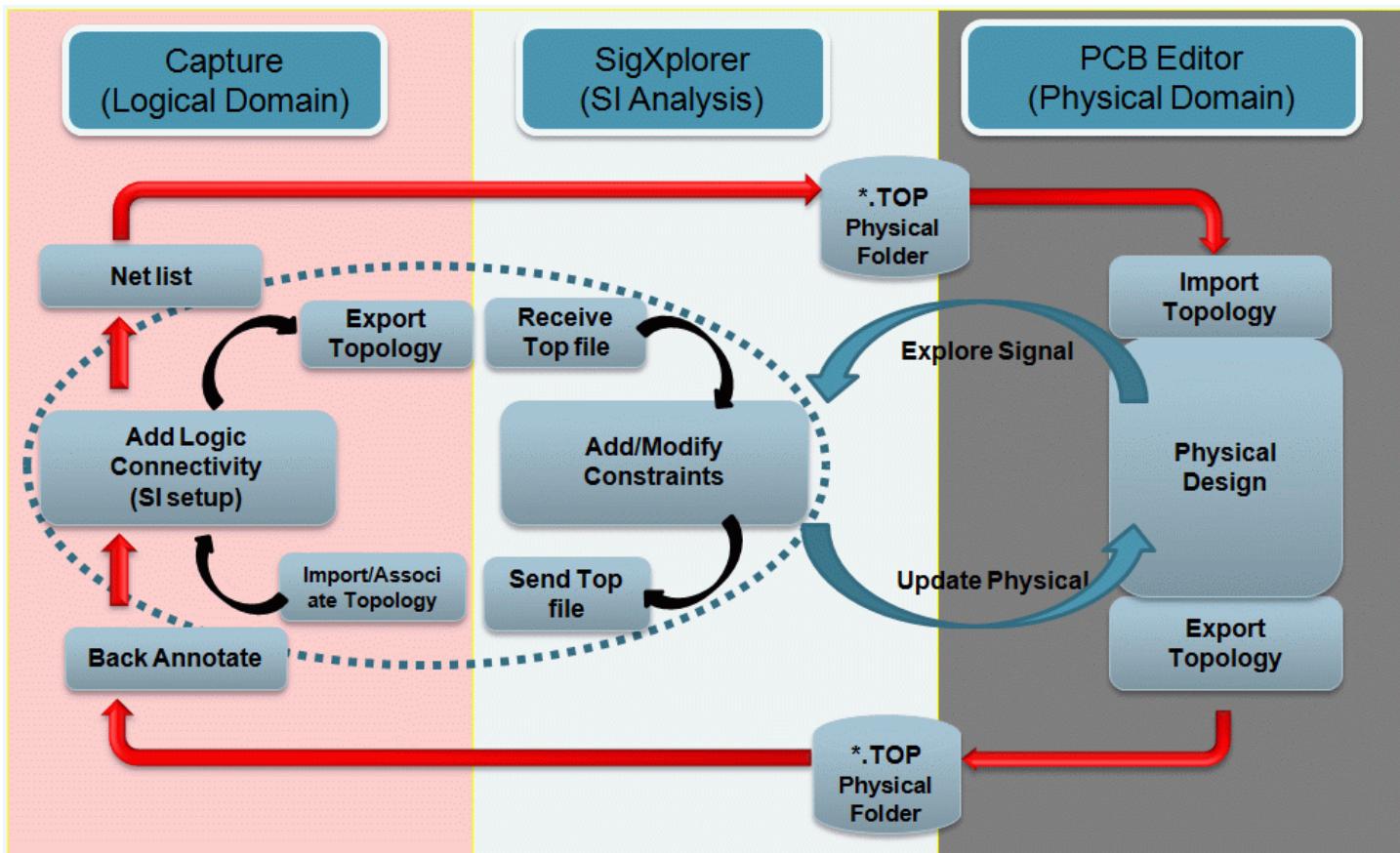
You can choose to change parameter values for discrete components in SigXplorer and save the topology file. In SigXplorer, choose *File – Save* to save the topology and then choose *File – Update Capture*.

⚠ 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 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 Capture using the *Import Electrical Cset* menu item.

⚠ You must load and save the Electrical Csets in SigXplorer before importing and applying them in 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, do the following :

1. Netlist the design for PCB Editor and launch Constraint Manager.
2. Import the topology file.
3. Choose *Audit – Topology* to perform audit of the topology templates.

You can also open and make changes to the topology file in Constraint Manager.

Managing Toolbars

The toolbars provide easy access to common actions. Each toolbar contains buttons that relate to the specific functional group, such as PSpice toolbar. If you need more clear space on the screen to view your work, you can hide a toolbar, and display it again when you wish to use one of the tools. All the tools on the toolbar are available as menu commands.

Displaying a Toolbar

To display a toolbar, do the following:

1. Choose the *View – Toolbar* menu command
2. Select the required toolbar from the submenu.

Alternatively, right-click the menu bar at the top of the Capture screen and select the required toolbar.

Similarly, to hide a toolbar being displayed, select it from the *View – Toolbar* menu.

Moving a Toolbar

To move a toolbar anywhere on the screen, press the left mouse button over the toolbar and move the mouse to the new location of the toolbar.

If you move the toolbar to an edge of the session frame, it snaps into place along the side of the window. Otherwise, it floats on the screen wherever it is released.

When you move the pointer over an icon on the toolbar, a tooltip displays its name.

This section provides a brief description of the various toolbars in the Capture environment and how to manage them:

- [Align Toolbar](#)
- [Capture Toolbar](#)
- [Draw Electrical Toolbar](#)
- [Draw Graphical Toolbar](#)

- Part Manager Toolbar
- PCB Toolbar
- PSpice Toolbar
- SI Analysis Toolbar
- Customizing Toolbars
- Docking Toolbars

Align Toolbar

The Align toolbar offers a quick and easy way to align selected objects vertically (top, middle, and bottom) or horizontally (left, center, and right) with reference to other object. It also provides an option to equally distribute selected objects horizontally or vertically across a schematic page.

You can also align selected objects with reference to a mouse click or equally distribute the selected objects within the area defined by mouse clicks using Mouse Mode.

