SIEMENS EDA

# Calibre® Interactive™ User's Manual

Software Version 2021.2
Document Revision 9

SIEMENS

Unpublished work. © 2021 Siemens

This material contains trade secrets or otherwise confidential information owned by Siemens Industry Software, Inc., its subsidiaries or its affiliates (collectively, "Siemens"), or its licensors. Access to and use of this information is strictly limited as set forth in Customer's applicable agreement with Siemens. This material may not be copied, distributed, or otherwise disclosed outside of Customer's facilities without the express written permission of Siemens, and may not be used in any way not expressly authorized by Siemens.

This document is for information and instruction purposes. Siemens reserves the right to make changes in specifications and other information contained in this publication without prior notice, and the reader should, in all cases, consult Siemens to determine whether any changes have been made. Siemens disclaims all warranties with respect to this document including, without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement of intellectual property.

The terms and conditions governing the sale and licensing of Siemens products are set forth in written agreements between Siemens and its customers. Siemens' **End User License Agreement** may be viewed at: www.plm.automation.siemens.com/global/en/legal/online-terms/index.html.

No representation or other affirmation of fact contained in this publication shall be deemed to be a warranty or give rise to any liability of Siemens whatsoever.

**TRADEMARKS:** The trademarks, logos, and service marks ("Marks") used herein are the property of Siemens or other parties. No one is permitted to use these Marks without the prior written consent of Siemens or the owner of the Marks, as applicable. The use herein of third party Marks is not an attempt to indicate Siemens as a source of a product, but is intended to indicate a product from, or associated with, a particular third party. A list of Siemens' trademarks may be viewed at: www.plm.automation.siemens.com/global/en/legal/trademarks.html. The registered trademark Linux® is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a world-wide basis.

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

# Revision History

| Revision | Changes | Status/ Date |
|---|---|---|
| 9 | Modifications to improve the readability and comprehension of the content. Approved by Lucille Woo.<br>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 |
| 8 | Modifications to improve the readability and comprehension of the content. Approved by Lucille Woo.<br>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 |
| 7 | Modifications to improve the readability and comprehension of the content. Approved by Lucille Woo.<br>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 |
| 6 | Modifications to improve the readability and comprehension of the content. Approved by Lucille Woo.<br>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/.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Table of Contents

**Revision History**

**Chapter 1**
**Introduction to Calibre Interactive . . . . . 23**

Calibre Interactive Overview . . . . . 23
Interactive Workflow for Calibre. . . . . 24
Calibre Interactive System Requirements . . . . . 26
Integration to Design Tools . . . . . 26
Syntax Conventions . . . . . 29

**Chapter 2**
**Calibre Interactive Basics . . . . . 31**

Files and Directories Used By Calibre Interactive . . . . . 31
About Calibre Interactive Runsets . . . . . 33
How Calibre Interactive Uses Runset Data . . . . . 33
Default Runsets in Calibre Interactive . . . . . 34
Calibre Interactive Runset Contents . . . . . 35
Creating and Saving a Calibre Interactive Runset . . . . . 36
Methods to Load, Save, and Use Calibre Interactive Runsets . . . . . 37
Loading Runsets on Invocation . . . . . 39
Populating the Load Runset File Dialog Box with Runsets on Invocation . . . . . 40
Using an Environment Variable to Load Calibre Interactive Runset(s) on Invocation . . 41
Loading Multiple Runsets with a Startup Preference. . . . . 42
Using Multiple Calibre Interactive Runsets to Combine Different Sets of Options . . . . . 44
Save All GUI Options to the Runset File . . . . . 44
Runset Type and Runset Info . . . . . 45
About Control Files . . . . . 46
Control File in Calibre Interactive . . . . . 46
Control Filenames. . . . . 48
Creating and Viewing the Control File Without Starting a Run. . . . . 48
Running Batch Calibre with a Calibre Interactive Control File . . . . . 49
About Rule Files . . . . . 52
Rule File Operations in Calibre Interactive . . . . . 52
How Rule Files Relate to Control Files . . . . . 53
Features of the User Interface . . . . . 56
Calibre Interactive Buttons . . . . . 56
Setup Menu in Calibre Interactive . . . . . 57
Environment Variables in Calibre Interactive GUI Fields . . . . . 58
Color Coding in Calibre Interactive . . . . . 59
File Browser Options . . . . . 60

**Chapter 3**
**Running Calibre Interactive . . . . . 63**

Invoking Calibre Interactive Tools . . . . . 64
Invoking Calibre Interactive From a Layout or Schematic Viewer . . . . . 64
Invoking Calibre Interactive from the Command Line . . . . . 66
License Timeout in Calibre Interactive . . . . . 67
calibre -gui . . . . . 68
Communication with the Design Tool . . . . . 72
Using a Viewer as the Source for Design Input . . . . . 72
Setting Layout and Schematic Export in Calibre Interactive . . . . . 73
Methods for Populating the Calibre Interactive GUI . . . . . 75
Populating the Calibre Interactive GUI . . . . . 75
Automatically Loading a Rule File . . . . . 76
Managing Rule File and Calibre Interactive GUI Precedence . . . . . 77
Third Party Database Options . . . . . 80
MGC_CALIBRE_DB_READ_OPTIONS Environment Variable for Database Options . 81
OpenAccess/LEFDEF Options Tab . . . . . 82
Example: Setting the Layer Map, Cell Map, and Object Map for OpenAccess Databases 85
Providing LEF/DEF Input . . . . . 88
Read LEF/DEF Input from Design Files . . . . . 88
Export DEF Layout from Design Tool . . . . . 89
Providing OpenAccess Input . . . . . 90
Setting Database and General Options in Calibre Interactive . . . . . 93
Specifying a Layout Text File in Calibre Interactive . . . . . 93
Specifying Database Precision and Resolution in Calibre Interactive . . . . . 94
Specifying Third Party Database Options in Calibre Interactive . . . . . 94
Including Other Rule Files and SVRF Commands in Calibre Interactive . . . . . 95
Specifying Net Annotation Options When Reading Layout Databases . . . . . 96
Specifying Virtual Net Connections in Calibre Interactive . . . . . 96
Running a Job Using Calibre Interactive . . . . . 98
Starting a Job with Calibre Interactive . . . . . 98
Transcript Pane in Calibre Interactive . . . . . 99
Calibre Quality Agent for Calibre Interactive . . . . . 101
Running Calibre Interactive in Batch Mode . . . . . 102
Starting a Basic Batch Run in Calibre Interactive . . . . . 102
Rule Check Selection in Calibre Interactive Batch Runs . . . . . 103
Calibre Interactive nmDRC and DFM Check Selection in Batch Mode . . . . . 103
Calibre Interactive nmLVS ERC Check Selection in Batch Mode . . . . . 103
Specifying Layout and Schematic Export for a Calibre Interactive Batch Run . . . . . 104
Calibre Interactive Operation with a TVF Rule File . . . . . 105
Check Selection in Calibre Interactive . . . . . 108
Check Selection in Calibre Interactive nmDRC and Calibre Interactive DFM . . . . . 109
Check Selection Recipe . . . . . 109
Specifying a Check Selection Recipe . . . . . 110
Creating a Custom Check Selection Recipe . . . . . 110
Basic Editing of Custom Check Selection Recipes . . . . . 111
Advanced Editing of Check Selection Recipes in Calibre Interactive . . . . . 113
Running Only Checks Selected in the Rule File with a Custom Check Selection Recipe 116

Check Selection Recipe Editor Dialog Box . . . . . 118
Check Selection for ERC in Calibre Interactive nmLVS . . . . . 125
Viewing and Editing Text Files in Calibre Interactive . . . . . 125
Layer Derivations Tree . . . . . 126
Viewing and Editing Environment Variables . . . . . 130
Set Environment Variables Dialog Box . . . . . 131
Examples for Creating and Modifying Environment Variables . . . . . 135
Setup for Environment Variable Examples . . . . . 135
Adding a Runset Environment Variable Example . . . . . 136
Editing a Runset or Rule File Environment Variable Example . . . . . 137
Changing the Value of a Shell Environment Variable Example . . . . . 139
Setup Preferences . . . . . 142
Saving Setup Preferences . . . . . 143
Startup Preferences . . . . . 144
Startup Tab for Setup Preferences . . . . . 145
Configuration Files for Startup Preferences . . . . . 146
Templates for File Naming . . . . . 148
Example for File Naming Template . . . . . 149
Disabling Templates So That Saved Runset Values Are Used . . . . . 150
Export from Viewer Preferences . . . . . 152
Disable Preferences . . . . . 154
Miscellaneous Preferences . . . . . 156

**Chapter 4**
**Using Calibre Interactive to Perform DRC . . . . . 161**

Workflow with Calibre Interactive nmDRC . . . . . 161
Calibre Interactive - nmDRC Requirements . . . . . 163
Specifying a Rule File and Run Directory for a DRC Run . . . . . 163
Specifying the Checks to Execute in a Calibre Interactive nmDRC Run . . . . . 165
Specifying the Run Type for a DRC Run . . . . . 166
Inputs for a Calibre Interactive nmDRC Run . . . . . 168
Reading Layout Input from an Open Design Tool in Calibre Interactive nmDRC . . . . . 168
Reading Layout Input from File in Calibre Interactive nmDRC . . . . . 170
Specifying Multiple Layouts for a DRC Run . . . . . 172
Specifying Automatic Waivers for DRC . . . . . 173
Specifying Outputs for a DRC Run . . . . . 173
Running Calibre Interactive nmDRC . . . . . 178
Specifying Run Control and Licensing Options . . . . . 179
DRC Options in Calibre Interactive . . . . . 180
Preserving Cells During DRC . . . . . 180
Limiting the DRC Results Count . . . . . 181
Specifying the Rule Check Text Written to the Results Database . . . . . 182
Reporting DRC Results in Local Cell Space or Top Cell Space . . . . . 183
Creating a Check Text Override (CTO) File Using Calibre Interactive nmDRC . . . . . 183
Creating a DRC HTML Report from Calibre Interactive nmDRC . . . . . 185
Specifying the DFM Defaults Statement . . . . . 188
Performing Area DRC . . . . . 189
FastXOR Layout Comparison in Calibre Interactive . . . . . 192

Performing a FastXOR Layout Comparison Run in Calibre Interactive . . . . . 192
Rule File Generation Tab . . . . . 196
Fast XOR Options Tab . . . . . 198

**Chapter 5**
**Calibre Auto-Waivers in Calibre Interactive . . . . . 201**

Introduction to Calibre Auto-Waivers in Calibre Interactive . . . . . 202
Requirements for Calibre Auto-Waivers . . . . . 203
Calibre Auto-Waivers Flow in Calibre Interactive and Calibre RVE . . . . . 204
Generating Waiver Shapes with Calibre Interactive . . . . . 205
Starting a Waiver Run in Calibre Interactive. . . . . 209

**Chapter 6**
**Using Calibre Interactive to Perform DFM . . . . . 215**

Introduction to Calibre Interactive DFM . . . . . 215
Specifying a Rule File for a DFM Run . . . . . 217
Specifying the Checks to Execute in a Calibre Interactive DFM Run . . . . . 218
Specifying the Run Type for a Calibre Interactive DFM Run. . . . . 219
Inputs for a Calibre Interactive DFM Run. . . . . 221
Reading Layout Input from an Open Design Tool in Calibre Interactive DFM. . . . . 221
Reading Layout Input from File in Calibre Interactive DFM. . . . . 223
Specifying Multiple Layouts for a DFM Run. . . . . 225
Specifying Automatic Waivers for DFM CFA . . . . . 226
Outputs for a Calibre Interactive DFM Run . . . . . 227
Specifying Outputs for a DFM Run . . . . . 227
Specifying Layers and Annotations for DFM Results Database Output . . . . . 229
Running a YieldServer Script on the DFM Database. . . . . 231
Run Control, Remote Execution and Licensing Options. . . . . 232
DFM Options . . . . . 233
Limiting the DFM Results Count . . . . . 233
Specifying DFM Database Options . . . . . 233
Reporting DFM Results in Cell Space . . . . . 234
Performing Area DFM . . . . . 234

**Chapter 7**
**Using Calibre Interactive to Perform LVS. . . . . 237**

Calibre Interactive nmLVS Overview . . . . . 237
Specifying a Rule File for a Calibre nmLVS Run . . . . . 239
Specifying the Calibre nmLVS Run and Analysis Type . . . . . 240
Inputs for a Calibre Interactive nmLVS Run. . . . . 243
Layout Input in Calibre Interactive nmLVS . . . . . 244
Reading Geometric Layout Input from an Open Design Tool in Calibre Interactive nmLVS 244
Reading Geometric Layout Input from File in Calibre Interactive nmLVS . . . . . 245
Specifying Multiple Layouts for an Calibre nmLVS Run . . . . . 247
Reading Layout Netlist Input from an Open Design Tool in Calibre Interactive nmLVS 248
Reading Layout Netlist Input from File in Calibre Interactive nmLVS. . . . . 249
Source Netlist Input in Calibre Interactive nmLVS . . . . . 250

Reading Source Netlist Data from an Open Schematic Viewer in Calibre Interactive nmLVS 250
Reading Source Netlist Data from File in Calibre Interactive nmLVS . . . . . 252
Setting Up the Verilog Translator (v2lvs) . . . . . 254
Providing Additional SPICE Netlist Files with the Source Netlist . . . . . 257
Supplying Hcell Data . . . . . 258
Performing Hcell Analysis in Calibre nmLVS . . . . . 260
Setting up the Hcell Analysis Run . . . . . 260
H-cells Tab for Viewing Hcell Analysis Results . . . . . 261
Specifying Automatic Waivers for ERC . . . . . 262
Specifying Outputs for a Calibre nmLVS Run . . . . . 263
Running Calibre Interactive nmLVS . . . . . 264
Run Control, Remote Execution and Licensing Options . . . . . 266
Calibre nmLVS Options in Calibre Interactive . . . . . 267
Setting Power Supply Options . . . . . 267
Aborting on Sconnect Errors . . . . . 268
Ignoring Ports . . . . . 269
Specifying Calibre nmLVS Report Options . . . . . 269
Reporting Seed Promotions . . . . . 270
Controlling Logic Gate Recognition . . . . . 271
Setting Split Gate and Parallel MOS Reduction Options . . . . . 271
Setting Filter Unused Device Options . . . . . 272
Changing Trace Property Rules . . . . . 274
Changing LVS Box Statements . . . . . 276
Invoking Short Isolation . . . . . 278
Setting Options for Debug with Calibre RVE for LVS . . . . . 280
Running ERC Checks . . . . . 282
Selecting ERC Checks . . . . . 284
Device Signature Creation with Calibre Interactive nmLVS . . . . . 286
Generating Device Signatures for a Calibre nmLVS Rule File Using Calibre Interactive 286
Layers File for Device Signature Creation . . . . . 290
Edit Parameters for Device Dialog Box . . . . . 293

**Chapter 8**
**Using Calibre Interactive to Perform Parasitic Extraction . . . . . 295**

Introduction to Calibre Interactive PEX . . . . . 295
Specifying a Rule File for PEX Run . . . . . 299
Inputs for a PEX Run . . . . . 301
Specifying the Layout Input in Calibre Interactive PEX . . . . . 301
Specifying Multiple Layouts for a PEX Run . . . . . 303
Specifying Schematic Netlist Input . . . . . 304
Defining Hcell and Xcell Input (Gate-Level, Hierarchical Extraction and ADMS Only) . 307
Manipulating Cell Extraction Options Using the Blocks Tab . . . . . 308
Specifying Probe Points in Calibre Interactive PEX . . . . . 310
Outputs for a PEX Run . . . . . 312
Specifying the Extraction Mode and Level for a Calibre Interactive PEX Run . . . . . 312
Specifying Extracted Netlist Options for a Calibre Interactive PEX Run . . . . . 313
Specifying Report Options for a Calibre Interactive PEX Run . . . . . 315

Specifying Database Output Options for a Calibre Interactive PEX Run . . . . . 316
Naming the Extracted Parasitic Netlist Based on the Extraction Type and Format . . . . . 317
Calibre Interactive PEX Execution . . . . . 320
Running Calibre Interactive PEX . . . . . 320
Controlling PEX Step Execution . . . . . 321
Run Control, Remote Execution and Licensing Options . . . . . 322
Setting PEX Options . . . . . 324
Including/Excluding Nets . . . . . 325
Specifying Netlist Formatting Options . . . . . 326
Specifying Netlist Reduction and Coupling Capacitance Options . . . . . 330
Specifying Advanced Netlist Options . . . . . 333
Specifying Via Reduction . . . . . 335
Specifying the Top Level Pin Order . . . . . 337
Specifying Which Parasitics to Include in the Netlist . . . . . 338
Specifying Parameters for Resistance Calculations . . . . . 338
Specifying Temperature Parameters for Resistance Extraction . . . . . 339
Specifying Extracted Netlist Annotation of Net Geometries to Calibre View . . . . . 339
Adding the PRIMETIME command to a PEX NETLIST Statement . . . . . 342
Setting PEX Power and Ground Nets . . . . . 342
Generating a Driver/Receiver File . . . . . 342
Specifying a SPICE Instance Prefix . . . . . 343
Controlling Inductance Extraction . . . . . 343
Setting Delay Calculation . . . . . 345
Controlling Message Output . . . . . 346
Changing Trace Property Rules for a PEX Run . . . . . 347
Setting LVS and Netlist Extraction Options in a PEX Run . . . . . 348
Setting LVS Options for PEX . . . . . 348
Logic Gate Recognition Options in PEX . . . . . 349
Split Gate and Parallel MOS Reduction Options in PEX . . . . . 349
Filter Unused Device Options in PEX . . . . . 349

**Chapter 9**
**Using Calibre Interactive xACT . . . . . 351**

Requirements for Calibre Interactive xACT . . . . . 352
Invoking the Calibre Interactive xACT GUI . . . . . 352
Calibre Interactive xACT Command Line . . . . . 355
Setting Up and Starting a Calibre Interactive xACT Run . . . . . 357
Tips for Using the Calibre Interactive xACT GUI . . . . . 359
Populating the Load Runset Files Dialog Box and Recent Files Lists . . . . . 360
Using a Calibre Interactive PEX Runset with Calibre Interactive xACT . . . . . 363
Loading a Classic Tcl Customization File in Calibre Interactive xACT . . . . . 365
Providing OpenAccess Input with Calibre Interactive xACT . . . . . 366
Specifying Probe Points in Calibre Interactive xACT . . . . . 368
Setting Options in Calibre Interactive xACT . . . . . 370
Connecting to a Design Tool in Calibre Interactive xACT . . . . . 370
Adding and Changing LVS Box Statements in Calibre Interactive xACT . . . . . 372
Calibre Interactive xACT GUI Reference . . . . . 375
Outputs Page in Calibre Interactive xACT . . . . . 378

LVS Page in Calibre Interactive xACT . . . . . 381
Database Page in Calibre Interactive xACT . . . . . 383
OpenAccess Page in Calibre Interactive xACT . . . . . 385
Environment Page in Calibre Interactive xACT . . . . . 389
Run Control Page in Calibre Interactive xACT . . . . . 392
Triggers Page in Calibre Interactive xACT . . . . . 395
Templates Page in Calibre Interactive xACT . . . . . 397
Search Toolbar in Calibre Interactive xACT . . . . . 399
Preferences Page in Calibre Interactive xACT . . . . . 401
Rules Preferences . . . . . 402
Runset Preferences . . . . . 404
Miscellaneous Preferences in Calibre Interactive xACT . . . . . 405
GUI Customization for Calibre Interactive xACT . . . . . 406
Loading a Configuration File in Calibre Interactive xACT . . . . . 406
Configuration File Reference . . . . . 409
Configuration File Format for xACT GUI Customization . . . . . 410
Global Object Definitions . . . . . 413
Arrays and Maps . . . . . 414
Tool Properties . . . . . 418
Option Properties . . . . . 421
GUI Signals in Calibre Interactive xACT . . . . . 422
Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT . . . 423
Finding Option Names to Use in the Configuration File . . . . . 425
Function Reference for GUI Customization in Calibre Interactive xACT . . . . . 427
General Configuration File Functions . . . . . 428
Configuration Commands to Execute Functions and Commands . . . . . 456
Configuration File Functions to Add #DEFINE and #UNDEFINE Statements . . . . . 467
Configuration File Functions to Add Variable Statements . . . . . 505
Configuration File Functions for Trigger Functions . . . . . 543
Configuration File Functions to Add Buttons . . . . . 548
Troubleshooting Configuration Files for the Calibre Interactive xACT GUI . . . . . 556

**Chapter 10**
**Using Calibre Interactive to Perform PERC . . . . . 557**

Introduction to Calibre Interactive PERC . . . . . 557
Specifying a Rule File for a Calibre PERC Run . . . . . 560
Specifying the PERC Run and Analysis Type . . . . . 561
Inputs for a Calibre Interactive PERC Run . . . . . 563
Specifying Inputs for a Calibre PERC Run . . . . . 563
Input Sources for a Calibre PERC Run . . . . . 564
Specifying Calibre PERC Inputs for Layout Analysis . . . . . 565
Reading Geometric Layout Input from an Open Design Tool in Calibre Interactive PERC 565
Reading Geometric Layout Input from File in Calibre Interactive PERC . . . . . 567
Specifying Calibre PERC Inputs for Layout Netlist Analysis . . . . . 568
Specifying Calibre PERC Inputs for Source Netlist Analysis . . . . . 569
Reading Source Netlist Data from a Schematic Viewer in Calibre Interactive PERC . . 569
Reading Source Netlist Data from File in Calibre Interactive PERC . . . . . 571

Supplying Hcell Data for a Calibre PERC Run . . . . . 572
Specifying a Calibre PERC LDL Probe Path . . . . . 573
Specifying Multiple Layouts for a Calibre PERC Run. . . . . 573
Specifying Waivers for a Calibre PERC Run . . . . . 574
Specifying the Calibre PERC Cell-Based Flow . . . . . 575
Creating a Calibre PERC Rule File with High Level Checks . . . . . 576
Specifying Outputs for a Calibre PERC Run. . . . . 578
Running Calibre Interactive PERC . . . . . 579
Run Control, Remote Execution, and Licensing Options . . . . . 581
Setting Calibre PERC Options. . . . . 582
Setting Power Supply Options . . . . . 582
Aborting Calibre PERC on Sconnect Errors. . . . . 583
Specifying PERC Report Options and Result Count . . . . . 584
Controlling Logic Gate Recognition in Calibre PERC. . . . . 585
Setting Split Gate and Parallel MOS Reduction Options in Calibre PERC . . . . . 585
Setting Filter Unused Device Options in Calibre PERC . . . . . 585
Changing Trace Property Rules for the Calibre PERC Run. . . . . 586

**Chapter 11**
**Customization Files. . . . . 587**

Customization Overview . . . . . 588
Getting Started with Customization Files . . . . . 589
Customization GUI Operation. . . . . 591
Methods to Specify a Customization File. . . . . 591
Methods to Open the Customization Settings Dialog Box. . . . . 593
Customization Settings Dialog Box . . . . . 594
Saving Customization Settings. . . . . 597
Customization Overrides . . . . . 598
cmnCustomFileOverrideValues Syntax. . . . . 599
Customization GUI Operation in Batch Mode . . . . . 602
Methods to Specify the Customization File for Calibre Interactive Batch Mode . . . . . 602
Saving Customization GUI Settings in Calibre Interactive Batch Mode Runs . . . . . 603
Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive. 604
Customization Language . . . . . 606
Tcl/Tk Basics . . . . . 606
Tcl Lists . . . . . 607
Tcl/Tk References. . . . . 608
Customization Commands. . . . . 610
Customization Control Summary . . . . . 610
CUSTOM::CHECK . . . . . 613
CUSTOM::DEFINE . . . . . 617
CUSTOM::LABEL. . . . . 622
CUSTOM::SEPARATOR . . . . . 623
CUSTOM::SETWINGEOM. . . . . 624
CUSTOM::VARIABLE . . . . . 625
CUSTOM::getCalibreVersionString . . . . . 632
CUSTOM::getRunsetVarValue . . . . . 633
CUSTOM::runsetVarExists . . . . . 635

CUSTOM::setRunsetVarValue . . . . . 636
CUSTOM::setOKCallback . . . . . 638
CUSTOM::setOKCallbackError . . . . . 640
Primary-Dependent Commands . . . . . 640
Input Type Checking for Customization GUI Entry Fields . . . . . 644
Config and Cget Options . . . . . 646
Customization File Examples . . . . . 648
Customization File to Read and Set Calibre Interactive GUI Controls . . . . . 648
Customization File to Select ERC Rule Checks . . . . . 651
General Customization Example . . . . . 652
Customization File to Load Multiple Customization Files . . . . . 655

**Chapter 12**
**Trigger Functions in Calibre Interactive . . . . . 657**

Trigger Overview . . . . . 657
Internal Triggers in Calibre Interactive . . . . . 659
Trigger Parameters for Internal Triggers in Calibre Interactive . . . . . 659
Setting Calibre Interactive Internal Triggers . . . . . 661
Trigger Access to the Design Tool Environment . . . . . 665
Trigger Options in the Runset . . . . . 665
Calibre Interactive Internal Trigger Examples . . . . . 668
Internal Trigger for Calling a UNIX Utility . . . . . 668
Internal Trigger for Calling a Script . . . . . 668
Internal Trigger to Set an SVRF Variable . . . . . 668
Internal Trigger Executed in Calibre DESIGNrev to Count Layout Cells . . . . . 670
Applying Triggers According to the Calibre Interactive Application . . . . . 671
External Triggers in Calibre Interactive . . . . . 673
External Trigger Types . . . . . 673
External Trigger Functions and File Locations . . . . . 675
External Trigger Definitions in Calibre DESIGNrev . . . . . 675
External Trigger Definitions in Cadence Virtuoso . . . . . 676
External Trigger Definitions in Synopsys IC Compiler . . . . . 677
Calibre Interactive External Trigger Examples . . . . . 678
External Trigger in Tcl . . . . . 678
External Trigger in SKILL for Cadence Virtuoso . . . . . 678

**Chapter 13**
**Run Control Setup and Licensing . . . . . 681**

Specifying the Host Processor and Run Options . . . . . 681
Computer Cluster Options . . . . . 686
Using the Spectrum LSF Option . . . . . 686
Example Spectrum LSF Configurations . . . . . 690
Using the Univa Grid Engine Option . . . . . 691
Using the Other Cluster Option . . . . . 693
Replaceable Parameters for the Other Cluster Queue Command . . . . . 694
Configuring for Distributed (MTflex) Execution . . . . . 696
Configuring the Remote Environment . . . . . 699
Setting Licensing Options . . . . . 701

Reference for Run Control Options . . . . . 703
Hyperscaling Mode . . . . . 703
Hyperscale Compare Option in Calibre Interactive nmLVS . . . . . 703
Hyperscale Pathchk Option . . . . . 703
Remote Data Server (RDS) Option . . . . . 704
Hyperscale Remote Option . . . . . 704
Backup Data Option . . . . . 704

**Appendix A**
**Interfacing Calibre Interactive to Layout and Schematic Viewers . . . . . 705**

Basic Interface . . . . . 706
Calibre Interactive . . . . . 706
Setting the Socket Port in Calibre Interactive . . . . . 707
Setting the Calibre Interactive Socket Port for a Layout Viewer . . . . . 707
Setting the Calibre Interactive Socket Port for a Schematic Viewer . . . . . 708
Custom Menu Items in Design Tools . . . . . 709
Creating Custom Menus in Design Tools . . . . . 710
Menu Customization File . . . . . 711
Menu Customization Commands . . . . . 712
mgc_calibre_get_viewer_name . . . . . 713
mgc_calibre_add_menu_item . . . . . 714
mgc_calibre_delete_menu_item . . . . . 718
Calibre Integration Menu Labels and Backward Compatibility for Menu Customization Files 718
Menu Customization Example . . . . . 719
Calibre DESIGNrev and WORKbench . . . . . 721
Setting Socket Connections with Calibre DESIGNrev or Calibre WORKbench . . . . . 721
Calibre Interactive with Calibre DESIGNrev or WORKbench . . . . . 722
IC Station/Pyxis . . . . . 723
Setting Socket Connections with Pyxis . . . . . 723
Using Calibre Interactive with Pyxis . . . . . 723
Siemens Tanner . . . . . 724
Cadence Virtuoso . . . . . 725
Creating an Interface to Cadence Virtuoso . . . . . 725
The Calibre Menu in Cadence Virtuoso . . . . . 727
Layout Export in Cadence Virtuoso . . . . . 730
Calibre Layout Export Setup Dialog Box in Cadence Virtuoso . . . . . 731
Using a Template File for Layout Export with Cadence Virtuoso . . . . . 734
SKILL Functions for Layout Export . . . . . 736
Direct Read of Cadence Virtuoso OpenAccess Database . . . . . 738
Automatically Saving the Express Pcell Cache . . . . . 738
Netlist Export in Cadence Composer . . . . . 740
Calibre Netlist Export Setup Dialog Box in Cadence Composer . . . . . 741
SKILL Functions for Netlist Export . . . . . 744
Using the Calibre - Virtuoso Interface . . . . . 745
Setting Socket Connections with Cadence Virtuoso . . . . . 747
Calling SKILL Functions from Calibre Interactive . . . . . 748

Creating a Calibre View . . . . . 749
What You Need Before Creating a Calibre View . . . . . 750
What a Calibre View Produces. . . . . 751
Automatic OpenAccess Version Detection in Calibre View . . . . . 751
Device Property Calculation in Calibre View Generation . . . . . 752
Handling of Nets With Inherited Connectivity . . . . . 753
Bus Delimiter in Calibre View. . . . . 754
Identifying Devices with a Cellmap File . . . . . 754
Creating the Cellmap File. . . . . 754
Syntax for Cellmap Statements . . . . . 756
Cellmap Examples . . . . . 766
Setting the Calibre View Cellmap File in Calibre Interactive . . . . . 773
Importing Schematic Properties for Calibre View. . . . . 774
Auto-Mapping Behavior, Control, and Debug in Calibre View Generation . . . . . 775
Calibre View Setup Dialog Box. . . . . 777
Creating a Post-LVS Calibre View . . . . . 780
Creating a Post-Parasitic Extraction Calibre View . . . . . 782
Creating a Gate-Level Calibre View . . . . . 786
Creating Calibre Views in a Multiple Corner Extraction Run . . . . . 789
Creating a SPECTRE Netlist from the Calibre View Netlist. . . . . 790
Creating a CalibreView Setup File. . . . . 791
Showing Parasitic Polygons in Calibre View. . . . . 792
CalibreView Setup File Format . . . . . 794
SKILL Functions for Creating a Calibre View . . . . . 798
Global Variables for Calibre View Generation . . . . . 800
Creating a Calibre View in Batch Mode . . . . . 801
Calibre View and IC Manage Revision Control. . . . . 802
Calibre View Generation Warnings . . . . . 802
Setting the Bus Delimiter for Schematic Netlists. . . . . 808
Convert Bus Name Delimiter During Schematic Netlist Export . . . . . 808
Automatically Convert Bus Name Delimiter for Calibre Interactive and Calibre RVE . 808
Mapping Schematic Nets to Calibre View Nets for Simulation and Plotting . . . . . 810
Mapping Schematic Nets to Calibre View Nets with a Single Extracted View for Simulation and Plotting. . . . . 810
Mapping Schematic Nets to Calibre View Nets with Multiple Extracted Views for Simulation and Plotting. . . . . 811
Annotating Point to Point Resistance in Calibre View . . . . . 812
Backannotation of Parasitic Values to the Schematic in Cadence Virtuoso. . . . . 813
Automating Calibre Interactive Runsets in the Cadence Design Environment . . . . . 814
Cadence Composer . . . . . 816
Cadence Encounter . . . . . 817
Creating an Interface to Cadence Encounter . . . . . 817
Importing saveNetlist Parameters for Verilog Export in Cadence Encounter . . . . . 818
Cadence Innovus . . . . . 820
Creating an Interface to Cadence Innovus . . . . . 820
Synopsys IC Compiler and Synopsys IC Compiler II. . . . . 822
Calibre Flow with Synopsys IC Compiler . . . . . 823
Calibre Interface to Synopsys IC Compiler . . . . . 824
Loading the Calibre Interface to Synopsys IC Compiler . . . . . 824

Loading the Calibre Interface to Synopsys IC Compiler II . . . . . 825
README File for the Calibre-IC Compiler Integration . . . . . 827
Calibre Menu in Synopsys IC Compiler . . . . . 827
Running Calibre Interactive with Synopsys IC Compiler . . . . . 828
Setting Socket Connections with Synopsys IC Compiler and IC Compiler II . . . . . 831
Troubleshooting the Calibre Interface with Synopsys IC Compiler. . . . . 832
Silvaco Expert . . . . . 834
Creating the Interface to Silvaco Expert. . . . . 834
Setting Socket Connections with Silvaco Expert . . . . . 834
Calibre Menu in Silvaco Expert . . . . . 834
Other Viewer Integrations . . . . . 835

**Appendix B**
**Environment Variables . . . . . 837**

Environment Variables for Calibre Interactive . . . . . 837
Environment Variables for Calibre View . . . . . 843

**Appendix C**
**Runset Options . . . . . 847**

Calibre Interactive Runset File Format . . . . . 847
Creating a Calibre Interactive Runset . . . . . 848
Option Descriptions . . . . . 849
DRC Runset Options . . . . . 849
DFM Runset Options . . . . . 859
LVS Runset Options . . . . . 875
PERC Runset Options. . . . . 887
PEX Runset Options . . . . . 895
Common Runset Options . . . . . 918
Waiver Setup Runset Options. . . . . 944
3DSTACK Runset Options. . . . . 949
Runset Options for Template Parameters . . . . . 960

**Appendix D**
**Runset Options for Calibre Interactive xACT. . . . . 965**

Calibre Interactive xACT Runset File Format. . . . . 965
General Runset Options for Calibre Interactive xACT . . . . . 966
xACT Runset Options . . . . . 990
Template Definitions in the Calibre Interactive xACT Runset . . . . . 1069
Trigger Functions for the Calibre Interactive xACT Runset . . . . . 1073

**Glossary**

**Index**

**Third-Party Information**

# List of Figures


Figure 1-1. Interactive Verification Flow . . . . . 25
Figure 2-1. Specifying Multiple Runset Files Using Setup Preferences . . . . . 43
Figure 2-2. Runset Type and Runset Info Fields . . . . . 45
Figure 2-3. Relating User Input to Control Files . . . . . 47
Figure 2-4. Relating Control Files to Rule Files . . . . . 55
Figure 2-5. File Browser . . . . . 61
Figure 3-1. Selecting the Calibre Interactive Application in Calibre DESIGNrev . . . . . 65
Figure 3-2. Specifying the Layer Map, Object Map, and Cell Map Files . . . . . 87
Figure 3-3. Transcript Pane in Calibre Interactive. . . . . 100
Figure 3-4. Calibre Quality Agent Dialog Box for Calibre Interactive . . . . . 101
Figure 3-5. Adding a Recipe Expression . . . . . 115
Figure 3-6. Check Selection Recipe Editor Basic Controls . . . . . 118
Figure 3-7. Check Selection Recipe Editor Advanced Controls . . . . . 119
Figure 3-8. Layer Derivations Tree . . . . . 127
Figure 3-9. Set Environment Variables Dialog Box . . . . . 132
Figure 3-10. Adding a Runset Environment Variable . . . . . 137
Figure 3-11. Editing a Runset or Rule File Environment Variable . . . . . 139
Figure 3-12. Changing the Value of a Shell Environment Variable . . . . . 141
Figure 3-13. Startup Preferences Tab . . . . . 145
Figure 3-14. Defining Templates for Generated Filenames . . . . . 150
Figure 3-15. Defining Export Preferences in the Setup Preferences Dialog . . . . . 153
Figure 4-1. Interactive Verification for Calibre nmDRC. . . . . 162
Figure 5-1. Calibre Auto-Waivers Flow in Calibre Interactive and RVE . . . . . 204
Figure 5-2. Waivers Tab for Waiver Creation . . . . . 206
Figure 6-1. Interactive Verification for DFM . . . . . 216
Figure 6-2. DFM Outputs Pane . . . . . 228
Figure 6-3. DFMDB Tab for DFM Options . . . . . 230
Figure 7-1. Generic LVS Flow . . . . . 238
Figure 7-2. H-Cells Analysis Result Tabs . . . . . 262
Figure 7-3. Properties Tab in Calibre Interactive nmLVS. . . . . 275
Figure 7-4. LVS Box Tab in Calibre Interactive nmLVS . . . . . 278
Figure 7-5. Select Checks Dialog Box. . . . . 285
Figure 7-6. Signatures Tab in LVS Inputs Pane . . . . . 287
Figure 7-7. Edit Parameters for Device Dialog Box . . . . . 293
Figure 8-1. Generic PEX Flow . . . . . 298
Figure 8-2. Inputs > Blocks Tab in Calibre Interactive PEX. . . . . 310
Figure 8-3. PEX Probes Tab . . . . . 311
Figure 8-4. Selecting PEX Steps for Execution. . . . . 322
Figure 8-5. Netlist > Vias Tab in Calibre Interactive PEX . . . . . 336
Figure 9-1. xACT Probes Page . . . . . 369

Figure 9-2. hyperlink Property Example . . . . . 424
Figure 10-1. Generic Calibre PERC Flow for Layout Analysis . . . . . 559
Figure 11-1. Customization Settings Dialog Box . . . . . 589
Figure 11-2. Customization Example to Set Calibre Interactive GUI Controls . . . . . 650
Figure 11-3. Customization GUI to Select Checks . . . . . 652
Figure 12-1. Trigger Pre- and Post-Execution Order . . . . . 658
Figure 12-2. Calibre Interactive Internal Triggers. . . . . 663
Figure 13-1. Spectrum LSF Dialog Box . . . . . 688
Figure 13-2. Univa Grid Engine Configuration. . . . . 693
Figure 13-3. Example Calibre MTflex Configuration With Local Host . . . . . 698
Figure 13-4. Remote MGC_HOME Table . . . . . 700
Figure A-1. Custom Menu Example for Calibre DESIGNrev. . . . . 710
Figure A-2. Cadence Technology File. . . . . 772
Figure A-3. Interactive Calibre Flow with Synopsys IC Compiler . . . . . 823
Figure A-4. Calibre Menu Contents in Synopsys IC Compiler. . . . . 828
Figure A-5. Calibre Menu in Synopsys IC Compiler . . . . . 829

# List of Tables

Table 1-1. Calibre Interactive Features . . . . . 23
Table 1-2. Calibre Integrations . . . . . 27
Table 1-3. Syntax Conventions . . . . . 29
Table 2-1. Filenames for Default Runsets . . . . . 34
Table 2-2. Loading, Saving, and Using Calibre Interactive Runsets . . . . . 37
Table 2-3. Automatically a Loading Calibre Interactive Runset on Invocation . . . . . 39
Table 2-4. Buttons in Calibre Interactive . . . . . 56
Table 2-5. Setup Menu Contents in Calibre Interactive . . . . . 57
Table 2-6. Advanced File Browser Options . . . . . 61
Table 3-1. Invoking Calibre Interactive from the Command Line . . . . . 66
Table 3-2. Overwrite File Dialog Box for Layout and Schematic Export . . . . . 74
Table 3-3. Methods for Setting Third Party Database Options . . . . . 80
Table 3-4. Read Options in the OpenAccess/LEFDEF Tab . . . . . 82
Table 3-5. Mapping Files Options in the OpenAccess/LEFDEF Tab . . . . . 83
Table 3-6. Additional Files Options in the OpenAccess/LEFDEF Tab . . . . . 84
Table 3-7. Other Options in the OpenAccess/LEFDEF Tab . . . . . 84
Table 3-8. Transcript Menu Contents in Calibre Interactive . . . . . 100
Table 3-9. Check Selection Recipe Editor — Action Buttons . . . . . 119
Table 3-10. Check Selection Recipe Editor — Basic Controls . . . . . 120
Table 3-11. Check Selection Recipe Editor — Advanced Controls . . . . . 121
Table 3-12. Expression Categories for Check Selection Recipes . . . . . 122
Table 3-13. Display Conventions for Layer Derivations Tree . . . . . 128
Table 3-14. Set Env Tab for Set Environment Variables Dialog Box . . . . . 133
Table 3-15. Setup Preferences Dialog Box Tabs . . . . . 142
Table 3-16. Customization and Check Recipe Settings in the Startup Preferences . . . . . 146
Table 3-17. Replaceable Parameters in Template Names . . . . . 148
Table 3-18. Export Preference Settings in the Setup Preferences Dialog . . . . . 153
Table 3-19. Disable Preferences . . . . . 154
Table 3-20. Miscellaneous Preferences . . . . . 156
Table 4-1. Sources for Layout Data . . . . . 168
Table 4-2. Output Options for All Results Database Formats (ASCII, GDS, OASIS) . . . . 176
Table 4-3. Output Options for an ASCII DRC Results Database . . . . . 176
Table 4-4. Output Options for a GDS or OASIS Output Database . . . . . 177
Table 4-5. Template Options for DRC HTML Report Creation . . . . . 187
Table 4-6. Format Options for DRC HTML Report Creation . . . . . 187
Table 4-7. Area DRC Options . . . . . 190
Table 4-8. Format Options for FastXOR Rule File Generation . . . . . 196
Table 4-9. Mapping Files Options for FastXOR Rule File Generation . . . . . 196
Table 5-1. Calibre Auto-Waivers Information Sources . . . . . 202
Table 6-1. DFM Run Modes and Result Formats . . . . . 219

Table 6-2. Sources for Layout Data in Calibre Interactive DFM . . . . . 221
Table 6-3. Area DFM Options . . . . . 235
Table 7-1. Inputs Pane in Calibre Interactive nmLVS . . . . . 243
Table 7-2. Verilog String Options . . . . . 255
Table 7-3. Verilog Translator Options . . . . . 256
Table 7-4. Changing Trace Property Statements in Calibre Interactive nmLVS . . . . . 275
Table 7-5. Changing LVS Box Statements in Calibre Interactive . . . . . 277
Table 7-6. Layers File for Device Signature Creation Contents . . . . . 290
Table 7-7. Device Parameter Fields for Create Device Signature . . . . . 293
Table 8-1. Files Required by the Calibre Interactive PEX GUI . . . . . 296
Table 8-2. Inputs Pane for PEX . . . . . 301
Table 8-3. Replaceable Parameters for Output PEX Netlist . . . . . 318
Table 8-4. PEX Netlist Formatting Options . . . . . 326
Table 8-5. Netlist Reduction and CC Options . . . . . 330
Table 8-6. Advanced Netlist Format Options . . . . . 333
Table 9-1. Tips for the Calibre Interactive xACT GUI . . . . . 359
Table 9-2. Loading a Classic Tcl Customization File . . . . . 365
Table 9-3. Calibre Interactive xACT GUI Items . . . . . 375
Table 9-4. Read Options on the OPENACCESS Page . . . . . 385
Table 9-5. Mapping Files Options on the OPENACCESS Page . . . . . 387
Table 9-6. Additional Files Options on the OPENACCESS Page . . . . . 387
Table 9-7. Other Options on the OPENACCESS Page . . . . . 387
Table 9-8. OpenAccess Environment Options on the OPENACCESS Page . . . . . 388
Table 9-9. Loading a Configuration File . . . . . 407
Table 9-10. Tool Properties . . . . . 418
Table 9-11. Common Option Properties . . . . . 421
Table 9-12. Signals from the Calibre Interactive GUI . . . . . 423
Table 9-13. Functions for FileIO Component . . . . . 447
Table 9-14. Obejct Properties for launchCommandReturnString . . . . . 465
Table 9-15. Object Properties for addChoicesDefine . . . . . 470
Table 9-16. Object Properties for addDefine . . . . . 474
Table 9-17. Object Properties for addInputDirTextDefine . . . . . 478
Table 9-18. Object Properties for addInputFileTextDefine . . . . . 481
Table 9-19. Object Properties for addIntegerDefine . . . . . 485
Table 9-20. Object Properties for addOutputDirTextDefine . . . . . 489
Table 9-21. Object Properties for addOutputFileTextDefine . . . . . 492
Table 9-22. Object Properties for addRealDefine . . . . . 495
Table 9-23. Object Properties for addTextDefine . . . . . 499
Table 9-24. Object Properties for addTextDefine . . . . . 503
Table 9-25. Object Properties for addChoicesVariable . . . . . 508
Table 9-26. Object Properties for addListVariable . . . . . 512
Table 9-27. Object Properties for addInputDirTextVariable . . . . . 515
Table 9-28. Object Properties for addInputFileTextVariable . . . . . 518
Table 9-29. Object Properties for addIntegerVariable . . . . . 521
Table 9-30. Object Properties for addMultiChoicesVariable . . . . . 524

Table 9-31. Object Properties for addOutputDirTextVariable . . . . . 528
Table 9-32. Object Properties for addOutputFileTextVariable . . . . . 531
Table 9-33. Object Properties for addRealVariable . . . . . 534
Table 9-34. Object Properties for addTextVariable . . . . . 538
Table 9-35. Object Properties for addTextVariable . . . . . 541
Table 9-36. Replaceable Trigger Parameters . . . . . 543
Table 9-37. Object Properties for addAction . . . . . 549
Table 9-38. Control Functions for addAction . . . . . 550
Table 9-39. Object Properties for addChoicesAction . . . . . 553
Table 9-40. Control Functions for addChoicesAction . . . . . 554
Table 10-1. Sources for Calibre PERC Input Data . . . . . 564
Table 11-1. Action Buttons in Customization Settings Dialog Box . . . . . 596
Table 11-2. Tcl Special Characters . . . . . 606
Table 11-3. Customization Command Types . . . . . 610
Table 11-4. Control File Statements in the Customization File . . . . . 611
Table 11-5. Runset Control Statements in the Customization File . . . . . 612
Table 11-6. Customization GUI Appearance Statements . . . . . 612
Table 11-7. Check State Table for CUSTOM::CHECK . . . . . 615
Table 12-1. Triggers in Calibre Interactive . . . . . 657
Table 12-2. Internal Trigger Parameters . . . . . 660
Table 12-3. Runset Options for Trigger Definition . . . . . 666
Table 12-4. File Locations for External Trigger Definitions in Calibre DESIGNrev . . . . . 675
Table 12-5. File Locations for External Trigger Definitions in Cadence Virtuoso . . . . . 676
Table 12-6. File Locations for External Trigger Definitions in Synopsys IC Compiler . . . 677
Table 13-1. Run Calibre On Options in the Run Control Pane . . . . . 682
Table 13-2. Run Calibre Execution Options in the Run Control Pane . . . . . 682
Table 13-3. Resource Option Variables for Spectrum LSF . . . . . 688
Table 13-4. Parameters for Other Cluster Queue Command . . . . . 695
Table A-1. Supported Design Tools for Custom Menu Items . . . . . 709
Table A-2. Return Value for mgc_calibre_get_viewer_name . . . . . 713
Table A-3. Verification Menu in Calibre DESIGNrev and WORKbench . . . . . 721
Table A-4. Calibre Menu in Cadence Virtuoso . . . . . 728
Table A-5. Calibre Setup Menu in Cadence Virtuoso . . . . . 728
Table A-6. Calibre Layout Export Setup Dialog Box Contents . . . . . 732
Table A-7. Calibre Netlist Export Setup Dialog Box Contents . . . . . 741
Table A-8. Defining Properties in the Cellmap File . . . . . 760
Table A-9. MGC_FDI_PIN_AUTOMAP_METHOD Values . . . . . 776
Table A-10. Calibre View Setup Dialog Box Contents . . . . . 777
Table A-11. CalibreView Setup File Keywords and Values . . . . . 794
Table A-12. Global Variables for Calibre View Generation . . . . . 800
Table A-13. Calibre View Warnings . . . . . 802
Table A-14. Calibre Menu in Cadence Encounter . . . . . 818
Table A-15. Calibre Menu in Cadence Innovus . . . . . 821
Table A-16. Calibre Menu in Silvaco Expert . . . . . 835
Table B-1. Calibre Interactive Environment Variables . . . . . 837

Table B-2. Calibre View Environment Variables . . . . . 844
Table C-1. DRC Runset Options . . . . . 849
Table C-2. DFM Runset Options . . . . . 859
Table C-3. LVS Runset Options . . . . . 875
Table C-4. PERC Runset Options . . . . . 887
Table C-5. PEX Runset Options . . . . . 895
Table C-6. Common Runset Options . . . . . 919
Table C-7. Waiver Setup Runset Options . . . . . 944
Table C-8. Calibre 3DSTACK Runset Options . . . . . 949
Table C-9. Runset Parameters for Template Settings . . . . . 960
Table D-1. General Runset Options for Calibre Interactive xACT . . . . . 967
Table D-2. General Runset Options for Calibre Interactive xACT . . . . . 991
Table D-3. Replaceable Parameters in Template Values for Calibre Interactive xACT . . . 1072

# Chapter 1
# Introduction to Calibre Interactive

Calibre® Interactive™ provides a GUI interface to several Calibre tools. The interface enables you to customize rule file settings and run-time options. You can submit jobs on local, remote, or distributed computer resources.

**Note**

As of the 2019.2 release, information on Calibre RVE is now in the *Calibre RVE User's Manual*.


**Calibre Interactive Overview** . . . . . 23
**Interactive Workflow for Calibre** . . . . . 24
**Calibre Interactive System Requirements** . . . . . 26
**Integration to Design Tools** . . . . . 26
**Syntax Conventions** . . . . . 29


## Calibre Interactive Overview

Calibre Interactive is a GUI interface to Calibre nmDRC, Calibre nmLVS, Calibre parasitic extraction tools, Calibre PERC, Calibre DFM, and Calibre 3DSTACK. You can easily adjust rule file settings and run-time options without editing the rule file, then submit the Calibre job on a local, remote, or distributed computer resource.

