SIEMENS EDA

Calibre® xACT™ User's Manual

Software Version 2021.2
Document Revision 14

SIEMENS

Support Center: support.sw.siemens.com
Send Feedback on Documentation: support.sw.siemens.com/doc_feedback_form

Revision History

Revision	Changes	Status/ Date
14	Modifications to improve the readability and comprehension of the content. Approved by Lucille Woo. All technical enhancements, changes, and fixes listed in the *Calibre Release Notes* for this products are reflected in this document. Approved by Michael Buehler.	Released April 2021
13	Modifications to improve the readability and comprehension of the content. Approved by Lucille Woo. All technical enhancements, changes, and fixes listed in the *Calibre Release Notes* for this products are reflected in this document. Approved by Michael Buehler.	Released January 2021
12	Modifications to improve the readability and comprehension of the content. Approved by Lucille Woo. All technical enhancements, changes, and fixes listed in the *Calibre Release Notes* for this products are reflected in this document. Approved by Michael Buehler.	Released October 2020
11	Modifications to improve the readability and comprehension of the content. Approved by Lucille Woo. All technical enhancements, changes, and fixes listed in the *Calibre Release Notes* for this products are reflected in this document. Approved by Michael Buehler.	Released July 2020

Author: In-house procedures and working practices require multiple authors for documents. All associated authors for each topic within this document are tracked within the Siemens EDA documentation source. For specific topic authors, contact the Siemens Digital Industries Software documentation department.

Revision History: Released documents maintain a revision history of up to four revisions. For earlier revision history, refer to earlier releases of documentation which are available on https://support.sw.siemens.com/.

Table of Contents

Revision History

Chapter 1
Calibre xACT Product Overview 17

The Calibre xACT Tools 17
Calibre xACT Digital Design Flow 17
Calibre xACT and the Full Custom Design Flow 18
Calibre xACT Digital Flow Overview 19
Calibre xACT Overview 20
Calibre xACT 3D Overview 21
Calibre xACT Inductance Flow Overview 22
Calibre xACTView Overview 23
Syntax Conventions 23

Chapter 2
Getting Started: Calibre xACT Parasitic Extraction For Transistor-Level Designs 25

Transistor-Level Extraction Setup 25
Creating a Top-level Control File 27
Performing Extraction with Layout Names 28
Performing Extraction with Source Names 30

Chapter 3 Getting Started: Calibre xACT Parasitic Extraction Using Calibre Interactive - xACT 33

Invoking Calibre Interactive - xACT 33
Loading a Calibre xACT Runset (Optional) 34
Specifying a Rule File 34
Input Specifications for Calibre xACT 36
Specifying Design Layout Input 36
Specifying Schematic Netlist Input 38
Output Specifications for Calibre xACT 41
Specifying Extraction Mode 41
Specifying Extraction Type 43
Specifying the Parasitic Netlist 45
Specifying Mask SVDB 48
Running Calibre xACT 49

Chapter 4
Getting Started: Calibre xACT Parasitic Extraction for Digital Designs 51

Digital Design Extraction Flow Setup 51
SVRF Statements Supported By the Digital Extraction Flow 53
Performing Digital Extraction With Calibre xACT 54

Chapter 5
Getting Started: Calibre xACT 3D Parasitic Extraction Using Batch Mode 59

Batch Mode Setup for Calibre xACT 3D 60
Direct Netlisting With Calibre xACT 3D 61
Using the Calibre xACT 3D PDB Flow 63
Step 1 — Creating the PHDB 63
Step 2 — Creating the Parasitic Database 64
Step 3 — Generating A Netlist or Report 67

Chapter 6
Getting Started: Calibre xACT 3D Parasitic Extraction Using Calibre Interactive 71

Invoking Calibre Interactive Parasitic Extraction (PEX) 71
Loading a Runset (Optional) 72
Specifying Rule File for PEX Run 72
Input Specification 74
Defining Input Data Names in the Extracted Netlist 74
Using Schematic Netlist Input in the Extracted Netlist (Optional) 75
Defining H-Cells Input (Gate-Level and Hierarchical Extraction Only) 75
Outputs for a PEX Run 77
Defining the Extraction Type 77
Specifying the Parasitic Netlist 78
Restricting the Nets (Optional) 79
Setting Up Reports (Optional) 79
Adding to the SVDB (Optional) 80
Running Calibre Interactive PEX 81
Setting PEX Options 81

Chapter 7
Getting Started: Calibre xACT Inductance Extraction Using Batch Mode 83

Inductance Extraction Flow Setup 83
SVRF Statements Supported by the Inductance Extraction Flow 84
Performing Inductance Extraction With Calibre xACT 85

Chapter 8
Types of Extraction 89

Comparison of Extraction Types 89
Calibre xACT Extraction Types 91
Calibre xACT Flat Transistor-Level Extraction 91
Selective Resistance Extraction 91
Selective Inductance Extraction 92
Multiple Netlist Generation 92
3D Select Extraction 93
Flat LEF/DEF Extraction 93
Block-by-block Hierarchical Extraction 94
Full Hierarchical LEF/DEF Extraction 95
GDS and OASIS Metal Fill Extraction 95
DEF Metal Fill Extraction 97
Extraction With GDS Macrocells 97

Multi-Temperature Extraction 97
Multi-Corner Multi-Temperature Extraction 98
TSV Extraction 98
Calibre xACT 3D Extraction Types 102
Hybrid xACT 3D/Rule-Based Extraction 102
Extraction Using Boundary Conditions 102
Hierarchical Memory Extraction 104
Mixed-Signal Hierarchical Extraction 107
Calibre xACT 3D Gate-Level Extraction 107
Calibre xACT 3D Flat Transistor-Level Extraction 108

Chapter 9
Producing Parasitic Models 109

Parasitic Devices 109
Types of Parasitic Models 110
Distributed Resistance 110
Distributed Resistance and Capacitance 111
Distributed Resistance and Coupled Capacitance 112

Chapter 10
Basic Extraction Methods 115

Prerequisites for Performing Parasitic Extraction 116
Running Gate-Level Extraction 117
Creating a Gate-Level Netlist with Calibre xACT 117
Creating a Gate-Level Netlist with Calibre xACT 3D 118
Creating a Gate-Level Netlist from Calibre Interactive with Calibre xACT 3D 119
Running Transistor-Level Extraction 121
Creating a Transistor-Level Netlist with Calibre xACT 121
Creating a Transistor-Level Netlist Using Calibre xACT 3D Direct Netlisting 122
Creating a Transistor-Level Netlist Using the Calibre xACT 3D PDB Extraction Flow . . 123
Creating a Transistor-Level Netlist from Calibre Interactive with Calibre xACT 3D 124
Running Full Hierarchical and Mixed-Signal Hierarchical Extraction 126
Creating a Hierarchical Netlist from the Command Line 126
Creating a Hierarchical Netlist from Calibre Interactive 127
Running TSV Extraction 130
Extracting TSV Coupling for an Analog Design 130
Extracting TSV Coupling for a Digital Design 132
Netlisting a Design Without Parasitics 136
Creating an Ideal Netlist from the Command Line 136
Creating an Ideal Netlist from Calibre Interactive 137
Backannotating Parasitics to a Source Netlist 138
Backannotating from the Command Line Using Calibre xACT 3D Direct Netlisting 138
Backannotating from the Command Line Using the Calibre xACT 3D PDB Flow 139
Backannotating from Calibre Interactive 140
Generating a Capacitance Summary Report 141
Net-to-Net Coupling Capacitance Report 142
Reporting Coupled Capacitance from the Command Line 142
Reporting Coupled Capacitance from Calibre Interactive 143

Point-to-Point Resistance Reports 145
Reporting Net Resistance from the Command Line 145
Reporting Net Resistance from Calibre Interactive 146
Top Level Only Extraction 148

Chapter 11
Handling Input 149

Hierarchy Control with Xcells 149
Importing GDS Cell Views 153
Metal Fill Modeling 155
Modeling Multiple Ground Regions 157
Varying Thickness with CMP Files 158

Chapter 12
Tuning Extraction 161

Calibre xACT Processing Control 161
Extracting Net Resistance By Layer 162
Device Extraction Without Parasitics 163
Select-Net Extraction with Calibre xACT and Calibre xACT 3D 164
Select-Layer Extraction with Calibre xACT and Calibre xACT 3D 166
Extracting Particular Nets with the Calibre xACT 3D PDB flow 167
Exclusion of Power and Ground Nets 168
Coupling Capacitance Output Control 169
Grounding Coupled Capacitors 169
Ignore or Extract Floating Nets 169
Resistance Extraction and PERC 171

Chapter 13
Controlling Netlisting 173

Netlisting Multiple Corners and Multiple Temperatures 173
Extracting Multiple Corners with Calibre xACT 175
Generating Multiple Netlists With Calibre xACT 178
Netlisting Only Direct Devices on a Selected Net 179
Methods for Correcting Pin Swapping 180
Using the Source Based Flow 181
How to Join a Disjoint Parasitic Model 183
Port Names for Net Names 184
Verification of Timing with Probe Points 184
Parasitic Extraction with Calibre View Device Properties 184

Chapter 14
Integration and Troubleshooting Topics 185

Integration 185
Optimization 188
Troubleshooting 191
Setting Up For Troubleshooting 191
Invocation Issues 191

Chapter 15
Handling Parasitic On-Chip Variation 193
On-Chip Variation in Parasitic Extraction 193
Parasitic Extraction Techniques for On-Chip Variation 196
In-Die Variation 196
CMP Modeling 197

Chapter 16
Calibre xACTView 201
Calibre xACTView Input 201
Calibre xACTView Invocation 201
Calibre xACTView GUI Features 205
Calibre xACTView Session Window 205
FS3 Display Window 206
Layer Palette 215
Axes Window 217
Menus 218
Toolbars 230
Ruler 232

Chapter 17
Calibre xACT Tool Invocation Reference 235
Setting the CALIBRE_HOME Environment Variable 235
Command Invocation Reference 236
calibre -lvs 237
calibre -xact 238
calibre -xact -phdb 241
calibre -xact -3d -pdb 243
calibre -xact -fmt 247
xactview 251

Appendix A
Reduction Techniques 253
Capacitive and Resistive Reduction 253
Threshold-Based Reduction 254
TICER 255

Appendix B
Error and Warning Messages 257
Error Messages 257
Warning Messages 258

Glossary

Index

Third-Party Information

List of Figures

Figure 1-1. Calibre xACT in the Digital Design Flow 18
Figure 1-2. Calibre xACT in the Full Custom Design Flow 19
Figure 2-1. Calibre xACT Extraction Flow 25
Figure 3-1. Calibre Interactive - xACT Initial View 35
Figure 3-2. Outputs Pane for xACT 3D xACT Mode Selection 43
Figure 4-1. Calibre xACT Digital Extraction Flow 51
Figure 5-1. Calibre xACT 3D Direct Netlisting Extraction 59
Figure 5-2. Calibre xACT 3D PDB Extraction 60
Figure 6-1. Loading Rules in Calibre Interactive 73
Figure 6-2. Providing Source Netlist to Calibre Interactive 75
Figure 6-3. Completing the H-Cells Tab 76
Figure 6-4. Extraction Type Settings 77
Figure 6-5. Describing the Output Format 79
Figure 6-6. Enabling Report Output 80
Figure 8-1. TSV Models 99
Figure 8-2. Distance Between TSVs 99
Figure 8-3. TSV Circuit for Analog Flow 100
Figure 8-4. TSV Circuit for Digital Flow 101
Figure 8-5. Top View of Cell for Boundary 103
Figure 8-6. Cell Array Showing REFLECTIVE Versus PERIODIC 104
Figure 8-7. Hierarchical vs. Actual RC Network Examples 106
Figure 9-1. Simplified Layout for Distributed Resistance 110
Figure 9-2. Distributed Resistance Extraction 111
Figure 9-3. Simplified Layout for Distributed Resistance and Capacitance 111
Figure 9-4. Distributed Resistance and Capacitance Extraction 112
Figure 9-5. Simplified Layout for Distributed Resistance With Coupled Capacitance 113
Figure 9-6. Distributed Resistance With Coupled Capacitance 113
Figure 10-1. Specifying HCell and XCell Files in Calibre Interactive 120
Figure 10-2. Extraction Mode 120
Figure 10-3. Gate Level Setting 120
Figure 10-4. Extraction Mode Setting 125
Figure 10-5. Transistor Level Setting 125
Figure 10-6. Inputs Pane for Hierarchical Extraction 128
Figure 10-7. Set Extraction Mode 128
Figure 10-8. Hierarchical Extraction Type Setting 129
Figure 10-9. Setting for No Parasitics 137
Figure 10-10. Coupling Capacitance Report Settings 143
Figure 10-11. Point-to-Point Resistance Report Settings 147
Figure 11-1. Simple Metal Fill on a Single Layer 156
Figure 11-2. Metal Fill on Multiple Layers With Multiple Nets 157

Figure 11-3. Non-Square Fill With Multiple Nets 157
Figure 12-1. Device Extraction With and Without Parasitics 164
Figure 12-2. Floating-Net Coupling Floating Example 170
Figure 12-3. Floating-Net Coupling Extraction Example 171
Figure 13-1. Gate-Level Logical Pin Swapping Example 180
Figure 13-2. Comparison of Normal and Source Based Flows 182
Figure 14-1. Example Shell Script to Run Calibre 188
Figure 15-1. Design Flow Showing In-Die Variation Techniques 195
Figure 15-2. Layout Structure Affects Local Density 196
Figure 15-3. Typical Effects of CMP 198
Figure 15-4. Metal Fill and CMP 198
Figure 16-1. Input Files Dialog 202
Figure 16-2. Select FS3 File Dialog 203
Figure 16-3. Select PDB Dialog 203
Figure 16-4. Select fs3.out file Dialog 204
Figure 16-5. Select cfcap.fs3.out file Dialog 204
Figure 16-6. Session Window For Calibre xACTView 206
Figure 16-7. Net and Shape Information Display 208
Figure 16-8. Net and Shapes Window 208
Figure 16-9. Net and Shape Highlighting 209
Figure 16-10. Devices Window 210
Figure 16-11. Highlighted Devices 210
Figure 16-12. Net Coupling Window 211
Figure 16-13. Shape Coupling Window 212
Figure 16-14. Layer Coupling Window 213
Figure 16-15. Ignored Capacitance:Layers Window 213
Figure 16-16. Ignored Capacitance:Layers From 214
Figure 16-17. Ignored Capacitance Highlighting 215
Figure 16-18. Layer Palette 216
Figure 16-19. Layer Transparency 217
Figure 16-20. Axes Window 218
Figure 16-21. File Menu 219
Figure 16-22. File Menu With Export Net/Layer Coupling Information 219
Figure 16-23. Save to file Dialog Box for Export View 220
Figure 16-24. Save to file Dialog Box for Export Net/Layer Coupling Information 220
Figure 16-25. View Menu 221
Figure 16-26. Layers Menu 223
Figure 16-27. Options Menu 225
Figure 16-28. Preferences Dialog Box - Ruler Tab 225
Figure 16-29. Preferences Dialog Box - Grid Tab 226
Figure 16-30. Preferences Dialog Box - Cross Section Tab 226
Figure 16-31. Preferences Dialog Box - View Tab 227
Figure 16-32. Reset Preferences 227
Figure 16-33. Save Layer Properties Dialog 228
Figure 16-34. Load Layer Properties Dialog 228

Figure 16-35. Window Menu. 229
Figure 16-36. Window Menu With Coupling 229
Figure 16-37. Hidden Windows. 230
Figure 16-38. Help Menu. 230
Figure 16-39. Horizontal Toolbar 230
Figure 16-40. Vertical Toolbar 232
Figure 16-41. Ruler Measurement 233
Figure 16-42. Info Dialog for Ruler Limit. 233
Figure 16-43. Multiple Ruler Measurements 234

List of Tables

Table 1-1. Syntax Conventions 23
Table 3-1. Invocation Methods for Calibre Interactive - xACT 34
Table 3-2. Inputs Pane Contents 36
Table 3-3. Fields in Outputs Pane 41
Table 5-1. Invocation Line for PDB Step in Calibre xACT 3D PDB Flow 67
Table 5-2. Formatter Options 68
Table 6-1. Invocation Methods for Calibre Interactive PEX 72
Table 6-2. Tabs in Inputs Pane 74
Table 6-3. Tabs in Outputs Pane 77
Table 6-4. Capabilities Available in PEX Options 82
Table 8-1. Tool Support For Extraction Types 90
Table 11-1. Xcell Flags 151
Table 11-2. Wildcards in an Xcell List 153
Table 14-1. Calibre Interactive Settings Sequence 186
Table 14-2. Best Practices for Shell Scripts 187
Table 16-1. File Menu Item Descriptions 219
Table 16-2. View Menu Item Descriptions 221
Table 16-3. Layers Menu Item Descriptions 223
Table B-1. Calibre xACT Error Messages 257
Table B-2. Calibre xACT Warning Messages 258

Chapter 1
Calibre xACT Product Overview

The Calibre xACT products consist of Calibre® xACT™, Calibre xACT 3D, and Calibre xACTView.

The following sections provide an overview of the Calibre xACT product line:

The Calibre xACT Tools 17
Calibre xACT Digital Design Flow 17
Calibre xACT and the Full Custom Design Flow 18
Calibre xACT Digital Flow Overview 19
Calibre xACT Overview 20
Calibre xACT 3D Overview 21
Calibre xACT Inductance Flow Overview 22
Calibre xACTView Overview 23
Syntax Conventions 23

The Calibre xACT Tools

The Calibre xACT products consist of Calibre xACT, Calibre xACT 3D, and Calibre xACTView.

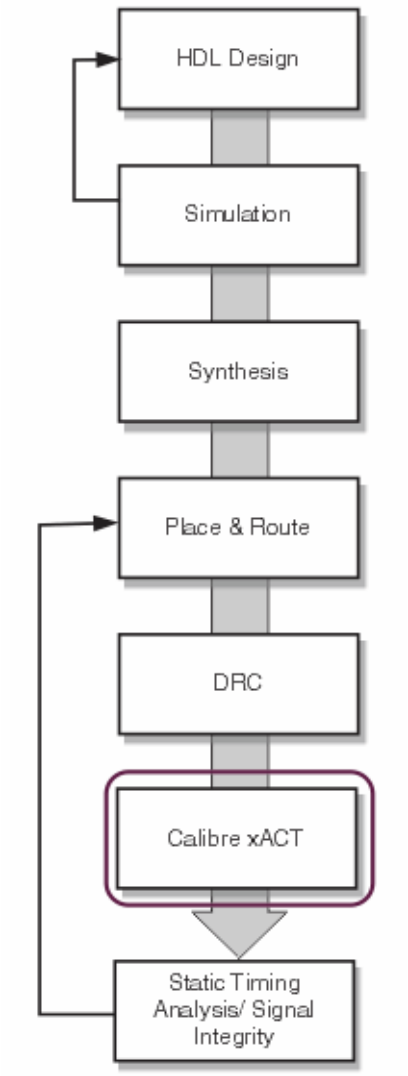
- Calibre xACT is a high-performance, high-accuracy parasitic extraction tool for leading-edge transistor-level and digital design.
- Calibre xACT 3D is a high-accuracy parasitic extraction tool that uses fast field solver technology for capacitance extraction.
- Calibre xACTView is a 3D layout viewer that provides the ability to graphically view the different polygons processed for capacitance extraction by Calibre xACT 3D.

Calibre xACT Digital Design Flow

Calibre xACT calculates parasitic and coupling effects. This information is essential for driving static timing analysis tools. Calibre xACT sits between Place & Route and static timing analysis in the digital design flow. You should run this tool only after it passes DRC and LVS checks.

Parasitic extraction runs before static timing analysis. The cycle of Place & Route, DRC, parasitic extraction, and static timing analysis is often repeated several times.

Figure 1-1. Calibre xACT in the Digital Design Flow



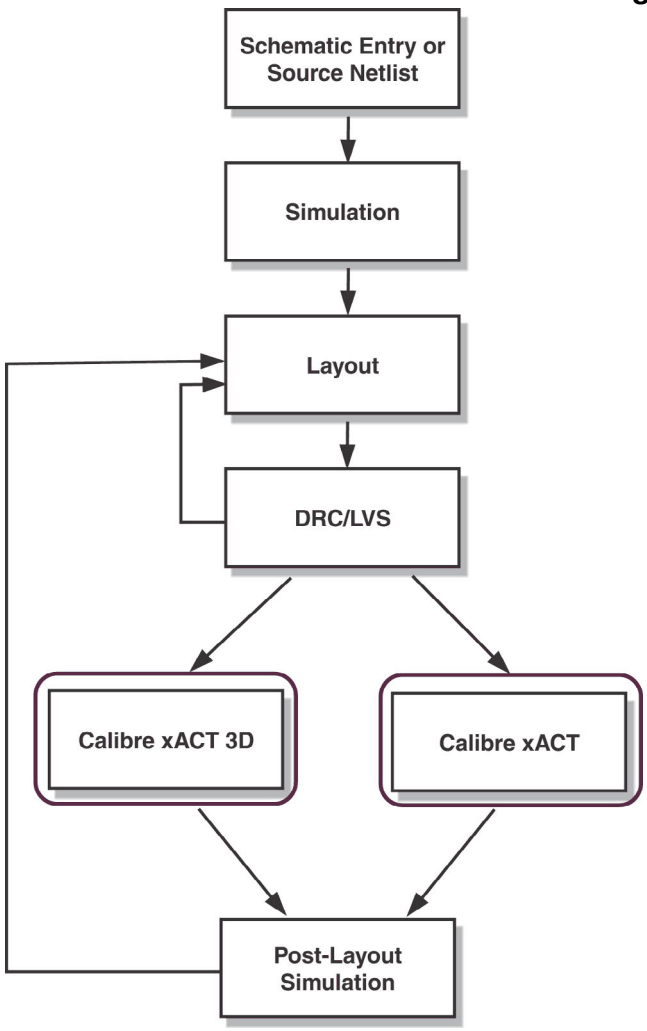
Calibre xACT and the Full Custom Design Flow

Calibre xACT and Calibre xACT 3D provide information about parasitic and coupling effects that are essential for full custom design flows. After full custom layout is completed, and DRC and LVS checks are clean, use Calibre xACT or Calibre xACT 3D to generate accurate netlists for use in simulation tools.

Calibre xACT and Calibre xACT 3D require clean connectivity data. You should run these tools only after your layout passes LVS checks. If the layout is not electrically correct, the parasitic results will not apply to the final design.

After running parasitic extraction, you may choose to simulate your design. You can also use the reports to identify the most affected nets. Usually the cycle of physical verification, parasitic extraction, and post-layout verification is repeated several times before tapeout.

Figure 1-2. Calibre xACT in the Full Custom Design Flow



Calibre xACT Digital Flow Overview

Calibre xACT encompasses sign-off parasitic extraction capabilities for cell-based (digital) flows as well as for transistor-based (custom). The digital extraction capabilities of Calibre xACT deliver the full-chip performance and sign-off accuracy needed for large digital designs. Calibre xACT reads LEF and DEF data directly, uses foundry-supported rule decks, and retains all connectivity information from the original design. It generates SPEF netlist output that is then used for static timing analysis. Although the Calibre xACT transistor-based (custom) flows only support 16nm and smaller processes, the Calibre xACT cell-based (digital) flows support all foundries and process geometries for which Calibre® xRC™ or Calibre xACT extraction rules are available.

Use Calibre xACT to calculate the parasitics in your design. Parasitic effects can slow down signals or add noise in your design, among other problems.

Calibre xACT Digital Design Flow Requirements

You must have the following to run the Calibre xACT tool in a digital flow:

- Design DEF file(s) with defined connectivity.
- A technology LEF file.
- All other LEF files such as cell libraries, IP blocks, and via definitions.
- A top-level SVRF control file containing Calibre-specific SVRF statements and operations.
- A layer definitions SVRF rule file.
- Resistance and capacitance rule files from the foundry.
- The Calibre and Calibre xACT software.
- All required product licenses. For Calibre xACT licensing information, see the "Licensing: Parasitic Extraction Products" section of the *Calibre Administrator's Guide*.
- Either the $MGC_HOME or $CALIBRE_HOME environment variables defined. Calibre tools require that the CALIBRE_HOME environment variable be set. See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

Run the Calibre xACT tool from the command line. See "Getting Started: Calibre xACT Parasitic Extraction for Digital Designs" for details.

Calibre xACT Overview

Calibre xACT is a high-performance high-accuracy parasitic extraction tool that calculates parasitic capacitance and resistance in an IC layout and generates transistor-level parasitic netlists and reports. The netlists can be simulated with transistor-level post-layout analysis tools.

Use Calibre xACT to calculate the parasitics in your design. Parasitics can slow down signals, add noise, or cause hot spots in your design, among other problems.

Calibre xACT Requirements

You must have the following to run the Calibre xACT tool:

- A layout database with defined connectivity.

- An optional source netlist, if you require schematic names for the nets in the parasitic netlist.
- A rule file containing Calibre-specific SVRF statements and operations.
- The Calibre software.
- All required product licenses. For Calibre xACT licensing information, see the "Licensing: Parasitic Extraction Products" section of the *Calibre Administrator's Guide*.
- Either the $MGC_HOME or $CALIBRE_HOME environment variables defined. Calibre tools require that the CALIBRE_HOME environment variable be set. See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

Run the Calibre xACT tool from the command line. For details, see "Getting Started: Calibre xACT Parasitic Extraction For Transistor-Level Designs".

Calibre xACT 3D Overview

Calibre xACT 3D is a high-accuracy parasitic extraction tool that generates parasitic netlists and reports using fast field solver technology for capacitance extraction. It calculates parasitic resistance and capacitance in an IC layout and outputs a simulatable netlist.

Use Calibre xACT 3D to calculate the parasitics in your design. Parasitic effects can slow down signals, add noise, or cause hot spots in your design, among other problems. When used in conjunction with the Calibre® xL product in the Calibre xACT 3D PDB flow, it can also calculate parasitic inductance.

Calibre xACT 3D Requirements

You must have the following to run the Calibre xACT 3D tool:

- A layout database with defined connectivity.
- An optional source netlist, if you require schematic names for the nets in the parasitic netlist.
- A rule file containing Calibre-specific SVRF statements and operations.
- The Calibre software.
- All required product licenses. For Calibre xACT 3D licensing information, see the "Licensing: Parasitic Extraction Products" section of the *Calibre Administrator's Guide*.
- Either the $MGC_HOME or $CALIBRE_HOME environment variables defined. Calibre tools require that the CALIBRE_HOME environment variable be set. See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

Run the Calibre xACT 3D tool using one of the following methods:

- From the command line using the direct netlisting flow. See "Direct Netlisting With Calibre xACT 3D".
- From the Calibre® Interactive™ graphical user interface. See "Getting Started: Calibre xACT 3D Parasitic Extraction Using Calibre Interactive".
- From the command line using the PDB flow. See "Using the Calibre xACT 3D PDB Flow".

Calibre xACT Inductance Flow Overview

As designs become more dense and interconnects get longer it is important to consider parasitic inductance effects for sensitive and high frequency designs. The inductance extraction capabilities of Calibre xACT are fully integrated within the Calibre LVS extraction solution. Use the Calibre xACT Inductance flow to generate optimized and accurate RLC netlists needed for fast post-layout simulations.

Analog/RF design teams can use this flow to consider inductance effects on sensitive and high frequency signals in their designs. Digital (LEF/DEF) design teams can use this flow to analyze inductance effects in high speed signals such as clocks or data lines, achieving better timing closure for their digital designs.

Calibre xACT Inductance Flow Requirements

You must have the following to run the Calibre xACT tool in an inductance flow:

- Digital or analog design files.
- A top-level SVRF control file containing Calibre-specific SVRF statements and operations.
- A layer definitions SVRF rule file.
- Resistance and capacitance rule files from the foundry.
- The Calibre and Calibre xACT software.
- All required product licenses including Calibre xACT and Calibre xL licenses. For complete Calibre xACT licensing information, see the "Licensing: Parasitic Extraction Products" section of the *Calibre Administrator's Guide*.
- Either the $MGC_HOME or $CALIBRE_HOME environment variables defined. Calibre tools require that the CALIBRE_HOME environment variable be set. See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

Run the Calibre xACT tool from the command line. See "Getting Started: Calibre xACT Inductance Extraction Using Batch Mode" for details.

Calibre xACTView Overview

Calibre xACTView is a 3D layout viewer used to visualize and debug test cases, and to view the technology stack used in integrated circuit designs. A 3D layout viewer is an essential component of standard field solvers. It provides the ability to graphically view the different polygons processed for capacitance extraction as seen by the field solver.

Use this tool to create golden reference parasitic capacitance for small interconnect structures or around devices. This is typically done at foundries in order to establish a golden reference against which other tools are qualified. This tool also verifies Calibre xACT 3D setup, which includes LVS and PEX rules.

Calibre xACTView Requirements

The requirements for running Calibre xACTView include the following:

- The FS3 file containing the layer descriptions, 3D layout polygon descriptions, and extraction commands for the field solver.
- A Linux machine with OpenGL installed. In order to invoke Calibre xACTView from a remote machine, you must use a VNC software version that supports OpenGL.
- The Calibre software installed.
- All required product licenses for Calibre xACT 3D. For licensing information, see the "Licensing: Parasitic Extraction Products" section of the *Calibre Administrator's Guide*. Calibre xACTView does not require a separate license.
- Either the $MGC_HOME or $CALIBRE_HOME environment variables defined. Calibre tools require that the CALIBRE_HOME environment variable be set. See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

Run the Calibre xACTView tool from the command line. See "Calibre xACTView" for details.

Syntax Conventions

The command descriptions use font properties and several metacharacters to document the command syntax.

Table 1-1. Syntax Conventions

Convention	Description
Bold	Bold fonts indicate a required item.

Table 1-1. Syntax Conventions (cont.)

Convention	Description
Italic	Italic fonts indicate a user-supplied argument.
`Monospace`	Monospace fonts indicate a shell command, line of code, or URL. A bold monospace font identifies text you enter.
Underline	Underlining indicates either the default argument or the default value of an argument.
UPPercase	For certain case-insensitive commands, uppercase indicates the minimum keyword characters. In most cases, you may omit the lowercase letters and abbreviate the keyword.
[ ]	Brackets enclose optional arguments. Do not include the brackets when entering the command unless they are quoted.
{ }	Braces enclose arguments to show grouping. Do not include the braces when entering the command unless they are quoted.
‘ ’	Quotes enclose metacharacters that are to be entered literally. Do not include single quotes when entering braces or brackets in a command.
| or ||	Vertical bars indicate a choice between items. Do not include the bars when entering the command.
…	Three dots (an ellipsis) follows an argument or group of arguments that may appear more than once. Do not include the ellipsis when entering the command.
Example: **DEVice** {***element_name*** [‘(’*model_name*‘)’]} ***device_layer*** {***pin_layer*** [‘(’*pin_name*‘)’] …} [‘<’*auxiliary_layer*‘>’ …] [‘(’*swap_list*‘)’ …] [BY NET | BY SHAPE]	

Note

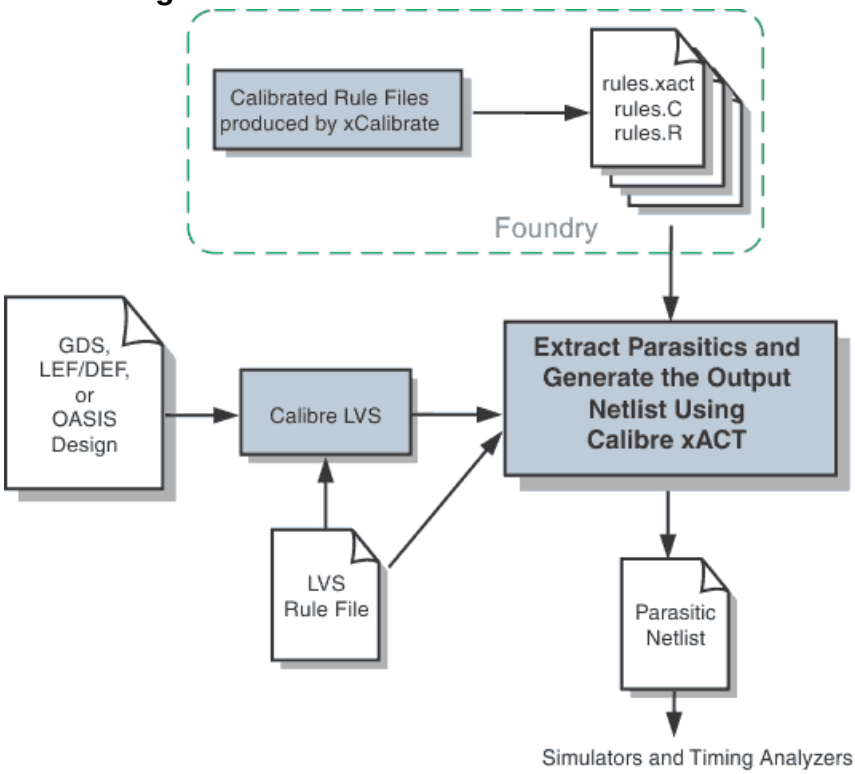
Omit braces, vertical bars, and notational fonts listed above when you actually enter a command.

Chapter 2
Getting Started: Calibre xACT Parasitic Extraction For Transistor-Level Designs

The Calibre xACT extraction flow for advanced process nodes offers many possible customizations based on your particular needs.

Figure 2-1 illustrates the Calibre xACT direct netlisting extraction flow.

Figure 2-1. Calibre xACT Extraction Flow



Transistor-Level Extraction Setup 25
Creating a Top-level Control File 27
Performing Extraction with Layout Names 28
Performing Extraction with Source Names 30

Transistor-Level Extraction Setup

Certain conditions must be met before running the Calibre xACT tool.

These instructions assume that everything is correctly set up and configured:

- Calibre software is installed.

  Calibre xACT is packaged in the AOI CALIBRE_HOME executable, which runs on Red Hat Enterprise Linux 6 (RHEL 6) or SLES 11 Service Pack 2 operating systems. Calibre xACT does not run on RHEL 5 or earlier, or SLES 11 Service Pack 1 or earlier.

- Calibre xACT and Calibre xL licenses are available.
- Either the $MGC_HOME or $CALIBRE_HOME environment variables are defined.

  See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

- A layer definitions file. This file defines the metal layers that Calibre xACT will extract, and the connectivity between them. The Layer and Connect statements are typically found in the Calibre LVS rule file available for download from the foundry web site. You may also create your own SVRF layer definitions based on a list of layer names and connection definitions.
- An LVS rule file. This file contains device recognition and parameter extraction information.
- Calibre xACT rule files (*rules.xact* and *rules.ptf*) created with xCalibrate.
- A capacitance rule file (*rules.C*) created with xCalibrate. Calibre xACT, Calibre xRC, and Calibre xACT 3D use the same capacitance rule file.
- A resistance rule file (*rules.R*) created with xCalibrate. Calibre xACT, Calibre xRC, and Calibre xACT 3D use the same resistance rule file.

  Use calibrated rule files generated with the 2014.2 or newer versions of xCalibrate. Using calibrations and rule files generated with previous versions will generate a warning.

- A top level SVRF control file with layers, connectivity, devices, and parasitic calculations specified.

Caution

SVRF rule files that contain deprecated or obsolete SVRF statements are not guaranteed to work properly with Calibre xACT.

Tip

Shell scripts are an excellent way to run Calibre from the command line. A script can explicitly set environment variables and record invocation combinations you use frequently.

Creating a Top-level Control File

The top-level control file is an SVRF rule file that contains all the settings necessary to run Calibre xACT. This example describes a set of statements used in a GDS flow.

Procedure

1. Create a Calibre xACT rule file called *xact_t.rules*.
2. Using a text editor, include the following statements in the *xact_t.rules* file:

```
// design layout information - GDS
LAYOUT PRIMARY "mytopcell"
LAYOUT PATH "design/mytopcell.gds"
LAYOUT SYSTEM GDSII

// design netlist information
SOURCE PRIMARY "mytopcell"
SOURCE PATH "design/mytopcell.src.net"
SOURCE SYSTEM SPICE

// specify extracted netlist output - DSPF
PEX NETLIST "netlist.dspf" DSPF SOURCENAMES
PEX EXTRACT EXCLUDE SOURCENAMES VDD VSS
PEX EXTRACT TEMPERATURE 27
PEX REDUCE ANALOG YES
UNIT CAPACITANCE fF // May already be in capacitance file

// specify LVS and parasitic extraction rule files
INCLUDE rules/calibre_layers.svrf
INCLUDE rules/calibre_lvs_rules.svrf
INCLUDE rules/rules.C
INCLUDE rules/rules.R
INCLUDE rules/rules.xact
```

This rule file uses the following SVRF statements:

- Layout System, Layout Path, and Layout Primary statements specify the layout database. The layout database, also called the design, is a physical representation of your IC.

  Calibre xACT accepts GDS, OASIS®[1], and LEF/DEF design formats.

- If you require source or schematic names in the generated netlist, use Source System, Source Path, and Source Primary statements to specify the source netlist.

  If you require source (schematic) names in the generated netlist, you need the source netlist. This netlist must be in SPICE format and is identified in the SVRF rule file.

1. OASIS® is a registered trademark of Thomas Grebinski and licensed for use to SEMI®, San Jose. SEMI® is a registered trademark of Semiconductor Equipment and Materials International.

- PEX Netlist generates a netlist with parasitic elements and places it in the specified file.

  Calibre xACT supports HSPICE, DSPF, SPEF, SPECTRE, ELDO, and CALIBREVIEW netlist formats. When performing extraction with source names be sure to specify the SOURCENAMES keyword.

- PEX Extract Exclude specifies to exclude the named nets from parasitic extraction. When performing extraction with source names be sure to specify the SOURCENAMES keyword.
- PEX Extract Temperature specifies a temperature dependency for the extracted resistance.
- PEX Reduce Analog specifies to perform reduction of extracted data from analog designs.
- Unit Capacitance specifies the capacitance scale factor for device calculations. This may already be defined in your *rules.C* file.
- Include specifies to include statements from the specified rule file in the top-level rule file.

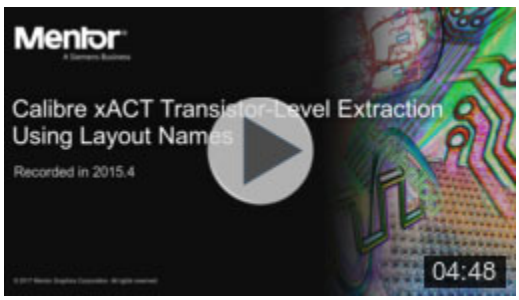
3. Save and close the file.

Performing Extraction with Layout Names

When performing extraction with layout names, Calibre xACT runs all stages of extraction including device recognition, parasitic calculations, and netlisting.

The following video:

- Introduces the Calibre xACT tool
- Reviews the Calibre xACT direct netlisting flow using layout names
- Reviews the extraction rule file contents
- Demonstrates tool execution
- Reviews results



Prerequisites

- You have met the requirements outlined in Transistor-Level Extraction Setup.
- You have created a top-level control file. See Creating a Top-level Control File for an example. Note: in the example, you must replace the SOURCENAMES keyword specified for the PEX Netlist and PEX Extract Exclude statements with the LAYOUTNAMES keyword to perform extraction with layout names.

Procedure

1. Verify the top-level control file contains the following statements to specify the "inputs", a layout database:

   - Layout System
   - Layout Path
   - Layout Primary

   The layout database, also called the design, is a physical representation of your IC. The Calibre xACT software can read GDS, OASIS, and LEF/DEF formats.

2. Determine which parasitics you need. The choices are:

   - **Resistance (-r)** — Exclusively writes resistance models into the netlist.
   - **Lumped Capacitance (-c)** — Exclusively writes capacitance models into the netlist.
   - **Resistance and Distributed Capacitance (-rc)** — Specifies distributed RC parasitic extraction and writes the distributed RC models into the netlist.
   - **Resistance with Distributed Capacitance and Coupled Capacitance between nets (-rcc)** — Specifies R-coupled-C extraction and writes fully-coupled models in to the netlist. This is the default if no option is specified.

   There is a trade-off between the amount of detail and how long the netlist takes to simulate. For example, a netlist with parasitics as lumped capacitance (-c) takes less time to simulate than one with coupled capacitance (-rcc) between nets, which takes less time than one with coupled capacitance between nets including floating nets.

3. Determine the output you need.

   - Netlists are useful for simulation. The format you need depends on your simulator. Only one format of netlist is generated at a time. Calibre xACT supports HSPICE, DSPF, SPEF, SPECTRE, ELDO, and CALIBREVIEW netlist formats.
   - Reports are useful for identifying the nets or cells most affected by parasitics, or for post-processing in spreadsheets or with scripts. Reports can be generated at the same time as netlists.

4. Output the netlist or report(s). Using the top-level control file, *xact_t.rules*, run Calibre xACT to create the DSPF netlist:

   ```
   calibre -xact -rcc xact_t.rules
   ```

   The output depends on what values are set in the top-level control file and what parasitics were specified on the command line.

Results

Calibre xACT ends the transcript with a summary of errors and warnings. Be sure to check for any errors; these invalidate results.

The working directory also contains the requested netlist and reports. For this example, the tool creates a netlist named *netlist.dspf*.

Related Topics

calibre -xact

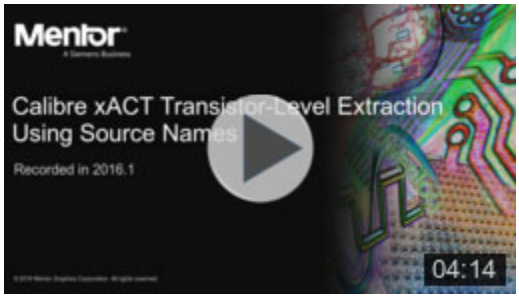
Creating a Top-level Control File

Performing Extraction with Source Names

When performing extraction with source names, Calibre LVS must be run for device recognition. Calibre xACT runs the remaining stages of extraction including parasitic calculations and netlisting.

The following video:

- Introduces the Calibre xACT tool
- Reviews the Calibre xACT direct netlisting flow using source names
- Reviews the extraction rule file contents
- Demonstrates tool execution
- Reviews results



Prerequisites

- You have met the requirements outlined in Transistor-Level Extraction Setup.
- You have created the top-level control described in Creating a Top-level Control File.

Procedure

1. Verify the SVRF rule file contains the following statements to specify the “inputs”, a layout database, and source netlist.

   a. Layout System, Layout Path, and Layout Primary statements specify the layout database.

      The layout database, also called the design, is a physical representation of your IC. You identify this database in your SVRF rule file. The Calibre xACT software can read GDS, OASIS, and LEF/DEF formats.

   b. Source System, Source Path, and Source Primary statements specify the source netlist.

      If you require source (schematic) names in the generated netlist, you need the source netlist. This netlist must be in SPICE format and is identified in the SVRF rule file. You must also specify the SOURCENAMES keyword for the PEX Netlist statement, as well as for statements PEX Extract Exclude or PEX Extract Include when used.

2. Determine which parasitics you need. The choices are:

   - **Resistance (-r)** — Exclusively writes resistance models into the netlist.
   - **Lumped Capacitance (-c)** — Exclusively writes capacitance models into the netlist.
   - **Resistance and Distributed Capacitance (-rc)** — Specifies distributed RC parasitic extraction and writes the distributed RC models into the netlist.
   - **Resistance with Distributed Capacitance and Coupled Capacitance between nets (-rcc)** — Specifies R-coupled-C extraction and writes fully-coupled models in to the netlist. This is the default if no option is specified.

   There is a trade-off between the amount of detail and how long the netlist takes to simulate. For example, a netlist with parasitics as lumped capacitance (-c) takes less time to simulate than one with coupled capacitance (-rcc) between nets, which takes less time than one with coupled capacitance between nets including floating nets.

3. Determine the output you need.

   - Netlists are useful for simulation. The format you need depends on your simulator. Only one format of netlist is generated at a time. Calibre xACT supports HSPICE, DSPF, SPEF, SPECTRE, ELDO, and CALIBREVIEW netlist formats.

- Reports are useful for identifying the nets or cells most affected by parasitics, or for post-processing in spreadsheets or with scripts. Reports can be generated at the same time as netlists.

4. Use the top-level control file, *xact_t.rules*, to run Calibre LVS and perform device recognition:

   ```
   calibre -lvs -hier xact_t.rules
   ```

5. Perform parasitic calculations and output the netlist and/or report(s):

   ```
   calibre -xact -rcc xact_t.rules
   ```

   The output is controlled by rules specified in the top-level control file and parasitics specified on the command line.

Results

Calibre xACT ends the transcript with a summary of errors and warnings. Be sure to check for any errors; these invalidate results.

The working directory contains the requested netlist and reports. The example top-level control file, *xact_t.rules*, creates a netlist named *netlist.dspf*.

Related Topics

calibre -xact

Creating a Top-level Control File

Chapter 3 Getting Started: Calibre xACT Parasitic Extraction Using Calibre Interactive - xACT

The Calibre® Interactive™ Graphical User Interface (GUI) provides an interface to the Calibre xACT tool.

The following sections describe how to use Calibre Interactive - xACT to perform parasitic extraction with Calibre xACT using the direct netlisting flow:

Invoking Calibre Interactive - xACT **33**

Loading a Calibre xACT Runset (Optional) **34**

Specifying a Rule File **34**

Input Specifications for Calibre xACT **36**

Specifying Design Layout Input 36

Specifying Schematic Netlist Input 38

Output Specifications for Calibre xACT **41**

Specifying Extraction Mode 41

Specifying Extraction Type 43

Specifying the Parasitic Netlist 45

Specifying Mask SVDB 48

Running Calibre xACT **49**

Invoking Calibre Interactive - xACT

The Calibre xACT tool has its own graphical user interface (GUI) in Calibre Interactive. The GUI can be invoked from the command line or from Calibre DESIGNrev.

Prerequisites

Environment correctly set up and configured:

- Calibre software installed and optionally integrated with layout editor.
- Calibre Interactive and Calibre xACT licenses are available. See "Licensing: Physical Verification Products" and "Licensing: Parasitic Extraction Products" in the *Calibre Administrator's Guide* for details.
- Either the $MGC_HOME or $CALIBRE_HOME environment variables defined. Calibre tools require that the CALIBRE_HOME environment variable be set. See

"Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

Procedure

Select the appropriate method:

Table 3-1. Invocation Methods for Calibre Interactive - xACT

From…	Use…
Cadence Virtuoso or other layout editor	Select **Calibre** > **Run xACT** from the menu
Calibre DESIGNrev	Select **Verification** > **Run xACT...**
Command line	Type: calibre -gui -xact

Loading a Calibre xACT Runset (Optional)

After you invoke Calibre Interactive - xACT, you are prompted to specify a runset. A runset sets default values and can be useful for managing your different types of extractions.

Procedure

1. Load a runset with the file browser by clicking the Browse (**…**) button.
2. After navigating to the runset, select it and click **Open**. In the main dialog, click **OK**.
3. To skip loading a runset, click **Cancel**.

Specifying a Rule File

Parasitic extraction requires a pre-defined rules file. Use this procedure to specify and load a rule file in Calibre Interactive - xACT.

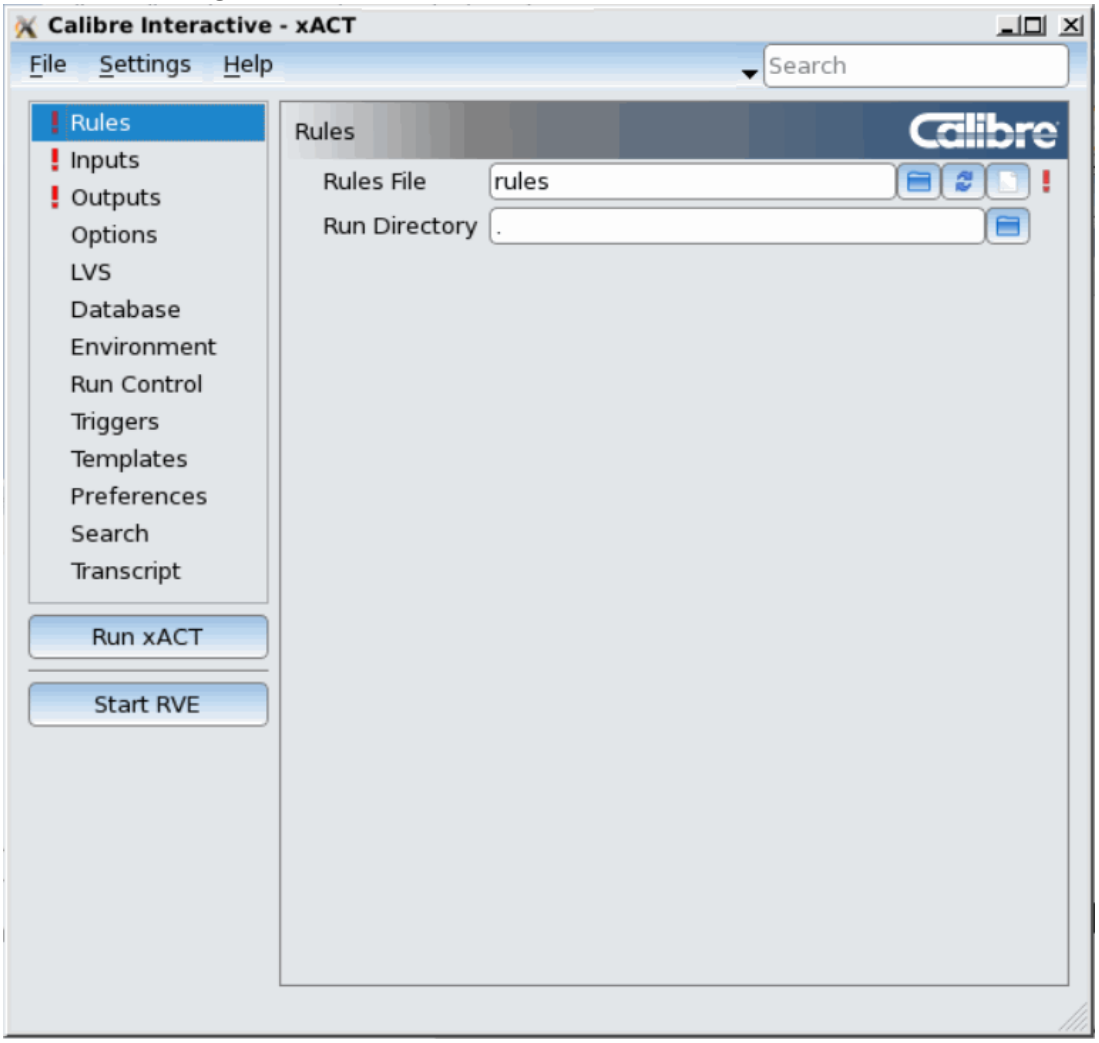
Prerequisites

- You have invoked Calibre Interactive -xACT
- You have a pex.rule file.

Procedure

1. Click **Rules**.

Figure 3-1. Calibre Interactive - xACT Initial View



2. Specify the run directory name. You can use the browse ( ) button to select the run directory name from a list.
3. Specify the rule file name. You can use the Browse button to select the rule filename from a list. You can use the view ( ) button to view or edit the rule file.
4. Click the load ( ) button. This sets the GUI fields and options based on rule file data.

Tip

After you load a rule file, any information you specify in the GUI supersedes information in your loaded rule file.

Input Specifications for Calibre xACT

The input data varies depending on how you invoked Calibre Interactive and the type of extraction you plan to run.

Table 3-2. Inputs Pane Contents

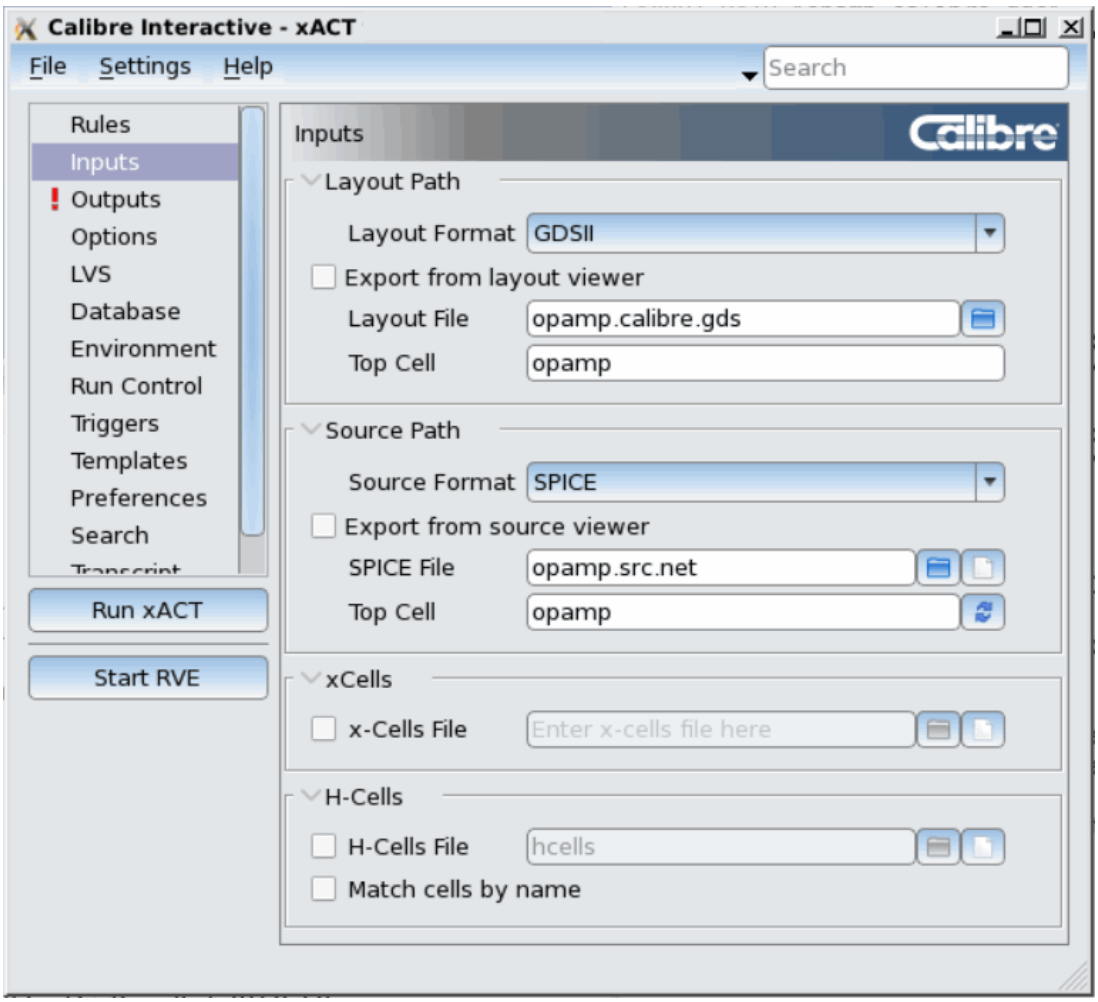
Group	Purpose	Required
Layout Path	Specify the design database and format information. See "Specifying Design Layout Input".	Yes
Source Path	Specify *schematic* netlist or source file information. This input is used when output requires information from the schematic. See "Specifying Schematic Netlist Input".	Yes

Specifying Design Layout Input

The design layout information is specified on the Inputs pane in the Layout Path field. A red exclamation point is displayed next to Inputs if any required settings are specified incorrectly or missing.

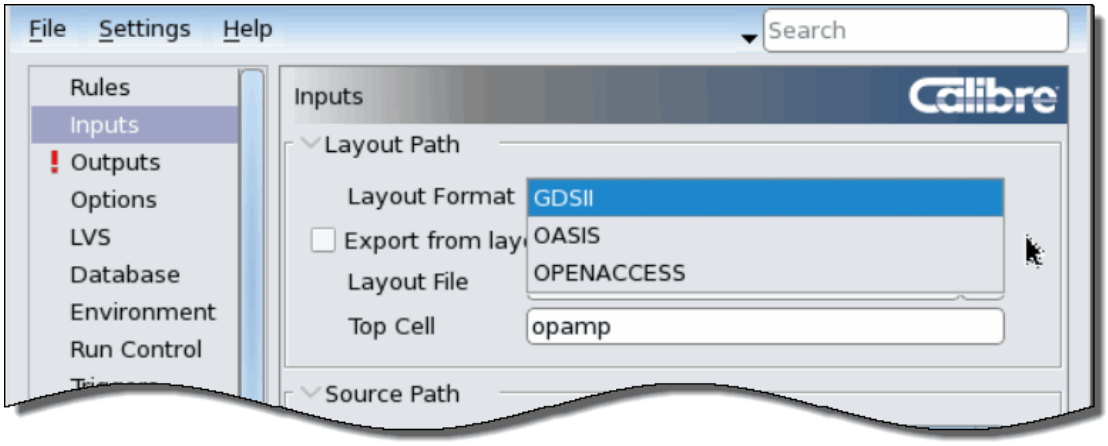
Procedure

1. Click **Inputs** to display the Inputs pane.

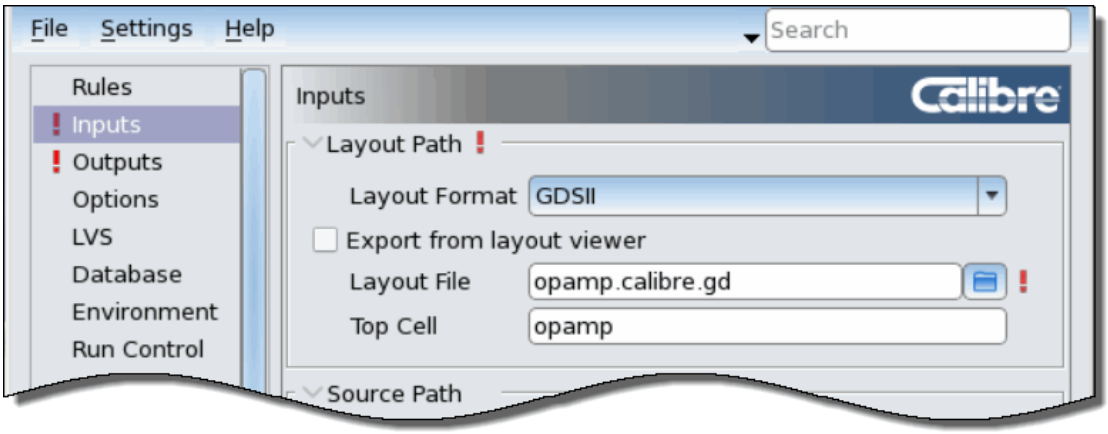


2. Click the dropdown list arrow to display the list of available Layout Format options.

Choices are GDSII, OASIS, and OPENACCESS. Select GDSII.



3. Specify the layout filename. A red exclamation point appears next to the entry box if the layout file name does not exist.



If you invoked Calibre Interactive from a layout editor, you can use the current layout by selecting the Export from layout viewer option. This saves a copy of the layout in the Layout File name you specified. If the filename already exists, the contents are overwritten.

4. Specify the layout top cell name in the Top Cell text box.

Specifying Schematic Netlist Input

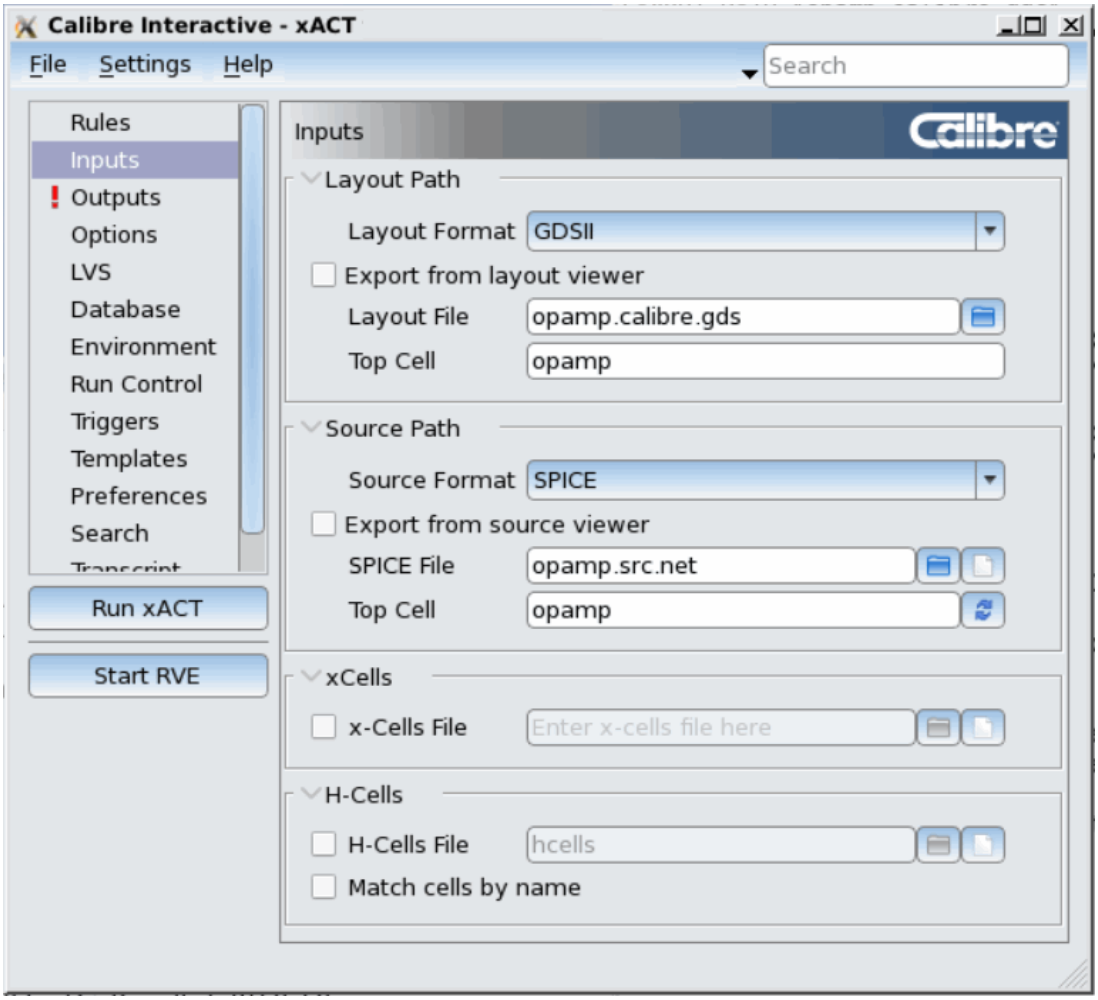
The schematic netlist or source file information is specified on the Inputs pane in the Source Path field. A red exclamation point is displayed next to Inputs if any required settings are specified incorrectly or missing.