Icon	Name	Description
	Align Left	<p>Use this command to left align the selected objects with respect to the leftmost selected object.</p> <p>In Mouse Mode:</p> <p>Use this command to left align the selected objects with respect to the mouse click.</p>
	Align Center	<p>Use this command to center align the selected objects with respect to the virtual selection bounding box. The virtual selection bounding box is formed with respect to the leftmost object and the rightmost object of the selected objects.</p> <p>In Mouse Mode:</p> <p>Use this command to center align the selected objects with respect to the mouse click.</p>

	Align Right	<p>Use this command to right align the selected objects with respect to the rightmost selected object.</p> <p>In Mouse Mode:</p> <p>Use this command to right align the selected objects with respect to the mouse click.</p>
	Align Top	<p>Use this command to top align the selected objects with respect to the topmost selected object.</p> <p>In Mouse Mode:</p> <p>Use this command to top align the selected objects with respect to the mouse click.</p>
	Align Middle	<p>Use this command to middle align the selected objects with respect to virtual selection bounding box. The virtual selection bounding box is formed with respect to the topmost object and the bottommost object of the selected objects.</p> <p>In Mouse Mode:</p> <p>Use this command to middle align the selected objects with respect to the mouse click.</p>
	Align Bottom	<p>Use this command to bottom align the selected objects with respect to the bottommost selected object.</p> <p>In Mouse Mode:</p> <p>Use this command to bottom align the selected objects with respect to the mouse click.</p>
	Distribute Horizontal	<p>Use this command to horizontally distribute the selected objects with equal spacing within virtual selection bounding box. The virtual selection bounding box is formed with respect to the leftmost object and the rightmost object of the selected objects.</p> <p>In Mouse Mode:</p> <p>Use this command to horizontally distribute the selected objects with equal spacing between the two selected points on a schematic page.</p>
	Distribute Vertical	<p>Use this command to vertically distribute the selected capture objects with equal spacing within virtual selection bounding box. The virtual selection bounding box is formed with respect to the topmost object and the bottommost object of the selected objects.</p> <p>In Mouse Mode:</p> <p>Use this command to vertically distribute the selected objects with equal spacing between the two selected points on a schematic page.</p>

	Mouse Mode	Use this command to enable or disable Mouse Mode.
---	-------------------	---

Capture Toolbar

The Capture toolbar provides shortcuts for many of the most frequently used generic Capture commands. The following table describes the icons on the toolbar.

Icon	Name	Description
	Create document	Creates a new document based on the active document. Equivalent to the New command on the File menu.
	Open document	Opens an existing document based on the active document. Equivalent to the Open command on the File menu.
	Save document	Saves the active schematic page or part. Equivalent to the Save command on the File menu.
	Print	Prints the active schematic page or part. Equivalent to the Print command on the File menu.
	Cut to clipboard	Removes the selected object and places it on the Clipboard. Equivalent to the Cut command on the Edit menu.
	Copy to clipboard	Copies the selected object to the Clipboard. Equivalent to the Copy command on the Edit menu.
	Paste from clipboard	Pastes the contents of the Clipboard at the cursor. Equivalent to the Paste command on the Edit menu.
	Undo	Undoes the last command performed. Equivalent to the Undo command on the Edit menu.
	Redo	Redoes the last command performed. Equivalent to the Redo command on the Edit menu.

	Snap to grid	<p>When Snap to Grid is turned on , then Capture automatically aligns all objects as you place or move them on the page, by snapping them to the page grid.</p> <p>When Snap to Grid is turned off , then Capture allows you to place or move objects anywhere on the page.</p>
	Area Select - Fully Enclosed Vs Intersecting	<p>If the button is in the Intersecting mode state , the objects are selected when the selection area border intersects them.</p> <p>If the button is in the Fully Enclosed state , the objects are selected only when they are completely enclosed in the selection area.</p> <p>This is equivalent to selecting the Area Select options under the Select tab in the Preferences dialog box.</p> <p>If the button is in the Enclosed mode state, ensure that the object along with its name and number are enclosed in the selection area. Otherwise, the object does not get selected.</p>
	Drag connected object	<p>If the button is in the  state then Capture allows to drag and place objects on the schematic, even if the connectivity changes.</p> <p>If the button is in the  state then the selected object attaches to the cursor and does not get placed on the schematic, if it changes the connectivity.</p>

	Most recently used	Displays the most recently placed part name in the drop-down list. Capture automatically adds part names as you select them from the Place Part dialog box. Select from the list to place parts again later.
	Zoom in	Zooms in to present a closer, enlarged view. Equivalent to the In command on the Zoom menu (on the View menu).
	Zoom out	Zooms out to present more of your document. Equivalent to the Out command on the Zoom menu (on the View menu).
	Zoom to region	Specifies an area of the schematic page or part to enlarge to fill the entire window. Equivalent to the Area command on the Zoom menu (on the View menu).
	Zoom to all	Views the entire document. Equivalent to the All command on the Zoom menu (on the View menu).
	Fisheye	Toggles the Fisheye mode on and off.
	Project manager	Displays a project manager window for the active document, providing an overview of project contents.
	Help	Displays the online help. Equivalent to the Help Topics command on the Help menu.
	Annotate	Assigns part references to parts on the selected schematic pages. Equivalent to the Annotate command on the Tools menu.
	Backannotate	Back annotates the selected schematic pages. Equivalent to the Back Annotate command on the Tools menu.

	Create netlist	Creates a netlist from the selected design. Equivalent to the Create Netlist command on the Tools menu.
	Cross reference parts	Creates a cross reference report of the selected schematic pages. Equivalent to the Cross Reference command on the Tools menu.
	Bill of materials	Creates a bill of materials report from the selected schematic pages. Equivalent to the Bill of Materials command on the Tools menu.

Draw Electrical Toolbar

The Draw Electrical toolbar provides shortcuts for commands to place components, pins, wires, and bus. The following table describes the icons on this toolbar.

Icon	Name	Description
	Select	Selects objects. This is the normal mode.
	Place part	Selects parts from a library for placement. Equivalent to the Part command on the Place menu.
	Place wire	Draws wires. Shift allows any angle drawing. Equivalent to the Wire command on the Place menu.
	Place NetGroup	Places a net group.
	Auto Connect two points	Switches the Schematic page editor into the Auto connect mode and allows you to connect two points on the page.
	Auto Connect multi points	Switches the Schematic page editor into the Auto connect mode and allows you to connect multiple points on the page.
	Auto Connect to Bus	Switches the Schematic page editor into the Auto connect mode and allows you to connect points to a bus.