Calibre Interactive is integrated to most standard industry design tools, as discussed in the section "Integration to Design Tools" on page 26. When Calibre Interactive is connected to an integrated design tool you can make changes in the layout or schematic, then run Calibre on the modified version; the current layout or schematic is exported to Calibre prior to the start of the run.

Calibre Interactive includes the features listed in the following table.

**Table 1-1. Calibre Interactive Features**

| Feature | Reference |
|---|---|
| Customize rule file settings and run-time options without editing the rule file | "Running Calibre Interactive" on page 63 |
| Submit jobs on local, remote, or distributed computer resources | "Run Control Setup and Licensing" on page 681 |




Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 1-1. Calibre Interactive Features (cont.)**

| Feature | Reference |
|---|---|
| Export the design from a connected design tool | "Using a Viewer as the Source for Design Input" on page 72 |
| Specify scripts and commands to run before and after the Calibre job | "Trigger Functions in Calibre Interactive" on page 657 |
| Run without opening the GUI (Calibre Interactive batch mode) | "Running Calibre Interactive in Batch Mode" on page 102 |
| Create a user-coded customization GUI to control certain rule file and GUI settings | "Customization Files" on page 587 |
| Set environment variables for the run | "Viewing and Editing Environment Variables" on page 130 |
| Save GUI settings in a "runset" and load them in a later Calibre Interactive session | "About Calibre Interactive Runsets" on page 33 |
| Add custom menu items to the Calibre integration menu in supported design tools | "Custom Menu Items in Design Tools" on page 709 |

**Note**

For details on using these feature in Calibre Interactive xACT, see "Using Calibre Interactive xACT" on page 351.

# Interactive Workflow for Calibre

Calibre® Interactive™ and Calibre® RVE™ are essential components of an interactive workflow using Calibre physical verification tools. Calibre Interactive and RVE are integrated to most standard industry design tools,

The following figure shows how you can use Calibre Interactive and Calibre RVE in an interactive design process.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 1-1. Interactive Verification Flow**

The preceding figure shows a general, high-level flow diagram; for more detailed flows, refer to the following topics:

- "Workflow with Calibre Interactive nmDRC" on page 161
- "Introduction to Calibre Interactive DFM" on page 215
- "Calibre Interactive nmLVS Overview" on page 237
- "Introduction to Calibre Interactive PEX" on page 295
- "Introduction to Calibre Interactive PERC" on page 557

For information on Calibre RVE, see the *Calibre RVE User's Manual*.

## Related Topics

Integration to Design Tools



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Interactive System Requirements

Certain requirements for the computing platform, licensing, and environment variables must be met in order to run Calibre Interactive.

The following list gives the requirements for Calibre Interactive:

- **Platform support** — Calibre Interactive is available on all supported platforms found in the *Calibre Administrator's Guide*.
- **Licensing** — Calibre Interactive requires a Calibre Interactive product license. Further license requirements depend upon the Calibre application that is run.

  See the *Calibre Administrator's Guide* for complete licensing information.

  See "License Timeout in Calibre Interactive" on page 67 for information on setting the license timeout period.
- **Environment variable for software tree** — Before invoking Calibre Interactive, you must set an environment variable to the location of your Calibre software tree. See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide* for details.
- **Layout Design Tool** (Optional) — Access to a layout editor or viewer with a Calibre integration is optional. Calibre Interactive can export the current design from a connected viewer.

  Integrated layout viewers include Calibre DESIGNrev, Siemens Sierra Olympus/Pinnacle, Siemens Tanner L-Edit, Cadence Virtuoso, and Synopsys IC Compiler. See "Integration to Design Tools" on page 26 for a complete list of integrated viewers.
- **Schematic Design Tool** (Optional) — Access to a schematic viewer with a Calibre integration is optional. Some applications, such as Calibre nmLVS, Calibre PERC, and PEX, require the source schematic as input. Calibre Interactive can export the current schematic from a connected schematic viewer.

  Integrated schematic viewers include Siemens IC Station/Pyxis Schematic, Siemens Tanner S-Edit, and Cadence Composer. See "Integration to Design Tools" on page 26 for a complete list of integrated viewers.

### Related Topics

Calibre Interactive Overview

# Integration to Design Tools

Calibre Interactive and Calibre RVE can establish connections to most standard industry design tools. This section describes where to find setup and operation information.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Table 1-2 lists the design tools that support a Calibre integration. The Setup and Operation Information column indicates whether the integration is automatic or requires configuration.

The Calibre Interfaces web page has additional information about the integration to different design tools:

www.mentor.com/products/ic_nanometer_design/calibre-integration/

**Table 1-2. Calibre Integrations**

| Design Tool | Setup and Operation Information | Support by |
|---|---|---|
| **Custom Design Tools** | | |
| IC Station®, Pyxis™ Layout, and Pyxis™ Schematic | No setup required. See IC Station/Pyxis for integration information. | Siemens Digital Industries Software |
| L-Edit and S-Edit | Setup required.<br>See "Tanner AMS and MEMS Flows" on mentor.com for general product information. | Siemens Digital Industries Software |
| Cadence® Virtuoso® | See Creating an Interface to Cadence Virtuoso and Cadence Virtuoso. | Siemens Digital Industries Software |
| Synopsys Laker™ (layout only) | No setup required. See "Synopsys" on the Calibre Interfaces web page for additional information. | Synopsys |
| Synopsys® Custom Compiler™ | See "Synopsys" on the Calibre Interfaces web page. | Synopsys |
| Keysight | See "Keysight" on the Calibre Interfaces web page for additional information. | Agilent |
| PhoeniX OptoDesigner | No setup required. See "PhoeniX Software" on the Calibre Interfaces web page for additional information. | Phoenix Software |
| Silvaco Expert | No setup required. See Silvaco Expert in this manual for more information.<br>Also see "Silvaco" on the Calibre Interfaces web page for additional information. | Silvaco |
| **Place and Route Tools** | | |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 1-2. Calibre Integrations (cont.)**

| Design Tool | Setup and Operation Information | Support by |
|---|---|---|
| Siemens Olympus-SoC™ | No setup required. | Siemens Digital Industries Software |
| Synopsys® IC Compiler™ and Synopsys IC Compiler II (ICC2) | See Synopsys IC Compiler and Synopsys IC Compiler II. | Siemens Digital Industries Software |
| Cadence® Encounter® | See Cadence Encounter. | Siemens Digital Industries Software |
| Cadence® Innovus® | See Cadence Innovus. | Siemens Digital Industries Software |
| **Layout Editors and Viewers** | | |
| Calibre® DESIGNrev™ and Calibre® WORKbench™ | No setup required. See Calibre DESIGNrev and WORKbench for operational information. | Siemens Digital Industries Software |
| Cadence® QuickView Signoff Data Analysis Environment | No setup required. The integration is to Calibre RVE only. See "Cadence QuickView" in the *Calibre RVE User's Manual*. | Cadence |
| TOOL Corporation LAVIS | See "TOOL" on the Calibre Interfaces web page.<br>Supports the interface to Calibre RVE; no setup required. | TOOL |
| AnaGlobe Technology | See "AnaGlobe Technology" on the Calibre Interfaces web page.<br>Supports the standard interfaces to Calibre Interactive and Calibre RVE. No setup required; on by default. | AnaGlobe Technology |
| Jedat a-SX Ismo | See "Jedat" on the Calibre Interfaces web page.<br>Supports the standard interfaces to Calibre Interactive and Calibre RVE. No setup required. | Jedat |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 1-2. Calibre Integrations (cont.)**

| Design Tool | Setup and Operation Information | Support by |
|---|---|---|
| **Schematic Viewers and Editors** | | |
| Siemens DFT Schematic Viewer in DFTInsight™ | No setup required. | Siemens Digital Industries Software |
| Cadence® Composer | Creating an Interface to Cadence Virtuoso and Cadence Composer. | Siemens Digital Industries Software |
| **Analysis Tools** | | |
| Ansys® Totem™ | See "Ansys" on the Calibre Interfaces web page. | Ansys |
| CWS SiPEX™ | See "CWS" on the Calibre Interfaces web page. | CWS |
| Helic® VeloceRF™ | See "Helic" on the Calibre Interfaces web page. | Helic |
| Integrand Software, Inc. | See "Integrand Software" on the Calibre Interfaces web page. | Integrand |
| Lorentz Solution, Inc. | See "Lorentz Solutions" on the Calibre Interfaces web page. | Lorentz Solution |
| NI AWR Analog Office® and Microwave Office® | See "NI (formerly AWR)" on the Calibre Interfaces web page. | NI |
| Magwel NV | See "Magwel NV" on the Calibre Interfaces web page. | Magwel NV |

# Syntax Conventions

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

You should enter literal text, that which is not in italics, exactly as shown.

**Table 1-3. Syntax Conventions**

| Convention | Description |
|---|---|
| **Bold** | Bold fonts indicate a required item. |
| *Italic* | Italic fonts indicate a user-supplied argument. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 1-3. Syntax Conventions (cont.)**

| Convention | Description |
|---|---|
| `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. |
| [ ] | 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:**<br>**mgc_calibre_add_menu_item -type menu -label “*label*”**<br>[-view_type {layout \| schematic}]<br>[-after “*label*” \| -before “*label*”] [-parent “*label*”] | |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 2
# Calibre Interactive Basics

Knowledge of some basic Calibre Interactive features such as runsets, controls files, preferences, and file naming templates is helpful before running Calibre Interactive.

**Files and Directories Used By Calibre Interactive** . . . . . 31
**About Calibre Interactive Runsets**. . . . . 33
**About Control Files** . . . . . 46
**About Rule Files**. . . . . 52
**Features of the User Interface** . . . . . 56

## Files and Directories Used By Calibre Interactive

Calibre Interactive uses several important files and directories.

### Files

- **Runset —** A Calibre Interactive generated file that stores the settings you specify in the GUI windows. Refer to "About Calibre Interactive Runsets" on page 33 for more information on this file.
- **Control File —** A system-generated file that controls the Calibre run. Refer to "About Control Files" on page 46 for more information on this file.
- **Rule File —** Your primary SVRF rule file for the Calibre run. Certain settings in this file can be overridden by Calibre Interactive, depending on the settings in the interface. Refer to "About Rule Files" on page 52 for more information on how Calibre Interactive uses rule files.
- **Preference Files —** System-generated files created by Calibre Interactive to save your preferences as defined in the Setup Preferences dialog box. Calibre Interactive creates a different preference file for each type of run. See "Setup Preferences" on page 142.
- **Customization File —** An optional Tcl file in which you program custom controls to create a Customization Settings Dialog Box. See "Customization Overview" on page 588.
- **Trigger Script —** A trigger script or function can be specified to execute at certain times in the Calibre Interactive execution process. For more information, see "Trigger Functions in Calibre Interactive" on page 657.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

The preceding list does not apply to Calibre Interactive xACT. See "Using Calibre Interactive xACT" on page 351 for information.

## Directories

- **Working Directory —** The directory from which you invoke Calibre Interactive.
- **Run Directory —** The directory location from which you run the Calibre application and from which all relative pathnames are resolved. By default, Calibre Interactive sets the run directory to your current directory. Changing the run directory may require that you update the paths to input and output files. The run directory is set from the Rules pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# About Calibre Interactive Runsets

A runset is a text file created by Calibre Interactive to store the settings you specify in the GUI. You can load a runset into Calibre Interactive to restore the GUI interface to the saved settings. You also use a runset to run Calibre Interactive in batch mode.

Runsets provide two important benefits:

- They reduce your work time. You no longer need to repeatedly enter all the information necessary to run Calibre.
- They help you manage your design environment. You can create a different runset for each design situation (process, cells, block-level verification, full-chip verification, and so on).

A Calibre Interactive runset includes all the information in the GUI that is not a default value, however, you can set an environment variable to save all GUI options to the runset. Choose **File > Save Runset** or **File > Save Runset As** to save a runset. Information saved in the runset includes:

- All GUI settings.
- Settings in the Setup Preferences dialog box (**Setup > Preferences**), except those in the "Startup Runset" and "Calibre Interactive Window Position" areas on the **Startup** tab.

**How Calibre Interactive Uses Runset Data** . . . . . 33
**Default Runsets in Calibre Interactive** . . . . . 34
**Calibre Interactive Runset Contents** . . . . . 35
**Creating and Saving a Calibre Interactive Runset** . . . . . 36
**Methods to Load, Save, and Use Calibre Interactive Runsets** . . . . . 37
**Loading Runsets on Invocation** . . . . . 39
**Using Multiple Calibre Interactive Runsets to Combine Different Sets of Options** . . . 44
**Save All GUI Options to the Runset File** . . . . . 44
**Runset Type and Runset Info** . . . . . 45

## How Calibre Interactive Uses Runset Data

Calibre Interactive treats runset data differently depending on whether you invoke the tool from a design tool or from the command line.

If you open the Calibre Interactive interface from the command line, loading a runset populates the GUI with the data in the runset. Unless you manually change these values, Calibre Interactive uses them when generating the control file for the run.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If you open Calibre Interactive from a design viewer, loading a runset populates the GUI with all the data in the runset, except the names of the system-generated input and output files. For these files, Calibre Interactive generates names based on the templates that are defined through Calibre Interactive preferences settings. Refer to "Templates for File Naming" on page 148 for more information on name-generation templates. Calibre Interactive combines the generated filenames with the other runset settings when generating the control file for the run.

Calibre Interactive has a certain set of default values. You can create a default runset if you want different default values. Default runsets are loaded only when no other runset is loaded.

### Related Topics

Default Runsets in Calibre Interactive

Loading Runsets on Invocation

Methods to Load, Save, and Use Calibre Interactive Runsets

## Default Runsets in Calibre Interactive

You can use a default runset to specify custom default settings for Calibre Interactive. Default runsets are used only when no other runset is loaded.

To create a default runset, create a runset with the settings you want, save it to one of the following filenames depending on your application, and place the file in your home directory:

**Table 2-1. Filenames for Default Runsets**

| Application | Filename |
|---|---|
| DRC | *.cgidrcdb.def* |
| DFM | *.cgidfmdb.def* |
| LVS | *.cgilvsdb.def* |
| PERC | *.cgipercdb.def* |
| PEX | *.cgipexdb.def* |

The default runset is read in the following cases:

- You invoke Calibre Interactive without loading a runset.
- You choose **File** > **New Runset** during a Calibre Interactive session.

Calibre Interactive searches your home directory for a default runset file in the above two cases. If a default runset is found those settings are used instead of the Calibre Interactive built-in default values.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

The default runset is not read when you load a runset. When you load a runset, the built-in Calibre Interactive defaults are applied first, then the settings in the runset being loaded are applied and overwrite the default settings.

### Related Topics

Save All GUI Options to the Runset File

Creating and Saving a Calibre Interactive Runset

Using Multiple Calibre Interactive Runsets to Combine Different Sets of Options

## Calibre Interactive Runset Contents

A Calibre Interactive runset is a text file that contains variable names and values corresponding to settings in the Calibre Interactive GUI. A runset is used to restore saved settings to the GUI and to control Calibre Interactive batch runs.

A runset includes the following settings:

- GUI settings.
- Settings in the Setup Preferences dialog box (**Setup > Preferences**), except those on the **Startup** tab in the “Startup Runset” and “Calibre Interactive Window Position” areas.

Here is a brief example:

```
*drcRulesFile: rules
*drcRunDir: .
*drcLayoutSystem: GDSII
*drcResultsFile: drc.results
*drcResultsFormat: ASCII
```

The basic form of each line is the following:

```
*<variable>: <value>
```

Each variable corresponds to an item you can enter or change in the GUI. The name of the variable indicates which GUI (DRC, LVS, PEX, DFM, PERC) is affected by the value, and which item in the GUI is being set to that value. Runset variables preceded by “cmn” are common to all applications.

If an option is not listed in the runset, the Calibre Interactive built-in default setting is used.

### Related Topics

Runset Options



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Creating and Saving a Calibre Interactive Runset

Save All GUI Options to the Runset File

# Creating and Saving a Calibre Interactive Runset

A Calibre Interactive runset is a text file that stores the Calibre Interactive GUI settings. You can load a runset in a later Calibre Interactive session to restore the GUI settings.

By default, Calibre Interactive only saves non-default settings to the runset. You can define an environment variable to have all settings saved to the runset.

**Note**

Runset files should be created using the Calibre Interactive application interface. Editing a runset file manually may lead to unexpected results.

## Prerequisites

- (Optional) To make Calibre Interactive save all GUI settings to a runset, set the environment variable MGC_CALIBRE_SAVE_ALL_RUNSET_VALUES to a non-zero value before starting Calibre Interactive.

  If you do not set this environment variable, only non-default settings are saved to the runset. This is usually more desirable than saving all GUI settings.

## Procedure

1. Start Calibre Interactive.
2. (Optional) If you have an existing runset that is close to your desired settings, load it with **File** > **Load Runset**.
3. Specify all GUI settings as desired.
4. (Optional) Choose **Setup** > **Preferences**, then the **Info** tab.

   Provide the Runset Type and Runset Info information. If a runset has been tagged with this information, it is displayed in the Load Runset File dialog box.
5. Choose **File** > **Save Runset** to save the runset.

   By default, Calibre Interactive saves runset files to the run directory.

   **Tip**

   You can choose **File** > **New Runset** to clear current GUI settings and return Calibre Interactive to default settings.

   If you launched Calibre Interactive from a design tool, then the input database, top cell, and other parameters related to the open design are filled in automatically using file naming templates.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Templates for File Naming

Runset Type and Runset Info

Invoking Calibre Interactive Tools

Methods to Load, Save, and Use Calibre Interactive Runsets

# Methods to Load, Save, and Use Calibre Interactive Runsets

You can load a Calibre Interactive runset from the command line, from the GUI, or by defining an environment variable. Preference settings allow you to save the runset automatically.

**Table 2-2. Loading, Saving, and Using Calibre Interactive Runsets**

| Action | Method and Description |
|---|---|
| Load a runset or multiple runsets from the GUI | **File > Load Runset**<br>To load multiple runsets, enter a space-separated list of runset names in the Runset File Path field. |
| Automatic runset loading, given in order of priority<br>(See "Loading Runsets on Invocation" on page 39) | |
| • Load a runset or multiple runsets from the command line<br>(Priority 1, highest) | -runset command line option:<br>-runset *my_runset*<br>-runset "*runset1 runset2*"<br>The -runset_options_display command line option can be used in batch operation. |
| • Populate the dropdown list of runsets with an environment variable<br>(Priority 2) | Define the environment variable MGC_CALIBRE_***_RUNSET_LIST to point to a file with the list of runsets. Replace *** with DRC, LVS, PERC, PEX, or DFM. Multiple files may be specified as a space-separated list. |
| • Load a runset or multiple runsets with an environment variable<br>(Priority 3) | Define the environment variable<br>MGC_CALIBRE_***_RUNSET_FILE with the runset(s); enclose multiple runsets in quotes. Replace *** with DRC, LVS, PERC, PEX, or DFM. |
| • Load a runset or multiple runsets with a startup preference setting<br>(Priority 4, lowest) | 1. Choose **Setup** > **Preferences**.<br>2. Choose the **Startup** tab.<br>3. Enable "Load runset files." To load multiple runsets, click the down arrow and enter one runset per line. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 2-2. Loading, Saving, and Using Calibre Interactive Runsets (cont.)**

| Action | Method and Description |
|---|---|
| Define a default runset | For DRC, name the default runset *.cgidrcdb.def*. Similar filenames are used for other applications. |
| Print runset options to the transcript in batch operation | Use the -runset_options_display command line option with the -batch switch. See the "calibre -gui" reference page for details. |
| Save a runset | **File > Save Runset**<br>Also see the option "Prompt to save changes before closing runset" on the **Misc** tab of the Setup Preferences dialog box. |
| Save the runset automatically | Enable one or both of these settings on the **Misc** tab of the Setup Preferences dialog box.<br>• "Save runset each time Calibre is run"<br>• "Save runset to current directory each time Calibre is run" (saves a copy of the runset to the directory from which Calibre Interactive was invoked) |
| Load the rule file when a runset is loaded | Enable "Automatically load rules file" on the **Misc** tab of the Setup Preferences dialog box. |

## Related Topics

Startup Preferences

calibre -gui

Default Runsets in Calibre Interactive

Loading Runsets on Invocation

Miscellaneous Preferences

Automatically Loading a Rule File

Using Multiple Calibre Interactive Runsets to Combine Different Sets of Options



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Loading Runsets on Invocation

There are several methods to load a Calibre Interactive runset automatically on invocation. You can use a command line option, an environment variable, or a startup preference setting.

**Table 2-3. Automatically a Loading Calibre Interactive Runset on Invocation**

| Priority | Method | Description |
|---|---|---|
| 1 | Load a runset or multiple runsets from the command line<br>(highest priority) | Use the -runset command line option:<br>-runset *my_runset*<br>-runset "*runset1 runset2*"<br>The -runset_options_display command-line option can be used in batch operation. |
| 2 | Populate the dropdown list of runsets with an environment variable | Define the environment variable MGC_CALIBRE_***_RUNSET_LIST to point to a file with the list of runsets. Replace *** with DRC, LVS, PERC, PEX, or DFM. Multiple files may be specified as a space-separated list. |
| 3 | Load a runset or multiple runsets with an environment variable | Define the environment variable<br>MGC_CALIBRE_***_RUNSET_FILE with the runset(s); enclose multiple runsets in quotes. Replace *** with DRC, LVS, PERC, PEX, or DFM. |
| 4 | Load a runset or multiple runsets with a startup preference setting<br>(lowest priority) | 1. Choose **Setup** > **Preferences**.<br>2. Choose the **Startup** tab.<br>3. Enable "Load runset files." To load multiple runsets, click the down arrow and enter one runset per line. |

The methods with priority 2, 3, and 4 are also described in the following sections:

**Populating the Load Runset File Dialog Box with Runsets on Invocation** . . . . . 40
**Using an Environment Variable to Load Calibre Interactive Runset(s) on Invocation** 41
**Loading Multiple Runsets with a Startup Preference** . . . . . 42



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Populating the Load Runset File Dialog Box with Runsets on Invocation

Define the environment variable MGC_CALIBRE_DRC_RUNSET_LIST to point to a file with the list of runsets to display in the Load Runset File dialog box. Replace DRC in the environment variable name with LVS, PERC, PEX, or DFM for those applications. The dialog box is displayed with the specified list of runsets when you invoke Calibre Interactive. Multiple files may be specified as a space-separated list.

**Note**

A runset specified on the command line takes precedence over a MGC_CALIBRE_***_RUNSET_LIST environment variable. See "Loading Runsets on Invocation" on page 39 for the complete order of precedence for automatic runset loading.

## Prerequisites

- You have a set of Calibre Interactive runsets for your application.

## Procedure

1. Create a text file with the path to each runset on separate lines. You can use absolute or relative pathnames.
2. Depending on your Calibre Interactive application, define one of the following environment variables with a text file containing the list of runsets:

| Application | Environment Variable |
|---|---|
| DRC | MGC_CALIBRE_DRC_RUNSET_LIST |
| LVS | MGC_CALIBRE_LVS_RUNSET_LIST |
| PERC | MGC_CALIBRE_PERC_RUNSET_LIST |
| PEX | MGC_CALIBRE_PEX_RUNSET_LIST |
| DFM | MGC_CALIBRE_DFM_RUNSET_LIST |

   If you have multiple files with runset lists, specify them as a space-separated list. The lists are combined when displayed in the Load Runset File dialog box.

3. Start Calibre Interactive.

## Results

Calibre Interactive displays the Load Runset File dialog box on invocation and displays the specified runsets in the "Recent Runsets" list.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Examples

Create the text file */user/calibre/my_runset_list* with the following contents:

```
/user/runsets/my_runset1
/user/runsets/my_runset2
/user/runsets/my_runset3
/user/runsets/my_runset4
```

Define the environment variable as follows for Calibre Interactive nmDRC.

**% setenv MGC_CALIBRE_DRC_RUNSET_LIST /user/calibre/my_runset_list**

If you have two runset list files:

**% setenv MGC_CALIBRE_DRC_RUNSET_LIST "runset_list1 runset_list2"**

### Related Topics

About Calibre Interactive Runsets

## Using an Environment Variable to Load Calibre Interactive Runset(s) on Invocation

Define the environment variable MGC_CALIBRE_DRC_RUNSET_FILE to point the runset(s) you want to load. Replace DRC in the environment variable name with LVS, PERC, PEX, or DFM for those applications. The runset(s) is loaded when you start Calibre Interactive.

**Note**

Runsets specified on the command line or with the MGC_CALIBRE_***_RUNSET_LIST environment variable both take precedence over a MGC_CALIBRE_***_RUNSET_FILE environment variable. See "Loading Runsets on Invocation" on page 39 for the complete order of precedence for automatic loading runsets in Calibre Interactive.

### Prerequisites

- You have a Calibre Interactive runset for your application. Multiple runsets may be specified.

### Procedure

1. Depending on your Calibre Interactive application, define one of the following environment variables with the pathname of the runset(s) to load:

| Application | Environment Variable |
|---|---|
| DRC | MGC_CALIBRE_DRC_RUNSET_FILE |
| LVS | MGC_CALIBRE_LVS_RUNSET_FILE |
| PERC | MGC_CALIBRE_PERC_RUNSET_FILE |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Application | Environment Variable |
|---|---|
| PEX | MGC_CALIBRE_PEX_RUNSET_FILE |
| DFM | MGC_CALIBRE_DFM_RUNSET_FILE |

Specify multiple runsets as a space separated list enclosed in double quotes.

2. Start Calibre Interactive.

### Results

The specified runset(s) are loaded when you invoke Calibre Interactive. If multiple runsets are used, they are loaded in order and the settings loaded last take precedence.

If you set one of these environment variables to “ ” (blank), no runset is opened.

### Examples

```
setenv MGC_CALIBRE_DRC_RUNSET_FILE "drc_runset"

setenv MGC_CALIBRE_PEX_RUNSET_FILE "pex_runset1 pex_runset2"
```

### Related Topics

About Calibre Interactive Runsets

Using Multiple Calibre Interactive Runsets to Combine Different Sets of Options

## Loading Multiple Runsets with a Startup Preference

You can load one or more runsets into Calibre Interactive using a startup preference. When multiple runsets are used, they are loaded in order and the settings loaded last take precedence.

**Tip**

You can also load multiple runsets from the command line or with the environment variable MGC_CALIBRE_***_RUNSET_FILE. See “Loading Runsets on Invocation” on page 39 for the order of precedence of these different methods. The startup preference has the lowest priority.

You can also load multiple runsets from the GUI. Select **File > Load Runset** and enter a space-separated list of runset names in the Runset File Path field.

### Prerequisites

- You have a Calibre Interactive runset for your application. Multiple runsets may be specified.

### Procedure

1. Open Calibre Interactive.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Choose **Setup** > **Preferences**.
3. Choose the **Startup** tab.
4. Enable “Load runset files.”
5. Specify a runset in the text field.
6. (Optional) If specifying multiple runsets, do the following:
   a. Click the down arrow to expand the file entry text box. Example 2-1 illustrates this step.
   b. Enter the runset filenames, one per line.
7. Exit Calibre Interactive.

## Results

The “Startup Runset” preference setting is automatically saved with your other startup preferences. The specified runset or set of runsets is loaded the next time you start Calibre Interactive. Multiple runsets are loaded in order and the settings loaded last take precedence.

## Examples

**Figure 2-1. Specifying Multiple Runset Files Using Setup Preferences**

## Related Topics

About Calibre Interactive Runsets

Startup Preferences

Using Multiple Calibre Interactive Runsets to Combine Different Sets of Options



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Using Multiple Calibre Interactive Runsets to Combine Different Sets of Options

When multiple runsets are loaded in Calibre Interactive, they are loaded in order and the settings loaded last take precedence. You can use multiple runsets to combine runsets that include different sets of options. For example, you can combine a basic runset with a second runset that contains settings for output options or host processor control.

Suppose you want to combine a basic DRC runset with a runset which only contains options for creating the DRC HTML report. You can create the following runset named *runset_HTML*:

```
*drcCreateDRCReport: 1
*drcDRCReportConfigFile: /user/drc_html.ini
*drcDRCReportConfigTemplate: Custom Template
```

To load *runset_HTML* with a runset named *runset_DRC*, issue the following on the command line:

**% calibre -gui -drc -runset "runset_DRC runset_HTML"**

If the runsets specify any of the same options, the settings in *runset_HTML* take precedence because it is loaded last.

For other methods to load multiple runsets, see "Loading Runsets on Invocation" on page 39. You can load multiple runsets with a startup preference, or with an environment variable.

You can also load multiple runsets from the GUI. Select **File** > **Load Runset** and enter a space-separated list of runset names in the Runset File Path field.

**Note**

If you load multiple runsets and the setting "Save runset to current directory each time Calibre is run" (**Setup** > **Preferences**, on the **Misc** tab) is enabled, the runset is saved as *runset.autosaved*.

### Related Topics

About Calibre Interactive Runsets

# Save All GUI Options to the Runset File

Calibre Interactive saves only non-default values to a runset file unless you set an environment variable to change this behavior. To save all GUI settings (both non-default and default settings) to the runset file, set the environment variable MGC_CALIBRE_SAVE_ALL_RUNSET_VALUES to a non-zero value.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For example:

```
% setenv MGC_CALIBRE_SAVE_ALL_RUNSET_VALUES 1
```

Save the runset with **File > Save Runset** or use one of the preference settings that instructs Calibre Interactive to automatically save the runset.

### Related Topics

Methods to Load, Save, and Use Calibre Interactive Runsets

Creating and Saving a Calibre Interactive Runset

About Calibre Interactive Runsets

## Runset Type and Runset Info

You can give a runset a label and descriptive comment using the Runset Type and Runset Info fields. These fields are located in the Setup Preferences dialog box (**Setup > Preferences**), on the **Info** tab.

If a runset has been saved with Runset Type and Runset Info, the information is displayed in the Load Runset File dialog box, as shown in Figure 2-2.

**Figure 2-2. Runset Type and Runset Info Fields**

### Related Topics

About Calibre Interactive Runsets



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# About Control Files

The control file in Calibre Interactive contains the settings in the GUI.

**Control File in Calibre Interactive** . . . . . 46
**Control Filenames** . . . . . 48
**Creating and Viewing the Control File Without Starting a Run** . . . . . 48
**Running Batch Calibre with a Calibre Interactive Control File** . . . . . 49

## Control File in Calibre Interactive

The Calibre Interactive control file provides all the information Calibre needs to process data, based on the settings you define in the user interface.

These settings include the following:

- The rule file pathname
- The input file pathnames
- The output file pathnames
- Special settings for controlling the run

The control file contains valid SVRF statements or TVF code if the main rule file is a TVF file. Environment variables are expanded in SVRF statements that specify a pathname. Environment variables for primary cell names are also expanded.

**Caution**

You should *not* modify control files generated by Calibre Interactive because this can cause unexpected conflicts with rule files that appear in Include statements. The statement DRC ICSTATION YES is internal to Calibre Interactive and should not be used in any rule file except the control file that Calibre Interactive internally generates.

The control file is a special type of rule file. Control files differ from standard rule files in the following ways:

- **Who creates them** — Calibre Interactive creates the control files based on the settings you define in the GUI; rule files are created separately, typically by the foundry or the design team.
- **File precedence** — Control files take precedence over rule files. Where there are possible conflicts, the control file settings are used for the run.

Figure 2-3 shows a Calibre Interactive control file and how the entries correspond to settings in the GUI. Also see “How Rule Files Relate to Control Files” on page 53 for more information.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 2-3. Relating User Input to Control Files**

**Tip**

You can instruct Calibre Interactive to create a control file that can be used in a command line Calibre run. The setting “Write control file suitable for command line run” is on the **Misc** tab in the Setup Preferences dialog box, as described in Table 3-20. See “Running Batch Calibre with a Calibre Interactive Control File” on page 49.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Control Filenames

Calibre Interactive writes the control file to the run directory with a name that is based upon the loaded rule file, with preceding and succeeding underscores (_).

```
_<ruleFileName>_
```

for example:

```
_testrun3A_
```

This control file is overwritten each time you run your job using the same rule file.

You can specify a different name for the control file with the setting “Specify control file name” on the **Misc** tab in the Setup Preferences dialog box (**Setup** > **Preference**).

## Related Topics

About Control Files

# Creating and Viewing the Control File Without Starting a Run

You can instruct Calibre Interactive to create a control file based on the current GUI settings. This is useful if you want to view the control file before starting the run.

## Procedure

1. Make all required GUI settings in Calibre Interactive.
2. Do one of the following:
   - Choose **File** > **Control File** > **View** from the main menu.
   - Hold down the Ctrl key and click the **Run** button on the left panel of Calibre Interactive.

## Results

Calibre Interactive generates the control file and opens it in a text viewer. In addition, the Transcript pane displays the Calibre command line.

## Related Topics

About Control Files

Control Filenames



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Running Batch Calibre with a Calibre Interactive Control File

Calibre Interactive can create a control file that you can use as the rule file for a command line Calibre run. The control file includes GUI settings from your Calibre Interactive setup, such as check selection, top cell name, and other settings.

The control file does not include the following settings from Calibre Interactive:

- Export from layout or schematic viewer

  Export from a design tool is only possible when using Calibre Interactive. You can run Calibre Interactive in batch mode and export the design from an open design tool. See "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104.

- Database read options for OpenAccess and LEF/DEF direct read

  If the layout database format is OPENACCESS or LEFDEF, then the database read options must be set with the MGC_CALIBRE_DB_READ_OPTIONS Environment Variable for Database Options; this is described in the following procedure.

- Environment variables defined in Calibre Interactive with the Set Environment Variables Dialog Box

  Environment variables defined within a Calibre Interactive runset are not included in the control file and must be defined in your shell environment prior to a command line Calibre run. This is described in the following procedure.

- Trigger settings

  Trigger functions defined within Calibre Interactive are not included in the control file; the trigger functions must be called separately. See "Trigger Functions in Calibre Interactive" on page 657 for more information on trigger functions.

## Procedure

1. Set up Calibre Interactive with the desired options or load a runset with the desired options.
2. Specify a control file for a command line run as follows:
   a. Choose **Setup** > **Preferences** to open the Setup Preferences dialog box.
   b. Click the **Misc** tab.
   c. Enable "Write control file suitable for command line run."
   d. (Optional) Enable "Specify control file name" and enter a filename in the text field.
   e. Click **OK** to close the Setup Preferences dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Determine if environment variables are defined within Calibre Interactive as follows:

   a. Choose **Setup** > **Set Environment** to open the Setup Environment Variables dialog box.

   b. View the **Set Env**. tab. A green check mark in the Runset column indicates that the environment variable is defined in the Calibre Interactive runset. Make a note of any environment variables defined in the runset and the corresponding runset value so the environment variable can be defined in the shell environment later.

   c. Click **Cancel** to close the Setup Environment Variables dialog box.

4. Determine if trigger functions are defined as follows:

   a. Choose **Setup** > **Preferences** to open the Setup Preferences dialog box.

   b. Click the **Triggers** tab. This tab displays the settings for Internal Triggers in Calibre Interactive.

   c. View the options on the **Triggers** tab and make a note of the settings for internal triggers.

   d. Review the section "External Trigger Functions and File Locations" on page 675, then determine if any external triggers are defined within your design tool environment. Make a note of any external trigger functions.

   You may want to call any external and internal trigger functions from a shell script. Consult the chapter "Trigger Functions in Calibre Interactive" on page 657 to determine the correct timing of the trigger functions in relation to the Calibre run.

5. Choose **File** > **Control File** > **View** to open a text viewer with the control file.

6. If your layout format is OPENACCESS or LEFDEF on the Inputs pane **Layout** tab, find lines similar to the following at the beginning of the control file:

   ```
   // MGC_CALIBRE_DB_READ_OPTIONS to:
   //
   //  -rundir ./data -abortOnEmptyPCell
   ```

   Make a note of the line beginning with "-rundir"—this is used later to set an environment variable with the database read options for the run. The database read options are determined from the settings on the OpenAccess/LEFDEF Options Tab on the Options pane (**Setup** > ***app*** **Options**, where *app* is one of DRC, LVS, DFM, PEX, or PERC).

7. Choose **File** > **Save As** to give the control file a new name or close the control file to keep the same filename.

   The name of the control file is *_rules_*, where *rules* is the rule file on the Rules pane, or the filename specified in Step 2.d. The control file is overwritten each time you run Calibre Interactive, so you may want to save this control file to a different filename.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

8. Click the **Transcript** button on the left panel of Calibre Interactive to view the Transcript pane. Make a note of the command line for the run, which is displayed in the Transcript pane.
9. Exit Calibre Interactive.
10. If runset environment variables were defined in Calibre Interactive (Step 3), define them in your shell environment.
11. If your layout format is OpenAccess or LEF/DEF, define the environment variable MGC_CALIBRE_DB_READ_OPTIONS as found in Step 6, using quotes around the option string. For example:

```
setenv MGC_CALIBRE_DB_READ_OPTIONS "-rundir ./data -abortOnEmptyPCell"
```

12. Execute any external pre-execution trigger functions found in Step 4, then internal pre-execution trigger functions.
13. Execute Calibre with the options used on the command line viewed on the Transcript pane (Step 8) and use the control file saved in Step 7 as the rule file. If you renamed the control file in Step 7, then the command line on the Transcript pane will not have the correct filename.
14. Execute any internal post-execution trigger functions found in Step 4, then external post-execution trigger functions.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# About Rule Files

Rule files define the processing Calibre performs during a run. You specify the rule file for a Calibre Interactive run on the Rules pane.

Calibre Interactive uses rule files specified in the following ways:

- **The primary rule file defined on the Rules pane** — This serves as the rule file on which the Calibre run is based.
- **Rule files included through the primary rule file** — These are included in any Calibre run regardless of whether Calibre is invoked from the command line or Calibre Interactive. Loading the primary rule file in Calibre Interactive causes included rule files to be loaded as well. (For more information, refer to the description of the Include command in the *Standard Verification Rule Format Manual*).
- **Rule files included through the GUI** — These are defined on the **Include** tab on the Options pane. These rule files act as included files called by the GUI. They allow you to specify included rule files outside of the primary rule file. They are listed in the control file within include statements.

**Rule File Operations in Calibre Interactive** . . . . . 52
**How Rule Files Relate to Control Files** . . . . . 53

## Rule File Operations in Calibre Interactive

When working with rule files in Calibre Interactive you can view and edit the rule file, check the rule file syntax, select the rule checks to execute, and load rule file settings into the Calibre Interactive GUI.

The following actions are available:

- **View and edit rule files** — You can view and edit any rule file using the Calibre Interactive text editor, as described in "Viewing and Editing Text Files in Calibre Interactive" on page 125. For the primary rule file, defined through the Rules pane, you can click the **View** button to view the rule file. When you click the **View** button, the rule file displays in a separate window.

  You can also view the Layer Derivations tree near the bottom of the Rules pane. Click the ⊞ icon to open the tree and continue to expand the tree as needed. See "Layer Derivations Tree" on page 126 for more information.
- **Specify additional rule files and rule file statements** — See "Including Other Rule Files and SVRF Commands in Calibre Interactive" on page 95.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Check rule file syntax**— Calibre Interactive causes your rule file to be compiled and reports any errors. Rule file compilation occurs as a preliminary step in each of the following tasks:
  - Loading a rule file (**Load** button on the Rules pane).
  - Choosing **Setup** > **Set Environment**.
  - Opening the Check Selection Recipe Editor Dialog Box (Calibre Interactive nmDRC and DFM) or the Select Checks dialog box for ERC checks (Calibre Interactive nmLVS).
  - Choosing **Setup** > **PEX Options** if the **PEX Options** button is not already displayed.
  - Selecting the **Probes** tab on the Inputs pane in Calibre Interactive PEX.
- **Choose which checks to run** — The Check Selection Recipe specifies the checks that are executed during a Calibre Interactive nmDRC or Calibre Interactive DFM run. You can use built-in check recipes or define your own custom check recipe. See "Check Selection in Calibre Interactive nmDRC and Calibre Interactive DFM" on page 109 for details.

  The Select Checks dialog box is used to select the ERC checks executed for a Calibre Interactive nmLVS run. See "Selecting ERC Checks" on page 284 for details.
- **Load values from the rule file into the GUI** — The **Load** button on the Rules pane causes Calibre Interactive to populate the GUI with information from the rule file.

  Two preference settings affect how rule file values are loaded:
  - **Automatically load rules file** — In the Setup Preferences dialog box on the **Misc** tab. This setting instructs Calibre Interactive to automatically load a rule file, as described in "Automatically Loading a Rule File" on page 76.
  - **Read Inputs/Outputs fields when loading rules file** —In the Setup Preferences dialog box on the **Misc** tab. When this setting is enabled, Calibre Interactive reads information for the Inputs and Outputs panes from the rule file. This setting is enabled by default.

### Related Topics

How Rule Files Relate to Control Files

Key Concepts [Standard Verification Rule Format (SVRF) Manual]

# How Rule Files Relate to Control Files

Calibre uses the control file to define the run. The primary rule file that you define in the Rules pane is included in the Calibre run as an Include file in the Calibre Interactive control file. When Calibre compiles a control file, the settings in the control file take precedence over any



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

corresponding settings in the rule file. In other words, the GUI settings at the time the job is initiated control how the job is run.

Because the rule file is included as an Include file in the control file, current settings in the rule file are always used each time Calibre Interactive starts a run. However, if a GUI setting differs from a rule file setting, the GUI setting takes precedence.

Also see the preference setting “Automatically load rules file” in Table 3-20 on page 156.

Figure 2-4 shows a control file and a rule file for a simple DRC run.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 2-4. Relating Control Files to Rule Files**

## Related Topics

Rule File Operations in Calibre Interactive

Including Other Rule Files and SVRF Commands in Calibre Interactive



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Features of the User Interface

Features such as color coding and setting environment variables simplify the use of Calibre Interactive.

The section "Populating the Calibre Interactive GUI" on page 75 includes information on the different ways to populate the GUI fields.

**Calibre Interactive Buttons** . . . . . 56
**Setup Menu in Calibre Interactive** . . . . . 57
**Environment Variables in Calibre Interactive GUI Fields** . . . . . 58
**Color Coding in Calibre Interactive** . . . . . 59
**File Browser Options** . . . . . 60

## Calibre Interactive Buttons

The left side of the Calibre Interactive window contains buttons that either select the GUI pane that is displayed or initiate an action.

Table 2-4 describes the buttons on the left panel of Calibre Interactive. Some buttons are optional and must be enabled as described in the table.

**Table 2-4. Buttons in Calibre Interactive**

| Button | Description |
|---|---|
| Rules | Displays the Rules pane for entering the rule file and run directory. You can also view the Layer Derivations Tree. |
| Inputs | Displays the Inputs pane for specifying input files and related options. |
| Outputs | Displays the Outputs pane for specifying the output database and related options. |
| Options | (Optional) Displays the Options pane for the application. Enable the Options button by choosing **Setup** > ***app*** **Options**, where *app* is one of DRC, LVS, DFM, PEX, or PERC. |
| Run Control | Displays the Run Control pane for specifying the host computer and other job control options. See "Run Control Setup and Licensing" on page 681. |
| Transcript | Displays the Transcript Pane in Calibre Interactive. |
| Customization | (Optional) Opens the Customization Settings Dialog Box, which is a user-coded GUI for specifying certain GUI controls and run parameters. Enable the Customization button by choosing **Setup** > **Preferences** and specifying a customization file on the **Startup** tab. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 2-4. Buttons in Calibre Interactive (cont.)**

| Button | Description |
|---|---|
| PEX steps | (Optional) These buttons allow you to individually select the Create PHDB, Create PDB, and Create Netlist stages of a PEX run. Enable the PEX steps buttons with "Show PEX steps in task button bar" on the **Advanced** tab of the Run Control pane. |
| Run | Starts the run. |
| Start RVE | Starts Calibre RVE and opens the results database specified on the Outputs pane. |

Buttons for setup panes are green if all information is available and red if information is missing. Action buttons, such as Start RVE, are always black.

### Related Topics

Color Coding in Calibre Interactive

Manually Invoking Calibre RVE from Calibre Interactive [Calibre RVE User's Manual]

Getting Started with Calibre RVE [Calibre RVE User's Manual]

## Setup Menu in Calibre Interactive

The contents of the **Setup** menu in Calibre Interactive depend on the application you are running.

The following table describes the contents of the **Setup** menu in Calibre Interactive.

**Table 2-5. Setup Menu Contents in Calibre Interactive**

| Menu Selection (application) | Description |
|---|---|
| Options<br>(all applications) | Displays the Options pane for the application and adds the Options button to the left panel of Calibre Interactive. |
| Set Environment<br>(all applications) | Opens the Setup Environment Variables dialog box, which allows you to view, edit, and define the environment variables for the run. See "Viewing and Editing Environment Variables" on page 130. |
| Verilog Translator<br>(nmLVS, PERC, PEX) | Opens the Setup Verilog Translator dialog box. See "Setting Up the Verilog Translator (v2lvs)" on page 254. |
| Delay Calculation<br>(PEX) | Opens the Setup Delay Calculation dialog box. See "Setting Delay Calculation" on page 345 for more information. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 2-5. Setup Menu Contents in Calibre Interactive (cont.)**

| Menu Selection (application) | Description |
|---|---|
| Select Checks<br>(3DSTACK only) | Opens the Select Checks dialog box to select the checks that are executed. See "Selecting Specific Checks" in the *Calibre 3DSTACK User's Manual*. |
| Layout Viewer<br>(all applications) | Opens the Setup Layout Viewer dialog box.<br>Communication with a layout viewer is automatically set up when you launch Calibre Interactive from the layout viewer and no additional setup is necessary in most cases. See "Communication with the Design Tool" on page 72 and "Setting the Calibre Interactive Socket Port for a Layout Viewer" on page 707. |
| Schematic Viewer<br>(nmLVS, PERC, PEX) | Opens the Setup Schematic Viewer dialog box.<br>Communication with a schematic viewer is automatically set up when you launch Calibre Interactive from the schematic viewer and no additional setup is necessary in most cases. See "Communication with the Design Tool" on page 72 and "Setting the Calibre Interactive Socket Port for a Schematic Viewer" on page 708. |
| Preferences<br>(all applications) | Opens the Setup Preferences dialog box. See "Setup Preferences" on page 142. |
| Show Tooltips<br>(all applications) | When enabled, causes Calibre Interactive to display pop-up tooltips when you hover the mouse over a button or field in the GUI. |

# Environment Variables in Calibre Interactive GUI Fields

You can use environment variables in Calibre Interactive fields and runsets with some restrictions.

The following restrictions apply:

- Environment variables are supported for the following fields and associated runset variables:
  - **Pathname fields** — Any field that gives a pathname or filename. For example, the layout input file and output database (drcLayoutPaths and drcResultsFile for a DRC runset).
  - **Top cell name for layout and source** — For example, drcLayoutPrimary in a DRC runset.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Environment variables are not allowed for any other fields.

- Environment variable names are delimited by the following characters:

  `/  "  #  .  $`

  end of string (white space)

  surrounding pair of braces: ${ENV_VAR}

  The tool recognizes the end of a variable name when it comes to one of the specified delimiter characters or the end of the string. For example, if *$DESIGN1/$TOP.gds* is entered into the GUI field for the layout file, Calibre Interactive interprets the "/" and "." characters as delimiters and expects to find environment variables named "DESIGN1" and "TOP".

  Environment variables may also be surrounded by braces, as in ${PROC}. This provides a way to join an environment variable with text without using one of the recognized delimiter characters.

### Example: Environment Variable Usage in Calibre Interactive GUI

For example, suppose the following environment variables are defined:

```
setenv PROC 90nm
setenv TOP a_top
setenv DESIGN mem_b
```

The following examples illustrate correct usage of these environment variables:

| | |
|---|---|
| ${PROC}_lvs.rul | resolves as 90nm_lvs.rul |
| $DESIGN/$TOP.gds | resolves as mem_b/a_top.gds |

The following examples do not resolve correctly:

| | |
|---|---|
| $PROC_lvs.rul | not resolved, variable PROC_lvs is not defined |
| $DESIGNA/$TOP.gds | not resolved, variable DESIGNA is not defined |

### Related Topics

Viewing and Editing Environment Variables

About Calibre Interactive Runsets

Runset Options

## Color Coding in Calibre Interactive

The user interface uses color coding to indicate status.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Status is indicated as follows:

- Color coding in data fields for file or directory names indicates whether the file exists or not.
  - **Red text** — Input file that does not exist. You must provide the information.
  - **Green text** — File exists. If a file of the same name is to be generated by Calibre Interactive, the tool will overwrite the existing file. Files loaded in a viewer connected to Calibre Interactive are not overwritten.
  - **Black text** — Output file that does not exist. The file will be created by Calibre Interactive.
- Color coding in the button pane indicates readiness for running the tool.
  - **Green Setup buttons** — Indicate that you have provided enough information of the type associated with this button, and it appears to be the right type of information.
  - **Red Setup buttons** — Indicate that you have not provided enough information.
  - **Black Action buttons** — All action buttons are black, regardless of readiness for running Calibre.

# File Browser Options

Most file entry fields in Calibre Interactive allow you to select the file from a file browser.

File browser dialog boxes that specify the run directory in Calibre Interactive or an output file or directory include a button to create a new directory. A new directory is automatically created if the pathname of an output file includes a directory that does not exist.