Prerequisites

- Design layout input has been specified.

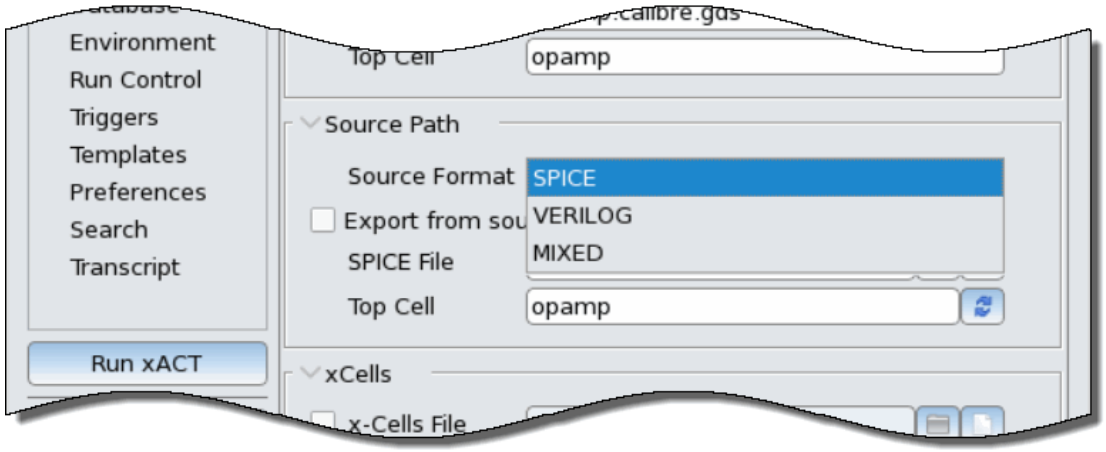
Procedure

1. Click **Inputs** to display the Inputs pane.



2. Click the dropdown list arrow to display the list of available Source Format options.

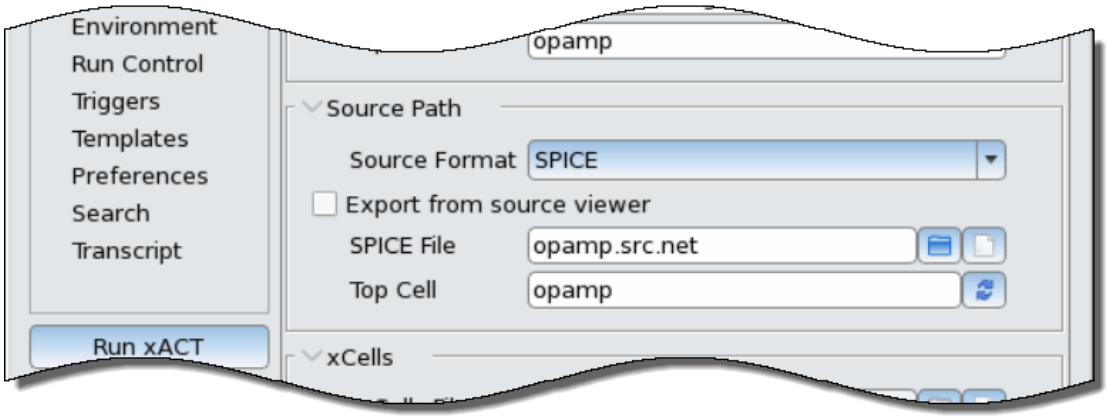
Choices are SPICE, VERILOG, and MIXED. SPICE is selected by default.



3. Specify the schematic file name. A red exclamation point appears next to the entry box if the filename entered does not exist.

   The file name text boxes that are displayed depend on the Source Format that is selected. When Source Format is:

   - **SPICE —** a SPICE File text box is visible.
   - **VERILOG —** a VERILOG File text box is visible.
   - **MIXED —** both a SPICE File and VERILOG File text boxes are visible.



   If you invoked Calibre Interactive from a schematic editor, you can use the current schematic by selecting the Export from source viewer option. This option saves a copy of the schematic in the SPICE or VERILOG File name you specified. If the filename already exists, the contents are overwritten. The Export from source viewer option is not available when MIXED is selected.

4. Specify the source netlist top cell name in the Top Cell text box.

Output Specifications for Calibre xACT

Use the Outputs pane to specify the type of extraction output that is generated.

In the Outputs pane, the PEX Netlist field requires an output netlist filename be specified. The remaining fields are optional with required options set by default.

Table 3-3. Fields in Outputs Pane

Field	Purpose	Required
None	Specify extraction mode. See "Specifying Extraction Mode".	Yes
Extraction Type	Specify extraction type. See "Specifying Extraction Type".	Yes
PEX XACT Corner | Corners	Specify multiple corner extraction using either the PEX XACT Corner statement or restrict the multi-corner run to specific process corners.	No
PEX Netlist	Specify the extracted netlist. See "Specifying the Parasitic Netlist".	Yes
Mask SVDB	Add information types to the SVDB. See "Specifying Mask SVDB".	No
Reports	Specify PEX Report to create a report of parasitic results for the extracted circuit.	No
PEX Report Point2Point	Specify to create a point-to-point report.	No

Specifying Extraction Mode

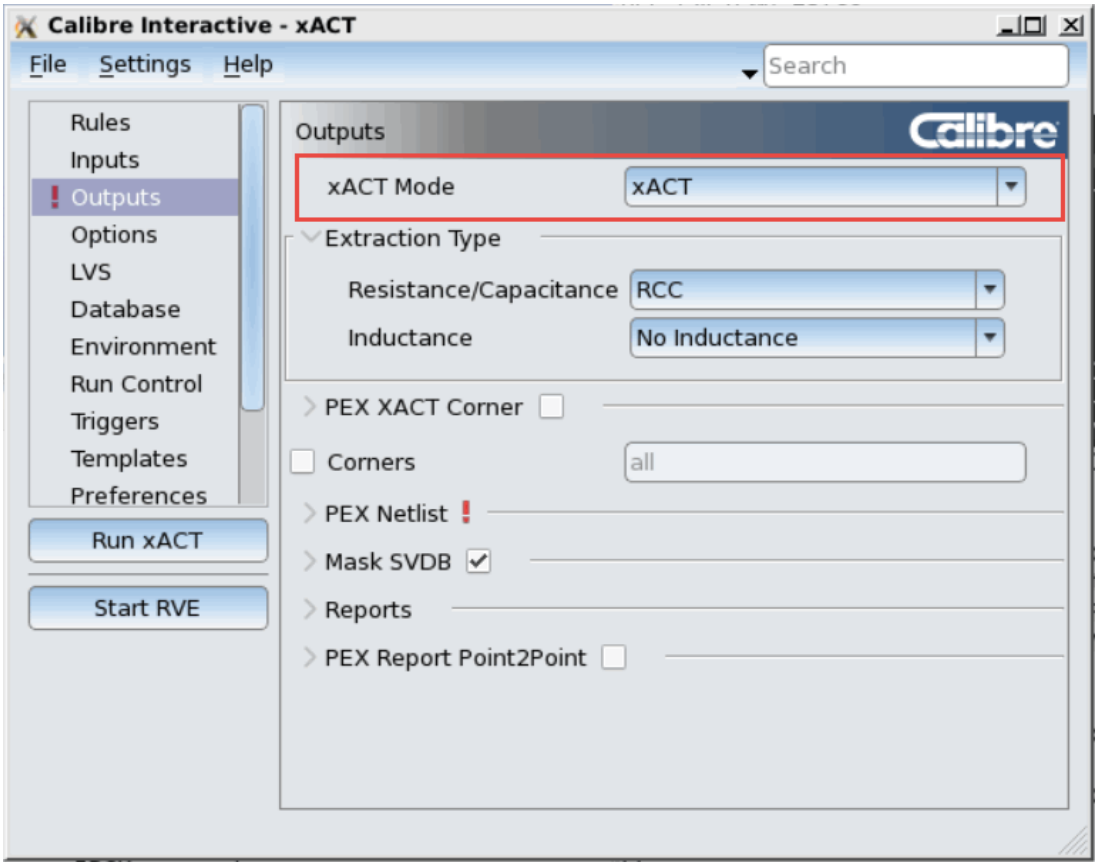
In Calibre Interactive - xACT the default extraction mode is set to xACT. Use xACT Mode to change the extraction mode.

Prerequisites

- The Inputs pane fields have been set. A red exclamation point displayed next to Inputs indicates that required settings are specified incorrectly or missing.

Procedure

1. Click **Outputs** to display the Outputs pane.



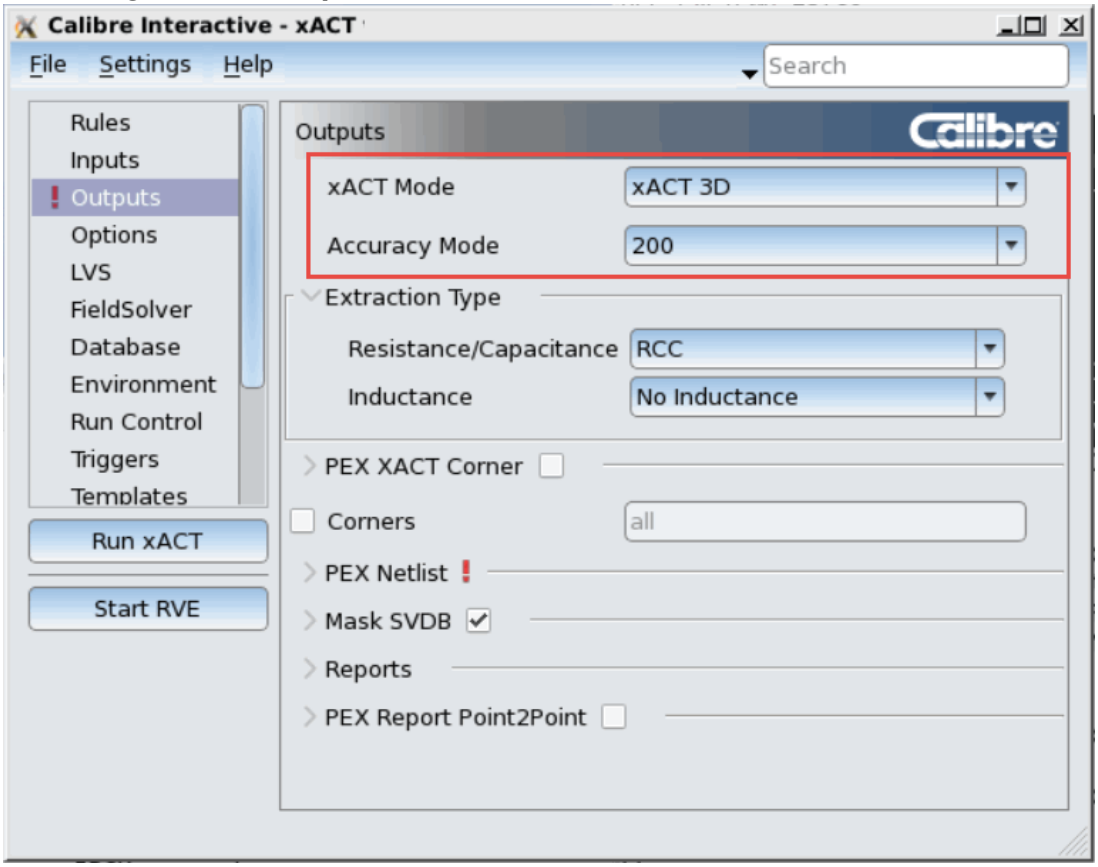
2. Select the xACT Mode. Click the dropdown list arrow to display the list of available xACT Mode options. Choices are:

   - **xACT —** Run Calibre xACT.
   - **xACT 3D —** Run Calibre xACT 3D. This choice invokes the 3D field solver for parasitic capacitance extraction and analysis.
   - **xACT 3D Select —** Run Calibre xACT 3D select net extraction using the 3D fieldsolver for parasitic capacitance on specific nets.

   Select xACT 3D.

   The xACT 3D and xACT 3D Select modes display the Accuracy Mode dropdown list. Choices for accuracy mode are 200 or 600. Setting the Accuracy Mode to 600 is more accurate, but will significantly increase runtime. The default field solver accuracy mode is 200.

Figure 3-2. Outputs Pane for xACT 3D xACT Mode Selection

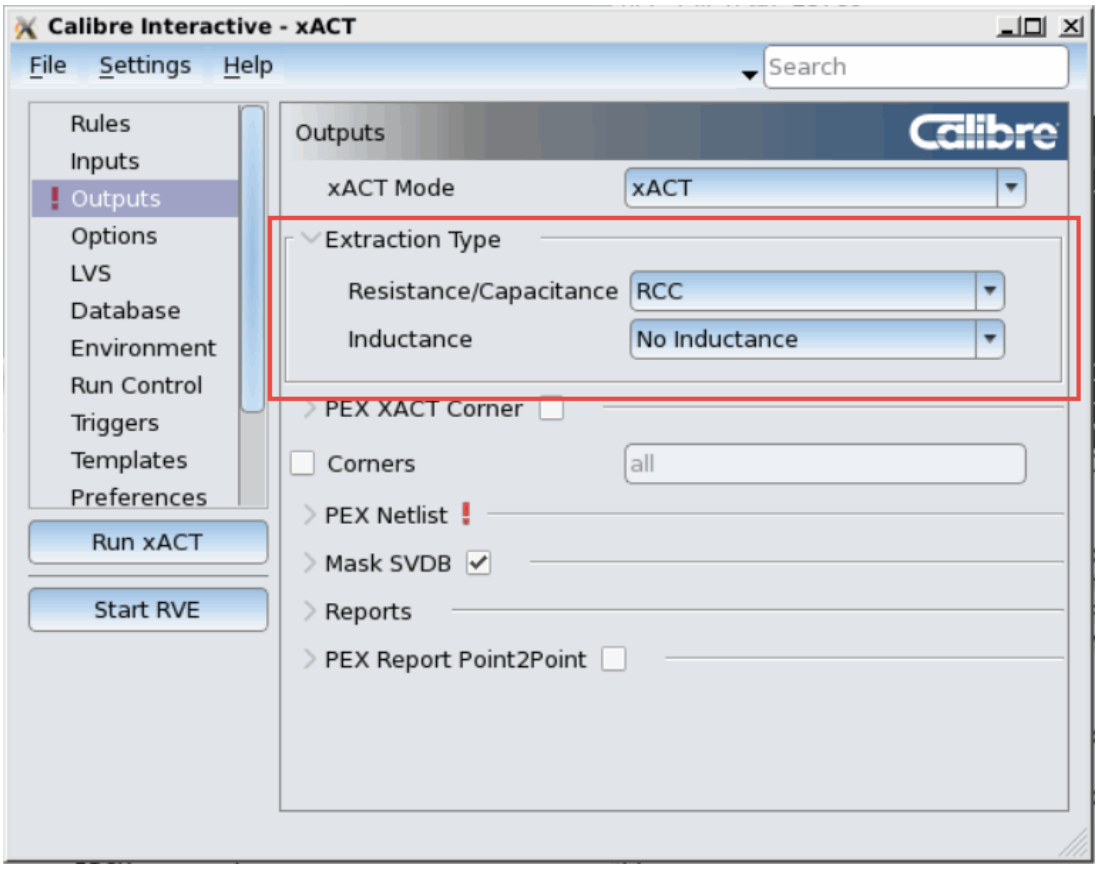


Specifying Extraction Type

Specify the type of extraction you would like to perform.

Procedure

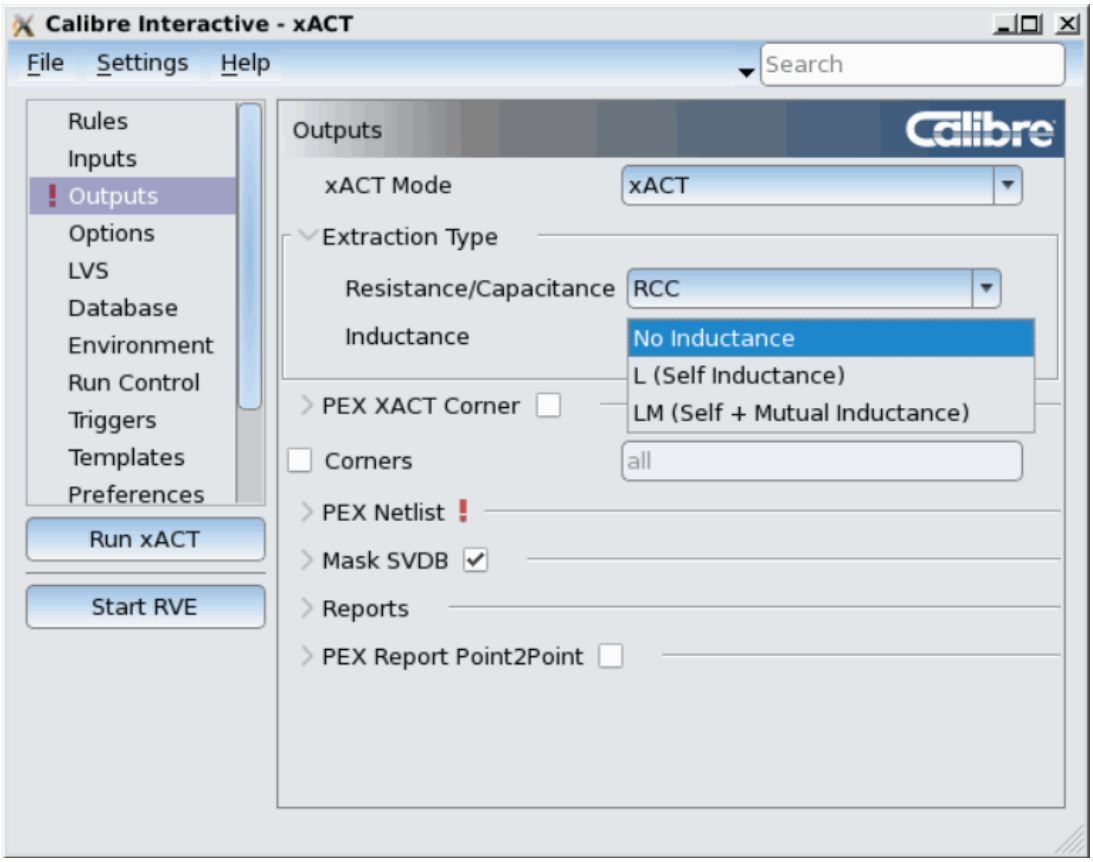
1. Click **Outputs**. The Extraction Type field should be expanded with dropdown list choices for Resistance/Capacitance and Inductance.



2. Select the desired Resistance/Capacitance extraction type from the dropdown list. Your choices are combinations of R (resistance), C (intrinsic capacitance), and CC (coupled capacitance). Valid choices for use with the Calibre xACT tool are R, C + CC, RC, RCC, and No R/C.

   The default extraction type is RCC, which extracts netlists with a distributed RCC network including resistors, coupling capacitors, and intrinsic capacitors.

3. Use the Inductance option to extract self-inductance and mutual-inductance parasitics. This requires an additional Calibre xL license to run. Inductance extraction is covered in detail in the *Calibre xL User's Manual*.



Choices for the Inductance dropdown list are:

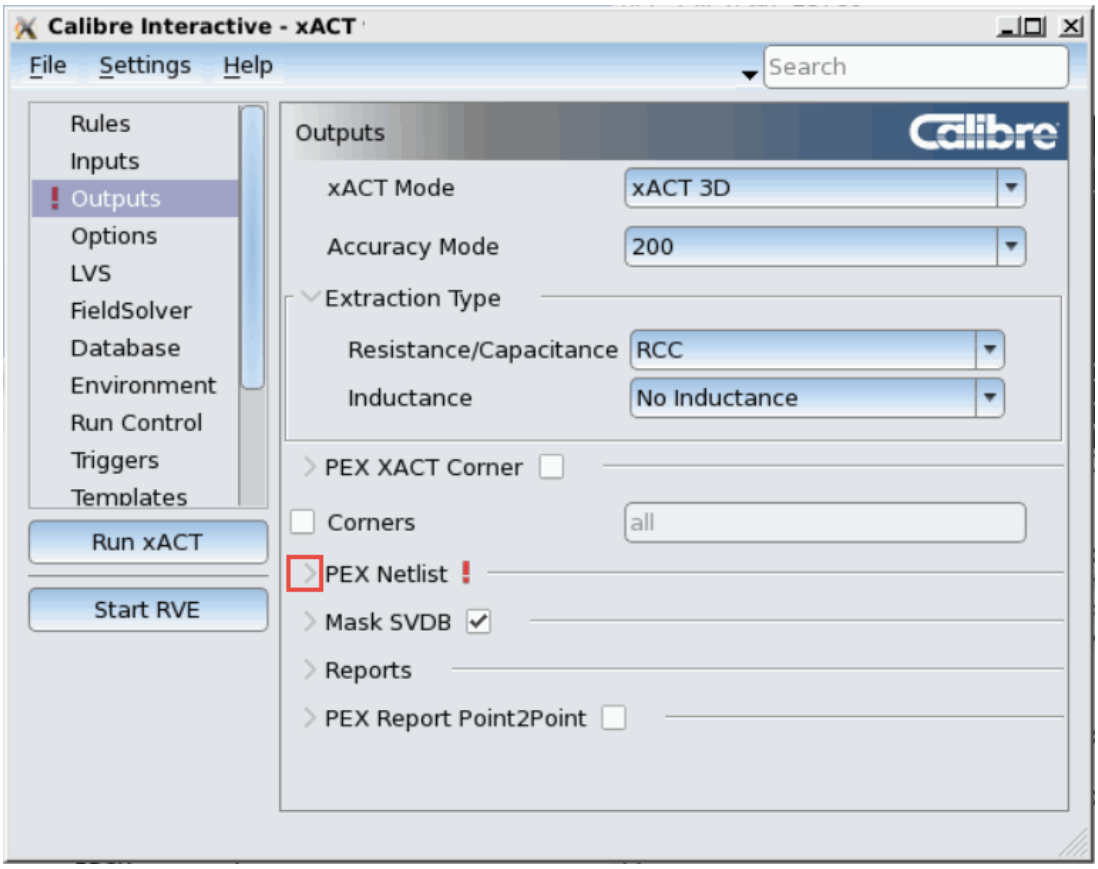
- **No Inductance —** No inductance is extracted. This is the default.
- **L (Self Inductance) —** Extracts a distributed network of inductors.
- **LM (Self + Mutual Inductance) —** Extracts a distributed network of inductors along with mutual inductors.

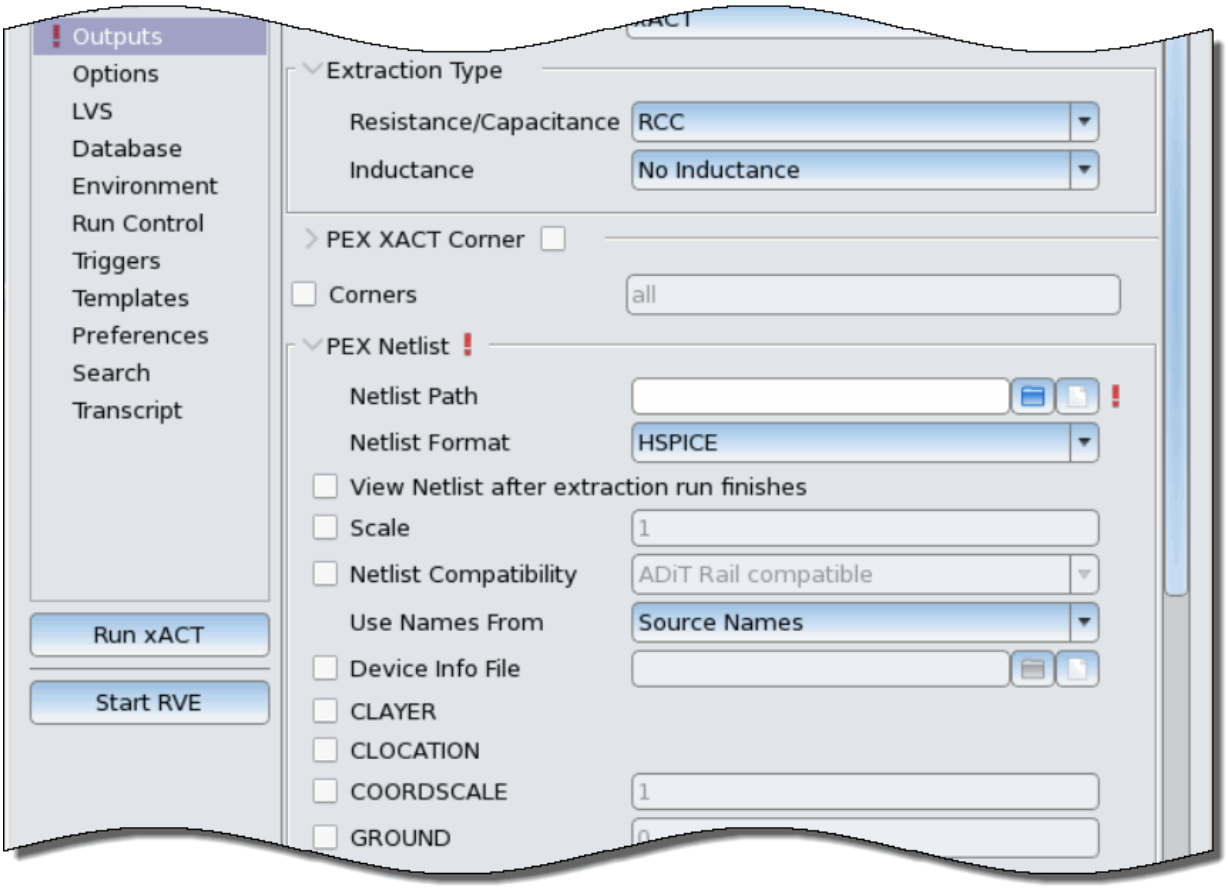
Specifying the Parasitic Netlist

Specify the PEX netlist output format, options, and filename for the parasitic netlist.

Procedure

1. Click the > symbol just to the left of PEX Netlist to display the contents of the PEX Netlist field.

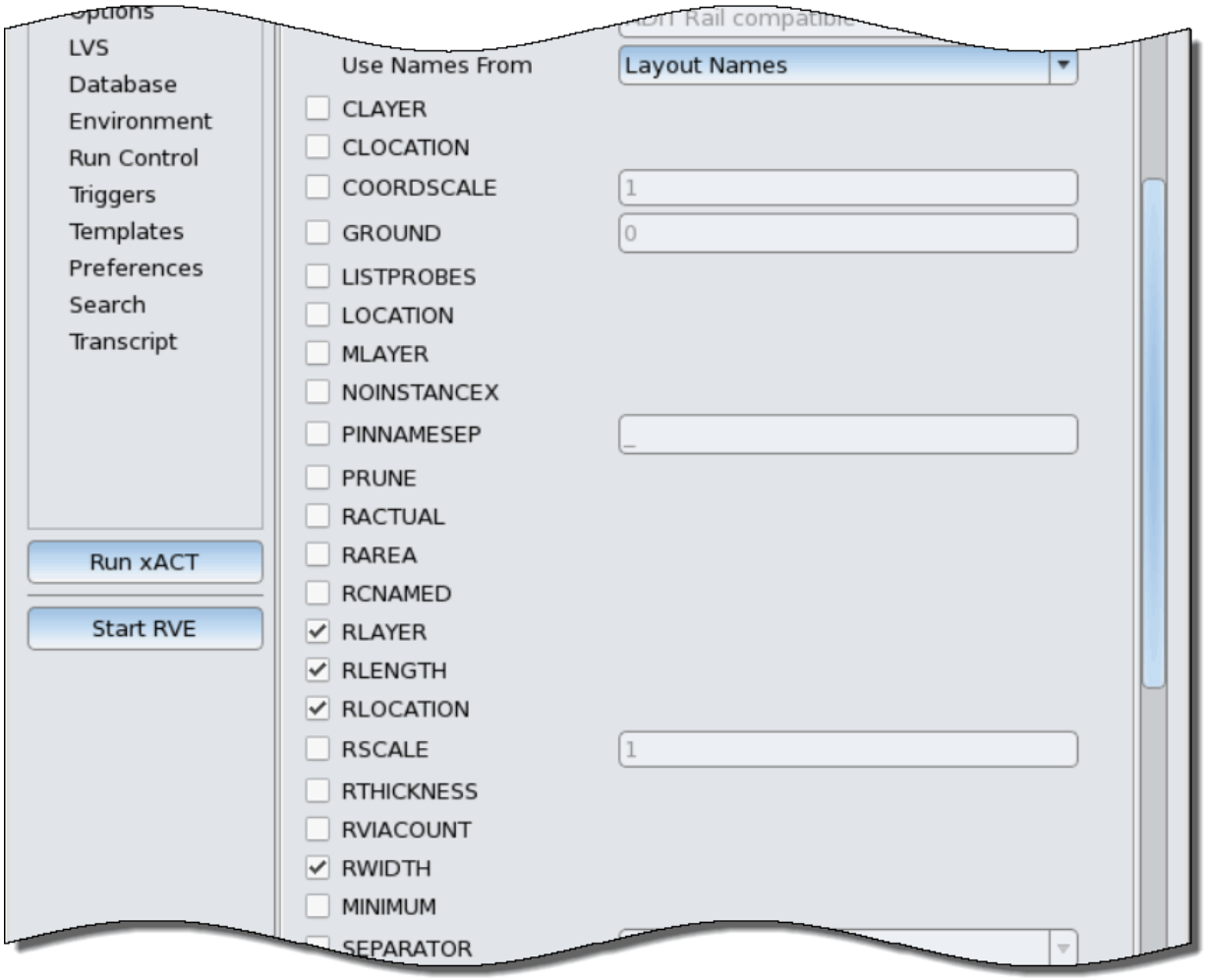




2. Enter the PEX netlist filename in the Netlist Path entry box.
3. Select the output format DSPF from the Netlist Format dropdown list.

   The PEX Netlist field options list changes based on the Netlist Format that is selected. Only options that can be used for the selected netlist format are displayed. You may need to scroll to see all the choices. Options that require additional input are disabled unless selected. In some cases a default value is shown.
4. Select Layout Names from the Use Names From dropdown list.

5. Select check boxes next to options RLAYER, RLENGTH, RLOCATION, and RWIDTH.

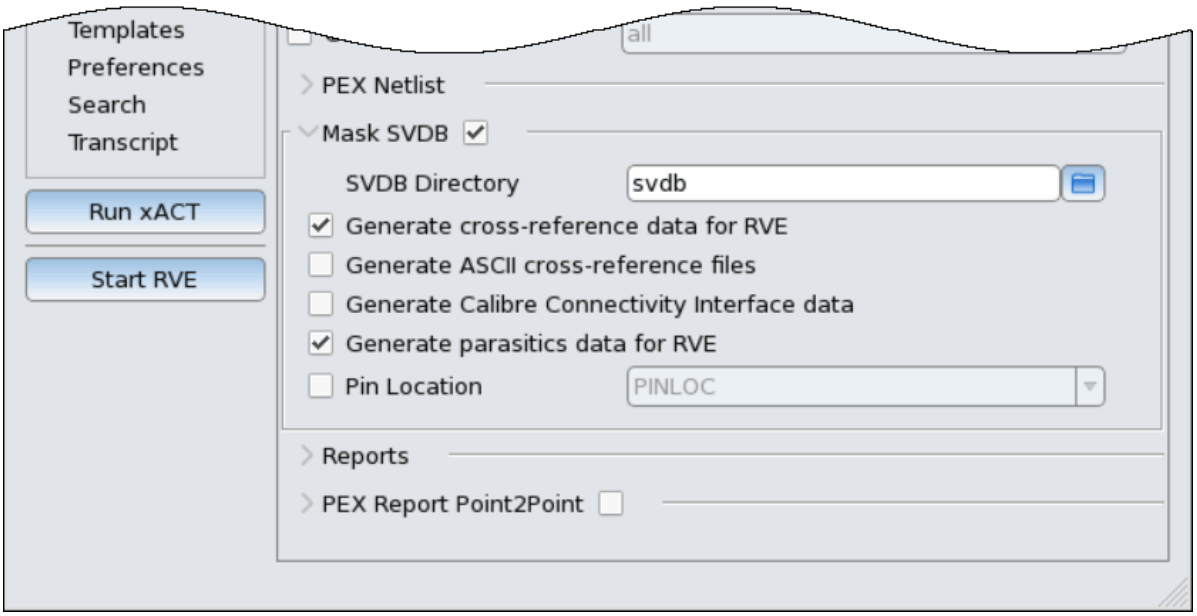


Specifying Mask SVDB

Use Mask SVDB to create the Standard Database Verification Directory (SVDB). The SVDB directory stores files and directories created by Calibre xACT and Calibre xACT 3D, and Calibre nmLVS/Calibre nmLVS-H for use by Calibre RVE. This option is enabled by default.

Procedure

1. Click the > symbol just to the left of Mask SVDB to display the contents of the Mask SVDB field.



   The SVDB Directory name is svdb by default.

2. Select Generate parasitics data for RVE. This option creates the parasitic database (PDB) that can be used with the Calibre RVE viewer.

Running Calibre xACT

After you have selected the inputs and the outputs, you are now ready to perform extraction.

Procedure

1. Click the Run xACT button.

   If any required fields have not been set a dialog displays with a message describing the settings that are missing.

   Tip

   A red exclamation point is displayed next to any required settings that are specified incorrectly or missing. Make sure there are no red exclamation points.

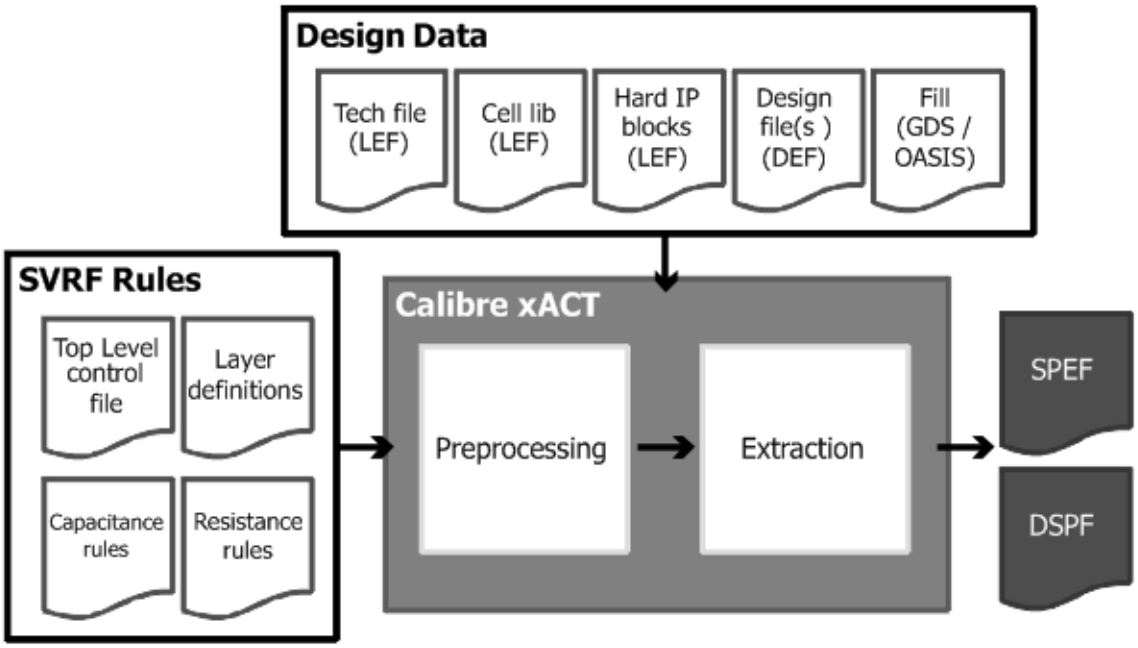
2. After the parasitic netlist is created, you can use Calibre RVE to highlight parasitic elements in the layout viewer.

Chapter 4
Getting Started: Calibre xACT Parasitic Extraction for Digital Designs

The Calibre xACT digital extraction flow generates either SPEF or DSPF file output.

This chapter describes the setup and execution of the Calibre xACT tool when extracting digital designs.

Figure 4-1. Calibre xACT Digital Extraction Flow



Digital Design Extraction Flow Setup 51
SVRF Statements Supported By the Digital Extraction Flow 53
Performing Digital Extraction With Calibre xACT 54

Digital Design Extraction Flow Setup

Running the Calibre xACT extraction tool in a digital flow requires certain conditions be met.

These instructions assume that everything is correctly set up and configured:

- Calibre and Calibre xACT software installed.

- Calibre xACT license available.

  See "Licensing: Parasitic Extraction Products" in the *Calibre Administrator's Guide* for details.

- Either the $MGC_HOME or $CALIBRE_HOME environment variables defined.

  See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

- A top-level SVRF rule file containing information used to control the extraction, such as the name of the SPEF file (using PEX Netlist SPEF), or other optional settings such as temperature. For a complete list SVRF statements supported by Calibre xACT, see "SVRF Statements Supported By the Digital Extraction Flow".

- Calibre xACT automatically excludes power and ground nets when using LEF and DEF as input. It identifies them with the keywords USE POWER and USE GROUND in the DEF file.

- A layer definitions file with layers, connectivity, and parasitic calculations specified. The Layer and Connect statements are typically found in the Calibre LVS rule file available for download from the foundry web site. You may also create your own SVRF layer definitions based on a list of layer names and connection definitions.

- A capacitance rule file (*rules.C*). Calibre xRC, Calibre xACT 3D, and Calibre xACT use the same capacitance rule file. Include the capacitance rule file in your top-level rule file using the Include statement.

- A resistance rule file (*rules.R*). Calibre xRC, Calibre xACT 3D, and Calibre xACT use the same resistance rule file. Include the resistance rule file in your top-level rule file using the Include statement.

- A Calibre xACT rule file (*rules.xact*) created with xCalibrate.

  Note

  The Calibre xACT digital extraction flow requires the *rules.xact* file only if it is provided in your foundry's PDK.

  Caution

  SVRF rule files that contain deprecated or obsolete SVRF statements are not guaranteed to work properly with Calibre xACT.

- The following design files:

  - A technology LEF file (TLEF)
  - A LEF cell library
  - LEF files for macros
  - One or more DEF files containing the design data

- GDS or OASIS fill data if available, GDS macro data if available

The type of required design files depends on which Calibre xACT digital flow you are using. The supported Calibre xACT digital flows are:

- Flat LEF/DEF Extraction
- Block-by-block Hierarchical Extraction
- Full Hierarchical LEF/DEF Extraction
- Extraction with dummy (metal) fill or virtual fill (GDS, OASIS, and DEF formats):
  - GDS and OASIS Metal Fill Extraction
  - DEF Metal Fill Extraction
- Extraction With GDS Macrocells
- Multi-Temperature Extraction
- Multi-Corner Multi-Temperature Extraction

For more information on these flows see "Types of Extraction".

SVRF Statements Supported By the Digital Extraction Flow

The Calibre xACT digital flow utilizes certain PEX SVRF statements.

The following statements are supported:

- PEX LEF Extract Cell Obstructions
- PEX DEF Extract Blockages
- PEX DEF Map
- PEX Density Estimate
- PEX Density Outside
- PEX Density Window
- PEX Extract Exclude
- PEX Extract Include
- PEX Extract Temperature
- PEX Magnify

- PEX Netlist *filename* {SPEF | DSPF}SOURCENAMES [SEPARATOR *sep_name*] — no other PEX Netlist formats or options are supported.
- PEX Netlist Subnode Section
- PEX XACT Corner
- PEX Report Opens
- PEX Report Shorts
- PEX Xcell
- PEX Xcell Extract Mode
- PEX XACT Virtual Fill ... Define
- PEX XACT Virtual Fill ... Use

Other SVRF statements supported by the Calibre xACT digital flow are:

- Capacitance Order
- Layout Path
- Layout Primary
- Layout System
- Layout Case
- Mask SVDB Directory
- Source Case

Performing Digital Extraction With Calibre xACT

Run Calibre xACT digital extraction to create the SPEF netlist.

Prerequisites

You have met the requirements outlined in Digital Design Extraction Flow Setup.

Procedure

1. Create a master directory for your design, *master_dir*.
2. Create a *rules* directory containing the following SVRF parasitic extraction rule files for the target technology and metal scheme:
   - capacitance rule file (*rules.C*)

- resistance rule file (*rules.R*)
- xact rule file (*rules.xact*) — required only if provided in your foundry's PDK.

3. Create a design data directory containing the LEF and DEF files (or links to them) for your design, *digital_design_dir*.
4. Create a rule file called *xact_layers.svrf* and specify DEF layer mapping statements using PEX DEF Map. This statement maps a DEF layer name to the input or original layer name. For example:

```
PEX DEF MAP metal1 metal1
PEX DEF MAP via1 via1
PEX DEF MAP metal2 metal2
PEX DEF MAP via2 via2
PEX DEF MAP metal3 metal3
PEX DEF MAP via3 via3
PEX DEF MAP metal4 metal4
PEX DEF MAP via4 via4
PEX DEF MAP metal5 metal5
PEX DEF MAP via5 via5
PEX DEF MAP metal6 metal6
PEX DEF MAP via6 via6
PEX DEF MAP metal7 metal7
PEX DEF MAP via7 via7
PEX DEF MAP metal8 metal8
PEX DEF MAP via8 via8
PEX DEF MAP metal9 metal9
PEX DEF MAP via9 via9
PEX DEF MAP metal10 metal10
```

5. Create a top-level control file called *xact.svrf*. The following example contains a set of required and recommended statements:

```
// design layout information - LEF/DEF
LAYOUT PRIMARY "top"
LAYOUT PATH "./Design/LEF/tech.lef"
            "./Design/LEF/gscl45nm.lef"
            "./Design/LEF/PADS.lef"
            "./Design/LEF/myram.lef"
            "./Design/LEF/dac6_op2.lef"
            "./Design/LEF/ADC_5bit_sc_5.lef"
            "./Design/DEF/top.def"
LAYOUT SYSTEM LEFDEF
LAYOUT CASE YES

// design netlist information
SOURCE CASE YES

// specify extracted netlist output - SPEF
PEX NETLIST "netlist.spef.gz" SPEF LAYOUTNAMES MAPNAMES NOINSTANCEX
PEX NETLIST CAPACITANCE UNIT fF
PEX NETLIST ESCAPE CHARACTERS OFF
MASK SVDB DIRECTORY "svdb" XACT

// Extraction options
PEX LEF EXTRACT CELL OBSTRUCTIONS YES
PEX DEF EXTRACT BLOCKAGES NO
PEX XACT FILL FLOATING
PEX EXTRACT TEMPERATURE 27

// Reduction options
PEX REDUCE CC ABSOLUTE 3 RATIO 0.03
PEX REDUCE DIGITAL YES
LAYOUT USE DATABASE PRECISION YES
DFM DATABASE "dfmdb" OVERWRITE

// Comment out the following line if Calibre complains that
// UNIT CAPACITANCE is already defined or change the units if
// the results are incorrect by a factor of 1000
UNIT CAPACITANCE fF

// specify LVS and parasitic extraction rule files
INCLUDE ./Rules/calibre_lvs_rules.svrf
INCLUDE ./Rules/xact_layers.svrf
INCLUDE ./Rules/pex_rules.C
INCLUDE ./Rules/pex_rules.R
INCLUDE ./Rules/pex_rules.xact
```

6. Using the top-level control file, *xact.svrf*, run Calibre xACT to create the SPEF netlist:

   The following example performs an extraction for resistance with distributed capacitance and coupled capacitance between nets using the settings specified in the file *xact.svrf*, and any files it includes:

```
calibre -xact -rcc xact.svrf
```

Results

Calibre xACT notifies you when it has successfully completed. Be sure to check the transcript for any other errors which may invalidate results.

Chapter 5
Getting Started: Calibre xACT 3D Parasitic Extraction Using Batch Mode

The Calibre xACT 3D parasitic extraction tool can generate a netlist in a single step or using a three step PDB batch process.

Figure 5-1 illustrates the Calibre xACT 3D direct netlisting extraction flow.

Figure 5-1. Calibre xACT 3D Direct Netlisting Extraction

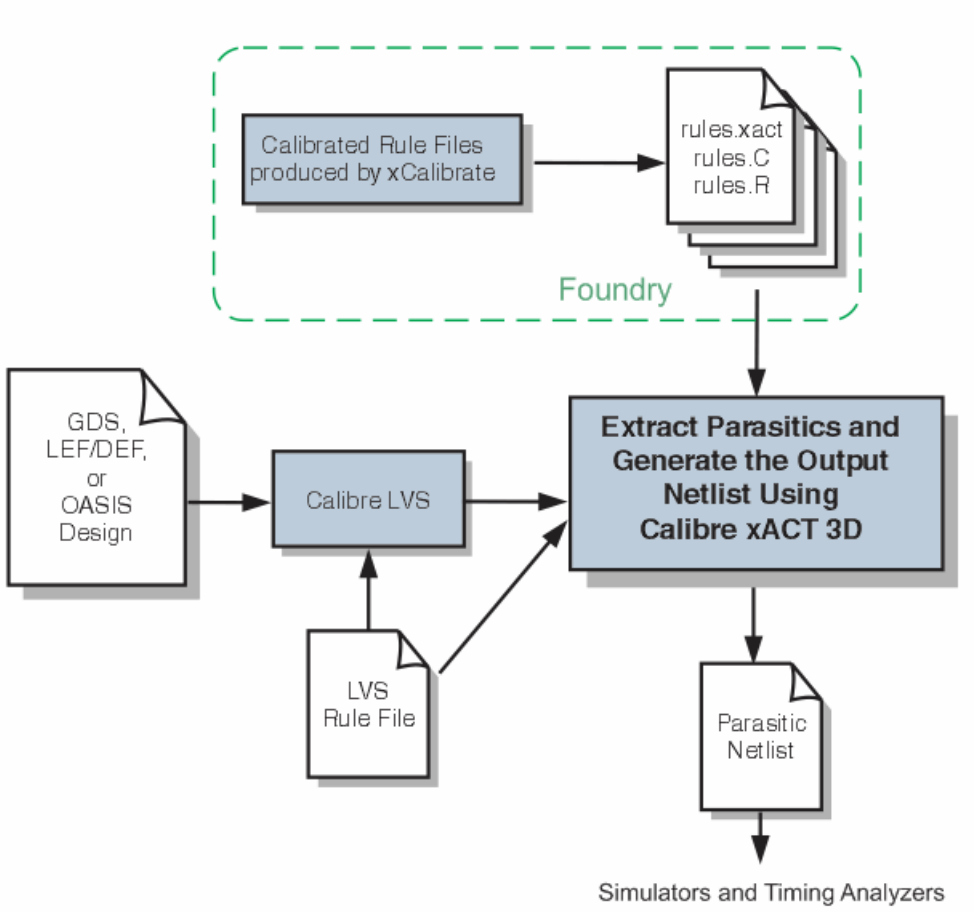
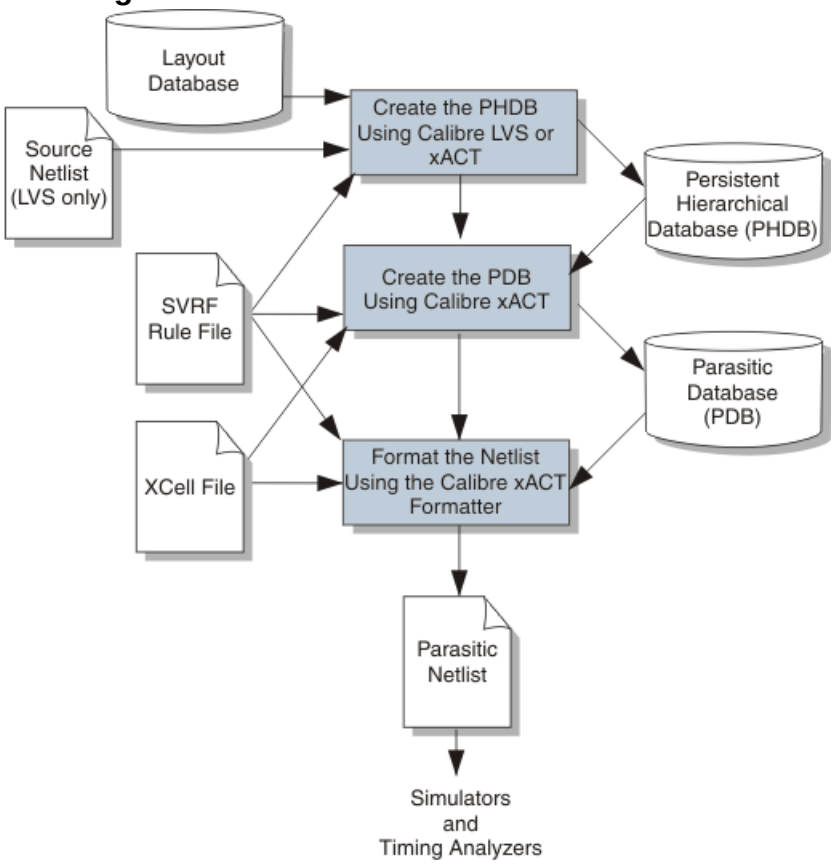


Figure 5-2 illustrates the Calibre xACT 3D PDB extraction flow. Each step has many possible customizations based on your particular needs.

Figure 5-2. Calibre xACT 3D PDB Extraction



Batch Mode Setup for Calibre xACT 3D **60**
Direct Netlisting With Calibre xACT 3D **61**
Using the Calibre xACT 3D PDB Flow **63**
Step 1 — Creating the PHDB 63
Step 2 — Creating the Parasitic Database 64
Step 3 — Generating A Netlist or Report 67

Batch Mode Setup for Calibre xACT 3D

Certain conditions must be met before running the Calibre xACT 3D tool.

These instructions assume that everything is correctly set up and configured:

- Calibre software is installed.
- Calibre xACT 3D license is available.
- Either the $MGC_HOME or $CALIBRE_HOME environment variables are defined.

  See “Setting the CALIBRE_HOME Environment Variable” in the *Calibre Administrator’s Guide* for details.

- A capacitance rule file. Calibre xRC, Calibre xACT 3D, and Calibre xACT use the same capacitance rule file.
- A resistance rule file. Calibre xRC, Calibre xACT 3D, and Calibre xACT use the same resistance rule file.

  For the direct netlisting extraction flow, use calibrated rule files generated with the 2014.2 or newer versions of xCalibrate.

  For the PDB extraction flow, use calibrated rule files generated with the 2008.4 or newer versions of xCalibrate. As of the 2012.3 release, using calibrations and rule files generated with previous versions will issue the following warning:

  ```
  WARNING: Calibre xACT 3D does not support pre-2008.4 release
  calibrations. Extraction results may be unexpected.
  ```

- An SVRF file with layers, connectivity, devices, and parasitic calculations is specified.

Caution

SVRF rule files that contain deprecated or obsolete SVRF statements are not guaranteed to work properly with Calibre xACT.

Tip

Shell scripts are an excellent way to run Calibre from the command line. A script can explicitly set environment variables and record invocation combinations you use frequently.

Direct Netlisting With Calibre xACT 3D

Use the Calibre xACT 3D direct netlisting flow to perform extraction and generate a netlist in a single step.

Prerequisites

- You have met the requirements outlined in Batch Mode Setup for Calibre xACT 3D.
- You have created a top-level control file. See Creating a Top-level Control File for an example. In this example, to perform extraction with layout names, make sure the LAYOUTNAMES keyword is specified for the PEX Netlist and PEX Extract Exclude statements.

Procedure

1. Verify the top-level control file contains the following statements to specify the “inputs”, a layout database:
   - Layout System
   - Layout Path

- Layout Primary

The layout database, also called the design, is a physical representation of your IC. The Calibre xACT software can read GDS, OASIS, and LEF/DEF formats.

2. Determine which parasitics you need. The choices are:

   - **Resistance (-r)** — Exclusively writes resistance models into the netlist.
   - **Lumped Capacitance (-c)** — Exclusively writes capacitance models into the netlist.
   - **Resistance and Distributed Capacitance (-rc)** — Specifies distributed RC parasitic extraction and writes the distributed RC models into the netlist.
   - **Resistance with Distributed Capacitance and Coupled Capacitance between nets (-rcc)** — Specifies R-coupled-C extraction and writes fully-coupled models in to the netlist. This is the default if no option is specified.

   There is a trade-off between the amount of detail and how long the netlist takes to simulate. For example, a netlist with parasitics as lumped capacitance (-c) takes less time to simulate than one with coupled capacitance (-rcc) between nets, which takes less time than one with coupled capacitance between nets including floating nets.

3. Determine the output you need.

   - Netlists are useful for simulation. The format you need depends on your simulator. Only one format of netlist is generated at a time. Calibre xACT supports HSPICE, DSPF, SPEF, SPECTRE, ELDO, and CALIBREVIEW netlist formats.
   - Reports are useful for identifying the nets or cells most affected by parasitics, or for post-processing in spreadsheets or with scripts. Reports can be generated at the same time as netlists.

4. Output the netlist or report(s). For example, using the top-level control file you created, *xact_3d.rules*, run Calibre xACT 3D to create a DSPF netlist:

   ```
   calibre -xact -3d -rcc xact_3d.rules
   ```

   The output depends on what values are set in the top-level control file and what parasitics were specified on the command line.

Results

Calibre xACT ends the transcript with a summary of errors and warnings. Be sure to check for any errors; these invalidate results.

The working directory also contains the requested netlist and reports. For this example, the tool creates a netlist named *netlist.dspf*.

Using the Calibre xACT 3D PDB Flow

Use the Calibre xACT 3D PDB flow to create a parasitic database and format a netlist.

Step 1 — Creating the PHDB 63
Step 2 — Creating the Parasitic Database. 64
Step 3 — Generating A Netlist or Report 67

Step 1 — Creating the PHDB

The Persistent Hierarchical Database, usually referred to as the PHDB, contains information about your design's layout, connectivity, and devices necessary for calculating the parasitic information. You can create the PHDB with Calibre® nmLVS-H™ or Calibre xACT 3D.

Procedure

1. Verify the SVRF rule file contains the following statements to specify the "inputs", a layout database and source netlist.

   a. Layout System, Layout Path, and Layout Primary statements specify the layout database.

      The layout database, also called the design, is a physical representation of your IC. You identify this database in your SVRF rule file, or through the Calibre Interactive Graphical User Interface. The Calibre software can read GDS, OASIS, and LEF/DEF formats.

   b. If you require source or schematic names in the generated netlist, use Source System, Source Path, and Source Primary statements to specify the source netlist.

      If you require source (schematic) names in the generated netlist, you will need the source netlist. This netlist must be in SPICE format and is identified in the SVRF rule file or through Calibre Interactive.

2. Create the PHDB.

   Note

   If your database is in LEF/DEF format, only layout names are supported. The PHDB must be created with calibre -xact -phdb.

   - **To use source names**:

     ```
     calibre -lvs -hier -spice directory_path/filename.sp SVRF_file
     ```

     For the output to be used in extraction, *directory_path* must match that specified in the Mask SVDB Directory statement, and *filename* must be the cell name of the Layout Primary statement.

- **To use layout names**:

  ```
  calibre -xact -phdb SVRF_file
  ```

  The PHDB is only generated once per layout. You do not need to regenerate the PHDB unless your design changes, or you modify the SVRF connectivity rules.

Results

Calibre notifies you when it has successfully completed. Check the transcript for any errors.

After creating the PHDB, these files are created:

- **PHDB** — The PHDB is stored in the Standard Verification Database (SVDB) and used in creating the PDB.
- **LVS Report file (default: *lvs.rep*)** — If you used Calibre® nmLVS™, it writes the results of the LVS run to this ASCII file. The file is not used in parasitic extraction.
- **LVS Extraction file (default: *lvs.rep.ext*)** — If you used Calibre nmLVS, it writes the results of the circuit extraction to this file. The file is not used in parasitic extraction.
- **layout netlist** — This is the SPICE netlist that the Calibre xACT tool uses for input in the next step. This file is named after the top level cell.

Related Topics

calibre -xact -phdb

Step 2 — Creating the Parasitic Database

Step 2 — Creating the Parasitic Database

Once you have created the PHDB, you create the parasitic database (PDB).

The PDB stores the parasitic models for each extracted net. This step can be run multiple times without regenerating the PHDB. You might want to do this if you are extracting different types of parasitics on different nets or if you are also extracting inductance with the Calibre® xL Parasitic Inductance Engine.

There are many decisions that affect how you create the PDB. The most important is speed versus level of detail, which affects accuracy. Parasitic extraction for an entire chip can take from hours to more than a day. (The exact duration depends on the capabilities of the computer on which you run the analysis and the number of nets in the IC design.)

Prerequisites

- **PHDB** created in Step 1 — Creating the PHDB.

  If you are doing multiple runs, you must use the same PHDB each time. The PHDB must be in the *SVDB* directory; you cannot separately specify the location.

- **Xcell file** (non-flat extraction only)

  When performing hierarchical extraction, the hierarchy is specified by means of an xcell file. If you are doing a transistor-level extraction, or working with LEF/DEF layouts, you do not need an xcell file.

- **SVRF rule file**.

  This should be the same file used in Step 1.

Procedure

1. Determine which parasitics you need. The choices are:

   - Resistance (-r)
   - Resistance and distributed capacitance (-rc)
   - Resistance with distributed capacitance and coupled capacitance between nets (-rcc)

   There is a trade-off between the amount of detail and how long the netlist takes to simulate. For example, a netlist with parasitics as lumped capacitance takes less time to simulate than one with coupled capacitance between nets, which takes less time than one with coupled capacitance between nets including floating nets.

2. Determine how much of the design you need to extract.

   The less of the design you perform parasitic extraction for, the faster simulation will run. Your choices are:

   - Transistor-level (“flat”) extraction, which flattens all design hierarchy and extracts parasitics for everything not explicitly excluded. This is the default if no xcell list is added.
   - Gate-level extraction, which ignores portions of the design listed in an xcell file. (The portions are usually devices or standard cells that have been verified separately.) The part of the design being extracted is flattened, just as with transistor-level extraction.
   - Full hierarchical extraction, which extracts each cell listed in an xcell file only once. This method is only recommended for designs with large, symmetrically placed cells such as memory chips.
   - Any of the above, with “select net”. This option extracts only nets explicitly specified. Use this with iterative extraction to handle critical nets: first most of the design is extracted with minimal detail, and then extraction is run again selecting the critical nets with more parasitic detail.

     Note

     If you are performing multiple extractions on the same design into a single PDB, the extractions need to all be the same type (for example, all gate-level).

Transistor-level (“flat”) extraction is the most accurate, but also slow. For large designs, flat extraction may produce netlists that are too large to simulate. However, where accuracy is critical, transistor-level extraction on a limited section of the design provides detailed information.

a. If you decide on gate-level or full hierarchical extraction, construct an xcell list. See “Hierarchy Control with Xcells” on page 149 for details.

b. If you decide on select-net extraction, add a PEX Extract Include statement to your rule file.

c. If you decide to extract a limited section of the design, and you are performing resistance and distributed capacitance (-rc) or resistance with distributed capacitance and coupled capacitance between nets (-rcc) extraction where highest accuracy is important, you can control the accuracy by adding the PEX Fieldsolver Mode statement to your rule file. Accuracy can be set to 200 or 600. The default setting for this statement is 200. Setting the accuracy to 600 is more accurate, but will significantly increase runtime.

3. Run the extraction.

   The PDB step always contains at least the following:

   ```
   calibre -xact -3d -pdb parasitic_switch SVRF_file
   ```

   where *parasitic_switch* is determined in Step 1. For example:

   ```
   calibre -xact -3d -pdb -rc rules.svrf
   ```

   The above example would perform a transistor-level extraction for resistance and distributed capacitance using the settings specified in the file *rules.svrf*, and any files it included. More simple examples are described in “Basic Extraction Methods”.

4. If you need additional detail on parts of the design (for example, getting precise couplings for critical nets), run extraction again using the same files.

   For example:

   ```
   calibre -xact -3d -pdb -rcclm -select rules.svrf
   ```

Table 5-1 shows common options for the PDB creation step. The decisions you made earlier let you choose among the extraction and parasitic options. Only one extraction option and one parasitic option can be specified per run.

Table 5-1. Invocation Line for PDB Step in Calibre xACT 3D PDB Flow

Required	Extraction Options[1]	Parasitic Options[1]	Additional Options	Required
calibre -xact -3d -pdb	no switch (flat)	-r	-noasic	***rule_file***
	-xcell ***xcell_file***	-rc	-select	
	-xcell ***xcell_file*** -full	-rcc		

1. Select only one from this column.

The procedures in "Basic Extraction Methods" give complete command line invocations.

Results

Calibre xACT 3D ends the transcript with a summary of errors and warnings. Be sure to check for any errors; these invalidate results.

This step creates the *PDB* directory in the *SVDB* directory. The PDB stores the parasitic models for each extracted net. These models consist of the net's name and the collection of device pins, ports, parasitic delays, and circuit elements.