	Place net alias	Places aliases on wires and buses. Equivalent to the Net Alias command on the Place menu.
	Place bus	Draws buses. Shift allows any angle drawing. Equivalent to the Bus command on the Place menu.
	Place junction	Places or deletes junctions. Equivalent to the Junction command on the Place menu.
	Place bus entry	Draws bus entries. Equivalent to the Bus Entry command on the Place menu.
	Place power	Places power symbols. Equivalent to the Power command on the Place menu.
	Place ground	Places ground symbols. Equivalent to the Ground command on the Place menu.
	Place hierarchical block	Places hierarchical blocks. Equivalent to the Hierarchical Block command on the Place menu.
	Place port	Places hierarchical ports on schematic pages. Equivalent to the Hierarchical Port command on the Place menu.
	Place H pin	Places hierarchical pins in the selected hierarchical block. Equivalent to the Hierarchical Pin command on the Place menu.
	Place pin	Places a pin on the schematic page.
	Place off-page connector	Places off-page connectors. Equivalent to the Off-Page Connector command on the Place menu.
	Place no connect	Places no-connect symbols on pins. Equivalent to the No Connect command on the Place menu.
	Place IEEE Symbol	Places IEEE symbols. Equivalent to the IEEE Symbol command on the Place menu. This command is available in the Part Editor.
	Place Pin Array	Places pin arrays. Equivalent to the Pin Array command on the Place menu. This command is available in the Part Editor.

Draw Graphical Toolbar

The Draw Graphical toolbar provides shortcuts for commands to drawing objects, such as arcs, polyline, ellipse, and text. The following table describes the icons on this toolbar.

Icon	Name	Description
	Place line	Draws lines. Equivalent to the Line command on the Place menu.
	Place polyline	Draws polylines. shift allows any angle drawing. Equivalent to the Polyline command on the Place menu.
	Place rectangle	Draws rectangles. shift constrains the shape to a square. Equivalent to the Rectangle command on the Place menu.
	Place ellipse	Draws ellipses. shift constrains the shape to a circle. Equivalent to the Ellipse command on the Place menu.
	Place arc	Draws arcs. Equivalent to the Arc command on the Place menu.
	Place elliptical arc	Draws elliptical arcs. Equivalent to the Elliptical Arc command on the Place menu.
	Place Bezier	Draws bezier curves. Equivalent to the Bezier Curve command on the Place menu.
	Place text	Places text. Equivalent to the Text command on the Place menu.

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 X Capture CIS. The following table describes the icons on the toolbar.

Icon	Name	Description
	Link Database Part	Links a part from the database to a schematic part. Equivalent to the Link Database command on the Tools menu.
	Update All Part Status	Checks all parts against the parts database. Equivalent to the Update All Part Status command on the Tools menu.
	Bill of materials	Creates a bill of materials. Equivalent to the Standard command on the CIS Bill of Materials sub-menu of the Reports menu.
	Crystal Reports bill of materials	Generates bill of materials using Crystal Reports interface. Equivalent to the Crystal Reports command on the CIS Bill of Materials sub-menu of the Reports menu.
	Variant Report	Generates a variant report. Equivalent to the Variant command on the Reports menu.
	Expand/Collapse Tree Item	Expands or collapses the tree view. Equivalent to the Expand/collapse tree item command on the View menu.
	Show/Hide Tree View	Shows or hides the tree view. Equivalent to the Show/Hide tree view command on the View menu.
	Resolve Ambiguity	Resolves ambiguity for selected item. Equivalent to the Resolve ambiguity command on the Tools menu.

PCB Toolbar

The PCB toolbar for the PCB flow available in Capture provides shortcuts for many of the most frequently used commands. The following table describes the icons on the toolbar.

Icons	Name	Description
-------	------	-------------

	Open Active Layout	Opens the active board file in the layout tool selected in the Design Sync Setup dialog box.
	Open Update Layout	Opens the Update Layout dialog box, which allows you to view and then commit to design connectivity changes in real-time from Schematic to Layout.
	Open Update Schematic	Opens the Update Schematic dialog box, which allows you to view and then commit to design connectivity changes in real-time from Layout to Schematic.
	Launch Constraint Manager	Opens the information window that explains the Capture-Constraint Manager flow.
	Design rules check	Opens the Design Rules Check window to run checks to identify design rules violations. Checks for design rules violations in the selected schematic pages. Equivalent to the Design Rules Check command on the Tools menu.

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. The following table describes the icons on the toolbar.

Icon	Name	Description
	New simulation profile	Creates a new simulation profile. Equivalent to the New Simulation Profile command on the PSpice menu.
	Edit simulation profile	Opens simulation profile for editing. Equivalent to the Edit Simulation Profile command on the PSpice menu.
	Run PSpice	Runs PSpice simulation for active profile. Equivalent to the Run PSpice command on the PSpice menu.
	View Simulation Results	Shows simulation results for the active profile. Equivalent to the View Simulation Results command on the PSpice menu.

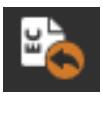
	Launch PSpice Part Search	Opens the PSpice Part Search pane.
	Modeling Applications	Opens the Modeling Application pane.
	Voltage/Level Marker	Places voltage/digital level markers. Equivalent to the Voltage Level command on the Markers submenu of the PSpice menu.
	Voltage Differential Marker	Places voltage differential marker. Equivalent to the Voltage Differential command on the Markers submenu of the PSpice menu.
	Current Marker	Places marker showing current into a pin. Equivalent to the Current Into Pin command on the Markers submenu of the PSpice menu.
	Power Dissipation Marker	Places marker showing power dissipation of a device. Equivalent to the Power Dissipation command on the Markers submenu of the PSpice menu.
	Displays a list of Plot Window Templates	Displays a list of Plot Window Templates available for marker placement. Equivalent to the Plot Window Template command in the Markers submenu of the PSpice menu.
	Enable Bias Voltage Display	Enables display of bias voltages. Equivalent to the Enable Bias Voltage Display command on the Bias Points submenu of the PSpice menu.
	Toggle Voltages on Selected Net(s)	Toggles display of voltage bias point value for selection. Equivalent to the Toggle Selected Bias Voltage Display command on the Bias Points submenu of the PSpice menu.
	Enable Bias Current Display	Enables display of bias currents. Equivalent to the Enable bias current display command on the Bias Points submenu of the PSpice menu.
	Toggle Currents on Selected Part(s)/Pin(s)	Toggles display of current bias point value for selection. Equivalent to the Toggle Selected Bias Current Display command on the Bias Points submenu of the PSpice menu.