An example of the file browser dialog box is as shown in the following figure for Calibre Interactive nmLVS.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 2-5. File Browser**

Advanced options are available by clicking the browser option icon. These options are described in the following table.

**Table 2-6. Advanced File Browser Options**

| Menu Item | Description |
|---|---|
| Detailed View | Show file modification date and size. |
| Show Hidden Files | As described. |
| Show Links | Show symbolic links. |
| Autocheck Entries | Check if the path is valid. |
| Multi Directory Select | Allows selection of files from multiple directories.<br>This selection is only available if the field allows multiple files to be specified. |
| Update Directory | Refresh the view. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 3
# Running Calibre Interactive

Some options and setup procedures pertain to all of the Calibre Interactive applications.

**Invoking Calibre Interactive Tools** . . . . . 64
**Communication with the Design Tool** . . . . . 72
**Methods for Populating the Calibre Interactive GUI** . . . . . 75
**Third Party Database Options** . . . . . 80
**Providing LEF/DEF Input** . . . . . 88
**Providing OpenAccess Input** . . . . . 90
**Setting Database and General Options in Calibre Interactive** . . . . . 93
**Running a Job Using Calibre Interactive** . . . . . 98
**Running Calibre Interactive in Batch Mode** . . . . . 102
**Calibre Interactive Operation with a TVF Rule File** . . . . . 105
**Check Selection in Calibre Interactive** . . . . . 108
**Viewing and Editing Text Files in Calibre Interactive** . . . . . 125
**Layer Derivations Tree** . . . . . 126
**Viewing and Editing Environment Variables** . . . . . 130
**Setup Preferences** . . . . . 142



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Invoking Calibre Interactive Tools

You can invoke Calibre Interactive from a supported design tool or from the command line.

Invocation methods:

- From a design tool with a Calibre interface.

  This method automatically sets up communication between the design tool and Calibre Interactive.

- From a custom menu item in a supported design tool.

  See "Custom Menu Items in Design Tools" on page 709.

- From the command line.

  When you invoke Calibre Interactive from the command line, you have the choice of running interactively (with the GUI) or in batch mode (without the GUI).

This section includes the following topics related to invoking Calibre Interactive:

**Invoking Calibre Interactive From a Layout or Schematic Viewer** . . . . . 64
**Invoking Calibre Interactive from the Command Line** . . . . . 66
**License Timeout in Calibre Interactive** . . . . . 67
**calibre -gui** . . . . . 68

## Invoking Calibre Interactive From a Layout or Schematic Viewer

When you start Calibre Interactive from a layout or schematic viewer, communication is automatically set up between Calibre Interactive and the viewer. You can specify that Calibre Interactive use the open layout or schematic as input for the Calibre run.

### Prerequisites

- A supported viewer with the interface to Calibre installed. See "Integration to Design Tools" on page 26 for a list of design tools with a Calibre interface, and "Interfacing Calibre Interactive to Layout and Schematic Viewers" on page 705 for instructions on setting up the interface to Calibre.
- Calibre Interactive is installed. See "Calibre Interactive System Requirements" on page 26.

### Procedure

1. Open your layout or schematic in a supported viewer.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Select the desired Calibre Interactive application. The menu selection for Calibre Interactive depends on the design tool.

| Design Tool | Menu Selection |
|---|---|
| Calibre DESIGNrev family of layout viewers | Choose **Verification**, then choose the run type as shown in Figure 3-1. |
| Cadence Virtuoso and other Cadence tools | Choose **Calibre**, then the run type |
| Synopsys tools | Choose **Calibre,** then the run type. |
| Other tools | Choose **Calibre**, then the run type. |

## Results

Figure 3-1 shows the **Verification** menu for Calibre DESIGNrev; the **Calibre** menu for other viewers will have similar entries and may have a submenu for the application.

When Calibre Interactive opens, certain fields are filled in based on the cell and view open in the design tool; see "Templates for File Naming" on page 148 for more information.

**Figure 3-1. Selecting the Calibre Interactive Application in Calibre DESIGNrev**

## Related Topics

Using a Viewer as the Source for Design Input



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Invoking Calibre Interactive from the Command Line

Command line invocation of Calibre Interactive can open the GUI or start a Calibre Interactive batch run.

## Prerequisites

- Calibre Interactive is installed. See "Calibre Interactive System Requirements" on page 26.
- Required input files are available. See "Files and Directories Used By Calibre Interactive" on page 31.

## Procedure

1. (Optional) Navigate to the run directory on your Calibre host computer.
2. Open Calibre Interactive with one of the following methods. Except for the first example, the examples are for Calibre Interactive nmDRC—substitute the appropriate command line argument of -lvs, -dfm, -perc, -pex, -xact, or -3dstack to open a different application.

**Table 3-1. Invoking Calibre Interactive from the Command Line**

| If you want to … | Do this: |
|---|---|
| Display the Calibre Interactive palette. | `calibre -gui`<br> <br><br>Choose the application you want to run. |
| Display the Calibre Interactive nmDRC window. | `calibre -gui -drc` |
| Open the Calibre Interactive nmDRC window with a runset automatically loaded. | `calibre -gui -drc -runset my_runset_file` |
| Run Calibre Interactive nmDRC in batch mode. | `calibre -gui -drc -runset drc_runset -batch` |

## Related Topics

calibre -gui

Running Calibre Interactive in Batch Mode

Invoking Calibre Interactive From a Layout or Schematic Viewer

About Calibre Interactive Runsets



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# License Timeout in Calibre Interactive

There are several options for setting up the timeout of your Calibre Interactive license.

The application looks at timeout settings in the following order of priority to determine when to release the license:

1. The server-side FLEXnet TIMEOUT option serves as top priority for when to release your Calibre Interactive application license. Refer to the *FLEXnet License Administration Guide* for information on setting the TIMEOUT option. See the section "Mentor Standard Licensing (MSL)" in the *Calibre Administrator's Guide* for information on where to find the FLEXnet licensing documentation.

2. The environment variable, MGC_CALGUI_RELEASE_LICENSE_TIME follows the FLEXnet setting in priority for setting license timeout.

   The default time unit is hours. You can specify the unit of time as hours, minutes, or seconds by appending h, m, or s to the time; a space between the time and the unit indicator is optional. The unit indicator may be a word—only the first character is used, therefore "sec", "seconds", and "s" all indicate seconds as the time unit. The 'h' to indicate hours is optional, as the default time unit is hours.

   The following examples all set a license timeout of 18 minutes:

   **setenv MGC_CALGUI_RELEASE_LICENSE_TIME 0.3**

   **setenv MGC_CALGUI_RELEASE_LICENSE_TIME 18m**

   **setenv MGC_CALGUI_RELEASE_LICENSE_TIME "18 min"**

   The license timeout for the Calibre PERC Rule Generator GUI (**Setup > Generate PERC Rules**) is set only by MGC_CALGUI_RELEASE_LICENSE_TIME.

3. The environment variable, MGC_RVE_RELEASE_LICENSE_TIME follows in priority for setting license timeout. The value is set as described for MGC_CALGUI_RELEASE_LICENSE_TIME.

4. The "Release license after: X inactive hours" option in the **Disable** tab of the Setup Preferences dialog box has the lowest priority. This choice stores your preference in the runset file and governs when your license is released if none of the other license timeout settings are specified.

   If the time-out period is set with a higher precedence setting, that value is displayed in the GUI and can not be changed.

If a license timeout is not set with one of the above methods, then there is no license timeout. License timeout values are set in hours; the timeout value must be greater than five minutes. The default timeout period is 500 hours.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# calibre -gui

Command line invocation for Calibre Interactive.

**Note**

For Calibre Interactive xACT, see "Invoking the Calibre Interactive xACT GUI" on page 352.

## Usage

### Interactive operation:

**calibre -gui** [ {-drc | -dfm | -lvs | -pex | -perc | -3dstack}
[-nowait | -wait *n*]
[*runset* | -runset *runset*]
[-custom *filename*] ]

### Batch operation:

**calibre -gui** {**-drc** | **-dfm** | **-lvs** | **-pex** | **-perc** | **-3dstack**}
[-nowait | -wait *n*]
{***runset*** | **-runset *runset*** |**-runset_options_display *runset***}
[-customgui *filename* | -custom *filename*]
**-batch**

### Calibre PERC Rule File Generator GUI:

**calibre -gui -perc_rule_gen** [-prgfile *rulegen_setup.prg*]

## Arguments

- **-gui**

  A required switch to open Calibre Interactive. When specified with -drc, -dfm, -lvs, -perc, or -pex, the respective user interface window opens. When specified alone, the Calibre Interactive Palette opens, as shown in "Invoking Calibre Interactive from the Command Line" on page 66. You can open the various application-specific GUIs from the Calibre Interactive Palette.

- -drc | -dfm| -lvs | -pex | -perc | -3dstack

  An optional switch used with -gui to specify the Calibre Interactive application. The interfaces are Calibre nmDRC, DFM, Calibre nmLVS, PEX, Calibre PERC, and Calibre 3DSTACK, respectively.

- -nowait

  An optional switch which specifies that Calibre Interactive does not wait for a license if one is not available immediately.

- -wait *n*

  An optional switch and argument which specifies the amount of time in minutes that Calibre Interactive waits for a license.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- *runset* | -runset *runset*

  A switch and argument specifying the pathname of a runset. When specified, the runset is automatically loaded when the user interface opens. A runset is required for batch operation (-batch); it is optional for interactive operation.

  You can load multiple runsets by specifying a space separated list of runsets enclosed in quotes; the runsets are loaded in the order specified and the settings loaded last take precedence.

  Also see "About Calibre Interactive Runsets" on page 33 and "Using Multiple Calibre Interactive Runsets to Combine Different Sets of Options" on page 44.

- -runset_options_display *runset*

  A switch and argument that specifies the pathname of a runset and also instructs the tool to display the runset options to the transcript. This option is only valid for batch operation (the -batch switch).

  You can specify -runset_options_display more than once to load multiple runsets, and you can combine the -runset_options_display and -runset switches. Runsets are loaded in the order given in the command line; options in runsets occurring later in the command line override options in runsets occurring earlier on the command line. Only options in runsets specified by -runset_options_display are echoed to the transcript. If an option in a -runset_options_display runset is overridden by a later runset specified with the -runset option, then the option is not echoed to the transcript.

- -custom *filename*

  An optional switch and argument specifying the pathname of a customization file.

  In interactive mode, the file is read when Calibre Interactive starts and the **Customization** button is added to the left panel of Calibre Interactive.

  In Calibre Interactive batch mode, the initial settings from customization file are applied to the batch run; the Customization Settings window is not opened. If the runset includes customization override settings, those take precedence over the initial settings in the customization file. See "Customization Files" on page 587 and "Customization GUI Operation in Batch Mode" on page 602.

  This switch overrides the MGC_CALIBRE_CUSTOMIZATION_FILE environment variable. This switch overrides a customization file specified in the Setup Preferences dialog box on the **Startup** tab and saved to the runset.

- -customgui *filename*

  An optional switch and argument specifying the pathname of a customization file. This option is only available in batch mode. The Customization Settings window appears at the beginning of the Calibre run. Once the customization selections are made and the window is closed, the Calibre run continues in batch mode, without the user interface window. For more information, see "Customization Files" on page 587 and "Customization GUI Operation in Batch Mode" on page 602.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

This switch overrides the MGC_CALIBRE_CUSTOMIZATION_FILE environment variable, if set.

- -batch

  An optional switch used to run Calibre Interactive in batch mode without opening the user interface. A runset must be specified for batch operation.

  Calibre Interactive starts the Calibre job using the runset and the optional customization file (-customgui or -custom switch). Layout and netlist databases need to be specified in the rule file that appears in the runset. See "Running Calibre Interactive in Batch Mode" on page 102 for some examples.

  In some cases Calibre Interactive must compile the rule file separately from the rule file compilation done for the Calibre run. Calibre Interactive in batch mode compiles the rule file in these cases:

  - The command line includes a check-related option.
  - You have specified a customization file.

  Most runset parameters can be specified in -batch mode as a command line argument. For example:

  ```
  calibre -gui -drc -runset rset -batch -drcLayoutPaths CHIP.gds
  -drcLayoutPrimary TOP
  ```

  The -drcLayoutPaths and -drcLayoutPrimary arguments are runset parameters. See "Calibre Interactive Runset Contents" on page 35 for an explanation of these parameters.

  **Note**

  The name given for the runset parameter in the command line argument must match the actual runset parameter name exactly. You can examine a runset file for any arguments you want to use in batch mode.

- **-gui -perc_rule_gen** [-prgfile *rulegen_setup.prg*]

  Argument set used to invoke the standalone Calibre PERC Rule Generator GUI.

  - **-gui -perc_rule_gen** — Required argument that specifies to open the Calibre PERC Rule Generator GUI.
  - -prgfile *rulegen_setup.prg* — An optional switch and argument specifying the path to a binary Calibre PERC Rule Generator setup file (*.prg* file). The setup file is used to populate settings in the GUI.

  See "Running the Rule File Generator GUI" in the *Calibre PERC User's Manual* for more information.

  You can also invoke the Calibre PERC Rule Generator GUI from Calibre Interactive PERC; see "Creating a Calibre PERC Rule File with High Level Checks" on page 576.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Related Topics

Syntax Conventions

Invoking Calibre Interactive Tools



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Communication with the Design Tool

When your Calibre Interactive session is connected to a supported layout or schematic viewer, you can use the open design as input for your Calibre run; this is referred to as layout and schematic export. Socket ports are used for communication between Calibre Interactive and the design tool.

Calibre Interactive can automatically set up communication with a design tool or you can manually make the connection.

**Automatic Connection** — If you invoke Calibre Interactive from a layout or schematic viewer, communication between the two applications is set up automatically. This invocation method is explained in "Invoking Calibre Interactive From a Layout or Schematic Viewer" on page 64.

**Manual Connection** — If you open a design tool after invoking Calibre Interactive, you can set up communication between the two tools through IPC sockets. See "Setting the Socket Port in Calibre Interactive" on page 707.

**Using a Viewer as the Source for Design Input** . . . . . 72
**Setting Layout and Schematic Export in Calibre Interactive** . . . . . 73

## Using a Viewer as the Source for Design Input

The "Export from layout" and "Export from schematic" settings on the Inputs pane of Calibre Interactive are used to specify the open layout or schematic in an connected viewer to be the input for your Calibre run. Layout and schematic export is possible for both interactive and batch runs. In batch mode, you must also disable the setting "Ignore layout and source export in batch mode" on the **Misc** tab in the Setup Preferences dialog box (**Setup** > **Preferences**) when doing layout or schematic export.

You can also define a preference to have the export setting depend on whether you started Calibre Interactive from a layout or schematic viewer. See "Export from Viewer Preferences" on page 152.

- For setting up design export, see "Setting Layout and Schematic Export in Calibre Interactive" on page 73
- For communication with the design tool, see "Setting the Socket Port in Calibre Interactive" on page 707
- For batch operation, see "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setting Layout and Schematic Export in Calibre Interactive

You can use the open design in a layout or schematic viewer as input for your Calibre run; this is referred to as layout and schematic export. Layout and schematic export can also be used in a batch mode Calibre Interactive run with an additional setting.

## Prerequisites

- The design is open in a supported layout or schematic viewer.
- Calibre Interactive is open to the desired application.

## Procedure

1. If exporting a layout from a connected layout viewer:

   a. Enable “Export from layout viewer” on the Inputs pane, **Layout** tab of Calibre Interactive.

   Layout data comes from the active cell in the layout editor.

   b. Verify the filename given to the extracted layout database in the File field.

   When “Export from layout viewer” is selected, the extracted layout database is saved to the filename in the File field. Calibre Interactive generates the filename based on the file naming template defined on the **Templates** > **Inputs** sub-tab in the Setup Preferences dialog box (**Setup** > **Preferences**). By default, the extracted layout filename is based upon the layout cell name. You can enter a different filename if you want. See “Templates for File Naming” on page 148.

   If the filename in the File field is red, the file does not exist and Calibre Interactive will create it.

   If the filename in the File field is green, the file exists and Calibre Interactive will overwrite the file. Enter another filename if you want to keep the existing file.

   **Caution**

   When “Export from layout viewer” is enabled, a layout file is extracted and saved. Do not use the extracted layout file named in the File field for tapeout or any other use.

2. If exporting a source schematic from a connected schematic viewer (for Calibre nmLVS, PEX, and Calibre PERC):

   a. Enable “Export from schematic viewer” on the Inputs pane, **Netlist** tab.

   Source netlist data comes from the active cell in the schematic editor.

   b. Verify the filename given to the extracted source netlist in the Files: field.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

When "Export from schematic viewer" is selected, the extracted netlist is saved to the filename in the Files: field. Calibre Interactive generates the filename based on the file naming template defined on the **Templates** > **Inputs** sub-tab in the Setup Preferences dialog box (**Setup** > **Preferences**). By default, the extracted netlist filename is based upon the schematic cell name. See "Templates for File Naming" on page 148.

If the filename in the File: field is red, the file does not exist and Calibre Interactive will create it.

If the filename in the File: field is green, the file exists and Calibre Interactive will overwrite the file. Enter another filename if you want to keep the existing file.

c. Verify the name of the top cell in the "Top Cell" field.

d. (Optional) If the runset will be used for a batch mode Calibre Interactive run, do the following:

Choose **Setup** > **Preferences**, select the **Misc** tab, and disable "Ignore layout and source export in batch mode."

3. Make remaining Calibre Interactive settings for the run.

## Results

When you click the **Run** button, Calibre Interactive instructs the design tool to export the design.

If the file for the extracted database already exists, Calibre Interactive displays the Overwrite File dialog box with the following options:

**Table 3-2. Overwrite File Dialog Box for Layout and Schematic Export**

| Button | Action |
|---|---|
| Overwrite | Overwrite the existing file. |
| Don't Overwrite | Keep the existing file and continue with the run. The Calibre run uses the existing extracted database. Use this selection if your design has not changed since the last export operation. |
| Cancel | Cancel the run. |

## Related Topics

Templates for File Naming

Communication with the Design Tool

Specifying Layout and Schematic Export for a Calibre Interactive Batch Run



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Methods for Populating the Calibre Interactive GUI

There are several methods for populating the fields in the Calibre Interactive GUI.

**Populating the Calibre Interactive GUI** . . . . . 75
**Automatically Loading a Rule File**. . . . . 76
**Managing Rule File and Calibre Interactive GUI Precedence** . . . . . 77

## Populating the Calibre Interactive GUI

The settings in the Calibre Interactive GUI can come from several sources, including direct user input, a Calibre Interactive runset, template settings, and the Customization GUI.

Calibre Interactive settings come from the following sources:

- **Default values** — Default values are programmed into Calibre Interactive. You see the default values when you open Calibre Interactive without a runset and not connected to a design tool. The default settings vary between applications.
- **Runset** — A Calibre Interactive runset is a text file that stores the Calibre Interactive GUI settings. You can load a runset in a later Calibre Interactive session to restore the GUI settings from a previous session. See "About Calibre Interactive Runsets" on page 33 for methods to save and load a runset.
- **Default runset** — A default runset allows you to specify your own default values for Calibre Interactive. The default runset is read when Calibre Interactive is invoked without a runset, as described in "Default Runsets in Calibre Interactive" on page 34.
- **System-generated parameters based on template settings** — When Calibre Interactive is invoked from a layout or schematic viewer, the default behavior is to overwrite certain GUI settings with values derived from the open database. For example, the output results database in Calibre Interactive nmDRC is automatically named *<top_cell>.drc.results*, where *<top_cell>* is the name of the primary layout cell in the open design. See "Templates for File Naming" on page 148.
- **Rule file** — Settings from the rule file populate the GUI when you click the **Load** button on the Rules pane. You can also instruct Calibre Interactive to load the rule file when a runset is loaded, as described in "Automatically Loading a Rule File."
- **Customization file** — The Customization GUI can set values in the Calibre Interactive GUI using runset control statements in the customization file, as described in the "Customization Files" chapter. The Customization GUI also adds statements to the Calibre Interactive control file.
- **User input** — Direct user input overrides any preset value or setting in the Calibre Interactive GUI.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The settings present in the Calibre Interactive GUI at the time of the run always take precedence over rule file settings if there is a conflict.

### Related Topics

About Rule Files

About Calibre Interactive Runsets

About Control Files

Managing Rule File and Calibre Interactive GUI Precedence

Environment Variables in Calibre Interactive GUI Fields

File Browser Options

Automatically Loading a Rule File

# Automatically Loading a Rule File

You can instruct Calibre Interactive to automatically load a rule file. The rule file is loaded when a runset is loaded and reloaded at runtime if the rule has been modified. You can choose whether or not to have a confirmation prompt displayed when the rule is about to be loaded.

### Procedure

1. Choose **Setup** > **Preferences**.
2. Choose the **Misc** tab.
3. Enable “Automatically load rules file.”

   The rule file is automatically loaded when you load a runset with a rule file defined, or when you start a run if the rule file has changed since it was last loaded.
4. If desired, disable the “Show prompt” setting, which is on by default.

   **Enabled** — Show a confirmation dialog before automatically loading the rule file.

   **Disabled** — Do not show a confirmation dialog before automatically loading the rule file. If “Show prompt” is disabled, the setting for “Automatically load rules file” applies during Calibre Interactive batch runs.
5. (Optional) Select a setting for “Read Inputs/Outputs fields when loading rules file.”

   This setting is enabled by default, and causes Calibre Interactive to read settings for the Inputs and Outputs panes from the rule file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Results

When “Automatically load rules file” is enabled, Calibre Interactive loads the rule file at these times:

- When you load a runset with a rule file defined.
- When you start a Calibre run if the rule file has changed since it was last loaded.

If “Show prompt” is enabled (the default), a confirmation dialog is displayed before loading the rule file; otherwise, the rule file is loaded without notification.

Loading or not loading the rule file has this effect:

- Rule file loaded — GUI fields are updated from the rule file. This may overwrite runset and user-defined GUI settings.
- Rule file not loaded — GUI fields are not updated from the rule file. The current rule file is used in the Calibre run, but GUI settings take precedence over rule file settings where there is a conflict.

**Note**

When you exit Calibre Interactive, you will be prompted to save the runset. Be aware that rule file settings may have overwritten the original runset settings.

### Related Topics

About Rule Files

Managing Rule File and Calibre Interactive GUI Precedence

# Managing Rule File and Calibre Interactive GUI Precedence

The settings present in the Calibre Interactive GUI at the time of the run always take precedence over rule file settings if there is a conflict. This behavior allows you to customize your Calibre Interactive run without editing the rule file. However, in some cases you may want to insure that rule file settings are used for the run, or disable changes in portions of the Calibre Interactive GUI. Several features are available to manage requirements like this.

You can use the following features in Calibre Interactive to help you manage the precedence of the rule file and the GUI settings:

Load the Rule File

Use Runsets

Disable GUI Settings

Use Customization GUI

Run Calibre Interactive in Batch Mode



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Load the Rule File

You can do the following to help insure that the rule file settings are used for the run:

- Choose **Setup** > **Preferences**, click the **Misc** tab, and enable "Automatically load rules file." This setting causes Calibre Interactive to load the rule file when it loads a runset. This setting applies in Calibre Interactive batch runs if the "Show prompt" preference setting is disabled.

  If you operate Calibre Interactive connected to a design tool and use Templates for File Naming, certain filenames and design-related parameters are determined from the open design database. Templates are applied after the rule file is automatically loaded as part of loading a runset, so template settings override the rule file settings if "Apply Templates" on the **Templates** tab in the Setup Preferences dialog box is enabled.

- Click **Load** on the Rules pane immediately before a run. This loads the rule file settings and overwrites any GUI settings that have been made since the last load of the rule file.

  If you perform this step and use Templates for File Naming, you may want to disable "Read Inputs/Outputs fields when loading rules file" on the **Misc** tab in the Setup Preferences dialog box to avoid overwriting template settings with rule file settings.

## Use Runsets

You can use Calibre Interactive runsets to specify the settings in the GUI, as discussed in "About Calibre Interactive Runsets" on page 33. You can define a runset for each allowed configuration of Calibre Interactive.

If you operate Calibre Interactive connected to a design tool, you can specify Templates for File Naming so that certain filenames and design-related parameters are determined from the open design database rather than the runset or rule file.

## Disable GUI Settings

Calibre Interactive includes several settings which disable portions of the GUI interface. These features are useful if you want to limit the changes made in the Calibre Interactive GUI.

The following settings are on the **Disable** tab of the Setup Preferences dialog box (**Setup** > **Preferences**):

- **Disable app Options** — Disable the settings in the indicated Options pane, where *app* is one of DRC, LVS, DFM, PEX, or PERC.
- **Disable check selection** — Disable changing the Check Selection Recipe in Calibre Interactive nmDRC and DFM.
- **Disable check selection recipe editing** — Disable editing of custom check selection recipes in Calibre Interactive nmDRC and DFM.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Disable environment changes** — Disable editing of environment variables defined in the rule file.
- **Disable customization changes** — Disable editing of customization settings.
- **Disable ERC changes (LVS only)** — Disable changes on the **ERC** tab of the LVS Options pane (**Setup** > **LVS Options**); changes to ERC check selection are also disabled.

In addition, Calibre Interactive nmLVS, Calibre Interactive PERC, and Calibre Interactive PEX have specific options for specifying rule file settings related to device reduction and filtering. The following options are located on the **Gates** tab of the LVS Options and PERC Options panes, and on the **LVS Options** tab of the PEX Options pane:

- **Use settings from rules** — (for "Split Gate Reduction") When enabled, the rule file settings for LVS Reduce Split Gates, LVS Reduce Parallel MOS, and LVS Short Equivalent Nodes are used; GUI settings are ignored. See "Setting Split Gate and Parallel MOS Reduction Options" on page 271.
- **Filter Unused Device Options** — When this checkbox is enabled, the rule file settings for LVS Filter Unused Option are used; GUI settings are ignored. See "Setting Filter Unused Device Options" on page 272.

## Use Customization GUI

You can also use a customization file to control access to the Calibre Interactive GUI. The Runset Control Statements in the Customization File can get and set values in the Calibre Interactive GUI. You can use these statements to construct a Customization GUI which only allows changes to certain Calibre Interactive GUI settings. See "Customization File to Read and Set Calibre Interactive GUI Controls" on page 648 for an example.

## Run Calibre Interactive in Batch Mode

When you run Calibre Interactive in batch mode, the GUI does not open; the settings for the run are controlled by the runset and an optional customization file. This mode of operation makes it easy to control the run parameters and still use the important features of Calibre Interactive, such as runsets, customization files, and export from layout.

See these topics for additional information on running Calibre Interactive in batch mode:

- "Running Calibre Interactive in Batch Mode" on page 102
- "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104
- "Customization GUI Operation in Batch Mode" on page 602



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Third Party Database Options

You can specify the options that Calibre uses when reading in OpenAccess and LEF/DEF databases.

You set third-party database options with one of the methods in the following table.

**Table 3-3. Methods for Setting Third Party Database Options**

| Method | Comment |
|---|---|
| With an environment variable | Set the environment variable MGC_CALIBRE_DB_READ_OPTIONS before starting Calibre Interactive. See "MGC_CALIBRE_DB_READ_OPTIONS Environment Variable for Database Options" on page 81. |
| In the GUI | Choose **Setup** > ***app*** **Options** and set options on the **Database** > **OpenAccess/LEFDEF** sub-tab, where *app* is one of DRC, LVS, DFM, PERC, or PEX. See "OpenAccess/LEFDEF Options Tab" on page 82. |

Third party database options include several mapping files, as explained in the section "Example: Setting the Layer Map, Cell Map, and Object Map for OpenAccess Databases" on page 85.

**Note**

If the environment variable MGC_CALIBRE_DB_READ_OPTIONS is defined, the **Database** > **OpenAccess/LEFDEF** sub-tab displays the options set with the environment variable; the GUI controls are not active.

**Caution**

If you use the statement SVRF FDI Options to set third-party database read options in the rule file, these options are *not* used in a run started from Calibre Interactive. When using Calibre Interactive, the FDI options must be set in the GUI or with MGC_CALIBRE_DB_READ_OPTIONS.

These additional topics may be useful:

- "Providing LEF/DEF Input" on page 88
- "Providing OpenAccess Input" on page 90



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# MGC_CALIBRE_DB_READ_OPTIONS Environment Variable for Database Options

You can specify database read options for third party databases with the environment variable MGC_CALIBRE_DB_READ_OPTIONS. The options apply to OpenAccess input for all Calibre Interactive applications, and to LEF/DEF input for all applications except PEX.

Define the environment variable MGC_CALIBRE_DB_READ_OPTIONS with the desired options before you start Calibre Interactive. See the Layout System statement and the table MGC_CALIBRE_DB_READ_OPTIONS in the *SVRF Manual* for a description of the environment variable.

For example, using the C shell:

```
setenv MGC_CALIBRE_DB_READ_OPTIONS "-layerMap layer_map.txt"
```

When Calibre Interactive is started, the **Database > OpenAccess/LEFDEF** tab on the Setup > DRC Options pane displays the options set with the environment variable; the GUI controls are not active.

Note that if MGC_CALIBRE_DB_READ_OPTIONS is specified, then the options displayed on the **Database > OpenAccess/LEFDEF** tab are not saved when you save a runset unless you also set MGC_CALIBRE_SAVE_ALL_RUNSET_VALUES to a non-zero value.

If your layout database format is OpenAccess or LEF/DEF and you are using a Calibre Interactive control file to perform a command line Calibre run, then you must set MGC_CALIBRE_DB_READ_OPTIONS in order for the database translation options to be set correctly. The required value for the environment variable is included as a comment at the beginning of the control file; an example is shown here:

```
//  If this rule file is run directly from the command line,
//  to set the environment correctly for database translation,
//  please set the value of the environment variable
//  MGC_CALIBRE_DB_READ_OPTIONS to:
//
//  -abortOnEmptyPCell
```

## Related Topics

OpenAccess/LEFDEF Options Tab

Providing LEF/DEF Input

Miscellaneous Preferences

Providing OpenAccess Input



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# OpenAccess/LEFDEF Options Tab

Access with: **Setup** > **Options**, the **Database** > **OpenAccess/LEFDEF** tab. (For PEX, it is the **Database** > **OpenAccess** tab.)

You can control database conversion options for third party databases using Calibre Interactive. These options apply to OpenAccess input for all Calibre Interactive applications, and to LEF/DEF input for all applications except PEX (parasitic extraction).

**Note**

If the environment variable MGC_CALIBRE_DB_READ_OPTIONS is defined, the **Database** > **OpenAccess/LEFDEF** sub-tab displays the options set with the environment variable; the GUI controls are not active.

**Tip**

When using direct read of third-party database formats (input Format set to LEFDEF or OPENACCESS) the database translation options for the fdi2gds utility are reported at the beginning of the control file after the line "// MGC_CALIBRE_DB_READ_OPTIONS". To review these settings, press the Ctrl key and click the **Run *app*** button to open the control file, then make corrections on the **Database** > **OpenAccess/LEFDEF** sub-tab if needed.

**Caution**

If you use the statement SVRF FDI Options to set third-party database read options in the rule file, these options are *not* used in a run started from Calibre Interactive. When using Calibre Interactive, the FDI options must be set in the GUI or with MGC_CALIBRE_DB_READ_OPTIONS.

## Description

The options on the tab are described in the following tables. See "Database Translation Utilities" in the *Calibre Layout Comparison and Translation Guide* for a more detailed description.

## Fields

**Table 3-4. Read Options in the OpenAccess/LEFDEF Tab**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| **Read**: | |
| Texts<br>(-noText) | Read text objects.<br>The default is to read text objects. |
| Pins<br>(-noPinShape) | Read pin shapes.<br>The default is to read pin shapes. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-4. Read Options in the OpenAccess/LEFDEF Tab (cont.)**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Blockages (-outputBlockages) | Read blockages for LEF/DEF or OpenAccess databases; see the documentation of the -outputBlockages option for important information.<br>The default is to not read blockages. |
| Instance Pins (-instPinAsShape) | Read instance pin shapes.<br>The default is to not read instance pin shapes. |
| Read Net Names As (-annotateNets) | Specifies to read net names and how to annotate them.<br>This option is off by default. |
| Read Pin Names As (-annotatePins) | Specifies to read pin names and how to annotate them.<br>This option is off by default. |
| Read Net Type As (-annotateNetType) | Specifies to read net types and annotate them as a property.<br>This option is off by default. |
| Read Instance Name As (-annotateInsts) | Specifies to read instance names and annotate them as a property.<br>This option is off by default. |
| Abort on Reading Empty PCells (-abortOnEmptyPCell) | Causes the Calibre application to exit when an empty PCell appears in an OpenAccess database.<br>This option is on by default in the GUI. |

**Table 3-5. Mapping Files Options in the OpenAccess/LEFDEF Tab**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Use Layer Map File (-layerMap) | Specifies the pathname of the layer mapping file to use when the database is read. The layer mapping file maps specific input layers to a user-specified list of output layers and datatypes. By default, the application maps objects with their corresponding layer number and a datatype of 0 (zero).<br>See the "Example: Setting the Layer Map, Cell Map, and Object Map for OpenAccess Databases" on page 85. |
| Use Object Map File (-objectMap) | Specifies the pathname of the object mapping file to use when the database is read. The object mapping file determines what objects are output to GDS, and how they are mapped.<br>See the "object type tables" in the *Calibre Layout Comparison and Translation Guide*, and the "Example: Setting the Layer Map, Cell Map, and Object Map for OpenAccess Databases" on page 85. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-5. Mapping Files Options in the OpenAccess/LEFDEF Tab (cont.)**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Use Net Properties File (-netProp) | Specifies the pathname of the net properties file to use when the database is read. |
| Use Cell Map File (-cellMap) | Specifies the pathname of the cell mapping file to use when the database is read. The cell mapping file maps input cell names to user-specified cell names. The contents of the cell mapping file are layout system dependent. By default, input cell names are used throughout the run.<br>See "Example: Setting the Layer Map, Cell Map, and Object Map for OpenAccess Databases" on page 85. |
| Use Input Exception File (-inputException) | Specifies the pathname to a file that gives exception levels for certain tool behaviors. |

**Table 3-6. Additional Files Options in the OpenAccess/LEFDEF Tab**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Use Cell GDS Files (-cellGDS) | Specifies that lower-level cells and any instantiated hierarchy should be read from the given files and directories. |

**Table 3-7. Other Options in the OpenAccess/LEFDEF Tab**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| OpenAccess View List (-viewList) | Specifies the order in which to open views for lower-level cells. By default the view list is empty. |
| Write Log File (-logFile) | Specifies to write a runtime log file for the translation. |
| Write Map Info File (-mapInfo) | Specifies to write a map information output file which reports the mappings applied from the -cellMap, -layerMap, and -propertyMap options. |
| Write Layer Names File (-layerNames) | Specifies to use an output layer naming file. |
| Set OA_HOME when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that OA_HOME is set to *$CDS_INST_DIR/share/oa* or *$CDS_INST_ROOT/share/oa* if CDS_INST_ROOT or CDS_INST_DIR is defined. CDS_INST_ROOT is used if both variables are set. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-7. Other Options in the OpenAccess/LEFDEF Tab (cont.)**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Set CALIBRE_READDB_LD_LIBRARY_PATH when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that *$CDS_INST_ROOT/tools/lib* and *$CDS_INST_ROOT/tools/lib/64bit* are added to CALIBRE_READDB_LD_LIBRARY_PATH if CDS_INST_ROOT is defined. The corresponding action is taken if CDS_INST_DIR is defined. CDS_INST_ROOT is used if both variables are set. |
| Set PATH when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that *$CDS_INST_ROOT/share/oa/bin*, *$CDS_INST_ROOT/tools/dfII/bin*, and *$CDS_INST_ROOT/tools/bin* are added to the PATH variable if CDS_INST_ROOT is defined. The corresponding action is taken if CDS_INST_DIR is defined. CDS_INST_ROOT is used if both variables are set. |
| Set CDS_EXP_PCELL_DIR when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that CDS_EXP_PCELL_DIR is set to *./expressPcells* if CDS_INST_ROOT or CDS_INST_DIR is defined. |
| Set CDS_EXP_ALL_PARAMS when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that CDS_EXP_ALL_PARAMS is set to true if CDS_INST_ROOT or CDS_INST_DIR is defined. |
| Set CDS_ENABLE_EXP_PCELL when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that CDS_ENABLE_EXP_PCELL is set to TRUE if CDS_INST_ROOT or CDS_INST_DIR is defined. |

# Example: Setting the Layer Map, Cell Map, and Object Map for OpenAccess Databases

This example describes how to specify the layer map, cell map, and object map files in Calibre Interactive when using OpenAccess as the Layout System.

To specify the mapping files, do one of the following, depending on the GUI you are using:

- Calibre Interactive nmDRC, nmLVS, DFM, PERC, and PEX
  a. Choose **Setup** > ***app*** **Options**.
  b. Click the **Database** tab, and **OpenAccess/LEFDEF** sub-tab.
  c. Choose the **Mapping Files** category.
- Calibre Interactive xACT



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

a. Click **OPENACCESS** on the left panel. If **OPENACCESS** is not present, go to the Inputs page and set the Layout Format to OPENACCESS.

b. Expand the Mapping Files area.

The following sections have examples for the layer map, cell map, and object map files.

## Layermap file

By default, input layers are mapped using the input layer numbers and a datatype record of 0. Assume you want to override this behavior and map specific input layers to a user-specified list of output layers and datatypes. You can do this by writing the following mapping to a text file.

Create a file called *layermap.txt*, as follows:

```
# input_layer_name   purpose_name   output_layer  datatype

        met1           drawing           11           1
        met2           drawing           12           1
        met3           drawing           13           1
```

This file maps the indicated input layers to the output layer numbers and datatypes. To specify the *layermap.txt* file, enable “Use Layer Map File” and specify the file; this is shown for Calibre Interactive— nmDRC in Figure 3-2.

## Cellmap file

By default, the cell names that appear in the OpenAccess database are used on input. Assume you want to override this behavior and rename the cells. Create a file called *cellmap.txt*, as follows:

```
# library   input_name  view_name  output_name
  DesLib     nand2.1     layout      nand2x1
  DesLib     nand2.2     layout      nand2x2
```

This file maps the indicated input cell name from the specified design library and view to the output cell name. To specify the *cellmap.txt* file, enable “Use Cell Map File” and specify the file; this is shown for Calibre Interactive—nmDRC in Figure 3-2.

## Objectmap file

You can use an object map file to specify which objects in the design are output. Create a file called *objectmap.txt* as follows:

```
#Type          SubType    LayerName(opt)    gdsLNum    gdsDType
LayerBlockage  fill       METAL1            31         121
Boundary       PR                           300        101
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

This object map indicates that fill LayerBlockage on layer METAL1 and PR Boundary objects are output. See the "object type table" in the *Calibre Layout Comparison and Translation Guide* for a complete list of object types and subtypes. To specify the *objectmap.txt* file, enable "Use Object Map File" and specify the file; this is shown for Calibre Interactive—nmDRC in Figure 3-2.

**Figure 3-2. Specifying the Layer Map, Object Map, and Cell Map Files**

## Related Topics

"Treatment of Third-Party Layout Databases [Standard Verification Rule Format (SVRF) Manual]

Specifying Net Annotation Options When Reading Layout Databases

Specifying Multiple Layouts for a DRC Run

Providing LEF/DEF Input

Providing OpenAccess Input



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Providing LEF/DEF Input

You provide LEF/DEF input on the Inputs pane **Layout** tab in Calibre Interactive. You can provide LEF/DEF input as database files, or export a DEF file from an open design tool.

**Read LEF/DEF Input from Design Files** . . . . . 88
**Export DEF Layout from Design Tool** . . . . . 89

## Read LEF/DEF Input from Design Files

You can read LEF/DEF input from design files using Calibre Interactive.

Not supported in Calibre Interactive PEX.

### Prerequisites

- Make sure environment variables for database read are set correctly. See "FDI Environment Variables" in the *Calibre Layout Comparison and Translation Guide*.

### Procedure

1. Click **Inputs** on the left panel of Calibre Interactive.
2. Click the **Layout** tab.
3. Select a Format of LEFDEF.
4. Disable "Export from layout viewer."
5. Provide DEF Input as follows:
   - **Files** — One or more DEF files.
   - **Directories** — DEF directories.

   At least one of the Files or Directories entries must be filled in for DEF input.

   Multiple files or directories should be entered on separate lines; use the arrow button to expand or collapse the text entry box.
6. Provide LEF Input as follows:
   - **Tech Files** — One or more LEF technology files.
   - **Files** — One or more LEF files.
   - **Directories** — LEF directories.

   At least one of the Tech File, Files, and Directories entries must be filled in for LEF input. The Tech File entry is optional as long as the technology information is one of the LEF file or directory entries. If the Tech File entry is not used, then the LEF technology file should be first in the list of LEF files.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

7. Specify the "Top Cell."
8. Set other database conversion options as described in "Third Party Database Options" on page 80.

**Tip**

You can define a file naming template for the DEF file. Choose **Setup** > **Preferences**, the **Templates** tab, the **Database** sub-tab, and specify a template in the "DEF File" field. See "Templates for File Naming" on page 148 for more information.

# Export DEF Layout from Design Tool

If your design tool is connected to Calibre Interactive and is capable of exporting the layout in DEF format, you can export the currently open design to Calibre.

Not available in Calibre Interactive PEX.

## Procedure

1. Click the **Inputs** button on the left panel of Calibre Interactive.
2. Click the **Layout** tab.
3. Select a Format of LEFDEF.
4. Enable "Export from layout viewer."
5. Specify the DEF output file in the DEF Files field. The design streamed out by the design tool is written to this file; the file is overwritten if it already exists.
6. Provide LEF Input as follows:
   - **Tech File** — The LEF technology file.
   - **Files** — One or more LEF files.
   - **Directories** — LEF directories.

   At least one of the Tech File, Files, and Directories entries must be filled in for LEF input. The Tech File entry is optional as long as the technology information is in one of the LEF file or directory entries. If the Tech File entry is not used, then the LEF technology file should be first in the list of LEF files.
7. Specify the "Top Cell."
8. Set streamout options in your design tool.
9. Set other database conversion options as described in "Third Party Database Options" on page 80.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

You can define a file naming template for the DEF file. Choose **Setup** > **Preferences**, the **Templates** tab, the **Database** sub-tab, and specify a template in the "DEF File" field. See "Templates for File Naming" on page 148 for more information.

# Providing OpenAccess Input

You provide OpenAccess database input in the **Layout** tab on the Inputs pane in Calibre Interactive. You specify database conversion options on the Options pane in the **Database** > **OpenAccess/LEFDEF** sub-tab.

## Prerequisites

- For Cadence Virtuoso users, make sure CDS_INST_ROOT or CDS_INST_DIR is set. CDS_INST_ROOT takes precedence if both CDS_INST_ROOT and CDS_INST_DIR are defined.
- Make sure LD_LIBRARY_PATH includes the libstdc++.so.6.0.8 compiler libraries. This is true by default for Linux®[1] machines with RHEL 4u6 or later.

  For example with Cadence Virtuoso 6.15.500:

  ```
  setenv LD_LIBRARY_PATH ${CDS_INST_DIR}/share/oa/lib/
  linux_rhel40_gcc44x_64/opt:
  ${CDS_INST_DIR}/tools/cdsSkillPcell/lib/64bit:
  ${CDS_INST_DIR}/tools/lib/64bit:${LD_LIBRARY_PATH}
  ```

  The actual path depends on the tool version.
- Make sure environment variables for database read are set correctly. See "FDI Environment Variables" in the *Calibre Layout Comparison and Translation Guide*.

  In particular, if your design tool requires OpenAccess version 22.50, specify this setting with one of the following options:

  - In the rules file, include the SVRF statement LAYOUT SYSTEM OA "22.50".
  - Set the environment variable MGC_FDI_OA_VERSION to 22.50.

  Calibre Interactive does this automatically when connected to Cadence Virtuoso versions 6.17 or later. The setting specified by the SVRF statement overrides MGC_FDI_OA_VERSION.

## Procedure

1. Click the **Inputs** button on the left panel of Calibre Interactive.

1. Linux® is a registered trademark of Linus Torvalds in the U.S. and other countries.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Click the **Layout** tab.
3. Select a Format of OPENACCESS.
4. Enter the path to the directory containing the library definitions file in the “Library Path” field. The “Library Path” should be a directory that contains either a cds.lib (if OA_HOME is set) or lib.defs file.

   If the library definitions file is in the current working directory (the directory you invoked Calibre Interactive from), it is not necessary to define “Library Path.”
5. Enter the name of the design library in the “Library Name” field. You can click the button to open a library browser. The library browser allows you to select the library, cell, and view, then automatically fills in the correct GUI fields.

   Note that the library browser will not work correctly if “Library Path” is not correctly defined.
6. Specify the Top Cell and View name if these have not been filled in by the library browser.
7. Choose **Setup** > ***app*** **Options** from the menu, then click the **Database** > **OpenAccess/LEFDEF** sub-tab.
8. For Cadence Virtuoso users, do the following to automatically set certain Cadence environment variables:

   a. Choose the **Other Options** category on the **OpenAccess/LEFDEF** tab.

   b. Enable “Set OA_HOME when CDS_INST_ROOT or CDS_INST_DIR is set.”

   c. Enable “Set CALIBRE_READDB_LD_LIBRARY_PATH when CDS_INST_ROOT or CDS_INST_DIR is set.”

   d. Enable “Set PATH when CDS_INST_ROOT or CDS_INST_DIR is set.”

   e. If you are using Express Pcells, enable “Set CDS_EXP_PCELL_DIR when CDS_INST_ROOT or CDS_INST_DIR is set.”

   f. If you are using Express Pcells, enable “Set CDS_EXP_ALL_PARAMS when CDS_INST_ROOT or CDS_INST_DIR is set.”

   g. If you are using Express Pcells, enable “Set CDS_ENABLE_EXP_PCELL when CDS_INST_ROOT or CDS_INST_DIR is set.”
9. Set other database conversion options as needed in the **Database** > **OpenAccess/LEFDEF** sub tab. For complete details, see “OpenAccess/LEFDEF Options Tab” on page 82.

   These options control what is read from the OpenAccess database and passed on as input to Calibre.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

You can define a file naming template for the Library and View. Choose **Setup** > **Preferences**, the **Templates** tab, the **Database** sub-tab, and specify templates in the "Library" and "View" fields. See "Templates for File Naming" on page 148 for more information.

## Related Topics

MGC_CALIBRE_DB_READ_OPTIONS Environment Variable for Database Options

Automatically Saving the Express Pcell Cache



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setting Database and General Options in Calibre Interactive

You can set options relating to the input databases using Calibre Interactive. You can also specify a Layout Text file and specify additional rules files and rule file statements. The method to set these options is similar in all Calibre Interactive applications.

**Specifying a Layout Text File in Calibre Interactive** . . . . . 93
**Specifying Database Precision and Resolution in Calibre Interactive** . . . . . 94
**Specifying Third Party Database Options in Calibre Interactive** . . . . . 94
**Including Other Rule Files and SVRF Commands in Calibre Interactive** . . . . . 95
**Specifying Net Annotation Options When Reading Layout Databases** . . . . . 96
**Specifying Virtual Net Connections in Calibre Interactive** . . . . . 96

## Specifying a Layout Text File in Calibre Interactive

You can specify a Layout Text File in Calibre Interactive. The Layout Text File is used to specify free-standing cell-based text in the rule file.

### Prerequisites

A text file of the correct syntax for the Layout Text File statement.

### Procedure

1. Click ***app* Options** on the left panel, where *app* is one of DRC, LVS, DFM, PEX, or PERC. If the ***app* Options** button is not present, choose **Setup** > ***app* Options** from the menu to display it.
2. Click the **Include** tab.
3. Enter the filename of a Layout Text File in the text entry field for “Include Layout Text File.”

   The Layout Text File statement is added to the Calibre Interactive control file. Layout Text statements in the Layout Text File are added to any Layout Text statements present in the rule file.

### Related Topics

Layout Text File [Standard Verification Rule Format (SVRF) Manual]

Layout Text [Standard Verification Rule Format (SVRF) Manual]

Use of Text in Calibre Applications [Calibre Verification User's Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Specifying Database Precision and Resolution in Calibre Interactive

You can specify the SVRF Precision and Resolution statements in the Calibre Interactive control file.

## Procedure

1. Click ***app* Options** on the left panel, where *app* is one of DRC, LVS, DFM, PEX, or PERC. If the button is not present, choose **Setup** > ***app* Options** from the menu to display it.
2. Click the **Database** tab.
3. Click the **Precision** tab.
4. To set the precision, enable the "Specify precision" checkbox, and enter the desired precision in the text entry field. See Precision in the *Standard Verification Rule Format (SVRF) Manual* for information on setting this parameter.

   A compiler error results if the database precision does not match the rule file precision. To change this error to a warning and allow the DRC run to proceed, enable "Override layout value."
5. To set the resolution, select "Specify resolution" and enter the desired resolution in the text entry field. See Resolution in the *Standard Verification Rule Format (SVRF) Manual* for information on setting this parameter.

# Specifying Third Party Database Options in Calibre Interactive

You can set several Calibre Interactive options that affect how your OpenAccess or LEF/DEF design is read.

## Procedure

1. Click ***app* Options** on the left panel, where *app* is one of DRC, LVS, DFM, PEX, or PERC. If the ***app* Options** button is not present, choose **Setup** > ***app* Options** from the menu.
2. Click the **Database** tab.
3. Click the **OpenAccess/LEFDEF** sub-tab.

   For information on the tab contents, refer to "Third Party Database Options" on page 80.

## Related Topics

Providing OpenAccess Input



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Providing LEF/DEF Input

Specifying Net Annotation Options When Reading Layout Databases

# Including Other Rule Files and SVRF Commands in Calibre Interactive

You can specify additional rule files and SVRF commands using the Calibre Interactive GUI. This allows you to make adjustments to the run without editing your rule file.

## Procedure

1. View the Options pane for your Calibre Interactive application. If the **Options** button is not present in the left side of the Calibre Interactive window, choose **Setup >** ***app*** **Options**, where ***app*** is one of DRC, LVS, DFM, PEX, or PERC.
2. Click the **Include** tab.
3. To include one or more rule files, enter one or more include filenames in the top text area.

   Enable "Include Rule Files and Statements After Main <*app*> Rules File" if you want the included rule files to be placed after the rule file specified on the Inputs pane. This is useful if the order of rule file statements matters; this can happen if the main rule file includes conditional compilation directives, such as #DEFINE, which can impact the compilation of statements found in an included file. This option is off by default.
4. Do the following to include individual rule file statements:
   a. Enable "Include Rule Statements."
   b. If the rule file is TVF, use the button dropdown list to indicate whether you are including SVRF or TVF statements.

      (Only SVRF statements may be included in an SVRF rule file.)
   c. Enter one or more commands into the text area; one command per line is allowed.

      The rule file statements are included in the Calibre Interactive control file; if SVRF statements are included in a TVF rule file, the SVRF statements are enclosed within a TVF::VERBATIM block.

      **Tip**

      When you click the **Load** button on the Rules pane or the **Run <*app*>** button, Calibre compiles all included rule files and rule file statements along with the rule file specified in the Rules pane.

## Related Topics

Pre-Processor Directives [Standard Verification Rule Format (SVRF) Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Calibre Interactive Operation with a TVF Rule File

Specifying a Layout Text File in Calibre Interactive

# Specifying Net Annotation Options When Reading Layout Databases

Calibre Interactive enables you to choose whether to read net names when loading LEF/DEF or OpenAccess layout databases. By default, net names are not annotated when reading these databases.

## Procedure

1. Click the ***app* Options** button on the left panel, where *app* is one of DRC, LVS, DFM, PERC, or PEX. If the button is not present, choose **Setup > *app* Options** from the menu to display it.
2. Click the **Database** tab, and then click the **OpenAccess/LEFDEF** tab.
3. Choose the **Read Options** category.
4. In the area for "Read Net Names As", select the method for adding annotations to the shapes that belong to a net:
   - **Text** — add annotations as text objects.

     **All Cells** — read net names from all cells.

     **Top Cell** — read net names from the top cell only.
   - **Property** — add annotations as properties using the specified property name.

## Related Topics

Third Party Database Options

Inputs for a Calibre Interactive nmDRC Run

# Specifying Virtual Net Connections in Calibre Interactive

Using the virtual net connections options, you can instruct Calibre to treat multiple, physically-disjoint layout net segments as if they were actually connected.

Corresponding rule file statements: Virtual Connect Colon, Virtual Connect Name, Virtual Connect Report, Virtual Connect Report Maximum



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

The Virtual Connect statements are primarily intended for Calibre nmLVS, PERC, and PEX applications.

Virtual connections in Calibre nmDRC and DFM applications can cause unintended results. The statements do not function when DRC Incremental Connect Yes is specified because virtual connections do not obey connectivity zones in a rule file.

## Procedure

1. Click the ***app* Options** button on the left panel, where *app* is one of DRC, LVS, DFM, PERC, or PEX. If the button is not present, choose **Setup > *app* Options** from the menu to display it.
2. Click the **Connect** tab.
3. Select the handling of nets with a colon (:).
   - **Connect nets with colon (:)** — Virtually connect physically disjoint nets whose names agree up to the first colon character found in the names. The name of the new net is the shared name leading up to the colon. For example, virtually connect net clock:1 to net clock:2 as the net clock.
4. Select the handling of named nets. Net name matching is case-insensitive for all options.
   - **Don't connect nets by name** — Treat synonymous disjoint nets as separate nets. This is the default.
   - **Connect all nets by name** — Virtually connect physically disjoint nets whose names are identical. For example, choosing this option virtually connects net RESET to net Reset and net reset.
   - **Connect nets named** — Virtually connect physically disjoint nets whose names are in the list. The wildcard character "?" matches zero or more characters.
5. Specify reporting options:
   - **Report connections made by name** — Instruct Calibre to document virtual connections in the report.
   - **Don't report connections that are physically connected** — Do not report virtual connections by name that are physically connected higher in the hierarchy; only virtual connections that are not physically connected higher in the hierarchy are included in the report. (Virtual Connect Report YES UNSATISFIED)
   - **Max. connected messages printed in report** — Select "All" or specify a "Count" to control the number of messages included in the report.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Running a Job Using Calibre Interactive

This section describes how to set up and run Calibre using Calibre Interactive.

**Note**

For information on running Calibre Interactive in batch mode, see "Running Calibre Interactive in Batch Mode" on page 102.

**Starting a Job with Calibre Interactive** . . . . . 98
**Transcript Pane in Calibre Interactive** . . . . . 99
**Calibre Quality Agent for Calibre Interactive** . . . . . 101

## Starting a Job with Calibre Interactive

Use these general steps to set up and run a Calibre job for any of the supported applications.

### Prerequisites

- Calibre Interactive System Requirements are met.

### Procedure

1. Start Calibre Interactive with one of the methods described in "Invoking Calibre Interactive Tools" on page 64.
2. Do one of the following to populate the Calibre Interactive GUI:
   - Load a runset with **File** > **Load Runset**.
   - Specify all rule file, input, output, and run control information by clicking the appropriate button in the left button panel, and entering information in the data entry panel. Refer to the application-specific chapters for detailed information.

   All setup buttons on the left side of the window must be green before clicking the Run button. A setup button changes from red to green when you have specified all required information associated with that button.

   Be aware that simply because all of the buttons are green does not necessarily mean the information in the GUI is correct. It means that there appears to be sufficient information to start a Calibre job.
3. (Optional) Specify trigger functions, as described in "Trigger Functions in Calibre Interactive" on page 657.
4. (Optional) Specify a customization file, as described in "Customization Files" on page 587.
5. (Optional) Choose **Setup** > **Preferences** to specify preferences; see "Setup Preferences" on page 142.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

You can enable "Run Calibre when runset is opened" on the **Misc** tab to have Calibre Interactive start the run automatically when the runset is loaded.

6. (Optional) Save a runset with **File** > **Save Runset**. This saves all the Calibre Interactive settings except those in the Select Checks dialog box for Calibre Interactive nmLVS and Calibre Interactive 3DSTACK.
7. Click the Run button when you have verified that the GUI setup is correct. The name of the button reflects the Calibre application you are invoking.

**Tip**

You can generate and view a control file without running Calibre by holding down the Ctrl key and clicking **Run**. In addition, the calibre command line for the run is written to the Transcript pane.

### Results

If an error occurs in Calibre Interactive before the Calibre run is started, the error is displayed in a pop-up window or in the Calibre Quality Agent for Calibre Interactive.

When the Calibre run starts, the view switches to the Transcript pane, and any run output, including errors and warnings, is displayed here. See "Transcript Pane in Calibre Interactive".

Some applications have options to open the results database in Calibre RVE when the run is complete; see "Getting Started with Calibre RVE" in the *Calibre RVE User's Manual*. In addition, the Calibre Interactive GUI may open a report file in a text viewer when the run completes; see "Viewing and Editing Text Files in Calibre Interactive" on page 125.

### Related Topics

Using Calibre Interactive to Perform DRC

Using Calibre Interactive to Perform DFM

Using Calibre Interactive to Perform LVS

Using Calibre Interactive to Perform Parasitic Extraction

Using Calibre Interactive to Perform PERC

Calibre Interactive for Calibre 3DSTACK [Calibre 3DSTACK User's Manual]

## Transcript Pane in Calibre Interactive

You display the Transcript pane by clicking the **Transcript** button on the left panel of Calibre Interactive. The Transcript pane displays the Calibre transcript (also known as the log), which provides valuable information about your Calibre run.

You can turn off transcript output if desired; choose **Setup** > **Preferences**, click the **Disable** tab, and enable "Disable transcript output."



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If your Calibre run generates errors or warnings, Calibre Interactive displays each error and warning in tabs in the bottom portion of the Transcript pane. In Calibre Interactive xACT, when multiple configurations are loaded, messages from each configuration file are appended to the transcript window.

You can select any error or warning to scroll the top half of the Transcript pane to the relevant portion of the run transcript, as shown in Figure 3-3. If there are multiple errors or warnings listed in a tab, you can scroll through them with the up and down arrow keys on the keyboard. The menu selection **Transcript** > **Save Errors and Warnings As** saves only the errors and warnings to a file.

**Figure 3-3. Transcript Pane in Calibre Interactive**

The following table describes the contents of the **Transcript** menu in Calibre Interactive.

**Table 3-8. Transcript Menu Contents in Calibre Interactive**

| Menu Selection | Description |
|---|---|
| Save As | Save the current transcript. |
| Echo to file | Write the transcript to the specified file as Calibre executes. |
| Save Errors and Warnings As | Save only the errors and warning from the transcript. The errors and warnings are grouped together with a count for each given at the top of each section. |
| Search | Search in the Transcript pane for a specific character string. |

## Related Topics

Creating and Viewing the Control File Without Starting a Run

Running Calibre Interactive



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Quality Agent for Calibre Interactive

The Calibre Quality Agent dialog box is displayed when Calibre Interactive encounters an application error. You use the dialog box to send feedback about the error conditions to the Calibre team.

The Calibre Quality Agent dialog box is shown in the following figure.

**Figure 3-4. Calibre Quality Agent Dialog Box for Calibre Interactive**

**Note**

No design data is sent with the Calibre Quality Agent emails.

The feedback from the Calibre Quality Agent email is very helpful to the Calibre team, especially when you include the steps leading up to the application error. The error and code traceback generated at the time of the error are included automatically, but are usually not enough information by themselves to correct an application error. A copy of the feedback email is automatically sent to the email address you provide.

Click the **Details** tab to view the automatically included error report and code traceback. Enable or disable “Include path to database” on the **Details** tab to choose whether the path to your runset (if any) is included.

The settings you choose for the Calibre Quality Agent dialog box remain in effect for the current Calibre Interactive session.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Running Calibre Interactive in Batch Mode

You can run Calibre Interactive in batch mode without opening the GUI. This mode of operation preserves the important features of Calibre Interactive, such as runsets, customization files, and export from layout, while letting you run in batch mode.

The following tasks describe how to run Calibre Interactive in batch mode with various options:

**Starting a Basic Batch Run in Calibre Interactive** . . . . . 102
**Rule Check Selection in Calibre Interactive Batch Runs** . . . . . 103
**Specifying Layout and Schematic Export for a Calibre Interactive Batch Run** . . . . . 104

## Starting a Basic Batch Run in Calibre Interactive

You can run Calibre Interactive in batch mode without opening the GUI. All Calibre Interactive options are set in the runset.

### Procedure

1. Set up Calibre Interactive as described in "Running a Job Using Calibre Interactive" on page 98, but do not start a run.
2. If you want Calibre RVE to start after the batch run and open the results database, choose **Setup** > **Preferences**, the **Misc** tab, and enable "Start RVE after a batch run."
3. Save a runset with **File** > **Save Runset**.
4. Exit Calibre Interactive.
5. Start the batch run with the following command:

   ```
   calibre -gui -app -runset_options_display my_runset -batch
      >& batch.log
   ```

   where *app* is one of drc, lvs, perc, pex, or dfm, and *my_runset* is the runset saved in Step 3.

   The option -runset_options_display causes the runset options to be echoed to stdout. This is useful so you can determine what Calibre Interactive settings might override rule file settings. If you do not want the runset options echoed, use the option -runset instead.

### Related Topics

Customization GUI Operation in Batch Mode

Running Batch Calibre with a Calibre Interactive Control File

Specifying Layout and Schematic Export for a Calibre Interactive Batch Run



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Rule Check Selection in Calibre Interactive Batch Runs

The method for selecting checks in a Calibre Interactive batch run depends on whether you are running Calibre Interactive nmDRC, Calibre Interactive DFM, or Calibre Interactive nmLVS.

**Calibre Interactive nmDRC and DFM Check Selection in Batch Mode . . . . . 103**

**Calibre Interactive nmLVS ERC Check Selection in Batch Mode. . . . . 103**

## Calibre Interactive nmDRC and DFM Check Selection in Batch Mode

The Check Selection Recipe is the preferred method for selecting the rule checks that are evaluated in a Calibre Interactive nmDRC or Calibre Interactive DFM batch run, just as for interactive GUI operation. The Check Selection Recipe is included in the runset specified for the batch run. The default check recipe is “Checks selected in the rules file”.

**Caution**

It is possible, but not recommended, to use a customization file to select checks for a batch run of Calibre Interactive nmDRC or Calibre Interactive DFM. If you are using CUSTOM::CHECK controls in a Customization GUI for Calibre Interactive nmDRC or DFM, make sure that the “Checks selected in the rules file” built-in check selection recipe is selected, as described in “Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive” on page 604.

See “Customization GUI Operation in Batch Mode” on page 602 for further instructions.

### Related Topics

Check Selection Recipe

Starting a Basic Batch Run in Calibre Interactive

## Calibre Interactive nmLVS ERC Check Selection in Batch Mode

ERC checks in batch mode for Calibre Interactive nmLVS are selected using a customization file.

The settings in the Select Checks dialog box for ERC cannot be used in batch mode because Calibre Interactive runsets do not save the check selections in the Select Checks Dialog Box. You must select ERC checks in the rule file, or select them using a customization file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

A customization file can include a CUSTOM::CHECK command to select the rule checks that are run. The customization file is specified with the -customgui or -custom command line options in batch mode.

For example:

```
calibre -gui -lvs -runset_options_display lvs_runset
   -customgui lvs_custom_checks -batch > batch_gui.log
```

Where *lvs_custom_checks* is the name of the customization file; an example can be found at "Customization File to Select ERC Rule Checks" on page 651.

### Related Topics

Invoking Calibre Interactive from the Command Line

Starting a Basic Batch Run in Calibre Interactive

Customization GUI Operation in Batch Mode

## Specifying Layout and Schematic Export for a Calibre Interactive Batch Run

You can export the design from an open design tool for use as input for the Calibre Interactive batch run.

### Procedure

1. Open the design in your design tool to the desired cell and view.
2. Make sure the socket connection is started and uses the default socket number (9189 for layout export, 9199 for schematic export).

   To do this in Calibre DESIGNrev, choose **Verification > RVE/CI Setup**, and click **Start Server** to start the socket connection. See "Interfacing Calibre Interactive to Layout and Schematic Viewers" on page 705 for instructions for other design tools.

   If your design tool is not using the default socket numbers, you need to specify the socket number. This can be done with an environment variable; see "Setting the Calibre Interactive Socket Port for a Layout Viewer" on page 707 and "Setting the Calibre Interactive Socket Port for a Schematic Viewer" on page 708.
3. Open Calibre Interactive and make the following settings:
   a. Enable "Export from layout viewer" on the **Layout** tab of the Inputs pane.
   b. (Optional) Enable "Export from schematic viewer" on the **Netlist** tab of the Inputs pane if you are running Calibre Interactive nmLVS, PERC, or PEX.

      In Calibre Interactive xACT, the setting is on the **Inputs** page when source names are used.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

c. Disable “Ignore layout and source export in batch mode” on the **Misc** tab in the Setup Preferences dialog box (**Setup** > **Preferences**).

   In Calibre Interactive xACT, the setting is on the **Preferences** page in the Misc section.

d. Check that all other Calibre Interactive settings for your run are correct.

4. Save the runset with **File** > **Save Runset**.
5. Exit Calibre Interactive.
6. Start the Calibre Interactive batch run. The following command is for a Calibre nmDRC run:

```
calibre -gui -drc -runset drc_runset -batch
```

**Tip**

For supported design tools, you can add a custom menu item to the integration menu of the design tool. The new menu item can be defined to start the Calibre Interactive batch run. See “Creating Custom Menus in Design Tools” on page 710.

### Related Topics

Starting a Basic Batch Run in Calibre Interactive

Setting Layout and Schematic Export in Calibre Interactive

# Calibre Interactive Operation with a TVF Rule File

You specify a TVF rule file in Calibre Interactive on the Rules pane just as you do for an SVRF rule file. Some additional options are available when running with a TVF rule file. Debugging methods can be slightly different with a TVF rule file.

The following sections are included:

- Specify the TVF Rule File
- Include TVF Statements with the Calibre Interactive Run
- Save the Generated SVRF Rule File
- Debugging Errors in a Calibre Interactive Run with a TVF Rule File

## Specify the TVF Rule File

Click the **Rules** button on the left panel on Calibre Interactive and specify the TVF rule file in the Rules File field.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Include TVF Statements with the Calibre Interactive Run

You can use Calibre Interactive to add TVF statements to a run that uses a TVF rule file as the main rule file on the Rules pane.

Choose **Setup** > ***app*Options** and click the **Include** tab. Enable "Include Rule Statements" and choose TVF in the button dropdown list. Enter statements in the text area; the statements are added to the control file when you start the run. If you choose SVRF in the button dropdown list the statements are wrapped in a TVF::VERBATIM block.

Also see "Including Other Rule Files and SVRF Commands in Calibre Interactive" on page 95.

## Save the Generated SVRF Rule File

You can enable a preference setting to save the translated SVRF code to a file. This option corresponds to the calibre -E command line option.

Choose **Setup** > **Preferences**, click the **Misc** tab, and enable "Save TVF Rules to SVRF before running Calibre." The saved file is named _<*tvf_rule_file*>_.*svrf*, where <*tvf_rule_file*> is replaced with the name of the TVF rule file specified on the Rules pane. The generated SVRF file includes statements from the Calibre Interactive control file.

## Debugging Errors in a Calibre Interactive Run with a TVF Rule File

If the TVF rule file or the SVRF code generated from it has a compilation error the run is stopped. The generated SVRF code is still saved if "Save TVF Rules to SVRF before running Calibre" is enabled on the **Misc** tab of the Setup Preferences dialog box *and* the run is on localhost. In such a case the generated SVRF file *does not* include statements from the Calibre Interactive control file.

If the generated SVRF code contains a compilation error, a pop-up message similar to the following is displayed:

```
Error while compiling rules file <tvf file>
Error xxx on line xxx of SVRF generated from TVF ...
```

where the line number refers to the line number in the generated SVRF code. If "Save TVF Rules to SVRF before running Calibre" is enabled you can examine the generated SVRF code to debug the error.

If the error occurs in the TVF code, an error similar to the following occurs:

```
Error while compiling rules file <tvf file>
Error xxx ...
(file <tvf file> line xxx)
```

where the line number refers to the TVF rule file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Related Topics

Running a Job Using Calibre Interactive

Running Calibre Interactive in Batch Mode

Compile-Time TVF [Standard Verification Rule Format (SVRF) Manual]

Error Handling in Compile-Time TVF Rule File [Standard Verification Rule Format (SVRF) Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Check Selection in Calibre Interactive

The method for the checks that are executed in a Calibre Interactive run depends on which application you are running.

**Check Selection in Calibre Interactive nmDRC and Calibre Interactive DFM** . . . . . 109
**Check Selection for ERC in Calibre Interactive nmLVS** . . . . . 125



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Check Selection in Calibre Interactive nmDRC and Calibre Interactive DFM

The Check Selection Recipe specifies the checks that are executed during a Calibre Interactive nmDRC or Calibre Interactive DFM run. The Check Selection Recipe is specified on the Rules pane.

Check selection recipes are explained in the following sections:

**Check Selection Recipe** . . . . . 109
**Specifying a Check Selection Recipe** . . . . . 110
**Creating a Custom Check Selection Recipe** . . . . . 110
**Basic Editing of Custom Check Selection Recipes** . . . . . 111
**Advanced Editing of Check Selection Recipes in Calibre Interactive** . . . . . 113
**Running Only Checks Selected in the Rule File with a Custom Check Selection Recipe** 116
**Check Selection Recipe Editor Dialog Box** . . . . . 118

## Check Selection Recipe

A Check Selection Recipe is a set of expressions for selecting the checks to execute during a Calibre run. You can use built-in check recipes or create user-defined check recipes tailored for specific cases. Check Selection Recipes are used in Calibre Interactive nmDRC and Calibre Interactive DFM. The recipes are saved to the runset.

Calibre Interactive defines the following two built-in check recipes:

- **Checks selected in the rules file** — Execute the same set of checks that would be executed in a batch Calibre run. This is the only built-in recipe that obeys DRC [Un]Select Check and DFM [Un]Select Check statements in the rule file.
- **All checks** — Executes *all* checks in the rule file except those excluded by pre-processor directives such as #IFDEF. DRC [Un]Select Check and DFM [Un]Select Check statements are *not* considered for the “All checks” recipe.

Custom Check Selection Recipes, also called user recipes, are useful for specific tasks or designs. User-defined check recipes allow you to tailor the rule checks that are executed to the type of design work you are doing. For example, you can define a custom recipe to only run checks for metal layers, or run all checks except those that include connectivity. User-defined check recipes and the currently active check recipe are saved to the Calibre Interactive runset.

You can disable selecting and editing the check recipe with the settings “Disable check selection” and “Disable check selection recipe editing” on the **Disable** tab of the Setup Preferences dialog box (**Setup** > **Preferences**).



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Specifying a Check Selection Recipe

The Check Selection Recipe specifies the checks that are executed in a Calibre Interactive nmDRC or DFM run. You can use a built-in check recipe or define your check recipe.

## Prerequisites

- Calibre Interactive nmDRC or Calibre Interactive DFM is running.
- "Disable check selection" is not enabled on the **Disable** tab of the Setup Preferences dialog box (**Setup** > **Preferences**).

## Procedure

1. Click the Rules button on the left panel to display the Rules pane.
2. Specify a rule file.
3. Choose a check recipe using the dropdown list for Check Selection Recipe, as shown in the following figure.

## Related Topics

Check Selection Recipe

# Creating a Custom Check Selection Recipe

You can create a new custom check recipe or copy an existing recipe from the Check Selection Recipe Editor dialog box. Custom check recipes (also called user recipes) allow you to control which Calibre rule checks are executed during a Calibre nmDRC or Calibre DFM run. Check recipe definitions are saved in the Calibre Interactive runset, along with the currently active check recipe.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Prerequisites

- Calibre Interactive nmDRC or Calibre Interactive DFM is running.
- A rule file is available.
- "Disable check selection" is not enabled on the **Disable** tab of the Setup Preferences dialog box (**Setup** > **Preferences**).

### Procedure

1. Click the **Rules** button on the left panel of Calibre Interactive to view the Rules pane.
2. Specify a rule file. You cannot open the check recipe editor without specifying a rule file.
3. Click the **Edit** button to the right of the Check Selection Recipe entry.

   A view of the recipe editor is shown in "Check Selection Recipe Editor Dialog Box" on page 118.
4. Do one of the following:

| To do this … | Take these steps … |
|---|---|
| Create a new recipe | 1. Click **New** in the top button bar.<br>A new recipe is initialized with the checks selected in the rule file. |
| Copy an existing recipe | 1. Choose the recipe to copy in the Recipe dropdown list.<br>2. Click **Copy** in the top button bar. |

5. Proceed to "Basic Editing of Custom Check Selection Recipes" on page 111 or "Advanced Editing of Check Selection Recipes in Calibre Interactive" on page 113.

### Related Topics

Check Selection Recipe

Check Selection Recipe Editor Dialog Box

## Basic Editing of Custom Check Selection Recipes

The basic editing controls allow you to construct check recipes based on a short list of common expressions, such as "Checks selected in the rule file" and "Checks with density". You can also include and exclude individual checks and check groups.

Custom check recipes (also called user recipes) allow you to control which Calibre rule checks are executed during a Calibre nmDRC or Calibre DFM run. Check recipe definitions are saved in the Calibre Interactive runset, along with the currently active check recipe.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Prerequisites

- Calibre Interactive nmDRC or Calibre Interactive DFM is running.
- A rule file is loaded on the Rules pane.
- “Disable check selection recipe editing” is not enabled on the **Disable** tab of the Setup Preferences dialog box (**Setup** > **Preferences**).

## Procedure

1. Click the **Rules** button on the left panel of Calibre Interactive to view the Rules pane.
2. Specify a rule file. You cannot edit a check recipe without specifying a rule file.
3. Choose the recipe you want to edit in the Check Selection Recipe dropdown list.

**Tip**

If you edit a built-in recipe, it is automatically saved as a user recipe with “(Modified)” appended to the recipe name. Only one modified version of each built-in recipe is saved at a time—new modifications overwrite any existing modified recipe.

4. Click the **Edit** button at the far right of the Check Selection Recipe entry to open the Check Selection Recipe Editor dialog box.
5. For basic recipe editing, make sure the dialog box appears as shown in the following figure. If it doesn’t, click the **Advanced** button to collapse the advanced controls.

6. Use the Include and Exclude selections to select the types of checks in the check recipe.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

7. Enable and disable individual checks and check groups in the Recipe Checks area. The **Groups** tab is only displayed if check groups are defined with the Group statement.
8. (Optional) Click **Export** to save the check recipe to a file; recipe files are given the *.rcp* file extension. The recipe file can be imported by other users with the **Import** button. Saved recipes can be imported by Calibre RealTime Custom and Calibre RealTime Digital.
9. Click **OK** to save the recipe and exit the dialog box.

**Tip**

If you edited a built-in recipe, you can click the **Rename** button to give the modified recipe a new name.

10. (Recommended) Choose **File > Save Runset** to save the recipe definition to a runset.

**Tip**

You must either save a runset or export the recipe to a file if you want the new check recipe to be available in later sessions of Calibre Interactive.

### Related Topics

Advanced Editing of Check Selection Recipes in Calibre Interactive

Check Selection Recipe Editor Dialog Box

Creating a Custom Check Selection Recipe

Check Selection Recipe

## Advanced Editing of Check Selection Recipes in Calibre Interactive

The advanced editing controls in the Recipe Editor allow you to select checks by layer and group, apply a wildcard filter, and select and unselect individual rule checks.

Custom check recipes (also called user recipes) allow you to control which Calibre rule checks are executed during a Calibre nmDRC or Calibre DFM run. Check recipe definitions are saved in the Calibre Interactive runset, along with the currently active check recipe.

### Prerequisites

- Calibre Interactive nmDRC or Calibre Interactive DFM is running.
- A rule file is loaded on the Rules pane.

### Procedure

1. Open a recipe for editing and make selections using the basic editing controls, as described in “Basic Editing of Custom Check Selection Recipes” on page 111.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Click the **Advanced** button to expand the advanced editing controls if they are not visible. The dialog box should look similar to that shown in Figure 3-7 in "Check Selection Recipe Editor Dialog Box" on page 118.

3. Do the following to add an expression to the recipe definition:

   a. Choose an expression category in the Expression area.

   b. Specify Addition or Subtraction below the Expression area to indicate whether you want to include or exclude checks with the expression.

   c. In the expression view area on the right, select items that you want to be part of the expression. The view in this area depends on the expression category. You can use the following methods to select checks:

      - Click in the checkbox next to the rule check (or layer name if selecting by layer).
      - Right-click for a selection menu. Note: In the **Layers** tab, you must right-click in a selection to open the menu. In addition, there is only one menu selection available for the **Layers** tab, unlike the other expression views.
      - If the option is available, specify a wildcard pattern.

      Some expression categories have a separate **Preview** tab to display the rule checks that are selected by the expression.

   d. Click **Apply expression** or **Apply checks** to update the check recipe definition.

      - **Apply expression** — Add the expression to the recipe definition; for example, "Checks by group: width_rules". Recipes built with expressions can often be used with a different rule file.
      - **Apply checks** — Resolve the selected expression into a list of rule checks and update the recipe definition. Recipes built with a list of rule checks may not be able to be used with a different rule file.

      **Note**

      The recipe definition is not modified until you click either **Apply expression** or **Apply checks**.

   Figure 3-5 shows Steps 3.a-3.d for the "Checks including selected layer" expression, which selects checks by layer.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 3-5. Adding a Recipe Expression**

4. Repeat Step 3 as necessary to add more expressions to the check recipe definition.

5. View the check recipe in the Recipe definition area. You can use the buttons to move an expression up or down in the list, remove an expression, or resolve an expression into the equivalent list of rule checks.

**Tip**

If you want to only run checks that are selected in the rule file when using a custom check recipe, go to the Expression area, select “Checks not selected in the rules file,” enable “Subtraction,” and click the **Apply expression** button. Make sure the expression “- Checks not selected in the rules file” is the last expression in the recipe definition.

6. View the list of included and excluded checks in the Recipe checks area. A **Groups** tab is included if check groups are created in the rule file (the Group statement).



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

A green check mark indicates that the rule check or group is included in the recipe. A red X indicates the item is not included. Click in the checkbox next to an item to include or exclude it from the recipe.

In the **Checks** tab you can right-click for a menu to add and subtract checks. You can also enter a filter pattern to filter the rule checks that are shown in the **Checks** tab.

7. (Optional) Click **Export** to save the check recipe to a file. The recipe file can be imported by other users with the **Import** button. Saved recipes can be imported by Calibre RealTime Custom and Calibre RealTime Digital.
8. Click **OK** to save the recipe and exit the dialog box.

**Tip**

If you edited a built-in recipe, you can click the **Rename** button to give the modified recipe a new name.

9. (Recommended) Choose **File > Save Runset** to save the recipe definition to a runset.

**Tip**

You must either save a runset or export the recipe to a file if you want the new check recipe to be available in later sessions of Calibre Interactive.

### Related Topics

Basic Editing of Custom Check Selection Recipes

Check Selection Recipe Editor Dialog Box

Creating a Custom Check Selection Recipe

Check Selection Recipe

Running Only Checks Selected in the Rule File with a Custom Check Selection Recipe

## Running Only Checks Selected in the Rule File with a Custom Check Selection Recipe

You can subtract the check expression “Checks not selected in the rules file” from a custom check recipe in order to run only checks that are selected in the rule file. This step is necessary because most check expressions do not obey DRC [Un]Select Check and DFM [Un]Select Check statements in the rule file.

### Prerequisites

- Calibre Interactive nmDRC or Calibre Interactive DFM is running.
- A rule file is loaded on the Rules pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Do one of the following to open a recipe for editing:
   - Create a custom check recipe. See "Creating a Custom Check Selection Recipe" on page 110.
   - Open an existing check recipe for editing:
     i. On the Rules pane, choose the recipe to edit in the "Check Selection Recipe" dropdown list.
     ii. Click **Edit**.

   Leave the recipe open in the Check Selection Recipe editor dialog box.
2. Select "Checks not selected in the rules file" in the Expression area.
3. Enable "Subtraction."
4. Click the **Apply expression** button.
5. View the "Recipe definition" area and confirm that "- Checks not selected in the rules file" is the last expression in the recipe definition.

## Examples

## Related Topics

Basic Editing of Custom Check Selection Recipes

Advanced Editing of Check Selection Recipes in Calibre Interactive

Check Selection Recipe Editor Dialog Box



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Check Selection Recipe Editor Dialog Box

To access: Select the Rules pane in Calibre Interactive, then click the **Edit** button to the right of the Check Selection Recipe dropdown list.

The Check Selection Recipe Editor dialog box includes controls to create, copy, save, and edit a check selection recipe. Basic and advanced editing controls are available.

## Description

The Edit Recipe Dialog box has two modes: basic and advanced.

Use the basic controls (Figure 3-6) for the following tasks:

- Include and exclude basic recipe expressions.
- Include and exclude individual checks or check groups.

Use the advanced controls (Figure 3-7) for the following tasks:

- Develop advanced recipe expressions involving selected layers, visible or recently edited layers, rule check comments, and check groups.
- View and edit the recipe definition.

Descriptions of the fields and selections in the dialog box follow the figures.

**Figure 3-6. Check Selection Recipe Editor Basic Controls**

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 3-7. Check Selection Recipe Editor Advanced Controls**

## Objects

**Table 3-9. Check Selection Recipe Editor — Action Buttons**

| Item | Type | Description |
|---|---|---|
| New | Button | Open a new recipe. |
| Import | Button | Open a recipe that was saved to file with the **Export** button.<br>**Tip:** You can specify check recipe files to import at Calibre Interactive startup on the **Startup** tab of the Setup Preferences dialog box (**Setup** > **Preferences**); check the "Import Recipe Files" checkbox and specify the recipe files.<br>You can import a check recipe file exported from Calibre RealTime Custom and Calibre RealTime Digital; however, the recipe expressions involving visible and recently edited layers have no meaning in Calibre Interactive. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-9. Check Selection Recipe Editor — Action Buttons (cont.)**

| Item | Type | Description |
|---|---|---|
| Export | Button | Save a recipe to file.<br>A recipe saved in this manner can be read using the **Import** button. Calibre RealTime Custom and Calibre RealTime Digital can import check recipes saved to file from Calibre Interactive. |
| Copy | Button | Create a copy of the selected recipe. |
| Clear | Button | Clear all expressions from the Recipe definition area and return to the default of "Checks selected in the rules file." |
| Rename | Button | Rename the selected check recipe. |
| Delete | Button | Delete the selected check recipe. |
| Undo | Button | Undo the last modification to the recipe definition. |
| Redo | Button | Redo the last modification to the recipe definition. |

**Table 3-10. Check Selection Recipe Editor — Basic Controls**

| Item | Type | Description |
|---|---|---|
| Recipe | Dropdown list | Specifies the active recipe. |
| Include | Part of Recipe edits pane | Specify checks to include:<br>• **Checks selected in the rules file** — Include checks selected in the rule file with DRC [Un]Select Check statements.<br>• **All checks** — Include all checks in the rule file.<br>• **None** — Do not include any checks. |
| Exclude | Part of Recipe edits pane | Specify checks to exclude:<br>• Checks with connectivity.<br>• Checks with density. |
| Recipe checks | Pane | Displays tabs with the rule checks and check groups in the rule file; the **Groups** tab is only displayed if check groups are defined in the rule file.<br>A green check mark indicates that the rule check or group is included in the recipe. A red X indicates the item is not included. Click in the checkbox next to an item to include or exclude it from the recipe.<br>In the **Checks** tab you can right-click for a menu to add and subtract checks. You can also enter a filter pattern to filter the rule checks that are shown in the **Checks** tab. |
| Check text | Pane | Displays the check text for the selected rule check. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-10. Check Selection Recipe Editor — Basic Controls (cont.)**

| Item | Type | Description |
|---|---|---|
| Advanced | Button | Toggle between basic and advanced editing controls. |

**Table 3-11. Check Selection Recipe Editor — Advanced Controls**

| Item | Type | Description |
|---|---|---|
| Recipe | Dropdown list | Specifies the active recipe. |
| Expression | Pane | The list of available expression categories for the Check Selection Recipe. See the expression categories table for the definition of each expression.<br>The expression view to the right of the expression categories displays the elements you select to make up the expression. Click in the checkbox next to a check name to include or exclude the check from the recipe or right-click for a menu to add and subtract checks. |
| Addition | Selection | Add the expression or equivalent checks to the check recipe. |
| Subtraction | Selection | Subtract (exclude) the expression or equivalent checks from the check recipe. |
| Apply expression | Button | Update the check recipe with the selected expression; for example, “Checks by group: width_rules”. Recipes built with expressions can often be used with a different rule file.<br>The recipe definition is not modified until you click either **Apply expression** or **Apply checks**. |
| Apply checks | Button | Resolve the selected expression into a list of rule checks and update the check recipe. Recipes built with a list of rule checks may not be able to be used with a different rule file.<br>The recipe definition is not modified until you click either **Apply expression** or **Apply checks**. |
| Recipe definition | Pane | Displays the expressions that make up the recipe definition. Use the following buttons to operate on the selected expression:<br>— Move expression up in the list<br>— Move expression down in the list<br>— Delete the expression<br>— Resolve recipe into equivalent list of rule checks |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-11. Check Selection Recipe Editor — Advanced Controls (cont.)**

| Item | Type | Description |
|---|---|---|
| Recipe checks | Pane | Displays tabs with the rule checks and check groups in the rule file; the **Groups** tab is only displayed if check groups are defined in the rule file.<br>A green check mark indicates that the rule check or group is included in the recipe. A red X indicates the item is not included. Click in the checkbox next to an item to include or exclude it from the recipe.<br>In the **Checks** tab you can right-click for a menu to add and subtract checks. You can also enter a filter pattern to filter the rule checks that are shown in the **Checks** tab. |
| Check text | Pane | Displays the check text for the selected rule check. |
| Advanced | Button | Toggle between basic and advanced editing controls. |

**Table 3-12. Expression Categories for Check Selection Recipes**

| Expression Category | Definition and Usage |
|---|---|
| The expression categories appear in the Expression pane in the dialog box, as shown in Figure 3-7 | |
| All checks | All checks in the rule file, except those excluded by preprocessor directives. DRC [Un]Select Check and DFM [Un]Select Check statements are *not* considered.<br>All checks are listed and enabled in the expression view. Click in the checkbox next to a check name to include or exclude the check from the recipe or right-click for a menu to add and subtract checks. |
| Checks selected in the rules file | Checks selected in the rule file. This expression and “Checks not selected in the rules file” are the only expressions that obey the DRC [Un]Select Check and DFM [Un]Select Check statements in the rule file.<br>Note that if the rule file does not include any DRC Select Check statements, then all checks are executed; this is the same behavior as for batch Calibre nmDRC.<br>Checks selected in the rule file are listed and enabled in the expression view. Click in the checkbox next to a check name to include or exclude the check from the recipe or right-click for a menu to add and subtract checks. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-12. Expression Categories for Check Selection Recipes (cont.)**

| Expression Category | Definition and Usage |
|---|---|
| Checks not selected in the rules file | Checks *not* selected in the rule file. This expression and "Checks selected in the rules file" are the only expressions that obey the DRC [Un]Select Check and DFM [Un]Select Check statements in the rule file.<br><br>Checks not selected in the rule file are listed and enabled in the expression view. Click in the checkbox next to a check name to include or exclude the check from the recipe or right-click for a menu to add and subtract checks.<br><br>**Tip:** You can subtract this expression at the end of a recipe definition to ensure that only checks selected in the rule file are included in the recipe. |
| Checks by rule check comments | Choose checks by filtering on the rule check comment.<br><br>Enter a wildcard pattern in the Pattern field. For example, enter *metal1* to have only checks that include "metal1" in the check text comment listed in the Preview pane. Click in the checkbox next to a check name to include or exclude the check from the recipe or right-click for a menu to add and subtract checks.<br><br>See "Rule Comments" in the Calibre Verification User's Manual. |
| Checks selected in recon mode | Choose checks automatically selected in Calibre nmDRC Reconnaissance mode. Requires a license for Calibre RealTime Digital.<br><br>Calibre nmDRC Recon selects checks that identify gross errors early in the design and assembly process. For more information, see "Chip-Level Verification" in the *Calibre Solutions for Physical Verification* manual. |
| Checks selected in recon inverse mode | Choose checks *not* automatically selected in Calibre nmDRC Reconnaissance mode. Requires a license for Calibre RealTime Digital. |
| Checks by group | Choose checks by group name.<br><br>Rule check groups are listed in the expression view. You can expand each group to view the rule checks. Click in the checkbox next to the group name or individual check to enable or disable it. Click the **Preview** tab to view the list of enabled and disabled checks. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-12. Expression Categories for Check Selection Recipes (cont.)**

| Expression Category | Definition and Usage |
|---|---|
| Checks with selected layer only | Choose checks by layer, where the rule check includes *only* the indicated layer.<br>Original layers that are the only layer in a rule check are displayed in the expression view. Layers can be expanded to view the rule checks.<br>Click in the checkbox next to a layer to enable the rule checks that include only the indicated layer or right-click for a menu to select checks. Click the **Preview** tab to view the list of enabled and disabled checks. |
| Checks including selected layer | Choose checks by layer, where the rule check includes the indicated layer and possibly other layers.<br>Original layers that are included in a rule check are displayed in the expression view. Layers that are included in a rule check can be expanded to view the rule checks.<br>Click in the checkbox next to a layer to enable the rule checks that include the indicated layer; rule checks that include other layers in addition to the indicated layer are also enabled. Click the **Preview** tab to view the list of enabled checks. |
| Checks with connectivity | Choose checks that include connectivity.<br>All checks with connectivity are listed and enabled in the expression view. Click in the checkbox next to a check name to include or exclude the check from the recipe or right-click for a menu to add and subtract checks. |
| Checks without connectivity | Choose checks that do not include connectivity.<br>All checks without connectivity are listed and enabled in the expression view. Click in the checkbox next to a check name to include or exclude the check from the recipe or right-click for a menu to add and subtract checks. |
| Checks with density | Choose Density checks.<br>All checks with the Density operation are listed and enabled in the expression view; this includes checks in which the Density operation is used to derive an input layer for the check. Click in the checkbox next to a check name to include or exclude the check from the recipe or right-click for a menu to add and subtract checks. |
| Checks without density | Choose checks that do not include the Density operation.<br>All checks without the Density operation are listed and enabled in the expression view. Click in the checkbox next to a check name to include or exclude the check from the recipe or right-click for a menu to add and subtract checks. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Check Selection for ERC in Calibre Interactive nmLVS

The ERC rule checks that are executed during the Calibre run are selected with the Select Checks dialog box in Calibre Interactive nmLVS. You can also use a customization file to select checks.

## Select Checks Dialog Box

Choose **Setup > LVS Options**, click the **ERC** tab, then click the **Select Checks** button to open the Select Checks dialog box. Selections made in the Select Checks dialog box are not saved to the runset. See these topics for details:


- "Selecting ERC Checks" on page 284
- "Running ERC Checks" on page 282


## Customization File

You can use a customization file to select ERC checks for a Calibre Interactive nmLVS run. This method is useful because customization settings are applied in both interactive and batch runs of Calibre Interactive and can be saved to the runset. See these topics for more information on customization files:


- "Customization Files" on page 587
- "Customization File to Select ERC Rule Checks" on page 651
- "CUSTOM::CHECK" on page 613
- "Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive" on page 604


# Viewing and Editing Text Files in Calibre Interactive

The Calibre Interactive text editor lets you view and edit text files without leaving the application. Most of the files you use with Calibre Interactive are text files; these include control files, runsets, rule files, transcripts, and summary reports.

## Procedure

1. Choose **File > View Text File**.
2. Enter the filename in the File Path field.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Click OK to open the view window.

   By default, the text editor opens in view-only (read-only) mode.

4. (Optional) Enable editing with one of the following:

   - Click the **Edit** button in the bottom of the text editor window. The Edit button is green when editing is enabled and red when editing is not enabled.
   - Choose **Options** > **Edits Are**, and enable **Allowed**.

5. Use the **Edit** and **Options** menus for standard editing functions, to open a search dialog box, and to control the view.

6. If you made changes, choose **File** > **Save** to save the file.

**Note**

If you edit a rule file, you must save the file and reload it in order for the changes to take effect for the next Calibre run.

**Tip**

If you want the text editor to open with editing enabled by default, choose **Setup** > **Preferences**, click the **Misc** tab, and enable "Enable editing for text file viewer."

# Layer Derivations Tree

The layer derivation tree displays the layers found in the rule file and the sequence of derivations that lead up to a particular derived layer. The layer derivation tree also displays the checks and device definitions found in the rule file and the corresponding layer derivations. The layer derivation tree is available for all Calibre Interactive applications.

To access: Click the **Rules** button to display the Rules pane, specify a rule file, and click the ⊞ icon next to "Layer Derivations" to open the tree. Expand the tree as needed.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 3-8. Layer Derivations Tree**

See the following sections for detailed descriptions:

- "Derivations Tree Listings" on page 127
- "Display Conventions for Derivations Tree" on page 128
- "Usage Notes" on page 128

## Derivations Tree Listings

The layer derivations are organized in the following different ways:

- Layers (Alphabetized, *n_layers*)

  Lists all layers alphabetically, where *n_layers* is the total number of layers. Each derived layer can be expanded to see the full derivation.

- Layers (Top-Down, *n_top* / *n_layers*)

  Lists all original and derived layers at the top of the derivation tree; layers at the top of the derivation tree are not used in further layer derivations.

- Layers (Bottom-Up, *n_bottom* / *n_layers*)

  Lists all layers at the bottom of the derivation tree; such layers are not the result of any layer derivation.

- Checks (*n_checks*)

  Lists all rule checks and the derived layer or layers that result from the check. Only present for run types that include checks.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Devices (*n_devices*)

  Lists the devices that are defined in the rule file, along with the layers that are part of the device definition.

A layer that is not part of any layer derivation is listed in both the top-down and bottom-up listings.

## Display Conventions for Derivations Tree

The display conventions for the layer derivation tree are shown in Table 3-13.

**Table 3-13. Display Conventions for Layer Derivations Tree**

| Convention | Definition |
|---|---|
| ◇ | Indicates no further expansion is possible.<br>For original layers, the layer derivation is displayed as follows:<br>"*orig_layer*" = OR "*orig_layer*" = *layer_number*<br>See Usage Notes for additional information. |
| ◎ | Indicates the layer is part of a Connect or Sconnect statement. |
| *rule_check*::<*num*> | Indicates the derived layer output from the check *rule_check*. Some rules have more than one output layer, and this is indicated by *num*. |
| *rule_check*::*layer* | Indicates that *layer* is derived within the check *rule_check*. |
| DEVICE … (*rule_file:line_num*) | Gives the device definition, ending with a link to the definition in the rule file. |

## Usage Notes

For most cases, examining the layer derivations tree in one of the available listings provides the information needed for debugging. The layer derivation tree can also be saved to a text file using the **Save** button if further examination with advanced search tools is needed.

Calibre Interactive does not access the Calibre database constructor when displaying the layer derivation tree. Some layer operations, such as Inside Cell, are processed within the database constructor; this can cause Calibre Interactive to display layers derived using such layer



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

operations with the convention used for original layers. For example, Calibre Interactive displays a layer derived using the Inside Cell operation in the following manner:

## Related Topics

About Rule Files



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Viewing and Editing Environment Variables

Calibre Interactive allows you to quickly view and edit environment variables relevant to your design. The environment variables referenced in your rule file are automatically listed in the Set Environment Variables dialog box. In addition to viewing rule file environment variables, you can view the environment variables defined in the shell session and create and edit runset environment variables.

Calibre Interactive maintains a distinction between rule file and runset environment variables, and between shell and runset values, as summarized here:

- **Rule file environment variable** — One referenced in the rule file.
- **Shell environment value** — The value defined in your shell environment for the environment variable.
- **Runset environment variable** — One created as part of your Calibre Interactive session or saved in a runset.
- **Runset environment value** — The value defined in your Calibre Interactive session or saved in the runset for the environment variable.

For information on using environment variables in GUI fields and runsets, see these topics:

- "Environment Variables in Calibre Interactive GUI Fields" on page 58
- "Calibre Interactive Runset File Format" on page 847

**Video**: How To Set Rule File Environment Variables in Calibre Interactive

The following sections explain the Set Environment Variables dialog box and provide examples for some common tasks:

**Set Environment Variables Dialog Box** . . . . . **131**
**Examples for Creating and Modifying Environment Variables** . . . . . **135**
Setup for Environment Variable Examples . . . . . 135
Adding a Runset Environment Variable Example . . . . . 136
Editing a Runset or Rule File Environment Variable Example . . . . . 137
Changing the Value of a Shell Environment Variable Example . . . . . 139



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Set Environment Variables Dialog Box

To access: Choose **Setup** > **Set Environment** to open the Set Environment Variables dialog box.

The Set Environment Variables dialog box allows you to view shell, runset, and rule file environment variables. You can add and edit runset variables, which are variables defined for the Calibre Interactive session and saved in the runset.

## Description

The Set Environment Variables dialog box has two tabs for viewing environment variables:

- **Set Env Tab** — Displays rule file and runset environment variables. You can also add and edit runset variables, and choose whether to use the shell value, use the runset value, or undefine the variable for the run.
- **Shell Env Tab**— Displays the shell environment variables and their values. You can specify a pattern for which environment variables are displayed.

The two tabs of the Set Environment Variables dialog box are shown in Figure 3-9. The controls for the **Set Env** tab are described in a table following the figure.

The use of the Set Environment Variables dialog box is explained with the examples in the section "Examples for Creating and Modifying Environment Variables" on page 135. The view in Figure 3-9 shows the setup used for the examples, which is described in "Setup for Environment Variable Examples" on page 135.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 3-9. Set Environment Variables Dialog Box**

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Objects

**Table 3-14. Set Env Tab for Set Environment Variables Dialog Box**

| Control or Column | Definition |
|---|---|
| Add | Adds a new runset environment variable. A new row appears in the table; click in the columns Name and Runset Value to enter values. |
| Delete | Deletes the selected environment variable. Only runset environment variables can be deleted (those with an ☒ in the In Rules column). |
| Name | The name of the environment variable. Click in the column to enter the name for a new variable or edit the name of a runset variable. |
| Env. | A green check mark in this column indicates that the value in the Env. Value column is used for the run. Click in the Env. column to select the Env Value for the run. |
| Env. Value | The value defined in the shell. This value cannot be changed. |
| Runset | A green check mark in this column indicates that the value in the Runset Value column is used for the run. Click in the Runset column to select the Runset Value for the run. |
| Runset Value | The value defined in the Calibre Interactive session or runset. Click in the column to enter a value for a new variable or edit an existing value. |
| Unset | A green check mark in this column indicates that the variable is considered undefined for the run. Click in the Unset column to unset the variable for the run. |
| In Rules | ☑ — Indicates a rule file environment variable. When a rule file is loaded, the referenced environment variables are automatically added to the **Set Env**. tab. You cannot change the name of a rule file environment variable or delete it.<br>☒ — Indicates a runset environment variable. Runset environment variables are created and defined in your Calibre Interactive session or saved in a runset. You can change the name and the value of a runset environment variable. |
| OK | Apply changes and close the dialog box. |
| Apply | Apply changes. |

## Related Topics

Environment Variables

CUSTOM::DEFINE



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Customization Files

Environment Variables in Calibre Interactive GUI Fields

Calibre Interactive Runset File Format



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Examples for Creating and Modifying Environment Variables

There are several methods for creating and modifying environment variables in Calibre Interactive. The rule file, shell environment, and GUI setup for the examples is explained first.

The video “How To Set Rule File Environment Variables in Calibre Interactive” demonstrates how to:

- View shell environment variables
- Define environment variables in the runset
- Unset environment variables

**Setup for Environment Variable Examples** . . . . . 135
**Adding a Runset Environment Variable Example** . . . . . 136
**Editing a Runset or Rule File Environment Variable Example** . . . . . 137
**Changing the Value of a Shell Environment Variable Example** . . . . . 139

## Setup for Environment Variable Examples

The examples for using environment variables all use the same rule file and shell environment setup.

The examples and the view shown in Figure 3-9 in the section “Set Environment Variables Dialog Box” on page 131 use the following setup:

- **Rule File**

  The rule file includes the following statements referencing the environment variable DRC_process:

```
#ifdef $DRC_process 7b
   LAYER top_m metal1
#else
   #ifdef $DRC_process 8a
    LAYER top_m metal3
   #else
    LAYER top_m metal2
   #endif
#endif
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Shell Environment**

  The shell environment variables DRC_process and MGC_DRC_output are defined in the shell as follows (using csh):

  **% setenv DRC_process 7b**

  **% setenv MGC_DRC_output lab2_results**

- **GUI Setup**

  In the Calibre Interactive nmDRC GUI, the DRC Results Database File on the Outputs pane is defined as $MGC_DRC_output.