Related Topics

Hierarchy Control with Xcells

Step 3 — Generating A Netlist or Report

Step 3 — Generating A Netlist or Report

As the last step, you produce a netlist or report using the Calibre xACT formatter. The netlist can be in any of several formats such as HSPICE or DSPF. You can also set the formatter to perform different types of reductions to produce netlists that are more easily simulated.

Prerequisites

- **PHDB** created in Step 1 — Creating the PHDB.

  The PHDB contains connectivity information.

- **PDB** created in Step 2 — Creating the Parasitic Database.

  The PDB contains the parasitic information.

- **SVRF rule file**.

This should be the same file as used in Steps 1 and 2.

Procedure

1. Determine the output you need.
   - Reports are useful for identifying the nets or cells most affected by parasitics, or for post-processing in spreadsheets or with scripts. Reports can be generated at the same time as netlists.
   - Netlists are useful for simulation. The format (SPICE, DSPF, CalibreView…) your format depends on your simulator. Only one format of netlist is generated at a time.
2. Verify the SVRF file contains the necessary statements for your output.
   - Reports: Any of the following:

Coupled Capacitance	PEX Report Coupling Capacitance
Total Capacitance	PEX Report Netsummary
Resistance	PEX Report Point2Point

   - Netlist: The format is specified in a PEX Netlist statement, depending on the parasitics that were extracted.

Lumped or Distributed	PEX Netlist
No Parasitics	PEX Netlist Simple

3. Output the netlist or report(s).

   The formatter step always contains at least the following:

   ```
   calibre -xact -fmt SVRF_file
   ```

   The output depends on what values are set in the SVRF file and what parasitics were extracted.

   For example, if the PDB contains RC values and the SVRF specifies “PEX Netlist design.spf SPEF PRIMETIME”, Calibre xACT 3D writes out a SPEF netlist suitable for the PrimeTime® static timing analysis program containing distributed resistance and capacitance parasitic models. Any specified reports are also created.

   The formatter provides the following additional options for specifying an output netlist:

Table 5-2. Formatter Options

Option	Result
-c	Writes only capacitance models into netlist, even if PDB contains more information.
-r	Writes only resistance models into the netlist, even if PDB contains more information.

Table 5-2. Formatter Options (cont.)

Option	Result
-rc	Writes distributed resistance and capacitance models into the netlist.
-rcc	Writes distributed resistance and coupled capacitance models into the netlist.
-all	Writes distributed results; does not produce lumped capacitance. This is the default and recommended for most netlists.
-simple	Writes only the intentional netlist, without parasitics. Useful for verifying that the design is being interpreted correctly.

Results

Calibre xACT 3D ends the transcript with a summary of errors and warnings. Be sure to check for any errors; these invalidate results.

The working directory also contains the requested netlist and reports.

Related Topics

calibre -xact -fmt

Controlling Netlisting

Chapter 6
Getting Started: Calibre xACT 3D Parasitic Extraction Using Calibre Interactive

The Calibre® Interactive™ PEX Graphical User Interface (GUI) provides an interface to the Calibre xACT 3D tool.

The following sections describe how to use Calibre Interactive to perform parasitic extraction with Calibre xACT 3D using the PDB flow:

Invoking Calibre Interactive Parasitic Extraction (PEX) **71**

Loading a Runset (Optional) **72**

Specifying Rule File for PEX Run **72**

Input Specification **74**

Defining Input Data Names in the Extracted Netlist 74

Using Schematic Netlist Input in the Extracted Netlist (Optional) 75

Defining H-Cells Input (Gate-Level and Hierarchical Extraction Only) 75

Outputs for a PEX Run **77**

Defining the Extraction Type 77

Specifying the Parasitic Netlist 78

Restricting the Nets (Optional) 79

Setting Up Reports (Optional) 79

Adding to the SVDB (Optional) 80

Running Calibre Interactive PEX **81**

Setting PEX Options **81**

Invoking Calibre Interactive Parasitic Extraction (PEX)

The way in which you start Calibre Interactive depends on your tool environment.

Prerequisites

Environment correctly set up and configured:

- Calibre software installed and optionally integrated with layout editor.
- Calibre Interactive and Calibre xACT 3D licenses are available. See "Licensing: Physical Verification Products" and "Licensing: Parasitic Extraction Products" in the *Calibre Administrator's Guide* for details.

- Either the $MGC_HOME or $CALIBRE_HOME environment variables defined. Calibre tools require that the CALIBRE_HOME environment variable be set. See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

Procedure

Select the appropriate method:

Table 6-1. Invocation Methods for Calibre Interactive PEX

From…	Use…
Cadence Virtuoso or other layout editor	Select **Calibre > Run PEX** from the menu
Calibre DESIGNrev	Select **Verification > Run PEX...**
Command line	Type: calibre -gui -pex

Loading a Runset (Optional)

After you invoke Calibre Interactive PEX, you are prompted to specify a runset. A runset sets default values and can be useful for managing your different types of extractions.

Procedure

1. Load a runset with the file browser by clicking the Browse (**…**) button.
2. After navigating to the runset, select it and click **Open**. In the main dialog, click **OK**.
3. To skip loading a runset, click **Cancel**.

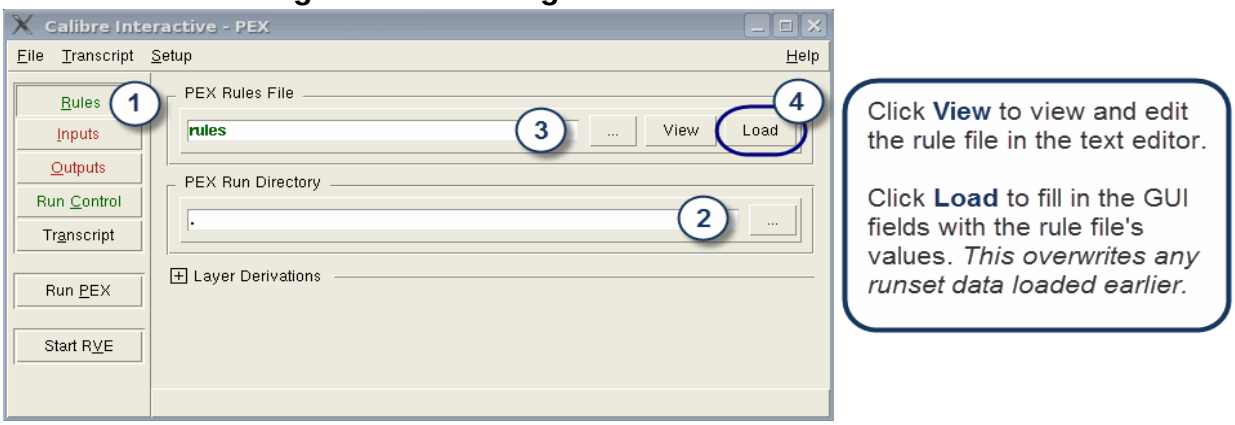
Specifying Rule File for PEX Run

Parasitic extraction requires a pre-defined rules file. Use this procedure to specify and load a rule file in Calibre Interactive.

Procedure

1. Click **Rules**.
2. Specify the run directory name. You can use the Browse (**…**) button to select the run directory name from a list.
3. Specify the rule file name. You can use the Browse button to select the rule filename from a list. You can use the **View** button to view or edit the rule file.
4. Click **Load**. This loads GUI fields and sets GUI options based on rule file data.

Figure 6-1. Loading Rules in Calibre Interactive



Tip

After you load a rule file, any information you specify in the GUI supersedes information in your loaded rule file.

Input Specification

The source of your layout data varies depending on how you invoked Calibre Interactive and the type of extraction you plan to run.

Table 6-2. Tabs in Inputs Pane

Tab	Purpose	Required
Layout	Specify design database and format. (See "Defining Input Data Names in the Extracted Netlist".)	Yes
Netlist	Specify *schematic* netlist or source files, used when output should have schematic's information. (See Using Schematic Netlist Input in the Extracted Netlist (Optional).)	No
H-Cells	Specify cell lists used for performing a non-flat extraction. (See Defining H-Cells Input (Gate-Level and Hierarchical Extraction Only).)	No
Probes	Specify probe points.	No

Defining Input Data Names in the Extracted Netlist

You must specify the design database and format on the **Layout** tab for the Calibre Interactive PEX run.

Procedure

1. Click **Inputs**.
2. Select **Layout** tab.
3. Specify the layout filename. The name appears in red text if the layout file does not yet exist.

   If you invoked Calibre Interactive from a layout editor, you can use the current layout by selecting the Export from layout viewer option. (This will save a copy of the layout in the filename you specify. If the filename already exists, the contents will be overwritten.)

   If there are multiple files for the layout:

   a. Enable the **PEX Options** pane by clicking **Setup > PEX Options**.

   b. Click the **Database** tab.

   c. Click the **Library** tab and use the **Add**, **Delete** and **Delete All** buttons to edit the list of additional layout files you would like to be available during your PEX run. Click **Inputs** to continue specifying input data.

4. Select a format type using the Format dropdown list.
5. Specify the layout Top Cell name.

Using Schematic Netlist Input in the Extracted Netlist (Optional)

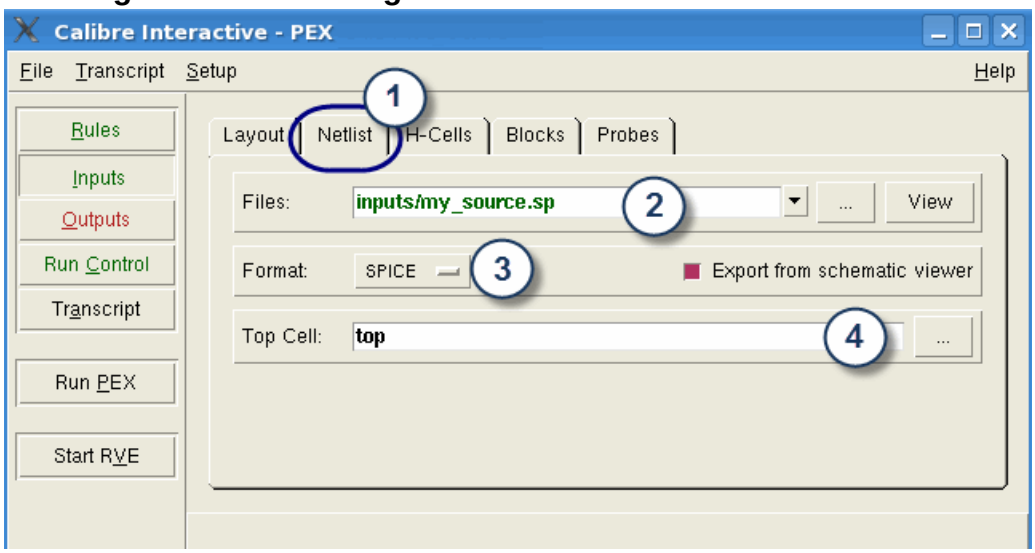
You can specify to use source names instead of layout names in the extracted netlist.

Procedure

1. Select the **Netlist** tab.
2. Specify the source netlist filename in the Files text box.

   If you are working with a schematic viewer, you can generate a new copy of a SPICE or Verilog netlist by choosing the Export from schematic viewer option. The schematic viewer must be running and the schematic data must be loaded in the schematic viewer's edit window.
3. Select the netlist file format from the Format dropdown list.
4. Specify the source netlist Top Cell name in the text box.

Figure 6-2. Providing Source Netlist to Calibre Interactive



Defining H-Cells Input (Gate-Level and Hierarchical Extraction Only)

Specify Hcell files using Calibre xACT 3D with Calibre Interactive.

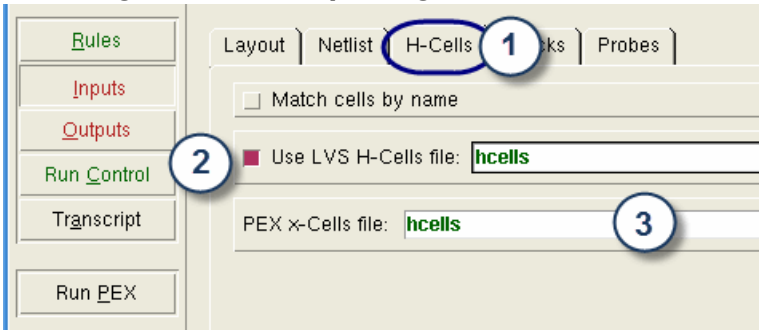
Procedure

Note

You do not need to perform these steps for flat extraction.

1. Select the **H-Cells** tab.
2. Select **Use LVS H-Cells file** and specify the hcell filename (not required if you specify the use of layout names in the pex netlist). The Calibre® nmLVS™ tool uses the hcell file. The Calibre xACT tool uses the xcell file. For more information on the xcell file, see "Hierarchy Control with Xcells".
3. Specify the xcell filename in the **PEX x-Cells file** text box.

Figure 6-3. Completing the H-Cells Tab



Outputs for a PEX Run

Use the Outputs pane to specify the type of extraction.

Table 6-3 lists the procedures associated with each tab in the Outputs pane for Calibre Interactive.

Table 6-3. Tabs in Outputs Pane

Tab	Purpose	Required
(no tab)	Specify extraction mode and extraction type. (See "Defining the Extraction Type".)	Yes
Netlist	Specify the type of netlist. (See "Specifying the Parasitic Netlist".)	Yes
Nets	Restrict the extraction to only some nets. (See "Restricting the Nets (Optional)".)	No
Reports	Set up LVS and xACT reports. (See "Setting Up Reports (Optional)".)	No
SVDB	Add information types to the SVDB. (See "Adding to the SVDB (Optional)".)	No

Defining the Extraction Type

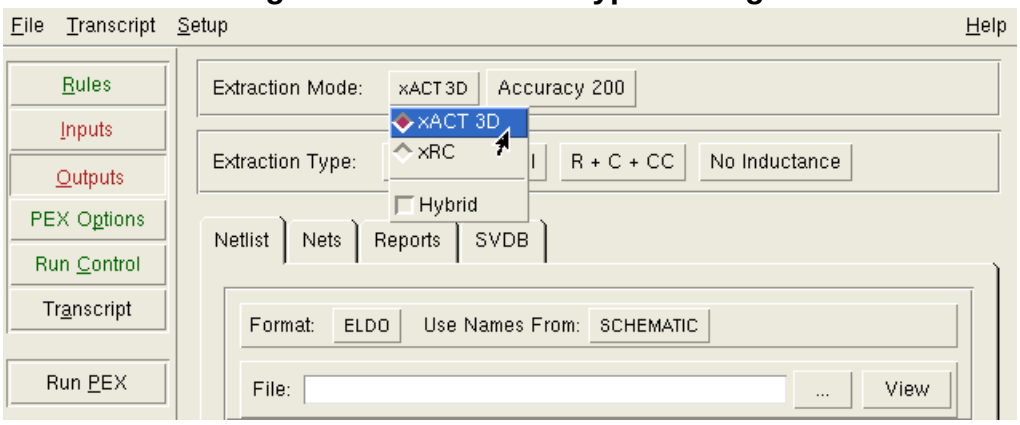
Specify the type of extraction you would like to perform.

Procedure

1. Click **Outputs**.
2. Select the **Extraction Mode** (xACT 3D or xRC).

   For xACT 3D, choose the **Accuracy** (200 or 600). The default is 200. Setting the accuracy to 600 is more accurate, but will significantly increase runtime.

Figure 6-4. Extraction Type Settings



3. Select the extraction level (Transistor Level, Gate Level, or Hierarchical) from the first button on the Extraction Type: line.

   Transistor Level is also known as “flat” extraction. Any cell placements are flattened into the top cell.

   Gate Level extracts parasitics for geometries within the top cell, down to the boundary of the xcells. Xcells are specified in the file provided to the **Inputs > H-Cells** tab.

   Hierarchical extracts parasitics for each identified xcell (not each cell placement) and the top cell. All geometries have parasitics extracted.

   See the “Types of Extraction” chapter for more details.

4. Select the desired extraction type from the second button on the Extraction Type line. Your choices are combinations of R (resistance), C (intrinsic capacitance), and CC (coupled capacitance). Valid choices for use with the Calibre xACT tool are R + C + CC, R + C, R, and No R/C.

   R + C extracts coupled capacitance between nets but represents the value by adding it to the intrinsic capacitance.

5. Select the Inductance option to extract self-inductance and mutual-inductance parasitics. This requires an additional Calibre xL license to run. Inductance extraction is covered in detail in the *Calibre xL User’s Manual*.

   Note

   The second button on the Extraction Type line controls the information that is extracted into the PDB. The **Setup > PEX Options** panel includes a Parasitics to output to RC netlist: field that controls the information from the PDB that is displayed in the netlist. In other words, you can set up your netlist to display a subset of what you have extracted to the PDB.

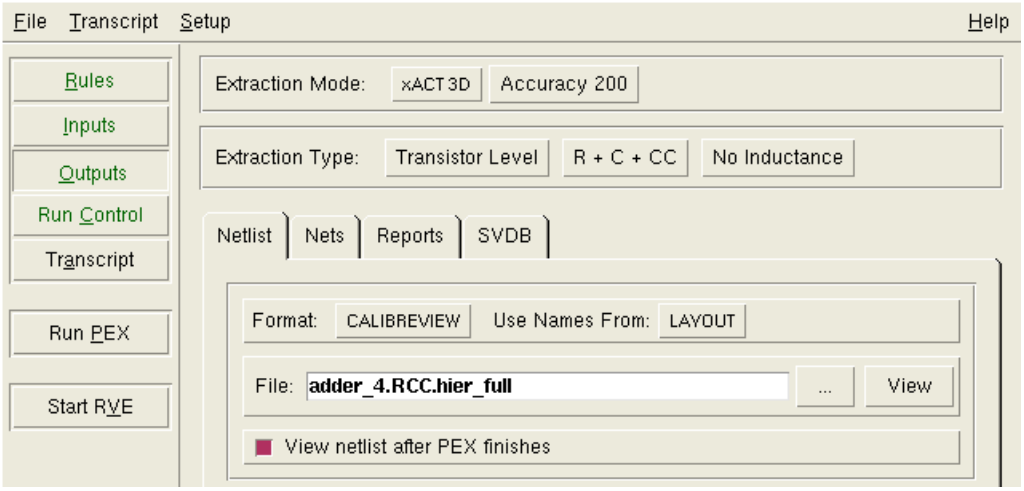
Specifying the Parasitic Netlist

Specify the PEX netlist filename and output format to generate the parasitic netlist.

Procedure

1. Click the **Netlist** tab.
2. Select the output format from the Format dropdown list.
3. Select the source (**SCHEMATIC** or **LAYOUT**) for PEX netlist net and instance names from the Use Names From dropdown list. If you choose SCHEMATIC, you must specify an LVS report name on the **Reports** tab.
4. Enter the PEX netlist filename in the File entry box.

Figure 6-5. Describing the Output Format



Restricting the Nets (Optional)

You can either exclude nets or use only the nets you specify. To decrease netlist size, it is useful to exclude power and ground nets unless they are central to the type of analysis you have planned.

Procedure

1. Click the **Nets** tab.
2. To exclude certain nets, select **Specified Nets** and then **Exclude**. Click the arrow button to browse the schematic viewer and select specific nets, or enter the netlist names manually. The arrow buttons are active only if SCHEMATIC has been chosen in the **Netlist** tab.

Tip

Specify source net names if you chose the SCHEMATIC option for naming nets in the extracted netlist; specify layout net names if you chose LAYOUT naming.

Setting Up Reports (Optional)

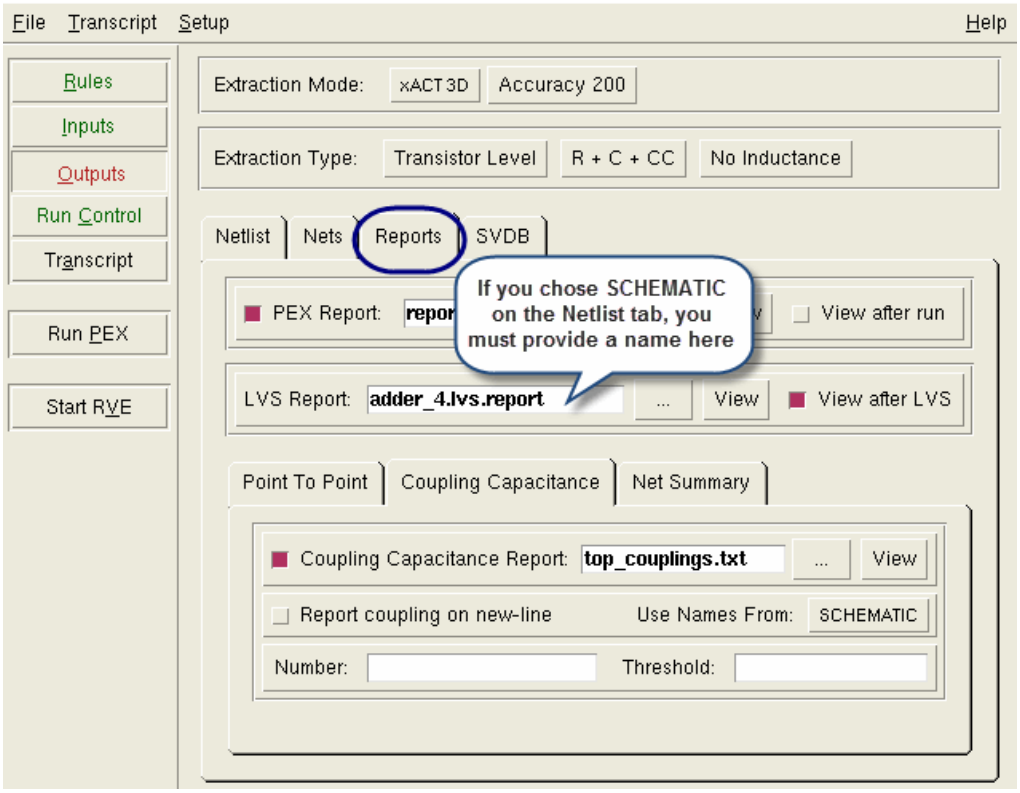
Report files are essential for debugging an extraction run. You can generate a PEX report file from Calibre Interactive.

Procedure

1. Click the **Reports** tab.
2. Specify filenames as appropriate. Select **PEX Report** and specify a report name, if you want to generate a PEX Report file. You must specify an LVS report name if you chose SCHEMATIC on the **Netlist** tab **Use Names From:** dropdown list.

3. There are five reports available:
   - The PEX Report summarizes capacitance from the run.
   - The LVS Report is the same as the one produced as part of a Calibre nmLVS run.
   - The Point to Point Resistance report calculates resistance between two points on the same net. See Reporting Net Resistance from Calibre Interactive.
   - The Coupling Capacitance report summarizes capacitance between pairs of nets along with total capacitance of each net. This quickly shows the net pairs with the most significant coupling. See Reporting Coupled Capacitance from Calibre Interactive.
   - The Net Summary Report generates a report which details the parasitic capacitance values as stored in the PDB.

Figure 6-6. Enabling Report Output



Adding to the SVDB (Optional)

You can specify SVDB output options in Calibre Interactive.

Procedure

1. Click the **SVDB** tab.

2. Specify the SVDB directory name.
3. If you need to start the RVE process on the selected SVDB file after PEX finishes, select the**Start RVE after PEX** option.
4. If you need to generate ASCII cross-reference (XDB) files in the SVDB database, select the **Generate ASCII cross-reference files** option.

   The Generate cross-reference data for RVE option (on by default) generates the binary cross-reference database (XDB) for RVE and the query server.
5. If you need to generate Calibre Connectivity Interface (CCI) data in the SVDB database, select the **Generate Calibre Connectivity Interface data** option. This option requires a CCI license. You may also specify to suppress pin-location information.
6. If you want to incrementally add to the PDB rather than re-generate the PDB every time the PDB generation step is run, select **Generate PDB incrementally**.

Running Calibre Interactive PEX

After you have selected a runfile, and the inputs and outputs, you are ready to perform extraction.

Procedure

1. Click the **Run PEX** button.

   Tip

   All left panel buttons must be green before clicking **Run PEX**. A button changes from red to green when you have specified all required information associated with that button.

2. After the parasitic netlist is created, you can use Calibre RVE to highlight parasitic elements in the layout viewer.

   To run Calibre RVE, the SVDB must have RVE cross-references. This is set in the **Outputs > SVDB** tab, and is on by default. For detailed information on using the Calibre Results Viewing Environment (RVE) for PEX, see "Using Calibre RVE for PEX" in the *Calibre RVE User's Manual*.

Setting PEX Options

You can override Calibre PEX defaults and customize the extracted parasitic netlist by selecting appropriate options which you access via the PEX Options panel.

Use the procedure below to add the PEX options panel to the GUI.

Procedure

1. Select the PEX Options in the **Setup** menu.
2. Select the **PEX Options** button in the left panel to display the options panel. Selecting the **PEX Options** item in the **Setup** menu adds the **PEX Options** button to the left panel button list.
3. Table 6-4 lists the capabilities available in PEX Options.

Table 6-4. Capabilities Available in PEX Options

To…	Set…
	Netlist tab
Convert coupling capacitance into grounded capacitance. (The value of each grounded capacitor equals the value of the original coupling capacitor.)	On the **Reduction and CC** tab select **Ground all coupling capacitors**
Specify a name other than 0 for ground	On the **Format** tab select **Ground node name**
Create multiple ground regions using a layer	On the **Format** tab select **Ground layer name**
Specify the character that separates levels of hierarchy in the netlist (default is /)	On the **Format** tab select **Hierarchy separator**
Include additional parasitic resistor information in the netlist comments	On the **Format** tab select: **Output intentional device …** **Output values for intentional R,C devices** **Output parasitic R**
Reduce RC and RCC netlists without affecting the time domain	On the **Reduction** and **CC** tab: Enable TICER reduction below
Reduce coupled capacitance parasitics	On the **Reduction** and **CC** tab: Enable CC reduction
Specify how floating nets such as metal fill are handled	On the **Format** tab: Extract floating nets
	Misc tab
Match the top-level pins to the order in the test bench	Create top level pin order
Include only some parasitic types in the netlist	Parasitics to output to RC netlist
Generate a driver or receiver file	Generate driver/receiver

4. For additional information on working in Calibre Interactive, see "Using Calibre Interactive to Perform Parasitic Extraction" in the *Calibre Interactive User's Manual.*

Chapter 7
Getting Started: Calibre xACT Inductance Extraction Using Batch Mode

The Calibre xACT inductance extraction flow extracts parasitic frequency-dependent resistance, self and mutual inductance of interconnects in IC layouts.

Inductance can be extracted for transistor-level, analog, and digital designs as a part of any Calibre xACT direct netlisting flow.

Inductance Extraction Flow Setup 83
SVRF Statements Supported by the Inductance Extraction Flow 84
Performing Inductance Extraction With Calibre xACT 85

Inductance Extraction Flow Setup

Certain conditions must be met before running inductance extraction with the Calibre xACT tool.

You must have the following to run the Calibre xACT Inductance extraction:

- Calibre software installed.
- Calibre xACT license available.
- Calibre xL license available. Additional licenses may be required depending on the operations performed. For licensing information, see the "Calibre xL Licensing" section of the *Calibre Administrator's Guide*.
- Either the $MGC_HOME or $CALIBRE_HOME environment variables defined.

  See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.
- An SVRF rule file containing Calibre-specific SVRF statements and operations. This should include the PEX Inductance Frequency statement. If power and ground nets are not defined using LVS Power Name and LVS Ground Name, then PEX Ground and optionally PEX Power must be specified.
- A layout database for which connectivity is defined.
- An optional source netlist if you require schematic names for the nets in the parasitic netlist.

Caution

SVRF rule files that contain deprecated or obsolete SVRF statements are not guaranteed to work properly with Calibre xACT.

Tip

Shell scripts are an excellent way to run Calibre from the command line. A script can explicitly set environment variables and record invocation combinations you use frequently.

SVRF Statements Supported by the Inductance Extraction Flow

The Calibre xACT inductance extraction flow utilizes certain PEX Inductance SVRF statements.

The following PEX Inductance statements are supported:

- PEX Ground
- PEX Power
- PEX Inductance Extract Layers
- PEX Inductance Frequency
- PEX Inductance Minlength
- PEX Inductance Mode
- PEX Inductance Range
- PEX Inductance Returnpath
- PEX Inductance Same Net Mutual
- PEX Inductance Switch Time
- PEX Inductance Victim
- PEX Reduce Minmutual

A warning is generated if the rule file contains PEX Inductance statements not supported by Calibre xACT. For example:

```
WARNING: Statement PEX Inductance Filter is not valid.
```

Performing Inductance Extraction With Calibre xACT

Run Calibre xACT inductance extraction to produce a netlist containing inductance parasitics.

You can perform inductance extraction using source names or layout names. When performing extraction with source names, Calibre LVS must be run for device recognition. Calibre xACT runs the remaining stages of extraction including parasitic calculations and netlisting. When performing extraction with layout names, Calibre xACT runs all stages of extraction including device recognition, parasitic calculations, and netlisting.

This example demonstrates how to perform inductance extraction with Calibre xACT using source names.

Prerequisites

- You have met the requirements outlined in Inductance Extraction Flow Setup.
- You have created the top-level control described in Creating a Top-level Control File.

Procedure

1. Verify the *xact_t.rules* rule file contains the following statements to specify the "inputs", a layout database, source netlist, and inductance specifications.

   a. Layout System, Layout Path, and Layout Primary statements specify the layout database.

      The layout database, also called the design, is a physical representation of your IC. You identify this database in your SVRF rule file. The Calibre xACT software can read GDS, OASIS, and LEF/DEF formats.

   b. Source System, Source Path, and Source Primary statements specify the source netlist.

      If you require source (schematic) names in the generated netlist, you need the source netlist. This netlist must be in SPICE format and is identified in the SVRF rule file. You must also specify the SOURCENAMES keyword for the PEX Netlist statement, as well as for statements PEX Extract Exclude or PEX Extract Include when used.

   c. PEX Inductance Frequency specifies the frequency at which the inductance parasitics are extracted. Parasitic inductance as well as parasitic resistance for some nets may vary with the frequency of extraction. For example, to use the broadband parasitic RLM model for post-extraction transient simulations with the extracted netlists, specify the PEX Inductance MAXIMUM Frequency statement. For details on the broadband model see Broadband Netlist Generation in the *Calibre xL User's Manual*.

d. PEX Ground and PEX Power specify the ground and power nets. Inductance extraction considers power and ground nets as return paths for currents on signal nets based on the specification of PEX Inductance Returnpath statement.

e. PEX Inductance Minlength specifies the minimum length that the signal net paths should exceed in order to be considered for inductance extraction.

2. Determine which inductance parasitics you need. The choices are:

   - **Resistance and Self Inductance (-rl)** — Writes parasitic resistance and self inductance data into the netlist.
   - **Resistance, Self Inductance, and Mutual Inductance (-rlm)** — Writes parasitic resistance, self inductance, and mutual inductance data into the netlist.
   - **Resistance, Intrinsic Capacitance, and Self Inductance (-rcl)** — Writes parasitic resistance, intrinsic capacitance, and self inductance data into the netlist.
   - **Resistance, Intrinsic Capacitance, Self Inductance, and Mutual Inductance (-rclm)** — Writes parasitic resistance, intrinsic capacitance, self inductance, and mutual inductance data into the netlist.
   - **Resistance, Coupled Capacitance, and Self Inductance (-rccl)** — Writes parasitic resistance, coupled capacitance, and self inductance data into the netlist.
   - **Resistance, Coupled Capacitance, Self Inductance, and Mutual Inductance (-rcclm)** — Writes parasitic resistance, coupled capacitance, self inductance, and mutual inductance data into the netlist.

   There is a trade-off between the amount of detail and how long the netlist takes to simulate. For example, a netlist with parasitics as lumped capacitance (-c) takes less time to simulate than one with coupled capacitance (-rcc) between nets, which takes less time than one with coupled capacitance between nets including floating nets.

3. Determine the output you need.

   - Netlists are useful for simulation. The format you need depends on your simulator. Calibre xACT supports HSPICE, DSPF, SPEF, SPECTRE, ELDO, and CALIBREVIEW netlist formats.
   - Reports are useful for identifying the nets or cells most affected by parasitics, or for post-processing in spreadsheets or with scripts. Reports can be generated at the same time as netlists.

4. Use the top-level control file, *xact_t.rules*, to run Calibre LVS and perform device recognition:

```
calibre -lvs -hier xact_t.rules
```

5. Perform parasitic calculations and output the netlist and/or report(s). For example to extract parasitic resistance, capacitance and self inductance use the following command line:

   ```
   calibre -xact -rcl xact_t.rules
   ```

   The output is controlled by rules specified in the top-level control file and parasitics specified on the command line.

Results

Calibre xACT ends the transcript with a summary of errors and warnings. Be sure to check for any errors; these invalidate results.

The working directory contains the requested netlist and reports. The example top-level control file, *xact_t.rules*, creates a netlist named *netlist.dspf*.

Chapter 8
Types of Extraction

The Calibre xACT extraction engine is able to extract interconnect parasitics in a variety of ways. For example, Hierarchical extraction is usually faster and requires less memory; the hierarchical netlists it creates are also smaller than equivalent non-hierarchical netlists and easier to simulate.

This chapter provides descriptions of various types of extraction.

Comparison of Extraction Types 89

Calibre xACT Extraction Types 91

Calibre xACT Flat Transistor-Level Extraction 91
Selective Resistance Extraction 91
Selective Inductance Extraction 92
Multiple Netlist Generation 92
3D Select Extraction 93
Flat LEF/DEF Extraction 93
Block-by-block Hierarchical Extraction 94
Full Hierarchical LEF/DEF Extraction 95
GDS and OASIS Metal Fill Extraction 95
DEF Metal Fill Extraction 97
Extraction With GDS Macrocells 97
Multi-Temperature Extraction 97
Multi-Corner Multi-Temperature Extraction 98
TSV Extraction 98

Calibre xACT 3D Extraction Types 102

Hybrid xACT 3D/Rule-Based Extraction 102
Extraction Using Boundary Conditions 102
Hierarchical Memory Extraction 104
Mixed-Signal Hierarchical Extraction 107
Calibre xACT 3D Gate-Level Extraction 107
Calibre xACT 3D Flat Transistor-Level Extraction 108

Comparison of Extraction Types

A design can be extracted in several different ways. Which one is right for you depends on your design stage and what you intend to do with the netlist.

Depending on your design stage, different extraction types may be more accurate. For example, flat extraction preserves all the interaction between components, but an extraction with xcells allows you to skip pre-characterized cells or devices. Similarly, if some parts of the design are

to be supplied later, a flat extraction provides misleading results compared to a gate-level extraction with xcells for the missing parts.

The other consideration is the size of the produced data and netlist. Large designs may require several gigabytes of memory for the PHDB and PDB. Flat extraction requires more space than any form of hierarchical extraction.

Table 8-1 lists commonly used extraction flows and identifies which Calibre xACT tools support them.

Table 8-1. Tool Support For Extraction Types

Extraction Type	Calibre xACT	Calibre xACT 3D
Gate-Level	Supported	Supported
Flat Transistor-Level	Supported	Supported
3D Select	Supported	Supported
Hybrid xACT 3D/Rule-Based	Not Supported	Supported
Extraction Using Boundary Conditions	Not Supported	Supported
Hierarchical Memory	Not Supported	Supported
Mixed-Signal Hierarchical	Not Supported	Supported
Flat LEF/DEF	Supported	Not Supported
Block-by-Block Hierarchical	Supported	Not Supported
Full Hierarchical LEF/DEF	Supported	Not Supported
GDS and OASIS Fill	Supported	Not Supported
DEF Metal Fill	Supported	Not Supported
GDS Macrocells	Supported	Not Supported
Multi-Temperature	Supported	Not Supported
Multi-Corner Multi-Temperature	Supported	Not Supported
TSV	Supported	Supported

Calibre xACT Extraction Types

Calibre xACT is recommended for digital , block-level, and full-chip GDS based extraction.

This section describes the types of extraction supported by Calibre xACT:

Calibre xACT Flat Transistor-Level Extraction **91**
Selective Resistance Extraction **91**
Selective Inductance Extraction **92**
Multiple Netlist Generation **92**
3D Select Extraction **93**
Flat LEF/DEF Extraction **93**
Block-by-block Hierarchical Extraction **94**
Full Hierarchical LEF/DEF Extraction **95**
GDS and OASIS Metal Fill Extraction **95**
DEF Metal Fill Extraction **97**
Extraction With GDS Macrocells **97**
Multi-Temperature Extraction **97**
Multi-Corner Multi-Temperature Extraction **98**
TSV Extraction **98**

Calibre xACT Flat Transistor-Level Extraction

The Calibre xACT tool extracts parasitic capacitance and resistance for interconnect nets and produces parasitic netlists for post-layout analysis.

Transistor-level extraction flattens the design's interconnect nets into a top-level cell. Calibre xACT extracts the nets in their entirety—from device pin to device pin.

Flat transistor-level extraction is the default.

Related Topics

Creating a Transistor-Level Netlist with Calibre xACT

Selective Resistance Extraction

Extract parasitic resistance on specified layers by performing selective resistance extraction with Calibre xACT.

Calibre xACT uses settings specified in the PEX XACT Control statement to control how certain nets are handled during parasitic extraction. In the EXTRACT_CONTROL section, use

the R_LAYERS_SELECT option to control parasitic resistance extraction on specific layers. Only the layers on a net that are specified in this way have parasitic resistance extracted, while all other layers are extracted with the global parasitic model. The specified layers must be original or derived layers.

Related Topics

Calibre xACT Processing Control

Extracting Net Resistance By Layer

Selective Inductance Extraction

Extract frequency-dependent resistance, self and mutual inductance on specified layers by performing selective inductance extraction with Calibre xACT.

Calibre xACT uses settings specified in the PEX XACT Control statement to control how certain nets are handled during parasitic inductance extraction. In the EXTRACT_CONTROL section, use the L_LAYERS_SELECT option to control parasitic inductance extraction on specific layers. Only the layers of the net that are specified in this way have parasitic inductance extracted. The specified layers must be original or derived layers. Use the L_LAYERS_EXCLUDE option to control which layers of the net are excluded from inductance extraction.

Multiple Netlist Generation

Calibre xACT allows multiple netlists to be created in a single run. Multiple netlists can also be generated with Calibre xACT 3D using the direct netlisting flow.

Calibre xACT uses settings in the PEX XACT Control statement to control netlist generation. The NETLIST_CONTROL directive specifies how Calibre xACT generates netlists and overrides any global net model specifications.

In the NETLIST_CONTROL section, use the NETLIST: directive to specify the netlist name and format. You can also include the NET: directive to control extraction on a specific net. If NET: is not specified then all nets are used. You can specify one or more NETLIST: directives. Specifying multiple NETLIST: directives generates the netlists simultaneously in a single run.

Related Topics

Calibre xACT Processing Control

Generating Multiple Netlists With Calibre xACT

3D Select Extraction

3D select extraction (-3dselect) extracts all nets selected for Calibre xACT and a subset of nets defined for Calibre xACT 3D, and combines the results into a single parasitic netlist. Use this flow for high-accuracy extraction of critical nets with field solver accuracy on all layers.

The command line option -3dselect activates the hybrid xACT/xACT 3D flow. You can specify the nets to be extracted by using either the PEX Extract Include or PEX Extract Exclude statement. If neither statement is specified, then all nets are extracted. The subset of nets to be extracted by Calibre xACT 3D are defined with PEX Fieldsolver Mode NETS statement. The extraction results from both tools are combined and written to the same netlist.

If PEX Fieldsolver Mode statement is specified without NETS or EXCLUDE_NETS keywords and the -3dselect option is specified on the command line, for example:

```
Command line: calibre -xact -3dselect -rcc rules.top
SVRF Statement: PEX FIELDSOLVER MODE 600
```

All layers on all nets are extracted by the field solver with the specified field solver mode of 600.

If PEX Fieldsolver Mode statement is specified with the NETS keyword and the -3dselect option is specified on the command line, for example:

```
Command line: calibre -xact -3dselect -rcc rules.top
SVRF Statement: PEX FIELDSOLVER MODE 600 NETS SOURCENAMES TOPLEVEL "clk"
```

All layers on net “clk” are extracted with field solver mode 600. All other nets are extracted Calibre xACT.

If PEX Fieldsolver Mode statement is specified with the NETS keyword and the -3dselect option is specified on the command line, for example:

```
Command line: calibre -xact -3dselect -rcc rules.top
SVRF Statements: PEX FIELDSOLVER MODE 200 EXCLUDE_NETS "B"
                 PEX FIELDSOLVER MODE 600 "A" "Z"
```

All nets are extracted by the field solver with mode 200 except for net “B”. Net B is extracted with Calibre xACT. Nets “A” and “Z” are extracted with mode 600.

The -3dselect option is not supported with the -3d, -full option, or -3d -pdb (Calibre xACT 3D PDB extraction flow).

Flat LEF/DEF Extraction

Calibre xACT performs flat LEF/DEF extraction in cases where there are no hierarchical sub-blocks in the design. All nets are extracted except for the power and ground nets.

For details on how to run flat extraction using Calibre xACT, see "Performing Digital Extraction With Calibre xACT".

Block-by-block Hierarchical Extraction

Block-by-block hierarchical extraction extracts each hierarchical sub-block individually, then extracts the top level of the design using black boxes for the sub-blocks. In addition to the top-level SVRF control file, you need a separate SVRF control file for each sub-block in the hierarchy.

For block-by-block hierarchical extraction:

- In each of the SVRF control files, set the DEF "DESIGN" name of each block using the Layout Primary statement. For example:

```
LAYOUT PRIMARY "subBlockAlpha"
```

- Use the Layout Path statement to define the name of the LEF technology file plus the name of the LEF files for the leaf cells (cell library plus hard IP blocks), and the name of the DEF file for the sub-block that you are extracting. For example:

```
LAYOUT PATH "myTechFile.lef"
"myCellLibrary.lef"
"subBlockAlpha.def"
```

- Run Calibre xACT using this command file, then change the name of the DEF file to point to the next hierarchical block. Then run it again.
- To extract the top-level DEF file, you need LEF file(s) containing macro definitions of each of the hierarchical sub-blocks in the design. Including LEF macros for the sub-blocks and excluding their DEF designs is equivalent to telling Calibre xACT to black box the hierarchical sub-blocks. Be sure to change the name of the top cell to make sure the tool can find it. For example:

```
LAYOUT PRIMARY "myTopCell"
LAYOUT PATH "myTechFile.lef"
"myCellLibrary.lef"
"subBlockAlpha.lef
"subBlockBravo.lef"
"subBlockCharlie.lef"
"myTopLevel.def"
```

- This flow generates a SPEF file for each sub-block, plus a SPEF file for the top level. Make sure that you rename the SPEF files after each iteration. Otherwise they will overwrite each other.

Full Hierarchical LEF/DEF Extraction

Full hierarchical LEF/DEF extraction reads in the complete design, which includes the top level and all of the hierarchical DEF blocks. It then flattens the design in memory. This gives the most accurate depiction of the hierarchical design because unique instances of the sub-blocks are extracted within the context of the design as a whole. Calibre xACT has sufficient performance and capacity even for extremely large full-chip designs.

Nets inside the sub-blocks that connect to nets at higher levels in the hierarchy, inherit the names of the higher level nets to which they connect. Nets contained completely within the sub-blocks are uniquified by pre-pending the component instance name of the sub-block followed by a forward slash.

For full hierarchical LEF/DEF extraction:

- Ensure that the Layout Path statement in your top-level SVRF control file includes the names of all the DEF files in your hierarchy. The Layout Path statement must also name all of the LEF files that describe the leaf cells (cell library plus any hard IP blocks). The Layout Path statement must list LEF files that describe LEF macros corresponding to the hierarchical sub-blocks in your hierarchy, otherwise the extraction run will terminate with an error. For example:

  ```
  LAYOUT PATH "myTechFile.lef"
  "myCellLibrary.lef"
  "subBlockAlpha.lef"
  "subBlockBravo.lef"
  "subBlockCharlie.lef"
  "subBlockAlpha.def"
  "subBlockBravo.def"
  "subBlockCharlie.def"
  "myTopLevel.def"
  ```

  To black-box hierarchical blocks, remove the corresponding DEF files from the LAYOUT PATH list.

- The Layout Primary statement in your top-level SVRF command file must define the name of the top cell in your design. For example:

  ```
  LAYOUT PRIMARY "myTopCell"
  ```

GDS and OASIS Metal Fill Extraction

The Calibre xACT digital flow can optionally model coupling effects between nets and fill cells, and can read fill data in DEF, GDS, and OASIS formats.

Calibre xACT uses the setting of the PEX XACT Fill statement to determine whether it will model fill shapes as floating (the default setting) or grounded.The name of the GDS fill top cell must match the name of the DEF design.

Although Calibre xACT can read gzipped fill data, there is typically no performance advantage to zipping GDS fill files.

Create a layer map file to map the metal fill layer numbers onto equivalent DEF layer names. The map file format consists of nine columns in the following order:

- column 1: *LEFDEFlayername*
- column 2: *GDSdefaultlayernumber*
- column 3: *GDSdefaultlayerdatatype*
- column 4: *GDSfilllayernumber*
- column 5: *GDSfilllayerdatatype*
- column 6: *GDStextlayernumber*
- column 7: *GDStextlayerdatatype*
- column 8: *GDScellboundarylayernumber*
- column 9: *GDScellboundarylayerdatatype*

Columns 8 and 9 (GDS cell boundary layer and datatype) are optional. Although they are technically not required at all for importing fill data, the same map file is used for importing GDS macrocells as for importing fill, so it is good practice to include the cell boundary information columns.

For example:

```
metal1 31 0 31 1 31 99 108 0
metal2 32 0 32 1 32 99 108 0
metal3 33 0 33 1 33 99 108 0
....
```

It is possible to map more than one GDS fill layer number to a single LEF/DEF layer. For example, when you want to combine dummy and OPC fill, GDS layers 31.1 and 31.7 respectively, into your M1 layer, repeat the line in the mapfile for the LEF/DEF layer(s) that you want to map:

```
M1 31 0 31 1 31 99 108 0
M1 31 0 31 7 31 99 108 0
```

To import GDS fill data specify the PEX Xcell ... FILL statement in your SVRF rule file. For example:

```
PEX XCELL mydesignname FILL GDS "design/fill/myfillfile_flat.gds" MAP
"mymapfile.txt"
```

You may also use this statement to translate the (x,y) location of the fill cell, which is useful if you are using different origins in your GDS fill file from those in your DEF file. See "Importing GDS Cell Views" for details.

Calibre xACT reads the GDS file and calculates coupling to the individual fill shapes during extraction. Typically there is small overhead for reading the GDS file, but no performance overhead for calculating coupling to the GDS fill shapes, which are treated by Calibre xACT as floating by default, or grounded depending on the setting of the PEX XACT Fill statement.

DEF Metal Fill Extraction

Calibre xACT automatically performs DEF metal fill extraction when the DEF file has metal fill defined in the FILLS section.

If your DEF file includes metal fill, then Calibre xACT instantiates the fill in the database by default and incorporates the effects of the fill the parasitic calculations. There is no way to disable the effects of fill if it is incorporated into the DEF file, other than by using a DEF file that does not contain the FILLS section.

Extraction With GDS Macrocells

The Calibre xACT digital flow can read macro cells in GDSII format and use the GDS geometries in place of those defined in the corresponding macro cells defined in the LEF files.

To import GDS data into Calibre xACT specify the PEX Xcell ... PRIMITIVE statement in your SVRF rule file. For example:

```
PEX XCELL "*" PRIMITIVE GDS "design/gds/mycell.gds" MAP "mymapfile.txt"
```

The wildcard "*" specifies to import all cells contained in the specified GDS file. You can include multiple PEX Xcell statements to import cells from different GDS files. If you rule file contains multiple instances of this statement, it is only necessary to specify the MAP keyword once. Importing specific cells from a single GDS file is not supported.

Multi-Temperature Extraction

The parasitic resistance of interconnects varies at different temperatures. To model the resistance variation, it is common to perform temperature variation extraction at the same process corner. By creating multiple netlists at the minimum, typical, and maximum temperature, and performing simulation at all three temperatures, designers can feel confident that the design will perform as expected at all temperatures.

Temperature extraction is controlled with the PEX Extract Temperature statement. You can create multiple SPEF netlists with different temperature settings by re-running the extraction

step for each temperature. Remember to give a new netlist name for each temperature run using the PEX Netlist statement.

Multi-Corner Multi-Temperature Extraction

The Calibre xACT digital flow supports extraction of multiple process corners and temperatures in a single run. There is a small overhead for each additional corner, typically less than 10%.

The following SVRF statements control the flow:

- PEX XACT Corner — Relates the name of a process corner to the corresponding process technology file used by Calibre xACT.
- PEX Extract Temperature — Specifies temperature dependence which modifies the resistance extracted during netlist generation.

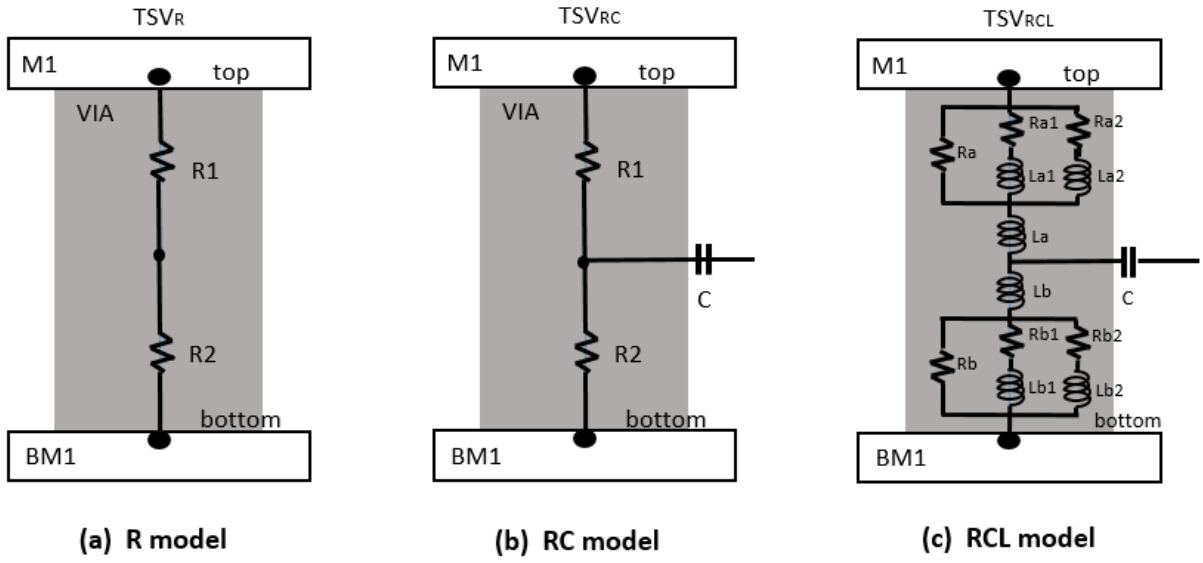
Calibre xACT creates one SPEF file for each process corner/temperature combination specified in the rule deck. For details on how to perform multiple-corner multiple-temperature extraction, see "Netlisting Multiple Corners and Multiple Temperatures".

TSV Extraction

The TSV extraction flow includes the extraction of die or interposer parasitics on both sides of the TSVs, TSV subcircuit insertion, and TSV-to-TSV coupling extraction. TSV extraction is supported in both LEF/DEF (digital) and GDS (analog) extraction flows.

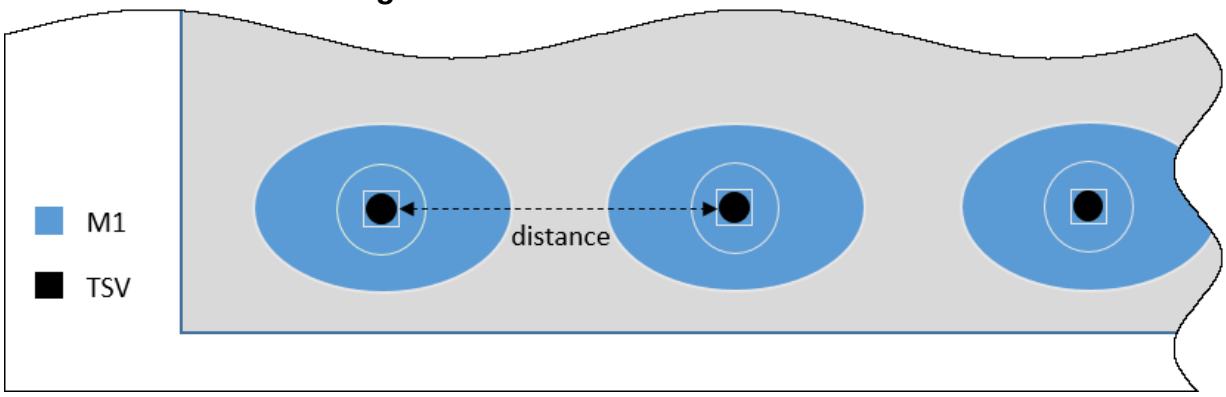
For TSV subcircuit insertion, you can insert TSV models of different complexity (Figure 8-1) at the TSV locations using the PEX Netlist 3DIC statement. The subcircuits are provided in SPICE format. Analog designs typically use R model, RC model, and RLC model. For digital designs only the R and RC models should be used. TSV-to-TSV coupling extraction requires TSV subcircuit insertion.

Figure 8-1. TSV Models



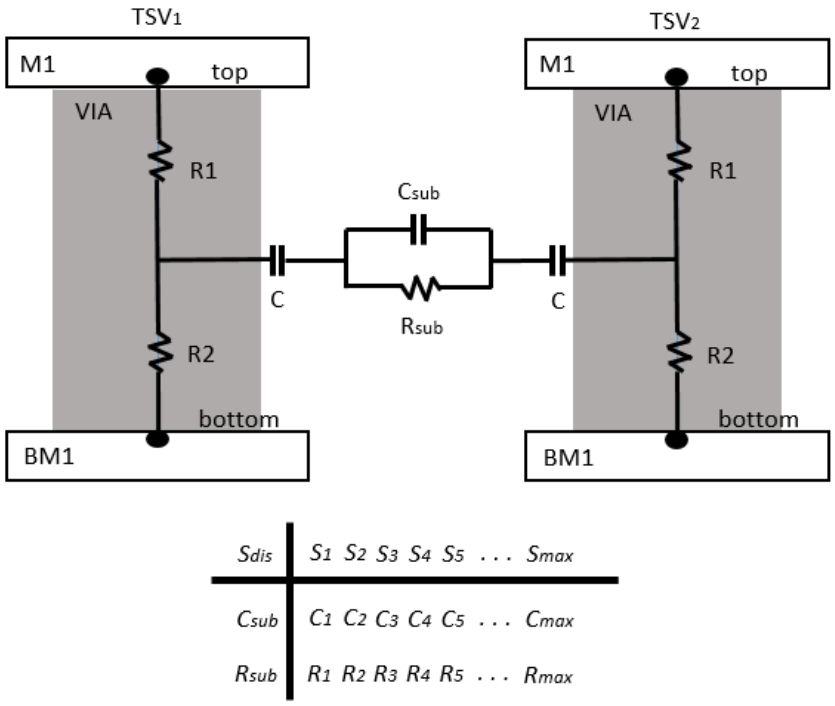
For TSV-to-TSV coupling extraction in analog circuits, a parallel RC circuit (R_{sub} and C_{sub}) is inserted between the edges of the TSVs. This is modeled by the foundry in the calibrated rule files using the analog syntax for TSV_CAPACITANCE Table and the TSV_RESISTANCE Table defined in the TSV via layer in the MIPT file. These tables provide the parasitic values as a function of distance (spacing) between the TSVs.

Figure 8-2. Distance Between TSVs



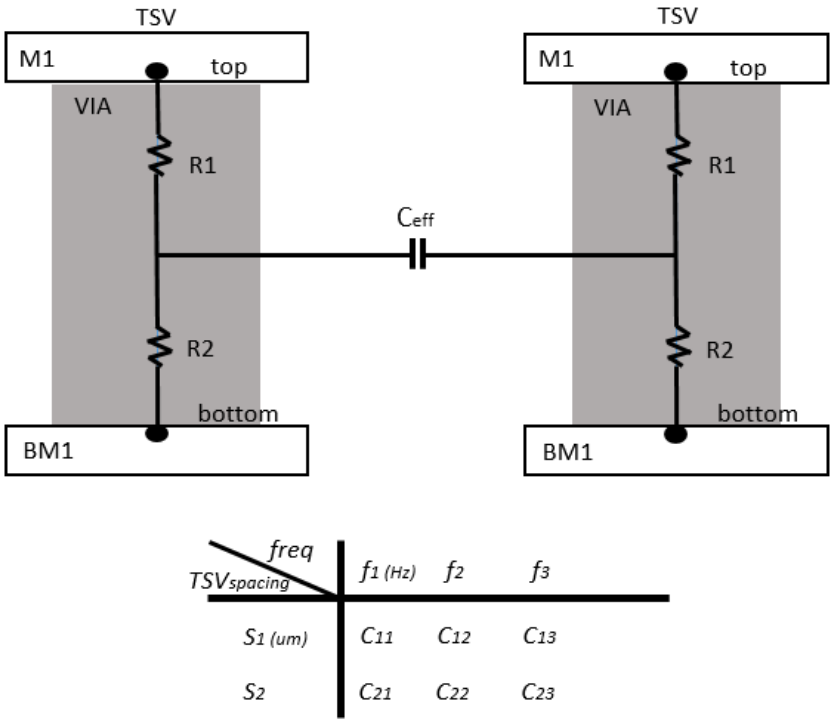
To use the R_{sub} and C_{sub} values from the provided tables and insert them into the extracted netlist, use the PEX 3DIC Coupling statement with the ANALOG keyword. The MAXDISTANCE parameter controls the maximum distance between the TSVs for which the coupling between is calculated (Figure 8-2). Any TSVs beyond the maximum distance are not considered.

Figure 8-3. TSV Circuit for Analog Flow



In digital circuits, the coupling between TSVs is a function of frequency and spacing. A single effective capacitance is inserted between the TSVs. This is modeled by the foundry in the calibrated rule files using the digital syntax for TSV_CAPACITANCE Table defined in the TSV via layer in the MIPT file.

Figure 8-4. TSV Circuit for Digital Flow



To use the C_{eff} values from the provided tables and insert them into the extracted netlist, use the PEX 3DIC Coupling statement with the DIGITAL keyword and frequency value. The MAXDISTANCE parameter controls the maximum distance between the TSVs for which the coupling between is calculated. Any TSVs beyond the maximum distance are not considered.

For examples on how to run TSV extraction for GDS and LEF/DEF flows, see “Extracting TSV Coupling for an Analog Design” on page 130 and “Extracting TSV Coupling for a Digital Design” on page 132.

Calibre xACT 3D Extraction Types

Calibre xACT 3D is recommended for parasitic extraction of small cell block and memory designs.

This section describes the types of extraction supported by Calibre xACT 3D:

Hybrid xACT 3D/Rule-Based Extraction **102**
Extraction Using Boundary Conditions **102**
Hierarchical Memory Extraction **104**
Mixed-Signal Hierarchical Extraction **107**
Calibre xACT 3D Gate-Level Extraction **107**
Calibre xACT 3D Flat Transistor-Level Extraction **108**

Hybrid xACT 3D/Rule-Based Extraction

Hybrid xACT 3D/Rule-Based extraction (-3d -hybrid) extracts all nets selected for Calibre xRC and a subset of nets defined for Calibre xACT 3D, and combines the results into a single parasitic database. Use this flow for high-accuracy extraction of critical nets in a full-chip sign-off process.

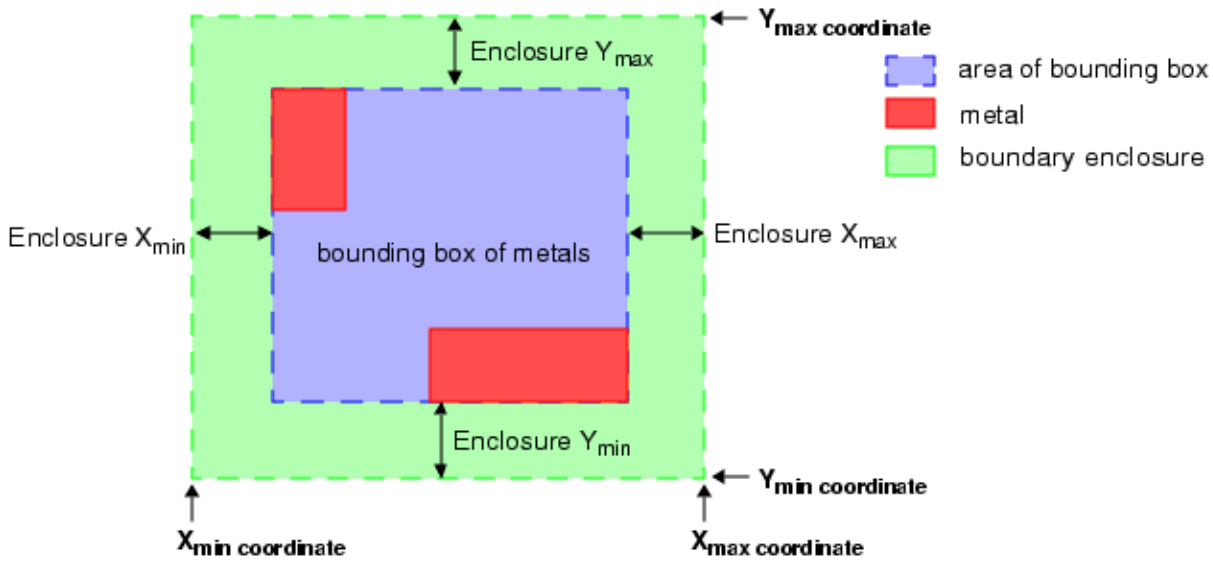
The command line option -hybrid used with the -3d option activates the hybrid xACT 3D/Rule-Based flow. Nets to be extracted with Calibre xRC are selected with the PEX Extract Include statement and the -select command line option. Nets to be excluded from extraction are specified with the PEX Extract Exclude statement. The subset of nets to be extracted by Calibre xACT 3D are defined with PEX Fieldsolver Mode NETS statement. The extraction results from both tools are combined into a single PDB. The -hybrid option is not supported with the -full option or in the direct netlisting flow.