	Enable Bias Power Display	Enables display of bias point quiescent power. Equivalent to the Enable bias power display command on the Bias Points submenu of the PSpice menu.
	Toggle Power on Selected Part(s)	Toggles display of bias point quiescent power for selection. Equivalent to the Toggle bias power command on the Bias Points submenu of the PSpice menu.
	Assigns Tolerance for Advanced Analysis	Opens the Tolerance Analysis window. Equivalent to the Assign Tolerance command on the Advanced Analysis submenu of the PSpice menu.
	Start Advanced Analysis - Sensitivity	Enables Sensitivity Analysis in PSpice Advanced Analysis. Equivalent to the Sensitivity Analysis command in the Advanced Analysis submenu of the PSpice menu.
	Start Advanced Analysis - Optimizer	Enables Optimizer Analysis in PSpice Advanced Analysis. Equivalent to the Optimizer Analysis command in the Advanced Analysis submenu of the PSpice menu.
	Start Advanced Analysis - Monte Carlo	Enables Monte Carlo Analysis in PSpice Advanced Analysis. Equivalent to the Monte Carlo Analysis command in the Advanced Analysis submenu of the PSpice menu.
	Start Advanced Analysis - Smoke	Enables Smoke Analysis in PSpice Advanced Analysis. Equivalent to the Smoke Analysis command in the Advanced Analysis submenu of the PSpice menu.
	Start Advanced Analysis - Parametric Plot	Enables Parametric Plot Analysis in PSpice Advanced Analysis. Equivalent to the Parametric Plot Analysis command in the Advanced Analysis submenu of the PSpice menu.

SI Analysis Toolbar

The SI Analysis toolbar for SI Analysis functions available in Capture provides shortcuts for many of the most frequently used SI Analysis commands. The following table describes the icons on the toolbar.

Icon	Name	Description

	SI Library Setup	Opens the Library Setup (SI Analysis) window to set the DML libraries.
	Auto Assign Discrete SI Models	Assigns discrete SI models to the design parts.
	Identify DC Nets in the Design	Identify all DC nets in the design and assign voltage property.
	Assign SI Model	Opens the SI Model Assignment window where you can assign models from the listed libraries. You can also autogenerate models.
	Explore Signal	Opens SigXplorer to explore the signal.
	Export Topology	Export Topology dumps a .top file for a selected net which needs to be used by SigXP for launching.
	Export Electrical Csets	Opens the Export Electrical Csets from design window. You can export a topology file that can then be updated using SigXplorer offline in a distributed design environment.
	Import Electrical Csets	Opens the Select Directory window where you can specify the directory from where you want to import an Electrical Cset. This allows you to import topology files that have been updated offline in a distributed design environment.
	Remove Electrical Cset Assignments	Opens the Remove ECSets from design window where you can selectively remove existing Electrical Csets from the design.
	Associate Electrical Cset	Associates an Electrical Csets using a topology file.
	Validate Electrical Cset Assignments	Opens the Validate ECSets in design window where you can selectively validate Electrical Csets in the design.

	Validate SI Model Assignments	Validates the SI model assignments.
	SI Model Integrity	Opens Model Integrity with the DML file for the devices.
	Export SI Models Used	Exports the SI models used in the design into a DML library file.
	Remove SI Model Assignments	Removes SI model assignments.

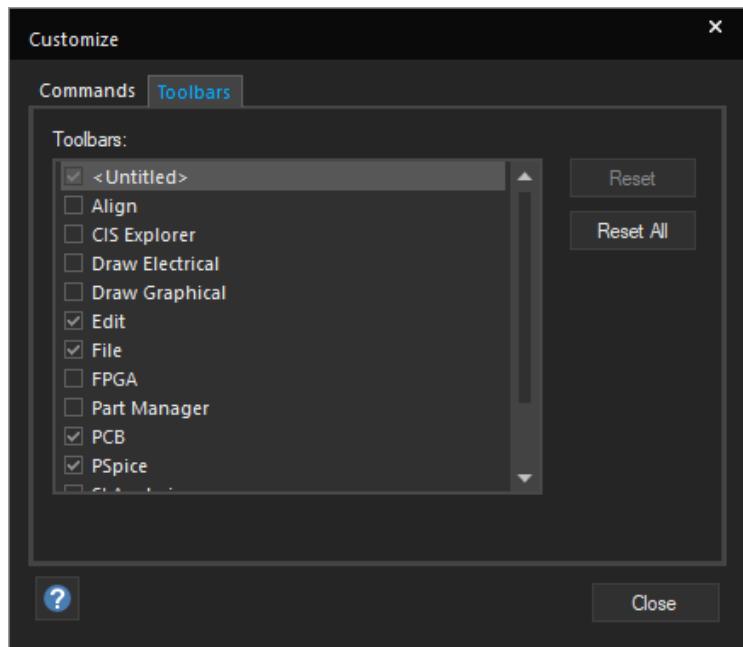
Customizing Toolbars

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

Changing the Display

To change the display of toolbars, do the following:

1. Select *Tools – Customize*.
2. Click the *Toolbars* tab.



3. Select the toolbars you want to display from this list.

The selected toolbar appears.