A previous Calibre Interactive run defined the runset environment variable testnum and assigned it a runset value of 2, as seen in Figure 3-9. The process for doing this is given in, "Adding a Runset Environment Variable Example".

# Adding a Runset Environment Variable Example

You can add an environment variable to your runset and set the value using the Set Environment Variables dialog box. Runset variables added in this way can be used in any of the Calibre Interactive GUI fields that accept environment variables.

This example shows how to add the runset environment variable testnum seen in Figure 3-9 in "Set Environment Variables Dialog Box" on page 131.

## Prerequisites

- The shell environment, GUI, and rule file are set up as described in "Setup for Environment Variable Examples" on page 135.

## Procedure

1. Choose **Setup** > **Set Environment** to display the Set Environment Variables dialog box.
2. Click the **Set Env** tab.
3. Click the **Add** button. A new row appears in the table of environment variables.
4. Click in the Name column of the new row and enter testnum for the name of the environment variable, as shown in Example 3-10.

   When you add a new variable, it is initialized with a green check mark in the Runset column, indicating that the Runset value is used in the Calibre Interactive session.
5. Click in the Runset Value column and enter a value for the variable.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

6. Click **Apply** to save the settings.

   When you save your runset (**File** > **Save Runset**), the new runset environment variable is saved with the runset.

**Tip**

To add an environment variable to the rule file, you must edit the rule file. After you save the rule file, click **Load** on the Rules pane to see the new rule file environment variable displayed in the **Set Env.** tab of the Set Environment Variables dialog box. You cannot use the Set Environment Variables dialog box to add an environment variable to be used in the rule file.

### Examples

Adding a runset environment variable is illustrated in the following example:

**Figure 3-10. Adding a Runset Environment Variable**

## Editing a Runset or Rule File Environment Variable Example

You can change the value of an environment variable by editing it in the Runset Value column of the **Set Env**. tab in the Set Environment Variables dialog box. This applies to both rule file and runset environment variables. If you want to use the new value in the Calibre Interactive session, you must make sure that the Runset column has a green box with a check mark to indicate that the runset value is used.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

This example shows how to assign the runset value 8a to the rule file environment variable DRC_process used in the example setup described in, "Setup for Environment Variable Examples" on page 135. This allows you to use a different value for the run without redefining the variable in the shell. You can then easily switch between using the shell value and the runset value in your Calibre Interactive session.

## Prerequisites

- The shell environment, GUI, and rule file are set up as described in "Setup for Environment Variable Examples" on page 135.

## Video

"How To Set Rule File Environment Variables in Calibre Interactive"

## Procedure

1. Choose **Setup** > **Set Environment** to display the Set Environment Variables dialog box.
2. Click the **Set Env** tab.
3. Click in the Runset Value column for the DRC_process environment variable, and enter the value 8a.
4. Click the ☒ icon in the Runset column to change it to a green check mark(☑).

   The green check mark indicates that the runset value is used in the Calibre Interactive session.

   See Example 3-11 on page 139 for an illustration of Steps 3 and 4.
5. Click **Apply** to save the changes.

   Now when you run Calibre, the value 8a is used for the environment variable DRC_process.

   **Tip**

   You can change the runset value for any environment variable listed in the **Set Env** tab and specify that the runset value be used.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Examples

**Figure 3-11. Editing a Runset or Rule File Environment Variable**

## Changing the Value of a Shell Environment Variable Example

You can change the value used for a shell environment variable by assigning it a runset value and specifying that the runset value be used during the Calibre Interactive session.

If the shell environment variable is listed in the **Set Env** tab of the Set Environment Variables dialog box, follow the same procedure described for, "Editing a Runset or Rule File Environment Variable Example" on page 137.

This example assigns a runset value to the shell environment variable MGC_DRC_output. As outlined in, "Setup for Environment Variable Examples" on page 135, this variable is defined in the shell, and is used in the GUI field for the output DRC database filename. The variable MGC_DRC_output is not listed in the **Set Env** tab (see Figure 3-9 in "Set Environment Variables Dialog Box" on page 131), and this example shows how to enter it and assign a runset value.

### Prerequisites

- The shell environment, GUI, and rule file are set up as described in "Setup for Environment Variable Examples" on page 135.

### Video

How To Set Rule File Environment Variables in Calibre Interactive

### Procedure

1. Choose **Setup** > **Set Environment** to display the Set Environment Variables dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Click the **Set Env** tab.
3. Click **Add** at the top right.

   A new row appears in the table of environment variables.
4. Click in the Name column of the new row and enter the name of the shell environment variable MGC_DRC_output.
5. Press Enter on the keyboard after entering the variable name.

   The value defined for the variable in the shell is listed in the Env. Value column.
6. Click in the Runset Value column and enter the new value.
7. Verify that the Runset column has a green check mark (✅), indicating that the runset value is used in the Calibre Interactive session.

   New variables are initialized with a green check mark in the Runset column. However, if this is not the case, click the X in the Runset column to change it to a green check mark.
8. Click **Apply** to save the changes.

**Tip**

To return to using the shell environment value for the variable, click the X in the Env. column to change it to a green check mark.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Examples

This procedure is illustrated in the following example:

**Figure 3-12. Changing the Value of a Shell Environment Variable**

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setup Preferences

You can control many behaviors of the Calibre Interactive GUI with setup preferences. Setup preferences are specified with the Setup Preferences dialog box, which is opened by selecting **Setup** > **Preferences**. Startup Preferences are saved automatically to special setup files, while all other preferences are saved to the runset.

See "Saving Setup Preferences" on page 143 for information on saving preferences.

The Setup Preferences dialog box has the following tabs:

**Table 3-15. Setup Preferences Dialog Box Tabs**

| Tab | Description |
|---|---|
| Startup | Defines startup behavior.<br>See "Startup Preferences" on page 144. |
| Templates | Defines file naming templates.<br>See "Templates for File Naming" on page 148 and "Export from Viewer Preferences" on page 152. |
| Triggers | Specifies trigger scripts.<br>See "Trigger Functions in Calibre Interactive" on page 657. |
| Disable | Disables certain GUI options and sets license timeout behavior.<br>See "Disable Preferences" on page 154 |
| Misc | Defines miscellaneous preferences.<br>See "Miscellaneous Preferences" on page 156 |
| Info | Defines optional descriptive information about the runset.<br>See "Runset Type and Runset Info" on page 45. |
| CTO File<br>(applies to DRC only) | Defines the viewer layer map file to use when creating a CTO file.<br>See ""Creating a Check Text Override (CTO) File Using Calibre Interactive nmDRC" on page 183. |
| HTML Report<br>(applies to DRC only) | Specifies options to use when creating the DRC HTML Report. Creation of the DRC HTML Report is enabled with the "Create HTML Report" checkbox on the Outputs pane of Calibre Interactive nmDRC.<br>See ""Creating a DRC HTML Report from Calibre Interactive nmDRC" on page 185. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Saving Setup Preferences

Most Setup Preferences are saved to the Calibre Interactive runset, while Startup Preferences are automatically saved to special configuration files.

## Procedure

1. Choose **Setup** > **Preferences** and make sure all the preference settings are correct.
2. Click **OK** to close the Setup Preferences dialog box.
3. Choose **File** > **Save Runset** or **File** > **Save Runset As** to save a runset.
4. Exit Calibre Interactive.

## Results

All preferences on the **Templates**, **Triggers**, **Disable**, **Misc**, and **Info** tabs, plus the customization file settings on the **Startup** tab, are saved to the Calibre Interactive runset. See "Creating and Saving a Calibre Interactive Runset" on page 36 for complete information.

Preferences on the **Startup** tab, except the customization file settings, are automatically saved to special setup files when you exit Calibre Interactive, as described in "Configuration Files for Startup Preferences" on page 146.

## Related Topics

About Calibre Interactive Runsets

Startup Preferences

Setup Preferences



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Startup Preferences

You define startup preferences from the **Startup** tab in the Setup Preferences dialog box, which is opened by selecting the **Setup** > **Preferences** menu item. The startup preferences define startup behavior for loading runsets, optional customization file settings, and the Calibre Interactive window position. Startup preferences are saved in special configuration files.

**Startup Tab for Setup Preferences** . . . . . 145
**Configuration Files for Startup Preferences** . . . . . 146



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Startup Tab for Setup Preferences

Access with: **Setup** > **Preferences**, **Startup** tab

The **Startup** tab in the Setup Preferences dialog box defines startup behavior for loading runsets, optional customization file settings, and the Calibre Interactive window position. For Calibre Interactive nmDRC and DFM, it also includes settings for imported check recipes files.

## Description

The runset opening behavior specified on the **Startup** tab is overridden by command line invocation with a runset, or by setting either of the environment variables MGC_CALIBRE_***_RUNSET_LIST and MGC_CALIBRE_***_RUNSET_FILE. See the discussion with "Loading Runsets on Invocation" on page 39.

Except for the customization file and imported check recipe settings, the settings on the **Startup** tab are automatically saved to a setup file each time you exit Calibre Interactive; see Configuration Files for Startup Preferences for more information.

The customization file and imported check recipe settings are saved to the runset.

**Figure 3-13. Startup Preferences Tab**

The customization file and check recipe settings are described in the following table; other settings are intuitive and not described.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Fields

**Table 3-16. Customization and Check Recipe Settings in the Startup Preferences**

| Option | Description |
|---|---|
| Customization File | Enable the checkbox and specify a customization file to use the Customization GUI. |
| Show customization dialog after opening runset<br><br>(Default: true) | Automatically open the Customization Settings dialog box when a runset is loaded. |
| Warn if customization file has no DRC/DFM/ LVS/PERC/PEX settings<br><br>(Default: false) | Display a warning message if the customization file does not have any settings that are valid for the current Calibre Interactive application. |
| Import Recipe Files<br>(DRC and DFM only) | Enable the checkbox and specify check recipe files to load on startup. You save a check recipe file with the **Export** button in the Check Selection Recipe editor dialog box. The listed check recipe files are imported at startup and listed in the USER RECIPES section of the check recipe list. |

### Related Topics

Customization Files

Check Selection Recipe Editor Dialog Box

## Configuration Files for Startup Preferences

Calibre Interactive automatically saves your startup preferences to special setup files, stored in your $HOME directory by default. Separate setup files allow you to define different startup preferences for the different Calibre Interactive applications.

The filenames are the following:

- *.cgidrcdb* for DRC
- *.cgidfmdb* for DFM
- *.cgilvsdb* for LVS
- *.cgipercdb* for PERC



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- *.cgipexdb* for PEX
- *.cgi3dstackdb* for 3DSTACK

These setup files contain the last 10 runsets opened in Calibre Interactive and are used to populate the dropdown list of runsets in the Open Runset dialog box. The setup files also contain the settings from the Startup Runset and Calibre Interactive Window Position sections of the **Startup** tab in the Setup Preferences dialog box (see "Startup Preferences" on page 144).

**Note**

Calibre Interactive also creates a *.cgi<tool>db* file in the current run directory. This file is for internal tool use only, and is not the same as the startup preference file in your home directory.

**Tip**

You can populate the dropdown list of runsets in the Open Runset dialog box using the environment variable MGC_CALIBRE_DRC_RUNSET_LIST; replace DRC with LVS, PERC, PEX, or DFM for other applications. See "Loading Runsets on Invocation" on page 39.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Templates for File Naming

You can use file naming templates to form the names that Calibre Interactive generates. The template preference settings are applied when you invoke Calibre Interactive from a layout or schematic editor. Template settings make use of the cell and view open in the design tool to name input and output files. You define file naming templates on the **Templates** tab of the Setup Preferences dialog box (**Setup** > **Preferences**).

If "Apply Templates" is enabled on the **Templates** tab the template settings are used when you invoke Calibre Interactive from a design tool. The settings formed by templates take precedence over other runset settings for the same fields. The template settings are saved with the runset.

The default behavior for extracted layout files, extracted SPICE netlist files, and output files is to generate the names for these files based upon layout and source top cell names given by the replaceable parameters %l and %s. You can tailor the file naming templates to fit your needs. Also see "Export from Viewer Preferences" on page 152 for important information about the "Export from layout viewer" and "Export from schematic viewer" template settings.

You can use absolute or relative pathnames on **Templates** tab; relative paths are relative to the run directory. The replaceable parameters given in Table 3-17 are supported in template names.

**Table 3-17. Replaceable Parameters in Template Names**

| Parameter | Definition |
|---|---|
| %l | The primary cell in the layout database. |
| %s | The primary cell in the source netlist (used only for LVS, PERC, and PEX runs). |
| %L | The name of the layout library for OpenAccess databases. Click the **Templates** > **Database** sub-tab to set the template. |
| %V | The name of the layout view for OpenAccess databases. Click the **Templates** > **Database** sub-tab to set the template. |
| %S | The name of the source library for OpenAccess databases. Click the **Templates** > **Database** sub-tab to set the template. |
| %W | The name of the source view for OpenAccess databases. Click the **Templates** > **Database** sub-tab to set the template. |
| Replaceable parameters for the output parasitic netlist filename. Click the **Templates** > **Outputs** sub-tab to set the "PEX Netlist" template. | |
| %X | The extraction type:<br>• R+C+CC — rcc<br>• R+C — rc<br>• R — r<br>• C+CC — cc<br>• No R/C — norc |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-17. Replaceable Parameters in Template Names (cont.)**

| Parameter | Definition |
|---|---|
| %x | The output netlist format:<br>• Calibre View — cv<br>• DSPF — dspf<br>• Eldo — spi<br>• HSPICE — sp<br>• Spectre — scs<br>• SPEF — spef |
| %o | The netlist format option:<br>• PRIMETIME — pt<br>• HSIM — hsim<br>• ADITRAIL — ar<br>• Net Geometry — ng |

See the following topics for more information.

- "Example for File Naming Template" on page 149 (includes video)
- "Disabling Templates So That Saved Runset Values Are Used" on page 150 (includes video)
- "Export from Viewer Preferences" on page 152 (includes video)
- "Runset Options for Template Parameters" on page 960
- "Naming the Extracted Parasitic Netlist Based on the Extraction Type and Format" on page 317

## Example for File Naming Template

You can use a template definition to create an output file name based on the top cell in the design. The method for doing this is shown in an example and with a short video.

**Video**: How To Fill in Calibre Interactive GUI Fields Based on the Open Design by Using Templates

For example, suppose you open a design with a top cell of "pllclk" in your design tool, then start Calibre Interactive nmDRC. File naming templates apply because Calibre Interactive was



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

opened from a design tool. The default file naming template for the output database is "%l.drc.results", where %l is replaced with the name of the topcell. When you view the DRC Results Database filename on the Outputs pane, it is set to "pllclk.drc.results".

Figure 3-14 shows the setup for this example. The figure shows the **Templates** > **Outputs** sub-tab of the Setup Preferences dialog box and a section of the Outputs pane for Calibre Interactive nmDRC.

**Figure 3-14. Defining Templates for Generated Filenames**

Template definitions are saved to a runset parameter with "Template" in the runset option name, as in the following two lines from a runset file:

```
*drcTemplate_SF: %l.summary
*cmnTemplate_RL: %l.rules
```

where drcTemplate_SF and cmnTemplate_RL define templates for the DRC summary file and the rule file, respectively. See "Runset Options for Template Parameters" on page 960 for a listing of all runset options for template parameters, along with defaults and the corresponding runset option for the resolved template.

## Disabling Templates So That Saved Runset Values Are Used

You may sometimes see that saved runset settings are not used when you open Calibre Interactive from a design tool. This occurs because templates are used to fill in certain GUI fields based on the open design when you launch Calibre Interactive from a design tool. You



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

can disable templates if you want to use only saved runset options. You can disable all templates or delete the template for a specific GUI field.

## Prerequisites

- Calibre Interactive is open.

## Video

The video demonstrates how to do the following:

- Turn off templates so that Calibre Interactive only uses saved runset values
- Delete the template for a field
- Disable all templates

## Procedure

1. Choose **Setup** > **Preferences** to open the Setup Preferences dialog box.
2. Click the **Templates** tab.
3. Do one of the following to disable templates:

| To … | Do this … |
|---|---|
| Disable all templates | Uncheck “Apply Templates.” |
| Disable a template for a specific GUI field | 1. View the different sub-tabs on the **Templates** tab and locate the setting for the GUI field.<br>2. Clear the text in the entry field for the template. |

4. Click **OK** to close the Setup Preferences dialog box.
5. Make sure all GUI fields have the correct values. You may need to correct GUI fields that were filled in automatically using template settings.
6. Save the runset.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

To insure that runset settings are always used, you may want to check the setting of "Read Inputs/Outputs fields while loading rules file" on the **Misc** tab of the Setup Preferences dialog box. When this setting is enabled, input and output statements from the rule file populate the Inputs and Outputs pane when the rule file is loaded (**Load** button on the Rules pane).

### Related Topics

Templates for File Naming

Runset Options for Template Parameters

# Export from Viewer Preferences

If Calibre Interactive is connected to a layout or schematic viewer, you have the option of specifying the open layout or netlist as the input file for the run; this is called layout or schematic export. You can set a preference for the export setting that is used when Calibre Interactive is invoked from a layout or schematic viewer. The preference setting is on the **Templates** > **Inputs** sub-tab of the Setup Preferences dialog box. This preference setting is not used in batch mode or command line execution.

Figure 3-15 shows the **Templates** > **Inputs** sub-tab of the Setup Preferences dialog box for Calibre Interactive nmLVS. The options for input file templates are different for different run types, and the "Export netlist from schematic viewer" setting is not present for Calibre nmDRC and DFM runs. Table 3-18 explains the possible settings of Yes, No, and Runset for the export preference.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 3-15. Defining Export Preferences in the Setup Preferences Dialog**

**Table 3-18. Export Preference Settings in the Setup Preferences Dialog**

| Setting | Description |
|---|---|
| Export layout from layout viewer<br><br>Default = Yes | If Calibre Interactive is invoked from a layout viewer:<br>• **Yes** — Export from layout viewer, regardless of the runset setting for “Export from layout viewer” on the **Layout** tab of the Inputs pane.<br>• **No** — Do not export from layout viewer, regardless of the runset setting for “Export from layout viewer” on the **Layout** tab of the Inputs pane.<br>• **Runset** — Use the setting from “Export from layout viewer” on the **Layout** tab of the Inputs pane. |
| Export netlist from schematic viewer<br><br>(for LVS, PERC, and PEX only)<br><br>Default = Runset | If Calibre Interactive is invoked from a schematic viewer:<br>• **Yes** — Export from schematic viewer, regardless of the runset setting for “Export from schematic viewer” on the **Netlist** tab of the Inputs pane.<br>• **No** — Do not export from schematic viewer, regardless of the runset setting for “Export from schematic viewer” on the **Netlist** tab of the Inputs pane.<br>• **Runset** — Use the setting from “Export from schematic viewer’ on the **Netlist** tab of the Inputs pane. |

The template settings in Table 3-18 affect the “Export from layout viewer” and “Export from schematic viewer” settings on the **Layout** and **Netlist** tabs of the Inputs pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Video**: The second half of the video “How To Fill in Calibre Interactive GUI Fields Based on the Open Design by Using Templates” shows the export from viewer preferences.

### Related Topics

Templates for File Naming

Using a Viewer as the Source for Design Input

# Disable Preferences

You can disable several Calibre Interactive options and changes to certain settings on the **Disable** tab on the Setup Preferences dialog box (**Setup** > **Preferences**).

**Table 3-19. Disable Preferences**

| Option | Remarks |
|---|---|
| Disable *app* Options | Disables editing of settings on the Options pane. |
| Disable waiver flow controls<br>(DRC/DFM/LVS only) | Disables editing of settings on the **Waivers** tab of the Inputs pane. |
| Disable check selection<br>(DRC/DFM only) | Disables changing the DRC or DFM Check Selection Recipe. |
| Disable check selection recipe editing<br>(DRC/DFM only) | Disables editing of custom check selection recipes. |
| Disable environment changes | Disables editing of environment variables defined in the rule file. |
| Disable customization changes | Disables editing of customization settings.<br>The **OK** button in the Customization Settings Dialog Box is made unavailable. The Customization GUI itself is still active but changes cannot be saved. |
| Disable device signature creation<br>(LVS only) | Disables the **Create Device Signature Creation** selection in the **Setup** menu. |
| Disable ERC changes<br>(LVS only) | Disables controls in the **ERC** tab of the LVS Options pane. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-19. Disable Preferences (cont.)**

| Option | Remarks |
|---|---|
| Disable transcript output | Disables all output to the transcript window except for reports of non-zero exit status.<br>You can choose **Transcript** > **Echo to File** to direct the transcript output to a file; however, you cannot use the other selections in the **Transcript** menu.<br>Note: Transcript output must be enabled when creating device signatures, as described in "Device Signature Creation with Calibre Interactive nmLVS" on page 286. |
| Disable Fast XOR max diff count control<br>(DRC only) | Disables the "Max Diff" control on the Inputs pane for FastXOR runs; this option is on by default, meaning the control is not displayed. Caution should be used when using this option. The "Max Diff" control adds the -maxdiff argument for DBdiff execution with FastXOR; see "FastXOR Database Comparison" in the *Calibre Layout Comparison and Translation Guide* for more information. |
| Disable Fast XOR options | Disables the **Fast XOR Options** tab on the Inputs pane for FastXOR runs. This disable preference is on by default, meaning the tab is not displayed. See "Fast XOR Options Tab" on page 198. |
| Disable Fast XOR generated rules file custom name | Disables the option to specify a custom name for the generated rule file in a FastXOR run. The "Rules File" option is on the **Rule File Generation** tab on the Inputs pane. This disable preference is on by default, meaning the option is not displayed. See "Rule File Generation Tab" on page 196. |
| Disable Fast XOR layer map and base layers combined usage | Enables you to simultaneously specify the Layer Map and Base Layer options on the Rule File Generation tab. By default, these options are mutually exclusive. |
| Release license after: ___ inactive hours. | Sets the time-out period for releasing the application's license; the setting is saved in the runset. The minimum time-out period is five minutes.<br>If the time-out period is set with a higher precedence setting, that value is displayed and cannot be changed; for more information, refer to "License Timeout in Calibre Interactive" on page 67. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Miscellaneous Preferences

The preferences on the **Misc** tab of the Setup Preferences dialog box in Calibre Interactive control warnings and prompts, automatic behaviors, and other options.

Choose **Setup** > **Preferences** and click the **Misc** tab to make changes to the miscellaneous preferences. The preferences are saved with the runset.

**Table 3-20. Miscellaneous Preferences**

| Option | Remarks |
|---|---|
| Automatically load rules file<br>(Default: false)<br><br>Show prompt<br>(Default: true) | When "Automatically load rules file" is enabled, Calibre Interactive automatically loads the rule file at these times:<br>• When you load a runset with a rule file definition.<br>• When you start a Calibre run and the rule file has been modified since it was last loaded.<br>**Show prompt** — Enable or disable displaying a confirmation dialog before automatically loading the rule file. If "Show prompt" is disabled, the setting for "Automatically load rules file" applies during Calibre Interactive batch operation.<br>See the section "Automatically Loading a Rule File" on page 76 for more details.<br>Also see the section "How Rule Files Relate to Control Files" on page 53. |
| Read Inputs/Outputs fields while loading rules file<br>(Default: true) | If enabled, input and output statements from the rule file populate the Inputs and Outputs pane when **Load** is clicked. |
| Run Calibre when runset opened<br>(Default: false) | Start a run immediately after a runset is opened. |
| Warn before overwriting layout database<br>(Default: true) | Recommended. Display a warning before overwriting the system- generated layout database. This occurs when the layout GDSII or OASIS®[1] database is extracted from the layout viewer. |
| Specify control file name<br>(Default: false) | Specifies a control filename. This value is stored in the runset. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-20. Miscellaneous Preferences (cont.)**

| Option | Remarks |
|---|---|
| Write control file suitable for command line run<br><br>(Default: true) | If enabled, the #IFDEF $MGC_CALIBRE_INTERACTIVE line is omitted before the DRC ICSTATION YES line in the control file, along with the corresponding #ENDIF. A control file saved with this option enabled can be used in a command line Calibre run.<br><br>See "Running Batch Calibre with a Calibre Interactive Control File" on page 49.<br><br>If the layout database format is LEF/DEF or OpenAccess, you also need to set database translation options with an environment variable; this is described in the referenced task. |
| Warn before overwriting source database<br>(LVS/PEX only)<br><br>(Default: true) | Display a warning before overwriting the source SPICE netlist. This occurs when the source SPICE netlist is extracted from the schematic viewer. |
| Prompt to save changes before closing runset<br><br>(Default: true) | When enabled, Calibre Interactive displays a prompt dialog when a modified runset is about to be closed. If this setting is not enabled, the prompt dialog is not displayed. |
| Save changed settings before closing runset<br><br>(Default: false) | Controls whether or not the runset is saved. This setting is only available if "Prompt to save changes before closing runset" is disabled. |
| Save runset each time Calibre is run<br><br>(Default: false) | When enabled, the runset is saved automatically each time Calibre is run. |
| Save runset to current directory each time Calibre is run<br><br>(Default: false) | When enabled, each time Calibre is run a copy of the runset is saved to the directory from which Calibre Interactive was invoked.<br><br>If you loaded multiple runsets and this setting is enabled, the runset is saved as *runset.autosaved*. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-20. Miscellaneous Preferences (cont.)**

| Option | Remarks |
|---|---|
| Close RVE when Calibre Interactive closes<br><br>(Default: true) | This option relates only to those Calibre RVE sessions invoked from Calibre Interactive. |
| Close RVE before running … <br><br>(Default: true) | Close the Calibre RVE session before starting a new Calibre run. |
| Display text of included rule files in transcript<br><br>(Default: false) | Displays contents of included rule files in the transcript when Calibre is run.<br><br>Note: This setting takes precedence over the environment variable CALIBRE_ECHO_RULE_FILE. |
| Start RVE as soon as results are available<br><br>(DRC and DFM only)<br><br>(Default: true) | Opens the ASCII results database when the first DRC check is completed instead of waiting until DRC checking completes. This is referred to as "dynamic RVE", since results are updated as they are made available. The option "Show results in RVE" on the Outputs pane must also be enabled.<br><br>This option has no effect in a Calibre Interactive nmDRC run with Calibre Auto-Waivers enabled. |
| Do not open empty main results database in RVE<br><br>(DRC and DFM only)<br><br>(Default: false) | When enabled, an empty main results database (RDB) is not opened if at least one side (auxiliary) RDB will open automatically. If no side RDBs are specified to open automatically, or if all side RDBs are empty, then an empty main RDB is opened.<br><br>Side RDBs can be specified to open automatically with the Calibre Interactive setting "Side RDBs to open" on the Outputs pane or with the Calibre RVE setting "Auto Open" in the "Side RDBs" table. See "Viewing Auxiliary Databases" in the *Calibre RVE User's Manual*.<br><br>This option is automatically disabled if "Start RVE as soon as results are available" is checked. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-20. Miscellaneous Preferences (cont.)**

| Option | Remarks |
|---|---|
| Verify checks are selected before running DRC<br>(DRC only)<br>(Default: true) | When enabled, Calibre Interactive verifies that at least one rule check is selected before starting the run. |
| Stop PEX if LVS indicates mismatch<br>(PEX only)<br>(Default: false) | Stops the parasitic extraction run if the LVS step detects a mismatch between the layout and source designs. |
| Enable editing for text file viewer<br>(Default: false) | Opens text files in edit mode by default. If this setting is not enabled, you must click the red edit box to unlock the text file for editing. |
| Save TVF Rules to SVRF before running Calibre<br>(Default: false) | Enables saving of TVF rules to an SVRF file.<br>See "Running Calibre Interactive PEX" on page 320 for a note on how this option works for PEX runs. Also see "Calibre Interactive Operation with a TVF Rule File" on page 105. |
| Only compile the rules file when Run button is pressed<br>(Default: false) | Prevents Calibre invocation when you click the Run button. Instead, only the rule file compiler is invoked. Any compilation errors are displayed in an error dialog box. If compilation succeeds, a message to that effect is printed in the Transcript. This setting is saved to the runset. |
| Ignore layout and source export in batch mode<br>(Default: true) | This setting is enabled by default, causing batch runs of Calibre Interactive to ignore the settings for "Export from layout viewer" and "Export from schematic viewer" on the Inputs pane.<br>Disable this setting if you want to run Calibre Interactive in batch mode and read the layout or source data from a connected viewer. See "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 3-20. Miscellaneous Preferences (cont.)**

| Option | Remarks |
|---|---|
| Check source netlist is complete before running LVS<br>Check source netlist is complete before running PEX<br>Check source netlist is complete before running PERC<br><br>(Default: true) | Verify that the source netlist is complete before starting the run; the source netlist check is case insensitive. If referenced cells are not defined or any .include files are missing, a warning is issued with the choice to continue or cancel the run. Up to eleven levels of subcircuits in the netlist are checked.<br>The GUI only displays the names of five subcircuits that are used but not defined. Checking stops if eleven subcircuits are used but not defined.<br>If an internal pre-execution trigger is defined, the check for a complete source netlist is done after the trigger executes. See "Internal Triggers in Calibre Interactive" on page 659. |
| Specify waiver setup file name<br>(DRC, LVS, and DFM only)<br><br>(Default: false) | When enabled, the waiver setup file generated by Calibre Interactive during waiver creation or during a waiver run is saved to the specified filename. By default, the generated waiver setup file is named *_waiver_setup*. |
| Start RVE after batch run<br>(Default: false) | When enabled, Calibre RVE is automatically opened after a Calibre Interactive batch run. |
| Warn when library layout file format differs from primary layout format<br>(Default: false) | When enabled, Calibre Interactive displays a warning at runtime if additional layout files on the **Options** > **Database** > **Library** tab are not the same format as the layout file on the Inputs pane. |

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.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 4
# Using Calibre Interactive to Perform DRC

The Calibre Interactive nmDRC Graphical User Interface (GUI) is an interface to Calibre nmDRC. The Calibre Interactive - nmDRC window allows you to specify command line options and select rule file statements that guide Calibre nmDRC.

This chapter assumes you are familiar with the terms and concepts presented in the chapter "Running Calibre Interactive" on page 63. The objectives of this chapter are:


**Workflow with Calibre Interactive nmDRC** . . . . . **161**
**Calibre Interactive - nmDRC Requirements** . . . . . **163**
**Specifying a Rule File and Run Directory for a DRC Run** . . . . . **163**
**Specifying the Checks to Execute in a Calibre Interactive nmDRC Run.** . . . . . **165**
**Specifying the Run Type for a DRC Run** . . . . . **166**
**Inputs for a Calibre Interactive nmDRC Run** . . . . . **168**
**Specifying Automatic Waivers for DRC** . . . . . **173**
**Specifying Outputs for a DRC Run** . . . . . **173**
**Running Calibre Interactive nmDRC** . . . . . **178**
**Specifying Run Control and Licensing Options** . . . . . **179**
**DRC Options in Calibre Interactive** . . . . . **180**
**Performing Area DRC.** . . . . . **189**
**FastXOR Layout Comparison in Calibre Interactive** . . . . . **192**


## Workflow with Calibre Interactive nmDRC

The most common scenario for performing DRC from Calibre Interactive assumes that a runset containing all the necessary settings already exists. Your time is spent running DRC and correcting errors in an interactive verification loop.

A typical workflow is illustrated in Figure 4-1. In this scenario, you spend very little time setting up to run Calibre because you can load the necessary settings from a runset. You can set up Calibre Interactive to load the runset automatically on invocation as described in "Loading Runsets on Invocation" on page 39.




Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 4-1. Interactive Verification for Calibre nmDRC**

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Using Calibre RVE for DRC [Calibre RVE User's Manual]

# Calibre Interactive - nmDRC Requirements

Several requirements must be met before running Calibre Interactive nmDRC.

The following items are required for running Calibre Interactive nmDRC:

- A SVRF rule file
- A layout database
- Licensing and platform requirements as described in "Calibre Interactive System Requirements" on page 26.

# Specifying a Rule File and Run Directory for a DRC Run

In general, it is best to begin setup for a DRC run in the Rules pane. Once you define a rule file, you have the option to use it to populate the other rule-file related fields. Calibre Interactive interprets any relative pathnames as relative to the run directory, which you define on this pane.

### Prerequisites

- "Calibre Interactive - nmDRC Requirements" on page 163



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Click Rules to display the Rules pane.

2. Specify the name of the rule file. You can use the View button to view and edit the rule file.

   Be aware that Calibre Interactive automatically recognizes a rule file with the literal name “rules” and displays the name in green. Because the name is shown in green, you may assume the file has been loaded but it has *not* been; the file must be explicitly loaded using the **Load** button.

3. (Optional) Click the **Load** button to populate the GUI with the settings in the rule file.

   A rule file can also be specified in a runset. After loading that runset, the rule file is included in the control file but its settings will not be loaded into the GUI until you explicitly click **Load**.

   **Tip**

   The setting “Read Inputs/Outputs fields while loading rules file” controls whether or not rule file settings populate the GUI options on the Inputs and Outputs panes when **Load** is clicked. This setting is enabled by default; choose **Setup** > **Preferences** and the **Misc** tab to change the setting. See “Read Inputs/Outputs fields while loading rules file” in Table 3-20 on page 156.

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

4. Specify the run directory name. If the directory you specify does not exist, Calibre Interactive queries as to whether you want to create the directory.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. (Optional) View the layer derivation tree at the bottom of the Rules pane; see "Layer Derivations Tree" on page 126 for information.
6. The typical next step is "Specifying the Checks to Execute in a Calibre Interactive nmDRC Run" on page 165.

### Related Topics

About Rule Files

About Calibre Interactive Runsets

Including Other Rule Files and SVRF Commands in Calibre Interactive

# Specifying the Checks to Execute in a Calibre Interactive nmDRC Run

The Check Selection Recipe specifies the checks that are executed in a Calibre Interactive nmDRC run. You can use a built-in check recipe or define your own check recipe.

### Prerequisites

- "Calibre Interactive - nmDRC Requirements" on page 163 are complete.
- "Disable check selection" is not enabled in the Setup Preferences dialog box on the **Disable** tab.

### Procedure

1. Click **Rules** to display the Rules pane.
2. Specify a rule file; see "Specifying a Rule File and Run Directory for a DRC Run" on page 163 for additional details and a view of the Rules pane.
3. Choose a check recipe using the dropdown list for Check Selection Recipe.

   BUILT-IN RECIPES

   - **Checks selected in the rules file** — (Default) Run only checks selected in the rule file. This is the only built-in recipe that obeys DRC Select Check and DRC Unselect Check statements in the rule file.

     Note that if the rule file does not include any DRC Select Check statements, then all checks are executed; this is the same behavior as for batch Calibre nmDRC.

     If you are using Customization Files to select checks for a Calibre Interactive nmDRC run, you should always use "Checks selected in the rules file" to avoid conflicts between the check recipe and the Customization GUI.

   - **All checks** — Run *all* checks in the rule file, except those excluded by pre-processor directives. DRC [Un]Select Check statements are *not* considered.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

USER RECIPES

- Select a previously defined custom check recipe.

  See "Basic Editing of Custom Check Selection Recipes" on page 111 for information on creating a Check Selection Recipe.

### Results

The checks specified in the check recipe are executed during the Calibre nmDRC run.

### Related Topics

Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive

# Specifying the Run Type for a DRC Run

The run type for a Calibre nmDRC run determines whether Calibre is run in hierarchical or flat mode, as a FastXOR run for Layout versus Layout comparison, or in Autowaiver Creation mode.

### Procedure

1. Click **Inputs** on the left panel of Calibre Interactive nmDRC.
2. In the Run area at the top of the pane, choose from the following selections:
   - **DRC (Hierarchical)** — Run Calibre nmDRC in hierarchical mode, with the -hier command line option.

     When selected, you can also optionally specify to run in Calibre nmDRC Reconnaissance mode, its inverse, or Calibre DRC Analyze:

     - Recon — Runs with the -recon command line option. Automatically selects a subset of DRC rulechecks that identify gross errors early in the design and assembly process.
     - Recon Inverse — Runs with the "-recon inverse" command line option. Runs the subset of checks that Calibre nmDRC Recon mode does not run.

       For more information on Calibre nmDRC Recon, see "Chip-Level Verification" in the *Calibre Solutions for Physical Verification* manual.
     - Analyze — Runs with the -analyze command line option. Creates a DFM database containing error distribution results data for gross analysis of areas in the design that require attention.

       When "Analyze" is enabled, all options in the Outputs page are disabled aside from "Show results in RVE". If "Show results in RVE" is enabled, the summary file opens after the DRC Analyze run completes.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For more information on DRC Analyze, see "Reviewing DRC Analyze Results" in the *Calibre Solutions for Physical Verification* manual.

- **DRC (Flat)** — Run Calibre nmDRC in flat mode, without the -hier command line option.
- **Calibre-CB (Flat)** — Use the Calibre Cell/Block license package. This license package is only used for flat runs.
- **Fast XOR** — Run FastXOR Layout Comparison. See the section "Performing a FastXOR Layout Comparison Run in Calibre Interactive" on page 192 for instructions.
- **Autowaiver Creation** — Generate a waiver shape database for the Calibre Auto-Waivers flow. See "Generating Waiver Shapes with Calibre Interactive" on page 205.

3. The typical next step is specifying layout input data; see "Inputs for a Calibre Interactive nmDRC Run" on page 168.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Inputs for a Calibre Interactive nmDRC Run

Layout input for a Calibre Interactive nmDRC run can be exported from an open design tool or read from file. You can also specify multiple layout inputs.

**Table 4-1. Sources for Layout Data**

| Open Cell in Design Tool | Files on Disk |
|---|---|
| Layout data comes from the active cell in the layout editor. Calibre Interactive prompts the editor to generate an intermediate layout file at run time.<br><br>Check “Export from layout viewer” on the Inputs pane. | Calibre Interactive reads layout data from one or more layout files.<br><br>Uncheck “Export from layout viewer” on the Inputs pane. |

**Tip**

You can set a preference for the setting of “Export from layout viewer” that depends on whether your Calibre Interactive session is connected to a layout viewer. By default, this preference is set so that “Export from layout viewer” is enabled if Calibre Interactive is started from a layout viewer, regardless of the saved runset value for the option. See “Export from Viewer Preferences” on page 152.

**Reading Layout Input from an Open Design Tool in Calibre Interactive nmDRC . . . 168**
**Reading Layout Input from File in Calibre Interactive nmDRC . . . . . 170**
**Specifying Multiple Layouts for a DRC Run . . . . . 172**

## Reading Layout Input from an Open Design Tool in Calibre Interactive nmDRC

You can use the open cell in a your design tool as the layout input for a Calibre Interactive nmDRC run. Enable “Export from layout viewer” on the Inputs pane to use this mode.

### Prerequisites

- Calibre Interactive nmDRC is open and connected to your design tool. See the following topics for information:

  “Invoking Calibre Interactive From a Layout or Schematic Viewer” on page 64

  “Communication with the Design Tool” on page 72

- Your design tool is properly set up to export the design to the layout format you select in Step 3.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Click **Inputs** on the left panel.
2. Click the **Layout** tab.

3. Use the “Format” dropdown button list and select the format of the *exported* layout data; GDSII, OASIS, and LEF⁄DEF are available. See “Export DEF Layout from Design Tool” on page 89 for more information when using LEF⁄DEF input.

   For a full list of supported layout databases, refer to the “Layout Databases” section of the *Calibre Verification User’s Manual*. Also see “Third Party Database Options” on page 80.

4. Enable “Export from layout viewer.”

   This specifies the layout viewer as the input source. The layout format must be GDSII, OASIS, or LEFDEF when “Export from layout viewer” is enabled.

   Batch runs: In addition, you must also disable the setting “Ignore layout and source export in batch mode.” Choose **Setup** > **Preferences** and the **Misc** tab to change the setting. Also see “Specifying Layout and Schematic Export for a Calibre Interactive Batch Run” on page 104 for details on the layout viewer connection for a batch run.

5. Verify the filename for the exported layout database in the “File” field.

   **Caution**

   When “Export from layout viewer” is enabled, the file specified in the File or DEF File field is overwritten if it exists already. Do not enter a tapeout database in the File field, or use the exported layout named in the File field for tapeout or any other use.

   When exporting a layout database from a connected viewer, an intermediate file is created and saved to the file named in the File field. This filename is generated



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

automatically based on the setting on the **Templates** > **Inputs** sub-tab of the Setup Preferences dialog box.

If the filename is red, the file does not exist and Calibre Interactive creates the file. If the filename is green, the file exists and will be overwritten. Enter another filename if you want to keep the existing file.

6. (Optional) If you have multiple layout inputs for your DRC run, choose **Setup** > **DRC Options** then the **Database** > **Library** tab to specify them. See "Specifying Multiple Layouts for a DRC Run" on page 172. The layout format must be GDSII or OASIS.
7. Verify the top cell name; the top cell is determined from the current cell in the layout viewer when you export a layout database.
8. (Optional) Enable the "Area" option to perform DRC checking on only a rectangular area of the layout. See "Performing Area DRC" on page 189 for complete information.
9. (Optional) You can apply waivers during the Calibre nmDRC run. Calibre Auto-Waivers controls are on the Inputs pane in the **Waiver** tab; see "Calibre Auto-Waivers in Calibre Interactive" on page 201.
10. Possible next steps are the following:
    - "Specifying Multiple Layouts for a DRC Run" on page 172
    - "Specifying Outputs for a DRC Run" on page 173

### Related Topics

Layout System [Standard Verification Rule Format (SVRF) Manual]

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Primary [Standard Verification Rule Format (SVRF) Manual]

## Reading Layout Input from File in Calibre Interactive nmDRC

Calibre Interactive can read layout input from database files. Uncheck "Export from layout viewer" on the Inputs pane to use this mode.

### Procedure

1. Click **Inputs** on the left panel.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

If you want to read information for the Inputs and Outputs panes from the rule file, choose **Setup** > **Preferences**, click the **Misc** tab, and enable the "Read Inputs/Outputs fields when loading rules file" option. (This option is enabled by default.) Rule file options that relate to the Inputs and Outputs panes are automatically applied when you click **Load** on the Rules pane.

2. Click the **Layout** tab.

3. Use the "Format" dropdown button list and select the layout format. For formats other than GDSII and OASIS, see the following topics for more information:

   - **LEFDEF** — see "Read LEF/DEF Input from Design Files" on page 88
   - **OPENACCESS** — see "Providing OpenAccess Input" on page 90

   For a full list of supported layout databases, refer to the "Layout Databases" section of the *Calibre Verification User's Manual*. Also see "Third Party Database Options" on page 80.

4. Uncheck "Export from layout viewer" to specify layout database files as the source of the layout. The checkbox "Export from layout viewer" is not displayed if the layout format does not allow layout export.

5. Specify the filename for the layout database in the "File" field.

6. (Optional) If you have multiple layout inputs for your DRC run, choose **Setup** > **DRC Options** then the **Database** > **Library** tab to specify them. See "Specifying Multiple Layouts for a DRC Run" on page 172. The layout format must be GDSII or OASIS.

7. Specify the top cell name.

8. (Optional) Enable the "Area" option to perform DRC checking on only a rectangular area of the layout. See "Performing Area DRC" on page 189 for complete information.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

9. (Optional) You can apply waivers during the Calibre nmDRC run. Controls for Calibre Auto-Waivers are on the Inputs pane in the **Waiver** tab; see "Calibre Auto-Waivers in Calibre Interactive" on page 201.
10. Possible next steps are the following:
    - "Specifying Multiple Layouts for a DRC Run" on page 172
    - "Specifying Outputs for a DRC Run" on page 173

### Related Topics

Layout System [Standard Verification Rule Format (SVRF) Manual]

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Primary [Standard Verification Rule Format (SVRF) Manual]

Setting Database and General Options in Calibre Interactive

# Specifying Multiple Layouts for a DRC Run

If your primary cell has references to other additional database files you may need to load multiple layouts. You specify multiple layouts using the DRC Options pane in the **Database** > **Library** sub-tab.

Calibre Interactive checks the format of the additional database files and only includes files that match the format of the primary layout database.

**Note**

The **Library** sub-tab is not available for FastXOR runs.

### Prerequisites

- Main layout database in GDSII or OASIS format.
- Additional layout files in GDSII or OASIS format. Files compressed with gzip may be used. The additional layout files must be the same format as the main layout database.

### Procedure

1. Click **Inputs**, then click the **Layout** tab.
2. Enter a layout file in the File field of the **Layout** tab.
3. Select GDSII or OASIS for the Format, as appropriate.
4. Choose **Setup** > **DRC Options** to open the DRC Options pane.
5. Click the **Database** tab; then click the **Library** sub-tab.
6. Use the buttons next to "Additional Layout Files" to add the required files.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The additional files are added to the Layout Path statement. Only files that match the format of the primary database are added.

7. If desired, enable the checkbox in the "Golden" column to add the GOLDEN keyword to the layout file specification in the Layout Path statement.

   The GOLDEN keyword is used when doing "golden" cell matching. See the Layout Path statement and the WITH MATCH GOLDEN keyword for the Extent Cell, Inside Cell, and Not Inside Cell statements for further information on the GOLDEN keyword.

8. (Optional) In order to receive a warning if the additional layout files are not the same format as the main layout database, select **Setup** > **Preferences**, click the **Misc** tab, and enable "Warn when library layout file format differs from primary layout format." The warning is displayed when you click the **Run DRC** button.

   Additional layout files that are not the same format as the primary database are ignored.

### Related Topics

Inputs for a Calibre Interactive nmDRC Run

# Specifying Automatic Waivers for DRC

Calibre Auto-Waivers allows you to waive certain results and run Calibre nmDRC with these waivers taken into account. The waiver_flow executable is used to generate waiver shapes and can be run from Calibre Interactive.

Controls for Calibre Auto-Waivers are on the Inputs pane in the **Waiver** tab; see "Calibre Auto-Waivers in Calibre Interactive" on page 201.

# Specifying Outputs for a DRC Run

You specify the results database and format on the Outputs pane. You can also select to have Calibre RVE open automatically and choose side RDBs to view. In addition, you can create a DRC HTML Report and a CTO file.

Be aware that Calibre Interactive creates new directories if you specify new pathnames. If you have the "Read Inputs/Outputs fields when loading rules file" option selected, relevant rule file settings will populate the Outputs pane when you click **Load** on the Rules pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Click the **Outputs** button on the left panel.

2. Check the following information and edit if necessary:

   - **File** — Specify the Results Database filename. Filenames that end in *.gz* are compressed using gzip. Filenames that end in *.bz2* are compressed using pbzip2. If either compression utility (gzip or pbzip2) is not in your path, you are prompted to supply the path.
   - **Format** — Specify the results database format.
     - **ASCII** — You must choose ASCII if you intend to use Calibre RVE to cross-probe the results.
     - **GDSII** or **OASIS** — If you choose GDSII or OASIS output format, you have the option of specifying a prefix and suffix for the cell name.
   - **Report results from pseudocells in** — Select from “User cells,” “User merged,” “Pseudocells,” “Top cell,” or “Combine.” See the tooltip for descriptions.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Specify whether to open results in Calibre RVE after the run is finished:

   If the main DRC results database is in GDS or OASIS format, only side RDBs are opened.

   - **Show results in RVE —** Specify whether to automatically open Calibre RVE when the DRC run completes.
   - **Side RDBs to open —** Click the button to open a dropdown list and select the RDBs you want to open in Calibre RVE. Side RDBs which are empty are not automatically opened.

   Also see these related settings on the **Misc** tab of the Setup Preferences dialog box: “Start RVE as soon as results are available,” “Start RVE after batch run,” and “Do not open empty main results database in RVE.”