Extraction Using Boundary Conditions

Extraction using boundary conditions allows the extraction of an isolated cell by simulating the parasitic effects of neighboring conductors. This is useful during memory bit cell design to simulate various cell placements and their effects on the memory cell geometries.

A boundary encloses the bounding box of the conductors (metal) in the cell. The size of the boundary is specified in terms of extent in each direction, X_{min} and X_{max} coordinates or Y_{min} and Y_{max} coordinates, or in terms of enclosure in the X or Y directions. Figure 8-5 shows an example of what a bounding box area might look like. Note the colors do not represent different layer types other than metal, but are useful for viewing a pattern in the geometry.

Figure 8-5. Top View of Cell for Boundary

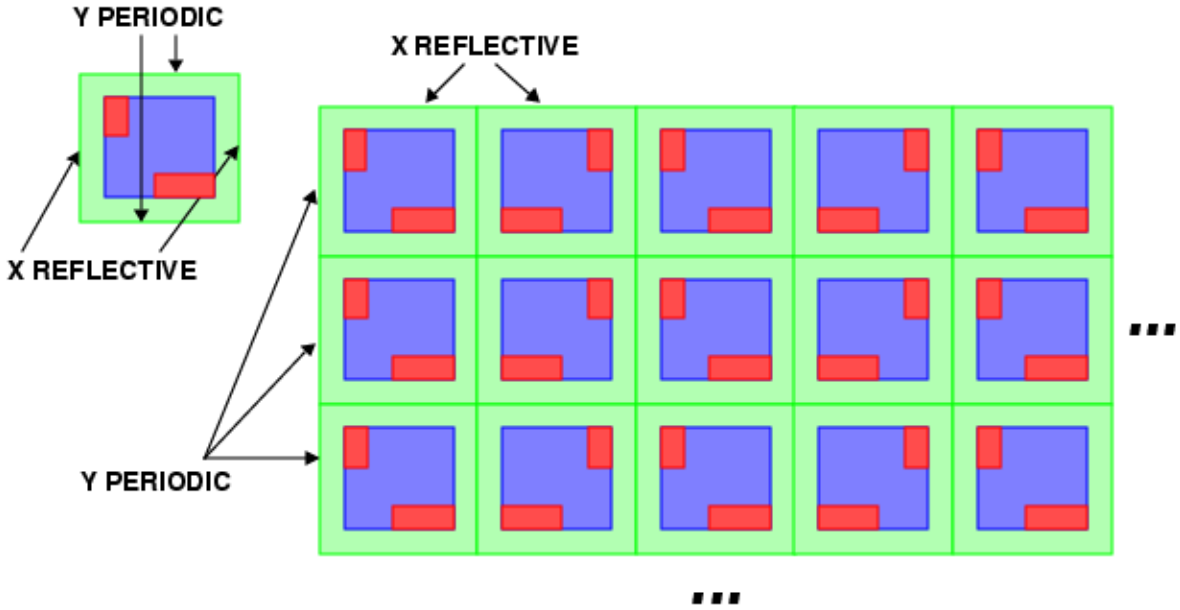


You may define a boundary condition in one of four possible ways:

- **Reflective Boundary** — This boundary condition forces the normal component of the electric field at the boundary to be zero. It is equivalent to placing a reflective wall at the boundary and placing a mirror image of the cell on the other side of this wall and the same distance from it.
- **Periodic Boundary** — This boundary condition replicates the enclosed geometries, creating copies on the other side of the boundary.
- **Reflective Cell Array** — This boundary condition places a mirror image of the enclosed geometries on the other side of the boundary and grounds all conductors in the mirror image.
- **Periodic Cell Array** — This boundary condition copies the enclosed geometries on the other side of the boundary and grounds all copies of conductor geometries.

Figure 8-6 shows an example of a cell_array with reflective setting for the X orientation and periodic setting specified for the Y orientation. The tool determines the number of cells.

Figure 8-6. Cell Array Showing REFLECTIVE Versus PERIODIC



Use the SVRF statements PEX Fieldsolver Boundary and PEX Fieldsolver Cell_array to specify the boundary conditions to be used by the capacitance solver.

Hierarchical Memory Extraction

Hierarchical memory extraction (-xcell -full) extracts data for each user-defined cell as well as the top level of the design. You can nest any number of cells making it ideal for memory arrays. Each cell is extracted only once, which shortens extraction time and netlist size. Hierarchical memory extraction can only be done in the PDB extraction flow.

The output is a netlist containing hierarchical levels matching the xcell levels. The tool defines each xcell in the final output netlist only once and models each net in the xcell. The resulting netlist is suitable for use with a hierarchical simulator.

In contrast to gate-level extraction, where the output includes netlist data only to the level of the cell's boundary, hierarchical memory extraction includes netlist data within the xcells. The n-level netlist contains instantiated xcells with subcircuit definitions. Additionally, this type of extraction:

- Flattens nets beginning in intermediate non-xcell hierarchical levels into the closest parent xcells
- Extracts the successive hierarchical xcell levels until it reaches the primitive device level

Note

Hierarchical memory extraction optimizes extraction runtimes and netlist size. The actual capacitance values for a net extracted hierarchically and flat will be slightly different.

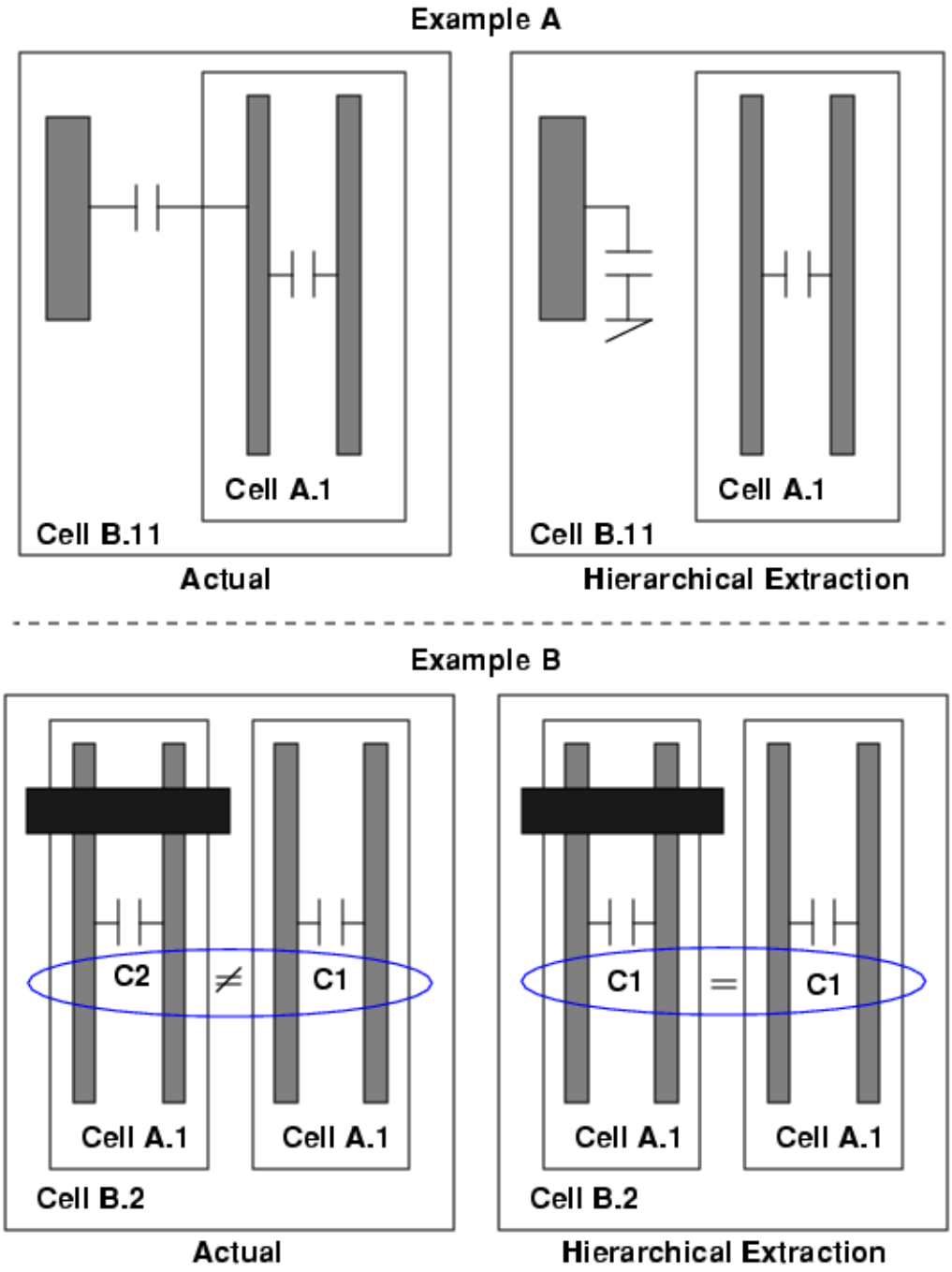
Because data is stored, analyzed, and processed once per cell instead of once for every flat placement of the cell, hierarchical RC or RCC netlisting cannot show the actual effect of geometries that overlap or abut each specific placement of the cell.

Figure 8-7 shows two examples of how information in a hierarchical netlist approximates the actual layout.

Example A shows the actual RC interaction between a cell, Cell A.1, and an adjacent geometry. It also shows how the same construction looks in a hierarchical netlist. Because Cell A.1 is an xcell and thus context-free by default, the RC component is shown between the adjacent geometry and ground.

Example B shows how the actual RC component of two identical cells, Cell A.1, compares when a metal layer is placed over only one of the cells. It also shows how the same construction looks in a hierarchical netlist. The actual RC component of the two cells differs, but because Cell A.1 is predefined, the RC component of the two cells is equal in the hierarchical netlist, despite the overlaying metal layer.

Figure 8-7. Hierarchical vs. Actual RC Network Examples



Related Topics

Running Full Hierarchical and Mixed-Signal Hierarchical Extraction

Mixed-Signal Hierarchical Extraction

Mixed-signal hierarchical extraction generates a parasitic netlist for mixed-signal designs. This type of extraction can only be done in the PDB extraction flow.

You may designate two distinct types of xcells in the xcell list:

- Cells that are extracted hierarchically at the transistor level. These cells are extracted and netlisted as subcircuits in the parasitic netlist.
- Cells that are treated as primitive cells. Primitive cells are identified by -P flags in the xcell list. For these cells, no parasitics are extracted and the contents are not netlisted. They are instantiated in the netlist as cell references.

In both cases the netlist preserves a level of hierarchy for the listed cell. For more information on the xcell file, see “Hierarchy Control with Xcells”.

Related Topics

Running Full Hierarchical and Mixed-Signal Hierarchical Extraction

Calibre xACT 3D Gate-Level Extraction

Gate-level extraction (-xcell) extracts global nets down to top-level cells (logic gates or blocks), which you specify using an xcell list. This type of extraction can only be done in the PDB extraction flow.

When you identify a cell using an xcell list, the Calibre xACT 3D tool extracts the parasitics down to the top-level cell, preserving the cell’s internal structure and hierarchy. The output is a netlist with the following two hierarchical levels:

- Top-level cell
- Instances for the highest-level xcells

You cannot use flat LVS for the PHDB with gate-level extraction.

The highest hierarchical xcell instances define the hierarchy. In gate-level extraction xcell instances do not contain other xcell instances. The Calibre xACT 3D tool ignores the connections within the xcell instances and their lower-level structures (cells, transistors, and nested xcells).

Unlike flat extraction, gate-level netlists include net data to the level of the cell’s boundary. Nets that cross the cell’s boundary are flattened into the parent. You generally use gate-level extraction when you have externally defined libraries containing standard cell data. In this case, include your standard cells in your xcell list.

Calibre xACT 3D supports both LEF/DEF and GDS designs. Power and ground nets are automatically excluded for LEF/DEF designs.

Related Topics

Running Gate-Level Extraction

Calibre xACT 3D Flat Transistor-Level Extraction

Transistor-level extraction flattens the design's interconnect nets into a top-level cell. These designs are typically less than 500,000 transistors and, subsequently, produce either lumped or distributed net models suitable for resistance and capacitance net models. The Calibre xACT 3D tool creates net models for your design in the form of netlists and models parasitic capacitance and resistance you use for input to analysis tools.

When using this type of extraction, the Calibre xACT 3D tool selects the nets in a design and flattens them to the top-level cell. The Calibre xACT 3D tool extracts the nets in their entirety—from device pin to device pin. Typically, the devices are transistors.

Netlists created from a PDB generated with flat extraction do not have any subcircuits.

Flat transistor-level extraction is the default. If the invocation does not contain "-xcell" in the command line, a flat netlist is created and the extraction includes the effects of all geometries.

Related Topics

Running Transistor-Level Extraction

Chapter 9
Producing Parasitic Models

The Calibre xACT tool uses different parasitic models to perform extraction.

This chapter includes the following sections that provide information for producing parasitic models.

Parasitic Devices **109**
Types of Parasitic Models **110**
Distributed Resistance 110
Distributed Resistance and Capacitance 111
Distributed Resistance and Coupled Capacitance 112

Parasitic Devices

There are several sources of parasitic effects in a design.

These parasitic effects include:

- **Intrinsic capacitance** — Capacitance between a wire and the substrate (ground).
- **Coupled capacitance** — Capacitance between two conductors (typically interconnects).
- **Parasitic resistance** — Resistance found as an inherent property of any conductive material.

Types of Parasitic Models

Parasitic models are generally divided into two main types: lumped and distributed.

A lumped parasitic model shows all the parasitic effects as a single element. Distributed parasitic models break up (distribute) the effects more evenly. Distributed parasitic effects can represent either resistance only (referred to as R-only), resistance and capacitance (referred to as RC) or resistance and coupled capacitance (referred to as RCC). When you use RC, coupled capacitance is included with the parasitic capacitance to ground.

The Calibre xACT tool produces the following parasitic model types: Distributed Resistance, Distributed Resistance and Capacitance, Distributed Resistance, and Coupled Capacitance.

With the separately licensed Calibre® xL extension, the Calibre xACT direct netlisting flow and the Calibre xACT 3D PDB flow extend the types of parasitic models to include frequency-dependent resistance, self inductance, and mutual inductance. For more information on extracting parasitic inductance using the direct netlisting flow, see "Getting Started: Calibre xACT Inductance Extraction Using Batch Mode".

Distributed Resistance **110**
Distributed Resistance and Capacitance **111**
Distributed Resistance and Coupled Capacitance **112**

Distributed Resistance

With distributed resistance extraction, the parasitic resistance of the net is broken into segments representing geometric regions. Capacitance is *not* modeled.

Figure 9-1 shows a simplified layout example with the equivalent distributed resistance extraction model. For Net_01, the net sections A to C represent how Calibre xACT segments a net for distributed R extraction. This is also the case for Net_02.

Figure 9-1. Simplified Layout for Distributed Resistance

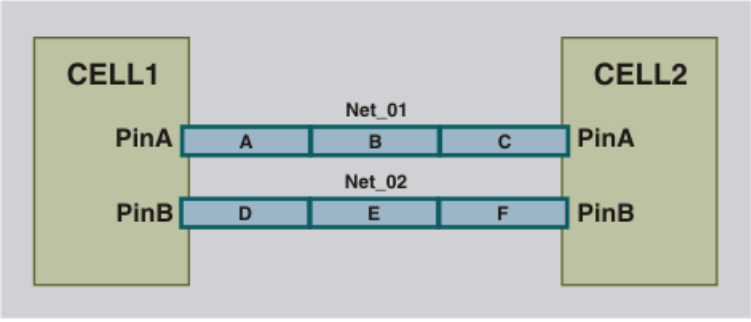
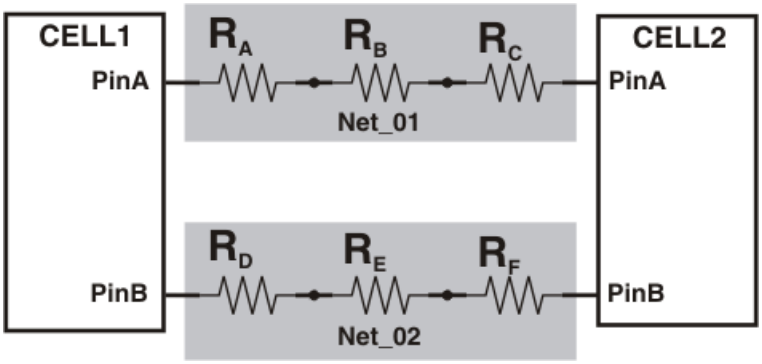


Figure 9-2 shows the extracted net model for distributed R, using the following command lines:

```
%calibre -xact -3d -r
%calibre -xact -fmt -r
```

Figure 9-2. Distributed Resistance Extraction



Distributed Resistance and Capacitance

With distributed resistance and capacitance extraction, the parasitic resistance of the net is broken into segments representing geometric regions. The parasitic capacitance is likewise divided into “local” segments going to the substrate, including the effect of coupled capacitance.

Figure 9-3 shows a simplified layout example for the distributed resistance and capacitance parasitic model. If you do not exclude devices for which you supply the models, your final netlist may double-count the parasitic capacitance.

Figure 9-3. Simplified Layout for Distributed Resistance and Capacitance

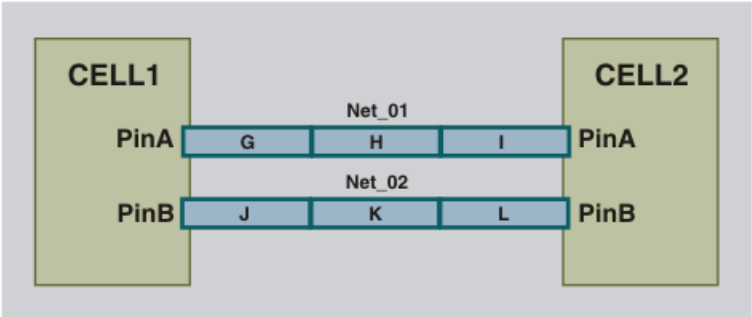


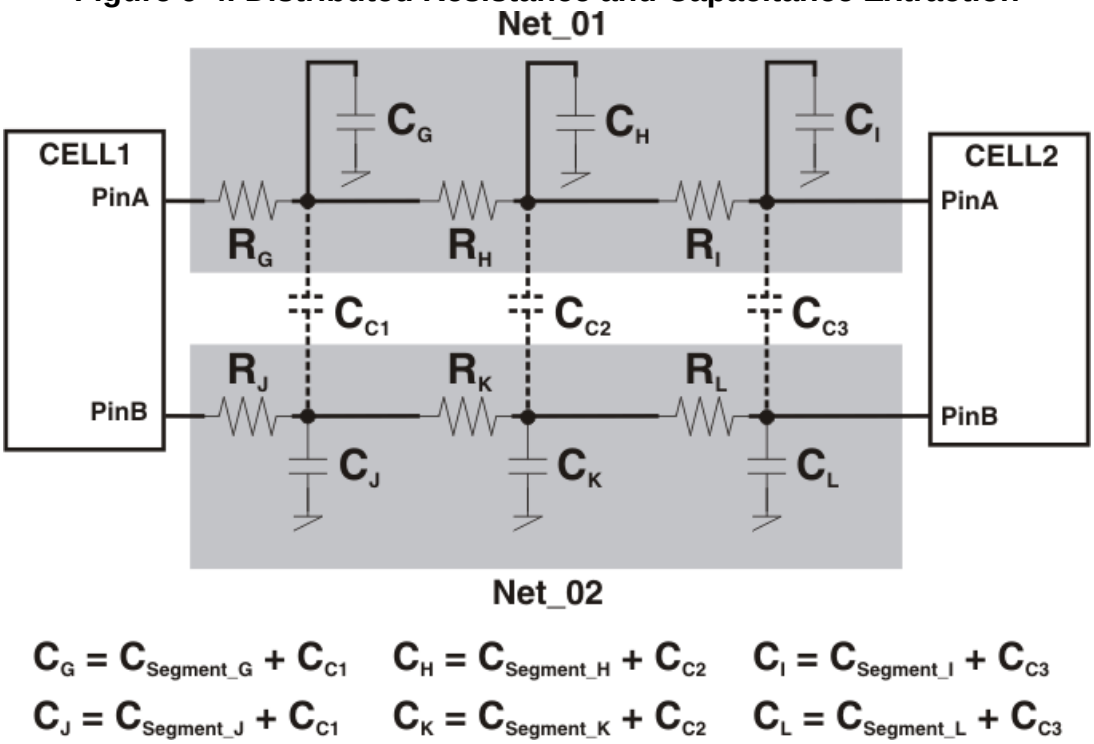
Figure 9-4 shows the extracted net model for distributed R and C, using the following command lines:

```
% calibre -xact -3d -rc
% calibre -xact -fmt -g [ -rc | -all ]
```

In the formatting step, the option -all is used for writing distributed results; it does not produce lumped capacitance.

Figure 9-4 also shows how the intrinsic capacitors for each net segment include the effect of the coupled capacitor, for example, capacitor CG is the sum of the intrinsic capacitance for segment G plus the coupled capacitance between segment G of Net_01 and segment J of Net_02. Each net is also shaded separately to show that they are not explicitly coupled together.

Figure 9-4. Distributed Resistance and Capacitance Extraction



$$C_G = C_{Segment\_G} + C_{C1} \quad C_H = C_{Segment\_H} + C_{C2} \quad C_I = C_{Segment\_I} + C_{C3}$$

$$C_J = C_{Segment\_J} + C_{C1} \quad C_K = C_{Segment\_K} + C_{C2} \quad C_L = C_{Segment\_L} + C_{C3}$$

Distributed Resistance and Coupled Capacitance

With distributed resistance and coupled capacitance, the parasitic resistance of the net is divided into segments; however, parasitic capacitance is modeled with separate elements for coupled capacitance and intrinsic capacitance (capacitance to substrate).

Figure 9-5 shows a simplified layout example for the distributed resistance and coupled capacitance parasitic model.

Figure 9-5. Simplified Layout for Distributed Resistance With Coupled Capacitance

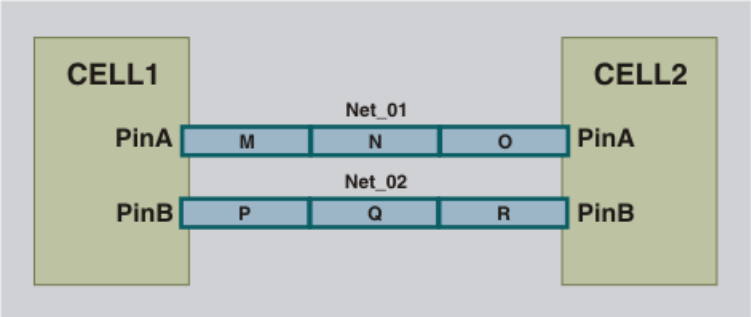
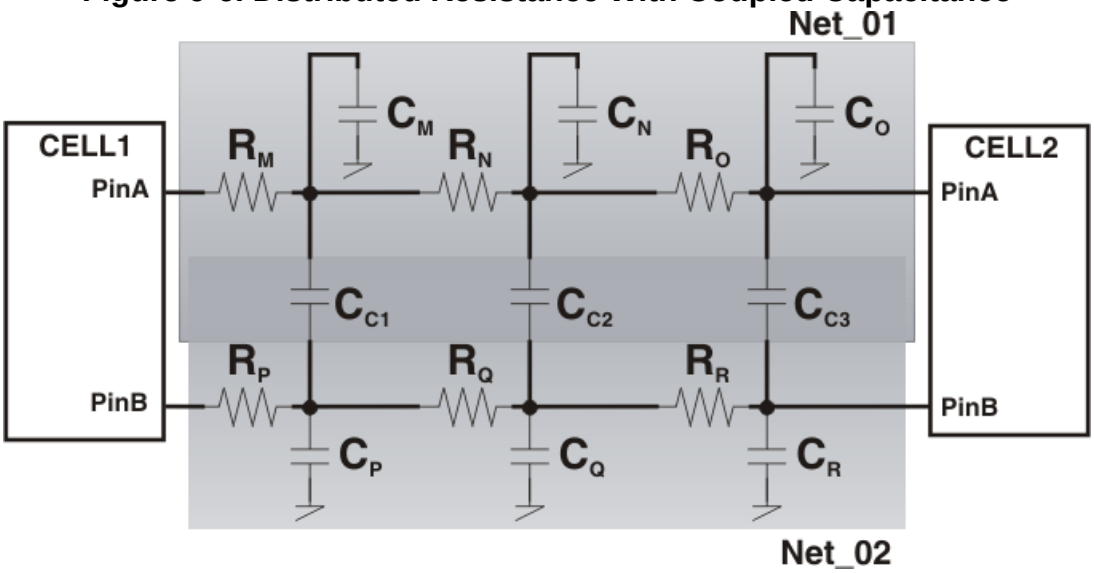


Figure 9-6 shows the extracted net model for distributed R with coupled C when you use the following commands:

```
% calibre -xact -3d -rcc
% calibre -xact -fmt [ -rcc | -all ]
```

In the formatting step, the -all option is used for writing distributed results; it does not produce lumped capacitance.

Figure 9-6. Distributed Resistance With Coupled Capacitance



The overlapping shaded areas show that Net_01 and Net_02 are capacitively coupled through CC1 to CC3. These coupled capacitors are explicitly included in the model and are not added to the intrinsic capacitors, as in the distributed R and C case.

Chapter 10
Basic Extraction Methods

Extraction can be performed in a variety of ways including extraction type, netlist type, or design hierarchy, and produce various netlists or reports depending on your verification needs.

This chapter provides procedures for common extraction runs using both the command-line interface and the Calibre Interactive GUI. Each procedure details the minimum required steps.

Prerequisites for Performing Parasitic Extraction 116

Running Gate-Level Extraction 117
- Creating a Gate-Level Netlist with Calibre xACT 117
- Creating a Gate-Level Netlist with Calibre xACT 3D 118
- Creating a Gate-Level Netlist from Calibre Interactive with Calibre xACT 3D 119

Running Transistor-Level Extraction 121
- Creating a Transistor-Level Netlist with Calibre xACT 121
- Creating a Transistor-Level Netlist Using Calibre xACT 3D Direct Netlisting 122
- Creating a Transistor-Level Netlist Using the Calibre xACT 3D PDB Extraction Flow . . 123
- Creating a Transistor-Level Netlist from Calibre Interactive with Calibre xACT 3D 124

Running Full Hierarchical and Mixed-Signal Hierarchical Extraction 126
- Creating a Hierarchical Netlist from the Command Line 126
- Creating a Hierarchical Netlist from Calibre Interactive 127

Running TSV Extraction 130
- Extracting TSV Coupling for an Analog Design 130
- Extracting TSV Coupling for a Digital Design 132

Netlisting a Design Without Parasitics 136
- Creating an Ideal Netlist from the Command Line 136
- Creating an Ideal Netlist from Calibre Interactive 137

Backannotating Parasitics to a Source Netlist 138
- Backannotating from the Command Line Using Calibre xACT 3D Direct Netlisting 138
- Backannotating from the Command Line Using the Calibre xACT 3D PDB Flow 139
- Backannotating from Calibre Interactive 140

Generating a Capacitance Summary Report 141

Net-to-Net Coupling Capacitance Report 142
- Reporting Coupled Capacitance from the Command Line 142
- Reporting Coupled Capacitance from Calibre Interactive 143

Point-to-Point Resistance Reports 145
- Reporting Net Resistance from the Command Line 145
- Reporting Net Resistance from Calibre Interactive 146

Top Level Only Extraction 148

Prerequisites for Performing Parasitic Extraction

Running Calibre xACT extraction requires that certain conditions be met.

These conditions include:

- Geometric database containing the layout.
- The design must be LVS clean, using the same device and connectivity definitions that will be used for parasitic extraction.
- To run any of the Calibre xACT tools, you must have the license files required by the tool. Refer to "Licensing: Parasitic Extraction Products" in the *Calibre Administrator's Guide* for details.
- An SVRF file for the type of parasitic extraction and the layout. In addition to your Calibre nmLVS information, the file should contain the following:
  - Parasitic calculation statements, usually provided as a separate file by the foundry and included by reference.

  For command-line execution, the file must also include the following statements:

  - A PEX Netlist statement to set netlist format and optional content.
  - Layout System, Layout Path, and Layout Primary statements to identify the layout format, design files, and the top cell of the design.

Running Gate-Level Extraction

The “gates” in gate-level extraction refer to logic gates. These and other predefined circuits are listed in the xcell file. Cells listed in the xcell file appear in the netlist as subcircuit instances. Gate-level extraction must follow the Calibre xACT 3D PDB extraction flow.

Note

If an xcell file is empty or the cell names do not match the input, the resulting netlist is flat.

Creating a Gate-Level Netlist with Calibre xACT 117
Creating a Gate-Level Netlist with Calibre xACT 3D 118
Creating a Gate-Level Netlist from Calibre Interactive with Calibre xACT 3D. 119

Creating a Gate-Level Netlist with Calibre xACT

Create a gate-level netlist using the Calibre xACT direct netlisting flow.

Prerequisites

- Hcell file or Hcell statement that includes all cells also listed in the xcell file.
- Xcell file listing cells which will not undergo parasitic extraction. All entries are treated as primitives. You can re-use the hcell file.
- Layout database that is LVS-clean.
- A valid Calibre xACT rule file for this layout. For more information refer to “Creating a Top-level Control File”.
- For more information refer to “Prerequisites for Performing Parasitic Extraction”.

Procedure

1. Build the database of intentional devices. Note: Flags that are specific to gate-level extraction are in **bold**.

   To use source names, use Calibre nmLVS-H:

   ```
   calibre -lvs -hier -hcell hcell_file -spice $svdb_dir/top_cell.sp
   rules
   ```

2. Extract parasitic effects with Calibre xACT.

   ```
   calibre -xact -xcell xcell_file -parasitic_switch rules
   ```

   where ***parasitic_switch*** indicates the type of parasitics to extract. For example, -rc for distributed resistance and capacitance.

Results

The transcript for a successful Calibre xACT run concludes with a count of errors and warnings. For example:

```
                CALIBRE xACT WARNING / ERROR Summary
------------------------------------------------------------------------
                        xACT Warnings   =   2
                          xACT Errors   =   0

========================================================================
```

If there are no errors, the directory also contains the transistor-level netlist you specified in the PEX Netlist statement. The exact number of files depends on the output format.

Creating a Gate-Level Netlist with Calibre xACT 3D

Create a gate-level netlist using the Calibre xACT 3D PDB flow from the command line.

Prerequisites

- Hcell file or Hcell statement that includes all cells also listed in the xcell file.
- Xcell file listing cells which will not undergo parasitic extraction. All entries are treated as primitives. You can re-use the hcell file.
- Layout database that is LVS-clean.
- A valid PEX rule file for this layout.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Build the database of intentional devices. Note: Flags that are specific to gate-level extraction are in **bold**.

   To use source names, use Calibre nmLVS-H:

   ```
   calibre -lvs -hier -hcell hcell_file -spice $svdb_dir/top_cell.sp
   rules
   ```

   To use layout names, use Calibre xACT:

   ```
   calibre -xact -phdb -hcell hcell_file rules
   ```

2. Extract parasitic effects with Calibre xACT 3D.

   ```
   calibre -xact -3d -pdb -xcell xcell_file -parasitic_switch rules
   ```

   where ***parasitic_switch*** indicates the type of parasitics to extract. For example, -rc for distributed resistance and capacitance.

3. Generate the netlist. You do not need to specify the xcell list.

```
calibre -xact -fmt rules
```

Results

The transcript for a successful Calibre xACT 3D run concludes with a count of errors and warnings. For example:

```
                  CALIBRE xACT WARNING / ERROR Summary
---------------------------------------------------------------------
                         xACT Warnings   =   2
                           xACT Errors   =   0

=====================================================================
```

If there are no errors, the directory also contains the transistor-level netlist you specified in the PEX Netlist statement. The exact number of files depends on the output format.

Related Topics

Calibre xACT Tool Invocation Reference

Calibre xACT 3D Gate-Level Extraction

Creating a Gate-Level Netlist from Calibre Interactive with Calibre xACT 3D

Create a gate-level netlist using Calibre xACT 3D from Calibre Interactive. Calibre Interactive uses the Calibre xACT 3D PDB extraction flow.

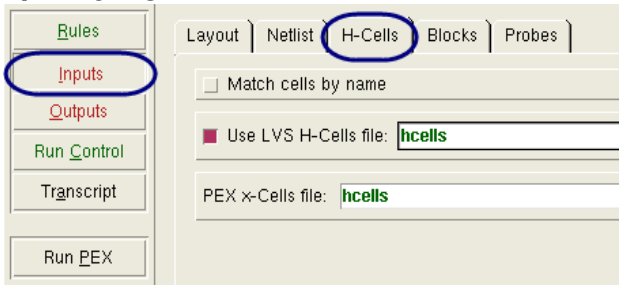
Prerequisites

- Hcell file or Hcell statement that includes all cells also listed in the xcell file.
- Xcell file listing cells which will not undergo parasitic extraction. All entries are treated as primitives. You can re-use the hcell file.
- Layout database that is LVS-clean.
- A valid PEX rule file for this layout.
- For more information refer to “Prerequisites for Performing Parasitic Extraction”.

Procedure

1. Start the PEX interface in Calibre Interactive:
2. Load a runset or rule file.
3. Specify the hcells and xcells by clicking the **Inputs** button in the left pane. Select the **H-Cells** tab. Specify the hcell and xcell files in the appropriate fields. (They can be the same file.)

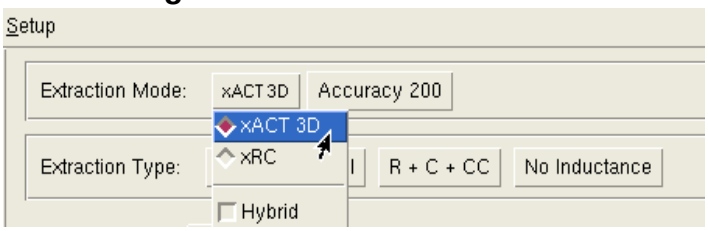
Figure 10-1. Specifying HCell and XCell Files in Calibre Interactive



4. Specify the extraction type by clicking the **Outputs** button in the left pane.

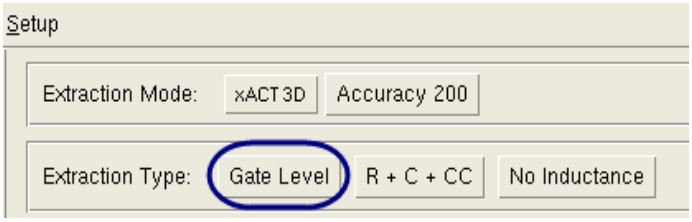
   a. Set Extraction Mode to **xACT 3D**.

Figure 10-2. Extraction Mode



   b. In the area above the tabs, set Extraction Type to **Gate Level**.

Figure 10-3. Gate Level Setting



5. Set other controls as needed.
6. Click the **Run PEX** button in the left pane.

Results

Check the Transcripts pane to verify the run completed with no errors. If you have selected "View netlist after PEX finishes" in the Outputs pane, a text viewer appears with the generated netlist loaded.

Related Topics

Getting Started: Calibre xACT 3D Parasitic Extraction Using Calibre Interactive

Calibre xACT 3D Gate-Level Extraction

Running Transistor-Level Extraction

In transistor-level extraction, devices and nets are examined without regard to cell boundaries. It is the most accurate form of standard Calibre xACT extraction, but also the most resource intensive.

Creating a Transistor-Level Netlist with Calibre xACT 121
Creating a Transistor-Level Netlist Using Calibre xACT 3D Direct Netlisting. 122
Creating a Transistor-Level Netlist Using the Calibre xACT 3D PDB Extraction Flow 123
Creating a Transistor-Level Netlist from Calibre Interactive with Calibre xACT 3D. 124

Creating a Transistor-Level Netlist with Calibre xACT

Transistor-level netlists do not have any specific required SVRF statements or environment variables. They also do not require hcell or xcell files.

Prerequisites

- Layout database that is LVS-clean.
- A valid Calibre xACT rule file for this layout. For more information refer to "Creating a Top-level Control File".
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Build the database of intentional devices.

   To use source names, use Calibre nmLVS-H:

   ```
   calibre -lvs -hier -spice $svdb_dir/top_cell.sp rules
   ```

   To use layout names, proceed to step 2.

2. Extract parasitic effects with Calibre xACT and generate the netlist.

   ```
   calibre -xact -parasitic_switch rules
   ```

   where ***parasitic_switch*** indicates the type of parasitics to extract. For example, for distributed resistance and coupling capacitance (-rcc), use the following:

   ```
   calibre -xact rules
   ```

   This is the default behavior if a parasitic switch is not specified. To produce a netlist that only contains lumped capacitance (-c), use the following:

   ```
   calibre -xact -c rules
   ```

Results

The transcript for a successful Calibre xACT run concludes with a count of errors and warnings. For example:

```
                  CALIBRE xACT WARNING / ERROR Summary
-------------------------------------------------------------------------
                          xACT Warnings   =   2
                            xACT Errors   =   0

=========================================================================
```

If there are no errors, the directory also contains the transistor-level netlist you specified in the PEX Netlist statement. The exact number of files depends on the output format.

Related Topics

Calibre xACT Flat Transistor-Level Extraction

Calibre xACT Tool Invocation Reference

calibre -xact

Creating a Transistor-Level Netlist Using Calibre xACT 3D Direct Netlisting

Create a transistor-level netlist using the Calibre xACT 3D direct netlisting flow.

Prerequisites

- Layout database that is LVS-clean.
- A valid PEX rule file for the layout.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Build the database of intentional devices.

   To use source names, use Calibre nmLVS-H:

   ```
   calibre -lvs -hier -spice $svdb_dir/top_cell.sp rules
   ```

   If you are using layout names, then you can skip this step.

2. Extract parasitic effects and generate the netlist.

   ```
   calibre -xact -3d -parasitic_switch -turbo rules
   ```

   where ***parasitic_switch*** indicates the type of parasitics to extract. For example, -rc for distributed resistance and capacitance.

   ```
   calibre -xact -3d -rc -turbo rules
   ```

Results

The transcript for a successful Calibre xACT 3D run concludes with a count of errors and warnings. For example:

```
                  CALIBRE xACT WARNING / ERROR Summary
------------------------------------------------------------------------
                          xACT Warnings   =   0
                            xACT Errors   =   0

========================================================================
```

If there are no errors, the directory also contains the transistor-level netlist you specified in the PEX Netlist statement. The exact number of files depends on the output format.

Creating a Transistor-Level Netlist Using the Calibre xACT 3D PDB Extraction Flow

Transistor-level netlists do not have any specific required SVRF statements or environment variables. They also do not require hcell or xcell files.

This example demonstrates how to create a transistor-level netlist using the Calibre xACT 3D PDB extraction flow.

Prerequisites

- Layout database that is LVS-clean.
- A valid PEX rule file for this layout.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Build the database of intentional devices.

   To use source names, use Calibre nmLVS-H:

   ```
   calibre -lvs -hier -spice $svdb_dir/top_cell.sp rules
   ```

   To use layout names, use Calibre xACT 3D:

   ```
   calibre -xact -phdb rules
   ```

2. Extract parasitic effects.

   ```
   calibre -xact -3d -pdb -parasitic_switch rules
   ```

   where ***parasitic_switch*** indicates the type of parasitics to extract. For example, -rc for distributed resistance and capacitance.

3. Generate the netlist.

If ***parasitic_switch*** included resistance, use the following:

```
calibre -xact -fmt -all rules
```

Results

The transcript for a successful Calibre xACT 3D run concludes with a count of errors and warnings. For example:

```
                  CALIBRE xACT WARNING / ERROR Summary
------------------------------------------------------------------------
                         xACT Warnings   =  2
                           xACT Errors   =  0

========================================================================
```

If there are no errors, the directory also contains the transistor-level netlist you specified in the PEX Netlist statement. The exact number of files depends on the output format.

Related Topics

Calibre xACT Tool Invocation Reference

Creating a Transistor-Level Netlist from Calibre Interactive with Calibre xACT 3D

Perform transistor-level extraction using the Calibre xACT 3D PDB extraction flow from Calibre Interactive.

Prerequisites

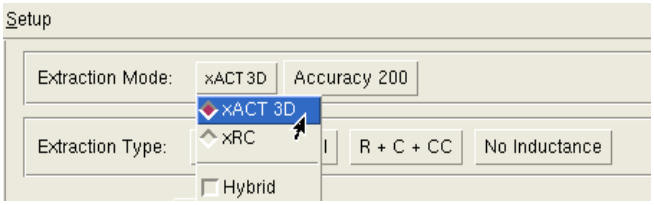
- Layout database that is LVS-clean.
- A valid PEX rule file for this layout.
- For more information refer to “Prerequisites for Performing Parasitic Extraction”.

Procedure

1. Start the PEX interface in Calibre Interactive.
2. Load a runset or rule file.
3. Specify the extraction mode.

   Click the **Outputs** button in the left pane. Set Extraction Mode to **xACT 3D**.

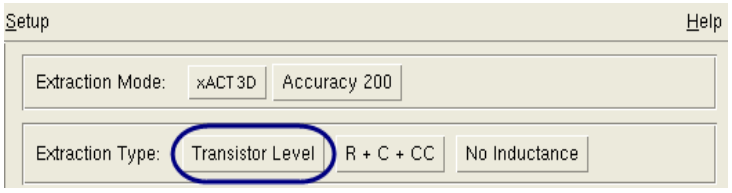
Figure 10-4. Extraction Mode Setting



4. Specify the extraction type.

   Set Extraction Type to **Transistor Level**.

Figure 10-5. Transistor Level Setting



5. Set other controls as needed.

   You do not need to clear the H-Cells field. Because Extraction Type is set to transistor level, the H-Cells settings are ignored.

6. Click the **Run PEX** button in the left pane.

Results

Check the Transcripts pane to verify the run completed with no errors. If you have selected "View netlist after PEX finishes" in the Outputs pane, a text viewer appears with the generated netlist loaded.

Related Topics

Getting Started: Calibre xACT 3D Parasitic Extraction Using Calibre Interactive

Running Full Hierarchical and Mixed-Signal Hierarchical Extraction

The difference between full hierarchical extraction and mixed-signal hierarchical extraction is whether or not all xcells are extracted in full and independent of the layout. The xcell file specifies whether each cell is a “regular” cell (no flag or a -c) to be extracted in full or a primitive cell (-p) and not extracted. The invocation and SVRF file are identical. This extraction type is only valid when using the Calibre xACT 3D PDB extraction flow.

Note

If an xcell file is empty or the cell names do not match the input, the resulting netlist is flat.

Full hierarchical extraction is intended for use on designs with significant amounts of repeated hierarchy such as memory.

Creating a Hierarchical Netlist from the Command Line **126**
Creating a Hierarchical Netlist from Calibre Interactive **127**

Creating a Hierarchical Netlist from the Command Line

Create a hierarchical netlist the Calibre xACT 3D from the command line using the PDB extraction flow.

Prerequisites

- A valid PEX rule file for this layout. If you are creating a DSPF netlist for use in a hierarchical simulator that accepts a position file, add PEX Netlist Position File.
- Hcell file or Hcell statement that includes all cells also listed in the xcell file.
- Xcell file listing cells to preserve in the parasitic netlist.
- Layout database that is LVS-clean.
- For more information refer to “Prerequisites for Performing Parasitic Extraction”.

Procedure

1. If you are netlisting to DSPF, use the PEX Netlist statement with the DSPF and HSIM keywords. It improves modeling of feedthrough nets, but only affects DSPF output.
2. Build the database of intentional devices. Note: Flags that are specific to hierarchical extraction are in **bold**.

To use source names, use Calibre nmLVS-H:

```
calibre -lvs -hier -hcell hcell_file -spice $svdb_dir/top_cell.sp
rules
```

To use layout names, use Calibre xACT:

```
calibre -xact -phdb -hcell hcell_file rules
```

3. Extract parasitic effects with Calibre xACT 3D.

```
calibre -xact -3d -pdb -xcell xcell_file -full -parasitic_switch
rules
```

where ***parasitic_switch*** indicates the type of parasitics to extract. For example, -rc for distributed resistance and capacitance.

4. Generate the netlist. You do not need to specify the xcell list.

```
calibre -xact -fmt -full rules
```

Results

Successful Calibre xACT 3D transcripts conclude with a count of errors and warnings as shown.

```
                CALIBRE xACT WARNING / ERROR Summary
------------------------------------------------------------------------
                        xACT Warnings   =   2
                          xACT Errors   =   0

========================================================================
```

If there are no errors, the directory also contains the netlist you specified in the PEX Netlist statement. Each xcell appears in the netlist as a subcircuit. The exact number of files depends on the output format.

Related Topics

Calibre xACT Tool Invocation Reference

Hierarchical Memory Extraction

Mixed-Signal Hierarchical Extraction

Creating a Hierarchical Netlist from Calibre Interactive

Create a netlist from Calibre Interactive with an SVRF file setup for your type of parasitic extraction, and specifying only the settings particular to producing hierarchical netlists. Extraction performed from Calibre Interactive follows the PDB extraction flow.

Prerequisites

- A valid PEX rule file for this layout. If you are creating a DSPF netlist for use in a hierarchical simulator that accepts a position file, add PEX Netlist Position File.
- Hcell file or Hcell statement that includes all cells also listed in the xcell file.
- Xcell file listing cells to preserve in the parasitic netlist.
- Layout database that is LVS-clean.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

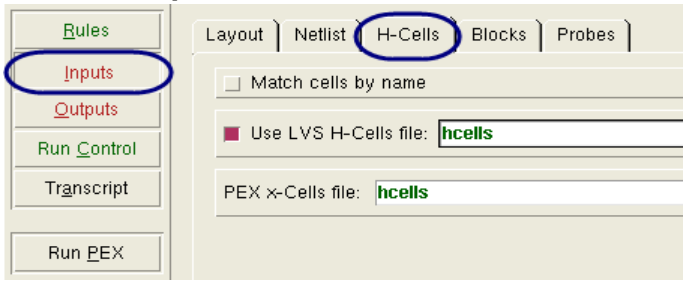
Procedure

1. Start the PEX interface in Calibre Interactive.

```
calibre -gui -pex
```

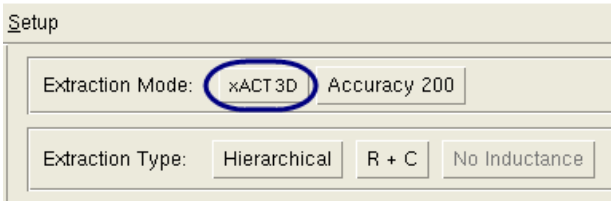
2. Load a runset or rule file.
3. Specify the hcells and xcells by clicking the **Inputs** button in the left pane. Select the **H-Cells** tab. Specify the hcell and xcell files in the appropriate fields. (They can be the same file.)

Figure 10-6. Inputs Pane for Hierarchical Extraction



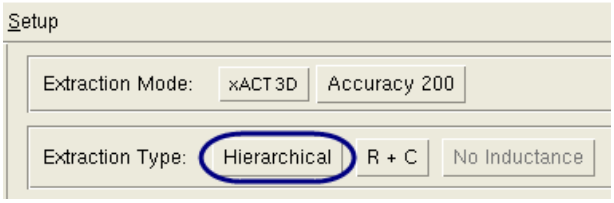
4. Specify the Extraction Type by clicking the **Outputs** button in the left pane.
   a. Set Extraction Mode to **xACT 3D**.

Figure 10-7. Set Extraction Mode



   b. In the area above the tabs, set Extraction Type to **Hierarchical**.

Figure 10-8. Hierarchical Extraction Type Setting.



5. Set other controls as needed.
6. Click the **Run PEX** button in the left pane.

Results

Check the Transcripts pane to verify the run completed with no errors. If you have selected "View netlist after PEX finishes" in the Outputs pane, a text viewer appears with the generated netlist loaded.

Related Topics

Getting Started: Calibre xACT 3D Parasitic Extraction Using Calibre Interactive

Hierarchical Memory Extraction

Running TSV Extraction

Calibre xACT supports die and interposer TSV extraction for both GDS and LEF/DEF designs. TSV extraction includes the extraction of the interposer parasitics on both sides of the TSVs, TSV subcircuit insertion, and TSV to TSV coupling extraction.

Extracting TSV Coupling for an Analog Design. 130
Extracting TSV Coupling for a Digital Design 132

Extracting TSV Coupling for an Analog Design

Create a netlist that inserts a TSV and includes substrate coupling between TSVs in an analog design using the GDS extraction flow.

Prerequisites

- Calibrated rule files that contain TSV resistance (R_{sub}) and TSV capacitance (C_{sub}) parasitic values as a function of spacing between the TSVs.

  To demonstrate the behavior this example assumes the following tables were defined in the MIPT file used to generate the calibrated rule files as a part of the TSV via layer definition.

```
...
table = tsv_capacitance {
        property = tsv_capacitance
        table_type = C
        dim_type = drawn
        value_type = absolute
        spacing = { 60 70 80 90 100 110 120 130 140 150 160 170 180
190 200 210 220 230 }
        value = { 40 35 31 28 25 23 21 19 18 17 16 15 14 13 12 11 10
9}

table = tsv_resistance {
        property = tsv_resistance
        table_type = R
        dim_type = drawn
        value_type = absolute
        spacing = { 60 70 80 90 100 110 120 130 140 150 160 170 180
190 200 210 220 230 }
        value = { 1200 1300 1350 1450 1550 1600 1700 1750 1800 1850
1900 1950 2000 2050 2090 2100 2130 2150}
...
```

  The values in bold are used to demonstrate the results.

- A TSV subcircuit definition in SPICE format. This example uses an RC subcircuit definition.
- A GDS design whose layout database is LVS-clean

- A valid top level GDS rule file. For more information refer to "Creating a Top-level Control File".

Procedure

1. The file *tsv_rc.subckt* contains an RC subcircuit definition for a TSV.

```
.subckt     tsv_rc    top    bottom
            R1        top    n2        5555
            R2        n2     bottom    5555
            C         n2     sub       7777f
.ends
```

2. Verify the top-level rule file, *top_gds.rules*, contains the following specification information.

   The layout database statements are used to specify the LVS and annotated GDS files generated by Calibre 3DStack for interposer extraction. For example:

```
LAYOUT PRIMARY "TOPCELL_3DIC"
LAYOUT PATH "./RC_interposer_gds/RC_interposer.gds"
INCLUDE ./RC_interposer_gds/RC_interposer.gds.map.svrf
```

   Specify the extracted netlist output in DSPF format using layout names with:

```
PEX NETLIST "tsv_netlist.dspf" DSPF LAYOUTNAMES LOCATION CLAYER
CLOCATION RLAYER RLOCATION RAREA RWIDTH
```

3. To include the TSV subcircuit in the extracted netlist add the PEX Netlist 3DIC statement to the *top_gds.rules* file.

```
PEX NETLIST 3DIC tsv_rc FILE "./inputs/tsv_rc.subckt" SUBNODE sub
```

   where *tsv_rc* is an LVS layer or a calibrated layer name, and *sub* is TSV circuit middle node.

4. To also include the couplings between the TSVs that are within a distance of 275 microns of each other, add the PEX 3DIC Coupling statement to the *top_gds.rules* file.

```
PEX 3DIC COUPLING tsv_rc ANALOG MAXDISTANCE 275
```

   When the **ANALOG** keyword is specified, the coupling capacitance between the TSVs is extracted based on a parallel combination of resistance and capacitance. The parasitic values are provided as a function of spacing between the TSVs.

5. Save and close the *top_gds.rules* file.

6. Run Calibre xACT to extract RC parasitic effects and generate the netlist.

```
calibre -xact -rc top_gds.rules
```

Results

If there are no errors, the directory should contain the DSPF netlist, *tsv_netlist.dspf*.

Examine the contents of the netlist. The netlist should contain the inserted TSV information. The resistance and capacitance values should match those from the subcircuit specified in *tsv_rc.subckt*.

```
...
R1180 TEST_SI_DIE1[14]:52 TEST_SI_DIE1[14]:58 5555 $X=542.5 $Y=582.5
+ $X2=542.5 $Y2=589.0
R1153 TEST_SI_DIE1[14]:52 TEST_SI_DIE1[14]:67 5555 $w=1.4e-05
+ $layer=metal1 $X=542.5 $Y=582.5 $X2=542.5 $Y2=589.0
Cc_1481 TEST_SI_DIE1[14]:52 TEST_SI_DIE1[14]:96 7777f $X=542.5 $Y=582.5
...
```

The netlist should also include the TSV coupling information:

```
...
Cc_2104 TEST_SI_DIE1[8]:107 TEST_SI_DIE1[4]:117 16f $X=182.5 $Y=764.875
Rc_1483 TEST_SI_DIE1[14]:96 TEST_SI_DIE1[18]:111 1900
...
```

The capacitance and resistance coupling values were obtained from the tsv_capacitance and tsv_resistance tables for the spacing value of 160 microns.

Extracting TSV Coupling for a Digital Design

Create a netlist that inserts a TSV and includes coupling capacitance between TSVs in a digital design using the LEF/DEF extraction flow.

Prerequisites

- Calibrated rule files that contain TSV capacitance (C_{eff}) parasitic values as a function of spacing between the TSVs and frequency.

  To demonstrate the behavior this example assumes the following table was defined in the MIPT file used to generate the calibrated rule files as a part of the TSV via layer definition.

```
...
table = tsv_capacitance_eff {
        property = tsv_capacitance
        table_type = C
        dim_type = drawn
        value_type = absolute
        frequency = { 5.0E+07 7.0E+07 9.0E+07 1.2E+08 1.4E+08
1.6E+08 1.9E+08 2.1E+08 2.3E+08 2.6E+08 2.8E+08 3.1E+08 3.3E+08
3.5E+08 3.8E+08 4.0E+08 4.2E+08 4.5E+08 4.7E+08 5.0E+08 }
        spacing = { 60 70 80 90 100 110 120 130 140 150 160 170 180
190 200 210 220 230 }
        value = { 2600 2400 2270 2120 2030 1900 1800 1790 1700 1680
1630 1600 1570 1550 1530 1510  1500 1490 ,
1770 1650 1550 1460 1380 1320 1260 1210 1170 1140 1110 1080 1060
1050 1030 1020 1015 1010 ,
1340 1250  1170 1100 1050  1000 950 910 880  860 840 820 800 790 780
775 770 760 ,
1070 1000 940 880 840 800 770 740 710 690 670 660 650 640 630 625 620
615 ,
900 840 780 740 700 670 640 610 590 580 560 550 540 530 525 520 515
510 ,
...
```

The values in bold are used to demonstrate the results.

- A TSV subcircuit definition in SPICE format in R or RC models. Inductance is not supported for digital designs. This example uses an RC subcircuit definition.
- A LEF/DEF design.
- A valid top level LEF/DEF rule file. For more information refer to "Performing Digital Extraction With Calibre xACT".

Procedure

1. The file *tsv_rc.subckt* contains an RC subcircuit definition for a TSV.

```
.subckt     tsv_rc    top    bottom
            R1        top    n2       5555
            R2        n2     bottom   6666
            C         n2     sub      7777f
.ends
```

2. Verify the top-level rule file, *top_lefdef.rules*, contains the following specification information.

   The layout input statements are used to specify the LEF/DEF input. For example:

```
LAYOUT PRIMARY "DIGITAL_interposer"
LAYOUT PATH "./input/tsv.lef"
            "./input/ubm.lef"
            "./input/DIGITAL_Interposer.tlef"
            "./input/DIGITAL_Interposer.def"
INCLUDE ./RC_interposer_LEFDEF/RC_interposer.lefdef.map.svrf
...
```

Specify the extracted netlist output in SPEF format using layout names with:

```
PEX NETLIST "tsv_netlist.spef" SPEF LAYOUTNAMES MAPNAMES NOINSTANCEX
CLAYER RLAYER RLOCATION RLENGTH RWIDTH
```

3. To include the TSV subcircuit in the extracted netlist add the PEX Netlist 3DIC statement to the *top_lefdef.rules* file.

   ```
   PEX NETLIST 3DIC tsv_rc FILE "./inputs/tsv_rc.subckt" SUBNODE sub
   ```

   where *tsv_rc* is an LVS layer or a calibrated layer name, and *sub* is TSV circuit middle node.

4. To also include the couplings between the TSVs that are within a distance of 325 microns of each other, add the PEX 3DIC Coupling statement to the *top_lefdef.rules* file.

   ```
   PEX 3DIC COUPLING tsv_rc DIGITAL 7.0e7 MAXDISTANCE 325
   ```

   When the **DIGITAL** keyword is specified, the coupling capacitance between the TSVs is extracted based on an effective capacitance table. The parasitic values are provided as a function of spacing between the TSVs and the operating frequency.

5. Save and close the *top_lefdef.rules* file.

6. Run Calibre xACT to extract RC parasitic effects, insert a TSV circuit, and generate the netlist.

   ```
   calibre -xact -rc top_lefdef.rules
   ```

Results

If there are no errors, the directory should contain the SPEF netlist, *tsv_netlist.spef*.

Examine the contents of the netlist. The netlist should contain the inserted TSV information. The resistance and capacitance values should match those from the subcircuit specified in *tsv_rc.subckt*. For example:

```
...
*CAP
...
23 *31:88 *31:22 7777
24 *31:22 *30:17 1.1124e+06
...
*RES
...
53 *31 *31:88 6666 // $x=182.5 $y=229.0 $x2=182.5 $y2=229.0
54 *31:88 *31:84 5555 // $x=182.5 $y=229.0 $x2=182.5 $y2=224.875
...
```

The netlist should also include the TSV coupling information:

```
...
*CAP
...
23 *31:88 *31:22 7777
24 *31:22 *30:17 1.110e+06
25 *31:22 *33:20 1.011e+06
26 *31:22 *34:24 1.011e+06
27 *31:22 *35:24 1.112e+06
28 *31:22 *37:19 1.011e+06
...
```

The capacitance values for TSV coupling were obtained from the effective tsv_capacitance table for the spacing value of 160 microns.

Netlisting a Design Without Parasitics

To validate Calibre xACT output you might create a netlist showing only the intentional devices. This can be done with or without the usual parasitic extraction step. The rule file still contains rules for parasitic extraction.

Creating an Ideal Netlist from the Command Line 136
Creating an Ideal Netlist from Calibre Interactive 137

Creating an Ideal Netlist from the Command Line

Netlists without parasitics can be created for any of the flows.

Prerequisites

- A valid PEX rule file for this layout including a PEX Netlist Simple statement specifying the output format.
- Layout database that is LVS-clean.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Build the database of intentional devices:

   To use source names, run Calibre nmLVS-H before Calibre xACT:

   ```
   calibre -lvs -hier -spice $svdb_dir/top_cell.sp rules
   ```

   To use layout names, skip step 1 and go directly to step 2.

2. To generate the netlist, run Calibre xACT:

   ```
   calibre -xact -simple rules
   ```

Results

The transcript for a successful Calibre xACT run concludes with a count of errors and warnings. For example:

```
                CALIBRE xACT WARNING / ERROR Summary
---------------------------------------------------------------------
                        xACT Warnings  =  2
                          xACT Errors  =  0

=====================================================================
```

If there are no errors, the directory also contains a file with the name you specified in the PEX Netlist Simple statement.

Creating an Ideal Netlist from Calibre Interactive

Create an ideal netlist from Calibre Interactive with a working SVRF file, and specifying only the settings particular to producing non-parasitic netlists.

Prerequisites

- A valid PEX rule file for this layout.
- Layout database that is LVS-clean.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

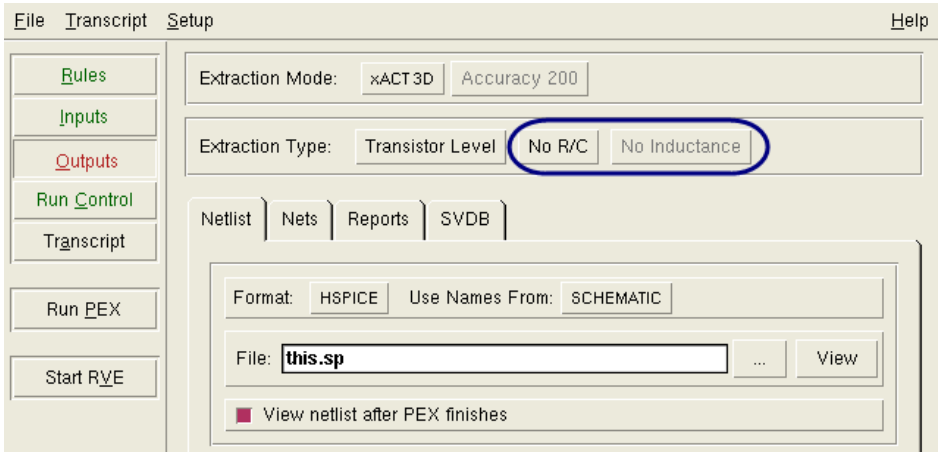
Procedure

1. Start the PEX interface in Calibre Interactive.

   ```
   calibre -gui -pex
   ```

2. Load a runset or rule file.
3. Set the extraction type to extract no parasitics - No R/C and No Inductance. (Any of the levels can be used; this figure happens to show transistor level.)