4. Click *Close*.

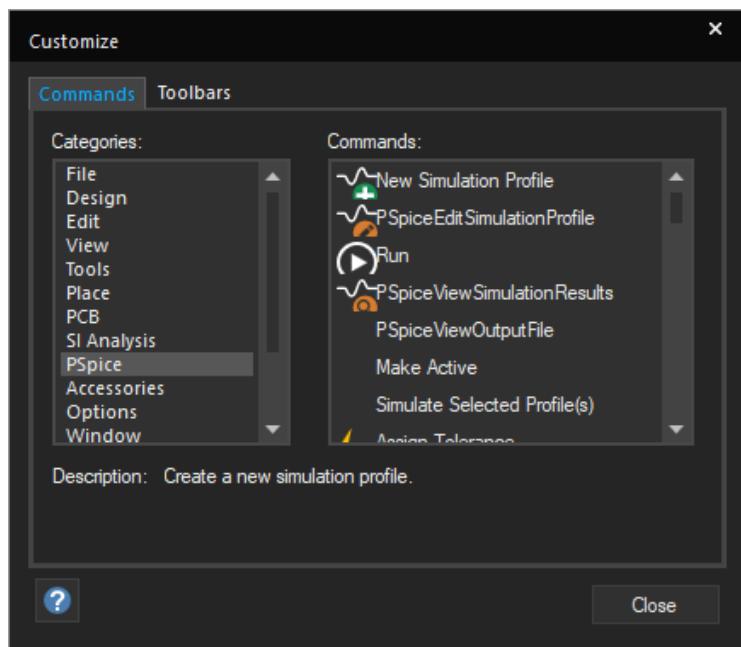
Alternatively, you can open the *Customize* dialog box using one of the following methods:

- Choose *View – Toolbar – Customize*.
- Right-click any menu bar or toolbar and choose *Customize*.

Adding Menus and Toolbar Icons

To add a menu command or a toolbar button as a new menu or in the toolbar, do the following:

1. Select *Tools – Customize*.
2. In the *Commands* tab, from the *Categories* list, select a category to display the related toolbar or menu bar options.



3. To add the selected option to a toolbar or to a menu, drag it from the *Customize* dialog box to any toolbar or menu bar displayed in the program window.
4. Click *Close* to close the dialog box.
 - To remove buttons from toolbars, drag the button away from the toolbar.

Resetting Toolbars to Default

To reset toolbars to their default settings, do the following:

1. Select *Tools – Customize*.
2. Click the *Toolbars* tab.
3. Click *Reset*.
4. Click *Close*.

 Changes are applied only to the currently selected toolbar.

Docking Toolbars

The toolbars in Capture can be docked, or made floating. This gives you 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, keeping the left mouse button pressed, drag the toolbar to the desired location on the application. Ensure that you do not click any of the toolbar buttons.
- To dock a floating toolbar, double-click its title bar.
- To move a floating toolbar, click the title bar and keeping the left mouse button down, drag the toolbar.
- To hide a toolbar, click the close (X) button on the right side of the title bar.
- To hide a toolbar, you can also right-click any toolbar or menu bar and deselect the toolbar name from the shortcut menu.

Shortcut Keys

In addition to providing menu access keys for menu commands, Capture provides shortcut keys for miscellaneous actions, such as scrolling across an editor's window. An example of an access key is ALT, E, T for the Cut command on the Edit menu. Shortcut keys include CTRL keys, Single letter keys (like P to place a part), and function keys (like F4 to repeat a command).

The description of each menu command includes keyboard, mouse, and other shortcuts.

- Many shortcuts are available while you use another command. For example, you can use I and O to zoom in and out while you move and place objects.
- You can run PSpice simulations from your Capture environment by pressing the F11 function key. Also, you can view the simulation results for the currently active profile by pressing the F12 function key.

You can select a product suite to open Capture without exiting the tool by choosing the *File – Change Product* menu command. If only start page is opened in the tool, press Alt, F, C, C, Enter for *File - Change Product* menu command, else press ALT, F, C when no window is opened in the tool. Also, you can select the Use as default check box to set the default product suite from which Capture should check out a license each time you start the tool.

All Capture windows

Key	Mouse click equivalent
ALT+F4	Exit
ALT, F, X	Exit
ALT, SPACEBAR, C	Exit
F1	Help

Text editor

Key	Mouse click equivalent
CTRL+F4	If you attach a VHDL file to a hierarchical block and descend the hierarchy, you will end up within the VHDL file. Use this shortcut to close the file and return to the top level of the schematic

Schematic page editor

Key	Mouse click equivalent
CTRL+A	Select All
CTRL+E	Edit properties
SHIFT+A	Ascend hierarchy
SHIFT+D	Descend hierarchy
B	Place bus
E	Place bus entry
F	Place power
G	Place ground
J	Place junction
N	Place net alias
P	Place part
T	Place text
W	Place wire
Y	Place polyline
X	Place No connect symbol
Z	Place Database Part

CTRL+SHIFT+A	Add part(s) to group
CTRL+SHIFT+R	Remove part(s) from group
CTRL+SHIFT+W	Select entire net
CTRL+I	Opens the Selection Filter dialog box where you can specify the objects that should be selected when the mouse pointer is dragged diagonally across the schematic page.
CTRL+SHIFT+Left Click	Lock component
CTRL+U	Ungroup objects
CTRL+G	Center the view on a specific location, grid references, or bookmark.
SHIFT+B	Create test bench
SHIFT+D	Compare an active test bench with the master design
CTRL+F4	Close Place Part dialog <ul style="list-style-type: none"> • To use this shortcut, the Place Part dialog must be currently selected.
ALT+D	Reloads Library Parts in Place Part dialog <ul style="list-style-type: none"> • To use this shortcut, the library must be selected in Place Part dialog.
SHIFT+I	Is the shortcut for the <i>Edit Source Component</i> option. Select the source component and use this shortcut to open the Independent Sources dialog box in the edit mode.
SHIFT+N	Is the shortcut for the <i>Edit PSpice Component</i> option. Select the modeling application PSpice component and then use this shortcut to open the PSpice Modeling Application in the edit mode.
F5	Redraw
B	Begin a wire, bus, or polyline (corresponding tool active)
C	Center the view at the pointer's current position
E	End a wire, bus, or polyline (corresponding tool active)

 Select a component on the schematic page and press SHIFT+S to place components directly from your Capture schematic design to Allegro PCB Editor.