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

4. Set other options as desired. The available options depend on the database format (ASCII, GDS, or OASIS), as described in the following two tables.

**Table 4-2. Output Options for All Results Database Formats (ASCII, GDS, OASIS)**

| Option | Description |
|---|---|
| Output cell errors in cell space | Instruct Calibre to use local cell space coordinates when reporting errors. Errors are output in top cell coordinates if this option is disabled. This option is for Calibre nmDRC-H only.<br>This option is on by default in Calibre Interactive. This option is recommended for Calibre Automatic-Waiver. |
| Specify the amount of rule check text | Specify the amount of rule check text that is written to the results database. See "Specifying the Rule Check Text Written to the Results Database" on page 182.<br>Rule file statement: DRC Check Text {{[COMMENTS \| ALL] [RFI]} \| NONE} |
| Check Text File | Specify a results database to write all selected rulechecks and check text as defined by the other output options to. Results are not written to this file. See "Specifying the Rule Check Text Written to the Results Database" on page 182.<br>This option is available if "Specify the amount of rule check text" is enabled. |

**Table 4-3. Output Options for an ASCII DRC Results Database**

| Option | Description |
|---|---|
| Create HTML Report | Create a DRC HTML report after the run completes. This option is only available if the layout format on the Inputs pane is GDSII or OASIS.<br>To set report options, choose **Setup** > **Preferences** and the **HTML Report** tab. See "Creating a DRC HTML Report from Calibre Interactive nmDRC" on page 185. |
| Create CTO file | Create a DRC RVE Check Text Override File (CTO File) during the run.<br>See "Creating a Check Text Override (CTO) File Using Calibre Interactive nmDRC" on page 183. To specify a layer map file, choose **Setup** > **Preferences**, the **CTO File** tab, and enter a filename for "Viewer Layer Map File." |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The options for GDS and OASIS output add keywords to the DRC Results Database statement in the control file.

**Table 4-4. Output Options for a GDS or OASIS Output Database**

| Option | Description |
|---|---|
| Cell prefix | Add the specified string as a prefix to output cell names (the PREFIX keyword). |
| Cell suffix | Add the specified string as a suffix to output cell names (the APPEND keyword). |
| CBLOCK | Embed compressed data in the OASIS results database. (OASIS only) |
| STRICT | Create strict mode output. (OASIS only) |
| AUTOMAP | Output each check to a separate layer. (OASIS only) |
| INDEX | Index the output database. (OASIS only) |
| NOVIEW | Optimize for direct input to Calibre MDP functions, but omit optimizations intended for interactive viewing. (OASIS only) |

Check “Write DRC summary report file” to specify the filename and options for the DRC summary report file. (DRC Summary Report statement)

If Area DRC is run, the summary report is modified to indicate the error filtering that occurred due to eliminating errors in the halo region. The number of filtered results is always written to the transcript. The summary report includes the following information:

- Number of errors filtered out and the final result count in the relevant RULECHECK line in the RULECHECK RESULTS STATISTICS section
- The total FILTERED and FINAL results counts in the final TOTAL DRC Results Generated line

Check “Reusable Hierarchical Database” to specify to save or restore an RHDB (Reusable Hierarchical Database) file. An RHDB is useful in some flows to save time by avoiding the overhead of repeatedly constructing a hierarchical database.

- Save — Saves a proprietary image of the hierarchical database, generated from all input files in the rule file, as an RHDB file to the specified file. The RHDB file is saved with an embedded checksum. This option disables CTO file generation, HTML report generation, Area DRC, and automatic RVE invocation.
- Restore — Restores an RHDB file from the specified file. A checksum is generated from all input files and compared with the checksum in the specified filename. A mismatch between the checksums generates an error.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. At this point, you may have specified all the information you need to run Calibre nmDRC. If so, then click **Run DRC**.

# Running Calibre Interactive nmDRC

Use the following procedure to start a Calibre nmDRC run from Calibre Interactive—nmDRC.

## Prerequisites

- A layout database.
- A SVRF rule file.

## Procedure

1. Enter all necessary information in the Rules, Inputs, Outputs, and Run Control panes.
2. Choose **Setup > DRC Options** and select each of the option tabs to set additional options. All left panel buttons will be green when the necessary information is entered.
3. (Optional) Choose **Setup > Preferences** and click the **Triggers** tab to specify pre- and post-execution trigger functions. See "Trigger Functions in Calibre Interactive" on page 657.
4. Click **Run DRC** on the left panel of Calibre Interactive.

   When the run starts, the view switches to the Transcript pane, and any run output, including errors and warnings, is displayed there.
5. View the transcript pane and examine the results of the run. See "Transcript Pane in Calibre Interactive" on page 99.

   **Tip**

   You can generate and view a control file without running Calibre by holding down the Ctrl key and clicking **Run DRC**. In addition, the calibre command line for the run is written to the transcript pane.

## Results

If an error occurs in Calibre Interactive before the Calibre run is started, the error is displayed in a pop-up window or in the Calibre Quality Agent for Calibre Interactive.

When the Calibre run starts, the view switches to the Transcript pane, and any run output, including errors and warnings, is displayed here. See "Transcript Pane in Calibre Interactive".

If "Show results in RVE" is checked, the results database is opened in Calibre RVE when the run is complete; see "Using Calibre RVE for DRC" in the *Calibre RVE User's Manual*. In addition, the Calibre Interactive GUI may open a report file in a text viewer when the run completes; see "Viewing and Editing Text Files in Calibre Interactive" on page 125.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Specifying a Rule File and Run Directory for a DRC Run

Specifying the Run Type for a DRC Run

Inputs for a Calibre Interactive nmDRC Run

Specifying Outputs for a DRC Run

Run Control Setup and Licensing

DRC Options in Calibre Interactive

Setting Database and General Options in Calibre Interactive

Performing Area DRC

Performing a FastXOR Layout Comparison Run in Calibre Interactive

# Specifying Run Control and Licensing Options

You can control how your Calibre nmDRC job is run by configuring options related to performance, remote and distributed computing, and license acquisition. The controls are on the Run Control pane.

These topics are covered in detail in “Run Control Setup and Licensing”.

### Related Topics

Trigger Functions in Calibre Interactive



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# DRC Options in Calibre Interactive

Calibre Interactive provides access to a wide range of Calibre nmDRC options. Some are available in the **Setup** menu, others from the Inputs and Outputs panes, and others are available through the DRC Options pane.

Some options are similar for all applications—see "Setting Database and General Options in Calibre Interactive" on page 93.

Do one of the following to access the DRC Options pane:

- Choose **Setup** > **DRC Options**. This adds the **DRC Options** button to the left side of the Calibre Interactive nmDRC GUI and opens the tab.
- Click the **DRC Options** button on the left panel if it is present.

**Preserving Cells During DRC** . . . . . 180
**Limiting the DRC Results Count** . . . . . 181
**Specifying the Rule Check Text Written to the Results Database** . . . . . 182
**Reporting DRC Results in Local Cell Space or Top Cell Space** . . . . . 183
**Creating a Check Text Override (CTO) File Using Calibre Interactive nmDRC** . . . . . 183
**Creating a DRC HTML Report from Calibre Interactive nmDRC** . . . . . 185
**Specifying the DFM Defaults Statement** . . . . . 188

## Preserving Cells During DRC

When performing hierarchical DRC, Calibre may expand certain cells for improved performance. The process of expanding a cell removes one level of hierarchy by promoting it into the parent cell. If a DRC error occurs in a cell that gets expanded, it is reported in the parent cell.

If you used Calibre RVE for DRC to waive certain errors from a previous Calibre nmDRC run, it is possible that an error in a waived cell will be reported in a parent cell if the cell with the waived result is expanded. For information on waiving results, see "Waiving, Marking, Sorting, and Grouping DRC Results in Calibre RVE" in the *Calibre RVE User's Manual*.

If you used Calibre RVE to waive cell-resident errors from a prior run and you do *not* want the cells containing those errors to be expanded, you can specify that certain cells are preserved from expansion.

### Prerequisites

- You are running hierarchical Calibre nmDRC.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- You know one of the following in order to specify which cells containing waived results should not be expanded during the Calibre nmDRC run:
  - Names of the cells that contain waived errors.
  - Name of the *.waived* file used by Calibre RVE for DRC. The file is automatically named *<drc_results_database>.waived*, where *<drc_results_database>* is the name of the DRC results database.

### Procedure

1. Click **DRC Options** on the left panel. If the DRC Options button is not present, choose **Setup > DRC Options** from the menu to display it.
2. Click the **Cells** tab.
3. If you wish to preserve all cells that contain waived results in the *.waived* file, enable "Preserve cells from RVE waiver files" and enter a filename in the associated text entry box.

   If you choose this option, the specified *.waived* file is automatically loaded into Calibre RVE for DRC if you start Calibre RVE from your Calibre Interactive session.
4. Enable "Preserve Cells from list" to specify individual cells that are protected from expansion. You can enter cell names in these ways:
   - Enter a space-separated list of cells in the text entry box.
   - Read cell names from an existing waiver file — Click the **Import from RVE Waiver File** button and select a *.waived* file. All the cells that contain waived results are entered into the cell list. You can then delete or add cell names as needed.

     **Note**

     Choosing to preserve cells from automatic expansion may increase run times in some cases.

   Corresponding rule file statements: Layout Cell List and Layout Preserve Cell List.

## Limiting the DRC Results Count

Use this option to control the number of results reported for each DRC rule check.

### Procedure

1. Click DRC Options.
2. Click the Output tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Select the maximum result count in “Max. errors generated per check”:
   - **All** — Report all results. Choose this option if you are using Calibre nmDRC to output a modified version of your layout as mask data
   - **Number** — Enter the maximum allowed result count.

   Corresponding rule file statement: DRC Maximum Results

# Specifying the Rule Check Text Written to the Results Database

Rule check text consists of rule file comments and rule file information, such as the rule file title and pathname. You can use Calibre Interactive to specify the amount of check text written to the ASCII results database.

If you specify DRC rule check comments for RVE in the rule file, you must output check text to the results database. See “DRC Rule Check Comments for Calibre RVE” in the *Calibre RVE User’s Manual*.

Rule file statement: DRC Check Text {{[COMMENTS | ALL] [RFI]} | NONE}

## Prerequisites

- You are running Calibre Interactive nmDRC or Calibre Interactive DFM.

## Procedure

1. Click **Outputs** on the left pane of Calibre Interactive.
2. Select “Specify the amount of rule check text.”
3. Specify the amount of check text. The corresponding keywords for the DRC Check Text statement are given in parentheses.
   - **Comments with RFI** — Output rule check comments (those preceded with an @ symbol) and rule file information. (COMMENTS RFI).
   - **Comments only** — Output rule check comments (those preceded with an @ symbol). (COMMENTS)
   - **All Check Text** — Output the entire text of the each rule check. (ALL)
   - **All Check Text with RFI** — Output the entire text of the each rule check and rule file information. (ALL RFI)
   - **No Check Text** — Do not output any rule check text or rule file information. (NONE)



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

4. (Optional) Select “Check Text File” and specify a file path. When specified, this database contains the rule checks selected for the run and the check text specified by the keyword options. Results are not included in this file.

# Reporting DRC Results in Local Cell Space or Top Cell Space

You can choose to report DRC results in local cell coordinates or top cell space. By default Calibre Interactive nmDRC reports errors in local cell space. This option is for Calibre nmDRC-H only.

## Procedure

1. Click the **Outputs** button on the left panel to display the Outputs pane.
2. Enable or disable “Output cell errors in cell space”:
   - **Enabled** — Use local cell space coordinates when reporting errors; this is the default for Calibre Interactive nmDRC. This option is required for Calibre Auto-Waivers. The following statement is included in the Calibre Interactive control file:

     ```
     DRC Cell Name YES CELL SPACE XFORM
     ```

   - **Disabled** —Use top cell space coordinates when reporting errors. The following statement is included in the Calibre Interactive control file:

     ```
     DRC Cell Name NO
     ```

     **Note**
     The default in command line Calibre runs is DRC Cell Name NO.

# Creating a Check Text Override (CTO) File Using Calibre Interactive nmDRC

Calibre Interactive—nmDRC can create a Check Text Override file to control layer visibility for highlight actions and to define categories for filtering by category. The CTO file is used by Calibre RVE for DRC and DRC HTML reporting. The automatically generated CTO file causes only the layers used in the rule check to be displayed when highlighting a result. Category definitions are optional.

The CTO file includes RVE Show Layers statements based on the rule checks in your DRC rule file. If your design tool uses a layer map file to control the layout export to GDS, the layer map file can be specified so that the layout viewer layer names are used in the CTO file.

Layer and Layer Map statements in the rule file are considered when creating the RVE Show Layers command in the CTO file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Restrictions and Limitations

- **Without a layer map file** — Layer Map statements in the rule file that have open-ended ranges for the layer or datatype are ignored and a warning is issued. The corresponding Layer statement alone is used to determine the layer number for the RVE Show Layers statement.
- **With a layer map file** — See "Layer Map File Usage in Check Text Override (CTO) File Creation" in the *Calibre RVE User's Manual*.

The effect to these restrictions is discussed in the referenced topic in the *Calibre RVE User's Manual*.

## Prerequisites

- A Calibre nmDRC rule file.
- (Optional) A layer map file in one of the formats described in "Layer Map File Usage in Check Text Override (CTO) File Creation".

  Use the layer map file if you want the RVE Show Layers statements in the CTO file to use layer and purpose pairs or layer names rather than layer numbers. The layer map file is also used when generating the Layers category definition; see Step 3.c.

## Procedure

1. Open Calibre Interactive—nmDRC.
2. Specify the rule file and other settings for the run.
3. Click the **Outputs** button.
   a. Select ASCII as the format of the results database.
   b. Enable "Create CTO file."
   c. (Optional) Enable "Generate CTO Categories from" and select Groups, Layers, and/or Recipes to include the selected categories in the CTO file. The category definitions are used to filter by category in Calibre RVE for DRC; see "Filtering DRC Results by Category" in the *Calibre RVE User's Manual*.
4. (Optional) Specify preferences, including a layer map file, using following steps.
   a. Select **Setup > Preferences**.
   b. Select the **CTO File** tab.
   c. (Optional) Specify a layer map file in the "Viewer Layer Map File" text area.

      This step is needed if you use a layer map file to control GDS export from a design tool and want your design tool layer and purpose pairs to be used in the RVE Show Layers statements rather than the GDS or OASIS layer numbers.

      See the "Prerequisites" section for information on layer maps.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

   d. If desired, enable “Warn before overwriting CTO file.”

   e. Click **OK** to close the Setup Preferences dialog box.

5. Click **Run DRC**.

6. Examine the resulting CTO file. If you received any warnings, examine the layers and rules checks affected by the warnings and correct the CTO file if necessary.

   If you are using a layer map file and there are Layer statements in your rule file without a corresponding Layer Map statement, all layer:purpose pairs mapped to the corresponding GDS layer in the layer map file are used in the RVE Show Layers statement for that layer.

### Results

A CTO file with RVE Show Layers statements for each rule check is created. If you specified categories in Step 3.c, the category definitions are included.

The format of the layer parameter in the RVE Show Layers statement is as follows:

- **Layer map file is used** — The layer is given as *layer:purpose*.
- **Layer map file not used** — The layer is given as *layer*, where *layer* is the layer number, or *layer.datatype* if Layer Map statements in the rule file provide a datatype.

Results with and without a layer map file are shown in the “Examples” section of “Layer Map File Usage in Check Text Override (CTO) File Creation” in the *Calibre RVE User’s Manual*

If a results database with the name *results_rdb* is open in Calibre RVE and the created CTO file is named *results_rdb.cto* and in the same directory, then the results database is reloaded and the CTO file is applied.

### Related Topics

Creating a Check Text Override (CTO) File Using Calibre RVE [Calibre RVE User's Manual]

DRC RVE Check Text Override File (CTO File) Format [Calibre RVE User's Manual]

## Creating a DRC HTML Report from Calibre Interactive nmDRC

You can specify to have Calibre Interactive nmDRC automatically create a DRC HTML Report after the Calibre run is finished. You can use a pre-defined configuration file or specify your own configuration file for the report format. Highlight images are automatically included. A shell script is created so that you can run the report at a later time from the command line.

DRC HTML report creation is only possible if the input layout format is GDS or OASIS.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Prerequisites

- Calibre Interactive nmDRC is open. See "Using Calibre Interactive to Perform DRC" on page 161.
- The input layout format is GDS or OASIS.
- (Optional) A Calibre DESIGNrev layer properties file named *<layout>.layerprops* where *<layout>* is the name of the layout specified on the Inputs pane. The file *<layout>.layerprops* is used automatically if it exists.
- (Optional) A DRC RVE Check Text Override File (CTO File) named *<drc_db>.cto*, where *<drc_db>* is the filename of the DRC results database. The file *<drc_db>.cto* should exist in the same directory as the results database. The file is read automatically during the creation of the DRC HTML report. See "DRC RVE Check Text Override File (CTO File) Format" in the *Calibre RVE User's Manual*.

  The CTO file is used to set layer visibility, highlight color, and other options for the rule checks in the results database.

  **Tip**

  If you do not have an existing CTO file, the following procedure describes how to create one automatically with Calibre Interactive.

- (Optional) Set the environment variable MGC_RVE_HTML_BROWSER to the path for a browser executable to specify the HTML browser that is opened when viewing the DRC HTML Report. A recent browser version is recommended, as unexpected results may occur in older browsers. Your default browser is used if this environment variable is not set.

## Procedure

1. Click the **Outputs** button on the left panel of Calibre Interactive nmDRC.
2. Enable "Create HTML Report."
3. (Optional) Enable "Create CTO file" if you do not have an existing CTO file and want highlight images to show only the layers used in each rule check.

   When enabled, Calibre Interactive automatically creates a CTO file with RVE Show Layers statements based on the rule file. The CTO file is used during HTML report creation.
4. (Optional) If you are creating a CTO file and want to use layer names rather than layer numbers, choose **Setup** > **Preferences** and specify a layer map file on the **CTO File** tab.

   See "Creating a Check Text Override (CTO) File Using Calibre Interactive nmDRC" on page 183 for the format of the layer map file.
5. (Optional) Choose **Setup** > **Preferences** to open the Setup Preferences dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

a. Click the **HTML Report** tab.

b. Choose the report configuration with the Template dropdown list:

**Table 4-5. Template Options for DRC HTML Report Creation**

| Template | Description |
|---|---|
| Cell-Check | Group the results tree by Cell/Check. |
| Check-Cell-SC | Group the results tree by Check/Cell/#SC, where #SC is the Shape Class calculated property.<br>Only one result per shape class is shown for each check/cell combination; this is to reduce the size of the report. |
| Check-Cell | Group the results tree by Check/Cell. |
| Check-SC | Group the results tree by Check/#SC, where #SC is the Shape Class calculated property.<br>Only one result per shape class is shown for each check/cell combination; this is to reduce the size of the report. |
| Custom Template | Provide a custom DRC HTML report configuration file; the file is specified in the Custom Template field, which appears when "Custom Template" is selected in the dropdown list. See "DRC HTML Report Configuration File Format" in the *Calibre RVE User's Manual*. |

c. Choose the output format with the Format dropdown list.

The Format option is not available if "Custom Template" is selected.

**Table 4-6. Format Options for DRC HTML Report Creation**

| Format | Description |
|---|---|
| HTML | Standard HTML format. |
| MHTML | Single file HTML format |

d. Specify the output directory in "Output Directory" field. The directory is created if it does not exist.

e. (Optional) Enable "View Report" to have the report opened automatically after the Calibre run finishes.

f. Click **OK** to close the Setup Preferences dialog box.

6. Make other Calibre Interactive nmDRC settings for the run.

7. Click **Run DRC**.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Results

The report is created in the output directory after the Calibre run finishes. The report is named *index.htm* for HTML formatted reports and *report.mht* for MHTML formatted reports. The report is opened automatically if "View Report" was enabled in Step 5.e.

The settings on the Run Control pane are used when creating the DRC HTML report.

The tool also performs the following actions:

- Creates a *report.ini* configuration file in the output directory.
- Creates *rdbSpec.txt* in the output directory if layout snapshots were included in the report. See "DRC HTML Report Database Map File" in the *Calibre RVE User's Manual*.
- Creates a shell script in the run directory; you can execute the shell script to recreate the HTML report at any time. The shell script is named *createReport.sh* if the leaf name of output directory is *report* (the default) or *createReport_<rptDir>.sh* otherwise, where *<rptDir>* is the leaf name of the output directory. The location of the shell script is noted in the Transcript pane.

  If the "View Report" option was enabled in Step 5.e, you can include the "-open" argument when running the script to automatically open the report in a web browser after it has been created.

## Specifying the DFM Defaults Statement

You can specify the statement DFM Defaults RDB {GDS | OASIS} using Calibre Interactive nmDRC. The setting is on the DRC Options pane on the **Database** > **DFM Defaults** sub-tab. You can set default options for the RDB filename, format, cell name prefix and suffix, and the TOP keyword.

### Prerequisites

- Calibre Interactive nmDRC is open.
- (Optional) Set a file-naming template for the DFM Defaults RDB filename. Choose **Setup** > **Preference**, **Templates** tab, **Outputs** tab, and set the "DFM Defaults File" template. See "Templates for File Naming" on page 148.

### Procedure

1. Click the **DRC Options** button on the left panel, or select **Setup** > **DRC Options**.
2. Select the **Database** tab.
3. Select the **DFM Defaults** sub-tab.
4. Uncheck the box "Use settings from rules" in order to use the DFM Defaults RDB statement specified in the GUI.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. Set the options as desired.

| Option | Description |
|---|---|
| File | The filename for the RDB. Required. |
| Format | The RDB format, which must be GDS or OASIS. |
| Transform geometries to the top-level cell | Adds the TOP keyword. |
| Cell prefix | Adds the specified string as a prefix to output cell names (the PREFIX keyword). |
| Cell suffix | Adds the specified string as a suffix to output cell names (the APPEND keyword). |

# Performing Area DRC

You can run DRC checks on only a specified rectangular area of the layout. Area DRC checks the specified area plus a *halo* region around the area. You can specify the halo size and whether results in the halo region are filtered out or kept.

## Prerequisites

- "Calibre Interactive - nmDRC Requirements" on page 163 are met

**Note**

When performing Area DRC in the Cadence Virtuoso integration (version ICADVM18.1 or later), it is recommended that you open the design in Cadence Virtuoso Layout XL or Layout EXL instead of using the Layout Viewer application.

## Procedure

1. Click **Inputs** on the left panel and enter the layout database information as described in "Inputs for a Calibre Interactive nmDRC Run" on page 168.
2. Enable the Area option.
3. Select the area coordinates with one of the following methods:
   - Enter space-separated coordinates for the rectangular window you want to check. Coordinates are the lower-left and upper-right corners of the rectangle, in that order. You must use this method if not connected to a layout viewer.
   - If connected to a layout viewer, click the browse button (...) to select the area you want to check in the layout viewer. Click and drag out an area selection rectangle In



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

the layout viewer. The coordinates of the area selection rectangle appear in the Area field of the Inputs pane.

**Tip**

When connected to a layout viewer, you can right-click the (...) button to highlight and zoom to the specified Area DRC region in the layout.

4. (Optional) Set area DRC options, such as halo size and coordinate scaling, as follows:
   a. Click the **DRC Options** button on the left panel, or choose **Setup** > **DRC Options** if the left panel button is not present.
   b. Click the **Area DRC** tab, then select the desired options described in Table 4-7.

**Table 4-7. Area DRC Options**

| Option | Description |
|---|---|
| Halo width for area DRC | Specifies the size of the halo which expands the area DRC window.<br>• **Automatic** — Width of halo is the greater of the width or height of the specified Area DRC region.<br>• **Size** — Width of halo is an explicit user-specified width in user-units.<br>Note: Depending on the geometry of your area selection, the automatic setting can result in a halo region that is much larger than needed, resulting in slow performance. |
| Remove results from halo region after area-DRC<br>(Only applies to the ASCII format results database.) | Enabled — (default). As described.<br>Disabled — Keep results in the halo region. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 4-7. Area DRC Options (cont.)**

| Option | Description |
|---|---|
| Apply coordinate scaling | Enabled — (default) Scale window coordinates by:<br>`(DRC Results Database Precision/Precision) *(Layout Magnify)`<br>Disabled — Do not apply scaling. |

5. Run DRC, as described in the section "Running Calibre Interactive nmDRC" on page 178.

## Results

The following changes in output occur when Area DRC is enabled:

- **Results database** — Results in the halo region are filtered out based on the setting "Remove results from halo region after area-DRC" on the **Area DRC** tab of the DRC Options pane. Result filtering only applies to ASCII format results databases.
- **Auxiliary databases** — If an auxiliary results database is specified in the rule file, the results are filtered as specified for the main results database.
- **Transcript and summary report output** — If the setting "Remove results from halo region after area-DRC" is enabled on the **Area DRC** tab of the DRC Options pane, then both the transcript and the DRC Summary Report report the number of results that were filtered out of the halo region.