Figure 10-9. Setting for No Parasitics



4. Set other controls as needed.
5. Click **Run PEX** to produce the netlist.

Results

Check the Transcripts pane to verify the run completed with no errors. If you have selected "View netlist after PEX finishes" in the Outputs pane, a text viewer appears with the generated netlist loaded.

Backannotating Parasitics to a Source Netlist

To add layout parasitics to a source netlist, you must use the source-based flow. This calculates parasitic effects based on the layout, matches the layout devices to source devices, and attempts to map parasitics. If your source and layout are not close matches (for example, two source resistors are laid out as 24), you will get nodes labeled as "_noxref".

Note

This method is not supported for full hierarchical extraction.

Backannotating from the Command Line Using Calibre xACT 3D Direct Netlisting . 138
Backannotating from the Command Line Using the Calibre xACT 3D PDB Flow . . . 139
Backannotating from Calibre Interactive 140

Backannotating from the Command Line Using Calibre xACT 3D Direct Netlisting

Backannotate parasitics to a source netlist from the command line using the Calibre xACT 3D PDB direct netlisting flow.

Prerequisites

- A valid PEX rule file for this layout.
  - If you need to change pin order, model names, or parameters, use PEX BA Mapfile.
  - The PEX Netlist statement should indicate SOURCEBASED.
- Layout database that is LVS-clean.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Build the database of intentional devices. For backannotation, you must use Calibre nmLVS:

   ```
   calibre -lvs -hier -spice $svdb_dir/top_cell.sp rules
   ```

2. Extract parasitic effects and generate the netlist with Calibre xACT 3D direct netlisting.

   ```
   calibre -xact -3d -parasitic_switch rules
   ```

   where ***parasitic_switch*** indicates the type of parasitics to extract. For example, -rc for distributed resistance and capacitance.

Results

The transcript for a successful Calibre xACT 3D run concludes with a count of errors and warnings. For example:

```
                  CALIBRE xACT WARNING / ERROR Summary
------------------------------------------------------------------------
                          xACT Warnings   =   0
                            xACT Errors   =   0

========================================================================
```

If there are no errors, the directory also contains the netlist you specified in the PEX Netlist statement. The exact number of files depends on the output format.

Backannotating from the Command Line Using the Calibre xACT 3D PDB Flow

Backannotate parasitics to a source netlist from the command line using the Calibre xACT 3D PDB extraction flow.

Prerequisites

- A valid PEX rule file for this layout.
  - If you need to change pin order, model names, or parameters, use PEX BA Mapfile.
  - The PEX Netlist statement should indicate SOURCEBASED.
- Layout database that is LVS-clean.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Build the database of intentional devices. For backannotation, you must use Calibre nmLVS:

   ```
   calibre -lvs -hier -spice $svdb_dir/top_cell.sp rules
   ```

2. Extract parasitic effects with Calibre xACT 3D.

   ```
   calibre -xact -3d -pdb -parasitic_switch rules
   ```

   where ***parasitic_switch*** indicates the type of parasitics to extract. For example, -rc for distributed resistance and capacitance.

3. Generate the netlist.

   If ***parasitic_switch*** included resistance, use the following:

   ```
   calibre -xact -fmt -all rules
   ```

To produce a netlist that only contains lumped capacitance, use the following:

```
calibre -xact -fmt -c rules
```

Results

The transcript for a successful Calibre xACT 3D run concludes with a count of errors and warnings. For example:

```
                  CALIBRE xACT WARNING / ERROR Summary
------------------------------------------------------------------------
                         xACT Warnings   =  2
                           xACT Errors   =  0

========================================================================
```

If there are no errors, the directory also contains the netlist you specified in the PEX Netlist statement. The exact number of files depends on the output format.

Backannotating from Calibre Interactive

Backannotate parasitics to a source netlist from Calibre Interactive.

There may be changes for your specific layout viewer or simulation flow. Backannotation is only available with Calibre xACT 3D. Extraction performed from Calibre Interactive follows the PDB extraction flow.

Prerequisites

- A valid PEX rule file for this layout.
  - The PEX Netlist statement should indicate SOURCEBASED.
- Layout database that is LVS-clean.
- For more information refer to “Prerequisites for Performing Parasitic Extraction”.

Procedure

1. Start the PEX interface in Calibre Interactive.
2. Load a runset or rule file.
3. Specify the extraction type and other settings. Generally, for backannotation you use names from the schematic.
4. Click **Run PEX** to produce the netlist.

Results

Check the Transcripts pane to verify the run completed with no errors. If you have selected “View netlist after PEX finishes” in the Outputs pane, a text viewer appears with the generated netlist loaded.

Related Topics

Getting Started: Calibre xACT 3D Parasitic Extraction Using Batch Mode

Types of Extraction

Running Gate-Level Extraction

Generating a Capacitance Summary Report

The capacitance summary report lists total capacitance and the ratio of coupling capacitance for all nets. It is controlled solely by the SVRF statement, PEX Report Netsummary. If the statement is present in the SVRF file, a run will produce the capacitance summary report regardless of other settings.

The report can be set up to provide details on only specific nets or cells. It can divide capacitance effects by cells, or report capacitance from top-level interconnect only.

Prerequisites

- A valid PEX rule file for this layout that includes a PEX Report Netsummary statement.
- A layout database that is LVS-clean.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Set up the PEX Report Netsummary information to provide the needed details.
2. Perform parasitic extraction for capacitance. (The extraction may also include resistance or induction effects.)

Results

Look for a file with the name specified in the PEX Report Netsummary statement. If no capacitance data was extracted, the report ends with "No meaningful analyzed data found."

Related Topics

Getting Started: Calibre xACT 3D Parasitic Extraction Using Batch Mode

Getting Started: Calibre xACT 3D Parasitic Extraction Using Calibre Interactive

Net-to-Net Coupling Capacitance Report

The coupling capacitance report shows the amount of capacitance between specific pairs of nets and calculates how much this coupling capacitance represents of the total coupling on each net.

Reporting Coupled Capacitance from the Command Line **142**
Reporting Coupled Capacitance from Calibre Interactive **143**

Reporting Coupled Capacitance from the Command Line

Create a coupled capacitance report from the command-line for any run that includes CC parasitics.

Prerequisites

- A valid PEX rule file for this layout including a PEX Report Coupling Capacitance statement.
- Layout database that is LVS-clean.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Set up the PEX Report Coupling Capacitance information to provide the needed details.
2. Run parasitic extraction for coupled capacitance using -rcc for the parasitic flag.

Results

The transcript for a successful Calibre xACT 3D run concludes with a count of errors and warnings. For example:

```
                  CALIBRE xACT WARNING / ERROR Summary
------------------------------------------------------------------------
                         xACT Warnings   =   2
                           xACT Errors   =   0

========================================================================
```

If there are no errors and the extraction included coupled capacitance data, the directory also contains a file by the name you specified in the PEX Report Coupling Capacitance statement.

If the file is not present, check the transcript for warnings regarding PEX REPORT COUPLING CAPACITANCE.

Related Topics

Getting Started: Calibre xACT 3D Parasitic Extraction Using Batch Mode

Reporting Coupled Capacitance from Calibre Interactive

Create a coupled capacitance report using Calibre Interactive for any run that includes CC parasitics. Extraction performed from Calibre Interactive follows the PDB extraction flow.

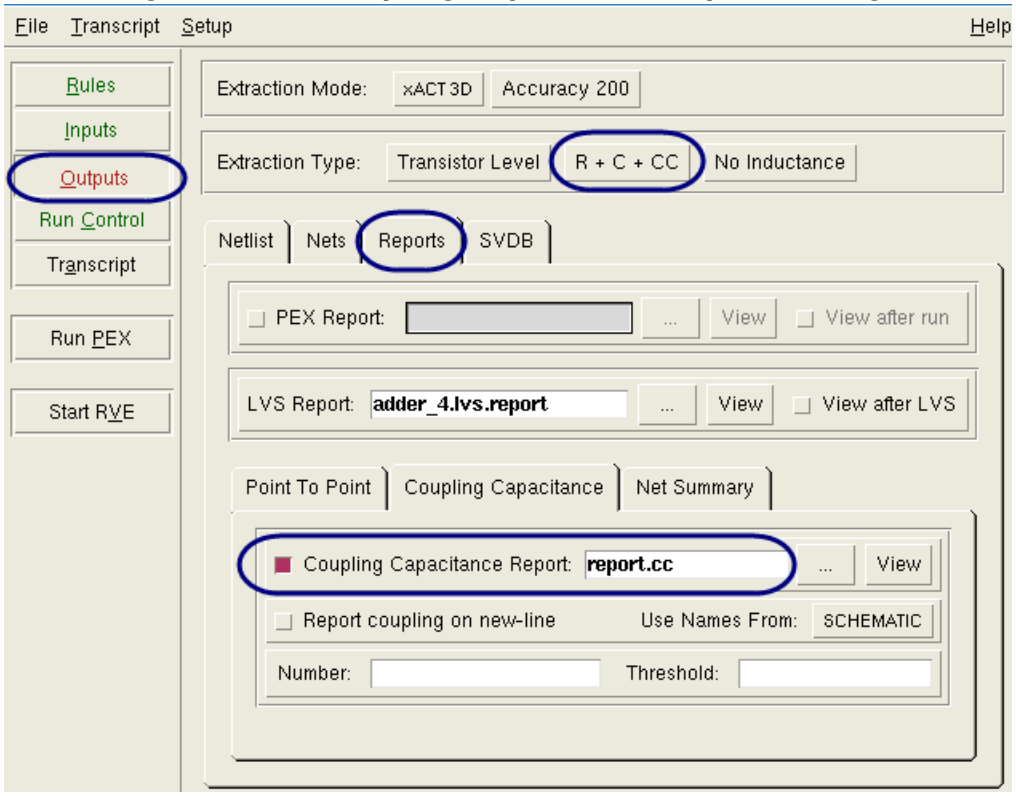
Prerequisites

- A valid PEX rule file for this layout.
- Layout database that is LVS-clean.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Start the PEX interface in Calibre Interactive.
2. Load a runset or rule file.
3. In the Outputs pane, set Extraction Mode to xACT 3D and set Extraction Type to R + C + CC.
4. Under the **Reports** tab, select the **Coupling Capacitance** tab. Select Coupling Capacitance Report.

Figure 10-10. Coupling Capacitance Report Settings



5. Provide a report name. (There is no default.) If needed, set the other options:
   - Number sets the maximum number of nets to include in the report. Only the most tightly coupled pairs are reported.
   - Threshold sets the capacitance threshold in farads below which the tool should not report.
6. Set other controls as needed.
7. Click **Run PEX** to produce the report (and netlist).

Results

Check the Transcripts pane to verify the run completed with no warnings about PEX Report Coupling Capacitance or errors.

If there are no errors, the directory contains the report file along with the netlist.

Related Topics

Getting Started: Calibre xACT 3D Parasitic Extraction Using Calibre Interactive

Point-to-Point Resistance Reports

Point-to-point resistance reports output the resistance along a single layout net. The points can be anywhere along the net. The Calibre Interactive interface allows you to turn the report on or off, but the contents of the report are controlled through a separate text file.

Note

It is easier to identify points if you label nets in the layout. Also, you will get more accurate results if you turn off PEX Reduce TICER when creating this report.

Reporting Net Resistance from the Command Line 145
Reporting Net Resistance from Calibre Interactive 146

Reporting Net Resistance from the Command Line

Create a net resistance report from the command-line using the Calibre xACT 3D PDB extraction flow.

Prerequisites

- A valid PEX rule file for this layout including a PEX Report Point2Point statement.
- Layout database that is LVS-clean.
- For more information refer to "Prerequisites for Performing Parasitic Extraction".

Procedure

1. Set up the control file for the report. Each entry should be of the form:

   ```
   RESISTANCE Netname Location Netname Location
   ```

   where:

   - *Netname* is the *layout* name of the net, and is same for both entries. (Resistance cannot be measured across devices.)
   - *Location* is one of the following:

PIN *name signature*	Name of the resistor pin
PORT | PROBE *p_name*	Name of the port or probe
COORD *x y layer*	Coordinates and layer name to be used to find the closest node on the desired net. The coordinates are in database units (DBU) or user specified (UNIT_LENGTH).

The file can contain multiple entries. Each should be on a separate line.

2. Run parasitic extraction for resistance using any of the -r switches for the parasitic flag.

Results

The transcript for a successful Calibre xACT run concludes with a count of errors and warnings. For example:

```
                  CALIBRE xACT WARNING / ERROR Summary
-------------------------------------------------------------------------
                         xACT Warnings   =  2
                           xACT Errors   =  0

=========================================================================
```

If there are no errors, the directory also contains a file by the name you specified in the PEX Report Point2Point statement. If an entry in the report says “No analyzed resistors on net” the points in the entry have insignificant resistance or the net was not extracted.

Related Topics

Getting Started: Calibre xACT 3D Parasitic Extraction Using Batch Mode

Reporting Net Resistance from Calibre Interactive

Create a net resistance report using Calibre Interactive for any run that includes parasitic resistance. Extraction performed from Calibre Interactive follows the PDB extraction flow.

Prerequisites

- A valid PEX rule file for this layout.
- Layout database that is LVS-clean.
- For more information refer to “Prerequisites for Performing Parasitic Extraction”.

Procedure

1. Set up the control file for the report. Each entry should be of the form:

   ```
   RESISTANCE Netname Location Netname Location
   ```

   where:

   - *Netname* is the *layout* name of the net, and is the same for both entries. (Resistance cannot be measured across devices.)
   - *Location* is one of the following:

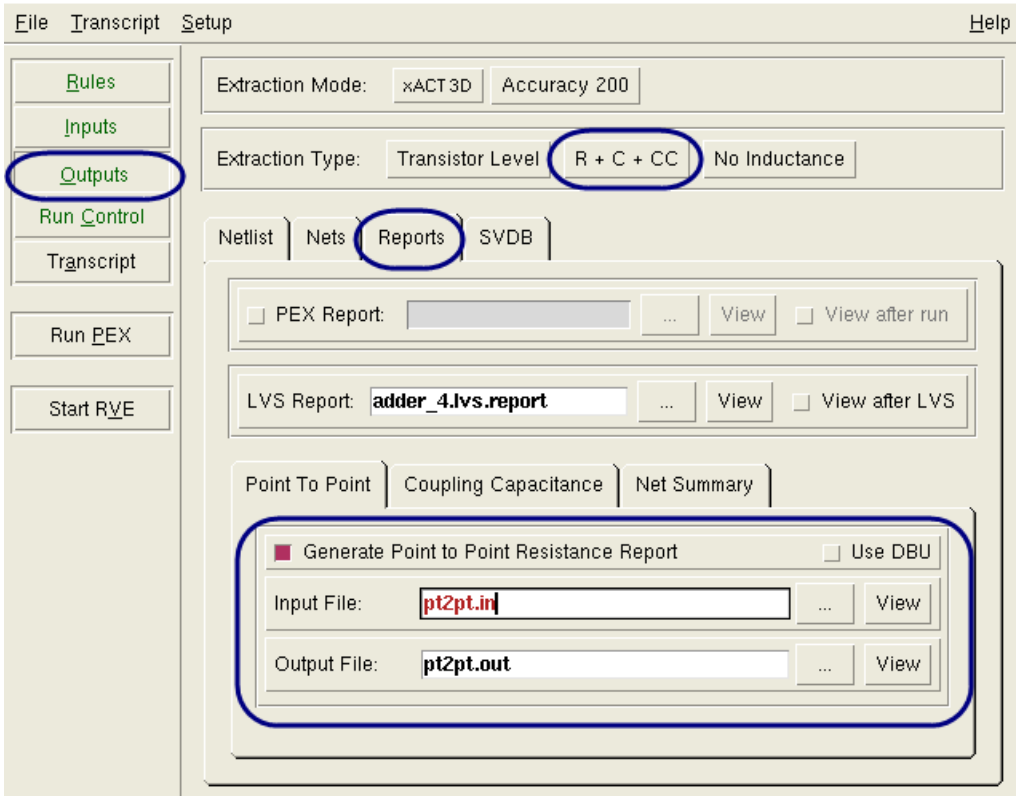
PIN *name signature*	Name of the resistor pin
PORT | PROBE *p_name*	Name of the port or probe

COORD *x y layer*	Coordinates and layer name used to find the closest node on the desired net. The coordinates are in database units (DBU) or user specified (UNIT_LENGTH).

The file can contain multiple entries. Each should be on a separate line. The PEX Report Point2Point statement in the *Standard Verification Rule Format (SVRF) Manual* has more information.

2. Start the PEX interface in Calibre Interactive.
3. Load a runset or rule file.
4. In the Outputs pane, set the extraction mode to xACT 3D and set the extraction type to R, R + C, or R + C + CC.
5. Under the **Reports** tab, select the **Point to Point** tab. Select Generate Point to Point Resistance Report.

Figure 10-11. Point-to-Point Resistance Report Settings



6. Enter the name of the control file you created in step 1 in the Input field. Enter another name in the Output field.

7. Set other controls as needed.
8. Click **Run PEX** to produce the report (and netlist).

Results

Check the Transcripts pane to verify the run completed with no errors. The directory contains the report file along with the netlist. If an entry in the report says “No analyzed resistors on net” the points in the entry have insignificant resistance or the net was not extracted.

Related Topics

Getting Started: Calibre xACT 3D Parasitic Extraction Using Calibre Interactive

Top Level Only Extraction

To extract only the top-level interconnect, list all cells instantiated in the top level in the xcell file. Use the cell names without IDs. Then run extraction using gate-level extraction. This occurs automatically with LEF/DEF layouts.

See “Running Gate-Level Extraction” on page 117 for instructions.

Chapter 11
Handling Input

Extraction flows can be customized to handle different types of input.

This chapter describes how to modify the basic extraction processes to handle the following types of variant input:

Hierarchy Control with Xcells **149**
Importing GDS Cell Views **153**
Metal Fill Modeling **155**
Modeling Multiple Ground Regions **157**
Varying Thickness with CMP Files **158**

Hierarchy Control with Xcells

When performing hierarchical extraction, you define your design's hierarchy for the Calibre xACT tool by identifying the lower-level cells in the design hierarchy.

You define this hierarchy by creating a cell correspondence list that matches source and layout names. This cell correspondence list, also known as an hcell list, is input into the Calibre nmLVS-H tool using the "-hcell" switch. An xcell list is derived from the hcell list; the xcell list can only contain cells that were named in the hcell list. The xcell list is specified as part of the Calibre xACT 3D invocation.

The Xcell List

An xcell list is an ASCII file that you create. The cells listed in the xcell list define the design hierarchy and designate the cells the extraction will preserve. Cells not in the xcell list are flattened into the top circuit.

During Parasitic Database (PDB) creation, the Calibre xACT 3D tool caches a copy of the xcell list you input into the tool, even if the xcell list is empty. If you supply an empty xcell list, or if none of the cells in the provided xcell list are found in the layout, the Calibre xACT 3D tool issues the following warning message at the conclusion of the PDB stage:

WARNING: Could not match any layout cell names against the XCELL file: *xcell_file_name*

An empty xcell list effectively produces a transistor-level (flat) PDB and, consequently, a flat netlist or report.

Calibre nmLVS-H and Calibre xACT 3D Cell List Compatibility

When performing source name extraction, you must provide the Calibre nmLVS-H tool with an hcell list that includes the cells you will use as xcells with the Calibre xACT 3D tool. The hcell list uses the following format:

```
layout_name source_name
```

With the Calibre nmLVS-H tool, you indicate the cell list using the "-hcell" switch. For example, the following command line invocations demonstrate a typical Calibre nmLVS-H and Calibre xACT tool run using the same cell list throughout, called hcell_list:

```
calibre -lvs -hier -spice svdb/design.sp -hcell hcell_list rules

calibre -xact -3d -rc -xcell hcell_list rules

calibre -xact -fmt -all rules
```

The cell list performs the following functions in each of the tools:

- **Calibre nmLVS-H tool** — Maps the layout's cell names to their corresponding source's cell names when you perform source name extraction. When invoking the Calibre nmLVS-H tool, you specify the hcell list by using the "-hcell hcell_list" switch; see "calibre -lvs" for details. See "Hcells" in the *Calibre Verification User's Manual* for information on hcells.
- **Calibre xACT 3D tool** — Defines the design hierarchy for global net extraction. You create an xcell list based on the hierarchy you want, and the Calibre xACT 3D tool analyzes and extracts the hierarchy.

Although an hcell list can be used as an xcell list, xcell lists have additional options. If you add xcell options to your hcell file, be sure to rename the file since these options will not work for Calibre nmLVS-H. Xcell entries can use wildcards to match multiple cells.

Xcell List Format

An xcell list can have either of two formats. You can use the hcell list format:

```
layout_name source_name
```

or you can use the xcell-specific format:

```
layout_name [source_name] [ -PCDEF [-INTRINSIC] [-CONLY]
                          | -P [-GDS gds_macro]
                          | -FILL file.gds cellname
                          | -I
                          | -NOBLOCK ] [//Comment ]
```

In both formats, each cell is on its own line and can only appear once. If the xcell file follows the hcell list format, you can use the same file for both source name extraction and parasitic

extraction. The second column (the source column) is ignored during parasitic extraction. The second, xcell-specific format can only be used for parasitic extraction and will not work as an hcell list for source name extraction.

The flags in the xcell-specific format can appear in either upper or lower case. Table 11-1 provides more information on each flag.

Table 11-1. Xcell Flags

Flag	Usage
-PCDEF	Indicates the cell is a pcell. The pcell contents are not extracted. The intentional device represented by the pcell will be netlisted. The parasitics outside the pcell boundary will be extracted. For nets that exit an xcell through an xcell pin, the coupling capacitance between nets outside the xcell and nets inside the xcell will be extracted and netlisted; this includes connections between the outside net and the xcell participating in the relationship. For nets totally enclosed within the xcell (they do not exit the xcell through any xcell pin), coupling capacitance is computed to the outside nets and is netlisted on the outside net dropped to ground. You can also use -PCDEF -I to generate an empty .subckt definition for the xcell, by specifying the xcell in an LVS Box statement and specifying LVS Preserve Box Cells YES in your rule file.
-INTRINSIC	Used only in the PCDEF flow. Specifies that the pin intrinsic capacitance will be ignored for the device layers within the cell. Can only be specified in conjunction with -PCDEF.
-CONLY	Used only in the PCDEF flow. Indicates that the SPICE model for the cell only contains parasitic capacitance. This means only parasitic resistance is extracted for the cell. These resistors are identified in the extracted netlist with the property called $pcell_res=1. Can only be specified together with -PCDEF.
-P	Indicates the cell is a primitive. The contents are not extracted. In gate-level extraction, all cells are treated as primitives.
-GDS *gds_macro*	Used only with LEF/DEF designs. Indicates that a cell is a GDS macro. Can only be specified in conjunction with -P.
-FILL *file.gds cellname*	Used only with the top cell of a LEF/DEF design. Specifies that metal fill is in a separate file, *file.gds*, in cell *cellname*.
-I	Indicates the cell is an ideal xcell. The contents are not extracted, but are written to the netlist. Only cells with the -I flag are treated as ideal xcells.
-NOBLOCK	Indicates the cell is not a pcell. The contents are still extracted and written to the netlist. Used only in cases where the cell name is part of a wildcard list and should be treated differently. Cannot be specified with -I.

The following is a valid example of an xcell file:

```
// layout_name source_name flag
NOR          NOR      -I      //treated as an ideal cell
NAND         NAND     -P      //use gds layout for extraction
INV          INV              //handling depends on the extraction type
NMOS         NMOS     -PCDEF //treated as a pcell
NMOS         PMOS     -PCDEF -CONLY //only extract resistance for the pcell
```

Note

The Calibre xACT tool disregards any Hcell statements you specify in the SVRF rule file. You must include any cell you identify with the Hcell statement in the xcell file *or* use the PEX Xcell statement.

Wildcards in Xcell List

You can use an asterisk (*) as a wildcard in the xcell list, to ease the task of creating an xcell file.When wildcards are used in xcell specifications, a cell name could match multiple xcell specifications. By default all matching xcell specifications are applied to the cell.

If the specifications use conflicting flags, Calibre xACT 3D generates an error and extraction stops. For example, the following wildcard xcell specifications generates an error:

```
pmos_rf* -I -P          // Treat as ideal xcell and primitive
pmos* -PCDEF            // Treat as pcell
```

You can use the PEX Xcell Precedence statement with the BEST keyword to resolve such conflicts. The BEST keyword finds the best matching wildcard xcell name and applies only the flags for that specification. All other matches are ignored. With the BEST keyword, cells whose names match pmos_rf* would be treated as primitive xcells and the xcell specification to indicate the cell is a pcell (-PCDEF) is not applied.

Caution

Use the asterisk (*) wildcard in the xcell list with discretion as it could potentially increase runtime. It is recommended to specify as much of the name as possible before using a wildcard.

Table 11-2. Wildcards in an Xcell List

Xcell file without wildcards	Same Xcell file with wildcards
pmos_rf1	pmos_rf*
pmos_rf2	nmos_rf*
pmos_rf3	
pmos_rf7	
...	
nmos_rf1	
nmos_rf2	
nmos_rf3	
nmos_rf5	
...	

Tips For Choosing Xcells for Full Hierarchical Extraction

For most GDS-based extractions, you can shorten extraction time by using only a subset of the LVS hcells as Calibre xACT xcells. The cells you specify affect performance and accuracy.

- Select cells that occur multiple times. The more times a cell appears, the faster extraction will be.
- Do not specify densely packed cells as xcells. In full hierarchical extraction, coupling between xcells is not modeled. For example, in a memory design, specify the cell containing an array rather than the cell containing a single bit.
- Do not specify cells that overlap another xcell. The contents in the overlapped area will be counted more than once.
- Do not specify cells with feedthrough nets as xcells, unless you are formatting the netlist in extended DSPF with the HSIM keyword set.

Importing GDS Cell Views

Import GDS descriptions of macrocells and standard cells with Calibre xACT, and use their GDS geometries in place of their LEF equivalents. The LEF and GDS cells must be geometrically identical to ensure accurate results from this flow.

Prerequisites

- Your LEF design files and GDS macrocell file(s).
- A layer mapfile to map the metal fill layer numbers onto equivalent DEF layer names. See “GDS and OASIS Metal Fill Extraction” for a description of the mapfile format.

- Your Calibre xACT rule file.

Procedure

1. To import GDS cells, include lines in your Calibre xACT SVRF rule file similar to the following example:

   ```
   PEX XCELL "*" PRIMITIVE GDS "./design/lef/gsc45nm.lef" MAP "map.txt"
   PEX XCELL "*" PRIMITIVE GDS "./design/lef/PADS.lef"
   PEX XCELL "*" PRIMITIVE GDS "./design/lef/myram.lef"
   PEX XCELL "*" PRIMITIVE GDS "./design/lef/dac6_op2.lef"
   PEX XCELL "*" PRIMITIVE GDS "./design/lef/ADC_5bit_sc_5.lef"
   ```

   Note that it is only necessary to specify the mapfile once in your rule file. Calibre xACT uses the same mapfile for GDS cell views as it does for GDS fill, so if you are also using a PEX Xcell ... FILL statement in the same rule file, then you only need to provide the path to the mapfile once.

2. Create a layer map file to map the metal fill layer numbers onto equivalent DEF layer names. The map file format consists of nine columns in the following order:

   - column 1: *LEFDEFlayername*
   - column 2: *GDSdefaultlayernumber*
   - column 3: *GDSdefaultlayerdatatype*
   - column 4: *GDSfilllayernumber*
   - column 5: *GDSfilllayerdatatype*
   - column 6: *GDStextlayernumber*
   - column 7: *GDStextlayerdatatype*
   - column 8: *GDScellboundarylayernumber*
   - column 9: *GDScellboundarylayerdatatype*

   For example:

   ```
   metal1 31 0 31 1 31 99 108 0
   metal2 32 0 32 1 32 99 108 0
   metal3 33 0 33 1 33 99 108 0
   ....
   ```

   If you do not know the metal fill layer numbers for your fill data, you can find them by visually inspecting the data using Calibre DESIGNrev.

   It is possible to map more than one GDS fill layer number to a single LEF/DEF layer. For example, if you want to combine GDS layers 31.0 and 31.20 and map them onto

your M1 layer, repeat the line in the mapfile for the LEF/DEF layer(s) that you want to map:

```
M1 31 0 31 1 31 99 108 0
M1 31 20 31 1 31 99 108 0
```

If you omit columns 8 and 9 from the mapfile, Calibre xACT uses a default layer 108.0 for cell boundary information. If no geometries are found on that layer, Calibre xACT uses the extent of the cell geometries on layers specified in the layer map file. The bottom left-most macro pin is frequently located at an offset from the origin. When this occurs, the cell boundary is calculated incorrectly causing the cell to be offset when imported into Calibre xACT and the pin locations not to match. To avoid this problem, specify a cell boundary layer and data type if it is different than the default. You can do this by specifying two additional columns in the Calibre xACT GDS mapfile. For example, the following mapfile entry:

```
M1 10 0 10 1 10 99 110 0
```

defines the following layers and data types:

```
LEF/DEF layer: M1
GDS cellview layer: 10.0
Fill layer: 10.1
Text layer: 10.99
Cell boundary layer: 110.0
```

3. If the GDS macrocells include text annotations on the layer that you specified in the mapfile, then the cell ports are named accordingly in Calibre xACT.

   This results in coupling between nets in proximity to the cell and shapes inside the cell. This is represented in the netlist as coupling capacitances between the two associated nets, rather than being lumped to the net's total capacitance.

4. If the GDS macrocells do not include text annotations, then the cells inherit port names from the corresponding LEF macro definitions.

Metal Fill Modeling

By default, the Calibre xACT 3D tool treats floating signal nets as fixed (grounded) and does not extract them. However, floating nets are not ignored when calculating the capacitance of non-floating nets.

Assuming that floating signal nets are grounded results in some error for floating nets like metal fill. You can control how metal fill, and floating nets in general, are treated during extraction with the PEX Extract Floating Nets statement. In the Calibre Interactive interface, this is accessed through **PEX Options > Netlist > Format > Extract floating nets**.

To model the coupling resulting from metal fill, run parasitic extraction in -rc or -rcc mode and specify the SVRF statement as follows:

```
PEX EXTRACT FLOATING NETS ALL
```

This allows the metal fill to float, which gives a better approximation of real conditions. This may increase extraction time and the netlist size. If the netlists are too large to simulate, use PEX Reduce CC for targeted reduction.

For lumped capacitance extraction, which treats all neighbor nets as grounded, you need to extract the metal fill nets so that they can be simulated. To extract nets associated with the metal fill, set the statement as follows:

```
PEX EXTRACT FLOATING NETS GROUNDED
```

Floating nets and signal nets are extracted in the same manner.

Note

If the metal fill has not been added yet, set the target density using the PEX Density Estimate statement. In-die tables must be included in the calibrated rule file for the process technology.

Many metal fill configurations are supported. Figure 11-1 shows an example of simple square or rectangular fill between nets drawn on the same metal layer. Figure 11-2 shows multi-layer fill and nets. Figure 11-3 also shows a non-square fill with multiple nets.

Figure 11-1. Simple Metal Fill on a Single Layer

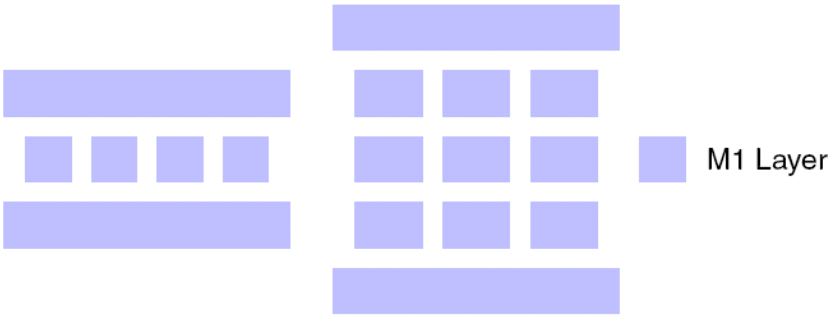


Figure 11-2. Metal Fill on Multiple Layers With Multiple Nets

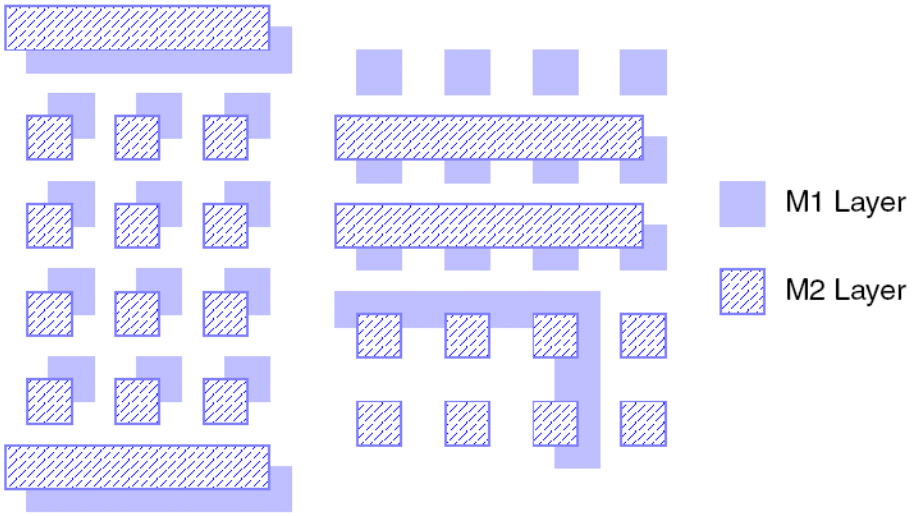
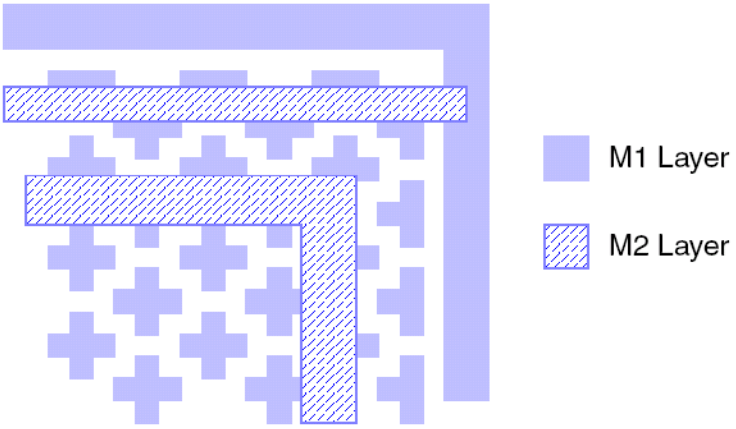


Figure 11-3. Non-Square Fill With Multiple Nets



For a more general discussion of floating nets, see "Ignore or Extract Floating Nets".

Modeling Multiple Ground Regions

Modeling multiple ground regions is performed when you need to reflect multiple ground regions of a design in your simulation, for example, when performing substrate noise analysis.

The steps result in ground regions "above" a global ground. When you extract parasitics, intrinsic capacitance goes to the named ground regions; grounded coupled capacitance goes to the global ground.

Note

You cannot model multiple ground regions with -rc extraction.

Prerequisites

- A design file with a separate layer for ground regions. Typically, this is your p-well layer.
- A complete SVRF rule file including statements for calculating intrinsic capacitance. These are a normal part of the foundry-supplied calibrated rules.

Procedure

1. If it does not already exist, create a layer with shapes for the different ground regions in your layout. Different regions can be on different layers.
2. In the SVRF rule file, verify all ground layers have connectivity. (If you are using a well layer, it should already have connectivity.) The layer name must be in a Connect statement. For example:

   ```
   CONNECT analog_regions
   ```

   You can also connect related regions to each other with a Virtual Connect Name statement. For example:

   ```
   VIRTUAL CONNECT NAME "digital_regions?"
   ```

3. Add PEX Ground Layer to the SVRF rule file. For example:

   ```
   PEX GROUND LAYER analog_regions digital_regions
   ```

4. To model ground regions not otherwise connected to a signal net, specify the PEX Extract Floating Nets statement in the rule file. You can select how the parasitic capacitance is calculated based on the GROUNDED, ALL, or REDUCED parameters for this statement.
5. Run extraction as usual. Distributed RC extraction and netlisting are not supported with multiple grounds, but all other types of parasitic models are. When running full hierarchical extraction, the ground regions are only reflected within the cells that contain the region-defining shapes.

Varying Thickness with CMP Files

If you have chemical mechanical polishing (CMP) models, such as CMPA, VCMP, or the database produced by Praesagus' Copper Prediction and Verification software, you can use its calculations for conductor thickness. The CMP data overrides all other methods of calculating thickness, such as PEX Thickness EQN or PEX Table. The Calibre xACT interpreter converts thickness values into the same units used by the extraction.

Because CMP data may not model all layers, you must still provide a process description by means of a technology file. When the CMP data and the process description both describe a layer, the CMP values are used.

Prerequisites

- Before you begin, you need a text version of the CMP data. For information on producing a text version of the database, see the documentation for the CMP modeling software.
- Make sure the layer names in the CMP data and the SVRF rule file are the same. Layers are matched by name and are not case sensitive. (For VCMP, the Calibre software treats all layer names as lowercase when looking for the corresponding file.)
- In order to use CMP data for conductor thickness, the extraction model must have a RHO_T or RSH_T table and a Thickness table for each conductor layer. For more information on the xCalibrate Rule File generator and table syntax, refer to the *xCalibrate Batch User's Manual*.

Procedure

1. Use the PEX CMP Mode statement in your extraction rule file to specify the use of CMP data during extraction.
2. Proceed with extraction using your usual methods. The CMP data can be used with all extraction modes.

Results

Calibre xACT uses the CMP thickness variation files for capacitance calculations, and RHO-based or RSH-based resistance calculations in place of the models defined in the calibrated rule files.

If you get an error message about zero or negative thickness, check the following:

- Does the conductive layer have the same name (no misspellings) in both the SVRF file and the CMP data?
- Does the text file show a zero or negative value for the layer? In a Praesagus text file, the layer thickness is given as the last two values on a line that begins with the layer name.
- Is a layer used in the SVRF file missing in both the CMP data and the technology file?

If the run seems to be using the wrong values for layer thickness, check that all layer names are distinct regardless of case. Because the file parser does not flag duplicate names, when layer names differ only by capitalization the wrong layers may be matched.

Chapter 12
Tuning Extraction

The basic extraction procedures can be fine-tuned to refine the analysis and extraction phase.

Calibre xACT Processing Control 161
Extracting Net Resistance By Layer 162
Device Extraction Without Parasitics 163
Select-Net Extraction with Calibre xACT and Calibre xACT 3D 164
Select-Layer Extraction with Calibre xACT and Calibre xACT 3D 166
Extracting Particular Nets with the Calibre xACT 3D PDB flow 167
Exclusion of Power and Ground Nets 168
Coupling Capacitance Output Control 169
Grounding Coupled Capacitors 169
Ignore or Extract Floating Nets 169
Resistance Extraction and PERC 171

Calibre xACT Processing Control

Calibre xACT runs all stages of extraction including device recognition, parasitic calculations, and netlisting in one step. By default, it performs R-coupled-C (RCC) extraction and generates a single netlist. Use Calibre xACT 3D direct netlisting to activate the fast field solver technology for capacitance extraction.

To customize Calibre xACT processing, use the PEX XACT Control SVRF statement. This statement controls the following:

- Specifies how one or more nets are handled during parasitic extraction with a directive called EXTRACT_CONTROL.
- Specifies how netlisting is performed with a directive called NETLIST_CONTROL.

These controls can be defined in a text file or directly in your top-level control file using special syntax; see the PEX XACT Control statement. When this statement is specified, Calibre xACT runs parasitic extraction and netlisting using the settings defined by the statement. If EXTRACT_CONTROL is defined, then extraction uses the specification in EXTRACT_CONTROL. If NETLIST_CONTROL is defined, then netlist generation uses the specifications in NETLIST_CONTROL. Settings specified in PEX XACT Control statement override the global command line and some SVRF statement specifications such as PEX Netlist.

The parasitic model command line switch (-rcc, -rc, etc.) controls extraction and netlisting globally. Net-specific parasitic model selections are also made using PEX XACT Control statement. Parasitic models have the following order of precedence: RCC > RC > R, C, Cg. It is important to note that you cannot run R only extraction and then request a C only netlist; This sequence generates an error. However, you can run RCC extraction, then request a C only netlist; In this case, the capacitance data required to create the netlist has been extracted.

Related Topics

Creating a Top-level Control File

Multiple Netlist Generation

Selective Resistance Extraction

Extracting Net Resistance By Layer

Calibre xACT extracts resistance for a net on a specified layer using the PEX XACT Control SVRF statement.

Prerequisites

- You have created a top-level control file called *xact_t.rules*. See “Creating a Top-level Control File” on page 27 for an example.
- You have met the requirements outlined in “Transistor-Level Extraction Setup” on page 25 for Calibre xACT.

Procedure

1. Using a text editor, create a file called *xact_extraction.control* file and enter the following statements:

```
PEX XACT CONTROL [
   EXTRACT_CONTROL {
      NET: VDD? VSS?
      MODEL: RCC
      R_LAYERS_SELECT: VIA M2 M1 VIA M3 M2 VIA M4 M3 VIA M5 M4
   }
]
```

   Save and close the file.

2. Using a text editor, add the PEX XACT Control statement to the *xact_t.rules* file as follows:

```
PEX XACT CONTROL [xact_extraction.control]
```

   In this example, the netlist name and format generated is specified with PEX Netlist.

   Save and close the file.

3. Run Calibre xACT to perform extraction:

   ```
   calibre -xact -rcc xact_t.rules
   ```

   Or optionally run Calibre xACT 3D to perform extraction using the field solver for the capacitance extraction:

   ```
   calibre -xact -3d -rcc xact_t.rules
   ```

Results

Capacitance and resistance is extracted for all nets and layers, and only resistance is extracted for vias between the metal layers on nets VDD? and VSS?.

Related Topics

Selective Resistance Extraction

Device Extraction Without Parasitics

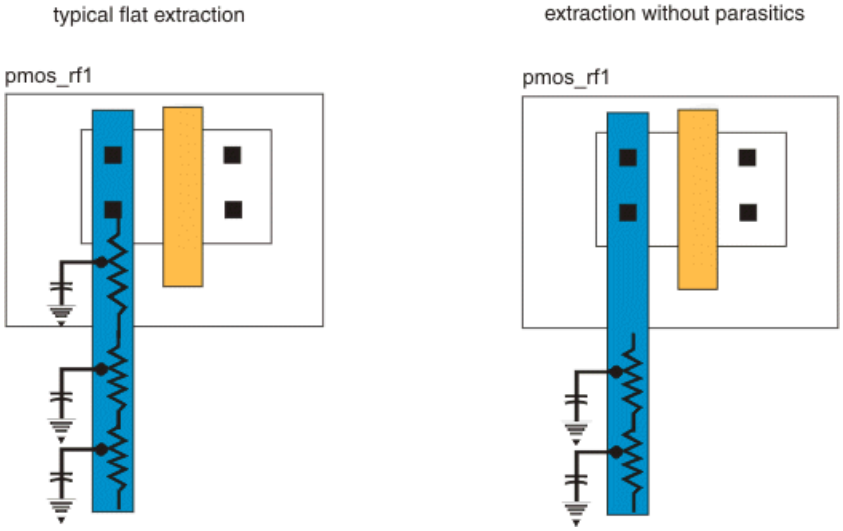
The Calibre xACT tool supports extraction of devices that do not contain parasitics in the xcells. This is useful for flows where the device models already include the parasitics. For example, when physical layout of devices is done by using device generators or parameterized cells (pcells), and the device models are parameterized to match the layout.

To ensure accurate simulation results when using parameterized models, the extraction tool must not extract parasitics inside the specified devices. Anything in the design's xcell list is treated as ideal, meaning that no parasitics are included, and flattened to the transistor level. This method of extraction is sometimes known as "gray box" extraction.

It is not the same as primitive xcells, because the netlist contains the nets for the xcell contents, which have been flattened within the xcell. All nets within the xcell will be ideal nets, that is, they will have no parasitic net models.

Figure 12-1 compares a typical transistor level extraction with a pcell extracted without parasitics. Note that this method of extraction ignores only the parasitics inside the pcell. The parasitics outside the pcell are still extracted.

Figure 12-1. Device Extraction With and Without Parasitics



Invocation

To invoke extraction of devices without parasitic net models in the xcells (ideal cells), mark the cells with a -i switch in the xcell file.

Requirements

In order for this extraction method to provide useful information, the following requirements must be met:

- The gate-level option (-xcell *xcell_file*) for Calibre xACT 3D must be set, even though the design is transistor level.
- Designs must use parameterized cells.
- The parameterized cell models must account for all of the parasitics in the cell.
- Each cell model must have an entry in the hcell file, and an identical entry in the xcell file. *All* entries will be treated as parameterized cells.

Select-Net Extraction with Calibre xACT and Calibre xACT 3D

Calibre xACT and Calibre xACT 3D can be used together to extract specific nets with field solver accuracy.

When performing select-net extraction with Calibre xACT and Calibre xACT 3D, you must perform the following operations:

1. Identify the selected nets by name using the PEX Fieldsolver Mode NETS statement in your SVRF rule file.
2. Use the "-3dselect" Calibre xACT command line switch.

If you omit the PEX Fieldsolver Mode NETS statements from your rule file and invoke the Calibre xACT tool using the "-3dselect" switch, then the tool will extract all nets with Calibre xACT.

Example 12-1. Selecting Nets with SOURCENAMES

```
SVRF: PEX FIELDSOLVER MODE 600 NETS SOURCENAMES "clk"

Command Line:    calibre -xact -3dselect -rcc rules.top
```

Explanation: When this SVRF statement is present in the rule file and -3dselect is specified on the command line, all layers on net "clk" are extracted with field solver mode 600. All other nets are extracted with Calibre xACT.

Wildcards and Search Level Specification for Nets

You can use wildcards to select nets to include or exclude when performing a parasitic extraction run. You can also specify where in the hierarchy to search for a matching net name.

- Use question mark (?) character as a wildcard matching zero or more characters in a net name.
- Use one of two mutually exclusive secondary keywords, TOPLEVEL and RECURSIVE, to specify where in the hierarchy to search for a matching name. TOPLEVEL is the default.

Example 12-2. Extracting Matching Names in Top Level

```
PEX FIELDSOLVER MODE 600 NETS LAYOUTNAMES TOPLEVEL "X0/X1/foo?"
```

Explanation: Any net in placement X0/X1 that is not ported out of that cell and has a name beginning with *foo* is extracted with fieldsolver mode 600. For instance, this statement would use fieldsolver mode 600 for top level net names foobar, foos, and foo1, but 1foo or ffoo would be extracted with Calibre xACT.

Example 12-3. Including Matching Names From All Levels of Hierarchy

```
PEX FIELDSOLVER MODE 600 LAYOUTNAMES RECURSIVE "?in?" "?out?"
```

Explanation: Includes any source net, from any level of the hierarchy, that is not ported out and has the character strings *in* or *out* somewhere within the net name. For instance, this statement

would extract pin and pout with fieldsolver mode 600, no matter what level of the hierarchy they occur on.

Note

The tool supports wildcards in the net name only, not in the path. For example, *X0/X1/foo?* is supported, but *?/X0/vdd* is not supported. In other words, the wildcard character does not match hierarchy.

Select-Layer Extraction with Calibre xACT and Calibre xACT 3D

Calibre xACT and Calibre xACT 3D can be used together to extract specific layers with field solver accuracy.

When performing select-layer extraction with Calibre xACT and Calibre xACT 3D, you must perform the following operations:

1. Identify the selected layers by name using the PEX Fieldsolver Mode LAYERS statement in your SVRF rule file.
2. Use the "-3dselect" Calibre xACT command line switch.

You can specify the accuracy level used for extraction on one or more specified layers using PEX Fieldsolver Mode LAYERS.

Example 12-4. Selecting Layers for Field Solver Extraction

```
SVRF: PEX FIELDSOLVER MODE 600 LAYERS M1

Command Line:  calibre -xact -3dselect -rcc rules.top
```

Explanation: When this SVRF statement is present in the rule file and -3dselect is specified on the command line, layer M1 is extracted with field solver mode 600. All other non-specified layers are extracted with Calibre xACT.

To specify the extraction accuracy level for both nets and layers, you can include both PEX Fieldsolver Mode LAYERS and PEX Fieldsolver Mode Nets statements in your rule file.

Example 12-5. Selecting Both Layers and Nets for Field Solver Extraction

```
SVRF: PEX FIELDSOLVER MODE 600 LAYERS M2
      PEX FIELDSOLVER MODE 600 NETS "clk"

Command Line:  calibre -xact -3dselect -rcc rules.top
```

Explanation: In this example, Calibre xACT extracts all layers on net clk with field solver mode 600 and extracts layer M2 on all other nets with field solver mode 600. All other non-specified layers and nets are extracted with Calibre xACT.

Extracting Particular Nets with the Calibre xACT 3D PDB flow

Calibre xACT 3D allows you to control how certain nets are handled during parasitic extraction.

When performing selected-net extraction using the Calibre xACT 3D PDB extraction flow, you must perform the following operations:

1. Identify the selected nets by name using the PEX Extract Include statement in your SVRF rule file.
2. During parasitic database (PDB) creation with the "-pdb" command line switch, use the "-select" Calibre xACT 3D command line switch.

If you omit the PEX Extract Include statements from your rule file and invoke the Calibre xACT 3D tool using the "-select" switch, then the tool will issue an error and terminate the run.

Wildcards and Search Level Specification

You can use wildcards to select nets to include or exclude when performing a parasitic extraction run. You can also specify where in the hierarchy to search for a matching net name.

- Use question mark (?) character as a wildcard matching zero or more characters in a net name.
- Use one of two mutually exclusive secondary keywords, TOPLEVEL and RECURSIVE, to specify where in the hierarchy to search for a matching name. TOPLEVEL is the default.

Example 12-6. Including Matching Names in Top Level

```
PEX EXTRACT INCLUDE LAYOUTNAMES TOPLEVEL "X0/X1/foo?"
```

Explanation: Includes any net in placement X0/X1 that is not ported out of that cell and has a name beginning with *foo*. For instance, this statement would include top level net names foobar, foos, and foo1, but would not include 1foo or ffoo.

Example 12-7. Including Matching Names From All Levels

```
PEX EXTRACT INCLUDE LAYOUTNAMES RECURSIVE "?in?" "?out?"
```

Explanation: Includes any source net, from any level of the hierarchy, that is not ported out and has the character strings *in* or *out* somewhere within the net name. For instance, this statement would include pin and pout, no matter what level of the hierarchy they occur on.

Note

The tool supports wildcards in the net name only, not in the path. For example, *X0/X1/foo?* is supported, but *?/X0/vdd* is not supported. In other words, the wildcard character does not match hierarchy.

Exclusion of Power and Ground Nets

Using the Calibre xACT tool, you exclude nets from the extraction run by including the PEX Extract Exclude Standard Verification Rule Format (SVRF) statement in the rule file. When you use this statement, you must list the nets by name you want excluded from the extraction run.

Excluded power and ground nets are not removed from the layout database. Their coupling effect on signal nets is included in the parasitic netlist. However, the final netlist will exclude parasitics attached to these nets.

You disable net exclusion by commenting out the PEX Extract Exclude statement in your SVRF rule file.

You can use wildcards to select nets to exclude when performing a parasitic extraction run. You can also specify where in the hierarchy to search for a matching net name.

- Use question mark (?) character as a wildcard matching zero or more characters in a net name.
- Use one of two mutually exclusive secondary keywords, TOPLEVEL and RECURSIVE, to specify where in the hierarchy to search for a matching name. TOPLEVEL is the default.

Example 12-8. Excluding Matching Names in Top Level

```
PEX EXTRACT EXCLUDE LAYOUTNAMES TOPLEVEL "?vdd?"
```

Explanation: Excludes any name in the top level namespace that has the character string *vdd* anywhere within it. For instance, this statement would exclude top level net names vdd, nvdd, and vdds.

Example 12-9. Excluding Matching Names From All Levels

```
PEX EXTRACT EXCLUDE LAYOUTNAMES RECURSIVE "?vdd?"
```

Explanation: Excludes any net, at any level of the hierarchy, that is not ported out and has a name containing the character string *vdd*. For instance, this statement would exclude nvdd and vdds, no matter what level of the hierarchy they occur on.

Coupling Capacitance Output Control

PEX Fieldsolver Mode selects the accuracy mode of Calibre xACT 3D. PEX Fieldsolver Mode 200 has a target of 5% total capacitance. Coupling capacitors that are at least 10% of the total capacitance are targeted to be within 10% of a reference field solver.

To preserve very small coupling capacitors for a design that also has very large capacitors, use PEX Fieldsolver Mode 600 with select net extraction. Setting the accuracy mode to 600 is recommended for increased accuracy of smaller coupling capacitances. For more information on select net extraction, see "Extracting Particular Nets with the Calibre xACT 3D PDB flow".

Grounding Coupled Capacitors

To decouple and ground any coupling capacitors in the parasitic netlist, use the "-g" command line option.

To force decoupling of coupled capacitors with Calibre xACT specify the optional "-g" switch on the command line. For example:

```
calibre -xact -c -g rule_file_name
```

The optional "-g" switch can also be specified when you invoke the Calibre xACT formatter. For example:

```
calibre -xact -fmt -c -g rule_file_name
```

Any coupled capacitors found in the parasitic models are decoupled and grounded in the parasitic netlist.

Ignore or Extract Floating Nets

By default, floating nets are treated as grounded during extraction. A floating net is defined as one not connected to a device and lacking port text. (The shapes comprising the floating net may have text attached in other ways such as by Attach without Port Layer Text. Because this does not define port placement, connectivity is not defined and the net remains floating.)

Treating floating nets as grounded may affect the accuracy of coupled capacitance, usually by overestimating it. There are two methods for improving accuracy:

- Floating Net Coupling Algorithm
- Extract Floating Nets

Both are controlled through the PEX Extract Floating Nets statement, which in the Calibre Interactive interface is set through **PEX Options > Netlist > Extract Floating Nets**.

Floating Net Coupling Algorithm

Set the PEX Extract Floating Nets statement to REDUCED to more accurately extract capacitance in the presence of floating nets. By default, when the Calibre xACT 3D tool performs RC extraction and converts coupling capacitances to grounded capacitances, it assumes floating signal nets are fixed (grounded) and does not extract them. This assumption results in some error for floating nets like metal fill.

When floating net coupling is enabled and a signal net is capacitively coupled to a floating net, the floating net is not assumed to be fixed. Instead, the Calibre xACT 3D tool approximates the effective capacitance of the floating net and computes the effective (series) capacitance to ground of the signal net through the floating net as shown in Figure 12-2.

Figure 12-2. Floating-Net Coupling Floating Example

This algorithm requires that there are signal nets and floating nets in the design, and is disabled if floating nets are extracted (that is, if PEX Extract Floating Nets is set to ALL). In contrast to extracting floating nets, the final netlist does not contain any floating nets.

Note

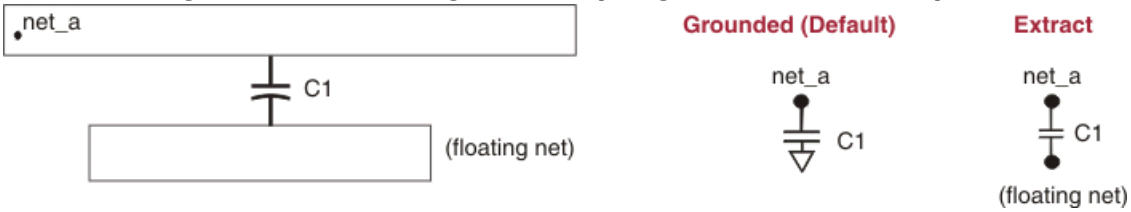
When extracting selected nets (-select switch), nets not selected are assumed to be grounded—this means that only selected nets can potentially float. To use the floating net coupling algorithm when extracting with selected nets, the floating nets must also be selected.

Extract Floating Nets

Setting the PEX Extract Floating Nets statement to ALL causes Calibre xACT 3D to output floating nets in the netlist and maintain the coupling between the signal and the floating net.

Figure 12-3 illustrates the Calibre xACT 3D tool's result when you set this. It shows a test structure with a signal net, net_a, and unnamed floating net. C1 represents parasitic coupling capacitance. When you extract floating nets, C1 is represented as a coupling capacitor between net_a and the floating net rather than lumped with intrinsic capacitance. The netlist contains both nets, and C1 is listed with other coupling capacitors.

Figure 12-3. Floating-Net Coupling Extraction Example



When extracting floating nets, both floating nets and signal nets are treated in the same manner. This increases extraction time and creates floating nets in the netlist.

Resistance Extraction and PERC

You can set up a rule file for parasitic resistance extraction so that it can be used with Calibre® PERC.

For more information on how to do this, see "Running Calibre PERC Using Calibre xRC Parasitic Resistance Netlist" in the *Calibre PERC User's Manual*.

Chapter 13
Controlling Netlisting

Netlist content can vary greatly depending on your design and analysis goals.

These topics describe how netlisting behaves and how to modify the basic extraction procedures to change the produced netlists.

Netlisting Multiple Corners and Multiple Temperatures **173**
Extracting Multiple Corners with Calibre xACT **175**
Generating Multiple Netlists With Calibre xACT **178**
Netlisting Only Direct Devices on a Selected Net **179**
Methods for Correcting Pin Swapping **180**
Using the Source Based Flow 181
How to Join a Disjoint Parasitic Model 183
Port Names for Net Names **184**
Verification of Timing with Probe Points **184**
Parasitic Extraction with Calibre View Device Properties **184**

Netlisting Multiple Corners and Multiple Temperatures

The Calibre xACT digital flow supports the extraction of multiple process corners and temperatures in a single run.

Follow this procedure when you want to netlist multiple process corners. Process corners refer to the variations on a “typical” process; for instance, metal thickness may not be exactly controlled.

Prerequisites

- A rules directory containing the following SVRF rule files for the target technology and metal scheme:
  - A top-level control file.
  - A capacitance rule file for each process corner.
  - A resistance rule file for each process corner.

- The following design files:
  - A technology LEF file (TLEF).
  - A LEF cell library.
  - LEF files for macros.
  - One or more DEF files containing the design data.

Procedure

1. Your top-level SVRF control file should look similar to the following:

```
LAYOUT PRIMARY "TOP"
LAYOUT PATH "./design/tech/tech.lef"
            "./design/lef/gsc145nm.lef
            "./design/lef/PADS.lef
            "./design/lef/myram.lef
            "./design/lef/dac6_op2.lef
            "./design/lef/ADC_5bit_sc_5.lef
            "./design/def/top.def
PEX NETLIST "netlist.spef.gz" SPEF SOURCENAMES MAPNAMES
PEX NETLIST CAPACITANCE UNIT fF
MASK SVDB DIRECTORY "svdb" XRC
LAYOUT SYSTEM LEFDEF
LAYOUT CASE YES
SOURCE CASE YES
PEX LEF EXTRACT CELL OBSTRUCTIONS YES
PEX DEF EXTRACT BLOCKAGES YES
PEX REDUCE CC ABSOLUTE 3 RATIO 0.03
PEX REDUCE DIGITAL YES
UNIT CAPACITANCE fF
INCLUDE xactdigital_layers.svrf
PEX EXTRACT TEMPERATURE 27 120
PEX XACT CORNER typical
PEX XACT CORNER cworst
// typical
#IFDEF $XACT_CORNER typical
   INCLUDE "./rules/7m_typical/rules.C"
   INCLUDE "./rules/7m_typical/rules.R"
   INCLUDE "./rules/7m_typical/rules.xact"
#ENDIF
// cworst
#IFDEF $XACT_CORNER cworst
   INCLUDE "./rules/7m_cworst/rules.C"
   INCLUDE "./rules/7m_cworst/rules.R"
   INCLUDE "./rules/7m_cworst/rules.xact"
#ENDIF
// default corner
#IFNDEF $XACT_CORNER
   INCLUDE "./rules/7m_typical/rules.C"
   INCLUDE "./rules/7m_typical/rules.R"
   INCLUDE "./rules/7m_typical/rules.xact"
#ENDIF
```

   For this example, this control file is named *xactdigital_multi.svrf*.

2. To extract multiple process corners, use the PEX XACT Corner statement. In this example the two process corners are typical and cworst. The PEX XACT Corner statement tells Calibre xACT the name of the process corner.
3. The PEX Extract Temperature statement defines two temperature corners, 27 and 120.
4. The variable $XACT_CORNER is set internally by Calibre and in each case carries the value of one of the process corners. The Include statements inside of the #IFDEF conditional use the value of the $XACT_CORNER variable to locate the capacitance, resistance, and xact decks for each corner. The Include statements following the #IFNDEF statement are required to make this SVRF code example compile in Calibre.
5. Run the extraction step. For example:

   ```
   calibre -xact -rcc xactdigital_multi.svrf
   ```

   The Calibre xACT extraction step generates one SPEF file for each temperature and process corner combination. In this example, the tool creates four SPEF netlists, one for each combination of the two process corners defined by the two PEX XACT Corner statements with the two PEX Extract Temperature corners.

   If the foundry rule deck contains double patterning corners, then Calibre xACT also generates four additional double patterning corner netlists; In this example, 8 netlists would be generated in total.
6. Optionally, to restrict the multi-corner run to specific process corners, use the -corner option. For example, the following command line extracts only the typical process corner:

   ```
   calibre -xact -corner typical xactdigital_multi.svrf
   ```

   The following command line extracts the typical and cworst corners:

   ```
   calibre -xact -corner typical,cworst xactdigital_multi.svrf
   ```

   For more information on command line options for Calibre xACT, see “calibre -xact” on page 238.

Extracting Multiple Corners with Calibre xACT

The Calibre xACT and Calibre xACT 3D direct netlisting flows support the extraction of multiple process corners in a single run.