Part editor

Key	Mouse click equivalent
CTRL+B	Previous part
CTRL+N	Next part

Property editor

Key	Mouse click equivalent
CTRL+B	Gives focus to the Filter by drop-down list
CTRL+D	Open the Display Properties dialog box to edit properties for a selected cell
CTRL+E	Edit properties
CTRL+F	Find a character or search string
CTRL+U	Open Propagation Delay dialog box
CTRL+L	Delete a property (cell value)
CTRL+N	Add a new column or row
CTRL+P	Apply changes
DELETE	Delete a character
CTRL+Z	Undo
CTRL+C	Copy
CTRL+V	Paste
CTRL+X	Cut

Page-Up/CTRL+<Up Arrow>	Go to the first cell in a column
Page-Down/CTRL+<Down Arrow>	Go to the last cell in a column
CTRL+<Left Arrow>	Go to the first cell in a row
CTRL+<Right Arrow>	Go to the last cell in a row
Esc	Undo edit in the selected cell
SHIFT+<Arrow key>	Select
CTRL+Home	Go to the beginning (top left corner) of the spreadsheet
CTRL+End	Go to the end (bottom right corner) of the spreadsheet
CTRL+F2	Select the contents of a cell
CTRL+F4	Close the spreadsheet
CTRL+F6	Switch to the other open windows (Works like CTRL+Tab)
CTRL+U	Opens the Propagation Delay dialog box or the Relative Propagation Delay dialog box depending on the property being edited in the Property Editor window. For example, if you select the grid corresponding to the Propagation Delay property and press CTRL+U, the Propagation Delay dialog box appears.

Schematic page and part editors

Key	Mouse click equivalent
CTRL+C	Copy
CTRL+F	Find

CTRL+G	Go to
CTRL+P	Print
CTRL+S	Save
CTRL+T	Cursor snap to grid (identical to the Preferences dialog box Grid display tab option).
CTRL+V	Paste
CTRL+X	Cut
CTRL+Y	Redo
CTRL+Z	Undo
F4	Repeat
DEL	Delete (Design and Edit menus)
DELETE	Delete (Design and Edit menus)
BACKSPACE	Delete (Design and Edit menus)
ENTER	Double-click
ESCAPE	Deselect all and switch to selection tool (arrow pointer)
SPACE	Click
UP ARROW	Move 1 grid up (grid on) or 0.1 grid up (grid off)
DOWN ARROW	Move 1 grid down (grid on) or 0.1 grid down (grid off)
LEFT ARROW	Move 1 grid left (grid on) or 0.1 grid left (grid off)
RIGHT ARROW	Move 1 grid right (grid on) or 0.1 grid right (grid off)
CTRL+UP ARROW	Snap pointer to the nearest grid and then move 5 grids up
CTRL+DOWN ARROW	Snap pointer to the nearest grid and then move 5 grids down
CTRL+LEFT ARROW	Snap pointer to the nearest grid and then move 5 grids left

CTRL+RIGHT ARROW	Snap pointer to nearest grid and then move 5 grids right
PAGE UP	Pan up
PAGE DOWN	Pan down
CTRL+PAGE UP	Pan left
CTRL+PAGE DOWN	Pan right
H	Mirror horizontally
I	Zoom in
O	Zoom out
R	Rotate
V	Mirror vertically
SHIFT+R	Place rectangle

Session log

Key	Mouse click equivalent
CTRL+DEL	Clear session log

 When an error or warning message displays in the session log, you can find the help topic for that particular error or warning by putting your cursor in the line containing the error/warning and pressing the F1 key.

Text boxes

Key	Mouse click equivalent
BACKSPACE	Delete selected text
DEL	Delete selected text
DELETE	Delete selected text
CTRL+C	Copy selected text to Clipboard
CTRL+V	Paste Clipboard contents
CTRL+X	Cut selected text to Clipboard
CTRL+Z	Undo last edit
DOUBLE CLICK	Select word and any following space
SHIFT+CLICK	Extend selection from insertion point to cursor
CTRL+RIGHT ARROW	Jump right one word
CTRL+LEFT ARROW	Jump left one word
HOME	Jump to beginning of line
END	Jump to end of line
CTRL+HOME	Jump to beginning of text box
CTRL+END	Jump to end of the text box
SHIFT+HOME	Extend selection from insertion point to beginning of multiple-line text box
SHIFT+END	Extend selection from insertion point to end of multiple-line text box

Browse spreadsheet editor

Key	Mouse click equivalent
CTRL+C	Copy value from a cell in the Browse spreadsheet editor
CTRL+V	Paste Clipboard contents onto another cell in Browse spreadsheet editor
CTRL+INSERT	Copy value from a cell in the Browse spreadsheet editor and paste it onto a cell in Microsoft Excel worksheet
SHIFT+INSERT	Paste value copied from Microsoft Excel onto a cell in the Browse spreadsheet editor

Fisheye View

Key	Mouse click equivalent
SHIFT+F11	Set Fisheye focus on selected component
SHIFT+CTRL+F11	Reset Fisheye focus
Q	Start Fisheye dynamic focus mode

PCB Editor Netlist Files

In the schematic to layout flow, user-defined flow through the `pstxnet`, `pstxpert` and `pstchip` files.

For a Constraint Manager-enabled design, in the schematic to layout flow, the user-defined properties are pushed to Allegro-CM. In the layout to schematic flow, the same user-defined properties become part of the Capture-CM flow and start flowing through both the constraint netlist file, `cmdbView` and the `pstxnet`, `pstxpert` and `pstchip` files.

In turn, user-defined properties are available in both Capture Property [Editor](#) and Capture-CM.

In this section:

- [PSTCHIP.DAT](#)
- [PSTXNET.DAT](#)
- [PSTXPRT.DAT](#)

PSTCHIP.DAT

The `PSTCHIP.DAT` file contains a description of each physical part used in a Capture design. The Capture netlister extracts this physical description from properties on all occurrences rather than just the instances.

In this section:

- [PSTCHIP File format](#)
- [PSTCHIP File Elements](#)
- [PSTCHIP Sample file](#)

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	<p>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:</p> <p>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</p>

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 be 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 File format](#)
- [PSTXNET Sample file](#)

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 be 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@TTL.74LS04.NORMAL(CHIPS)':
'I':;
NODE_NAME U2 4
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679613@FULLADD.74LS08_0.NORMAL(CHIPS)':
'-'-

```

```
'I0':;
NODE_NAME U2 9
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679614@FULLADD.74LS08_0.NORMAL(CHIPS)':
'I0':;
NODE_NAME U1 8
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679615@FULLADD.74LS32_0.NORMAL(CHIPS)':
'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@FULLADD.74LS32_0.NORMAL(CHIPS)':
'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@TTL.74LS04.NORMAL(CHIPS)':
'I':;
NODE_NAME U2 12
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679611@FULLADD.74LS08_0.NORMAL(CHIPS)':
'I0':;
NODE_NAME U4 5
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679614@FULLADD.74LS08_0.NORMAL(CHIPS)':
'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@TTL.74LS04.NORMAL(CHIPS)':
'I':;
NODE_NAME U4 1
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679613@FULLADD.74LS08_0.NORMAL(CHIPS)':
'I0':;
NODE_NAME U4 4
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679614@FULLADD.74LS08_0.NORMAL(CHIPS)':
'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@TTL.74LS04.NORMAL(CHIPS)':
'I':;
NODE_NAME U2 1
```

```
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679611@FULLADD.74LS08_0.NORMAL(CHIPS)':
'I0':;
NODE_NAME U2 10
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679614@FULLADD.74LS08_0.NORMAL(CHIPS)':
'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@FULLADD.74LS08_0.NORMAL(CHIPS)':
'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@FULLADD.74LS08_0.NORMAL(CHIPS)':
'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)':
'X_BAR':
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@TTL.74LS04.NORMAL(CHIPS)':
'O':;
NODE_NAME U2 5
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679613@FULLADD.74LS08_0.NORMAL(CHIPS)':
'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@FULLADD.74LS08_0.NORMAL(CHIPS)':
'
```

```
'I1':;
NODE_NAME U3 4
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679612@TTL.74LS04.NORMAL(CHIPS)':
'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@FULLADD.74LS08_0.NORMAL(CHIPS)':
'O':;
NODE_NAME U1 4
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679615@FULLADD.74LS32_0.NORMAL(CHIPS)':
'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@FULLADD.74LS32_0.NORMAL(CHIPS)':
'I1':;
NODE_NAME U2 6
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679613@FULLADD.74LS08_0.NORMAL(CHIPS)':
'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@TTL.74LS04.NORMAL(CHIPS)':
'O':;
NODE_NAME U4 2
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679613@FULLADD.74LS08_0.NORMAL(CHIPS)':
'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@FULLADD.74LS08_0.NORMAL(CHIPS)':
'I1':;
NODE_NAME U3 8
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679612@TTL.74LS04.NORMAL(CHIPS)':
'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@FULLADD.74LS08_0.NORMAL(CHIPS)';  
'O':;  
NODE_NAME U1 9  
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679615@FULLADD.74LS32_0.NORMAL(CHIPS)';  
'IO':;  
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@FULLADD.74LS32_0.NORMAL(CHIPS)';  
'I1':;  
NODE_NAME U4 3  
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679613@FULLADD.74LS08_0.NORMAL(CHIPS)';  
'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 File format](#)
- [PSTXPRT Sample file](#)

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 be 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='occurrence_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
```

```
SECTION_NUMBER 1
 '@FULLADD.FULLADD(SCH_1):I505679590@FULLADD.74LS32.NORMAL(CHIPS)':
 C_PATH='@fulladd.fulladd(sch_1):i505679590@fulladd.\74ls32.normal\chips',
 P_PATH='@fulladd.fulladd(sch_1):page1_i505679590@fulladd.\74ls32.normal\chips',
 PRIM_FILE='.\pstchip.dat',
 SECTION='A';

SECTION_NUMBER 2
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679615@FULLADD.74LS32.NORMAL(CHIPS):
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505679615@fulladd.\74ls32.norm
 al\chips',
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1):page2_i505679615@fulladd.
 \74ls32.normal\chips',
 PRIM_FILE='.\pstchip.dat',
 SECTION='B';

SECTION_NUMBER 3
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679615@FULLADD.74LS32.NORMAL(CHIPS):
 C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505679615@fulladd.\74ls32.norm
 al\chips',
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1):page2_i505679615@fulladd.
 \74ls32.normal\chips',
 PRIM_FILE='.\pstchip.dat',
 SECTION='C';

PART_NAME
 U2 'ANDGATE':;

SECTION_NUMBER 1
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679611@FULLADD.74LS08.NORMAL(CHIPS):
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505679611@fulladd.\74ls08.norm
 al\chips',
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1):page2_i505679611@fulladd.
 \74ls08.normal\chips',
 PRIM_FILE='.\pstchip.dat',
 SECTION='A';

SECTION_NUMBER 2
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679613@FULLADD.74LS08.NORMAL(CHIPS):
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505679613@fulladd.\74ls08.norm
 al\chips',
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1):page2_i505679613@fulladd.
 \74ls08.normal\chips',
 PRIM_FILE='.\pstchip.dat',
 SECTION='B';

SECTION_NUMBER 3
 '@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679614@FULLADD.74LS08.NORMAL(CHIPS):
 C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505679614@fulladd.\74ls08.norm
 al\chips',
```

```
P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1):page2_i505679614@fulladd.\74ls08.normal\chips',
PRIM_FILE='.\pstchip.dat',
SECTION='C';

SECTION_NUMBER 4
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679611@FULLADD.74LS08.NORMAL(CHIPS
):
C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505679611@fulladd.\74ls08.norm
al\chips',
P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1):page2_i505679611@fulladd.
\74ls08.normal\chips',
PRIM_FILE='.\pstchip.dat',
SECTION='D';

PART_NAME
U3 'NOTGATE':;

SECTION_NUMBER 1
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679610@FULLADD.74LS04.NORMAL(CHIPS
):
C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505679610@fulladd.\74ls04.norm
al\chips',
P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1):page2_i505679610@fulladd.
\74ls04.normal\chips',
PRIM_FILE='.\pstchip.dat',
SECTION='A';

SECTION_NUMBER 2
'@FULLADD.FULLADD(SCH_1):HALFADD_A@FULLADD.HALFADD(SCH_1):I505679612@FULLADD.74LS04.NORMAL(CHIPS
):
C_PATH='@fulladd.fulladd(sch_1):halfadd_a@fulladd.halfadd(sch_1):i505679612@fulladd.\74ls04.norm
al\chips',
P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_a@fulladd.halfadd(sch_1):page2_i505679612@fulladd.
\74ls04.normal\chips',
PRIM_FILE='.\pstchip.dat',
SECTION='B';

SECTION_NUMBER 3
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679610@FULLADD.74LS04.NORMAL(CHIPS
):
C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505679610@fulladd.\74ls04.norm
al\chips',
P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1):page2_i505679610@fulladd.
\74ls04.normal\chips',
PRIM_FILE='.\pstchip.dat',
SECTION='C';