Follow this procedure when you want to simultaneously extract and netlist all process, multi-patterning, and/or temperature corners in a single run using Calibre xACT. Process corners refer to the variations on a “typical” process; for instance, metal thickness may not be exactly controlled. The flow can be run on multiple remote machines or on one multi-CPU machine.

Prerequisites

- A rules directory containing the following SVRF rule files for the target technology and metal scheme:
  - A top-level control file (*top.rules*).
  - A capacitance rule file (*rules.C*) for each process corner.
  - A resistance rule file (*rules.R*) for each process corner.

    Note

    Calibrated capacitance and resistance rule files must be created with xCalibrate version 2015.1 or later.

- A Calibre xACT rule file (*rules.xact*) for each process corner.
- An LVS rule file.
- The following design files:
  - GDS or OASIS file.
  - Source netlist, if source names are desired.

Procedure

1. Create a top-level SVRF control file. It should look similar to the following:

```
LAYOUT PRIMARY "topcell"
LAYOUT PATH "./design/topcell.gds"
LAYOUT SYSTEM GDSII
SOURCE PRIMARY "topcell"
SOURCE PATH "./design/topcell.src.net"
SOURCE SYSTEM SPICE
UNIT CAPACITANCE fF
INCLUDE calibre_lvs_rules.svrf

PEX EXTRACT EXCLUDE SOURCENAMES VDD VSS
PEX NETLIST CAPACITANCE UNIT fF
PEX EXTRACT TEMPERATURE 27
PEX REDUCE ANALOG YES
```

2. To netlist multiple process corners use the PEX Netlist statement. For example:

```
PEX NETLIST
"netlist_%xactProcessCorner%_%xactDpCorner%_%xactTemperature%.dspf"
DSPF SOURCENAMES
```

   In the example, the file name uses substitution variables.

   - %xactProcessCorner% — is set to the process corner name.
   - %xactDpCorner% — is set to a particular corner name if double patterning corners are present.

- %xactTemperature% — is set to temperature values defined using either PEX Extract Temperature or PEX XACT Corner ... TEMPERATURE statements.

These variables are not case sensitive and can be placed anywhere within the filename. They are replaced during execution with predefined names for all possible combinations.

3. To extract multiple process corners, use the PEX XACT Corner statement. The PEX XACT Corner statement tells Calibre xACT the name of the process corner. Add these statements to your top-level SVRF control file. In this example the three process corners are typical, cworst, and rcbest.

```
PEX XACT CORNER typical
PEX XACT CORNER cworst
PEX XACT CORNER rcbest TEMPERATURE 27 140
#IFDEF $XACT_CORNER typical
   DFM DATABASE dfmdb.typical OVERWRITE
   INCLUDE "./typical/rules.C"
   INCLUDE "./typical/rules.R"
   INCLUDE "./typical/rules.XACT"
#ENDIF
#IFDEF $XACT_CORNER min
   DFM DATABASE dfmdb.cworst OVERWRITE
   INCLUDE "./cworst/rules.C"
   INCLUDE "./cworst/rules.R"
   INCLUDE "./cworst/rules.XACT"
#ENDIF
#IFDEF $XACT_CORNER max
   DFM DATABASE dfmdb.rcbest OVERWRITE
   INCLUDE "./rcbest/rules.C"
   INCLUDE "./rcbest/rules.R"
   INCLUDE "./rcbest/rules.XACT"
#ENDIF
#IFNDEF $XACT_CORNER
   DFM DATABASE dfmdb OVERWRITE
   INCLUDE "./typical/rules.C"
   INCLUDE "./typical/rules.R"
   INCLUDE "./typical/rules.XACT"
#ENDIF
```

The variable $XACT_CORNER is set internally by Calibre and each case carries the value of one of the process corners. The Include statements inside of the #IFDEF conditional use the value of the $XACT_CORNER variable to locate the capacitance and resistance decks for each corner. The Include statements following the #IFNDEF statement are required for Calibre.

4. To netlist a single double patterning process corner, include the following statement:

```
PEX XACT CORNER cworst DP "CCworst"
```

5. Run the extraction step. For example:

```
calibre -xact -rcc -turbo xact_multicorner.svrf
```

Results

Calibre xACT generates one or multiple netlists for each PEX XACT Corner statement, depending on the number of DP and/or temperature corners specified.

Generating Multiple Netlists With Calibre xACT

Calibre xACT and Calibre xACT 3D support the creation of multiple netlists in a single run.

This procedure demonstrates how Calibre xACT can be used to generate multiple output netlists in different formats and net models using the PEX XACT Control SVRF statement.

Prerequisites

- You have created a top-level control file called *xact_t.rules*. See "Creating a Top-level Control File" for an example.
- You have met the requirements outlined in "Transistor-Level Extraction Setup" on page 25 for Calibre xACT.

Procedure

1. Using a text editor, create a file called *xact_netlist.control* file and enter the following statements:

```
// Netlist specifications for RC Models
NETLIST_CONTROL {
   MODEL: RC
   NETLIST: pex1.dspf NETLIST_FORMAT: DSPF NET: ?VDDCOL? MODEL: Cg
}
// Netlist specifications for RCC Models
NETLIST_CONTROL {
   MODEL: RCC
   NETLIST: pex2.spef NETLIST_FORMAT: SPEF NET: ?VDDCOL? MODEL: Cg
   NETLIST: pex3.hspice NETLIST_FORMAT: HSPICE
}
```

   Save and close the file.

2. Using a text editor, add the following statement in the *xact_t.rules* file:

```
PEX XACT CONTROL FILE xact_netlist.control
```

   Save and close the file.

   Note

   Extraction stops with an error if you do not have a PEX Netlist statement specified in your rule file.

3. Run Calibre xACT to perform extraction:

```
calibre -xact -rcc xact_t.rules
```

and produce three netlists: *pex1.dspf*, *pex2.spef*, and *pex3.hspice*.

Results

The first netlist, *pex1.dspf*, has all nets netlisted with the RC net models except for the ?VDDCOL? nets that are netlisted with Cg net models.

The second netlist, *pex2.spef*, has all nets netlisted with RCC net models, except for the ?VDDCOL? nets that are netlisted with Cg net models.

The third netlist, *pex3.hspice*, has all nets netlisted with RCC net models.

Related Topics

Multiple Netlist Generation

Netlisting Only Direct Devices on a Selected Net

To output a net, its devices, and the extracted parasitics without additional nets or devices, use the PRUNE keyword as part of the SVRF statement PEX Netlist and perform selected-net extraction using the Calibre xACT 3D PDB flow.

Procedure

1. In the PEX top-level control file, include these statements:
   - PEX Extract Include (to specify selected nets)
   - PEX Netlist … PRUNE … (to specify netlisting options)
2. During parasitic database (PDB) creation step of the Calibre xACT 3D PDB flow, use the “-select” command line switch. For example:

   ```
   calibre -xact -3d -pdb -rcc -select my_rules
   ```

   For more information on the Calibre xACT 3D PDB flow, see “Using the Calibre xACT 3D PDB Flow” on page 63.
3. Run netlisting as usual.

Results

Only the selected net and its devices and parasitics will appear in the netlist. Although associated nets and devices are extracted and in the PDB, they are not written to the netlist.

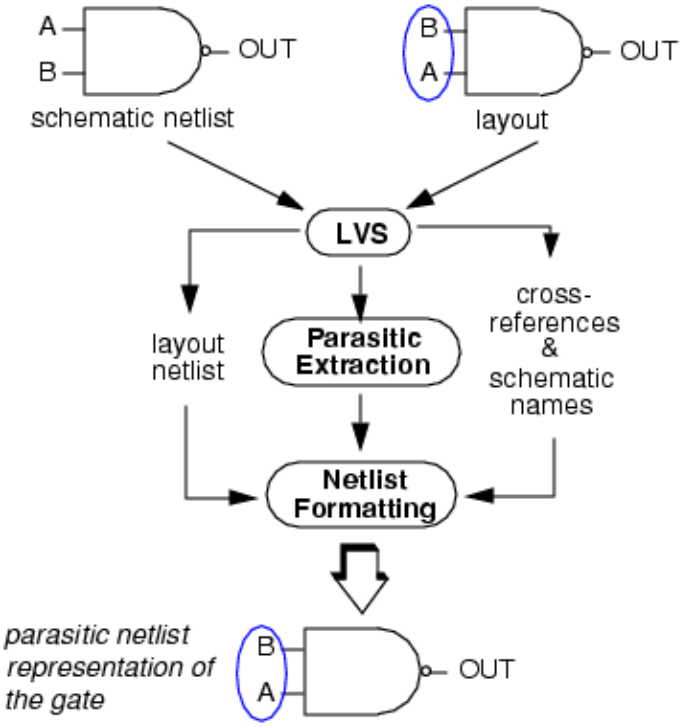
Methods for Correcting Pin Swapping

When performing parasitic extraction with the Calibre xACT 3D tool, you can encounter issues with logical pin swapping in a distributed RC parasitic netlist.

The Calibre nmLVS-H tool performs logical pin swapping and propagates the swapped pins into the layout-to-source cross-reference files. The Calibre xACT 3D formatter then uses the cross-reference files when constructing a distributed RC netlist. Logical pin swapping at the gate level can create problems when using the netlist in a downstream simulation tool—the layout is LVS clean, but there is a pin mismatch between the schematic and the layout.

Figure 13-1 shows an example of logical pin swapping the Calibre xACT 3D tool produces by default for a schematic NAND gate and its representation in the layout.

Figure 13-1. Gate-Level Logical Pin Swapping Example



In this example, the Calibre nmLVS-H tool reports the design is LVS clean and creates the layout-to-source cross-reference files using the swapped pins. By default, the Calibre xACT 3D tool subsequently uses the swapped pins when creating the distributed RC netlist.

You can prevent pin ordering anomalies by using DSPF format. This format bases pin order on the pin names instead of the pin connections. If there are port direction restrictions in a certain output format, the formatter makes the appropriate adjustment. In SPEF, the formatter uses “b” for type “x” pins because SPEF grammar does not include an any-direction notation.

There are four methods to correct pin order. The one to use depends on whether you are using source-based flow, in which nets are based on source netlists, or the default layout-based flow (using layout names or schematic/source names).

- If you are using source-based flow, see "Using the Source Based Flow".
- If the pins are on intentional models, see "PEX BA Mapfile".
- If you are using a layout-based flow and the pins are on primitives or the top circuit only, see "PEX Pin Order".
- If you need to specify pin direction or pin order on intermediate circuits, see "How to Join a Disjoint Parasitic Model".

Using the Source Based Flow

Normally, the Calibre xACT 3D formatter produces a layout netlist containing source names. The formatter uses the circuit pin order and cell hierarchy in the source netlist instead of the layout netlist when generating the extracted netlist. When you use this flow, the Calibre xACT 3D formatter produces a source netlist containing the extracted parasitics.

The source-based flow is not supported for *full hierarchical extraction*.

Use the Calibre xACT 3D formatter's source-based flow to:

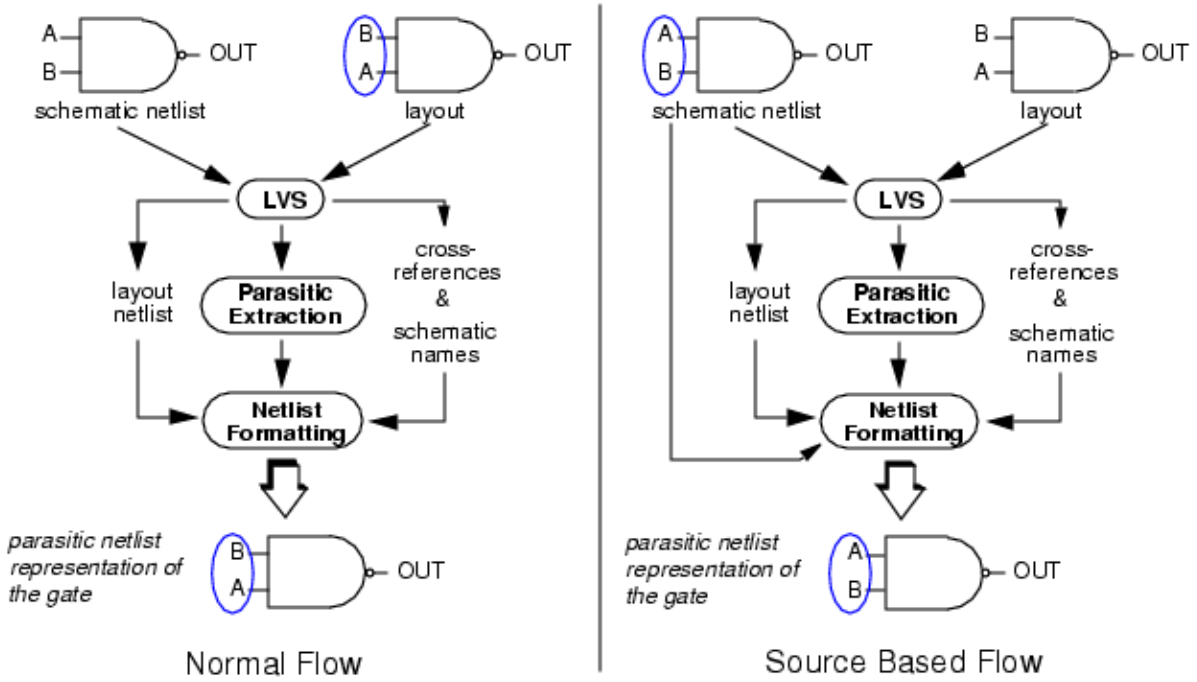
- correct logical pin swapping in the parasitic netlist.
- collapse multi-fingered devices back into one device.

If you are creating the source netlist with the Calibre nmLVS software, you must use Calibre nmLVS-H and the design must pass LVS. Because source-based extraction uses the schematic for formatting, it is very important that the layout pass LVS checks with no errors. If it is not, your formatting stage quits with the following error:

```
ERROR: Export name mismatch caused fatal error.
Please verify that this design is LVS CLEAN.
```

Figure 13-2 provides a comparison of the normal and the source based Calibre xACT 3D formatter flows. When creating a parasitic netlist with the source based flow, the Calibre xACT 3D formatter also uses the schematic (source) netlist.

Figure 13-2. Comparison of Normal and Source Based Flows



Prerequisites

Before using this source based flow, you must modify your Standard Verification Rule Format (SVRF) rule file by adding or modifying the following SVRF statements:

- Mask SVDB Directory BY GATE
- Source Case YES

Note

When using the source-based flow, the Calibre xACT formatter automatically sets the Source Case SVRF statement to YES regardless of its setting in your SVRF rule file.

Procedure

1. To output each multi-fingered transistor as a separate device using the source-based flow, specify the PEX Netlist statement with the SOURCENAMES keyword in your rule file.
2. To collapse multi-fingered devices back into one device using the source-based flow, specify the PEX Netlist statement with the SOURCEBASED keyword in your rule file.
3. Generate the netlist.

Results

You can verify whether the Calibre xACT 3D formatter used the source-base flowed by comparing the output netlist to the original LVS source netlist and seeing that the transistors are the same.

For example, given the original LVS source netlist sample:

```
MM7 net054 net37 vss! vss! N W=15u L=4u M=4
```

The Calibre xACT 3D netlist output with PEX NETLIST SOURCENAMES specified is:

```
MM7 MM7:d MM7:g MM7:s MM7:b N L=4e-06 W=1.5e-05
MM7@4 MM7@4:d MM7@4:g MM7@4:s MM7@4:b N L=4e-06 W=1.5e-05
MM7@3 MM7@3:d MM7@3:g MM7@3:s MM7@3:b N L=4e-06 W=1.5e-05
MM7@2 MM7@2:d MM7@2:g MM7@2:s MM7@2:b N L=4e-06 W=1.5e-05
```

Here the output netlist shows each multi-fingered transistor as a separate device.

The Calibre xACT 3D netlist output with PEX NETLIST SOURCEBASED specified is:

```
MM7 MM7:d MM7:g MM7:s MM7:b N L=4e-06 W=1.5e-05 M=4
```

Here the output netlist shows the multi-fingered device as one device with a multiplicity of 4 (M=4), like in the original LVS source netlist.

How to Join a Disjoint Parasitic Model

Use the PEX Netlist Virtual Connect statement to join a disjoint parasitic model.

When you specify this statement with the YES parameter, the Calibre xACT 3D formatter performs the following processing operations:

- Looks for disjoint net model fragments created during LVS by Virtual Connect specification statements.
- Subsequently connects these fragments to the “trunk” of the net model using a small-value resistor.

You can allow for incomplete net routing using the Virtual Connect SVRF specification statements during LVS and connect like-named net fragments. The Calibre xACT 3D tool will extract and produce a PDB containing a net model consisting of several disjoint parasitic model fragments—a true representation of the actual drawn design. Consequently, this will create problems for post-extraction simulation because there is no physical connection between these net fragments.

Note

You should use this statement in specific limited cases where you require a “clean” LVS when using Virtual Connect specification statements for locally disjoint power or signal nets. Ultimately, your design should be LVS clean without using either Virtual Connect statements.

When you use the PEX Netlist Virtual Connect statement, the Calibre xACT 3D formatter checks for disjoint net fragments. If the formatter finds such a fragment, the application selects a node on the fragment and connects this node using a low-value resistor to a node on the "trunk" of the net model. Although the formatter makes a reasonable choice of nodes when connecting, this connection is essentially arbitrary.

If you want a listing of the connections the Calibre xACT 3D formatter makes, then use the -fmt_warnings switch on the command line during Calibre xACT 3D formatter invocation.

Port Names for Net Names

The standard behavior is for the Calibre xACT 3D formatter to use the name of a connected port for the name of a net, even if the name of the port is different from the net name in the parasitic model.

Verification of Timing with Probe Points

Within the parasitic model, each probe point exists as an internal connection with a unique *probe_name*. You will be able to access each probe point by net and *probe_name*, depending on your simulator's behavior.

The PEX Probe File statement allows you to verify timing from specific points on each net by setting up probe points. For more information on the PEX Probe File SVRF statement, refer to the *Standard Verification Rule Format (SVRF) Manual*.

Parasitic Extraction with Calibre View Device Properties

Layout properties are calculated and represented in the CalibreView netlist produced by Calibre xACT.

Calibre xACT follows simple steps when calculating device properties. For details, see "Device Property Calculation in Calibre View Generation" in the *Calibre Interactive User's Manual*.

Chapter 14
Integration and Troubleshooting Topics

Some troubleshooting and integration techniques are provided to address common problems.

Integration **185**
Optimization **188**
Troubleshooting **191**
Setting Up For Troubleshooting 191
Invocation Issues 191

Integration

Integration information is provided for CAD engineers who are deploying pre-configured tools to their engineering groups.

Instructions for integrating the Calibre Interactive interface with layout viewers are available in the "Setting the Calibre Interactive Socket Port for a Layout Viewer" appendix of the *Calibre Interactive User's Manual*.

Guide to Calibre Interactive Files

The Calibre Interactive environment reads several types of configuration files. Settings in any of these affect how Calibre runs initiated from Calibre Interactive behave. These files are the following:

- **Preferences file**

  The Calibre Interactive preferences file for parasitic extraction is *.cgipexdb*. The default location is*$HOME/.cgipexdb*. The preferences files save the settings on the Startup tab of the Setup Preferences dialog and previously referenced runsets. When Calibre Interactive starts it automatically reads the preference file.

- **Runsets**

  A runset is a text file created by Calibre Interactive to store the settings specified in the interface. Only non-default data is recorded. Users are typically prompted to load a runset at the start of each interactive session.

- **Run scripts**

  A run script is any script which can be executed. For Calibre Interactive, run scripts are typically part of a trigger as described in the "Trigger Functions in Calibre Interactive"

chapter of the *Calibre Interactive User's Manual*. More generally, shell run scripts may also be used at the command line to execute a series of Calibre Interactive runs.

- **Rule files**

  A rule file is a file that contains SVRF and TVF statements specifying the details of the Calibre run. Many SVRF specifications are only accessible in the rule file; a user cannot override them from Calibre Interactive.

Calibre Interactive loads files in the following sequence when invoked:

Table 14-1. Calibre Interactive Settings Sequence

Stage	Action
1	Load the preferences file specified by $MGC_CALIBRE_PEX_RUNSET_FILE. By default, this points to *$HOME/.cgipexdb*. The preferences file populates the runset list and startup preferences.
2	Load a runset file and fill in the fields in the GUI. If the Load Runset File dialog has not been disabled by the preferences file, Calibre Interactive prompts the user to specify a runset. If the dialog has been disabled or the user cancels the dialog, no runset is loaded and default values are used in the fields.
3	Load files as user specifies. Typically, the user loads at least one rule file but may also load additional runsets. *Rule file settings are not loaded into the GUI until the user clicks Load.* The rule file and additional runsets can change fields already set in stages 1 and 2. The most recently loaded settings file (or manual setting by the user) is used at execution.
4	At execution, Calibre Interactive prepares a control file and runs any pre- and post-execution trigger run scripts.
5	At exit, Calibre Interactive prompts the user to save current settings to the loaded runset file. If the user made any changes, they will overwrite previous settings.

In stage 4, Calibre Interactive prepares a control file. This file includes the rule file and GUI settings. Its name is based on the rule file name. For example, if the rule file is "rules", then the control file is "_rules_". This file is written to the working directory. Subsequent runs overwrite control files with the same name.

Batch Shell Script Creation in Calibre Interactive

Many CAD engineers find it easiest to create new scripts by working in Calibre Interactive until they have a debugged setup, and then use Calibre Interactive's files as the basis of a shell script which they deliver to their customers.

For information on how to run a batch shell script in Calibre Interactive see "Running Batch Calibre with a Calibre Interactive Control File" in the *Calibre Interactive User's Manual*.

Best Practices for Shell Scripts

An example run script and tips for writing shell scripts are provided for you. The tips are based on experience helping customers set up their solutions.

Table 14-2. Best Practices for Shell Scripts

Tip	Reason
Explicitly set the shell.	This ensures the script will run for all users, no matter their shell preference.
Use redirects to capture messages.	Many examples in training and on Support Center use "| tee" to capture logs as well as display messages in the terminal. However, "| tee" does not copy the standard error (stderr) output to the log file. Use ">&" or ">&!" in most shells to redirect both standard output and standard error to the same file.
Have the script remove any Calibre-generated files before invoking Calibre.	Calibre leaves databases in the working directory so that users can come back to their results later. However, subsequent runs will overwrite data in existing files and this could cause misleading results - for example, left over netnames appearing in a netlist or erroneously swapped pins. Because some users may have a local version of rm, we recommend setting it explicitly as: `/bin/rm -rf` or as appropriate for your operating system.
Set environment variables explicitly in the script rather than the user environment.	Setting environment variables explicitly in the script serves several purposes: • Easier to verify when debugging. • Reduces problems due to variables remaining set in the shell; they are only active while the script is executing. • Less typing than setting in the shell each time. • Easier to package complete testcases.

The following is an example shell script that follows best practices. Alternatives are provided together, with all but one commented out. This example run script can be used as the basis of your scripts.

Figure 14-1. Example Shell Script to Run Calibre

```
#!/bin/csh -f
## Ensure which version will be run
setenv MGC_HOME /net/tools/calibre/<calibre_release>/lv_micro.ixl
setenv PATH $MGC_HOME/bin:$PATH
## Clean the working directory
/bin/rm -r *svd* *SVD* lvs.log phdb.log pdb.log fmt.log *sum tr*
####### Use this environment variable for debugging ###############
#####################################################################
#setenv CALIBRE_ECHO_RULE_FILE YES
######## Run-Specific Environment Variables ######################
#####################################################################
#            Mentor Graphics Variables
#            Your Flow Variables
#setenv PROCESS_SIZE 90
#setenv PROCESS_TYPE CU
#setenv ADD_FILL NO
## Hcells for LVS and PHDB
set h = ""
#set h = "-hcell hcells"
## Xcells for xACT PDB
set x = ""
#set x = "-xcell xcells"
set r = "rules"
set c = "layout_primary"
###### Run PHDB, PDB, and FMT steps
$MGC_HOME/bin/calibre -lvs -hier -spice svdb/$c.sp $h $r >&! lvs.log
#$MGC_HOME/bin/calibre -xact -phdb                  $x $r >&! phdb.log
$MGC_HOME/bin/calibre -xact -3d -rcc               $x $r >&! pdb.log
$MGC_HOME/bin/calibre -xact -fmt -all               $x $r >&! fmt.log
```

Optimization

Tips are provided for common optimization problems that CAD engineers are asked to tackle. Optimization is a broad topic, and these suggestions are by no means exhaustive.

Runtime Improvement Methods

There are several methods to make Calibre xACT runs faster. Not all of the following may apply to your situation.

- Run on multiple CPUs (-turbo).

  By default, Calibre xACT consumes one license and runs on two CPUs no matter how many are available. Adding "-turbo" to the PDB stage causes the run to use as many CPUs as it has licenses for. For more information on this option, refer the "Calibre xACT Tool Invocation Reference" and the *Calibre Administrator's Guide*.

- Extract only the details you need.

Extraction can take a long time to run not just because of all the calculations but also because of the memory resources needed to process all the geometries of very large designs. Reducing the number of polygons by restricting input reduces both the number of calculations and the memory load.

- Use gate-level extraction to ignore devices.

  Gate-level extraction uses an xcell file to indicate areas where parasitics are not extracted. Typically the contents of xcells are devices or standard cells, which already include parasitic effects in the simulation models.

- Exclude global nets such as power and ground.

  Many types of analysis such as cross talk or static timing analysis only require data on signal nets. Parasitic effects on non-signal nets in these cases rarely modify simulation significantly. Both Calibre run time and simulation time will be sped up by omitting global nets in these cases.

Netlist Reduction Techniques

There are two ways to create smaller netlists:

- Decrease the quantity of parasitic elements. This is known as reduction. Several reduction techniques are discussed in “Reduction Techniques” on page 253.
- Avoid long net names by minimizing the number of characters.

Generally, reduction techniques are preferred because they also reduce simulation times. If you need to use other methods to create more compact netlists, consider using the following SVRF statements:

PEX Netlist Global Nets

PEX Netlist . . . SPEF

PEX Netlist . . . DSPF

PEX Netlist . . . HSPICE SHORTPINNAMES

PEX Netlist Noxref Net Names

PEX Netlist Create Smashed Device Names

Best Way to Resize Designs

You can use Calibre to check whether layouts are suitable for process migration. Scaling designs and running DRC is relatively straightforward, but smaller sizes also require you to recalculate parasitic effects.

When you are working with parasitics, you should scale your layouts using the PEX Magnify statement instead of the DRC methods of Magnify or Precision and Resolution because PEX Magnify can also scale your device properties.

If you are also scaling the layout for Calibre nmDRC and nmLVS, be sure that your combination of statements does not just change the scale. See “Input Layout Database Magnification” in the *Calibre Verification User’s Manual* for more details.

Troubleshooting

Analysis techniques are provided for CAD engineers with some simple heuristics to aid in-house troubleshooting. This includes many of the steps used by Siemens support representative for initially diagnosing problems.

Setting Up For Troubleshooting **191**
Invocation Issues **191**

Setting Up For Troubleshooting

There are steps you can follow when you get a call from a user who is having trouble.

Prerequisites

- Working Calibre installation.
- Knowledge of shell commands for setting environment variables and creating log files.

Procedure

1. Get the full rule file by setting CALIBRE_ECHO_RULE_FILE to ON. This will echo out all included statements as well as the top rule file.
2. If you are using Calibre Interactive, be sure to also collect the control file, typically named “_rules_”.
3. If you are using scripts or entering commands directly at the prompt, be sure to redirect standard out and standard error to log files.
4. Be sure to use case-insensitive searching such as “grep -i” when looking through the log files. SVRF is generally not case sensitive, and Calibre messages may use mixed case.

Related Topics

Integration

Invocation Issues

If you are getting a usage message when you try to invoke Calibre, there is an error in the command line.

Check for the following:

- Are any of the switches incompatible? See “Calibre xACT Tool Invocation Reference”.
- Are there any errors in syntax such as missing arguments or switched order?

If Calibre invokes but quickly exits, there is typically an explanatory error message, such as “Could not get requested number of CPUs” or “problem with access of file”. If the run did not

cleanly exit with an error message, and you can reproduce the problem, please open a service request on support.sw.siemens.com.

Chapter 15
Handling Parasitic On-Chip Variation

Calibre xACT can help you handle parasitic on-chip variation.

This chapter contains information about parasitic on-chip variation.

On-Chip Variation in Parasitic Extraction **193**
Parasitic Extraction Techniques for On-Chip Variation **196**
In-Die Variation 196
CMP Modeling 197

On-Chip Variation in Parasitic Extraction

On-chip variation refers to the inter- and intra-die variation in physical process properties. These variations can cause poor yields.

Layout data contains shapes that are at desired or “drawn” dimensions. The actual width of each line varies from the drawn dimension. This variation has both random and deterministic attributes. The random variations are caused by fluctuations in the manufacturing process. The deterministic variations are caused by the interaction of fabrication technologies and layout.

Sources of On-Chip Variation

There are several sources of on-chip variation:

- Fabrication equipment can vary. A wafer processed in an etch station will not be identical to a wafer processed in a neighboring etch station. This is sometimes referred to as “process variation”.
- Relative placements interact. The way lines and fill are placed (both spacing and internal width) changes how the actual drawn shapes appear. RET/OPC corrects as much of this as possible, but there is always some effect.
- Metals and dielectric can bulge or sag. Additionally, steps such as reactive ion etching cause some lines to be etched deeper than others. The changes in height are referred to as “loading”.
- The chemical-mechanical polishing (CMP) step will grind down the interiors of large polygons, and some regions of the wafer, more than others.

How Calibre Parasitic Extraction Models On-Chip Variation

The Calibre parasitic extraction software provides two techniques for modeling on-chip variation effects:

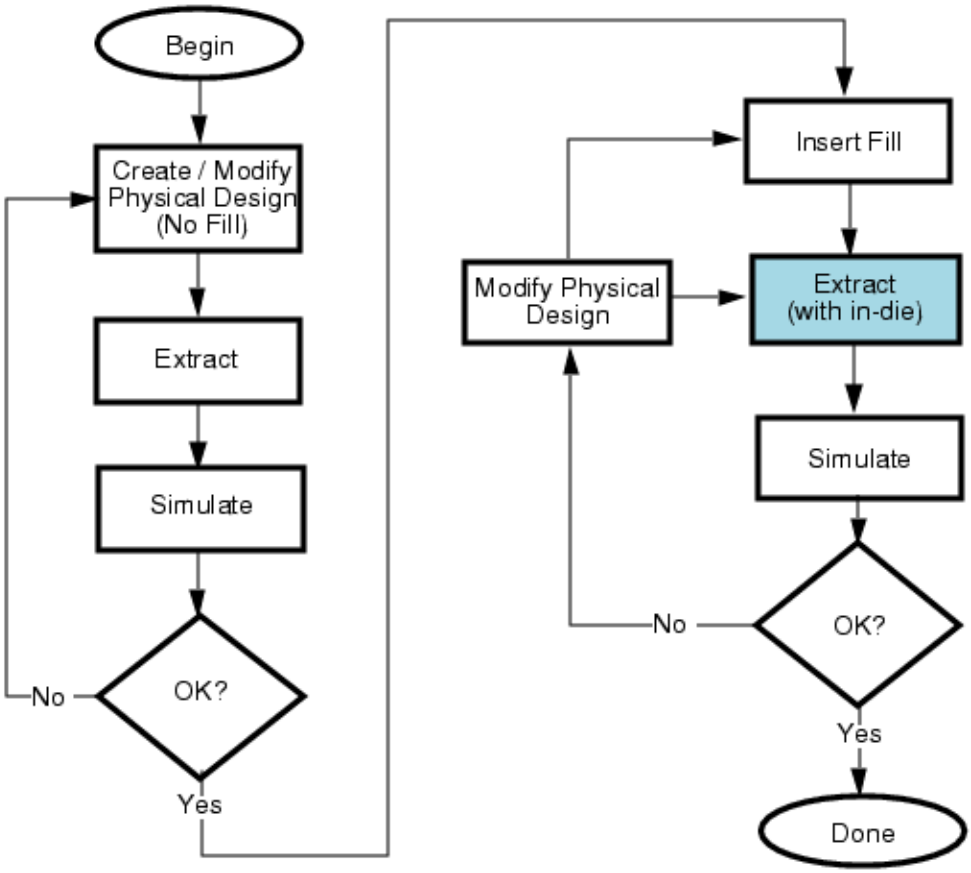
- **In-die variation** — Local density is used in conjunction with specialized rules to make adjustments to the drawn shapes. The adjusted shapes are used for calculating parasitic effects. Some foundries also supply loading effects in their in-die information. See "In-Die Variation" for more detail.
- **CMP modeling** — If your foundry provides appropriate CMP modeling files, you can have Calibre xACT read them in and make adjustments to metal thickness.

When to Use On-Chip Variation During Parasitic Extraction

The on-chip variation algorithms make small adjustments to parasitic extraction. They also increase the required amount of memory and can make the extraction step take longer to complete. They are recommended for use on near-final designs, as shown in Figure 15-1.

If you have not placed your metal fill yet, you almost certainly do not want to use in-die variation or CMP modeling. These techniques are sensitive to local density, which will change when you add metal fill.

Figure 15-1. Design Flow Showing In-Die Variation Techniques



Parasitic Extraction Techniques for On-Chip Variation

In-die variation and CMP modeling are independent of each other and may be singly or jointly used in extraction.

These sections explain the techniques for on-chip variation and how to use them.

In-Die Variation **196**
CMP Modeling **197**

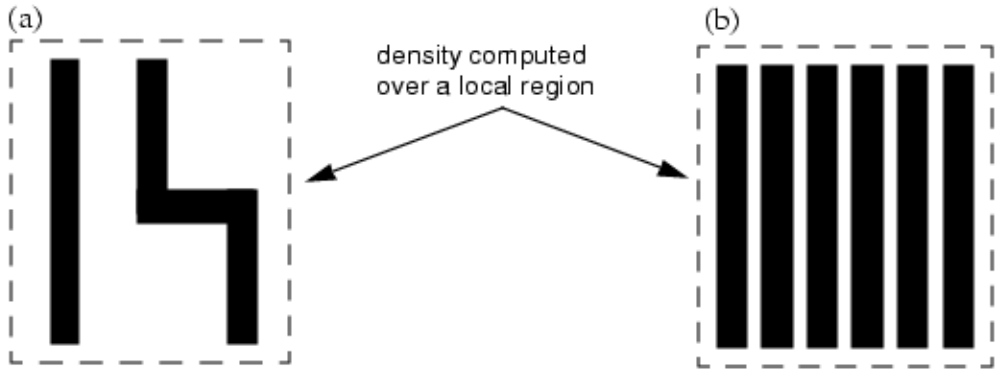
In-Die Variation

The in-die variation workflow takes into account the in-die variation of resistance and capacitance values in the extraction process.

When appropriately calibrated, the Calibre xACT tool uses the drawn dimensions of each conductor along with the local density of the material in a region around the conductor to determine the actual width, spacing, and thickness of each line.

The concept of local density is key to predicting the actual dimensions of each line. The density of material in a region or window around each conductor affects the thickness of the line, and indirectly, the width. Figure 15-2 shows how local density can vary from region to region.

Figure 15-2. Layout Structure Affects Local Density



In window (a), the area occupied by the layer material is relatively small compared to the area of the window itself; the local density is low. In window (b), the ratio of conductor area to window area is much higher, and local density is therefore much higher. Even though all of the lines in (a) and (b) have the same drawn width, the actual widths and thicknesses will differ between (a) and (b).

Prerequisites

In-die variation is generally enabled by the foundry. The Calibre xACT user can turn it on or off, but not change its values.

Caution

Calibrated rules created before version 2008.2 allowed the use of manually created Parasitic Variation statements for similar effects, but the newer encrypted calibrated rules are not compatible with this.

The following method is preferred, because the xCalibrate rule file generator can more accurately calculate the effect of in-die variation on the capacitance and resistance equations.

Procedure

1. The foundry performs process measurements and determines how density and edge-to-edge distance affects properties such as edge displacement, thickness, temperature coefficients, and resistance.
2. The foundry creates in-die tables that follow the format described in Table Syntax in the *xCalibrate Batch User's Manual*.
3. The foundry runs calibration and provides calibrated rules as part of its design kit.
4. The CAD engineer at the chip design company provides wrapper scripts and SVRF files for the Calibre users. These may turn on or off in-die effects.
5. If the wrapper scripts do not turn on in-die functionality, the engineer running Calibre xACT may choose to enable in-die variation.
6. Or, for manually-created calibrated rules, you can include in-die variation with the following technique:

   Obtain information on in-die variation from the foundry.
7. Add Parasitic Variation statements to your rule file to model the effects of in-die variation on width, thickness, and resistance. Because you will not always want to include in-die effects, it is recommended to enclose the lines within a #ifdef preprocessor directive.

   Note

   Older calibrated rule files may already include some Parasitic Variation statements.

CMP Modeling

The CMP modeling workflow for modeling chemical-mechanical polishing adds some specialized steps to the extraction process.

CMP models predict how much erosion of the dielectric and dishing of metal occur on the wafer surface because of the CMP step in manufacturing. The models create a grid of the wafer and, for each square, determine the amount of polishing force or erosion. (The exact effect calculated by the CMP model depends on which modeling program you are using.)

Figure 15-3 shows the potential effects of CMP. The metal is softer than the dielectric, and so is abraded more rapidly, resulting in dishing. Where dense stretches of metal switch to dielectric, the polishing agent can "pile up" on one side, resulting in uneven erosion. Design geometry interacts with the CMP forces to dictate exactly how thickness will vary.

Figure 15-3. Typical Effects of CMP

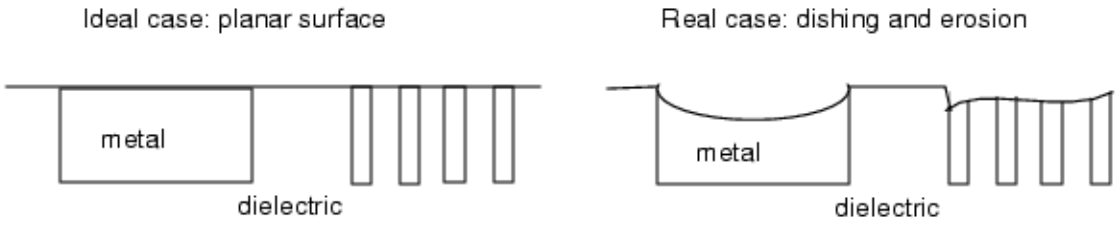
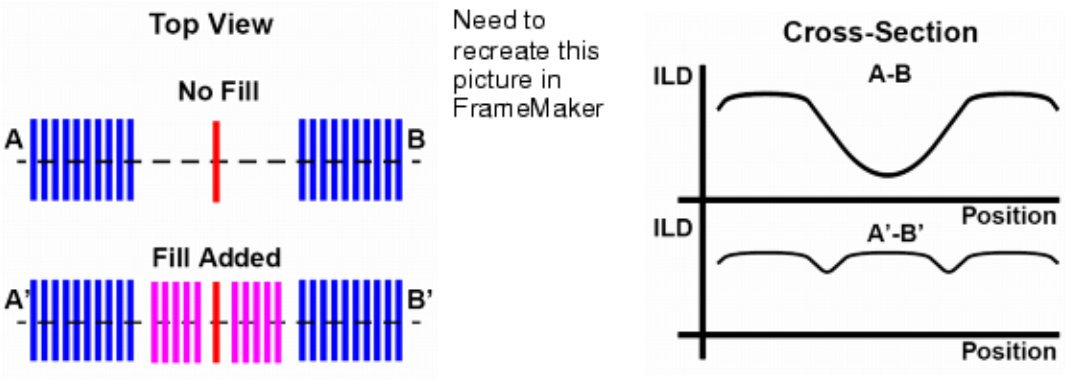


Figure 15-4 illustrates a side effect of the interaction: metal fill placement completely changes the profile of the CMP erosion. Notice, however, that even very regular metal fill still does not eliminate the uneven wear.

Figure 15-4. Metal Fill and CMP



CMP models only predict how thickness will be affected. They can be used in conjunction with in-die variation, which also calculates density effects on edge placement. Because having a full view of neighboring shapes is important to both techniques, transistor-level (flat) extraction is best. However, this requires large amounts of disk space to complete the extraction.

Prerequisites

Most manufacturing processes use chemical-mechanical polishing (CMP) to planarize the wafer surface after deposition. The effect of the CMP step is calculated using complex software which creates data files. The foundry may supply those files as part of a design kit.

If you have CMP modeling files in the Praesagus or VCMP format, you can have Calibre xACT use them to modify thickness for the conductors. When CMP data is used in conjunction with in-die variation, the CMP data is used instead of the in-die data.

Procedure

1. The foundry runs CMP modeling software and adds the data to the design kit.
2. The CAD group places the CMP files in a location accessible to the engineers running Calibre xACT.
3. The CAD group sets up the environment variables and files for running CMP modeling. See “Varying Thickness with CMP Files” for more information.
4. When appropriate, the engineer running Calibre xACT enables CMP input during extraction.

   CMP effects are sensitive to neighborhood density. Most of the time you would not benefit from including CMP data before placing metal fill.

Chapter 16
Calibre xACTView

Calibre xACTView is a 3D layout viewer used to visualize technology stack and design structures for integrated circuit designs in 2D and 3D views. It provides the ability to graphically view the different polygons processed for capacitance extraction as seen by Calibre xACT 3D.

This chapter describes the features of the Calibre xACTView tool.

Calibre xACTView Input **201**

Calibre xACTView Invocation **201**

Calibre xACTView GUI Features **205**

Calibre xACTView Session Window 205
FS3 Display Window 206
Layer Palette 215
Axes Window 217
Menus 218
Toolbars 230
Ruler 232

Calibre xACTView Input

Calibre xACTView accepts an FS3 file as input. The FS3 file is an intermediate file created by the Calibre xACT 3D field solver that contains the technology stack and design structure information.

An FS3 file, called *fs_1.fs3*, is temporarily created during extraction in the *pex.db* directory found in the directory name specified by the Mask SVDB Directory statement.

To write and save an FS3 file to a different directory use the PEX Report FS3 statement. Depending on the size of the design, multiple FS3 files may be created and numbered sequentially, *fs_1.fs3*, *fs_2.fs3*, ... , *fs_n.fs3*. The FS3 file may also be converted from another tool.

Calibre xACTView Invocation

The Calibre xACTView tool is invoked from the command line.

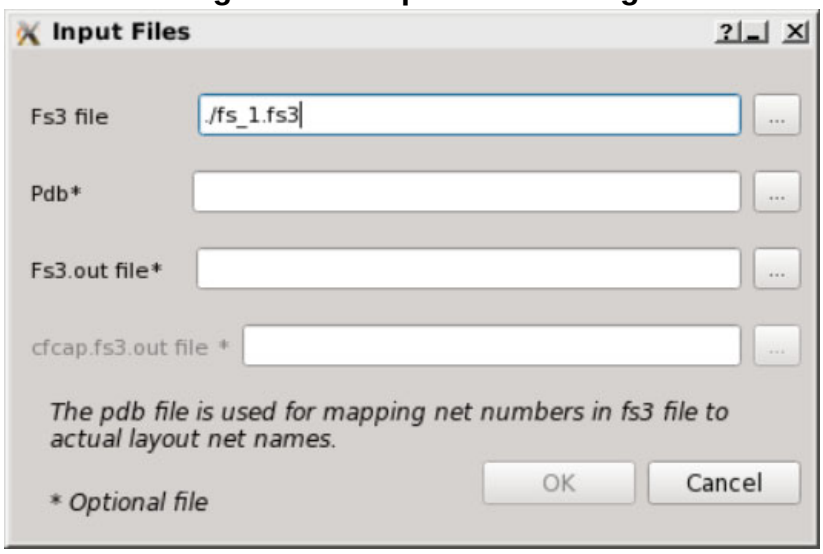
To invoke Calibre xACTView, enter the following at a shell prompt:

```
xactview [-fs3 filename.fs3 [-pdb pdb_name]
   [-fs3out fs3out_file [-cfcap cfcap_file]]
   [-layers layers]
   [-window x1 y1 x2 y2]]
```

- **-fs3 *filename* —** Supply the -fs3 *filename*.fs3 option to automatically load and display the FS3 file in Calibre xACTView.
- **-pdb *pdb_name* —** Optionally, supply the parasitic database (PDB) with the -pdb switch to display the actual layout net names instead of the net ids from the FS3 file in Calibre xACTView.
- **-fs3out *fs3out_file* [-cfcap *cfcap_file*] —** Specify the -fs3out and -cfcap options to optionally view coupling between shapes.
- **-layers *layers* —** Specify the -layers option with the -fs3 option to specify which layers are visible in the FS3 Display Window upon invocation of the Calibre xACTView tool.
- **-window *x1 y1 x2 y2* —** Specify the -window option with the -fs3 option to display a specified area of a large FS3 file upon invocation.

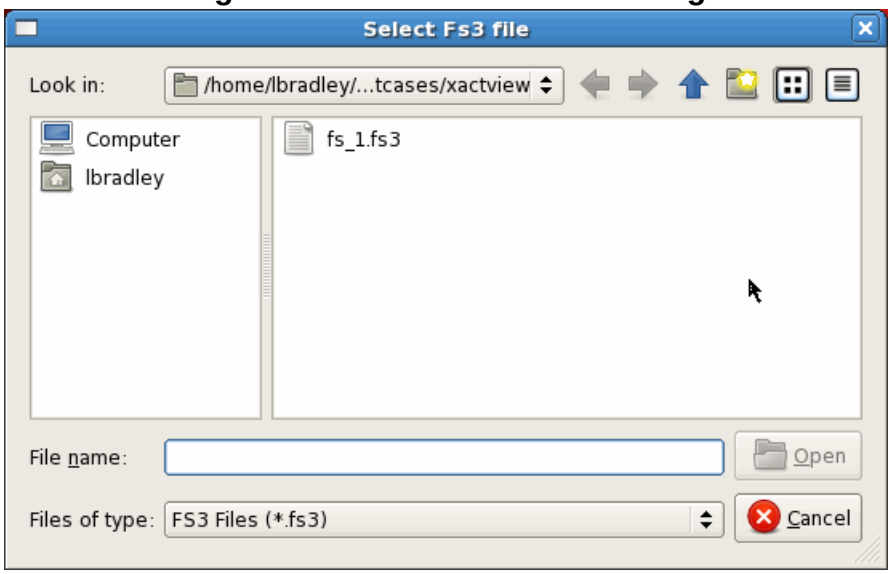
If you do not specify the -fs3 option, the Calibre xACTView tool opens with nothing displayed. Select **File > Open** from the menu or the **Open** tool bar button to display the Input Files dialog box (Figure 16-1).

Figure 16-1. Input Files Dialog



Enter the name of the FS3 file you want to view or click the "**...**" button to display the Select FS3 File dialog (Figure 16-2).

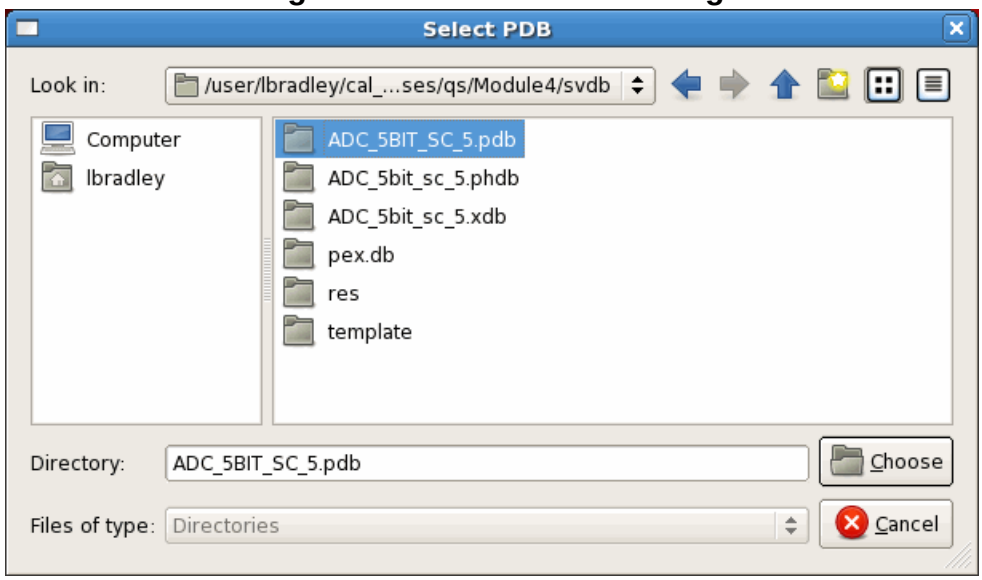
Figure 16-2. Select FS3 File Dialog



Optionally load the parasitic database (PDB) associated with the FS3 file to view net names instead of net ids for polygons. The parasitic database is generated in the PDB step of the Calibre xACT 3D PDB extraction flow. If you are using the Calibre xACT direct netlisting flow, then you must specify the PEX XACT PDB YES statement in your rule file in order to create the parasitic database.

Enter a parasitic database (PDB) name or click the "**...**" button to open the Select PDB dialog (Figure 16-3). For more information on displaying net information, see "Net and Shape Information Display" on page 207.

Figure 16-3. Select PDB Dialog



Optionally load the Fs3.out file typically named *fs_1.fs3.out*. This file is located in the parasitic database associated with the FS3 file. Use this file to view coupling capacitance between

polygons. A *cfcap.fs3.out* file can also be specified with the *fs_1.fs3.out* file to view correction file capacitance between polygons where available. Enter an Fs3.out file name or click the "**...**" button to view the Select fs3.out file dialog (Figure 16-4). Once an Fs3.out file name has been specified the cfcap.fs3.out file entry is enabled to allow specification of the associated correction file (Figure 16-5).

Figure 16-4. Select fs3.out file Dialog

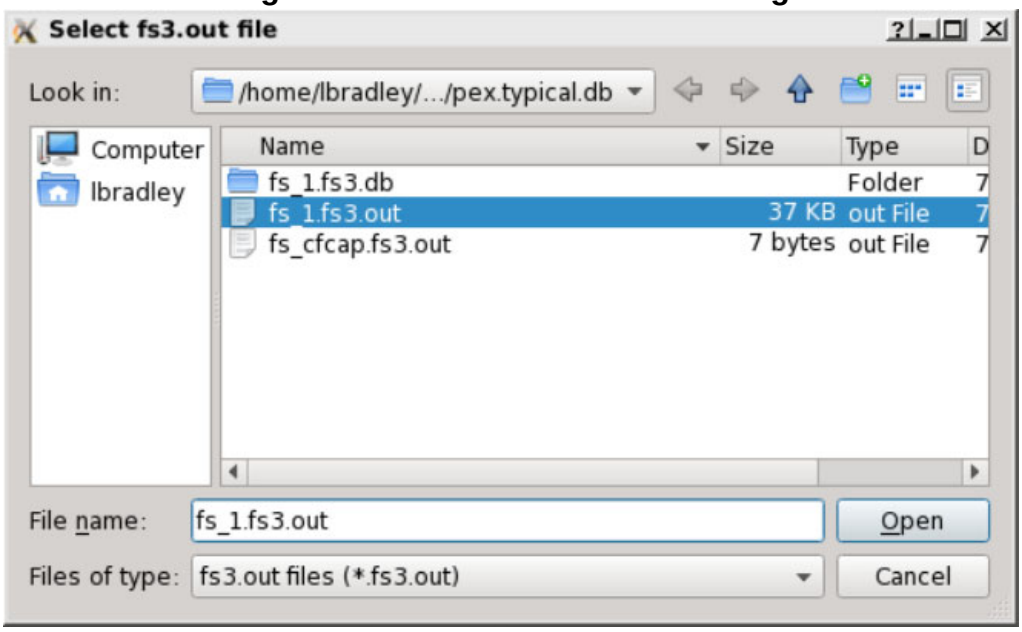
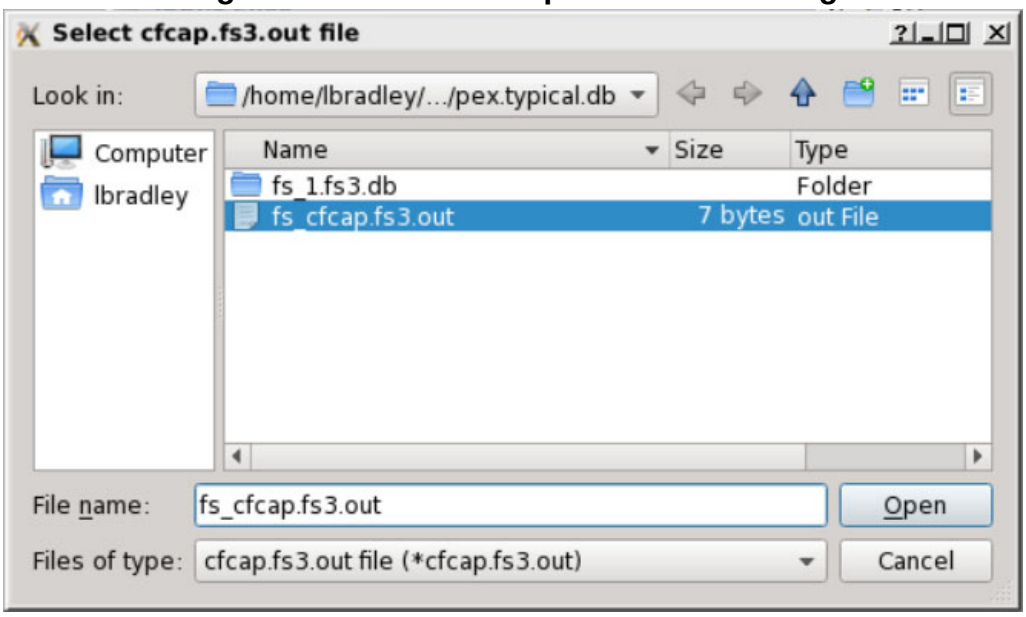


Figure 16-5. Select cfcap.fs3.out file Dialog



For more information on Calibre xACTView invocation options, see "xactview" on page 251. For details on where to find your FS3 file, see "Calibre xACTView Input" on page 201.

Calibre xACTView GUI Features

The Calibre xACT View graphical user interface (GUI) has several features.

These features include the following:

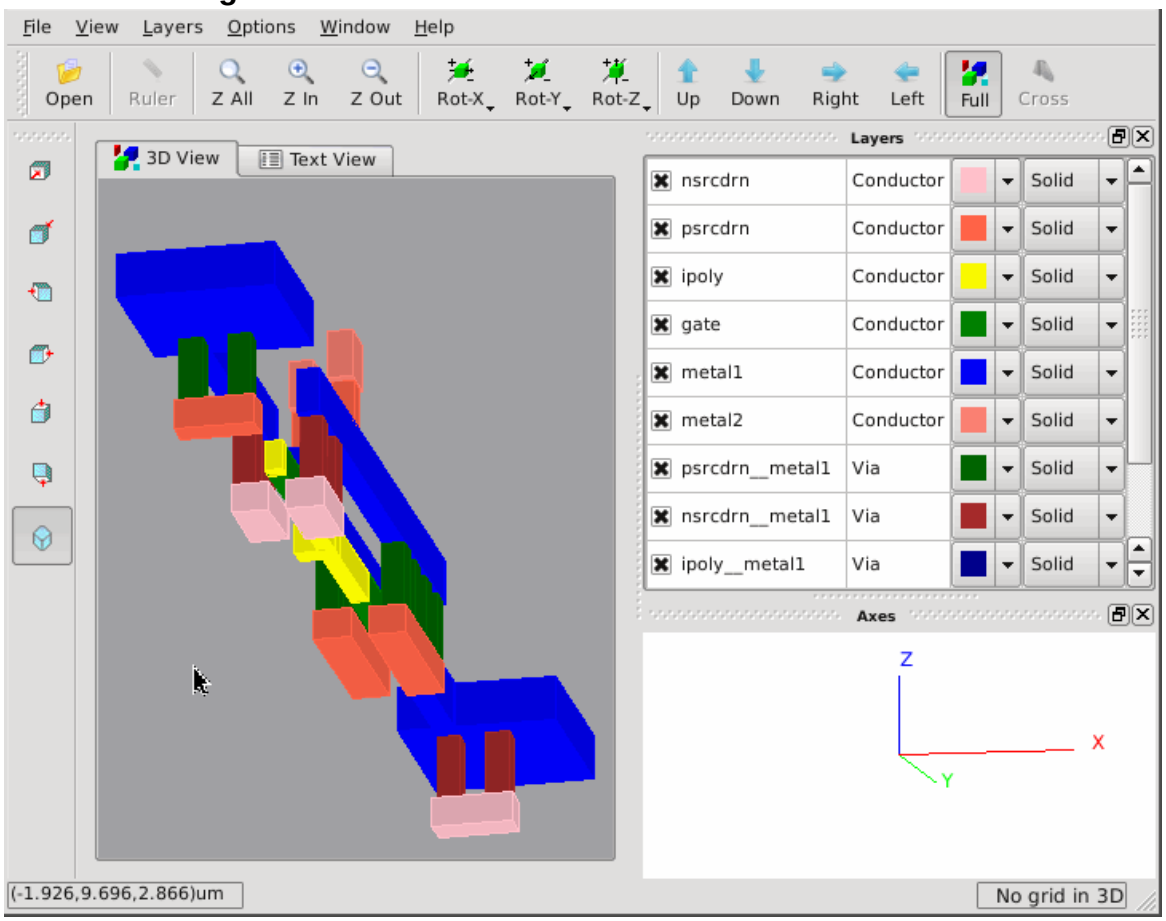
Calibre xACTView Session Window **205**
FS3 Display Window **206**
Layer Palette **215**
Axes Window **217**
Menus **218**
Toolbars **230**
Ruler **232**

Calibre xACTView Session Window

The Calibre xACTView session window opens upon invocation of the tool.

Figure 16-6 shows an example of the session window for Calibre xACTView.

Figure 16-6. Session Window For Calibre xACTView



The Calibre xACTView session window is divided into three main areas:

- FS3 Display Window — Here the FS3 data can be viewed graphically from the 3D View tab or as a text file from the **Text View** tab.
- Layer Palette — Use this palette to control the display of the layers shown in the graphical view.
- Axes Window — Use this view to track the current orientation of the structure shown in the 3D View window.

FS3 Display Window

The FS3 display window allows you to view the FS3 file data either graphically or as a text file. Select the **3D View** tab to graphically display the structure described in the FS3 file. Select the **Text View** tab to display the contents of the FS3 text file. Note that the Text View is only a viewer; the contents cannot be edited.

Display Controls

Options for display manipulation are accessible from the Calibre xACTView toolbars, menu picks, or hotkeys. You can specify the direction of the rotation for all axes.

The following views are available to control the display shown in the **3D View** tab's graphical view:

- Front view
- Back view
- Left view
- Right view
- Top view
- Bottom view
- 3D view

Pressing the left mouse button down while dragging the cursor across the display window automatically enables the 3D view and rotates the image.

The right mouse button can be used to zoom the image in and out. While pressing the right mouse button down:

- Dragging the cursor diagonally upwards right or left zooms out.
- Dragging the cursor diagonally downwards left or right zooms in.

You can pan the image in 2D or 3D view by:

- Using the arrow keys on your keypad.
- Using the Up, Down, Left, and Right toolbar buttons.
- Pressing the middle mouse button down while dragging the cursor across the display window.

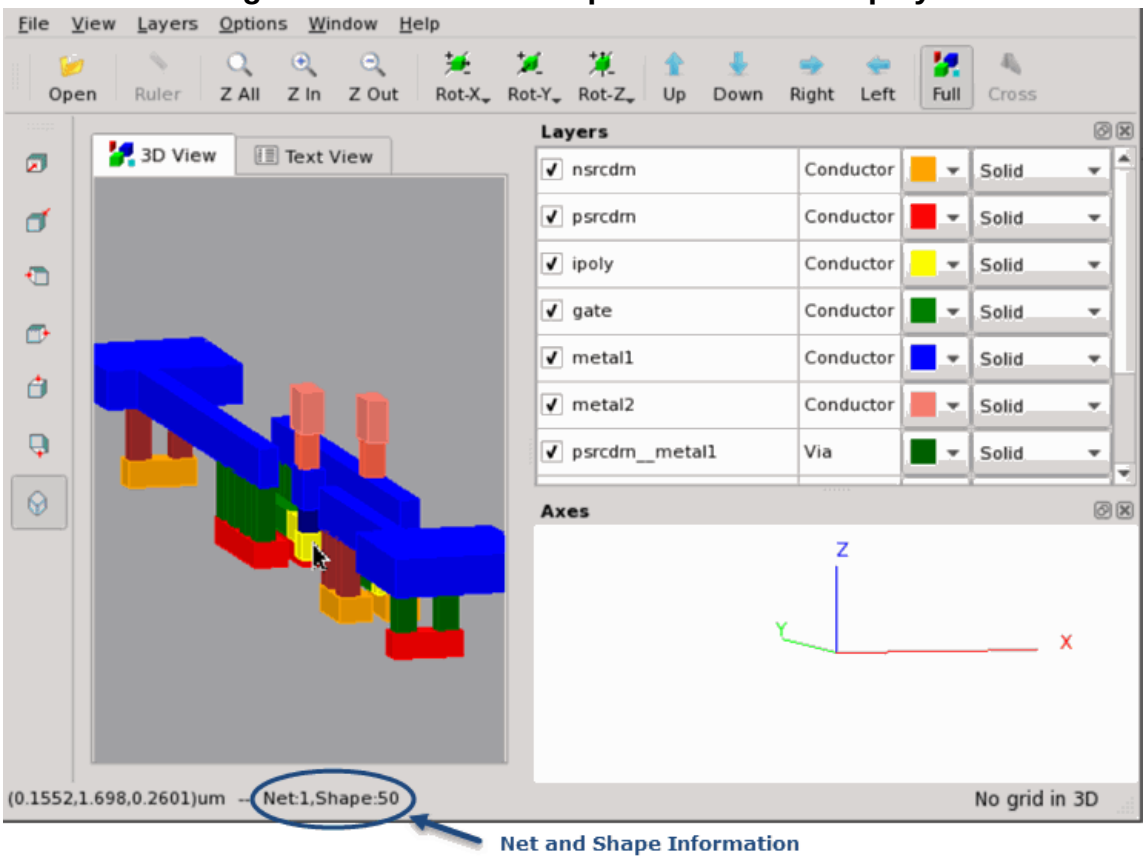
Net and Shape Information Display

Net and Shape information is shown upon request in the status bar (Figure 16-7). To display the net information associated with a polygon in the status bar, position the cursor over a polygon and press the control key. You may also hold the control key down and move the cursor over the different polygons to automatically view the net information for each shape as the cursor passes over them.