SECTION_NUMBER 4
'@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679612@FULLADD.74LS04.NORMAL(CHIPS
):
C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505679612@fulladd.\74ls04.norm
al\chips',
P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1):page2_i505679612@fulladd.
\74ls04.normal\chips',
```

OrCAD X Capture User Guide
PCB Editor Netlist Files--PSTXPRT.DAT

```
\V74LS08.NORMAL\(\chips) ,
PRIM_FILE='.\pstchip.dat',
SECTION='D';

PART_NAME
U4 'ANDGATE':;

SECTION_NUMBER 1
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679613@FULLADD.74LS08.NORMAL(CHIPS
)':
 C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505679613@fulladd.\74ls08.norm
al\chips',
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1):page2_i505679613@fulladd.
\74ls08.normal\chips',
 PRIM_FILE='.\pstchip.dat',
 SECTION='A';

SECTION_NUMBER 2
 '@FULLADD.FULLADD(SCH_1):HALFADD_B@FULLADD.HALFADD(SCH_1):I505679614@FULLADD.74LS08.NORMAL(CHIPS
)':
 C_PATH='@fulladd.fulladd(sch_1):halfadd_b@fulladd.halfadd(sch_1):i505679614@fulladd.\74ls08.norm
al\chips',
 P_PATH='@fulladd.fulladd(sch_1):page1_halfadd_b@fulladd.halfadd(sch_1):page2_i505679614@fulladd.
\74ls08.normal\chips',
 PRIM_FILE='.\pstchip.dat',
 SECTION='B';

END.
```

SDT Configuration Files

Capture requires either an `SDT.CFG` file or an `SDT.BCF` file when translating SDT designs into Capture. If you do not have one of these files, you can create an `SDT.CFG` in a text editor using the sample file provided.

You may need to edit the following items in your `SDT.CFG` file to ensure that schematic folders are imported correctly:

- 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 uses the `SDT.CFG` file to locate the libraries and parts needed to translate the schematic folders.

In this section:

- [Path and libraries](#)
- [Page dimensions and units](#)
- [Part fields](#)

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. You can have only one '`PLIB=`' line in your `SDT.CFG` file, and 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` file where `pathname` specifies the directory path to the `.LIB` files, and `filename` specifies the name of one `.LIB` file:

```
PLIB = 'pathname\*.LIB'  
LIB = '.\filename.LIB'
```

Page dimensions and units

If the design uses metric units, replace the corresponding lines with the following lines:

```
HOR = 264000 374300 548300 795300 1143300  
VRT = 177000 264000 374300 548300 795300  
P2P = 2540 2540 2540 2540 2540  
UNTS = 'METRIC'
```

Part fields

Use the part field lines in the `SDT.CFG` file to map the 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'  
FN5 = '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
```

Netlist Examples

This chapter provides a brief overview of some of the netlist formats available from Capture.

In this section:

- [Accel netlist format](#)
- [Algorex netlist format](#)
- [Altera ADF netlist format](#)
- [AppliconBRAVO netlist format](#)
- [AppliconLEAP netlist format](#)
- [Cadnetix netlist format](#)
- [Calay90 netlist format](#)
- [Calay netlist format](#)
- [Case netlist format](#)
- [CBDS netlist format](#)
- [Computervision netlist format](#)
- [DUMP netlist format](#)
- [EDIF 2 0 0 netlist format](#)
- [EEDesigner netlist format](#)
- [Futurenet netlist format](#)
- [HiLo netlist format](#)
- [Intel ADF netlist format](#)
- [Intergraph netlist format](#)
- [Mentor netlist format](#)
- [Multiwire netlist format](#)

- OHDL netlist format
- PADS 2000 netlist format
- PADS PCB netlist format
- PCAD netlist format
- PCADnlt netlist format
- PCBII and PCBIIL netlist formats
- PDUMP netlist format
- PLD netlist format
- Protel2 netlist format
- RecalRedac netlist format
- RINF netlist format
- Scicards netlist format
- Tango netlist format
- Telesis netlist format
- Vectron netlist format
- Verilog netlist format
- VHDL netlist format
- VST Model netlist format
- WinBoard netlist format
- WireList netlist format

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 CADD3 and CADD4X 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"))
/autostart orcad ttp
```

```
(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,IO_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,IO_B
)
(SIG,,N00019,1-1,5,N00019
PIN,1-1,1,U1,21,O_A
PIN,1-1,2,U2,23,IO_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,IO_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 X 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 (OO,"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) | U1
| G257 (SELECT,Q12,I12,O12,Q13,I13,O13,-,O14,I14,Q14,O15,I15,Q15,GND) | U10
| G257 (SELECT,Q0,I0,OO,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) | U6
| 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, -) | U9
| }
|
| 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 PCBIIIL 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 PCBIIIL netlist formats are identical with the following exception: the PCBIIIL.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 PCBIIIL) 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.

 The Telesis netlist created from Capture cannot be imported into PCB Editor. In the Create Netlist dialog box, use the *PCB* tab to generate a netlist that can be used in PCB Editor.

Example

Telesis netlists 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...
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:
```

```
```verilog
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',
^ ^ OPEN
```

```
U_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\	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

---

# 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 X 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 X 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 Editor 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												
48	XX												
72	XX												
96	X-	--	--	--	--	--	--	--	--	--	--	--	--
120	--	--	--	--	--	--	--	X-	--	--	--	--	--
144	--	--	--	-X	--	--	--	--	X-	--	--	--	--
168	--	--	--	X-	--	--	--	--	-X	--	--	--	--
192	--	--	-X	--	--	--	--	--	--	--	--	--	--
216	XX												
240	--	--	--	--	-X	--	--	--	--	--	--	--	--
264	--	--	--	--	--	--	--	--	--	--	-X	--	--
288	--	--	--	--	--	-X	--	--	--	-X	--	--	--
312	XX												
336	XX												
360	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

### **heterogeneous 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 [heterogeneous 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 X [EDA](#) tools to share information for display and transfer.

**ITC**

Intertool Communication. A capability available with OrCAD X 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](#), [heterogeneous 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.

**polygon**

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 X 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 X 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 X 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 [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.