By default, Calibre xACTView displays the net id and shape number information found in the FS3 file. To display the net name and shape number, load the PDB associated with the design.

The Calibre xACTView tool uses the PDB file to map the net ids in the FS3 file to the actual layout net names. For more information on loading the PDB, see “Calibre xACTView Invocation” on page 201.

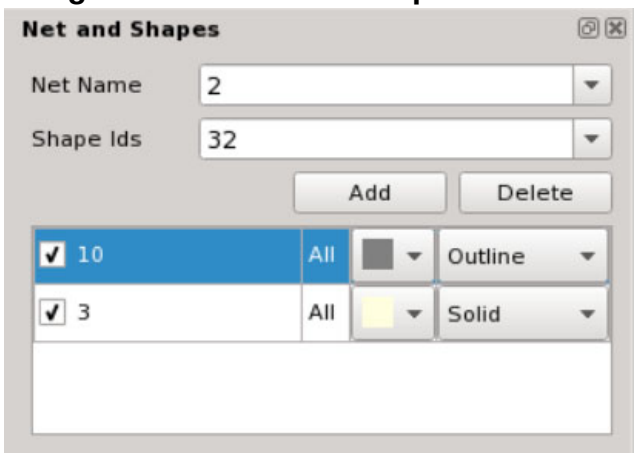
Figure 16-7. Net and Shape Information Display



Net and Shape Highlighting

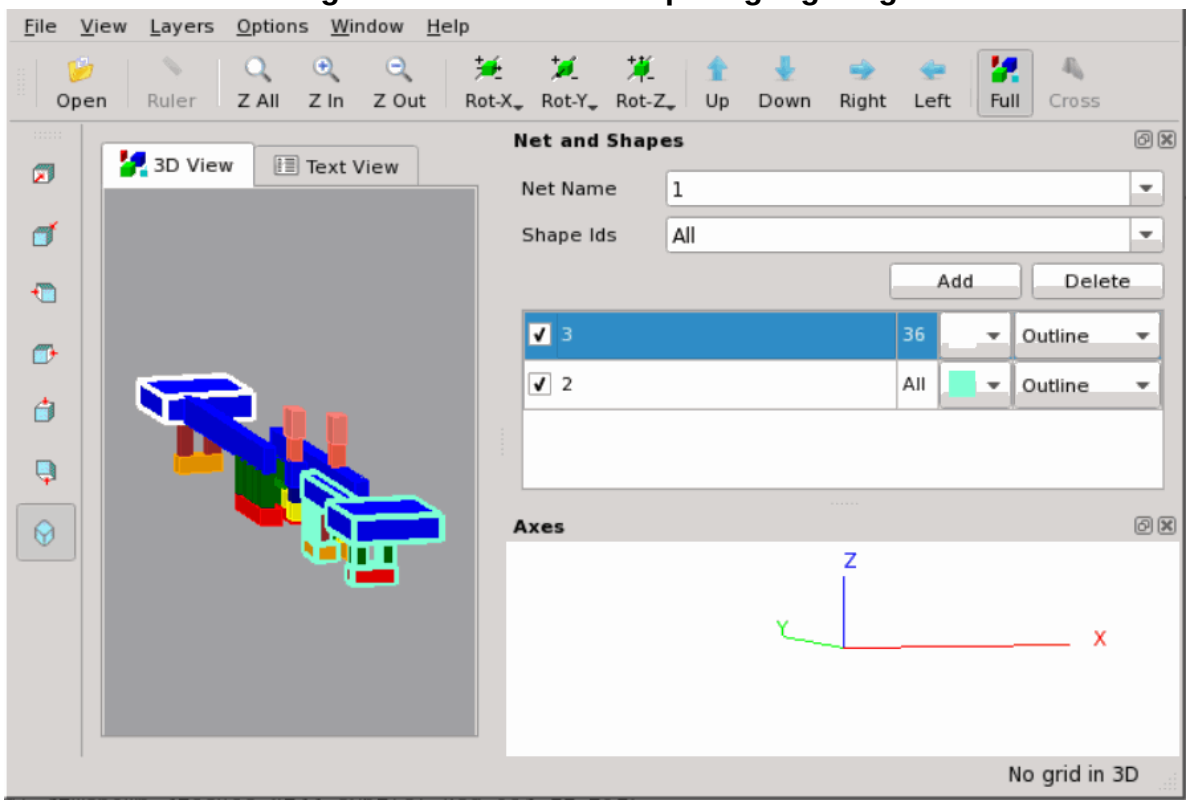
Net and shape highlighting is controlled by the Net and Shapes window (Figure 16-8).

Figure 16-8. Net and Shapes Window



Use this window to specify the net ids and their associated shapes that you want to highlight in the 3D View display window (Figure 16-9). Each net entry has controls that turn the highlighting on or off, change the color of the highlight, or control the transparency. By default, Calibre xACTView lists the net id and shape number information found in the FS3 file. To display the net name and shape number, load the PDB associated with the design.

Figure 16-9. Net and Shape Highlighting

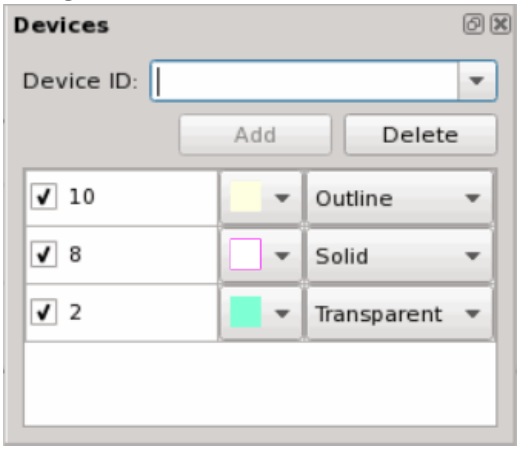


By default, the Net and Shapes window is hidden. Use the **Window > Net and Shapes** menu selection to display the Net and Shapes window. This window may be undocked from the application window or closed to increase the 3D viewing area.

Device Highlighting

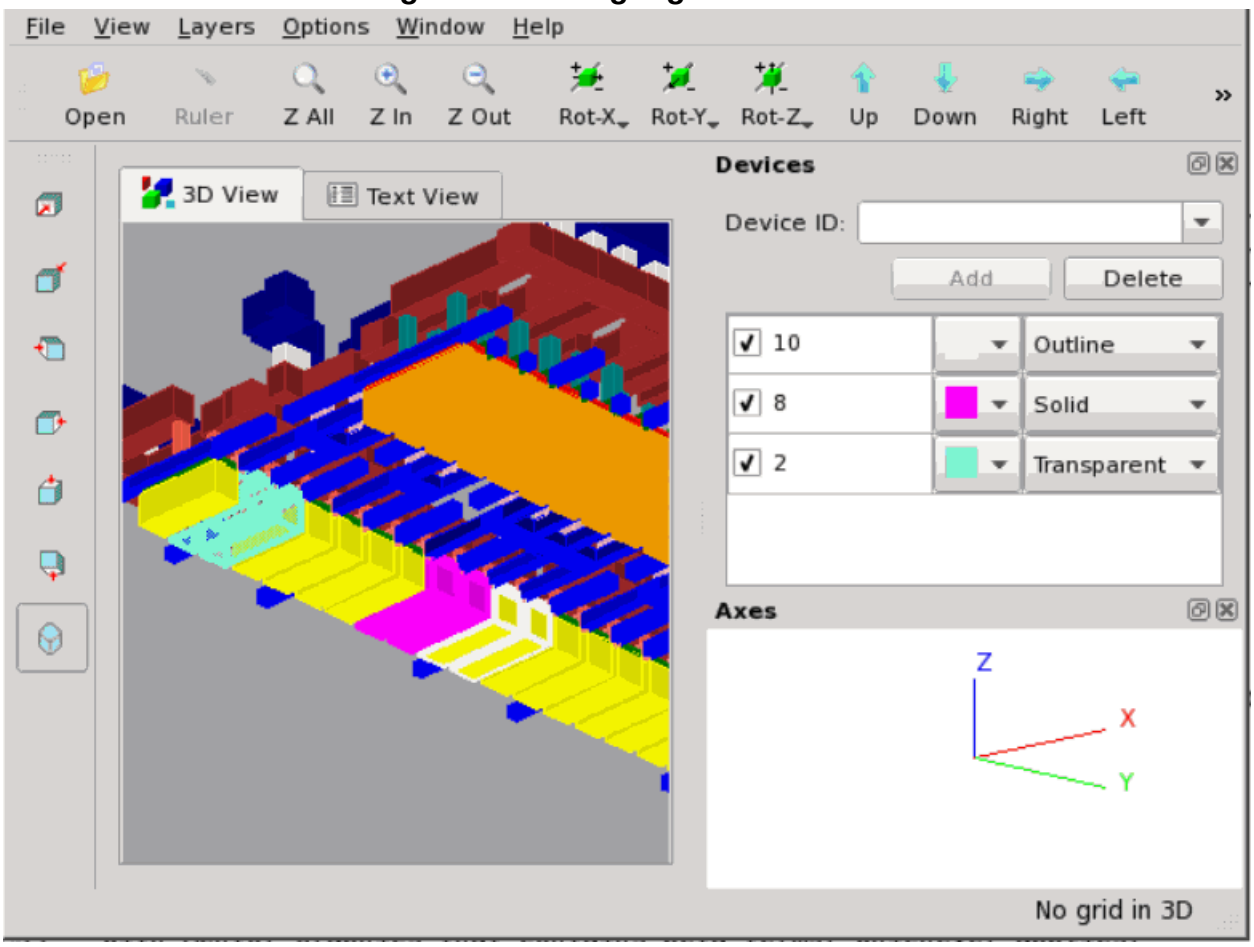
Device highlighting is controlled by the Devices window (Figure 16-10).

Figure 16-10. Devices Window



Use this window to specify the devices that you want to highlight in the 3D View display window (Figure 16-11). Each Device ID entry has controls that turn the highlighting on or off, change the color of the highlight, or control the transparency. By default, Calibre xACTView lists the device id information found in the FS3 file.

Figure 16-11. Highlighted Devices



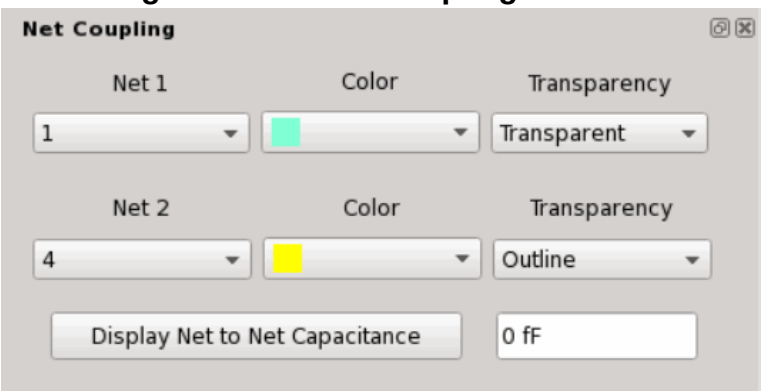
By default, the Devices window is hidden. Use the **Window > Devices** menu selection to display the Devices window. This window may be undocked from the application window or closed to increase the 3D viewing area.

Coupling Capacitance Highlighting

Coupling capacitance can be highlighted for nets, shapes, and layers. Select the highlighting method you would like to use from the **Window** menu. Only one method of highlighting at a time is allowed.

Coupling capacitance highlighting for nets is controlled by the Net Coupling window. Use this window to specify the net ids whose coupling you want to highlight in the 3D View display window (Figure 16-12). Select a net id for **Net 1** dropdown list and a net id for **Net 2** dropdown list, then click **Display Net to Net Capacitance** to highlight the net shapes that have coupling between them. There are controls that change the color of the highlight and control the transparency of Net 1 and Net 2. By default, Calibre xACTView lists the net id information found in the FS3 file. To display the net name, load the PDB associated with the design.

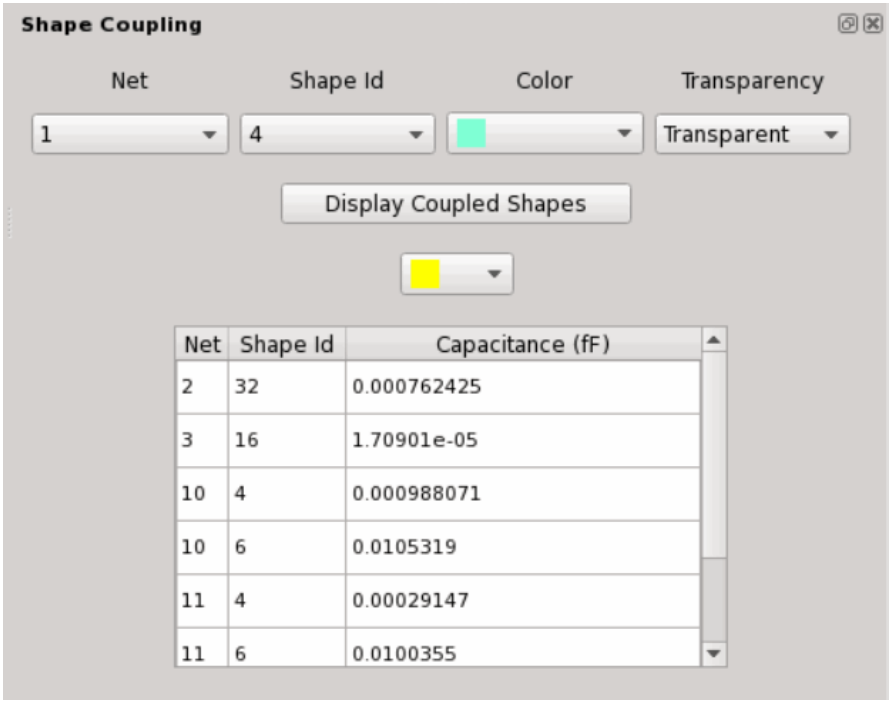
Figure 16-12. Net Coupling Window



By default, the Net Coupling window is hidden. Use the **Window > Net Coupling** menu selection to display the Net Coupling window. This window may be undocked from the application window or closed to increase the 3D viewing area.

Coupling capacitance highlighting for shapes is controlled by the Shape Coupling window. Use this window to highlight the coupling between a specified net shape, and all the shapes this net shape has coupling with in the 3D View display window (Figure 16-13). Select a net id for **Net** dropdown list and a shape id for **Shape Id** dropdown list, then click **Display Coupled Shapes** to highlight the shapes that have coupling between them. There are controls that change the color of the highlight and control the transparency of Net and Shape Id. There is only a color control for the coupled shapes. By default, Calibre xACTView lists the net id and shape number information found in the FS3 file. To display the net name and shape number, load the PDB associated with the design.

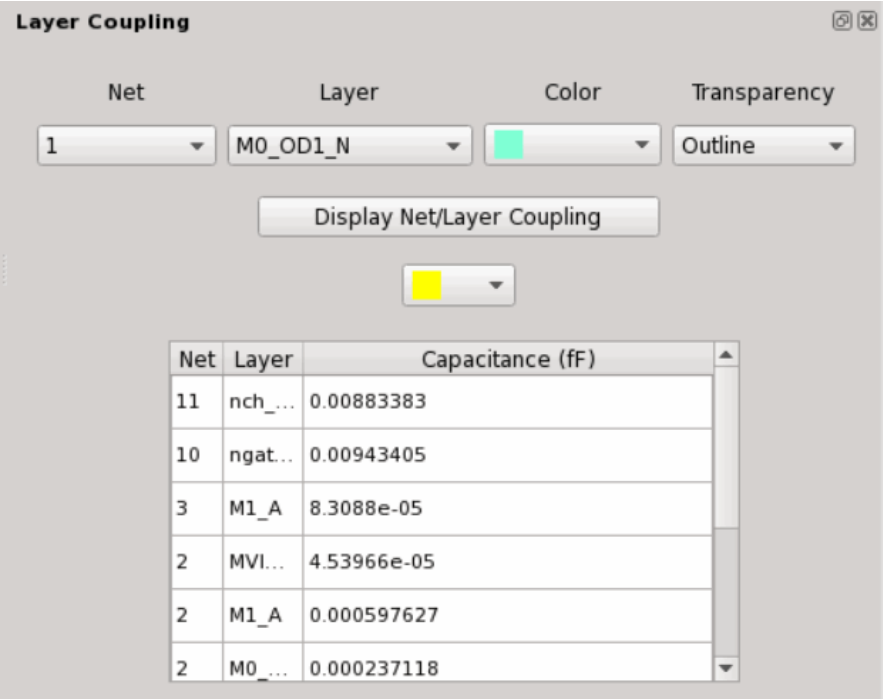
Figure 16-13. Shape Coupling Window



By default, the Shape Coupling window is hidden. Use the **Window > Shape Coupling** menu selection to display the Shape Coupling window. This window may be undocked from the application window or closed to increase the 3D viewing area.

Coupling capacitance highlighting for layers is controlled by the Layer Coupling window. Use this window to highlight the coupling between a specified net and layer, and all the layers this net layer has coupling with in the 3D View display window (Figure 16-14). Select a net id for **Net** dropdown list and a layer name for **Layer** dropdown list, then click **Display Net/Layer Coupling** to highlight the shapes that have coupling between them. There are controls that change the color of the highlight and control the transparency of Net and Layer. There is only a color control for the Net/Layer coupled shapes. By default, Calibre xACTView lists the net id information found in the FS3 file. To display the net name, load the PDB associated with the design.

Figure 16-14. Layer Coupling Window

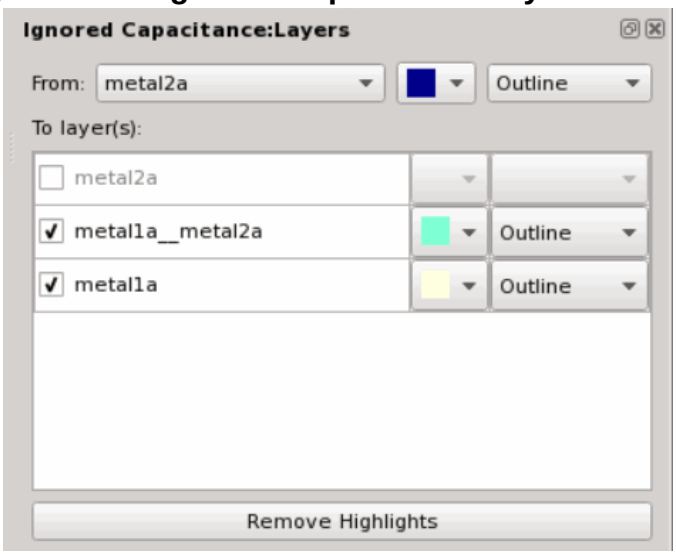


By default, the Layer Coupling window is hidden. Use the **Window > Layer Coupling** menu selection to display the Layer Coupling window. This window may be undocked from the application window or closed to increase the 3D viewing area.

Ignored Capacitance Highlighting

Ignored capacitance highlighting is controlled by the Ignored Capacitance:Layers window (Figure 16-15).

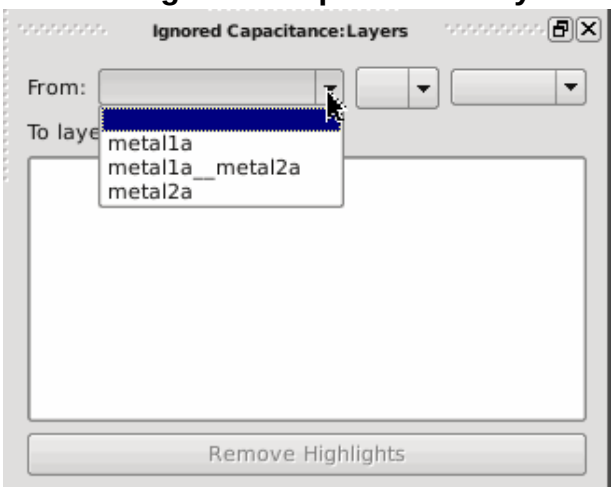
Figure 16-15. Ignored Capacitance:Layers Window



Use this window to specify the layers with ignored capacitance that you want to highlight in the 3D View display window (Figure 16-17). Each layer entry has controls that turn the highlighting on or off, change the color of the highlight, and control the transparency.

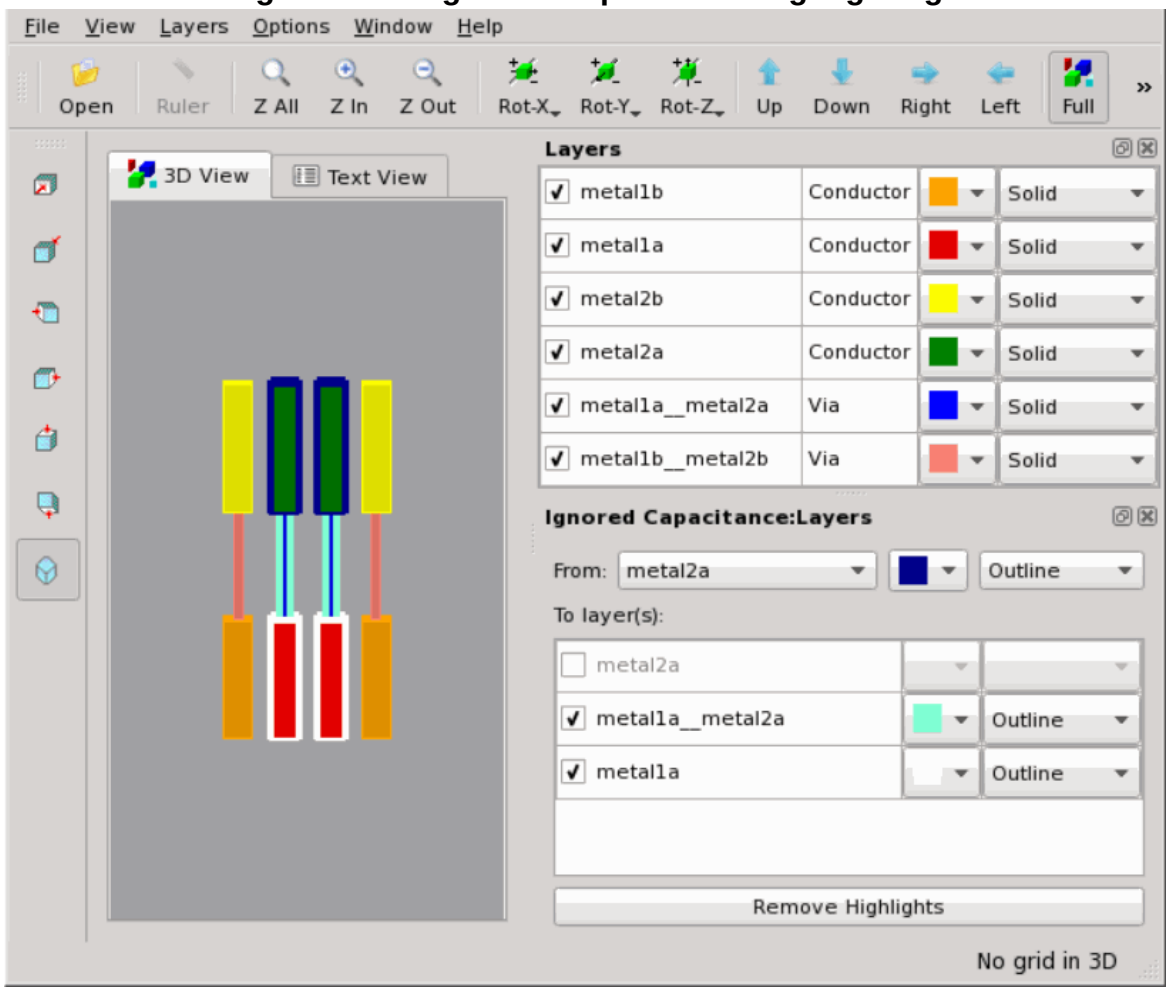
You can view ignored capacitance from one layer at a time. Select a layer name from the **From** layer dropdown list (Figure 16-16) to display the associated **To layer(s)** in the list box.

Figure 16-16. Ignored Capacitance:Layers From



Select the **Remove Highlights** button to clear the highlights from the display and clear the selected layers from the Ignored Capacitance:Layers window.

Figure 16-17. Ignored Capacitance Highlighting



Use the **Window > Ignored Capacitance:Layers** menu selection to toggle the display of the Ignored Capacitance:Layers window. This window may be undocked from the application window or closed to increase the 3D viewing area.

Layer Palette

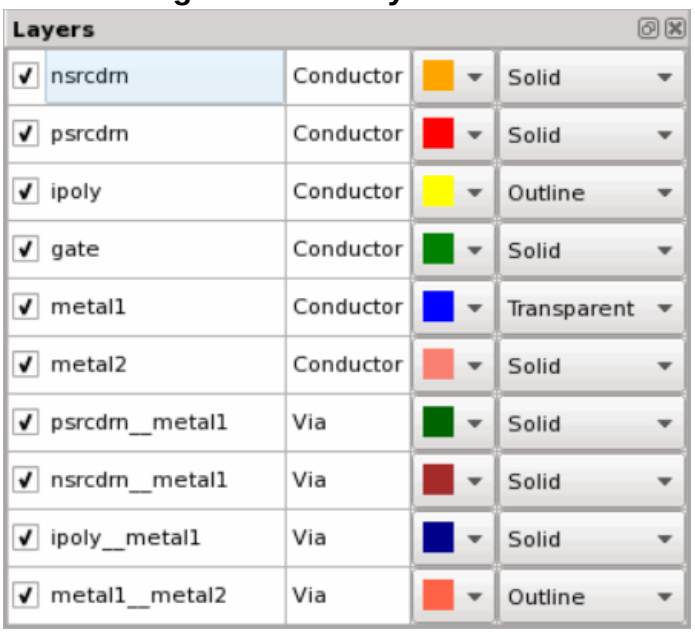
The Calibre xACTView layer palette controls the appearance of the layers for the displayed structure. The palette lists the layer names and types specified in FS3 file. The layer names are listed in the order they appear in the FS3 file. Use this palette to turn the display of specific layers on or off, change the color of a layer, or control the layer transparency.

The layer palette settings can be saved, loaded, or reset using controls found on the Options menu. For example, use the **Options > Save Layer Properties** menu selection to save these settings to a layer properties file. For more information on layer property controls, see the “Options Menu” on page 224.

The layer palette window is open by default when the tool is invoked. This window may be undocked from the application window or closed to increase the 3D viewing area. You may also use the **Window > Layers** menu selection to toggle the display of the layer palette.

Figure 16-18 shows the layer palette (Layers window) for Calibre xACTView.

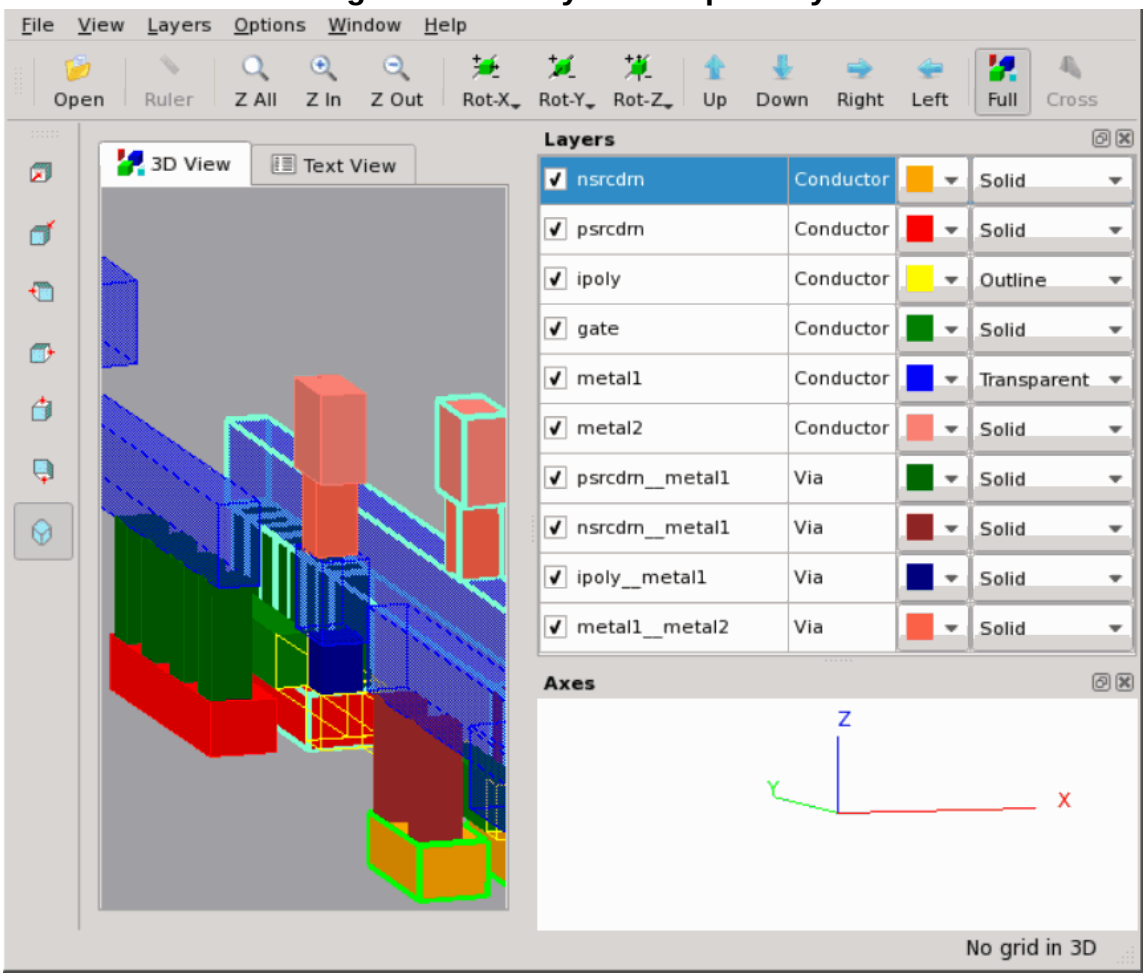
Figure 16-18. Layer Palette



The layer transparency dropdown list controls the display of each layer shape shown in the FS3 Display window. The transparency choices are:

- **Solid** — Displays the layers as solid shapes.
- **Outline** — Displays the layers as outlined shapes. Layer ipoly is shown as Outline in Figure 16-19.
- **Transparent** — Displays the layers as transparent shapes. Layer metal1 is shown as transparent in Figure 16-19.

Figure 16-19. Layer Transparency



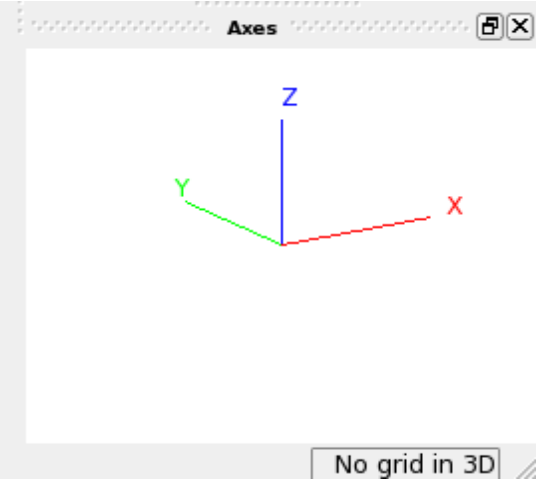
Axes Window

The Axes window displays the X, Y, and Z coordinate axes. This view tracks the orientation of the structure shown in the FS3 display window.

This window may be undocked from the application window or closed to increase the 3D viewing area. You may also use the **Window > Axes** menu selection to toggle the display of the Axes window.

Figure 16-20 shows the Axes window for Calibre xACTView.

Figure 16-20. Axes Window



Menus

The Calibre xACTView GUI has the following menus:

- **File** — A menu that contains the open, export, and exit commands.
- **View** — A menu that contains the view manipulation selections. These operations are also accessible from the horizontal and vertical toolbars.
- **Layers** — A menu that controls the types of layers shown in the 3D View window and Layers window.
- **Options** — A menu containing the **Preferences** menu selection.
- **Window** — A menu that controls the display of the Layers, Axes, Net and Shapes, Devices, and Ignored Capacitance:Layers windows. It also controls the display of the Net Coupling, Shape Coupling, and Layer Coupling windows if the coupling data is loaded.
- **Help** — A menu that contains the **About** menu selection. The **About** menu selection displays an informational dialog box containing the Calibre xACTView version information.

File Menu

The **File** menu is context sensitive. Figure 16-21 shows the **File** menu contents for Calibre xACTView when an FS3 file is opened.

Figure 16-21. File Menu

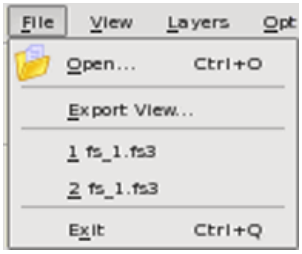


Figure 16-22 shows the **File** menu contents for Calibre xACTView when coupling capacitance information is available.

Figure 16-22. File Menu With Export Net/Layer Coupling Information

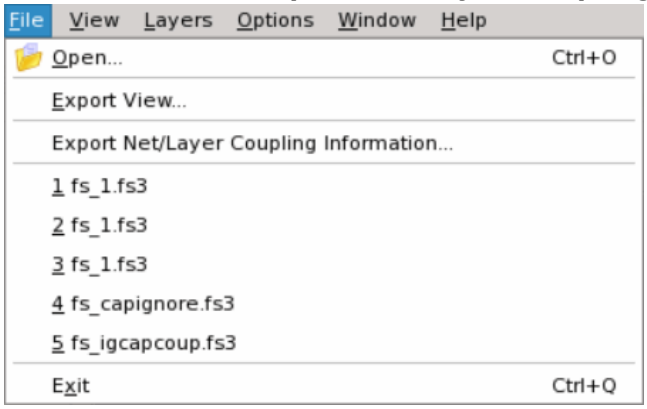


Table 16-1 describes the **File** menu contents.

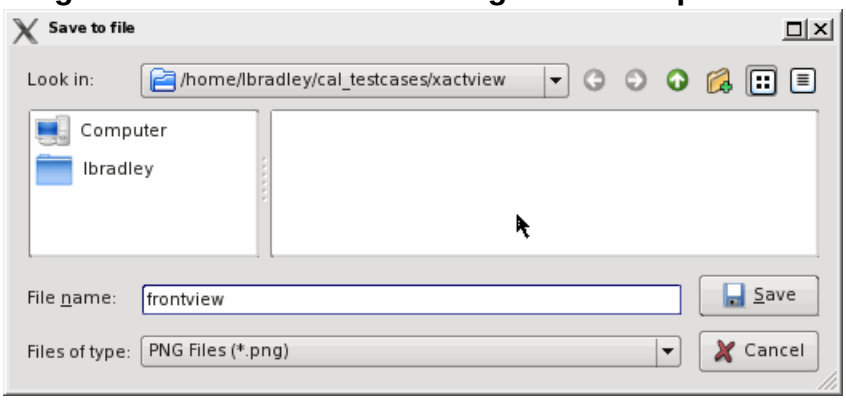
Table 16-1. File Menu Item Descriptions

Menu Item	Description
Open	Select **File > Open** to display the Input Files dialog box.
Export View	Select **File > Export View** to display the Save to file dialog box. Used to save the current 3D View image to a *.png* file.
Export Net/Layer Coupling Information	Select **File > Export Net/Layer Coupling Information** to display the Save to file dialog box. This menu selection is only available if coupling data is loaded. Used to save net/layer coupling data to a text file.
file history	A list of previously opened files. Each filename is preceded by a number. The file history saves up to the last five files opened.
Exit	Select **File > Exit** to close the Calibre xACTView tool.

"Calibre xACTView Invocation" on page 201 shows the Input Files dialog box. Use the Input Files dialog box to specify the FS3 file you want to display in the Calibre xACTView window. Optionally, you may also specify the PDB name to display the layout net names instead of the net id when selecting nets in the display.

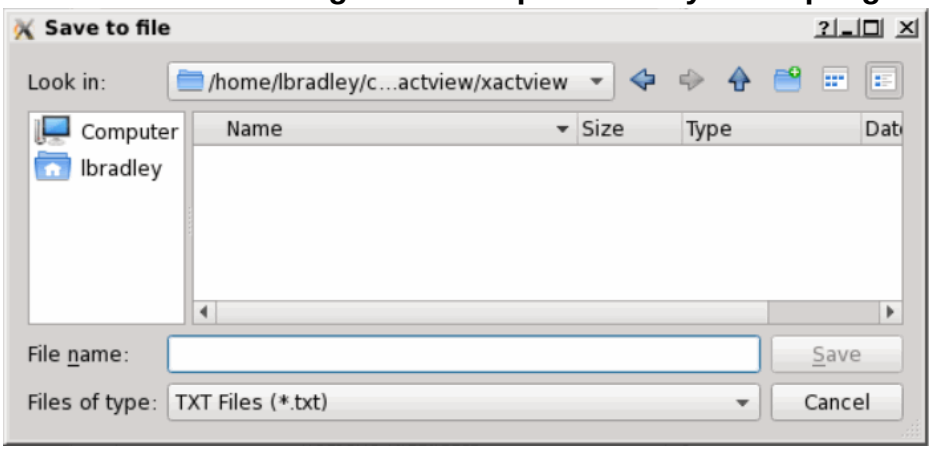
Figure 16-23 shows the Save to file dialog box used by Export View. Use this Save to file dialog box to export the current view to a PNG (**.png*) file.

Figure 16-23. Save to file Dialog Box for Export View



If coupling information is available, then the **File** menu includes the **Export Net/Layer Coupling Information** option. Figure 16-24 shows the Save to file dialog box displayed by **Export Net/Layer Coupling Information** menu selection. Use this Save to file dialog box to export the net/coupling information to a text (**.txt*) file.

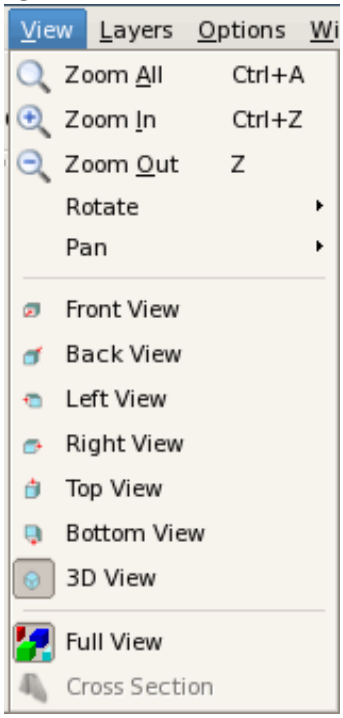
Figure 16-24. Save to file Dialog Box for Export Net/Layer Coupling Information



View Menu

Figure 16-25 shows the **View** menu contents for Calibre xACTView.

Figure 16-25. View Menu



The **View** menu contains selections that control the display of the structure shown in the 3D View window. Most of these selections can also be found on the toolbars. For more information on the toolbars, see “Toolbars”.

Table 16-2 describes the **View** menu contents.

Table 16-2. View Menu Item Descriptions

Menu Item	Description
Zoom All	Select**View** > **Zoom All** to display the entire structure in the 3D View window.
Zoom In	Select **View** > **Zoom In** to enlarge the image shown in the 3D View window by 20% increments.
Zoom Out	Select **View** > **Zoom Out** to shrink the image shown in the 3D View window by 20% increments.
Rotate	Select **View** > **Rotate** to display a cascading menu with the selections **X-axis**, **Y-axis**, and **Z-axis**.
Rotate > **X-axis**	Select **View** > **Rotate** > **X-axis** to display a cascading menu with the selections **Clockwise** and **Counter-Clockwise**. The current rotation mode is indicated by a dot to the left of the menu item.
Rotate > **X-axis** > **Clockwise**	Select **View** > **Rotate** > **X-axis** > **Clockwise** to set the rotation mode of the image shown in the 3D View window to clockwise along the X axis.

Table 16-2. View Menu Item Descriptions (cont.)

Menu Item	Description
Rotate > X-axis > Counter-Clockwise	Select **View > Rotate > X-axis > Counter-Clockwise** to set the rotation mode of the image shown in the 3D View window to counterclockwise along the X axis.
Rotate > Y-axis	Select **View > Rotate > Y-axis** to display a cascading menu with the selections **Clockwise** and **Counter-Clockwise**. The current rotation mode is indicated by a dot to the left of the menu item.
Rotate > Y-axis > Clockwise	Select **View > Rotate > Y-axis > Clockwise** to set the rotation mode of the image shown in the 3D View window to clockwise along the Y axis.
Rotate > Y-axis > Counter-Clockwise	Select **View > Rotate > Y-axis > Counter-Clockwise** to set the rotation mode of the image shown in the 3D View window to counterclockwise along the Y axis.
Rotate > Z-axis	Select **View > Rotate > Z-axis** to display a cascading menu with the selections **Clockwise** and **Counter-Clockwise**. The current rotation mode is indicated by a dot to the left of the menu item.
Rotate > Z-axis > Clockwise	Select **View > Rotate > Z-axis > Clockwise** to set the rotation mode of the image shown in the 3D View window to clockwise along the Z axis.
Rotate > Z-axis > Counter-Clockwise	Select **View > Rotate > Z-axis > Counter-Clockwise** to set the rotation mode of the image shown in the 3D View window to counterclockwise along the Z axis.
Pan	Select **File > Pan** to display a cascading menu with the selections **Up**, **Down**, **Left**, and **Right**.
Front View	Select **View > Front View** to display the front of the structure in 2D.
Back View	Select **View > Back View** to display the back of the structure in 2D.
Left View	Select **View > Left View** to display the left side of the structure in 2D.
Right View	Select **View > Right View** to display the right side of the structure in 2D.
Top View	Select **View > Top View** to display the top of the structure in 2D.
Bottom View	Select **View > Bottom View** to display the bottom of the structure in 2D.

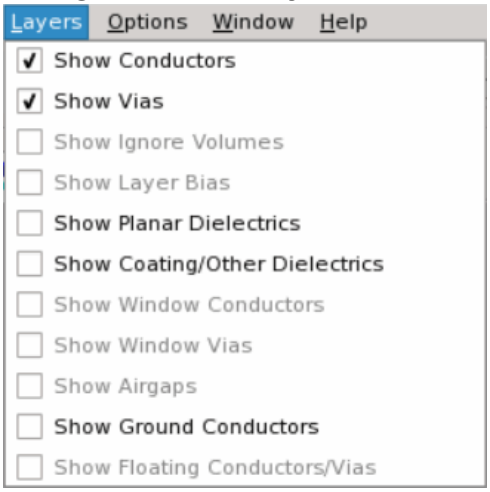
Table 16-2. View Menu Item Descriptions (cont.)

Menu Item	Description
3D View	Select **View > 3D View** to display a 3D view of the structure.
Full View	Select **View > Full View** to return to the structure display to its entirety after a cross section operation.
Cross Section	Select **View > Cross Section** to display the Cross section view tools in the toolbar and initiate the creation of a cross section.

Layers Menu

Figure 16-26 shows the **Layers** menu contents for Calibre xACTView.

Figure 16-26. Layers Menu



The **Layers** menu controls which layers are shown in the 3D View window and the Layers window. Toggle the display of a layer by selecting the menu item. A check mark or x next to the menu item indicates that layer type is currently shown in the display. Dimmed Layer menu items are layers that do not exist in the open FS3 file. For example, Show Window Vias is dimmed because the open FS3 file does not contain any Window Via layers.

Table 16-3 describes the **Layers** menu contents.

Table 16-3. Layers Menu Item Descriptions

Menu Item	Description
Show Conductors	Select **Layers > Show Conductors** to toggle the display of conductor layers in the displayed structure and the Layers window.
Show Vias	Select **Layers > Show Vias** to toggle the display of via layers in the displayed structure and the Layers window.

Table 16-3. Layers Menu Item Descriptions (cont.)

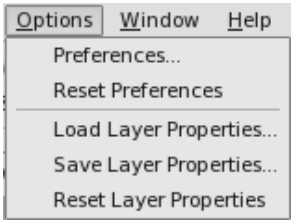
Menu Item	Description
Show Ignore Volumes	Select **Layers > Show Ignore Volumes** to toggle the display of ignore volume layers in the displayed structure and the Layers window.
Show Layer Bias	Select **Layers > Show Layer Bias** to toggle the display of layer bias layers in the displayed structure and the Layers window.
Show Planar Dielectrics	Select **Layers > Show Planar Dielectrics** to toggle the display of planar dielectric layers in the displayed structure and the Layers window.
Show Coating/Other Dielectrics	Select **Layers > Show Coating/Other Dielectrics** to toggle the display of coating or other dielectric layers in the displayed structure and the Layers window.
Show Window Conductors	Select **Layers > Show Window Conductors** to toggle the display of window conductor layers in the displayed structure and the Layers window.
Show Window Vias	Select **Layers > Show Window Vias** to toggle the display of window vias layers in the displayed structure and the Layers window.
Show Airgaps	Select **Layers > Show Airgaps** to toggle the display of airgap layers in the displayed structure and the Layers window.
Show Ground Conductors	Select **Layers > Show Ground Conductors** to toggle the display of ground conductor layers in the displayed structure and the Layers window.
Show Floating Conductors/Vias	Select **Layers > Show Floating Conductors/Vias** to toggle the display of floating conductor and floating via layers in the displayed structure and the Layers window.

By default, extracted layers display as solid shapes. Window conductors and window vias display as outlined shapes. Shapes defined with PEX Fieldsolver Boundary and PEX Fieldsolver Cell_array statements are window shapes and display as outlined shapes.

Options Menu

Figure 16-27 shows the **Options** menu contents for Calibre xACTView.

Figure 16-27. Options Menu



Select **Options** > **Preferences** to display the Preferences dialog box. Use the Preferences dialog box to set Ruler, Grid, Cross Section, and View preferences. Select **Options** > **Reset Preferences** to reset all preferences to their defaults.

Figure 16-28 shows the **Ruler** tab in the **Preferences** dialog box. Use this dialog to control the appearance and behavior of the ruler.

Figure 16-28. Preferences Dialog Box - Ruler Tab

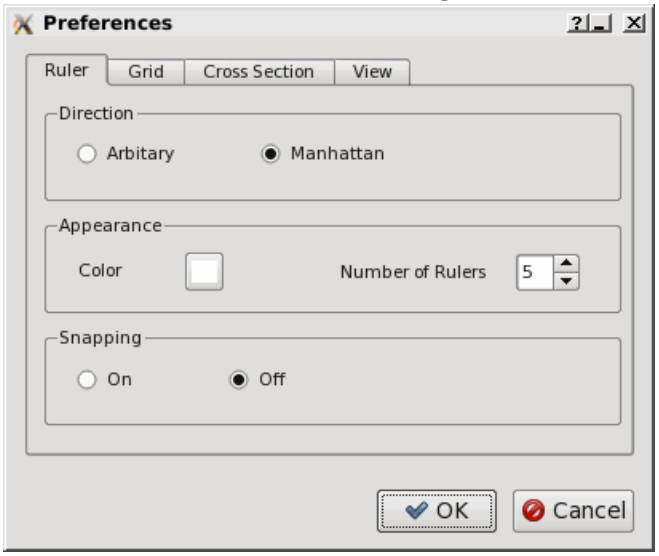


Figure 16-29 shows the **Grid** tab in the Preferences dialog box. Use this dialog to control the appearance and positioning of the grid.

Figure 16-29. Preferences Dialog Box - Grid Tab

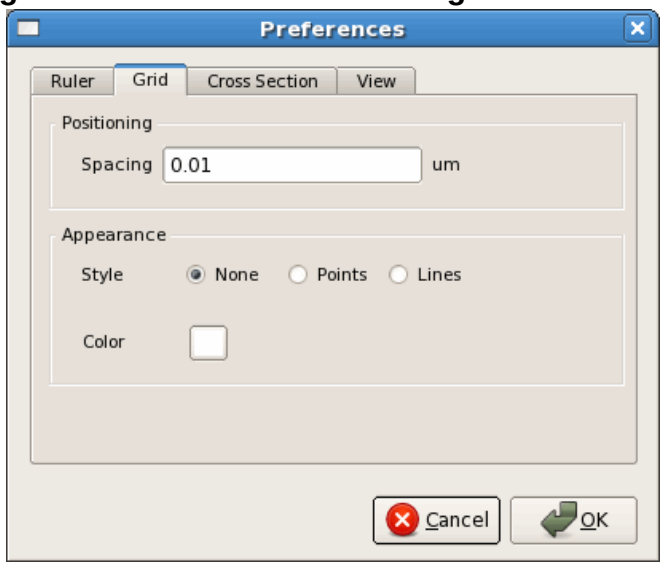


Figure 16-30 shows the **Cross Section** tab in the Preferences dialog box. Use this dialog to set the color of the cutline and the step value used when tuning the position of the cutline.

Figure 16-30. Preferences Dialog Box - Cross Section Tab

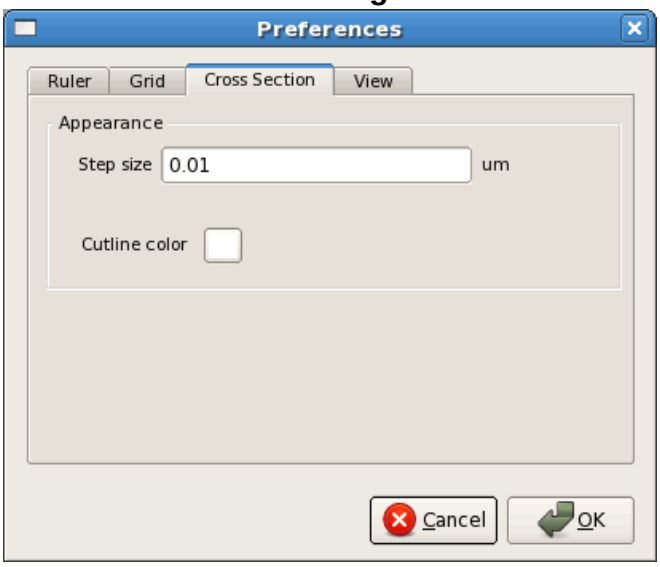
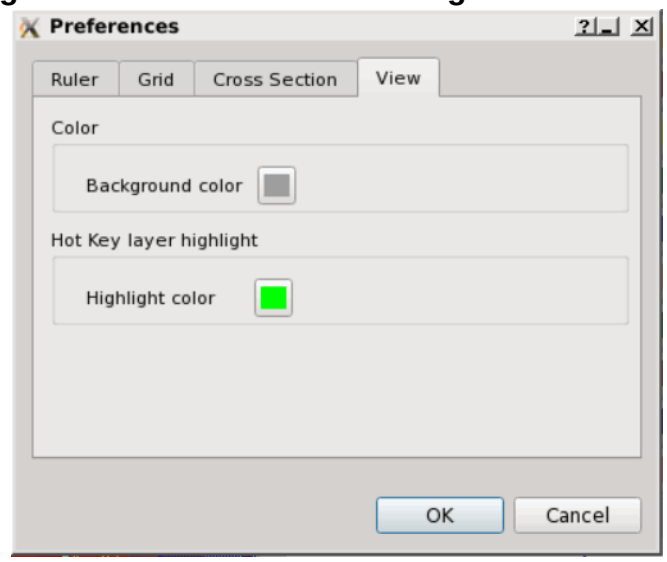


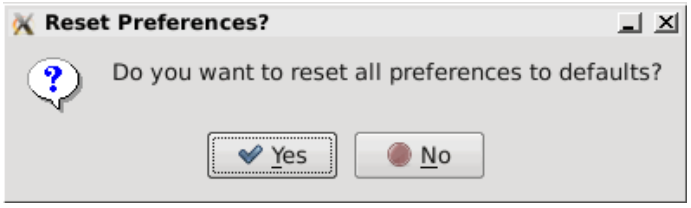
Figure 16-31 shows the **View** tab in the Preferences dialog box. Use this dialog to set the background color for the 3D View display window.

Figure 16-31. Preferences Dialog Box - View Tab



Select **Options** > **Reset Preferences** to reset the preferences to the tool defaults (Figure 16-32).

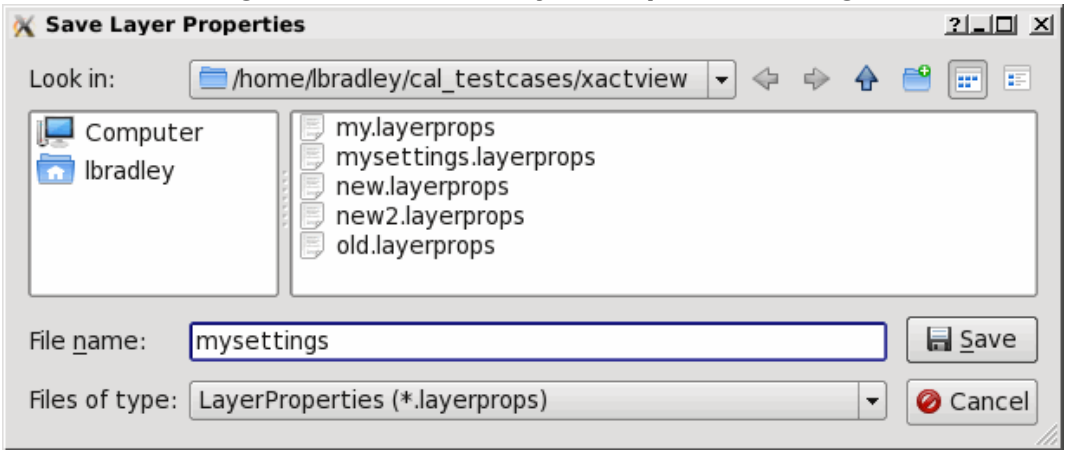
Figure 16-32. Reset Preferences



The layer properties file is an ASCII text file created with the Calibre xACTView tool that contains the layer names, color, and transparency settings defined in the layer palette for each displayed layer. Use the layer property controls on the **Options** menu to optionally create or save, load, or reset your layer palette settings to the layer properties file (*.*layerprops*).

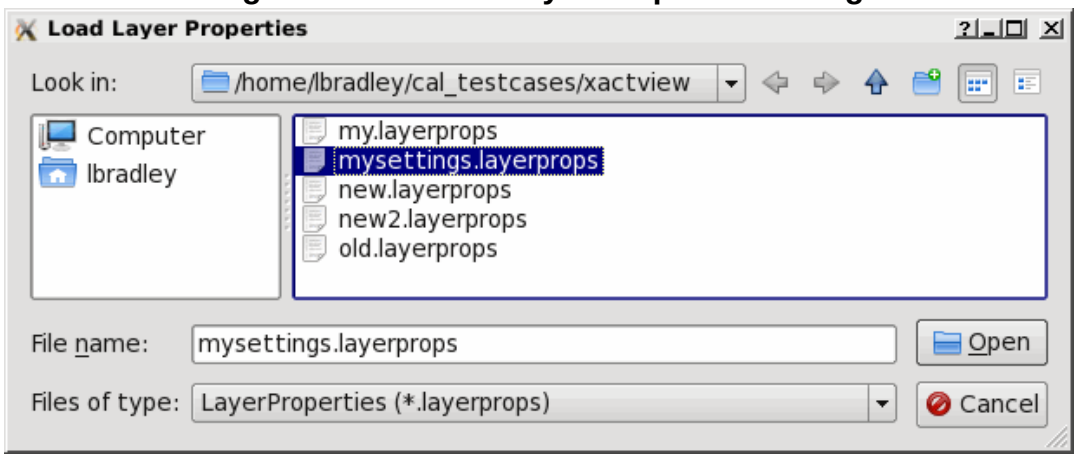
Select **Options** > **Save Layer Properties** to display the **Save Layer Properties** dialog (Figure 16-33). To save the color and transparency settings for all layers currently shown in the layer palette to a layer properties file, enter a file name and choose **Save**. If a file exists by the name specified in this field, you are prompted to overwrite the file.

Figure 16-33. Save Layer Properties Dialog



Select **Options > Load Layer Properties** to display the Load Layer Properties dialog (Figure 16-34). To load previously saved layer palette settings, select a layer property file name (**.layerprops*) and choose **Open**. The color and transparency settings for displayed layers in the layer palette are updated in the layer palette and 3D View display window.

Figure 16-34. Load Layer Properties Dialog



Select **Options > Reset Layer Properties** to reset all palette display settings to the tool defaults.

Window Menu

The **Window** menu contents change depending on the FS3 file information that is loaded. Figure 16-35 shows the **Window** menu contents for Calibre xACTView when only an FS3 file has been opened. **Ignored Capacitance Layers** menu item is dimmed when ignored layer data is not available in the open FS3 file.

Figure 16-35. Window Menu

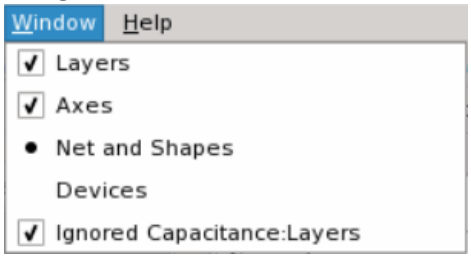
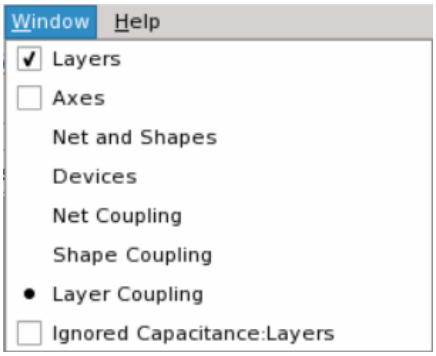


Figure 16-36 shows the **Window** menu contents for Calibre xACTView when an FS3 file has been opened with an *FS3.out* file and a Cfcap (*fs_cfcap.fs3.out*) file.

Figure 16-36. Window Menu With Coupling

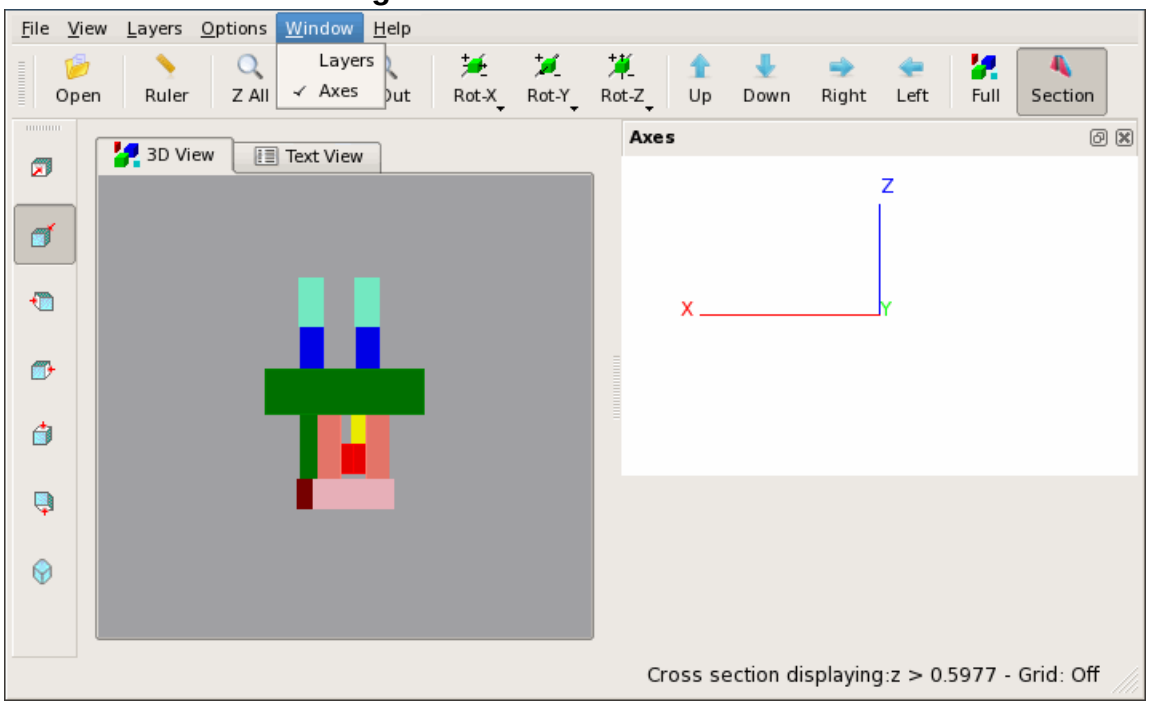


The **Window** menu toggles the display of the Layers, Axes, and Ignored Capacitance: Layers windows. Change the window display by selecting the menu item. A check mark or x next to the menu item indicates that window is currently shown. No check mark or x indicates the window is hidden.

The **Window** menu also contains radio buttons used to control the display the Net and Shapes, or Devices windows. Use the Net and Shapes, and Devices windows to highlight structures in the display window. For more information on highlighting structures, see "Net and Shape Highlighting" or "Device Highlighting".

Figure 16-37 shows the Calibre xACTView tool with its Layers, Ignored Capacitance Layers, Devices, and Net and Shapes windows hidden.

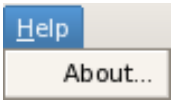
Figure 16-37. Hidden Windows



Help Menu

Figure 16-38 shows the **Help** menu contents for Calibre xACTView.

Figure 16-38. Help Menu



Select **Help** > **About** to display the About Calibre xACTView dialog box. This dialog box displays the tools version information.

Toolbars

The Calibre xACTView toolbars provide quick access to the most commonly used operations. By default the application window contains a horizontal toolbar and a vertical toolbar. These toolbars may be undocked from the main application window or moved within the window allowing you to customize your view.

Figure 16-39 shows the horizontal toolbar for Calibre xACTView.

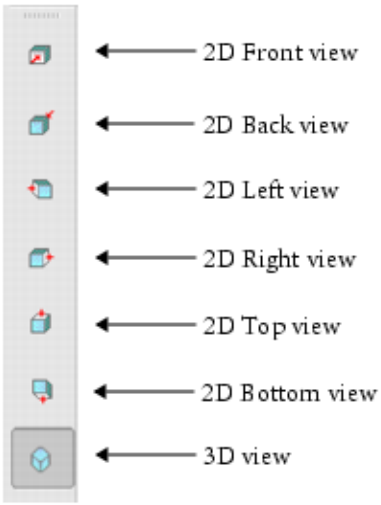
Figure 16-39. Horizontal Toolbar

The tools on the horizontal toolbar include:

- **Open** — Displays the Input Files dialog. Use this dialog to select and open an FS3 file.
- **Ruler** — Enables Ruler Mouse Mode. This item is available when you are in one of the 2D views. Use this tool to do measurements of your structure.
- **Z All** — Zoom all displays the entire structure in the 3D View window.
- **Z In** — Zoom in enlarges the image shown in the 3D View window by 20% increments.
- **Z Out** — Zoom out shrinks the image shown in the 3D View window by 20% increments.
- **Rot-X** — Rotates the image in small increments along the x-axis. The direction of the rotation can be set to either Clockwise or Counter-Clockwise from the **View > Rotate > X-axis** menu.
- **Rot-Y** — Rotates the image in small increments along the y-axis. The direction of the rotation can be set to either Clockwise or Counter-Clockwise from the **View > Rotate > Y-axis** menu.
- **Rot-Z** — Rotates the image in small increments along the z-axis. The direction of the rotation can be set to either Clockwise or Counter-Clockwise from the **View > Rotate > Z-axis** menu.
- **Up** — Pans the window view north.
- **Down** — Pans the window view south.
- **Right** — Pans the window view west.
- **Left** — Pans the window view east.
- **Full** — Restores the structure to its original form. Use this button to return to the full structure display after a cross section operation.
- **Cross** — Displays the Cross section view tools in the toolbar and initiates the creation of a cross section. This button is accessible in the 2D view only. The entries for the Cross section view tools are disabled until a cut line is drawn. Once the cut line is drawn, select the **Apply** button to make the cut and display the cross section.

Figure 16-40 shows the vertical toolbar for Calibre xACTView.

Figure 16-40. Vertical Toolbar



The tools on the vertical toolbar include:

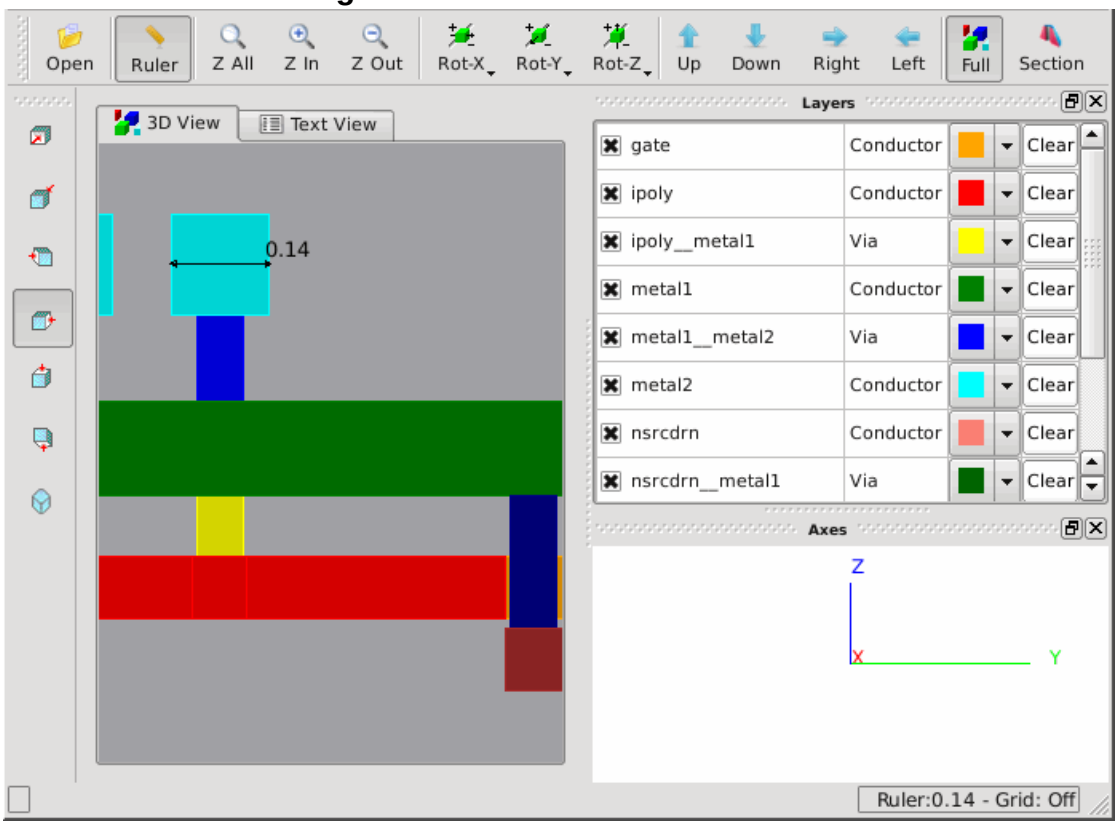
- **Front view** — Displays the front of the structure in 2D.
- **Back view** — Displays the back of the structure in 2D.
- **Left view** — Displays the left side of the structure in 2D.
- **Right view** — Displays the right side of the structure in 2D.
- **Top view** — Displays the top of the structure in 2D.
- **Bottom view** — Displays the bottom of the structure in 2D.
- **3D view** — Displays the 3D view of the structure.

Ruler

Ruler Mouse Mode is available when you are in one of the 2D views. Use this tool to measure your structure in microns (um).

Figure 16-41 shows how the ruler measurements are displayed in both the 3D View display window and the status bar.

Figure 16-41. Ruler Measurement



The **Ruler** tab in the Preferences dialog box controls the appearance and behavior of the ruler (see Figure 16-28). Use the **Number of Rulers** option in this dialog to specify the number of rulers that can be drawn in the display; the minimum setting is 1 and the maximum setting is 10. The following dialog appears when the maximum number of rulers have been drawn:

Figure 16-42. Info Dialog for Ruler Limit

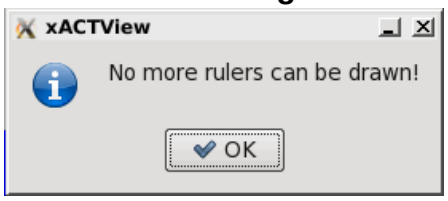
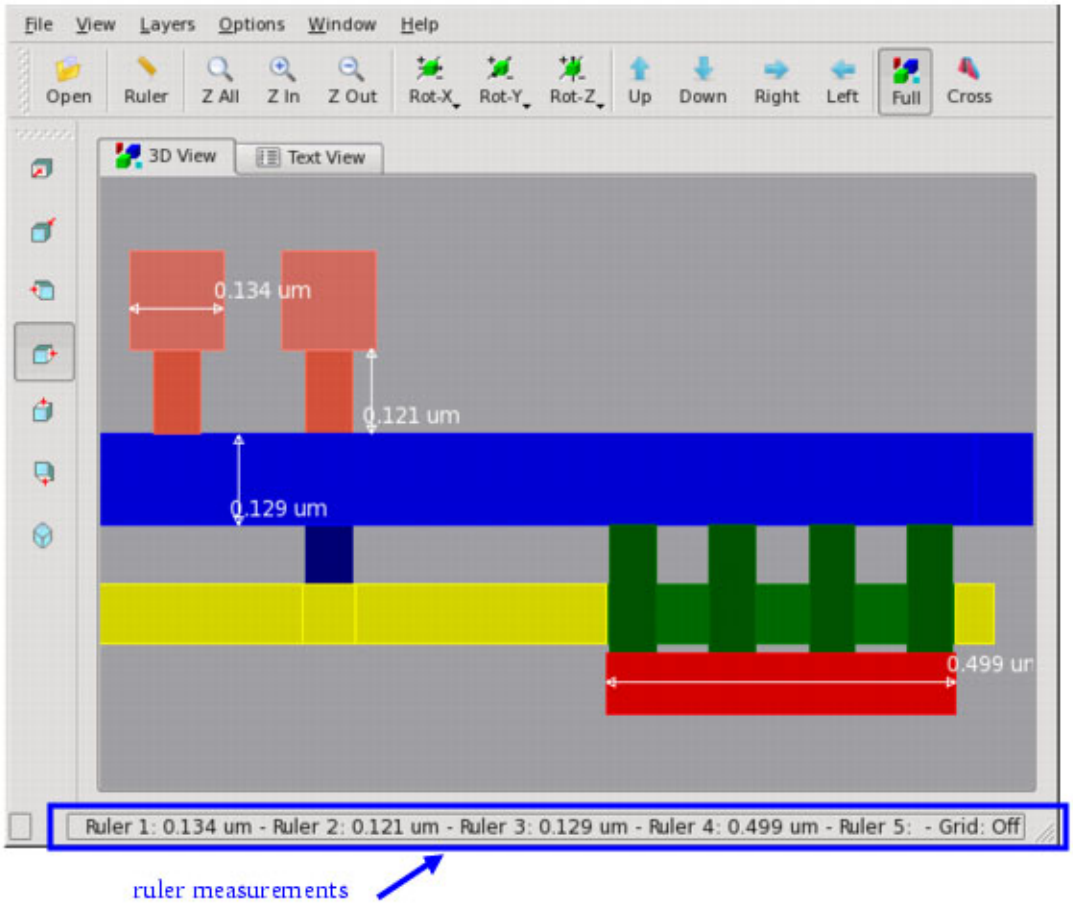


Figure 16-43 shows how multiple ruler measurements are displayed in both the 3D View display window and the status bar. In this example, Number of Rulers is set to 5, and 4 out of 5 rulers have been drawn in the display. Note that Ruler 5 in the status bar has no value yet.

Figure 16-43. Multiple Ruler Measurements



Select the **Ruler** toolbar button or change to 3D view to clear the drawn rulers from the display.

Chapter 17
Calibre xACT Tool Invocation Reference

Calibre xACT command line options allow you to execute the tool from the command line or with a run script.

This chapter contains the Calibre xACT tool invocation reference information.

Setting the CALIBRE_HOME Environment Variable 235
Command Invocation Reference 236
calibre -lvs. 237
calibre -xact. 238
calibre -xact -phdb 241
calibre -xact -3d -pdb 243
calibre -xact -fmt. 247
xactview 251

Setting the CALIBRE_HOME Environment Variable

Calibre tools require that the CALIBRE_HOME environment variable be set.

See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.

Command Invocation Reference

Reference pages describe the command line invocation syntax for the Calibre xACT and Calibre xACTView tools.

This section covers the following tool invocation commands:

- Source-Based connectivity and device recognition — calibre -lvs
- Full chip extraction for 16nm and 14nm and extraction of nets using Calibre xACT direct netlisting — calibre -xact
- Layout-Based connectivity and device recognition — calibre -xact -phdb
- Extraction and analysis of nets using the Calibre xACT 3D PDB extraction flow — calibre -xact -3d -pdb
- Format the netlist or report — calibre -xact -fmt
- View the FS3 file in a 3D graphical viewer — xactview

calibre -lvs

Runs the Calibre nmLVS-H tool and creates the persistent hierarchical database (PHDB) for use by the Calibre xACT tool.

Usage

calibre -lvs -hier -spice *directory_path/layout_primary.sp rule_file_name*

Description

Runs the Calibre nmLVS-H tool and creates the Persistent Hierarchical Database (*PHDB*) for use by the Calibre xACT tool.

When you invoke the Calibre nmLVS-H tool, you must specify an explicit path to the SVDB directory using the following syntax:

directory_path/layout_primary.sp

where:

- *directory_path* is the path you specified in the Mask SVDB Directory statement in the rule file
- *layout_primary.sp* is the design's top-level cell you specified with the Layout Primary statement in the rule file

Note

In standard LVS usage, you can specify any directory name for the SPICE netlist. To use the LVS output in the PDB stage, however, *directory_path* **must** match the Mask SVDB Directory setting.

Arguments

None. For complete Calibre nmLVS-H information, see the "Calibre nmLVS and Calibre nmLVS-H Command Line" section of the *Calibre Verification User's Manual*.

Examples

The SVRF rule file is design.rules and contains the following SVRF statements:

```
LAYOUT PRIMARY "ring"
MASK SVDB DIRECTORY "/scratch1/design/svdb" xACT
```

You invoke the Calibre nmLVS-H tool from the command line using the following syntax:

```
calibre -lvs -hier -spice /scratch1/design/svdb/ring.sp design.rules
```

calibre -xact

Based on the input design data specified in the rule file, Calibre xACT extracts parasitic data, performs parasitic analysis on the preprocessed data, and generates the output netlist.

Usage

calibre -xact [-3d | -3dselect] [-c | -r | -rc | -rcc | -rl | -rlm | -rcl | -rclm | -rccl | -rcclm | -simple]
[-g] [-turbo [*number_of_cpus*] [-turbo_all]]
[-xcell *xcell_list*]
[-corner {*corner*[,*corner*]… }] ***rule_file_name***

Arguments

- -3d

  An optional argument that invokes the 3D field solver for parasitic capacitance extraction and analysis.

- -3dselect

  An optional argument that invokes the 3D select extraction. Specify this option when you specify the PEX Fieldsolver Mode statement in your rule file to extract specific nets with the Calibre xACT 3D fieldsolver tool. Do not use with -3d, -full, or -pdb.

- [-c |-r | -rc | -rcc | -rl | -rlm | -rcl | -rclm | -rccl | -rcclm | -simple]

  An optional argument that specifies the parasitic model. Select one of the following options:

-c	Exclusively writes capacitance models into netlist.
-r	Exclusively writes resistance models into netlist.
-rc	Specifies distributed RC parasitic extraction. This mode calculates coupling capacitors but grounds them in the output netlist.
-rcc	Specifies R-coupled-C extraction. This mode creates netlists with a distributed RCC network including resistors, coupling capacitors, and intrinsic capacitors. This parameter is the default if a parasitic model option is not specified.
-rl	Specifies the extraction of a distributed network of resistors and inductors.
-rlm	Specifies the extraction of a distributed network of resistors and inductors along with mutual inductance.
-rcl	Specifies the extraction of a distributed network of resistors, capacitors, and inductors.
-rclm	Specifies the extraction of a distributed network of resistors, capacitors, and inductors along with mutual inductors.
-rccl	Specifies the extraction of a distributed network of resistors, intrinsic and coupled capacitors, and inductors. Coupled capacitors are not grounded.

-rcclm	Specifies the extraction of a distributed network of resistors, inductors, intrinsic and coupled capacitors along with mutual inductors. Coupled capacitors are not grounded.
-simple	Produces a simple netlist with no parasitic elements. The SVRF file must contain a PEX Netlist Simple statement. This parameter overrides the PEX XACT Control statement if specified.

- -g

  An optional argument that grounds any coupling capacitors.

- -turbo [*number_of_cpus*]

  An optional argument set that specifies to use multi-threaded parallel processing for extraction. The number_of_cpus argument is a positive integer specifying the number of processors to use in the processing. If you omit this number, the Calibre xACT tool runs on the maximum available for which you have licenses. If you do not apply the -turbo option, it defaults to running on two processors if available. To force the Calibre xACT tool to run on only one processor, specify "-turbo 1" on the command line.

  For more information on this option, refer to the *Calibre Administrator's Guide*.

- -turbo_all

  An optional argument used with the -turbo argument. This parameter halts the invocation if it cannot secure the exact number of CPUs specified using -turbo.

- -xcell *xcell_list*

  An optional argument set that specifies to extract hierarchically. Specifies the path to and the name of the file containing a list of cells to be preserved during extraction (xcells). When -xcell is specified, the primary cell is extracted to the boundaries of the xcells. For more information on xcells, see "Hierarchy Control with Xcells" on page 149.

- -corner {*corner*[,*corner*]… }

  An optional argument set that specifies one or more process corners to be extracted in a multi-corner run. If you specify two or more corner names, specify the names as a comma separated list with no spaces. If the rules define multiple corners and you do not use this parameter, only the typical corner is netlisted.

- ***rule_file_name***

  A required argument that specifies the path to and name of the SVRF rule file.

Examples

To create an output netlist with a distributed RCC network (distributed resistance, coupling capacitance, and intrinsic capacitance) for a layout name flow using an annotated GDS file (PEX Netlist LAYOUTNAMES), use the following command:

```
calibre -xact -rcc my_rules
```

For a source name flow (PEX Netlist SOURCENAMES), you must run Calibre LVS before running Calibre xACT. For example:

```
calibre -lvs -hier my_rules
calibre -xact -rcc my_rules
```

This example extracts resistance and distributed capacitance in a direct netlisting flow.

```
calibre -xact -3d -rc my_rules
```

calibre -xact -phdb

Performs connectivity extraction and device recognition, and creates the persistent hierarchical database (PHDB) for use by the Calibre xACT tool. This command also generates a layout netlist. Use only in the Calibre xACT 3D PDB flow.

Usage

calibre -xact -phdb [-noasic]
[-turbo [*number_of_cpus*] [-turbo_all] [-remote *host*[,*host*…] | -remotefile *filename*]]
[-hcell *hcell_list*] ***rule_file_name***

Description

Performs connectivity extraction and device recognition, and writes the results to a *PHDB*. This command also generates a layout netlist. After this step, you create the Parasitic Database (*PDB*) and format the netlist or report.

PHDBs must be re-generated if they are inconsistent with a current run; this can occur if you change the rule file.

Arguments

- **-xact**

  A required argument that invokes the Calibre xACT tool. For the PHDB stage, the netlists contain layout names. Use calibre -lvs for schematic names.

- **-phdb**

  A required argument that runs connectivity extraction and device recognition on the layout database specified in the rule file, and generates a PHDB.

  The resulting PHDB is named *layout_primary.phdb*, where layout_primary is the name specified by the Layout Primary specification statement in the rule file.

  This command also generates a layout netlist named layout_*primary*.sp which is input to the formatter.

  The Calibre xACT tool places both the PHDB and the layout netlist in the SVDB.

- -noasic

  An optional argument that specifies to disable the ASIC extraction mode when including GDS metal fill or layout blocks. By default, ASIC optimizations are on for LEF/DEF, and off for other formats. Specifying this option ensures that these optimizations are not used for fill extraction that involves GDS data.

- -turbo [*number_of_cpus*]

  An optional argument set that specifies to use multi-threaded parallel processing for PHDB creation. The *number_of_cpus* argument is a positive integer specifying the number of processors (CPUs) to use in the processing. If you omit this number, the Calibre xACT tool runs on the maximum available for which you have licenses. If you do not apply the -turbo

option, it defaults to running on two processors if available. To force the Calibre xACT tool to run on only one processor, specify "-turbo 1" on the command line.

For more information on this option, see the *Calibre Administrator's Guide*.

- -turbo_all

  An optional argument used with the -turbo argument to halt the invocation if the tool cannot secure the exact number of CPUs specified using -turbo.

- -remote *host*[, *host*…]

  An optional argument set used to run the software on remote hosts using the MTflex multi-threaded, parallel processing architecture. It must be specified in conjunction with the -turbo argument. It enables multi-threaded operation on remote hosts of a distributed network. You must specify at least one host name. A list of hosts is comma-delimited and specifies that multiple hosts participate in multi-threaded operations. You must have the required number of licenses for your job.

  For more details, see the *Calibre Administrator's Guide*.

- -remotefile *filename*

  This optional argument set is part of the MTflex multi-threaded, parallel processing architecture, which enables multi-threaded operation on remote hosts of a distributed network. It must be specified in conjunction with the -turbo argument, which specifies the number of processors you are using, including those on the remote hosts. The *filename* specifies the pathname of a configuration file containing information for the local and remote hosts. You must have the required number of licenses for your job.

  For more details, see the *Calibre Administrator's Guide*.

- -hcell *hcell_list*

  An optional argument set that specifies the path to and name of the hcell file. It is used to create hierarchical PHDBs. See "Hierarchy Control with Xcells" for an in-depth discussion of hcells and xcells.

- ***rule_file_name***

  A required argument that specifies the path to and name of the SVRF rule file.

Examples

The following command invokes PHDB creation using a Calibre xACT license:

```
calibre -xact -phdb -hcell hcells my_rules
```

The following command runs the same job on a computer named lsf3:

```
calibre -xact -phdb -turbo -remote lsf3 -hcell hcells my_rules
```

The following command runs the same job on two remote host computers named lsf2 and lsf3:

```
calibre -xact -phdb -turbo -remote lsf2,lsf3 -hcell hcells my_rules
```

calibre -xact -3d -pdb

Extracts parasitic data and performs parasitic analysis on the data within the PHDB. It also generates the Parasitic Databases (PDBs) containing the electrical data for each net.

Usage

calibre -xact {**-r**} | {**-3d -pdb**[-hybrid] {**-rc** | **-rcc** | **-l** | **-m** | **-rl** | **-rlm** | **-rcl** | **-rclm** | **-rccl** | **-rcclm**}}
[-turbo [*number_of_cpus*] [-turbo_all] [-remote *host*[,*host*…] | -remotefile *filename*]]
[-xcell *xcell_list* [-full]]
[-select] [-noasic]
[-nocheck] [-pdb_info]
[-E *output*] [-tvfarg *argument*] ***rule_file_name***

Arguments

- **-r**

  A required argument used to perform parasitic resistance-only extraction and analysis on data in the PHDB, and generate the PDB. Places the R-only models in the PDB. Capacitance is not extracted.

- **-3d**

  A required argument used to invoke the 3D field solver for parasitic capacitance extraction and analysis.

- **-pdb**

  A required argument used to perform parasitic extraction and analysis on data in the PHDB and generate a parasitic database (PDB) containing the electrical data for each net. Used only in the PDB extraction flow. For more information, see “Using the Calibre xACT 3D PDB Flow” on page 63.

- -hybrid

  An optional argument used to invoke Hybrid xACT 3D/Rule-Based extraction. This argument can be used together with -select. Specify this option to extract all nets selected for xRC and all nets selected for xACT, then combine the extraction results into a single PDB. Do not use with -full or in the direct netlisting flow. For more information, see “Hybrid xACT 3D/Rule-Based Extraction” on page 102.

- {**-rc** | **-rcc** | **-l** | **-m** | **-rl** | **-rlm** | **-rcl** | **-rclm** | **-rccl** |**-rcclm**} (choose one)

  A required argument that specifies the parasitic model. Select one of the following:

-rc	Specifies distributed RC parasitic extraction and places the distributed RC models in the PDB. This mode calculates coupling capacitors but grounds them in the output netlist.
-rcc	Specifies R-coupled-C extraction and places fully-coupled models in the PDB.

-l	Specifies self inductance extraction.
-m	Specifies mutual inductance extraction.
-rl	Specifies the extraction of a distributed network of resistors and inductors.
-rlm	Specifies the extraction of a distributed network of resistors and inductors along with mutual inductance.
-rcl	Specifies the extraction of a distributed network of resistors, capacitors, and inductors.
-rclm	Specifies the extraction of a distributed network of resistors, capacitors, and inductors along with mutual inductors.
-rccl	Specifies the extraction of a distributed network of resistors, intrinsic and coupled capacitors, and inductors. Coupled capacitors are not grounded.
-rcclm	Specifies the extraction of a distributed network of resistors, inductors, intrinsic and coupled capacitors along with mutual inductors. Coupled capacitors are not grounded.

- -turbo [*number_of_cpus*]

  An optional argument set that specifies to use multi-threaded parallel processing for PDB creation. The *number_of_cpus* argument is a positive integer specifying the number of processors to use in the processing. If you omit this number, the Calibre xACT tool runs on the maximum available for which you have licenses. If you do not apply the -turbo argument, it defaults to running on two processors if available. To force the Calibre xACT tool to run on only one processor, specify "-turbo 1" on the command line.

  For more information on this option, refer to *Calibre Administrator's Guide*.

- -turbo_all

  An optional argument used with the -turbo argument. This argument halts the invocation if it cannot secure the exact number of CPUs specified using -turbo.

- -remote *host*[, *host*…]

  An optional argument set used to run the software on remote hosts using the MTflex multi-threaded, parallel processing architecture. It must be specified in conjunction with the -turbo option. It enables multi-threaded operation on remote hosts of a distributed network. You must specify at least one host name. A list of hosts is comma-delimited and specifies that multiple hosts participate in multi-threaded operations. You must have the required number of licenses for your job.

  For more details, see the *Calibre Administrator's Guide*.

- -remotefile *filename*

  This optional argument set is part of the MTflex multi-threaded, parallel processing architecture, which enables multi-threaded operation on remote hosts of a distributed

network. It must be specified in conjunction with the -turbo option, which specifies the number of processors you are using, including those on the remote hosts. The *filename* specifies the pathname of a configuration file containing information for the local and remote hosts. You must have the required number of licenses for your job.

For more details, see the *Calibre Administrator's Guide*.

- -xcell *xcell_list*

  An optional argument set used to create the PDB hierarchically. The *xcell_list* specifies the path to and the name of the file containing a list of cells to be preserved during extraction (xcells). For more information on xcells, see "Hierarchy Control with Xcells" on page 149. Used only in the PDB extraction flow.

  When -xcell is specified without -full, the primary cell is extracted to the boundaries of the xcells.

- -full

  An optional argument used in conjunction with the -xcell option. Performs hierarchical extraction, with fully netlisted xcells. Used only in the PDB extraction flow. Do not use with -hybrid. For more information, see "Hierarchical Memory Extraction" on page 104.

- -select

  An optional argument that specifies to exclusively extract nets using the net names you specify with PEX Extract Include SVRF statement. For more information, see "Extracting Particular Nets with the Calibre xACT 3D PDB flow" on page 167. Use together with -hybrid to extract all nets selected for xRC and all nets selected for xACT, then combine the extraction results into a single PDB. Used only in the PDB extraction flow.

- -noasic

  By default, ASIC optimizations are off for all Calibre xACT 3D supported formats.

- -nocheck

  An optional argument that specifies to continue the extraction run with only a warning if file date stamps (commented checksums) are inconsistent with each other. If -nocheck is not specified and the date stamps are inconsistent, the extraction run stops.

- -pdb_info

  An optional argument that controls whether messages from the analyzer are printed to the transcript.

- -E *output*

  An optional argument set that specifies an output file name for SVRF code generated by the TVF processor. If ***rule_file_name*** contains no TVF statements, *output* is empty. TVF code is processed before the run is started.

- -tvfarg *argument*

  An optional argument set that specifies an *argument* that is passed to a compile-time TVF script. The *argument* cannot contain space characters. The *argument* is read by the

tvf::get_tvf_arg command in the TVF rule file. For more information about TVF see the *Standard Verification Rule Format (SVRF) Manual*.

- ***rule_file_name***

  A required argument that specifies the path to and name of the SVRF rule file.

Examples

This example shows a select net run using the PDB extraction flow.

```
calibre -xact -phdb -hcell hcells my_rules
calibre -xact -3d -pdb -rcc -select my_rules
calibre -xact -fmt -all my_rules
```

This example shows a hybrid run using the PDB extraction flow.

```
calibre -xact -phdb -hcell hcells my_rules
calibre -xact -3d -pdb -hybrid -rcc -select my_rules
calibre -xact -fmt -all my_rules
```

calibre -xact -fmt

Produces netlists and reports from the PDB contents. Use only in the Calibre xACT 3D PDB flow.

Usage

calibre -xact -fmt [-netmodel] {**-c** | **-r** | **-rc** | **-rcc** | | **-l** | **-m** | **-rl** | **-rlm** | **-rcl** | **-rclm** | **-rccl** | **-rcclm** | **-all** | **-simple**}
[-g]
[-xcell *xcell_list* [-full]]
[-corner {*corner*[,*corner*]… | all}]
[-fmt_warnings] [-fmt_info]
[-nocheck]
rule_file_name

Description

Produces netlists and reports from the PDB contents. Standard output formats are generated from:

- Parasitic models stored in PDB(s)
- Source-to-layout cross-reference files, if they exist
- SPICE layout netlist
- Rule file specification statements
- Specified environment variables

The supported output formats include HSPICE, DSPF, Spectre, Eldo, SPEF, CalibreView, and ASCII reports.

You specify the output formats and filename locations with the PEX Netlist, PEX Netlist Select, PEX Netlist Simple, and PEX Report statements in the rule file.

Arguments

- **-fmt**

  A required argument that produces netlists and reports from parasitic data stored in the PDB.

  This argument generates netlists and reports from parasitic data generated during selected and flat extraction. Selected nets and flattened global nets are processed with the same command line parameters because, in both cases, the parasitic model represents nets extracted in their entirety and flattened to the top-level cell. Netlists and reports use layout netlist names unless source names are specified in the rule file. Parasitic models are named cell_name%net_name in the output netlists.

- -netmodel

  An optional argument that specifies to perform netlist formatting using a file containing selected nets with assigned net models. The SVRF rule file must contain a PEX Netlist Select File statement. This option cannot be used with **-c**, **-r**, **-rc**, **-rcc**, or **-all** options. It can only be used with the **-simple** option.

- {**-c** | **-r** | **-rc** | **-rcc** | **-l** | **-m** | **-rl** | **-rlm** | **-rcl** | **-rclm** | **-rccl** | **-rcclm** | **-all** | **-simple**}

  A required argument that specifies the output mode. Select one of the following:

  -c Exclusively writes capacitance models into netlist. Do not use when -netmodel is specified.

  -r Exclusively writes resistance models into netlist. Do not use when -netmodel is specified.

  -rc Distributed netlist; suppresses writing of any inductors; if coupled capacitance data exists, it is grounded; exits if no parasitic resistance data is in the PDB. Do not use when -netmodel is specified.

  -rcc Distributed netlist; suppresses writing of any inductors; exits if no parasitic resistance or coupled capacitance data is in the PDB. Do not use when -netmodel is specified.

  -l Distributed netlist; suppresses writing of any parasitic capacitors, resistors, or mutual inductance; exits if the PDB does not contain self inductance data.

  -m Distributed netlist; suppresses writing of any parasitic capacitors, or resistors, or self inductance; exits if the PDB does not contain mutual inductance data.

  -rl Distributed netlist; suppresses writing of any parasitic capacitors or mutual inductance; exits if the PDB does not contain self inductance or resistance data.

  -rlm Distributed netlist; suppresses writing of any parasitic capacitors; exits if the PDB does not contain resistance, self inductance, or mutual inductance data.

  -rcl Distributed netlist; suppresses writing of any mutual inductance; exits if the PDB does not contain capacitance, resistance, or self inductance data.

  -rclm Distributed netlist; grounds coupled capacitance; exits if the PDB does not contain resistance, capacitance, self inductance, or mutual inductance data.

  -rccl Distributed netlist; exits if the PDB does not contain resistance, coupled capacitance, self inductance, or mutual inductance data.

  -rcclm Distributed netlist; exits if the PDB does not contain resistance, coupled capacitance, self inductance, or mutual inductance data.

-all	Distributed netlist; writes the contents of the PDB into the netlist specified by the PEX Netlist statement. This switch is optional and is the default. Do not use when -netmodel is specified.
-simple	Produces a simple netlist with no parasitic elements. The SVRF file must contain a PEX Netlist Simple statement.

- -g

  An optional argument that grounds any coupling capacitors.

- -xcell *xcell_list*

  An argument set that is required if the PDB contains only cells extracted with -incontext. Formerly, required with all hierarchical PDBs to specify the path to and name of the xcell file. Retained for backwards compatibility in scripts.

- -full

  An optional argument which specifies that the formatter should produce a fully hierarchical netlist.

- -corner {*corner*[,*corner*]… | all}

  An optional argument set that selects the process corners to write out. If you specify two or more corner names, do not place a space between names. If the rules define multiple corners and you do not use this option, only the typical corner is netlisted.

  When -corner is used in the formatter stage, sensitivity netlisting is disabled and sensitivity variations do not appear in the netlist.

- -fmt_warnings

  An optional argument used to display warning messages while the Calibre xACT formatter runs. Otherwise, messages are not displayed.

- -fmt_info

  An optional argument used to display informational messages while the Calibre xACT formatter runs. Otherwise, messages are not displayed.

- -nocheck

  An optional argument that specifies to continue the extraction run with only a warning if file date stamps (commented checksums) are inconsistent with each other. If -nocheck is not specified and the date stamps are inconsistent, the extraction run stops.

- ***rule_file_name***

  A required argument that specifies the path to and name of the SVRF rule file.

Examples

To write resistance with distributed capacitance and coupled capacitance between nets to the output netlist, use the following invocation:

```
calibre -xact -fmt -rcc my_rules
```

When using a net file to define how specific nets should be formatted, use the following invocation:

```
calibre -xact -fmt -netmodel my_rules
```

The *my_rules* file specifies the net file *my_selected_nets* as follows:

```
PEX NETLIST SELECT FILE my_selected_nets
```

For more information on PEX Netlist Select File see the *Standard Verification Rule Format (SVRF) Manual.*

xactview

Invokes the Calibre xACTView 3D layout viewer.

Note

Using the Exceed on Demand (EoD) client installed on the Windows platform to run Calibre xACTView can cause the session to fail or the display panel to appear blank. Use an EoD client installed on a Linux RH6 platform to run Calibre xACTView instead.

Usage

xactview [-fs3 *fs3_file_name* [-pdb *pdb_name*] [-fs3out *fs3out_name* [-cfcap *cfcap_name*]] [-layers *layers*] [-window *x1 y1 x2 y2*]] | [-version | -help]

Description

Invokes the Calibre xACTView 3D layout viewer. The parameters are optional. You can invoke the viewer then load an FS3 file from the Calibre xACTView GUI. Use the viewer to graphically analyze the different polygons processed for capacitance extraction by Calibre xACT 3D.

Arguments

- -fs3 *fs3_file_name*

  An optional argument set used to open the FS3 file in Calibre xACTView tool.

- -pdb *pdb_name*

  An optional argument set used with the -fs3 option that loads the parasitic database associated with the FS3 file. Use this option to view net names instead of net ids for polygons in the Calibre xACTView tool.

- -fs3out *fs3out_name*

  An optional argument set used with the -fs3 option that loads the coupling capacitance information associated with the FS3 file. Use this option to view coupling capacitance information in the Calibre xACTView tool.

- -cfcap *cfcap_name*

  An optional argument set used with the -fs3out option that loads the optional correction file associated with the FS3.out file. Use this option to view coupling capacitance correction information for polygons in the Calibre xACTView tool.

- -layers *layers*

  An optional argument set used with the -fs3 option to specify which layers are visible in the FS3 display window upon invocation of the Calibre xACTView tool. The *layers* argument is a list of one or more layer names found in the FS3 file separated by spaces.

- -window *x1 y1 x2 y2*

  An optional argument set used with the -fs3 option to specify the top view display of a small portion of a large FS3 file in the FS3 display window upon invocation of the Calibre xACTView tool. This option is not available from within the Calibre xACTView GUI.

- -version

  An optional argument used to output the version of the Calibre xACTView tool.

- -help

  An optional argument used to output the usage for the Calibre xACTView tool.

Examples

This example invokes the Calibre xACTView tool.

```
xactview
```

This example invokes the Calibre xACTView tool and loads the FS3 file, displaying a graphical view of the 3D structure.

```
xactview -fs3 my.fs3
```

This example invokes the Calibre xACTView tool, loads the FS3 file, and loads the PDB associated with the design. Loading the PDB allows you to view net names instead of net ids for polygons in the Calibre xACTView tool.

```
xactview -fs3 my.fs3 -pdb my.pdb
```

This example invokes the Calibre xACTView tool and loads the FS3 file displaying only the layers ipoly, gate, metal1, and metal2. Upon invocation, the display option for all other layers is set to off.

```
xactview -fs3 my.fs3 -layers ipoly gate metal1 metal2
```

Appendix A
Reduction Techniques

Various extraction reduction techniques are available to streamline the extraction process.

Capacitive and Resistive Reduction 253
Threshold-Based Reduction 254
TICER 255

Capacitive and Resistive Reduction

Use capacitive and resistive reduction to control netlist size and simulation performance.

There are SVRF statements that reduce the number of capacitors and SVRF statements that reduce the number of resistors:

- **PEX Reduce CC** — reduces the number of coupled capacitors by converting coupled capacitors that meet some constraint to lumped capacitance. The lumped capacitance on a net is represented as a single value coupled to ground, thus reducing the overall netlist size.
- **PEX Reduce Mincap** — reduces both intrinsic and coupled capacitors based on a user-defined threshold value. The command can specify to remove or combine capacitors.
- **PEX Reduce Minres** — reduces parasitic resistors by combining them based on a user-defined threshold value. The command combines parasitic resistors.
- **PEX Reduce Parallel** — reduces two or more resistors connected in parallel between the same pair of nodes into a single equivalent resistor in the extracted netlist.

Note

The PEX Reduce CC reduction overall provides the best control. It is the only one which bounds the error that can be introduced by aggressive reduction.

PEX Reduce CC

The PEX Reduce CC specification is applied during netlisting. The total coupling capacitance between two nets is compared to a constraint, either an absolute value such as 3 femtofarads or a percentage of the total net capacitance for either net. If the coupling capacitance is less than the constraint, it is decoupled from the nets and included in the lumped capacitance to ground. Note the percentage constraint must hold for both nets.

PEX Reduce CC runs before PEX Reduce TICER. For more information, see PEX Reduce CC in the *Standard Verification Rule Format (SVRF) Manual*.

PEX Reduce Mincap

The PEX Reduce Mincap specification is applied after the PDB generation, during the formatting stage. PEX Reduce Mincap implements two types of reduction, merging and removing, based on a user defined threshold.

By setting the REMOVE threshold, any capacitors that fall below the threshold value are either grounded or removed.

By setting the COMBINE threshold, any capacitors that fall below the threshold value are combined with neighboring capacitors on the same net.

For more information see PEX Reduce Mincap in the *Standard Verification Rule Format (SVRF) Manual*.

PEX Reduce Minres

The PEX Reduce Minres specification is applied after the PDB generation, during the formatting stage. PEX Reduce Minres COMBINE merges resistors based on a user-defined threshold.

For more information see PEX Reduce Minres in the *Standard Verification Rule Format (SVRF) Manual*.

PEX Reduce Parallel

The PEX Reduce Parallel specification controls whether or not resistors in parallel are combined in the netlist during the formatting stage.

For more information see PEX Reduce Parallel in the *Standard Verification Rule Format (SVRF) Manual*.

Threshold-Based Reduction

The PEX Reduce Digital statement decreases the size of netlists and databases.

When you use the PEX Reduce Digital statement in your rule file, the Calibre xACT tool uses the threshold you define for distributed RC parasitic extraction; if a distributed RC model meets the threshold, then the Calibre xACT formatter converts the model into a lumped C model by discarding the resistors.

TICER

TICER stands for “Time Constant Equilibration Reduction”.

You specify this reduction method in your SVRF rule file using the following SVRF statement and keyword:

```
PEX REDUCE TICER frequency
```

where *frequency* is a user-defined calculated number controlling which nodes in the circuit the tool can select for subsequent elimination; specifically, the tool selects nodes with time constants less than the *frequency* parameter.

In your rule file, specify the calculated frequency parameter in hertz and express the value in the PEX Reduce TICER statement using scientific notation. For example:

```
PEX REDUCE TICER 40e9
```

Using the Calibre Interactive PEX interface, specify TICER reduction by selecting “Enable TICER reduction below” and entering a frequency. The “Enable TICER reduction below” option is in the PEX Options pane. To enable PEX Options, select **Setup > PEX Options** in the Calibre Interactive - PEX menu.

How TICER Works

The TICER reduction method preserves the frequency response of an RC interconnect circuit from DC up to the frequency parameter. Consequently, the frequency parameter provides you with a control for trading off compression with accuracy.

- Setting the frequency parameter to a higher value results in larger (more R and C elements) interconnect circuits having accuracy over a wider bandwidth.
- Setting the frequency parameter to a lower value results in more compression and an earlier roll-off in accuracy.

Calculating the frequency Parameter

You must calculate the frequency parameter using the following formula:

$$frequency = \frac{tradeoff\_value}{transition\_time\_minimum}$$

where:

tradeoff_value — a number between 4 and 10. A larger tradeoff_value results in a higher frequency parameter value and, consequently, more accurate reduction results.

transition_time_minimum — the shortest rise or fall time you expect in your design. In general, you can estimate this value using 1/5 of your design's switching delay.

TICER and Temperature Sensitivity Effects

If you include temperature sensitivity in the extraction process by using the PEX Extract Threshold statement, the reduced netlist will include parasitic capacitors that have small changes in value due to temperature coefficients.

These changes are caused by how RC delays are calculated when using TICER. Including temperature coefficients in the extraction process affects the final values of parasitic resistors. During TICER reduction the values for resistors and capacitors are recalculated so that the RC delayer in the network remains unchanged. Including temperature coefficients for resistors affects the recalculated values for the parasitic capacitors in the final netlist.

Appendix B
Error and Warning Messages

Error and warning messages are generated by the Calibre xACT tools at run time.

Error Messages 257
Warning Messages 258

Error Messages

Error messages must be corrected to continue a Calibre xACT run.

Table B-1. Calibre xACT Error Messages

Message	Possible Causes
Source could not be read.	The specified source file is corrupt or incomplete.
Can not open source netlist file *filename* for input.	The source file does not exist or the file name is spelled incorrectly.
Failure to open input file *filename* for read access.	Incorrect layout file name specified. Check that the name is spelled correctly.
A LAYOUT PRIMARY is not specified in the rule file.	Layout Primary statement is not specified in the rule file.
A LAYOUT PATH is not specified in the rule file.	Layout Path statement is not specified in the rule file.
Cannot determine the LAYOUT SYSTEM from rule file *filename.rule*	Layout System statement is not specified in the rule file.
MASK SVDB DIRECTORY was not specified in the rules file.	Mask SVDB Directory statement is not specified in the rule file.
Error OPEN1 - problem with access, file type, or file open of file: *filename*. Compilation of the SVRF rules failed.	One or more of the calibrated rule files (*rules.C* or *rules.R*) or the LVS rule file has not been found.
SVRF syntax error on encrypted line *linenumber* of *filename*. Compilation of the SVRF rules failed.	One or more of the calibrated rule files (*rules.C* or *rules.R*) or the LVS rule file has not been found.
Compilation of the SVRF rules failed.	One or more of the calibrated rule files (*rules.C* or *rules.R*) or the LVS rule file has not been found.

Table B-1. Calibre xACT Error Messages (cont.)

Message	Possible Causes
No output files specified for formatter run (PEX NETLIST/REPORT).	A form of PEX NETLIST or PEX REPORT statement is not specified in the rule file.
Error INP1 on line ## of SVRF generated from TVF - superfluous or invalid input object: *svrf_statement_name*.	The SVRF statement syntax is incomplete or incorrect.
Error INP7 on line ## of *rulefilename* - incomplete keyword specification: *svrf_statement_name*.	The SVRF statement syntax is incomplete or incorrect.
Error SPC1 on line ## of *rulefilename* - superfluous specification statement: *svrf_statement_name*.	The SVRF statement can only be specified once. Remove any duplicate statement specifications.
No default profile in SVRF rules.	A calibrated rule file is not valid or does not exist.
Error PEX40 - Incompatible statements: PEX MAP REQUIRED: missing mapping or mapping to undefined layer for LVS layer *layer_name*.	The *layer_name* is not a calibrated layer and has not been specified in a PEX Map statement.
FATAL ERROR: Layer stack control statement is required for xACT Transistor to proceed. Contact support. Exiting...	An incomplete or wrong foundry rule file specified.
FATAL ERROR: No output files specified for formatter run (PEX NETLIST/REPORT).	The LUMPED keyword was specified in the PEX Netlist statement specification.
FATAL ERROR: Gate Layer "*gate_layer_name*" is not mapped to a calibrated layer.	Use PEX Map statement to map the gate layer to a calibrated layer.

Warning Messages

Warning messages should be reviewed to determine if they indicate a real problem in your design.

Table B-2. Calibre xACT Warning Messages

Message	Possible Causes
PEX FIELDSOLVER MODE is not supported in Calibre xACT.	The PEX Fieldsolver Mode statement cannot be used in the Calibre xACT transistor flow.

Table B-2. Calibre xACT Warning Messages (cont.)

Message	Possible Causes
No match found for XCELL name "*name**" in file: *rule_file_name*.	An XCELL name was specified in the rule file but does not exist in the design.
Cannot exclude net '*netname*'. Name not found in layout.	The *netname* was specified in a PEX Extract Exclude statement. Make sure the *netname* is correct.
Layer "*layername*" is not mapped to a calibrated layer.	The *layername* is not mapped to a calibrated layer. See PEX Map statement.
CAPACITANCE ORDER statement is incomplete.	The Capacitance Order statement is missing a mapped layer name.
Rules precision (1000) does not match with layout precision (2000). Layout precision (2000) will be used.	The rules precision is set by the Precision statement. The layout (database) precision is read from the layout file such as GDS/LEF/DEF files. Check to see how the Layout Use Database Precision statement is specified. For this warning the Layout Use Database Precision is set to YES. If Layout Use Database Precision is set to NO, then rules precision will be used.
Duplicate net '*netname*' in (SPECIAL)	Check the DEF file. The DEF file has a "special" net section with a list of nets containing special pins. This message is generated when a duplicate *netname* is detected in this special net section.

Glossary

backannotation
A process where extracted parasitics are added to the source netlist for parasitic re-simulation.

black box extraction
An extraction method where the contents of cells are ignored during parasitic extraction.

coupled capacitance
The capacitance between two conductors.

disjoint parasitic
A parasitic element associated with an incompletely routed net.

distributed capacitance
Parasitic capacitance modeled with separate elements distributed over a net that is divided into segments.

FS3 file
A generated file containing the interconnect process technology data (layer descriptions), the layout polygon 3D data (geometry and shape descriptions), and the extraction commands for the field solver.

floating net
A net that is not electrically grounded through connection to a device or xcell port.

floating net coupling
Coupling capacitance between signal nets and floating nets.

formatter
Extraction step which formats the netlist according to command line options and SVRF statements found in the extraction rule file.

fringe capacitance
The capacitance between the side of a conductor and either the substrate (intrinsic) or the bottom or top of another conductor (crossover).

full hierarchical extraction
A type of hierarchical extraction in which nets are extracted down to user-defined cells, and the contents of the cells are also extracted, preserving hierarchy. See “Hierarchical Memory Extraction”.

gate-level extraction
A type of hierarchical extraction in which nets are extracted down to user-defined cells, but no further. The PDB and PHDB contain no information about cell contents. See "Calibre xACT 3D Gate-Level Extraction".

gray box extraction
An extraction method where the parasitic elements between top-level routing cell geometries are included in the extraction. These parasitics are added to the intrinsic capacitance of a top-level net crossing over a cell.

hcell
A user-specified hierarchical cell used by Calibre nmLVS.

hierarchical extraction
A type of extraction which extracts data for each user-defined cell as well as the top level of the design. See "Hierarchical Memory Extraction".

mixed-signal hierarchical extraction
A type of hierarchical extraction which extracts some user-defined cells in full, and stops at the boundary of others. Contrast full hierarchical extraction and gate-level extraction.

in-context cells
Cells that are specified in an xcell file for in-context extraction.

in-die variation
During parasitic extraction, the drawn dimensions of conductors along with the local density of the material in a region around the conductor are used to determine the actual width, spacing, and thickness of each line.

intrinsic capacitance
The capacitance between a net and substrate (ground).

lumped capacitance
The amount of parasitic capacitance for a net. The lumped capacitance is represented as a single parasitic capacitor between net and ground and includes all intrinsic and coupled capacitance effects.

map file
A file which maps layout layer names to layer names used in a SVRF rule file.

mutual inductance
Defined as the ratio of electromotive force (emf) generated between two inductors, or the full emf effect of one current loop over another.

nearbody capacitance
The capacitance between the sides of two conductors, either on the same layer or different layers.

net exclusions
Nets in the design for which no parasitic model is extracted.

parasitic models
A set of multi-variable polynomial equations that compute parasitic effects.

parasitic netlist
A netlist containing models of the parasitic effects. The exact format and types of parasitics are specified by SVRF statements and command-line options.

PHDB
The Persistent Hierarchical Database, a database that stores information about your layout.

PDB
The Parasitics Database created by the extraction step. This database contains information about the parasitic capacitance and resistance.

plate capacitance
The capacitance between the lower surface of a conductor and the substrate, or the lower surface of a conductor and the upper surface of another conductor.

primitive cell
A cell that a designer provides from a standard library (for example, nand, xor, or). In hierarchical extraction, a primitive cell is designated with a -P in the xcell file and does not have parasitics extracted.

probe points
User-specified points on a net that are labeled and used to verify timing.

process corners
The variations on a “typical” process: for instance, metal thickness may not be exactly controlled.

process variation
Deterministic or random variability resulting from manufacturing process steps responsible for creating devices and interconnect in an integrated circuit.

signal net
A net that either has connections to devices or xcell ports, or is designated a port.

replicated device
Devices or cells that are repeated and connected together in a series or parallel combination. Replicated devices may correspond to one device on the source side (netlist or schematic).

self inductance
The change in a magnetic field of a conductor due to a change in current flow.

sensitivity aware
A type of extraction where electronic or physical sensitivities are taken into account during the extraction process. See also “process variation”.

smashed devices

Devices or cells that consist of drawn layout polygons at the same hierarchical level, also called "flattened".

source-based extraction

A type of extraction where the extracted parasitics from the layout are included in the source netlist, where layout devices are matched to source devices. See "Backannotating Parasitics to a Source Netlist" and "Using the Source Based Flow".

SVDB

The Standard Verification Database. The term is also used to indicate the directory named in the MASK SVDB DIRECTORY statement. The SVDB directory also contains the PHDB and PDB.

SVRF rule file

An ASCII file containing Calibre-specific statements. These statements are described in the *Standard Verification Rule Format (SVRF) Manual*.

TICER

TIme **C**onstant **E**quilibration **R**eduction method, used to reduce parasitic networks. See "TICER".

transistor-level extraction

A type of extraction where the design's interconnect nets are flattened into a top-level cell. For more information, see "Calibre xACT Flat Transistor-Level Extraction".

xcell

A user-specified extraction cell. The xcell appears in the generated netlists as a circuit. Every xcell must also be an hcell.

xcell file

An ASCII file that maps xcells to cells defined in the layout. For certain types of extraction, the xcell file settings may also affect whether parasitics are extracted.

Index

— Symbols —

[], 24
{}, 24
|, 24

— Numerics —

3-step invocation
  extraction with xACT 3D, 63, 246, 247

— A —

Accuracy trade-offs, 64

— B —

Backannotation, 138
Block-by-block hierarchical extraction, 94
Bold words, 23
Boundary conditions, 102

— C —

Calibre Interactive
  backannotating, 140
  coupled capacitance report, 143
  creating batch script, 187
  files used, 185
  gate-level extraction, 119
  hierarchical extraction, 127
  netlist without parasitics, 137
  point-to-point resistance report, 146
  preferences file, 185
  running, 81
  runset, 185
  runset loading, 72
  transistor-level extraction, 124
Calibre nmLVS-H
  hcell list, 150
  invoking, 237
Calibre xACT
  digital design flow, 17, 20, 22
  extracting with layout names, 28
  extracting with source names, 30
  full custom design flow, 18
  invoking the formatter, 247
  invoking xACT 3D, 243
  licensing, 20, 21, 22
  requirements, 20
  running, 21
  running digital design extraction, 51
  running in batch mode, 26
  setting up, 25
  supported SVRF statements for the digital flow, 53
  tool summary, 17
Calibre xACT 3D
  accuracy, 169
  creating netlist and reports, 67
  creating PDB, 64
  creating PHDB, 63
  direct netlisting, 61
  licensing, 21
  PDB flow, 63, 64
  requirements, 21
  running, 22
  running in batch mode, 60
  setting up, 60
  xcells, 150
Calibre xACTView
  axes window, 217
  display window, 206
  highlighting, 208, 209
  ignored capacitance highlighting, 213
  invoking, 201, 251
  layer palette, 215
  licensing, 23
  menus, 218
  requirements, 23
  ruler, 232
  running, 23
  session window, 205
  summary, 202
  toolbars, 230

CALIBRE_HOME variable, 235
Checksums, bypassing, 245
CMP data, 158
CMP modeling, 197
Command line switches
-fmt_info, 249
-fmt_warnings, 249
-noasic, 241
-simple, 239, 249
-turbo, 188
Command reference, 23
Command syntax, 23
Comparison of extraction types, 89
Connectivity extraction
with Calibre nmLVS, 237
with Calibre xACT, 241
Coupled capacitance, 112
Coupled capacitors, grounding, 169
Coupling capacitance, controlling, 169
Courier font, 24

— D —

DEF metal fill extraction, 97
Design hierarchy. See Hierarchy
Design, resizing, 189
Device extraction without parasitics, 163
Devices, removing from netlist, 179
Distributed resistance, 110
Distributed resistance and capacitance, 111
Distributed resistance and coupled capacitance, 112
Double pipes, 24

— E —

Error messages, 257
Errors, 257
Excluding nets, 168
Extraction
3d select extraction, 93
backannotating parasitics, 138
block-by-block, 94
comparison of types, 89
DEF metal fill, 97
devices without parasitics, 163
flat LEF/DEF, 93
flat transistor-level, 91, 108
full hierarchical and mixed-signal, running, 126
full hierarchical LEF/DEF, 95
gate-level, 107
gate-level, running, 117
GDS and OASIS metal fill, 95
GDS macrocells, 97
GDS metal fill, 95
hierarchical memory extraction, 104
hybrid, 102
mixed-signal hierarchical, 107
multi-corner multi-temperature, 98
multiple netlists, 92
multi-temperature, 97
net resistance by layer, 162
requirements, 116
selective resistance, 91
simple mode, 136
temperature, 97
top-level only, 148
transistor-level, running, 121
TSV, 98
with boundary conditions, 102
without parasitics, 136

— F —

Feedthrough nets, hierarchical extraction, 153
Flat LEF/DEF extraction, 93
Flat transistor-level extraction, 91, 108
Floating nets, ignore or extract, 169
fmt_info switch, 249
fmt_warnings switch, 249
Font conventions, 23
Formatter
-all switch, 248
-c switch, 248
-r switch, 248
-simple switch, 239, 249
Full hierarchical LEF/DEF extraction, 95

— G —

Gate-level extraction
defined, 107
improving runtime, 189
GDS cell views, 153
GDS metal fill extraction, 95

Global nets, excluded with Calibre xACT, 52
Global nets, excluding, 189
Gray box, 163
Ground
  excluding nets, 168
  excluding nets to improve runtime, 189
  multiple regions, 157
  nets excluded for LEF/DEF designs, 94, 108
  nets excluded with Calibre xACT, 52
  nets, excluding with Calibre Interactive, 79
Grounding coupled capacitors, 169
Grounding, -g command line switch, 239, 249

— H —

Hcells
  compared to xcells, 150
  in Calibre nmLVS-H, 150
Heavy font, 23
Hierarchical memory extraction, 104
Hierarchy
  overview, 107
  xcell list, 150
Highlighting Ignored Capacitance, 213
Hybrid xACT 3D/Rule-Based extraction, 102

— I —

Importing GDS cell views, 153
In-die variation, 196
interposer parasitics, 98
Invocation
  calibre -lvs, 237
  calibre -xact, 238
  calibre -xact -phdb, 241
  Calibre xACTView, 251
  CALIBRE_HOME requirement, 235
  syntax conventions, 23
  troubleshooting, 191
Isolated devices, 179
Italic font, 24

— L —

Layer map, for GDS metal fill, 96
Licensing information, 20, 22
Logic gates, gate-level extraction, 107
lvs.rep, 64
lvs.rep.ext file, 64

— M —

messages, 257, 258
Metal fill
  floating nets, 155
  modeling, 155
Minimum keyword, 24
Mixed-signal hierarchical extraction, 107
Multi-corner multi-temperature extraction, 98
Multiple corner extraction, 98
Multiple ground regions, 157
Multiple Netlist generation, 92

— N —

Net names, as port names, 184
Net Resistance Extraction by Layer, 162
Netlisting
  multiple grounds, 157
  process corners, 173, 175
  reduction techniques, 189
  without parasitics, 136
Netlists
  corners, 178
  multi-temperature, 97
Nets
  excluding to improve runtime, 189
  excluding with Calibre Interactive, 79
  joining disjoint fragments, 183
  pruning devices, 179
noasic switch, 241

— O —

On-chip variation
  CMP data, 158
  CMP modeling, 197
  description, 193
  in-die variation, 196
  modeling, 194
  sources, 193
Output files
  lvs.rep, 64
  lvs.rep.ext, 64

— P —

Parasitic devices, 109
Parasitic extraction. See Extraction

Parasitic models, 110
Parasitic models, disjoint, 183
Parasitics to output to RC netlist, 79
Parentheses, 24
PDB
  creating with source names, 63
  creating with xACT 3D, 64, 243
  creation in selected-net extraction, 167
  from empty xcell list, 150
  from flat transistor-level extraction, 91, 108
  from hybrid extraction, 102
  generate incrementally with Calibre Interactive, 81
  selecting information with Calibre Interactive, 78
  size considerations, 90
  turbo option, 188
PERC, and resistance extraction, 171
Performance trade-offs, 64
Persistent Hierarchical Database. See PHDB
PEX Ground Layer, 158
PEX Netlist statements, with formatter invocation, 247
PEX Probe File, 184
PEX Reduce CC, 253
PEX Reduce Digital, 254
PEX Reduce Mincap, 254
PEX Reduce Minres, 254
PEX Reduce Parallel, 254
PEX Reduce TICER, 255
PEX Report statements, 247
PHDB
  creati on for transistor-level extraction, 121
  creating for full hierarchical extraction, 127
  creating for gate-level extraction, 117, 118
  creating for ideal netlist, 136
  creating with layout names, 64, 241
  creating with source names, 237
  creation for transistor-level extraction, 122, 123
  files created with, 64
  for gate-level extraction, 107
  size considerations, 90
  using to create PDB, 64
Pin swapping, correcting, 180
Pipes, 24
Point-to-point resistance report, 145
Port names, as net names, 184
Power
  excluding nets to improve runtime, 189
  nets excluded for LEF/DEF designs, 94, 108
  nets excluded with Calibre xACT, 52
  nets, excluding, 168
  nets, excluding with Calibre Interactive, 79
praesagus. See CMP
Primitive cells, defined, 263
Probe points, setting, 184
Process variation, CMP, 158
Processing Control for Calibre xACT, 161

— Q —

Quotation marks, 24

— R —

Reports
  capacitance summary, 141
  coupling capacitance, 142
  point-to-point resistance, 145
Resistance extraction and PERC, 171
Resize design, 189
Runtime improvement, 188

— S —

Scale design, 189
Selected-net extraction, 165, 166, 167
Selective Resistance extraction, 91
Set probe points, 184
Setting CALIBRE_HOME, 235
Simple output mode, 239, 249
Slanted words, 24
Source name extraction, 237
Square parentheses, 24
subcircuit insertion, 98, 131, 134
Substrate noise analysis, 157
SVRF rule file
  echoing, 191
  excluding nets from extraction, 168
  for block-by-block extraction, 94
  for full hierarchical LEF/DEF extraction, 95

for multi-corner extraction, 98
for netlisting multiple corners, 173, 175
import GDS cells, 154
in DEF metal fill extraction, 97
metal fill and floating nets, 155
minimum requirements, 116
multiple ground regions, 158
reduction statements, 253
selected-net extraction, 165, 166, 167
temperature extraction, 97
TICER reduction, 255
with CMP data, 159
SVRF statements
PEX Reduce CC, 253
PEX Reduce Digital, 254
PEX Reduce Mincap, 254
PEX Reduce Minres, 254
PEX Reduce Parallel, 254
reduction statements, 254
supported statements in Calibre xACT digtal flow, 53
Swapped pins, correcting, 180
Syntax conventions, 23

— T —

Temperature
effects in TICER, 256
netlisting multiple, 173, 175
Temperature extraction, 97
Thickness, CMP based, 158
TICER, 255
Timing verification, 184
Top-level control file, 27
Trade-offs, 64
Troubleshooting, 191
TSV Extraction, 98
TSV extraction
analog flow, 130
digital flow, 132
TSV models, 98
turbo switch, 188

— U —

Underlined words, 24
Usage syntax, 23
Use Port Names for Ne t Names, 184

— V —

VCMP. See CMP
Verify timing, 184

— W —

Warnings, 258
Wildcards
in selected-net extraction, 165, 167
in xcell list, 152
xcell lists, 165, 166, 167

— X —

Xcells
and design hierarchy, 150
compared to hcells, 150
feedthrough nets as, 153
hierarchy control, 149
in Calibre xACT 3D, 150
tips for choosing, 153
wildcards in, 152
xcell list, 149
xcell list format, 150

Third-Party Information

Details on open source and third-party software that may be included with this product are available in the *<your_software_installation_location>/legal* directory.