  For example, the following line appears in both the transcript and the summary report:

```
TOTAL DRC Results Generated:     422 (783), FILTERED 397,
   FINAL Results Generated = 25
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# FastXOR Layout Comparison in Calibre Interactive

You can perform a Calibre FastXOR layout comparison using Calibre Interactive.

**Performing a FastXOR Layout Comparison Run in Calibre Interactive** . . . . . 192
**Rule File Generation Tab** . . . . . 196
**Fast XOR Options Tab** . . . . . 198

## Performing a FastXOR Layout Comparison Run in Calibre Interactive

Calibre FastXOR is targeted for comparison between design iterations and can complete XOR comparison many times faster than traditional XOR flows. Calibre FastXOR uses pre-processing by DBdiff to analyze two layout databases to determine the objects that differ between the two databases. Then, Calibre nmDRC-H is used to perform XOR operations on just the objects and layers that differ as determined by DBdiff.

### Prerequisites

- Two complete GDSII, OASIS, LEF/DEF, or OpenAccess layout databases to be compared. The input layouts should not use 64-bit coordinates—the layouts should use 32-bit coordinate space.

  If the layout format is OpenAccess or you are comparing two different LEF files, then a layer mapping file is needed. See "FastXOR Layer Mapping and Comparison" in the *Calibre Layout Comparison and Translation Guide* for instructions on providing a layer mapping file.

- (Optional) An XOR comparison rule file; this is not required if you instruct Calibre Interactive to generate the required XOR rule file.

  The XOR comparison rule file should meet the requirements given in the "Restrictions" section of the *Calibre Layout Comparison and Translation Guide*.

  In addition, the rule file should handle layer mapping correctly so the appropriate layers are compared. For instructions on creating the rule file using DBdiff, see "Generating the LVL Comparison Rule File" in the *Calibre Layout Comparison and Translation Guide*.

- (Optional) Set DBdiff options for FastXOR using the environment variable CALIBRE_FX_DB_DIFF_OPTIONS or the FastXOR template file. See the section "DBdiff Command Execution in Fast XOR" in the *Calibre Layout Comparison and Translation Guide* for details.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

In particular, the -transform option for DBdiff may be useful in some cases. The -transform option specifies transformations to apply to the design specified on the Inputs pane **Layout** tab.

## Procedure

1. Click **Inputs** on the left panel to display the Inputs pane.
2. Choose a run type of "Fast XOR" in the Run dropdown list.
3. Specify the rule file with one of the following methods. You can supply a FastXOR rule file or instruct Calibre Interactive to generate the required XOR rule file before the run.
   - Calibre Interactive generates the required XOR rule file:
     i. On the Inputs pane, enable "Create XOR rules file."

     When this setting is enabled, DBdiff creates the rule file using the -write_xor_rules option. The created rule file is named *rules.fxor*; this filename is entered as the rule file on the Rules pane and the DRC Rules File field is inactive.

     ii. (Optional) Set output and DBdiff options for rule file generation on the Rule File Generation Tab.

     Recommended: "Use Base Layers" in the **Mapping Files** category. This option adds the Layout Base Layer statement with the specified layers to the generated rule file.
   - Specify an existing FastXOR rule file:
     i. Click **Rules** on the left panel to display the Rules pane.
     ii. Enter the FastXOR comparison rule file in the "DRC Rules File" entry field. Make sure the rule file meets the requirements given in the "Prerequisites" section.
     iii. (Optional) Click **Load** to populate the GUI with the settings in the rule file.

     *After you load a rule file, any information you specify in the GUI supersedes the information in your loaded rule file.*
4. Click **Rules** on the left panel to display the Rules pane, then specify the run directory in the "DRC Run Directory" field.
5. Specify the current (modified) layout as follows:
   a. Click **Inputs** on the left panel to display the Inputs pane, then click the **Layout** tab. Do one of the following, depending on the source of the layout:
      - Layout is read from file — Enter the layout database filename in the File field and disable "Export from layout viewer."



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Layout is exported from attached viewer — Enable "Export from layout viewer" and verify the filename for the extracted layout database in the File field.

  When exporting a layout database from a connected viewer, an intermediate file is extracted and saved to the file named in the File field. This filename is generated automatically based on the setting on the **Templates** > **Inputs** sub-tab of the Setup Preferences dialog box. Also see "Using a Viewer as the Source for Design Input" on page 72.

  If the filename is red, the file does not exist and Calibre Interactive creates the file. If the filename is green, the file exists and will be overwritten. Enter another filename if you want to keep the existing file.

  **Caution**

  When "Export from layout viewer" is enabled, the file specified in the File field is overwritten if it exists already. Do not enter a tapeout database in the File field, or use the extracted layout named in the File field for tapeout or any other use.

  For batch runs you must also disable the setting "Ignore layout and source export in batch mode." Choose **Setup** > **Preferences** and the **Misc** tab to change this setting. Also see "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104 for details on the layout viewer connection for a batch run.

b. Select the database format in the Format dropdown list. Only GDSII, OASIS, LEF/DEF, and OpenAccess formats are allowed.

   If the layout format is OpenAccess or you are comparing two different LEF files, then a layer mapping file is needed, as mentioned in the "Prerequisites."

c. Enter the top cell name.

6. Click the **Layout2** tab on the Inputs pane and specify the reference (previous) layout as described in the previous step.

   This layout is compared to the layout specified on the **Layout** tab.

7. (Optional) Set database options as described in "Specifying Third Party Database Options in Calibre Interactive" on page 94.

8. (Optional) Check Calibre Interactive file template settings.

   Choose **Setup** > **Preferences**, then the **Templates** tab and check the file naming templates on each tab. In particular, you may want to change the default templates for the output files.

   See "Templates for File Naming" on page 148.

9. Click **Outputs** on the left panel and check the settings on the Outputs pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

See "Specifying Outputs for a DRC Run" on page 173 if needed.

10. Do the following to set additional options for the run, such as the DBdiff report file, template file, and compare text option.

    a. Select **Setup** > **Preferences**, the **Disable** tab, and clear the checkbox for "Disable Fast XOR options."

    b. Click the **Inputs** button, select the **Fast XOR Options** tab, and set the desired options. See "Fast XOR Options Tab" on page 198.

11. Choose **Setup** > **DRC Options** and check the settings on the DRC Options pane.

    See "DRC Options in Calibre Interactive" on page 180 if needed. The selections on the **Connect** tab are not valid for a FastXOR run.

12. Click **Run Control** on the left panel and check the settings on the Run Control pane.

    See "Run Control Setup and Licensing" on page 681 if needed.

13. Click **Run DRC** on the left panel.

## Results

The transcript output from the Calibre run appears in the Transcript window as the run progresses. Calibre RVE for DRC opens when the run finishes if you enabled "Show Results in RVE" on the Outputs pane.

If your FastXOR comparison rule file was created using DBdiff, then the following output files are specified in the rule file, where *rules.fxor* is the name of the FastXOR comparison rule file:

- DRC Summary Report file: *rules.fxor.summary*
- ASCII DRC Results Database: *rules.fxor.asc*
- Geometric output database: *rules.fxor.gds*

If you clicked **Load** on the Rules pane to load the rule file settings into the GUI and did not later change the output filenames in the GUI, then these are the files created by a FastXOR run when using the default XOR comparison rule file created by DBdiff.

However, if you did not load the rule file settings into the GUI, then the filenames in the GUI for the DRC Summary Report and DRC Results database take precedence over the filenames found in the rule file. If you started Calibre Interactive from a layout editor and are using file naming templates, then the output filenames are formed automatically using the templates.

For more detail about the output from a FastXOR run, see the "Results" section in the topic "Using FastXOR to Perform LVL" in the *Calibre Layout Comparison and Translation Guide*.

## Related Topics

Templates for File Naming



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Rule File Generation Tab

Access with: The **Rule File Generation** tab on the Inputs pane (only available when "Create XOR rules file" is enabled for the "Fast XOR" run type in Calibre Interactive nmDRC)

The **Rule File Generation** tab for FastXOR allows you to specify output file and DBdiff options when the FastXOR rule file is created by Calibre Interactive.

## Objects

**Table 4-8. Format Options for FastXOR Rule File Generation**

| Option | Description |
|---|---|
| Result Format | Select any combination of ASCII, GDS, and OASIS for the format of the output results database(s). |
| Rules File | Specifies the name of the generated rule file. The default is *rules.fxor*.<br>This option is not displayed by default. To display the option, select **Setup** > **Preferences**, click the **Disable** tab, clear the checkbox for "Disable Fast XOR generated rules file custom name," and click **OK** to close the Setup Preferences dialog box. |
| Output Files Prefix | Enter a prefix for the created output files. See the "prefix" option in the DBdiff -write_xor_rules syntax.<br>Note: The prefix specified here does not override output filenames specified on the Outputs pane of Calibre Interactive, since GUI filenames take precedence over output filenames specified in the rule file. |
| Add New and Missing shapes rules | Adds checks in the FastXOR rules file which use the NOT operation to identify new and missing shapes. |
| Common Only | When enabled, only compare layers that are common to both designs. |

**Table 4-9. Mapping Files Options for FastXOR Rule File Generation**

| Option | Description |
|---|---|
| Use Layer Map File | Specifies a layer mapping file. See the -layermap argument for -write_xor_rules option in DBdiff. |
| Use Base Layers | Specifies layers for the -base_layers argument for the -write_xor_rules option in DBdiff. This option adds the Layout Base Layer statement with the specified layers to the generated rule file. |
| Use Include Rules File | Specifies a file containing SVRF statements that are imported directly into the generated FastXOR rule file. See the -include_rules argument to DBdiff. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 4-9. Mapping Files Options for FastXOR Rule File Generation (cont.)**

| Option | Description |
|---|---|
| Use Include Layers File | Specifies a file listing layers to compare. See the -include_layer argument to DBdiff. |
| Use Exclude Layers File | Specifies a file listing layers to exclude from comparison. See the -exclude_layer argument to DBdiff. |

**Note**

The Use Layer Map File and Use Base Layers options are mutually exclusive by default. To use both options simultaneously, select **Setup** > **Preferences**, click the **Disable** tab, clear the checkbox "Disable Fast XOR layer map and base layers combined usage," and click **OK** to close the Setup Preferences dialog box.

## Related Topics

Performing a FastXOR Layout Comparison Run in Calibre Interactive



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Fast XOR Options Tab

To access: 1) Select **Setup** > **Preferences**, click the **Disable** tab, and clear the checkbox for "Disable Fast XOR options." 2) Click the **Inputs** button and select the **Fast XOR Options** tab.

The **Fast XOR Options** tab includes settings for the report file, RDB file, template file, and other options for a FastXOR run in Calibre Interactive.

**Note**

If an option is set with the environment variable CALIBRE_FX_DB_DIFF_OPTIONS or CALIBRE_FX_PASS_THROUGH_LAYER_ENABLE, the GUI setting reflects the environment variable setting and the GUI control is not active.

If you use an existing FastXOR rule file for the run and include the rule file statement SVRF DBdiff Options to set FastXOR options, those options are *not* used in a run started from Calibre Interactive. When using Calibre Interactive, the DBdiff options must be set in the GUI or with CALIBRE_FX_DB_DIFF_OPTIONS.

## Objects

| Setting | Description (command-line option) |
|---|---|
| Report File | Specifies the name of the DBdiff report file.<br>(-report) |
| RDB File | Specifies the name of the RDB output file with optional filter settings. Enter the following:<br>*filename* [FC \| FP \| FT] [-sortlayer]<br>FC — exclude cell differences<br>FP — exclude polygon differences<br>FT — exclude text differences<br>-sortlayer — generate output on a sorted layer basis<br>(-rdb) |
| Template File | Specifies a template file that contains command line options for DBdiff.<br>(-template) |
| Compare Text | Enables text comparison.<br>(-comparetext) |
| Compare All Placed Cells | Specifies to compare all cells that are referenced in the primary cells in both designs. By default, if a child cell exists in both designs but is placed in different parent cells, the child cell is reported as a missing or new instance in the parent cell, but the child cell is not compared.<br>(-compareallplacedcells) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Setting | Description (command-line option) |
|---|---|
| Enable Pass-Through Layers | Sets the environment variable CALIBRE_FX_PASS_THROUGH_LAYER_ENABLE, which improves performance if one or more layers between the compared designs has been added, significantly modified, or removed before a FastXOR comparison. Setting this option instructs FastXOR to pass unique layers directly to the traditional DRC XOR engine, bypassing unnecessary DBdiff operations. |

**Note**

See "Layout Comparison Command Line Syntax" and "FastXOR Database Comparison" in the *Calibre Layout Comparison and Translation Guide* for more information.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 5
# Calibre Auto-Waivers in Calibre Interactive

When verifying a design, some rule check violations may be waived by the foundry. A waived result is a violation that is considered acceptable for the design and process. Designers often want to eliminate waived errors from their Calibre results so they can focus on the errors that need to be fixed. You can perform many of the tasks related to Calibre Auto-Waivers using Calibre Interactive.

Calibre provides the followings two methods for managing waivers:

- Calibre RVE Waived Results

  Results can be waived using the Calibre RVE interface; results can be waived individually, by check, or by cell. Waived results are stored in a *drc_db.waived* file.

  This flow is simple and easy to use, but hierarchy and cell name changes are difficult to handle. See "Waived Results in Calibre RVE for DRC" in the *Calibre RVE User's Manual* for more information.

- Calibre Auto-Waivers

  Calibre Auto-Waivers manages waivers by generating waiver shapes that are merged with your design so that waivers can be applied automatically during a verification run. These waivers take into account hierarchical polygon interactions, as well as interactions of multiple waivers that may apply to a single result.

  See the *Calibre Auto-Waivers User's and Reference Manual*.

See "Calibre Auto-Waivers Versus the .waived File Method" in the *Calibre Auto-Waivers User's and Reference Manual* for a more detailed comparison of these two methods.

You can run Calibre Auto-Waivers in Calibre Interactive. You should be familiar with Calibre Auto-Waivers and Calibre Interactive.

**Introduction to Calibre Auto-Waivers in Calibre Interactive** . . . . . 202
**Requirements for Calibre Auto-Waivers** . . . . . 203
**Calibre Auto-Waivers Flow in Calibre Interactive and Calibre RVE** . . . . . 204
**Generating Waiver Shapes with Calibre Interactive** . . . . . 205
**Starting a Waiver Run in Calibre Interactive** . . . . . 209



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Introduction to Calibre Auto-Waivers in Calibre Interactive

Calibre Auto-Waivers automates the process of applying waivers during design verification.

Calibre Auto-Waivers can be used to waive the following types of results:

- Calibre nmDRC results
- ERC results from a Calibre nmLVS run
- CFA (Critical Feature Analysis) results in Calibre DFM (Design for Manufacturability).

You can use Calibre Interactive to do the following:

- Read in an existing waiver setup file and save a waiver setup file
- Read waiver setup parameters from a runset and save waiver setup parameters to a runset
- Adjust waiver setup parameters
- Run the waiver_flow tool to create waiver shapes
- Start a verification run with waivers

Since the Calibre Interactive controls for specifying Calibre Auto-Waivers are similar for each of the supported applications, the instructions in this chapter apply to all supported applications.

See the topics in the following table for additional information on Calibre Auto-Waivers.

**Table 5-1. Calibre Auto-Waivers Information Sources**

| **Section** (*manual*) | **Description** |
|---|---|
| "Getting Started With Calibre Auto-Waivers" (*Calibre Auto-Waivers User's and Reference Manual*) | General information and getting started procedures. |
| "Calibre Auto-Waivers Overview" (*Calibre Auto-Waivers User's and Reference Manual*) | General overview. |
| "ERC Automatic Waivers" and "DFM Automatic Waivers" (*Calibre Auto-Waivers User's and Reference Manual*) | Summary information for ERC and DFM-CFA waivers. |
| "Calibre PERC Waiver Flows" (*Calibre PERC User's Manual*) | The PERC waiver process, which is separate from Calibre Auto-Waivers. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Requirements for Calibre Auto-Waivers

In addition to the standard requirements for the Calibre application, there are additional licensing, rule file, and input file requirements when running Calibre Auto-Waivers with Calibre Interactive.

The requirements are as follows:

- A Calibre Auto-Waivers product license, plus other licenses as described in "Requirements for Running Calibre Auto-Waivers" in the *Calibre Auto-Waivers User's and Reference Manual*.
- A rule file that meets the requirements described in "Requirements for Running Calibre Auto-Waivers" in the *Calibre Auto-Waivers User's and Reference Manual*.
- A design in a format supported by Calibre Interactive: GDS, OASIS, LEF/DEF, or OASIS.
- (Optional) A waiver setup file. See "Writing the Waiver Setup File for Waiver Verification" and "Writing the Waiver Setup File for Waiver Generation" in the *Calibre Auto-Waivers User's and Reference Manual*. Calibre Interactive creates the waiver setup file at runtime based on the current GUI settings. You can read in an existing waiver setup file to populate the GUI settings, or specify the waiver setup parameters using the GUI.
- The waiver descriptions.
  - A waiver criteria file and a waiver cells file. The waiver criteria file is required for both waiver generation and the waiver run; the waiver cells file is only required for waiver generation. See "Waiver Description Files" in the *Calibre Auto-Waivers User's and Reference Manual*.

  If you are doing a waiver run, you have an additional option for providing the waiver criteria:

  - A waiver geometry file that has waiver criteria embedded in it. In this case "Use criteria from the waiver shape files" must be selected on the Inputs pane, **Waivers** tab, **DRC Run** category.

    The waiver_flow tool always adds the waiver criteria text object to the waiver shapes. If you are using Calibre RVE to generate the waiver shape database, then "Annotate waiver cells with waiver criteria text" must be enabled. See "Text Object Annotations" in the *Calibre Auto-Waivers User's and Reference Manual*.
- The waiver shapes if performing a Calibre verification run with waivers applied. The waiver shapes can be embedded in the design or exist in a separate database.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Auto-Waivers Flow in Calibre Interactive and Calibre RVE

Calibre Auto-Waivers can be used with Calibre Interactive and Calibre RVE. You can generate waivers, set many options, and start a waiver run.

**Figure 5-1. Calibre Auto-Waivers Flow in Calibre Interactive and RVE**

## Related Topics

Requirements for Calibre Auto-Waivers

Waiver Shape Generation with Calibre RVE [Calibre Auto-Waivers User's and Reference Manual]

Generating Waiver Shapes with Calibre Interactive

Starting a Waiver Run in Calibre Interactive

waiver_flow [Calibre Auto-Waivers User's and Reference Manual]

drc_waiv_asc2gds [Calibre Auto-Waivers User's and Reference Manual]

Physical Verification with Waivers [Calibre Auto-Waivers User's and Reference Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Generating Waiver Shapes with Calibre Interactive

You can use Calibre Interactive to generate a waiver shape database for the Calibre Auto-Waivers flow. All the waiver setup parameters are controlled through Calibre Interactive, then the waiver_flow tool is used to create the waiver database. The waiver cells file defines the cells from which waivers are generated and the rule checks to waive.

All settings on the **Waivers** tab of the Inputs pane are saved to the runset (see "About Calibre Interactive Runsets" on page 33). If you have a runset loaded with desired settings, you can skip to the end of the following procedure.

**Tip**

Calibre RVE can also be used to create a waiver database. Calibre RVE is often preferable to using the waiver_flow tool because individual results can be waived while you examine the result database. Only the waived results are exported to the waiver database. See "Waiver Shape Generation with Calibre RVE" in the *Calibre Auto-Waivers User's and Reference Manual* for more information.

## Prerequisites

- "Requirements for Calibre Auto-Waivers" on page 203
- You are running Calibre Interactive nmDRC, Calibre Interactive nmLVS with ERC, or Calibre Interactive DFM with Critical Feature Analysis (CFA).

## Procedure

1. Click **Rules** on the left panel of Calibre Interactive and specify the rule file and run directory.
2. Click **Inputs** on the left panel of Calibre Interactive and specify information for the layout database on the **Layout** tab. The layout format must be GDSII or OASIS for Calibre Auto-Waivers.
3. Choose a run type of "Autowaiver Creation" in the dropdown list.

   The label of the Run button on the left panel changes to **Create Autowaivers**.
4. (Optional) Specify the name of the generated waiver setup file.

   A waiver setup file is generated by Calibre Interactive based on the GUI parameters and named *_waiver_setup_* by default. To specify a different file name, choose **Setup** > **Preferences**, click the **Misc** tab, enable the option "Specify waiver setup file name," and specify the file name.
5. Click the **Waivers** tab on the Inputs pane; a view of the **Waivers** tab is shown in Figure 5-2 for Calibre Interactive nmDRC.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 5-2. Waivers Tab for Waiver Creation**

6. (Optional) If you have a waiver setup file already created and want to use it for waiver shape generation, click the **Import Waiver Setup File** button and select the file. This populates the GUI fields with the values in the selected waiver setup file; only GUI settings specified in the imported file are overwritten.

   Use the following steps to confirm that the imported waiver setup parameters are correct. The waiver setup parameter that corresponds to the GUI control is given in parentheses.

7. Specify the Waiver Criteria File(s) in the Description Files pane (see Figure 5-2). Click the button to edit the file in a text editor. See "Waiver Setup File Format for Waiver Generation" in the *Calibre Auto-Waivers User's and Reference Manual* for complete information. (WAIVER_CRITERIA)

8. (Optional) Specify the directory for temporary files in the Temp Files Directory field. The default is the working directory. (TMP_DIR)

9. Choose the category **Waiver Shapes** and specify the waiver shape file.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

a. Specify a merged or standalone waiver database:

   - **Merge waivers with existing design** *ON* — Embed waiver shapes in the design specified on the **Layout** tab on the Inputs pane. (MERGE YES)
   - **Merge waivers with existing design** *OFF* — Create a waiver database that contains only waiver geometry. (MERGE NO)

b. Specify the name of the output waiver database file in the External waiver File(s) field. This entry is optional—the default output waiver database filename is *waived.gds*. (WAIVER_DATABASE)

c. (Optional) Select the setting for "Use rule file precision":

   - **Enabled** — (default) Specifies PRECISION_CONVERSION YES in the waiver setup file.
   - **Disabled** — Specifies PRECISION_CONVERSION NO in the waiver setup file.

   This setting controls how waiver shape precision and magnification is handled if the rule file precision differs from the layout database precision; it has no effect if the precisions are the same. See "Waiver Cell Precision and Magnification Considerations" in the *Calibre Auto-Waivers User's and Reference Manual* for details. (PRECISION_CONVERSION)

10. Choose the **Waiver Creation** category and specify settings for the following options:

- **Waiver Cell File(s)** — Specify one or more Waiver Cell files. Click the button to open the Edit Waiver Cells File dialog box; you can create a waiver cells file by entering a new filename, then clicking . Select "Set IP_MATCH for added cell" to automatically enable IP_MATCH when adding new cells. See "Waiver Cells File Format" in the *Calibre Auto-Waivers User's and Reference Manual* for complete information. (WAIVER_CELLS)
- **Summary File** — Specify a name for the waiver summary report file. The summary file is always created and named *waiver.summary* by default if you do not specify a filename. (WAIVER_SUMMARY)
- **Text Magnification** — Specify a magnification factor for texts written during waiver creation (waiver_flow). Text magnification only applies when the output format is GDS. The default is 0.005. (TEXT_MAG)
- **Magnify Text Spacing** — Magnify the spacing between text written during waiver creation; this only applies when the output format is GDS. The magnification level specified for the "Text Magnification" option is used; the default of 0.005 is used if "Text Magnification" is not specified. (MAGNIFY_TEXT_SPACING)
- **Place waiver cells in intermediate cells** — Place waiver cells in intermediate cells instead of the parent cell. See the section "Intermediate Container Cells" in the



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

*Calibre Auto-Waivers User's and Reference Manual*.
(ADD_WAIVER_HIERARCHY)

11. Choose the **Waiver Creation > Layer Operations** category in the navigation panel, and specify active and inactive layer operations. (NOT IGNORE and IGNORE)

    Also see the section "SVRF Command Support" in the *Calibre Auto-Waivers User's and Reference Manual*.

12. Choose the **Layers and Text** category and specify layer numbers and datatypes for the waiver shapes and text annotations, if desired.

    See "Waiver Setup File Format for Waiver Generation" in the *Calibre Auto-Waivers User's and Reference Manual* for a description of the options and their defaults; also see "Text Object Annotations" in the same manual.

13. (Optional) Click the **View Waiver Setup File** button to view the waiver setup parameters. You can choose **File > Save As** to save the file.

14. (Optional) Choose **File > Save Runset** to save the runset.

15. Click the **Create Autowaivers** button on the left panel to create new waivers.

    This step runs the waiver_flow tool with the specified parameters.

## Results

The following files are created:

- **Waiver setup file** — A waiver setup file is generated by Calibre Interactive based on the GUI parameters and named *_waiver_setup_* unless you specified a different filename in Step 4.

  The RULE_FILE, WORKING_DIRECTORY, INPUT_LIBRARY, and LAYOUT_SYSTEM parameters are taken from the GUI settings on the Rules pane and the **Layout** tab of the Inputs pane.

- **Waiver shape file** — The filename is given on the Waiver Shapes pane, or *waived.gds* by default.

- **Waiver summary file** — The filename is given on the Waiver Creation pane, or *waiver.summary* by default.

See "Generating Waivers with the waiver_flow Tool" and "Waiver Cell Description" in the *Calibre Auto-Waivers User's and Reference Manual* for complete information.

## Related Topics

Creating Embedded Waiver Shapes Using Calibre RVE [Calibre Auto-Waivers User's and Reference Manual]

Creating Non-Embedded Waiver Shapes Using Calibre RVE [Calibre Auto-Waivers User's and Reference Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Exporting Waiver Shapes to a Cadence Virtuoso Database Using Calibre RVE [Calibre Auto-Waivers User's and Reference Manual]

Getting Started With Calibre Auto-Waivers [Calibre Auto-Waivers User's and Reference Manual]

# Starting a Waiver Run in Calibre Interactive

You can use Calibre Interactive to start a Calibre verification run with Calibre Auto-Waivers. The waiver setup parameters are specified on the **Waivers** tab of the Inputs pane. The waiver run adds the "-waiver" option to the Calibre command line.

All settings on the **Waivers** tab are saved to the runset (see "About Calibre Interactive Runsets" on page 33). If you have a runset loaded with desired settings, you can skip to the end of the following procedure.

## Prerequisites

- "Requirements for Calibre Auto-Waivers" on page 203
- You are running Calibre Interactive nmDRC, Calibre Interactive nmLVS with ERC, or Calibre Interactive DFM with Critical Feature Analysis (CFA).
- If you are using waivers from multiple sources, then all waiver shapes must be on the same layer/datatype. This may not be the case if waivers are provided by different IP vendors and a foundry, for example. For additional requirements and instructions for handling this situation, see "Performing a DRC Waiver Run Using Multiple Waiver Criteria Files" in the *Calibre Auto-Waivers User's and Reference Manual*.

## Procedure

1. Click **Rules** on the left panel of Calibre Interactive and specify the rule file and run directory.
2. Click **Inputs** on the left panel of Calibre Interactive. Specify information for the layout database on the **Layout** tab. The layout format must be GDSII or OASIS for Calibre Auto-Waivers.
3. Click the **Waivers** tab on the Inputs pane.
4. Enable "Use Waivers when running DRC/LVS/CFA."
5. (Optional) If you have a waiver setup file already created and want to use it for the waiver run, click the **Import Waiver Setup File** button. This populates the GUI fields with the values in the selected waiver setup file; only GUI settings specified in the imported file are overwritten.

   Use the following steps to confirm that the imported setup parameters are correct. The waiver setup parameter that corresponds to the GUI control is given in parentheses.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

6. Choose the **Description Files** category; the Description Files pane is shown for Calibre Interactive nmDRC.

a. Specify the “Waiver Criteria File(s).” It is not necessary to specify the Waiver Criteria file if you specify “Use criteria from the waiver shape files” in the **DRC Run** category in Step 9.

   Click the button to edit the file in a text editor. See “Waiver Criteria File Format” in the *Calibre Auto-Waivers User’s and Reference Manual* for complete information. (WAIVER_CRITERIA)

   **Note**

   The corresponding parameter in the waiver setup file is given in parentheses.

b. (Optional) Specify the directory for temporary files in the Temp Files Directory field. The default is the working directory. (TMP_DIR)

7. Choose the category **Waiver Shapes** and specify the waiver database as follows, depending on the method of waiver shape creation:

| If … | Do this … |
|---|---|
| *All* of the waiver shapes are embedded in the main design specified on the **Layout** tab of the Inputs pane. | Set Waiver Database to Embedded.<br>(MERGE YES) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| If … | Do this … |
|---|---|
| At least some waiver shapes exist in separate waiver shape files, and the waivers embedded in the design should be taken into consideration. | 1. Set Waiver Database to Both. (MERGE NO)<br>2. Specify the waiver shape file(s). (WAIVER_DATABASE)<br>The specified waiver shape files are combined at runtime with the design file on the **Layout** tab of the Inputs pane. |
| At least some waiver shapes exist in separate waiver shape files, and the waivers embedded in the design should *not* be taken into consideration. | • Set Waiver Database to External. (MERGE IGNORE_DESIGN_WAIVERS)<br>• Specify the waiver shape file(s). (WAIVER_DATABASE)<br>Only the external waivers are taken into consideration when applying waivers. |

8. (Optional) Select the setting for "Use rule file precision" (in the **Waiver Shapes** category):

   - **Enabled** — (default) Specifies PRECISION_CONVERSION YES in the waiver setup file.
   - **Disabled** — Specifies PRECISION_CONVERSION NO in the waiver setup file.

   This setting controls how waiver shape precision and magnification is handled if the rule file precision differs from the layout database precision; it has no effect if the precisions are the same. See "Waiver Cell Precision and Magnification Considerations" in the *Calibre Auto-Waivers User's and Reference Manual* for details. (PRECISION_CONVERSION).

9. Choose the category **DRC Run** and specify run options as desired:

   - **Use criteria from the waiver criteria files** — As described. Use this setting when all waiver criteria is contained in the specified file(s). (WAIVER_CRITERIA file)
   - **Use criteria from the waiver shape files** — Extract waiver criteria from the waiver shape file. The specified waiver criteria file is overwritten or created if it does not exist. (WAIVER_CRITERIA EXTRACT file)

     **Note**

     If both "Use criteria from the waiver criteria files" and "Use criteria from the waiver shape files" are enabled, then APPEND is used in the WAIVER_CRITERIA specification.

   - **Verify waiver cell checksums** — Verifies that the contents of the waiver cells match the checksum. (RUN_LAYOUT_CHECKSUM)



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Verify Calibre version** — Verifies the Calibre version in waiver cells. (RUN_CALIBRE_VERSION_CHECKSUM)
- **Retain edge and error result types** — Retain unwaived error and edge results from a waiver run instead of converting them to polygons. (RETAIN_RESULT_TYPE)
- **Verify IP_MATCH checksums** — As described. (RUN_IP_MATCH)
- **Verify rule checksums** — Use rule checksums to verify that the rule that produced the waiver shape has not changed. (RUN_RULE_CHECKSUM)
- **Ignore waiving density operation** — Do not apply waivers to rules with density checks. (IGNORE_DENSITY)
- **Enable dynamic results reporting** — Enables viewing of waived results in Calibre RVE as soon as they are available. (ENABLE_DYNAMIC_RESULTS_REPORTING)

10. Choose the **Waiver RDB Files** category and specify settings as desired to control the output of RDB files. (WAIVER_RDB, UNUSED_WAIVER_RDB, USED_WAIVER_RDB, DENSITY_WAIVER_RDB, GENERATE_COMBINED_RESULTS)

    Select “Enable waiver cell reporting” to add the cell names as an extra property in generated waiver RDB files. This option is disabled by default and corresponds to ENABLE_WAIVER_CELL_REPORTING.

    The option “Ignore waiver statistics in summary file” corresponds to OPTIMIZE_NONE_WAIVER_STATISTICS. The option is available if any of the waived, unused, or waived RDB options are set to NONE.

    Set “Generate combined results” to YES to generate a combined waivers results database with both the used and unused waiver shapes. The entries in the combined database include additional properties to indicate whether the waiver was applied or unused.

11. Click the **Layers and Text** category and specify settings as desired.

    These settings control the layer and datatype for the waiver shapes and text annotations. During a waiver run these settings specify the layer and datatype pairs that contain the waiver shapes and text annotations.

    See “Waiver Setup File Format for Waiver Verification” in the *Calibre Auto-Waivers User’s and Reference Manual* for a description of the options and their defaults; also see “Text Object Annotations” in the same manual.

12. (Optional) Click the **View Waiver Setup File** button to view the waiver setup parameters. You can choose **File > Save As** to save the file.

13. Make other settings as desired.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

In Calibre Interactive nmDRC, the preference setting "Start RVE as soon as results are available" has no effect if Calibre Auto-Waivers is enabled.

14. (Optional) Choose **File** > **Save Runset** to save the runset.
15. Click **Run DRC** when you are ready to start your Calibre nmDRC run.

**Note**

When waiver shapes are saved in a separate file ("Waivers have been merged with existing design" OFF) rather than embedded in the design, it is possible for the precision of the waiver database to be different than the precision of the layout database. Calibre adjusts the magnification automatically if this happens, as described in the section "Waiver Cell Precision and Magnification Considerations" in the *Calibre Auto-Waivers User's Manual*.

## Results

In addition to the output files specified on the Outputs pane, the following files are created or modified during a waiver run:

- **Waiver setup file** — A waiver setup file is generated by Calibre Interactive based on the GUI parameters and named *_waiver_setup_* by default. To specify a different file name, choose **Setup** > **Preferences**, click the **Misc** tab, enable the option "Specify waiver setup file name," and specify the file name.

  The parameter WORKING_DIRECTORY is set to the run directory specified on the Rules pane of Calibre Interactive.

- **Waiver criteria file** — If "Use criteria from the waiver shape files" is specified on the **DRC run** category pane, then the waiver criteria file is created or modified based on the waiver criteria included in text objects in the waiver shape file(s).
- *calibre-waiver.summary* — Contains runtime messages and results statistics.

The presence of the following RDB files depends on the settings on the **Waivers** tab in the **Waiver RDB Files** category:

- *waived.rdb*
- *unused_waiver.rdb*
- *used_waiver.rdb*
- *density_waived.rdb*

See these sections in the *Calibre Auto-Waivers User's Manual* for a complete description of the output from a waiver verification run:

- "Results" section in "Performing a Waiver Run"



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- "Calibre Auto-Waivers File Usage"



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 6
# Using Calibre Interactive to Perform DFM

You can use Calibre Interactive to run Calibre DFM and Calibre® LFD™.

This chapter assumes you are familiar with the terms and concepts presented in the chapter "Running Calibre Interactive" on page 63. The objectives of this chapter are:

- To guide you in setting up and performing a Calibre DFM or Calibre LFD run using Calibre Interactive.
- To provide you with the knowledge you need to change existing settings so you can tailor each run to fit your current needs. When you are satisfied with the settings, you can save your settings as a new runset for use in your next session.

**Introduction to Calibre Interactive DFM** . . . . . 215
**Specifying a Rule File for a DFM Run**. . . . . 217
**Specifying the Checks to Execute in a Calibre Interactive DFM Run** . . . . . 218
**Specifying the Run Type for a Calibre Interactive DFM Run** . . . . . 219
**Inputs for a Calibre Interactive DFM Run** . . . . . 221
**Specifying Automatic Waivers for DFM CFA** . . . . . 226
**Outputs for a Calibre Interactive DFM Run** . . . . . 227
**Run Control, Remote Execution and Licensing Options** . . . . . 232
**DFM Options** . . . . . 233

## Introduction to Calibre Interactive DFM

The Calibre Interactive DFM Graphical User Interface (GUI) is an interface to Calibre DFM. The Calibre Interactive - DFM window allows you to specify command line options and select rule file statements that guide Calibre DFM.

The most common scenario for performing DFM from Calibre Interactive assumes that a runset containing all the necessary settings already exists. Your time is spent running DFM and correcting errors in an interactive verification loop, as illustrated in Figure 6-1. In this scenario, you spend very little time setting up to run Calibre because you can load the necessary settings from a runset. You can set up Calibre Interactive to load the runset automatically on invocation as described in "Loading Runsets on Invocation" on page 39.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 6-1. Interactive Verification for DFM**

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Using Calibre RVE for DFM [Calibre RVE User's Manual]

# Specifying a Rule File for a DFM Run

In general, it is best to begin setup for a DFM run in the Rules pane. After you define a rule file, you can load the rule file to populate other GUI fields with rule file settings. Calibre Interactive interprets any relative pathnames as relative to the run directory, which you define on this pane.

### Procedure

1. Click Rules to display the Rules pane.

2. Specify the name of the rule file. You can use the **View** button to view and edit the rule file. After you load a rule file, any information you specify in the GUI supersedes the information in your loaded rule file.

   Be aware that Calibre Interactive automatically recognizes a rule file with the literal name “rules” and displays the name in green. Because the name is shown in green, you may assume the file has been loaded but it has not been; the file must be explicitly loaded using the **Load** button.

3. (Optional) Click **Load** to populate the GUI with the settings in the rule file.

   A rule file can be specified in a runset. After loading that runset, the rule file is included in the control file but its setting will not be loaded until you click **Load**.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

The setting "Read Inputs/Outputs fields while loading rules file" (**Setup** > **Preferences**, **Misc** tab) controls whether or not rule file settings populate the GUI options on the Inputs and Outputs panes when **Load** is clicked. This setting is enabled by default. See "Read Inputs/Outputs fields while loading rules file" in Table 3-20 on page 156.

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

4. Specify the run directory name. If the directory you specify does not exist, Calibre Interactive queries as to whether you want to create the directory.
5. (Optional) View the layer derivation tree at the bottom of the Rules pane; see "Layer Derivations Tree" on page 126 for information.

### Related Topics

About Rule Files

Including Other Rule Files and SVRF Commands in Calibre Interactive

# Specifying the Checks to Execute in a Calibre Interactive DFM Run

The Check Selection Recipe specifies the checks that are executed in a Calibre Interactive DFM run. You can use a built-in check recipe or define your own check recipe.

See "Check Selection Recipe" on page 109 for a full description of check recipes.

### Prerequisites

- The setting "Disable check selection" (**Setup** > **Preferences**, **Disable** tab) is not enabled.

### Procedure

1. Click Rules to display the Rules pane.
2. Specify a rule file; see "Specifying a Rule File for a DFM Run" on page 217 for additional details and a view of the Rules pane.
3. Select a check recipe using the dropdown list for Check Selection Recipe.

   BUILT-IN RECIPES

   - **Checks selected in the rules file** — (Default) Run only checks selected in the rule file. This is the only built-in recipe that obeys DFM [Un]Select Check and DRC [Un]Select Check statements in the rule file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

See the definition of DFM Select Check for the behavior of Calibre DFM if no DFM Select Check statements are specified in the rule file.

If you are using Customization Files to select checks for a Calibre Interactive DFM run, you should always use "Checks selected in the rules file" to avoid conflicts between the check recipe and the Customization GUI.

- **All checks** — Run *all* checks in the rule file, except those excluded by pre-processor directives. DFM [Un]Select Check and DRC[Un]Select Check statements are *not* considered.

USER RECIPES

- Select a previously defined custom check recipe.

  See "Basic Editing of Custom Check Selection Recipes" on page 111 for information on creating a Check Selection Recipe.

### Results

The checks specified in the check recipe are executed during the Calibre DFM run.

### Related Topics

Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive

# Specifying the Run Type for a Calibre Interactive DFM Run

The run mode in Calibre Interactive DFM determines whether you run Critical Area Analysis (CAA), Critical Feature Analysis (CFA), the CMP Analyzer, or Litho Friendly Design (LFD).

### Procedure

1. Click **Inputs** on the left panel of Calibre Interactive.
2. Select a run mode using the Run dropdown list. The following table explains how Calibre is invoked for each combination of Run Mode and Results Format:

**Table 6-1. DFM Run Modes and Result Formats**

| Run mode | Command-line argument | |
|---|---|---|
| | ASCII, GDSII, or OASIS Results Format | DFM Database Results Format |
| Critical Area Analysis (CAA) | N/A | -dfm |
| Critical Feature Analysis (CFA) | N/A | -dfm |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 6-1. DFM Run Modes and Result Formats (cont.)**

| Run mode | Command-line argument | |
|---|---|---|
| | ASCII, GDSII, or OASIS Results Format | DFM Database Results Format |
| CMP Analyzer | N/A | -dfm -cmp |
| Litho Friendly Design (LFD) | -lfd | -dfm -lfd |

## Results

The following additional tabs appear, depending on the run mode:

- **CMP Tab** — When you select the CMP Analyzer run mode, a **CMP** tab appears in the Inputs pane.
- **CAA Tab** — When you select the Critical Area Analysis (CAA) run mode, a **CAA** tab appears in the Inputs pane.

| Try It! | **Calibre CAA and CFA Tutorial and Example Kits (eKits)** |
|---|---|
| | These two eKits demonstrate basic procedures for using Calibre Critical Area Analysis and Calibre Critical Feature Analysis, and include all files and design data needed to run the tools.<br>Go to this page on Support Center to download the eKits (Documentation tab, Document Types=Getting Started Guide). The link goes to the latest release. |

## Related Topics

Calibre Interactive GUI Items for Calibre CMPAnalyzer [Calibre CMPAnalyzer User's Manual]

Calibre Litho-Friendly Design User's Manual

Calibre CMPAnalyzer Users Manual



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Inputs for a Calibre Interactive DFM Run

Layout input for a Calibre Interactive DFM run can be exported from an open design tool or read from file. You can also specify multiple layout inputs.

**Table 6-2. Sources for Layout Data in Calibre Interactive DFM**

| Open Cell in Design Tool | Files on Disk |
|---|---|
| Layout data comes from the active cell in the layout editor. Calibre Interactive prompts the editor to generate an intermediate layout file at run time. | Layout data exists as one or more layout files used for input. No viewer is connected to Calibre Interactive. |
| Check “Export from layout viewer” on the Inputs pane. | Uncheck “Export from layout viewer” on the Inputs pane. |

**Tip**

You can set a preference for the setting of “Export from layout viewer” that depends on whether your Calibre Interactive session is connected to a layout viewer. By default, this preference is set so that “Export from layout viewer” is enabled if Calibre Interactive is started from a layout viewer, regardless of the saved runset value for the option. See “Export from Viewer Preferences” on page 152.

**Reading Layout Input from an Open Design Tool in Calibre Interactive DFM** . . . . . 221
**Reading Layout Input from File in Calibre Interactive DFM**. . . . . 223
**Specifying Multiple Layouts for a DFM Run** . . . . . 225

## Reading Layout Input from an Open Design Tool in Calibre Interactive DFM

You can use the open cell in a your design tool as the layout input for a Calibre Interactive DFM run. Enable “Export from layout viewer” on the Inputs pane to use this mode.

### Prerequisites

- Calibre Interactive nmDRC is open and connected to your design tool. See the following topics for information:

  “Invoking Calibre Interactive From a Layout or Schematic Viewer” on page 64

  “Communication with the Design Tool” on page 72

- Your design tool is properly set up to export the design to the layout format you select in Step 3.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Click **Inputs** on the left panel of Calibre Interactive.
2. Click the **Layout** tab.

3. Use the “Format” dropdown button list and select the format of the *exported* layout data from GDSII, OASIS, or LEFDEF. For LEF/DEF format, see “Export DEF Layout from Design Tool” on page 89 for more information.
4. Enable “Export from layout viewer.”

   This specifies the layout viewer as the input source.

   Batch runs: In addition, you must also disable the setting “Ignore layout and source export in batch mode”; choose **Setup** > **Preferences**, then the **Misc** tab to find the setting. Also see “Specifying Layout and Schematic Export for a Calibre Interactive Batch Run” on page 104 for details on the layout viewer connection for a batch run.
5. Verify the filename for the extracted layout database in the “File” field. This file will be overwritten if it exists already.

   When exporting a layout database from a connected viewer, an intermediate file is extracted and saved to the file named in the “File” field. This filename is generated automatically based on the setting on the **Templates** > **Inputs** tab of Setup Preferences dialog box.

   If the filename is red, the file does not exist and Calibre Interactive creates the file. If the filename is green, the file exists and will be overwritten. Enter another filename if you want to keep the existing file.

   **Caution**

   When “Export from layout viewer” is enabled, do not use the extracted layout named in the “File” field for tapeout or any other use.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

6. (Optional) If you have multiple layout inputs for your DFM run, choose **Setup** > **DFM Options** then the **Database** > **Library** tab to specify them. See "Specifying Multiple Layouts for a DFM Run" on page 225. The layout format must be GDSII or OASIS.
7. Verify the top cell name. The top cell is determined from the current cell in the viewer.
8. (Optional) Select the "Area" option to perform DFM checking on only an area of the layout. See "Performing Area DFM" on page 234 for complete information.
9. (Optional) If you are running Critical Feature Analysis (CFA) you can apply waivers during the run. Calibre Auto-Waivers controls are on the Inputs pane on the **Waiver** tab; see "Calibre Auto-Waivers in Calibre Interactive" on page 201.

### Related Topics

Layout System [Standard Verification Rule Format (SVRF) Manual]

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Primary [Standard Verification Rule Format (SVRF) Manual]

## Reading Layout Input from File in Calibre Interactive DFM

Calibre Interactive can read layout input from database files. Uncheck "Export from layout viewer" on the Inputs pane to use this mode.

### Procedure

1. Click **Inputs** on the left panel of Calibre Interactive.

**Tip**

If you want to read information for the Inputs and Outputs panes from the rule file, choose **Setup** > **Preferences**, click the **Misc** tab, and enable the "Read Inputs/Outputs fields when loading rules file" option. (This option is enabled by default.) and rule file options that relate to the Inputs and Outputs panes are automatically applied when you click **Load** on the Rules pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Click the **Layout** tab.

3. Select a layout format using the Format dropdown list. For formats other than GDSII or OASIS, see the following topics for more information:

   - LEFDEF — see "Read LEF/DEF Input from Design Files" on page 88
   - OPENACCESS — see "Providing OpenAccess Input" on page 90

   For a full list of supported layout databases, refer to the "Layout Databases" section of the *Calibre Verification User's Manual*. Also see "Third Party Database Options" on page 80.

4. Uncheck "Export from layout viewer" to specify layout database files as the source of the layout. The checkbox "Export from layout viewer" is not displayed if the layout format does not allow layout export.

5. Specify the filename for the layout database in the "File" field.

6. (Optional) If you have multiple layout inputs for your DFM run, choose **Setup** > **DFM Options** then the **Database** > **Library** tab to specify them. See "Specifying Multiple Layouts for a DFM Run" on page 225. The layout format must be GDSII or OASIS.

7. Specify the top cell name.

8. (Optional) Select the "Area" option to perform DFM checking on only an area of the layout. See "Performing Area DFM" on page 234 for complete information.

9. (Optional) If you are running Critical Feature Analysis (CFA) you can apply waivers during the run. Calibre Auto-Waivers controls are on the Inputs pane on the **Waiver** tab; see "Calibre Auto-Waivers in Calibre Interactive" on page 201.

## Related Topics

Layout System [Standard Verification Rule Format (SVRF) Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Primary [Standard Verification Rule Format (SVRF) Manual]

Setting Database and General Options in Calibre Interactive

# Specifying Multiple Layouts for a DFM Run

If your primary cell has references to other additional database files you may need to load multiple layouts. You can specify multiple layouts on the **Database** > **Library** tab of the DFM Options pane.

Calibre Interactive checks the format of the additional database files and only includes files that match the format of the primary layout database.

## Prerequisites

- Main layout database in GDSII or OASIS format.
- Additional layout files in GDSII or OASIS format. Files compressed with gzip may be used. The additional layout files must be the same format as the main layout database.

## Procedure

1. Click the **Inputs** button, then click the **Layout** tab.
2. Enter a layout file in the “File” field of the **Layout** tab.
3. Select GDSII or OASIS for the format, as appropriate.
4. Enable the DRC Options pane by choosing **Setup** > **DFM Options**.
5. Click the **Database** tab; then click the **Library** sub-tab.
6. Use the buttons next to “Additional Layout Files” to add the required files.

   The additional files are added to the Layout Path statement. Only files that match the format of the primary database are added.
7. If desired, enable the checkbox in the “Golden” column to add the GOLDEN keyword to the layout file specification in the Layout Path statement.

   The GOLDEN keyword is used when doing “golden” cell matching. See the Layout Path statement and the WITH MATCH GOLDEN keyword for the Extent Cell, Inside Cell, and Not Inside Cell statements for further information on the GOLDEN keyword. It is also possible to specify multiple layouts using wildcards.
8. (Optional) In order to receive a warning if the additional layout files are not the same format as the main layout database, select **Setup** > **Preferences**, click the **Misc** tab, and enable “Warn when library layout file format differs from primary layout format.” The warning is displayed when you click the **Run DFM** button.

   Additional layout files that are not the same format as the primary database are ignored.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Inputs for a Calibre Interactive DFM Run

# Specifying Automatic Waivers for DFM CFA

Calibre Auto-Waivers with Critical Feature Analysis allows you to waive certain results and run Calibre DFM with these waivers taken into account. The waiver_flow executable is used to generate waiver shapes, and can also be run from Calibre Interactive - DFM when you are performing Critical Feature Analysis (CFA).

The controls for Calibre Auto-Waivers are on the Inputs pane on the **Waivers** tab; see "Calibre Auto-Waivers in Calibre Interactive" on page 201 for instructions on using this tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Outputs for a Calibre Interactive DFM Run

You specify the format of the output database on the Outputs pane of Calibre Interactive DFM. You can specify options for a DFM database on the **DFMDB** tab of the DFM Options pane.

**Specifying Outputs for a DFM Run** . . . . . 227
**Specifying Layers and Annotations for DFM Results Database Output** . . . . . 229
**Running a YieldServer Script on the DFM Database** . . . . . 231

## Specifying Outputs for a DFM Run

You specify the format of the output database on the Outputs pane of Calibre Interactive DFM.

Throughout this section, be aware that Calibre Interactive creates new directories if you specify new pathnames. If you have the “Read Inputs/Outputs fields when loading rules file” option selected on the **Misc** tab of the Setup Preferences dialog box, relevant rule file settings will populate the Outputs pane.

### Procedure

1. Click the Outputs button on the left panel of Calibre Interactive.
2. If there are multiple tabs on the Outputs pane, select the **Common** tab.
3. Select the format of the output database from the “Results Format” selections:
   - DFM Database
   - ASCII
   - GDSII
   - OASIS

   The fields present in the Outputs pane depend on the “Results Format” you choose. For a view of the Outputs pane with a results format of DFM Database, see the following figure.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 6-2. DFM Outputs Pane**

4. Fill in the required information for the results format you chose:

   - DFM Database — enter the directory path.

     The path specified in the Directory field is the parent directory of the DFM Database. The DFM database is created as a subdirectory within that parent directory and is named *<topcell>.dfmdb*, where *<topcell>* is the top cell name from the Inputs pane.

     See "Specifying Layers and Annotations for DFM Results Database Output" on page 229 for additional settings.

     Corresponding rule file statement: DFM Database.

     - ASCII, GDSII, and OASIS — Enter the filename for results database. Additional options are available depending on the database type; see the command DRC Results Database in the *SVRF Manual* for a description of the options.

       - **Pseudocells in** — Selections: "User cells", "User merged," or "Pseudocells," and "Top cell." See the tooltip for descriptions.
       - **Cell prefix** and **Cell suffix** — Available for GDSII and OASIS formats.
       - **Specify the amount of rule check text** — Specify the amount of rule check text that is written to the results database. This option only applies to ASCII results databases. See "Specifying the Rule Check Text Written to the Results Database" on page 182.
       - **CBLOCK** — Available for OASIS results format.
       - **STRICT** — Available for OASIS results format.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **INDEX** — Available for OASIS results format.
- **NOVIEW** — Available for OASIS results format if INDEX is enabled.

5. (Optional) Select "Show Results in RVE" to automatically start Calibre RVE for DFM when the run is done. Only DFM Database and ASCII results formats can be opened in Calibre RVE.

   If you want RVE to open after a batch run, choose **Setup** > **Preferences**, then the **Misc** tab, and enable "Start RVE after batch run."

   **Side RDBs to Open** — This selection is available if your results database format is ASCII, GDS, or OASIS. Click the button to open a dropdown list with the auxiliary databases (side RDBs) defined in the rule file; select the RDBs you want to open by checking the box in the "Open" column of the dropdown list. If the main database is in GDS or OASIS format, only the side RDBs will be opened in Calibre RVE. Side RDBs which are empty are not automatically opened.

   Also see the setting "Do not open empty main results database in RVE" on the **Misc** tab of the Setup Preferences dialog box.

   When Calibre RVE for DFM is started from Calibre Interactive, the settings on the Run Control pane are used to set the -turbo option for Calibre RVE; see the section "Multithreaded Parallel Processing for Calibre RVE Operations (-turbo Option)" in the *Calibre RVE User's Manual*.

6. (Optional) Specify a Summary Report File by enabling Write Summary Report File, and entering the filename. Adjust the other options for the summary report if needed.

7. (Optional) See "Running a YieldServer Script on the DFM Database" on page 231 to specify a YieldServer post-processing file and arguments.

8. At this point, you may have specified all the information you need to run DFM. If so, then click **Run DFM**. There are more options to consider, and these are discussed next.

# Specifying Layers and Annotations for DFM Results Database Output

If you are performing a run that generates a DFM Database, you can specify which layers you would like saved to a DFM Database.

## Procedure

1. Choose **Setup** > **DFM Options** to open the DFM Options pane.
2. Click the **DFMDB** tab, as shown in the following figure.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 6-3. DFMDB Tab for DFM Options**

3. Click the button next to “Layers to Save” and specify the layers that you would like saved to the DFM Database. Note: These layers correspond to keywords in the DFM Database operation.

   - ORIGINAL — Write original (merged) layers.
   - DEVICES — Device layers. This option causes device extraction to be performed.
   - ANALYZE — Input layers to all DFM ANALYZE operations.
   - MEASURE — Input layers to all DFM MEASURE operations.
   - PROPERTY — Input layers to all DFM PROPERTY operations.

4. If desired, select “Save layers involved in Connect statements.” This control selects the CONNECT or NO CONNECT keyword in the DFM Database operation.

5. Select either the “Use these” option or the “Use from rules” option if you want to specify database-level annotations that correspond to annotations added to the DFM Database statement.

   **Use these** — Annotations added to the table using the **Add** button are used in database output. Annotations listed in the rule file are ignored.

   **Use from rules** — Annotations listed in the DFM Database Annotations table are ignored. Only annotations listed in the rule file under the DFM Database statement are used.

   For more information, see “Database-Level Annotations” in the *Calibre YieldAnalyzer and YieldEnhancer User’s and Reference Manual*.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

6. If you chose “Use these” in the previous step, do the following to fill in the DFM Database Annotations table:

   a. (Optional) Click the **Load from Rules** button below the DFM Database Annotations table to populate the table with annotations present in the rule file. This step overwrites the existing table entries.

      Perform this step if you want to edit the rule file annotations.

   b. Use the **Add** and **Delete** buttons to specify annotation names and the corresponding values you would like to be saved. These options are also available using the right-click menu over the Annotations table.

      Click in the Name or Value cell the change the value.

# Running a YieldServer Script on the DFM Database

You can run a YieldServer script on DFM database after it is created.

## Prerequisites

- The output results format is DFM Database on the **Common** tab of the Outputs pane.

## Procedure

1. Choose **Setup** > **DFM Options** to open the DFM Options pane.
2. Click the **DFMDB** tab, as shown in Figure 6-3.
3. Enable “Run YS Script.”
4. Select one of the following:

   - **Extract from Rule File** — Run the YieldServer script in autostart mode. See DFM YS Autostart.
   - **File Path** — Provide the path to a YieldServer script.

     Enter the arguments for the YieldServer command in the “Arguments” field. Internal trigger parameters, such as %r (rule file) and %l (layout path), may be used for the arguments. The trigger parameters can be viewed by choosing **Setup** > **Preferences**, then the **Triggers** tab.

## Related Topics

Internal Triggers in Calibre Interactive

Calibre YieldServer Reference Manual



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Run Control, Remote Execution and Licensing Options

You can control how your Calibre DFM job is run by configuring options related to performance, remote and distributed computing, and license acquisition.

These topics are covered in detail in ““Run Control Setup and Licensing” on page 681”.

You can also specify triggers by choosing **Setup** > **Preferences**, then the **Triggers** tab. See “Trigger Functions in Calibre Interactive” on page 657.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# DFM Options

Calibre Interactive provides access to a wide range of Calibre DFM options. Some are available in the **Setup** menu, others are available through the DFM Options pane, which you access by clicking the **DFM Options** button on the left panel.

Some options are similar for all applications—see "Setting Database and General Options in Calibre Interactive" on page 93.

If the **DFM Options** button is not visible, choose the **Setup > DFM Options** menu item to add the **DFM Options** button to the left side of the Calibre Interactive.

**Limiting the DFM Results Count** . . . . . 233
**Specifying DFM Database Options** . . . . . 233
**Reporting DFM Results in Cell Space** . . . . . 234
**Performing Area DFM** . . . . . 234

## Limiting the DFM Results Count

You can control the number of results reported for each DFM rule check. This reporting option is only applicable to ASCII RDB output.

### Procedure

1. Click the DFM Options button on the left panel, or choose **Setup > DFM Options** if the button is not visible.
2. Click the ASCII Results tab.
3. Select the maximum result count in "Max errors generated per check":
   - **All** — Report all results. Choose this option if you are using Calibre DFM to output a modified version of your layout as mask data.
   - **Number** — Enter the maximum allowed result count.

## Specifying DFM Database Options

You can specify options for the DFM Database on the **DFMDB** tab of the DFM Options pane.

See the following topics:

- "Specifying Layers and Annotations for DFM Results Database Output" on page 229
- "Running a YieldServer Script on the DFM Database" on page 231



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Reporting DFM Results in Cell Space

Calibre reports DFM results in top-cell coordinates by default. Choose the “Output cell errors in cell space” option to instruct Calibre to use local cell space coordinates when reporting errors. This option is only for DFM-H runs that output ASCII format results database.

## Procedure

1. Choose **Setup** > **DFM Options** to display the DFM Options pane.
2. Click the **ASCII Results** tab.
3. Check the “Output cell errors in cell space” option.

# Performing Area DFM

You can run Calibre DFM on only a specified rectangular area of the layout. Area DFM runs on the specified area plus a *halo* region around the area. You can specify the halo size and whether results in the halo region are filtered out or kept.

## Procedure

1. Click **Inputs** on the left panel and enter the layout database information as described in “Inputs for a Calibre Interactive DFM Run” on page 221.
2. Enable the Area option.
3. Select the area coordinates with one of the following methods:
   - Enter space-separated coordinates for the rectangular window you want to check. Coordinates are the lower-left and upper-right corners of the rectangle, in that order. You must use this method if not connected to a layout viewer.
   - If connected to a layout viewer, click the browse button (...) to select the area you want to check in the layout viewer. Click and drag out an area selection rectangle In



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

the layout viewer. The coordinates of the area selection rectangle appear in the Area field of the Inputs pane.

**Tip**

When connected to a layout viewer, you can right-click the (...) button to highlight and zoom to the specified Area DFM region in the layout.

4. (Optional) Set Area DFM options, such as halo size and coordinate scaling. Some options depend on the run mode. Options are not available for Critical Area Analysis (CAA) runs.

   a. Click the **DFM Options** button on the left panel, or choose **Setup** > **DFM Options** if the left panel button is not present.

   b. Click the **Area DFM** tab, then select the desired options described in the following table.

**Table 6-3. Area DFM Options**

| Option | Description |
|---|---|
| Halo width for area DFM | Specifies the size of the halo which expands the Area DFM window.<br>• **Automatic** — Width of halo is the greater of the width or height of the specified Area DFM region.<br>• **Size** — Width of halo is an explicit user-specified width in user-units.<br>Note: Depending on the geometry of your area selection, the automatic setting can result in a halo region that is much larger than needed, resulting in slow performance. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 6-3. Area DFM Options (cont.)**

| Option | Description |
|---|---|
| Remove results from halo region after area-DFM<br>(Not available for CMP Analyzer runs. Only applies to the ASCII format results database.) | Enabled — (default). As described.<br>Disabled — Keep results in the halo region. |
| Apply coordinate scaling | Enabled — (default) Scale window coordinates by:<br>`(DFM Results Database Precision/Precision) *(Layout Magnify)`<br>Disabled — Do not apply scaling. |
| Shrink factor<br>(Only available for CMP Analyzer runs.) | Apply a shrink factor to the layout. This is useful if you want to simulate the effect of an optical shrink factor on the layout.<br>Adds Layout Magnify and DRC Magnify Results statements to the Calibre Interactive control file. |
| Extent layer<br>(Only available for CMP Analyzer runs.) | Specifies an extent layer.<br>The extent layer can be specified as a layer number or in *layer_number.datatype* format. If you specify only a layer number, then the datatype is assumed to be 0 (zero). |

5. If all other options are set, click the **Run DFM** button to start the run.

## Results

The following changes in output occur when Area DFM is enabled:

- **Results database** — Results in the halo region are filtered out based on the setting “Remove results from halo region after area-DFM”. Result filtering only applies to ASCII format results databases.
- **Auxiliary databases** — If an auxiliary results database is specified in the rule file, the results are filtered as specified for the main results database.
- **Transcript and summary report output** — If the setting “Remove results from halo region after area-DFM” is enabled, then both the transcript and the DFM Summary Report report the number of results that were filtered out of the halo region.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 7
# Using Calibre Interactive to Perform LVS

The Calibre Interactive nmLVS GUI allows you to specify command line options and select rule file statements for a Calibre nmLVS run. You can save GUI settings to a Calibre Interactive runset and load them in future runs.

This chapter assumes you are familiar with the terms and concepts presented in the chapter "Running Calibre Interactive" on page 63.


**Calibre Interactive nmLVS Overview** . . . . . 237
**Specifying a Rule File for a Calibre nmLVS Run** . . . . . 239
**Specifying the Calibre nmLVS Run and Analysis Type** . . . . . 240
**Inputs for a Calibre Interactive nmLVS Run** . . . . . 243
**Specifying Outputs for a Calibre nmLVS Run** . . . . . 263
**Running Calibre Interactive nmLVS** . . . . . 264
**Run Control, Remote Execution and Licensing Options** . . . . . 266
**Calibre nmLVS Options in Calibre Interactive** . . . . . 267
**Device Signature Creation with Calibre Interactive nmLVS** . . . . . 286


## Calibre Interactive nmLVS Overview

The Calibre Interactive nmLVS Graphical User Interface (GUI) is an interface to Calibre nmLVS. The Calibre Interactive - nmLVS window allows you to specify command line options and selected rule file statements that guide Calibre nmLVS.

A common scenario for performing LVS from Calibre Interactive assumes that a runset file containing all the necessary settings already exists. Your time is spent running Calibre nmLVS and correcting errors in an interactive verification loop, as illustrated in Figure 7-1. In this scenario, you spend very little time setting up the Calibre run because you can load the necessary settings from a runset. You can set up Calibre Interactive to load the runset automatically on invocation as described in "Loading Runsets on Invocation" on page 39.

You may want to refer to the *Calibre Verification User's Manual* for information on Calibre nmLVS.




Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 7-1. Generic LVS Flow**

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Using Calibre RVE for LVS [Calibre RVE User's Manual]

# Specifying a Rule File for a Calibre nmLVS Run

You need to specify a rule file for any Calibre nmLVS run. You specify the rule file on the **Rules** tab of Calibre Interactive. You can click the **Load** button to load the rule file settings into the GUI fields.

If you want to use device signatures in the device identification process, you can use the Create Device Signature dialog to create the device signatures and add the appropriate statements to your rule file. See "Device Signature Creation with Calibre Interactive nmLVS" on page 286.

### Procedure

1. Click the Rules button to display the Rules pane.

2. Specify the rule filename in the LVS Rules File field. You can use the View button to view and edit the rule file.

   Be aware that Calibre Interactive automatically recognizes a rule file with the literal name "rules" and displays the name in green. Because the name is shown in green, you may assume the file has been loaded but it has *not* been; the file must be explicitly loaded using the **Load** button.

3. (Optional) Click **Load** to populate the GUI with the settings in the rule file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

A rule file can also be specified in a runset. After loading that runset, the rule file is included in the control file but its settings will not be loaded into the GUI until you click **Load**.

**Tip**

The setting "Read Inputs/Outputs fields while loading rules file" (**Setup** > **Preferences**, **Misc** tab) controls whether or not rule file settings populate the GUI options on the Inputs and Outputs panes when **Load** is clicked. This setting is enabled by default. See "Miscellaneous Preferences" on page 156.

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

4. Specify the run directory name in the LVS Run Directory field.
5. (Optional) View the layer derivation tree at the bottom of the Rules pane; see "Layer Derivations Tree" on page 126 for information.

### Related Topics

About Rule Files

About Calibre Interactive Runsets

Inputs for a Calibre Interactive nmLVS Run

Specifying Outputs for a Calibre nmLVS Run

Including Other Rule Files and SVRF Commands in Calibre Interactive

Calibre nmLVS Options in Calibre Interactive

# Specifying the Calibre nmLVS Run and Analysis Type

You specify the type of Calibre nmLVS run on the Inputs pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Click the **Inputs** button on the left panel to display the Inputs pane.

2. Choose the run type. This selects the type of run and the license to use.

| Run Type | Description |
|---|---|
| Hierarchical | Run in hierarchical mode. |
| Flat | LVS comparison and circuit extraction are performed flat. The -flatten command line option is used. |
| Recon | Run Calibre nmLVS Reconnaissance Short Isolation. The -si command line option is used. Several options not used when running Calibre nmLVS Recon SI are hidden, and the “Step” option changes to “Mode”:<br>• **All Shorts** — Perform both IO and Power Ground short isolation<br>• **Power Ground Shorts** — Perform only Power Ground short isolation<br>• **IO Shorts** — Perform only IO short isolation |
| Calibre CB | Use the Calibre CB (Cell/Block) flat license package. LVS comparison and circuit extraction are performed flat. The -cb and -flatten command line options are used. |

**Note**

See “Generating Waiver Shapes with Calibre Interactive” on page 205 for instructions on using the “Autowaiver Creation” run type.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Choose the analysis step from the “Step” dropdown list:.

| Analysis Step | Description |
|---|---|
| Layout vs Netlist | Perform layout netlist extraction and LVS comparison in one step. You supply a layout database and the source netlist. Calibre extracts the layout netlist from the layout database. If the Run type is “Hierarchical”, the extracted layout netlist is saved. |
| Netlist vs Netlist | Perform LVS comparison in a separate step from layout netlist extraction. You supply a layout netlist file in SPICE format and a source netlist. |
| Netlist Extraction | Perform layout netlist extraction in a separate step from LVS comparison. You supply a layout database. The extracted layout netlist is saved. |

4. Supply other required data on the Inputs pane; see “Inputs for a Calibre Interactive nmLVS Run” on page 243 for more information.

## Related Topics

Calibre nmLVS and Calibre nmLVS-H Command Line [Calibre Verification User's Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Inputs for a Calibre Interactive nmLVS Run

You specify input data and other input-related options in separate tabs on the Inputs pane.

The Inputs pane contains the following tabs:

**Table 7-1. Inputs Pane in Calibre Interactive nmLVS**

| Tab | Description |
|---|---|
| Layout | Specify the layout input (geometric database or layout netlist) on the **Layout** tab; see "Layout Input in Calibre Interactive nmLVS" on page 244. |
| Netlist | Specify the source input (netlist) on the **Netlist** tab; see "Source Netlist Input in Calibre Interactive nmLVS" on page 250. |
| H-Cells | Specify hcell input and perform hcell analysis; see "Supplying Hcell Data" on page 258 and "Performing Hcell Analysis in Calibre nmLVS" on page 260. |
| Signatures | Create device signatures; see "Device Signature Creation with Calibre Interactive nmLVS" on page 286. |
| Waivers | Specify waivers for ERC checks; see "Specifying Automatic Waivers for ERC" on page 262. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Layout Input in Calibre Interactive nmLVS

Layout data for a Calibre Interactive nmLVS run is specified on the Inputs pane in the **Layout** tab. The input data and options you specify depend on the analysis type and whether you are reading layout data from file or exporting it from an open design tool. Depending on the format of the layout input, you can specify multiple layouts and other options.

**Layout vs Netlist and Netlist Extraction runs** — see these topics:

"Reading Geometric Layout Input from an Open Design Tool in Calibre Interactive nmLVS" on page 244

"Reading Geometric Layout Input from File in Calibre Interactive nmLVS" on page 245

"Specifying Multiple Layouts for an Calibre nmLVS Run" on page 247

**Netlist vs Netlist runs** — see these topics:

"Reading Layout Netlist Input from an Open Design Tool in Calibre Interactive nmLVS" on page 248

"Reading Layout Netlist Input from File in Calibre Interactive nmLVS" on page 249

## Reading Geometric Layout Input from an Open Design Tool in Calibre Interactive nmLVS

Layout data for a Calibre Interactive nmLVS run is specified on the Inputs pane on the **Layout** tab. You can export the layout from an open design tool. Geometric layout data is used in Layout vs Netlist and Netlist Extraction runs. The layout netlist is extracted from the geometric layout.

### Prerequisites

- The Calibre nmLVS analysis type is "Layout vs Netlist" or "Netlist Extraction." See "Specifying the Calibre nmLVS Run and Analysis Type" on page 240. The options available for layout input depend on the run and analysis type.

### Procedure

1. Click the **Inputs** button on the left panel.
2. Click the **Layout** tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Enable "Export from layout viewer" to specify that the layout is exported from a connected design tool; see "Communication with the Design Tool" on page 72 for more information.
4. Select the *exported* layout format using the "Format" dropdown button list. The selections GDSII, OASIS, and LEFDEF are available if "Export from layout viewer" is selected. See "Providing LEF/DEF Input" on page 88 for more information on LEF/DEF input.
5. Specify the filename for the *exported* layout database in the "File" or "DEF File" field.

   If you are performing a Calibre Interactive batch run, see "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104.
6. Verify the name of the top cell name in the "Top Cell" field. The top cell is determined from the current cell in the viewer.
7. Enter the name for the extracted layout netlist in the "Layout Netlist" field. This option is for hierarchical runs only; Calibre nmLVS extracts the layout netlist from the geometric database and writes it to this filename.

   When connected to a layout viewer, a system-generated SPICE netlist name appears in the "Layout Netlist" field by default. This name is taken from the "Extract to File" template setting (**Setup** > **Preferences**, the **Templates** > **Outputs** sub-tab). However, you can choose a different filename.
8. (Optional) If you have multiple layout inputs for your run, you can specify them using the **Database** > **Library** sub-tab on the LVS Options pane. See "Specifying Multiple Layouts for an Calibre nmLVS Run" on page 247.
9. Possible next steps are:
   - "Specifying a Layout Text File in Calibre Interactive" on page 93
   - "Source Netlist Input in Calibre Interactive nmLVS" on page 250

### Related Topics

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Primary [Standard Verification Rule Format (SVRF) Manual]

Layout System [Standard Verification Rule Format (SVRF) Manual]

Setting Database and General Options in Calibre Interactive

## Reading Geometric Layout Input from File in Calibre Interactive nmLVS

Layout data for a Calibre Interactive nmLVS run is specified on the Inputs pane on the **Layout** tab. You can read the layout data file design files saved on disk. Geometric layout data is used in



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Layout vs Netlist and Netlist Extraction runs. The layout netlist is extracted from the geometric layout.

## Prerequisites

- The Calibre nmLVS analysis type is “Layout vs Netlist” or “Netlist Extraction.” See “Specifying the Calibre nmLVS Run and Analysis Type” on page 240. The options available for layout input depend on the run and analysis type.

## Procedure

1. Click the **Inputs** button on the left panel.
2. Click the **Layout** tab.
3. Uncheck “Export from layout viewer” to specify that the layout data is contained in the specified files.
4. Select the layout format using the “Format” dropdown button list:
   - GDSII — GDS format
   - OASIS — OASIS format
   - LEFDEF — see “Providing LEF/DEF Input” on page 88
   - OPENACCESS — see “Providing OpenAccess Input” on page 90

   For a full list of supported layout databases, refer to the “Layout Databases” section of the *Calibre Verification User’s Manual*. Also see “Third Party Database Options” on page 80.
5. Specify the filename for the layout database in the “File” field.
6. Specify the top cell name in the “Top Cell” field.
7. Enter the name for the extracted layout netlist in the “Layout Netlist” field. This option is not used in flat runs. In hierarchical runs, Calibre nmLVS extracts the layout netlist from the geometric database and writes it to this filename.
8. (Optional) If you have multiple layout inputs for your run, you can specify them using the **Database > Library** sub-tab on the LVS Options pane. See “Specifying Multiple Layouts for an Calibre nmLVS Run” on page 247.
9. Possible next steps are:
   - “Specifying a Layout Text File in Calibre Interactive” on page 93
   - “Source Netlist Input in Calibre Interactive nmLVS” on page 250

## Related Topics

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Primary [Standard Verification Rule Format (SVRF) Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Layout System [Standard Verification Rule Format (SVRF) Manual]

Setting Database and General Options in Calibre Interactive

# Specifying Multiple Layouts for an Calibre nmLVS Run

If your primary cell has references to other additional database files you may need to load multiple layouts. You can specify multiple layouts using the **Database** > **Library** tab on the LVS Options pane.

Calibre Interactive checks the format of the additional database files and only includes files that match the format of the primary layout database.

## Prerequisites

- Main layout database in GDSII or OASIS format.
- Additional layout files in GDSII or OASIS format. Files compressed with gzip may be used. The additional layout files must be the same format as the main layout database.

## Procedure

1. Click the **Inputs** button on the left panel, then click the **Layout** tab.
2. Enter a layout file in the “File” field of the **Layout** tab.
3. Select GDSII or OASIS for the “Format”, as appropriate.
4. Enable the LVS Options pane by choosing **Setup** > **LVS Options**.
5. Click the **Database** tab then click the **Library** sub-tab.
6. Use the buttons next to “Additional Layout Files” to add the required files.
7. (Optional) In order to receive a warning if the additional layout files are not the same format as the main layout database, select **Setup** > **Preferences**, click the **Misc** tab, and enable “Warn when library layout file format differs from primary layout format.” The warning is displayed when you click the **Run LVS** button.

   Additional layout files that are not the same format as the primary database are ignored.

## Results

The additional files are added to the Layout Path statement. Only files that match the format of the primary database are added.

## Related Topics

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Input in Calibre Interactive nmLVS



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Reading Layout Netlist Input from an Open Design Tool in Calibre Interactive nmLVS

Layout netlist data for a Calibre Interactive nmLVS run is specified on the Inputs pane on the **Layout** tab. You can export the layout netlist from an open design tool. The layout netlist is used in Netlist vs Netlist runs.

## Prerequisites

- The Calibre nmLVS analysis type is "Netlist vs Netlist." See "Specifying the Calibre nmLVS Run and Analysis Type" on page 240. The options available for layout data depend on the run and analysis type.
- Calibre Interactive is connected to a supported schematic viewer; see "Communication with the Design Tool" on page 72.
- The connected schematic viewer is set up to export the netlist in SPICE format.

## Procedure

1. Click the **Inputs** button on the left panel.
2. Click the **Layout** tab.
3. Enable "Export from schematic viewer" to specify that the layout netlist is exported from a connected design tool.

   The design tool must be set up to export the layout netlist in SPICE format.

   Batch runs: In addition, you must also disable the setting "Ignore layout and source export in batch mode" in the Setup Preferences dialog box on the **Misc** tab. Also see "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104 for details on the viewer connection for a batch run.
4. Specify the filename for the *exported* layout netlist in the "File" field.

   The exported netlist is written to the filename in the File field. When connected to a layout viewer, a system-generated SPICE netlist name appears in the "File" field by default. This name is taken from the "Extract to File" template setting (Choose **Setup** > **Preferences**, then click the **Templates** > **Outputs** sub-tab). However, you can choose a different filename.
5. Verify the name of the top cell in the "Top Cell" field.

   When your netlist is exported from a connected schematic viewer, the top cell is determined from the current cell in the viewer.
6. Possible next steps are:
   - "Source Netlist Input in Calibre Interactive nmLVS" on page 250



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Templates for File Naming

Communication with the Design Tool

Layout Input in Calibre Interactive nmLVS

## Reading Layout Netlist Input from File in Calibre Interactive nmLVS

Layout netlist data for a Calibre Interactive nmLVS run is specified on the Inputs pane on the **Layout** tab. You can read the layout netlist from an existing file. The layout netlist is used in Netlist vs Netlist runs.

### Prerequisites

- The Calibre nmLVS analysis type is “Netlist vs Netlist.” See “Specifying the Calibre nmLVS Run and Analysis Type” on page 240. The options available for layout data depend on the run and analysis type.
- A layout netlist in SPICE format. The layout netlist may be one extracted and saved from a previous Layout vs Netlist or Netlist Extraction run.

### Procedure

1. Click the **Inputs** button on the left panel.
2. Click the **Layout** tab.
3. Uncheck “Export from schematic viewer” to specify that the layout netlist is read from file.
4. Specify the filename for the layout netlist in the “File” field.
5. Specify the name of the top cell in the “Top Cell” field.

### Related Topics

Layout Input in Calibre Interactive nmLVS

Source Netlist Input in Calibre Interactive nmLVS



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Source Netlist Input in Calibre Interactive nmLVS

Calibre nmLVS comparison runs require a source (or schematic) netlist. Source data can be exported from an connected schematic viewer or exist in database files. Netlist extraction runs do not use a source netlist.

| Input Method | Comment |
|---|---|
| Source netlist input from a connected schematic viewer | See "Reading Source Netlist Data from an Open Schematic Viewer in Calibre Interactive nmLVS" on page 250.<br>The schematic viewer should be set up to export the netlist in SPICE or Verilog format. |
| Source netlist input from file | See "Reading Source Netlist Data from File in Calibre Interactive nmLVS" on page 252.<br>The netlist format can be SPICE, Verilog, or a combination of both types of files. |

These topics may also be of interest:

- "Setting Up the Verilog Translator (v2lvs)" on page 254
- "Providing Additional SPICE Netlist Files with the Source Netlist" on page 257

## Reading Source Netlist Data from an Open Schematic Viewer in Calibre Interactive nmLVS

You can read source netlist data from a connected schematic viewer. A source netlist is required in Layout vs Netlist and Netlist vs Netlist runs in Calibre Interactive nmLVS.

### Prerequisites

- The Calibre nmLVS analysis type is "Layout vs Netlist" or "Netlist vs Netlist." See "Specifying the Calibre nmLVS Run and Analysis Type" on page 240. The options available for layout data depend on the run and analysis type.
- Calibre Interactive is connected to a supported schematic viewer; see "Communication with the Design Tool" on page 72.
- The connected schematic viewer is set up to export the netlist in SPICE or Verilog format.

### Procedure

1. Click the **Inputs** button on the left panel,



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Click the **Netlist** tab.

3. Enable “Export from schematic viewer” to specify that a connected schematic viewer supplies the schematic source netlist.

   Batch runs: In addition, you must also disable the setting “Ignore layout and source export in batch mode” in the Setup Preferences dialog box on the **Misc** tab. Also see “Specifying Layout and Schematic Export for a Calibre Interactive Batch Run” on page 104 for details on the viewer connection for a batch run.

4. Specify the format of the *exported* source netlist using the “Format” dropdown button list.

   - **SPICE** — The exported netlist is in SPICE format.
   - **VERILOG** — The exported netlist is in Verilog format. If you specify Verilog as the source format, Calibre Interactive calls a utility named v2lvs to translate the Verilog netlist to SPICE. Choose **Setup > Verilog Translator** to set up the translation options; see the section “Setting Up the Verilog Translator (v2lvs)” on page 254. The translated SPICE netlist is then used for LVS comparison; the SPICE netlist name is taken from the “Files” field.

   The “MIXED” option is not available when “Export from schematic viewer” is enabled.

5. Specify the filename for the *exported* source netlist.

   The exported netlist is written to the filename in the “Files” field.

6. Verify the name of the top cell in the “Top Cell” field. The top cell is determined from the cell currently open in the viewer when “Export from schematic viewer” is enabled.

7. (Optional) Specify additional SPICE netlist files. This may be necessary to provide INCLUDE files for the source netlist.

   a. Click the **LVS Options** button or choose **Setup > LVS Options**.

   b. Click the **Database** tab and the **Library** sub-tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

c. Use the buttons next to “Additional SPICE Files” to specify the required SPICE files.

The additional files are included before the netlist specified on the Inputs pane in the **Netlist** tab. Calibre Interactive creates a file named *_source.net_* with .INCLUDE statements for each netlist file; *_source.net_* is used as the source netlist for the Source Path statement in the control file.

**Note**

Calibre Interactive verifies that the source netlist is complete before starting the run. You can turn off this check by disabling “Check source netlist is complete before running LVS” (**Setup** > **Preferences**, **Misc** tab). See “Miscellaneous Preferences” on page 156 for information on the source netlist check.

### Results

The appropriate statements are written to the Calibre Interactive control file at runtime.

### Related Topics

Communication with the Design Tool

Using Calibre Interactive to Perform LVS

Source Path [Standard Verification Rule Format (SVRF) Manual]

Source Primary [Standard Verification Rule Format (SVRF) Manual]

Source System [Standard Verification Rule Format (SVRF) Manual]

## Reading Source Netlist Data from File in Calibre Interactive nmLVS

You can read source netlist data from file. A source netlist is required in Layout vs Netlist and Netlist vs Netlist runs in Calibre Interactive nmLVS.

### Prerequisites

- The Calibre nmLVS analysis type is “Layout vs Netlist” or “Netlist vs Netlist.” See “Specifying the Calibre nmLVS Run and Analysis Type” on page 240. The options available for layout data depend on the run and analysis type.

### Procedure

1. Click the **Inputs** button on the left panel,



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Click the **Netlist** tab.

3. Uncheck “Export from schematic viewer” to specify that schematic netlist files provide the source schematic.

4. Specify the format of the source netlist using the “Format” dropdown button list.

   - **SPICE** — The netlist is in SPICE format.

   - **VERILOG** — The netlist is in Verilog format. If you specify Verilog as the source format, Calibre Interactive calls a utility named v2lvs to translate the Verilog netlist to SPICE. Choose **Setup** > **Verilog Translator** to set up the translation options; see the section “Setting Up the Verilog Translator (v2lvs)” on page 254. The translated SPICE netlist is then used for LVS comparison; the SPICE netlist name is taken from the “Verilog Files” field.

   - **MIXED** — Select the “Mixed” option to use a mixed mode of both SPICE and Verilog files. Separate netlist entry fields are displayed for the SPICE and Verilog files. Verilog files are handled as noted for the VERILOG option.

5. Specify the filename for the source netlist. To specify multiple files, click the down arrow and enter one netlist file per line. Alternatively, you can enter additional SPICE files in the LVS Options pane as described in a later step in this procedure.

6. Specify the top cell name in the “Top Cell” field.

7. (Optional) Specify additional SPICE netlist files. This may be necessary to provide INCLUDE files for the source netlist.

   a. Click the **LVS Options** button or choose **Setup** > **LVS Options**.

   b. Click the **Database** tab and the **Library** sub-tab.

   c. Use the buttons next to “Additional SPICE Files” to specify the required SPICE files.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

Calibre Interactive verifies that the source netlist is complete before starting the run. You can turn off this check by disabling "Check source netlist is complete before running LVS" (**Setup** > **Preferences**, **Misc** tab). See "Miscellaneous Preferences" on page 156 for information on the source netlist check.

### Results

The appropriate statements are written to the Calibre Interactive control file at runtime.

If you specified multiple source netlist files, Calibre Interactive creates a file named *_source.net_* with .INCLUDE statements for each netlist file. The file*_source.net_* is used as the source netlist for the Source Path statement in the control file. Files entered in the **Database** > **Library** tab on the LVS Options pane are listed before files in the **Netlist** tab on the Inputs pane. If the source netlist format was MIXED, the SPICE netlist that was translated from the Verilog input is listed last.

### Related Topics

Communication with the Design Tool

Using Calibre Interactive to Perform LVS

Source Path [Standard Verification Rule Format (SVRF) Manual]

Source Primary [Standard Verification Rule Format (SVRF) Manual]

Source System [Standard Verification Rule Format (SVRF) Manual]

## Setting Up the Verilog Translator (v2lvs)

You can specify Verilog netlists as input source netlists for your Calibre nmLVS run. The Verilog netlist is translated into an LVS SPICE netlist before Calibre nmLVS is run using the v2lvs translator provided in the Calibre software tree.

The settings made in the Verilog Translator dialog box are saved in the runset.

### Procedure

1. Choose **Setup** > **Verilog Translator**.
2. Click the **Libraries** tab in the Setup Verilog Translator dialog box. Enter the location of the library files (Verilog or SPICE) in the corresponding "Verilog Library Files" or "SPICE Library Files" fields. If you have included SPICE files, specify the path in the "Include SPICE files" field.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Click the **Strings** tab in the Setup Verilog Translator dialog box to specify options with string parameters. Table 7-2 lists the command options for this tab. Enter the values for the options in the corresponding text field.

**Table 7-2. Verilog String Options**

| Dialog Box Option | Equivalent v2lvs Option | Description |
|---|---|---|
| Prefix for gate level primitives | -p *prefix* | Adds the *prefix* to Verilog gate-level primitive cells. This allows construction of specific SPICE subcircuits that correspond to the Verilog gates used. |
| Prefix for unnamed pins & gates | -u *unnamed_pin_prefix* | Adds the *prefix* to unnamed pin connections in module instantiations, and to unnamed primitive and gate instantiations. |
| Change array delimiters to | -a *delimiter1 delimiter2* | Changes the array delimiter characters to the ones you specify. The default delimiters are brackets ( [] ). Use this option to change characters used to describe Verilog net selections as SPICE nets. |
| Change illegal SPICE characters to | -c *char1* [*char2*] | Sets the substitution characters for characters illegal in SPICE. Parameter *char1* replaces “$”, “,” and “-”. Parameter *char2* replaces “/”. When characters normally illegal in SPICE are encountered in Verilog, the *char1* and *char2* characters are substituted. |
| Connect pins with 0 value to net | -s0 *groundnet* | Specifies that the default global ground is changed to *groundnet*. The *groundnet* is declared as a .GLOBAL net. |
| Connect pins with 1 value to net | -s1 *powernet* | Specifies that the default global ground is changed to *powernet*. The *powernet* is declared as a .GLOBAL net. |
| More v2lvs program options | None | Allows you to specify any v2lvs program option that you wish to pass to the translator. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

4. Click the **Options** tab in the Setup Verilog Translator dialog box to specify command options for the v2lvs translator. The options are described in Table 7-3 along with the equivalent command line options used by the v2lvs translator.

**Table 7-3. Verilog Translator Options**

| Dialog Box Option | Equivalent v2lvs Option | Description |
|---|---|---|
| Use aggregate mode for bus bits | -lsr<br>-lsp | Unrolls bus bits from the input Verilog files. If this option is not selected, bus net names are treated as non-aggregate nets. |
| Preserve backslash in escaped identifiers | -b | Preserves the backslash character in escaped identifiers. |
| Treat warnings as errors | -w0<br>-w1<br>-w2<br>-w3<br>-w4 | Sets warning levels to be treated as errors when encountered:<br>Level 0: No warnings.<br>Level 1: Issues warnings about skipped modules and multiple declarations of modules.<br>Level 2: All Level 1 warnings and calls to undeclared modules and pin array width wider than port.<br>Level 3: All level 2 warnings and called port array width mismatches and unsupported compiler directives.<br>Level 4: All level 3 warnings and all ignored constructs. |
| Don’t connect supply0/supply1 nets to global supply | -sk | supply0 and supply1 nets are not connected to the global power/ground. |
| Add pins to each subckt | -addpin | Adds specified pins to each subcircuit, if they do not already exist. This option is not compatible with the “Use positional pin order” option. |
| Use positional pin order (don’t use $PINS) | -i | Sets instance pin connections in positional order. If this option is not selected, pins are connected with $PINS. |
| Create numbered nets for unconnected pins | -n | All unconnected pins get numbered net connections starting from 1000. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 7-3. Verilog Translator Options (cont.)**

| Dialog Box Option | Equivalent v2lvs Option | Description |
|---|---|---|
| Generate xRC source template file | -t | The translator outputs an xRC source template file to the svdb database directory. |
| Prefer Calibre-CB license during license search | -cb | The translator will attempt to check out a Calibre Cell/Block license. If it fails, the translator will attempt to check out a Calibre nmLVS license. |
| Always translate before running Calibre | None | Translator is run before every Calibre run. If unchecked, the translator does not run if the Verilog files are older than the translated SPICE file. |

5. Click **OK** to close the dialog box.

### Related Topics

V2LVS [Calibre Verification User's Manual]

Source Netlist Input in Calibre Interactive nmLVS

## Providing Additional SPICE Netlist Files with the Source Netlist

You specify additional SPICE netlist files on the **Database** > **Library** sub-tab of the LVS Options pane. This may be needed to provide INCLUDE files for the main source netlist. This option is also available for Calibre Interactive PERC and Calibre Interactive PEX.

**Tip**

To specify both Verilog and SPICE files for the source netlist, select the MIXED format on the **Netlist** tab of the Inputs pane, as described in "Source Netlist Input in Calibre Interactive nmLVS" on page 250.

### Prerequisites

- Calibre Interactive nmLVS is open.

### Procedure

1. Click the **LVS Options** button or choose **Setup** > **LVS Options**.
2. Click the **Database** tab and the **Library** sub-tab.
3. Use the buttons next to "Additional SPICE Files" to specify the required SPICE files.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Results

The additional files are included before the netlist specified on the Inputs pane in the **Netlist** tab. Calibre Interactive creates a file named *_source.net_* with .INCLUDE statements for each netlist file; **_source.net_** is used as the source netlist for the Source Path statement in the control file.

### Related Topics

Source Netlist Input in Calibre Interactive nmLVS

## Supplying Hcell Data

Hcell files are used in hierarchical Calibre nmLVS to define the correspondence between layout and source cell names.

### Prerequisites

- Calibre Interactive nmLVS is open.

### Video

If you don't have an hcell list, you can use Calibre Interactive to create an initial hcell list, as demonstrated in the video "How To Create an Initial Hcell List with Calibre Interactive."

### Procedure

1. Click the **Inputs** button on the left panel.
2. Select "Hierarchical" for the "Run" type.
3. Click the H-Cells tab.
4. Select the desired options:
   - **Match cells by name** — Specifies automatic correspondence by name for cells in Calibre nmLVS-H comparison and should only be used if the layout cells actually contain the same devices as the source subcircuits of the same name.

     This option applies to LVS comparison (the -automatch command line option); this option does not apply to netlist extraction.

     If "Use H-Cells file" is also specified, automatched cells are added to the hcell list.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Use H-Cells file** — Specifies to use an hcell file. The hcell file specifies cell correspondence for hierarchical LVS comparison. This option preserves hcells as subcircuits during both netlist extraction and LVS comparison.
- **Automatically generate an H-Cell list** — Specifies to automatically generate an hcell list during circuit extraction using the Query Server Tcl shell. This option adds the -genhcells option to the command line. This option cannot be specified with "Match cells by name."

  Check "Use default thresholds as hcell selection criteria" to use the default thresholds when selecting hcells. This option adds the -select keyword to the command line.

  Check "Use H-Cells Query Server Tcl script" to execute a user-defined Tcl script to generate the hcell list. If not specified, a default Tcl script is used.

  If "Use H-Cells file" is also specified, the hcell lists are concatenated.

  For more details, see the -genhcells option in the section "Calibre nmLVS and Calibre nmLVS-H Command Line" of the *Calibre Verification User's Manual*.
- **Expand H-Cells to resolve false discrepancies** — Specifies to examine LVS discrepancies and, in certain cases, to automatically perform additional comparison runs with fewer hcells. Enable the checkbox and select "Yes" to use this option. Uses the LVS EXPAND ON ERROR {YES | NO } statement.

**Note**

The option "Match cells by number of placements" does not apply during LVS comparison or netlist extraction. It only applies during hcell analysis in Calibre Interactive, as described in "Performing Hcell Analysis in Calibre nmLVS" on page 260.

## Related Topics

Hcell [Standard Verification Rule Format (SVRF) Manual]

Performing Hcell Analysis in Calibre nmLVS

"Hcells [Calibre Verification User's Manual]

Hcell and Hierarchical Analysis [Calibre Query Server Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Performing Hcell Analysis in Calibre nmLVS

You can use Calibre Interactive nmLVS to calculate the effectiveness of hcells in reducing the total hierarchical instance count of your design. You can also extract and view the hierarchical structure of the design.

**Setting up the Hcell Analysis Run** . . . . . 260
**H-cells Tab for Viewing Hcell Analysis Results** . . . . . 261

## Setting up the Hcell Analysis Run

A number of options affect the output of your hcell analysis.

### Prerequisites

- Calibre Interactive nmLVS is open with the layout netlist and source netlist specified (Inputs pane, **Layout** and **Netlist** tabs).

  If the layout netlist file does not exist it will be extracted from the layout. If the source netlist file does not exist and “Export from schematic viewer” is enabled, the source netlist will be exported from a connected schematic viewer.

### Procedure

1. Click the **Inputs** button on the left panel.
2. Click the **H-Cells** tab.
3. Select hcell options. You must select at least one of the following in order to perform an analysis.
   - **Match cells by name** — Select this option if you want hcell candidates to be selected and analyzed based upon cell names that match between layout and source. (Do this only if you are certain the subcircuit definitions match between such cells.)

     This option applies to both hcell analysis and LVS comparison; it does not apply to netlist extraction.
   - **Match cells by number of placements** — Select this option if you want to compare cells with the same number of placements and equivalent pin counts as hierarchical entities. Use caution when specifying this option, since it can identify invalid hcell pairings, as described in hcells::placementmatch and NETLIST PLACEMENTMATCH in the *Calibre Query Server Manual*.

     This option only applies to hcell analysis in Calibre Interactive; it does not apply during LVS comparison or circuit extraction.
   - **Use H-Cells File** — Select this option if you have an existing hcell file that you want to analyze.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

This option applies to hcell analysis, LVS comparison, and netlist extraction.

4. Click the **H-cells Analysis** button. The H-Cells Analysis Options dialog box appears.
5. Select analysis options in the H-Cells Analysis Options dialog box:
   - **Evaluate Current H-Cells during analysis** — Calibre Interactive evaluates cells in the specified hcells file against a specific threshold during the analysis.
   - **Total Hier Inst Count Savings Threshold** — This setting defaults to 30%, and specifies the effectiveness threshold for saving an hcell candidate. The effectiveness is a measurement of computational savings based on the reduction of Total Hierarchical Instance Count (THIC) for additional hcells. Only hcell candidates that collectively are at or above this threshold are saved after analysis. This setting specifies the first parameter (*THIC_threshold*) for the NETLIST EVALUATION THRESHOLD and "hcells::select -threshold" Query Server commands.
   - **Total Flat Inst Count Percentage Threshold** — This setting specifies the cutoff threshold for instance count evaluation. If a cell's instance count is greater than (total flat instance count * inst_count_threshold / 100), it appears in the hcell list even if it does not improve the computational effectiveness. The default is 1%. This setting specifies the second parameter (*flat_threshold*) for the NETLIST EVALUATION THRESHOLD and "hcells::select -threshold" Query Server commands.

   If both "Evaluate Current H-Cells during analysis" and "Match cells by name (automatch)" are selected, the hcell analysis considers cells from the specified hcell list and cells determined by automatch.
6. Click the **OK** button to continue. Calibre Interactive nmLVS performs the analysis.

   Results are shown in the **H-Cells** tab, as explained in the section "H-cells Tab for Viewing Hcell Analysis Results" on page 261.

## H-cells Tab for Viewing Hcell Analysis Results

Hcell analysis results are shown in four sub-tabs within the H-Cells tab of the Calibre Interactive nmLVS Inputs pane.

See the task "Setting up the Hcell Analysis Run" on page 260 for a description of how to perform Hcell analysis.

When Hcell analysis is complete, the results are displayed as shown in Figure 7-2 in the following four sub-tabs:

- **H-Cells Evaluation** — This tab displays the results of your Hcells evaluation. Use the **Save H-cells** button to save the selected hcell names to an hcell file. When asked whether to use this as an hcell file, selecting yes automatically enables the "Use H-Cells file" option.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Source Analysis** — This tab displays information about the contribution of devices present given certain conditions and the resulting memory savings. Hover your mouse over the column heading for a description of the data provided.
- **Layout Analysis** — This tab displays information similar to the **Source Analysis** tab, but for the layout netlist.
- **Hierarchy** — This tab displays the hierarchical structure of both the layout and source.

**Figure 7-2. H-Cells Analysis Result Tabs**

| Total Hier Inst Count Layout | Total Hier Inst Count Source | Inst Count In this Cell Layout | Inst Count In this Cell Source | Saved By this Cell | Total Savings So Far | Potential Remaining Saving | Layout Cell Name | Source Cell Name |
|---|---|---|---|---|---|---|---|---|
| 707 | 481 | - | - | 0.0% | 0.0% | 47% | pllclk | pllclk |
| 576 | 350 | - | 12 | 22% | 22% | 32% | DFFRHQS1 | DFFRHQS1 |
| 448 | 288 | - | 22 | 21% | 38% | 14% | vco2phase | vco2phase |
| 406 | 250 | - | 40 | 11% | 45% | 3.7% | phaseDetPath | phaseDetPath |
| 407 | 251 | - | 42 | - | 45% | 4% | vco | vco |

Also see "Hcell and Hierarchical Analysis" in the *Calibre Query Server Manual*.

# Specifying Automatic Waivers for ERC

You can specify the ERC rule checks to waive and the cells in which the waivers apply using the Calibre Auto-Waivers flow.

Calibre Auto-Waivers allows various classifications for the waivers that can be applied, such as how to handle multiple waivers that may intersect a result region and hierarchical interactions. See "Calibre Auto-Waivers" and "ERC Automatic Waivers" in the *Calibre Auto-Waivers User's and Reference Manual* for complete information.

Calibre Auto-Waivers controls are on the Inputs pane in the **Waiver** tab; see "Calibre Auto-Waivers in Calibre Interactive" on page 201.

See "ERC Results Tab" in the *Calibre RVE User's Manual* for information on waiving ERC results using Calibre RVE.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Specifying Outputs for a Calibre nmLVS Run

The Outputs pane in Calibre Interactive nmLVS specifies the LVS report file, the SVDB (Standard Verification Database), and its options.

## Procedure

1. Click the Outputs button on the left panel to display the Outputs pane.

2. Click the **Report/SVDB** tab.
3. (Optional) Enable “View Extraction Report after LVS finishes.”
4. (Optional) Enable “View Summary Report after LVS finishes.” If LVS Summary Report is not specified in the rule file, then Calibre Interactive adds the statement to the control file with a report name of *lvs.summary*.
5. Specify the LVS report filename.
6. (Optional) Enable the “View Report after LVS finishes” option to automatically display the LVS report in a text editor window at the completion of your LVS run.
7. Enable the “Create SVDB Database” option and specify the SVDB directory name, as desired. Creating the SVDB database is recommended, since it is required to run Calibre RVE.
8. Enable “Start RVE after LVS finishes” to automatically start Calibre RVE at the completion of your Calibre nmLVS run. If you are creating a runset for a batch run, you may want to enable “Start RVE after batch run” on the **Misc** tab of the Setup Preferences dialog box.

   When Calibre RVE is started from Calibre Interactive nmLVS, the settings on the Run Control pane are used to set the -turbo option for Calibre RVE; see the section



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

"Multithreaded Parallel Processing for Calibre RVE Operations (-turbo Option)" in the *Calibre RVE User's Manual*.

9. If you want to output data for parasitic extraction, enable the "Generate data for Calibre-xRC" option.

   Once enabled, the netlist file extracted by Calibre nmLVS is automatically copied to the SVDB directory so that SVDBs generated by Calibre nmLVS can be seamlessly imported into PEX.

10. If you want to generate ASCII cross-reference (XDB) files in the SVDB database, select the "Generate ASCII cross-reference files" option.

    If the run type is flat the -nxf and -ixf command line options are also used.

11. If you want to generate Calibre Connectivity Interface (CCI) data in the SVDB database, select the "Generate Calibre Connectivity Interface data" option. This option requires a Calibre Connectivity Interface (CI) product license.

    If desired, select "Suppress pin-location information" to add the NOPINLOC keyword to the Mask SVDB Directory statement. This option is only available when "Generate Calibre Connectivity Interface data" is selected.

**Tip**

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

### Related Topics

LVS Report [Standard Verification Rule Format (SVRF) Manual]

Mask SVDB Directory [Standard Verification Rule Format (SVRF) Manual]

# Running Calibre Interactive nmLVS

Use the following procedure to start a Calibre nmLVS run from Calibre Interactive—nmLVS.

### Prerequisites

- All necessary input data and files.

### Procedure

1. Enter all necessary information in the Rules, Inputs, Outputs, and Run Control panes.
2. Choose **Setup** > **LVS Options** and select each of the option tabs to set additional options. All left panel buttons will be green when the necessary information is entered.
3. (Optional) To specify triggers, choose **Setup** > **Preferences**, then the **Triggers** tab. See "Trigger Functions in Calibre Interactive" on page 657.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

4. Click **Run LVS** on the left panel of Calibre Interactive.

   Calibre Interactive first verifies that the source netlist is complete; the source netlist check is case insensitive. If an internal pre-execution trigger is defined, the check for a complete source netlist is done after the trigger executes. You can turn off the source netlist check by disabling the setting "Check source netlist is complete before running LVS" (**Setup > Preferences**, **Misc** tab). See "Miscellaneous Preferences" on page 156 for information on the source netlist check.

   When the run starts, the view switches to the Transcript pane, and any run output, including errors and warnings, is displayed there.

5. View the transcript pane and examine the results of the run. See "Transcript Pane in Calibre Interactive" on page 99.

**Tip**

You can generate and view a control file without running Calibre by holding down the Ctrl key and clicking **Run LVS**. In addition, the calibre command line for the run is written to the transcript pane.

## Results

If an error occurs in Calibre Interactive before the Calibre run is started, the error is displayed in a pop-up window or in the Calibre Quality Agent for Calibre Interactive.

When the Calibre run starts, the view switches to the Transcript pane, and any run output, including errors and warnings, is displayed here. See "Transcript Pane in Calibre Interactive".

If "Start RVE after LVS finishes" is checked, the results database is opened in Calibre RVE when the run is complete; see "Using Calibre RVE for LVS" in the *Calibre RVE User's Manual*. In addition, the Calibre Interactive GUI may open a report file in a text viewer when the run completes; see "Viewing and Editing Text Files in Calibre Interactive" on page 125.

## Related Topics

Specifying a Rule File for a Calibre nmLVS Run

Inputs for a Calibre Interactive nmLVS Run

Specifying Outputs for a Calibre nmLVS Run

Run Control, Remote Execution and Licensing Options

Calibre nmLVS Options in Calibre Interactive

Setting Database and General Options in Calibre Interactive

LVS Results [Calibre Verification User's Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Run Control, Remote Execution and Licensing Options

You can control how your Calibre nmLVS job is run by configuring options related to performance, remote and distributed computing, and license acquisition.

These topics are covered in detail in "Run Control Setup and Licensing" on page 681.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre nmLVS Options in Calibre Interactive

You can specify options for your Calibre nmLVS run from the LVS Options pane. These settings generate SVRF specification statements in your control file. The options you specify in the GUI override the rule file settings.

Some options are similar for all applications—see "Setting Database and General Options in Calibre Interactive" on page 93.

If the **LVS Options** button is not visible, choose the **Setup > LVS Options** menu item to add the **LVS Options** button to the left side of the Calibre Interactive.

**Setting Power Supply Options** . . . . . 267
**Aborting on Sconnect Errors** . . . . . 268
**Ignoring Ports** . . . . . 269
**Specifying Calibre nmLVS Report Options** . . . . . 269
**Reporting Seed Promotions** . . . . . 270
**Controlling Logic Gate Recognition** . . . . . 271
**Setting Split Gate and Parallel MOS Reduction Options** . . . . . 271
**Setting Filter Unused Device Options** . . . . . 272
**Changing Trace Property Rules** . . . . . 274
**Changing LVS Box Statements** . . . . . 276
**Invoking Short Isolation** . . . . . 278
**Setting Options for Debug with Calibre RVE for LVS** . . . . . 280
**Running ERC Checks** . . . . . 282
**Selecting ERC Checks** . . . . . 284

## Setting Power Supply Options

You use the **Supply** tab to specify power and ground net names and to choose related options.

### Procedure

1. Click LVS Options to display the Options pane. If the LVS Options button is not available, choose Setup > LVS Options.
2. Click the Supply tab.
3. Enter the power and ground net names using one of these methods:
   - Enter a space-separated list of net names in the text field.
   - Click the **Load from file** button and specify a file. The file should be an ASCII text file with the net names specified as a space-separated list or given on separate lines.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For example, loading the following file:

```
A B C D
NET_IN1
NET_OUT2
```

results in the list “A B C D NET_IN1 NET_OUT2” being entered in the text field.

4. Enable “Abort LVS on power/ground net errors,” if desired. This option causes Calibre nmLVS to abort on encountering any power/ground net problems. An abort can be caused by any of the following reasons:
   - The power or ground names conflict with another name found on the net.
   - The same power or ground name was found on two disjoint nets.
   - An unattached power or ground name was found.
   - A stamp error occurred in a power or ground net.

   **Tip**

   You must specify power and ground net names if you invoke LVS logic gate recognition.

### Related Topics

LVS Abort On Supply Error [Standard Verification Rule Format (SVRF) Manual]

LVS Power Name [Standard Verification Rule Format (SVRF) Manual]

LVS Ground Name [Standard Verification Rule Format (SVRF) Manual]

## Aborting on Sconnect Errors

If you use Sconnect statements in your rule file and encounter errors during your run, you can have Calibre nmLVS abort automatically on soft connection conflicts.

### Procedure

1. Click the LVS Options button on the left panel to display the Options pane. If the LVS Options button is not available, choose Setup > LVS Options.
2. Click the Supply tab.
3. Enable “Abort LVS on Softchk errors.”

### Related Topics

LVS Abort On Softchk [Standard Verification Rule Format (SVRF) Manual]

Sconnect [Standard Verification Rule Format (SVRF) Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Ignoring Ports

Choose this option to disable the use of cell ports as initial correspondence points for connectivity comparison.

## Procedure

1. Click the LVS Options button on the left panel to display the Options pane. If the LVS Options button is not available, choose Setup > LVS Options.
2. Click the Supply tab.
3. Enable “Ignore layout and source ports during comparison.”

## Related Topics

LVS Ignore Ports [Standard Verification Rule Format (SVRF) Manual]

# Specifying Calibre nmLVS Report Options

You can adjust the information included in the LVS Report file. The settings on the **Report** tab are all optional, and in most cases the default settings are adequate.

## Procedure

1. (Optional) To view the rule file settings for report options, click the **Rules** button in the left panel and click the **Load** button. This loads the rule file settings into the GUI.
2. Click the LVS Options button on the left panel to display the Options pane. If the LVS Options button is not available, choose Setup > LVS Options.
3. Click the Report tab.
4. Select options for the LVS Report:

   For most Calibre nmLVS runs, you should not specify any of these options, unless you actually need the information they provide. The default Calibre nmLVS report is usually all that is needed for most LVS runs.

   a. Click the button next to “LVS Report Options.”

   b. Select the desired report options. The button label changes to show the selected options.

      You can also disable report options that are specified in the rule file.
5. Set the maximum discrepancy count:

   Select the setting for “Max. discrepancies printed in report” as follows:

   - **All** — Include all discrepancies in the Calibre nmLVS report.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Count** — Enter a value for the maximum discrepancy count. The default of 50 is adequate for most cases.

6. Specify to report seed promotions:

   The Seed Promotions Report provides detailed information regarding the promotion of individual polygons to higher-level cells for the purpose of device recognition. This is an advanced option for Calibre nmLVS-H. It should be used for test purposes only.

   a. Enable "Create Seed Promotions Report."

   b. Specify the maximum number of polygons to include in the report for each seed promotion.

### Related Topics

LVS Report Option [Standard Verification Rule Format (SVRF) Manual]

LVS Report Maximum [Standard Verification Rule Format (SVRF) Manual]

LVS Report [Standard Verification Rule Format (SVRF) Manual]

LVS Show Seed Promotions [Standard Verification Rule Format (SVRF) Manual]

LVS Show Seed Promotions Maximum [Standard Verification Rule Format (SVRF) Manual]

## Reporting Seed Promotions

You can specify to report seed promotions during your Calibre nmLVS run. The Seed Promotions Report provides detailed information regarding the promotion of individual polygons to higher-level cells for the purpose of device recognition. This is an advanced option for Calibre nmLVS-H and may increase run time. It should be used for test purposes only.

### Procedure

1. (Optional) To view the rule file settings for report options, click the **Rules** button in the left panel, and click the **Load** button. This loads the rule file settings into the GUI,
2. Click LVS Options to display the Options pane. If the LVS Options button is not available, choose Setup > LVS Options.
3. Click the Report tab.
4. Enable "Create Seed Promotions Report."
5. Specify the maximum number of polygons to include in the report for each seed promotion.

### Related Topics

LVS Show Seed Promotions [Standard Verification Rule Format (SVRF) Manual]

LVS Show Seed Promotions Maximum [Standard Verification Rule Format (SVRF) Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Controlling Logic Gate Recognition

These options control how Calibre performs logic gate recognition.

## Procedure

1. Click the **LVS Options** button on the left panel to display the Options pane. If the **LVS Options** button is not available, choose **Setup > LVS Options**.
2. Click the Gates tab on the LVS Options pane.
3. Choose the desired gate recognition option from the following selections:
   - **Recognize all gates** — Uses the ALL keyword in LVS Recognize Gates.
   - **Recognize simple gates** — Uses the SIMPLE keyword in LVS Recognize Gates.
   - **Turn off** — Uses the NONE keyword in LVS Recognize Gates.
4. Decide whether to select the "Mix subtypes" option. If you do not select this option, transistors in series-parallel branches must all have the same subtype.

**Tip**

You must specify power and ground net names if you enable gate recognition. For more information on this topic, refer to "Setting Power Supply Options" on page 267.

## Related Topics

Setting Filter Unused Device Options

Setting Split Gate and Parallel MOS Reduction Options

Logic Gate Recognition [Calibre Verification User's Manual]

# Setting Split Gate and Parallel MOS Reduction Options

These options control how split gates are reduced.

## Procedure

1. Click **LVS Options** on the left panel to display the LVS Options pane. If th**e LVS Options** button is not available, choose **Setup > LVS Options**.
2. Click the Gates tab on the LVS Options pane.
3. Select GUI or rule file control of split gate reduction as follows:
   - **"Use settings from rules" enabled** — Use rule file setting and disable GUI control. No further GUI settings are needed.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **"Use settings from rules" disabled** — Use GUI settings, as described in the following steps.

4. Enable or disable split gate reduction. Some options depend on the setting for "Reduce split gates."

   - **Enable "Reduce split gates"** — This specifies "LVS Reduce Split Gates YES" to the Calibre Interactive control file. The following options are available:

     - **Semi-split gates** — Adds the SEMI ALSO keyword.
     - **Only when within tolerance** — Adds the WITHIN TOLERANCE keyword.
     - **Mix types during reduction** — Adds the MIX TYPES keyword.
     - **Series-parallel split gate** — Adds the SP ALSO keyword.
     - **Only when the input order is same** — Adds the SAME ORDER keyword.

     The controls for "Reduce parallel MOS transistors" and "Short equipotential nodes together" are not available when "Reduce split gates is enabled."

   - **Disabled "Reduce split gates"** — This specifies "LVS Reduce Split Gates NO" to the Calibre Interactive control file. The following options are available:

     - **Reduce parallel MOS transistors** — As described; see LVS Reduce Parallel MOS.

       Note: This setting is on by default when "Reduce split gates" is enabled, since split gate reduction implies parallel MOS reduction.

     - **Short equipotential nodes together** — Short together the internal nets of basic split gate structures if the nodes are equipotential; see LVS Short Equivalent Nodes.

### Related Topics

Setting Filter Unused Device Options

Controlling Logic Gate Recognition

LVS Reduce Split Gates [Standard Verification Rule Format (SVRF) Manual]

LVS Reduce Parallel MOS [Standard Verification Rule Format (SVRF) Manual]

LVS Short Equivalent Nodes [Standard Verification Rule Format (SVRF) Manual]

## Setting Filter Unused Device Options

You can filter out specific unused devices in your layout and source using the Filter Unused Device Options table in the **Gates** tab of the LVS Options pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. (Optional) To view the rule file settings for filter options, click the **Rules** button in the left panel, and click **Load**. This loads the rule file settings into the GUI.
2. Click **LVS Options** on the left panel. If the **LVS Options** button is not available, choose **Setup** > **LVS Options**.
3. Click the Gates tab on the LVS Options pane.
4. Specify GUI or rule file control of the LVS Filter Unused Option [LAYOUT | SOURCE] statement as follows:
   - **To use rule file settings** — Disable the "Filter Unused Device Options" checkbox.
   - **To enable GUI control** — Enable the "Filter Unused Device Options" checkbox.
5. If GUI control is enabled, do the following to specify options:
   a. Select the desired filter options, either individually or for multiple options as follows:
      - **Individual selection** — Specify device filtering using the checkboxes in the L (layout) or S (source) columns.
      - **Multiple selections** — Shift- and Ctrl-click in the filter description text to select multiple filter types, then right-click to display a filter options menu and make the desired selection.

      **Tip**
      Click to the right of the L or S in the column header to sort the options by selected state. This is useful to list all selected options at the top.

   b. (Optional) You can specify that an option only apply to specific device types and subtypes, rather than to all devices. Click in the Devices column and use the list box to select the desired device(subtype) combinations.

      Note: Some filter options are specific only to certain device types. In those cases, the device(subtype) specified in the Devices column must be consistent with the types implied by the LVS Filter Unused command option.

      **Note**
      If the "Filter Unused Device Options" checkbox is enabled, but no options are selected in the table, then this unselects all LVS Filter Unused Option settings, including any that might be specified in the rule file.

## Related Topics

LVS Filter Unused Option [Standard Verification Rule Format (SVRF) Manual]

Controlling Logic Gate Recognition



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Setting Split Gate and Parallel MOS Reduction Options

Automatically Loading a Rule File

# Changing Trace Property Rules

You can use the **Properties** tab on the Options pane in Calibre Interactive to override the Trace Property statements in your rule file. You can add new statements, edit existing statements, and disable statements. This feature is available in Calibre Interactive nmLVS, Calibre Interactive PERC, and Calibre Interactive PEX.

## Prerequisites

- Calibre Interactive nmLVS, Calibre Interactive PERC, or Calibre Interactive PEX is open.
- A rule file is specified on the Rules pane.

## Procedure

1. Click **LVS Options**, **PERC Options**, or **PEX Options** to display the Options pane. If the button is not available, choose **Setup** > **LVS Options**, **Setup** > **PERC Options**, or **Setup** > **PEX Options**.
2. Click the **Properties** tab.
3. Disable “Use settings from rules.”

   When “Use settings from rules” is disabled, only the Trace Property statements enabled in the Trace Property table are used in the Calibre run.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 7-3. Properties Tab in Calibre Interactive nmLVS**

**Note**

Trace Property statements that include trace property computation procedures are not loaded into the Trace Property table.

4. Use the following table to determine the steps to take depending on the action you want:

**Table 7-4. Changing Trace Property Statements in Calibre Interactive nmLVS**

| Action | Steps |
|---|---|
| Disable all Trace Property statements in the rule file. | Do one of the following:<br>• Do *not* load the rule file to populate the Trace Property table.<br>• If the Trace Property table is already populated, disable all checkboxes in the Include column or click to remove all statements. |
| Disable or enable specific Trace Property statements. | 1. Click **Load** on the Rules pane to populate the table.<br>2. Use the checkboxes in the Include column to select the statements that are included in the run. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 7-4. Changing Trace Property Statements in Calibre Interactive nmLVS**

| Action | Steps |
|---|---|
| Edit the parameters for a Trace Property statement. | 1. Click **Load** on the Rules pane to populate the table.<br>2. Click in the Value column and edit the parameters. |
| Add a new Trace Property statement. | 1. (Optional) Click **Load** on the Rules pane to populate the table with rule file statements. Do this if you want to run the rule file Trace Property statements in addition to the new statement.<br>2. Click to add a new statement, then fill in the parameters in the Value column. |

5. (Optional) Save the runset with **File > Save Runset**. The statements in the Trace Property table and the Include column settings are saved to the runset.

### Results

When the Trace Property "Use settings from rules" checkbox is disabled, only the statements enabled in the Trace Property table are used in the Calibre run.

**Note**

When you load a rule file, all corresponding GUI settings are set to the settings in the rule file. Therefore, any custom settings in the Trace Property table are reset to rule file settings when you load the rule file.

## Changing LVS Box Statements

You can use Calibre Interactive to override the LVS Box related statements in your rule file; this is done on the **LVS Box** tab on the LVS Options pane. You can add new statements, edit existing statements, and disable statements. This feature is available in Calibre Interactive nmLVS, Calibre Interactive PERC, and Calibre Interactive PEX.

### Prerequisites

- Calibre Interactive is open.
- A rule file is specified on the Rules pane.

### Procedure

1. Click **LVS Options** to display the Options pane. If the button is not available, choose **Setup > LVS Options**; similar steps apply for Calibre Interactive PERC and PEX.
2. Click the **LVS Box** tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Disable “Use LVS Box related settings from rules.”

   When “Use LVS Box related settings from rules” is unchecked, only the LVS Box and LVS Black Box Port statements enabled in the **LVS Box** tab are used in the Calibre run.

4. The following table lists possible actions:

**Table 7-5. Changing LVS Box Statements in Calibre Interactive**

| Action | Steps |
|---|---|
| Disable all LVS Box related statements in the rule file. | Do one of the following:<br>• Do *not* load the rule file to populate the LVS Box and LVS Black Box Port tables.<br>• If the tables are already populated, disable all checkboxes in the Include columns or click to remove all statements. |
| Disable or enable specific statements. | 1. Click **Load** on the Rules pane to populate the tables.<br>2. Use the checkboxes in the Include columns to select the statements that are included in the run. |
| Edit the parameters for a statement. | 1. Click **Load** on the Rules pane to populate the tables.<br>2. Click in the Value column and edit the parameters. |
| Add a new LVS Box or LVS Black Box Port statement. | 1. (Optional) Click **Load** on the Rules pane to populate the tables with rule file statements. Do this if you want to run the rule file statements in addition to the new statement.<br>2. Click to add a new statement, then fill in the parameters in the Value column. |

**Note**

The LVS Black Box Port statement is required if LVS Box BLACK is specified. If you add a LVS Box BLACK statement, you must also add the LVS Black Box Port statement—the GUI does not do this automatically.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 7-4. LVS Box Tab in Calibre Interactive nmLVS**

5. (Optional) Edit the settings for LVS Black Box Port Depth.

### Results

When the “Use LVS Box related settings from rules” checkbox is disabled, only the statements enabled in the tables in the **LVS Box** tab are used in the Calibre run. The statements enabled in the tables are written to the Calibre Interactive control file.

Although multiple cells and pin swap lists can be specified in a single LVS Box statement, such statements in the rule file are broken up into multiple LVS Box statements when loaded into the table in **LVS Box** tab.

**Note**

When you load a rule file, all corresponding GUI settings are set to the settings in the rule file. Therefore, any custom settings in the LVS Box tables are reset to rule file settings when you load the rule file.

## Invoking Short Isolation

Calibre has a short isolation feature to help pinpoint the location of layout shorts between texted nets. You can specify this option in Calibre Interactive.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Click LVS Options to display the LVS Options pane. If the LVS Options button is not available, choose **Setup > LVS Options**.
2. Click the Shorts tab.
3. Enable the “Run short isolation” option and specify a type, if desired.
   - **Yes** — Perform short isolation, subject to specified cell and name filters. Adds “LVS ISOLATE SHORTS YES”.
   - **Power Shorts Only** — Isolate shorts between power net labels. Adds “&& NAME LVS_POWER_NAME” and disables the name option.
   - **Ground Shorts Only** — Isolate shorts between ground net labels. Adds “&& NAME LVS_GROUND_NAME” and disables the name option.
   - **Power Ground Shorts** — Isolate shorts between power net labels, shorts between ground net labels, and shorts between power-ground nets. Adds “&& NAME LVS_POWER_NAME LVS_GROUND_NAME” and disables the name option.
   - **IO Shorts** — Isolate shorts between signal nets and shorts between signal and power-ground nets. Adds “&& !NAME LVS_POWER_NAME LVS_GROUND_NAME” and disables the name option.

   Note: Enabling short isolation automatically adds the option SI to the Mask SVDB Directory statement.
4. Select general options:

   **Output shorts by layer** — Group results by layer. If not selected, polygons from all layers involved in the short are grouped together into one report cluster. This option adds the BY LAYER keyword.

   **Run flat short isolation** — Select this to choose flat short isolation. Hierarchical isolation is the default.

   **Output shorts by cell** — Report the cell of origin for every polygon written to the shorts isolation database. This option adds the BY CELL ALSO keyword to the LVS ISOLATE SHORTS command.

   **Exclude contact polygons** — Remove contact polygons from the output. This option adds the NO CONTACTS keyword.
5. Choose desired Hierarchical Options:

   These options are not available if the “Run flat short isolation” option is selected.

   a. Specify the depth of short isolation.
      - **Isolate shorts in top cell only** — Equivalent to the CELL PRIMARY keyword set in LVS Isolate Shorts.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Isolate shorts in all cells** — Equivalent to the CELL ALL keyword set in LVS Isolate Shorts.

b. Specify how to combine the depth and name sub-options:

- **AND** — A short must satisfy both conditions. For example, Calibre reports the short only if it occurs in the top-level cell *and* is between nets named VDD and VSS.
- **OR** — Any short satisfying either condition is reported. For example, report the short if it is between any top-level nets *or* if the short is between VSS and VDD at any level in the hierarchy.
- **NOT** — Select this setting to exclude the names specified in step c.

c. Choose the desired name option. This option specifies how the tool processes text names when identifying shorts. Adds the NAME keyword.

- **Between all names** — Isolate shorts between all text object names.
- **Between names** — Isolate shorts between the net names specified in the text field.

d. Enable "Identify common paths," if desired. This option instructs the tool to identify a common path, if it exists, shared by multiple paths between shorted text objects. This option adds the ACCUMULATE keyword.

**Tip**

Invocation of short isolation creates a short isolation database which can be viewed using Calibre RVE for LVS. Calibre RVE for LVS includes tools for isolating the short, such as highlighting by cell, layer, or net. See "Interactive Short Isolation" and "Verifying Short Repairs" in the *Calibre RVE User's Manual*.

6. Enable "Maximum Short Results," if desired. If set to "Count," this option limits the maximum number of isolated short paths that are identified and reported to the provided value. When not enabled or set to "All," all short paths are reported.

### Related Topics

LVS Isolate Shorts [Standard Verification Rule Format (SVRF) Manual]

Mask SVDB Directory [Standard Verification Rule Format (SVRF) Manual]

# Setting Options for Debug with Calibre RVE for LVS

You can use Calibre Interactive nmLVS to set up the run so that you can view the results in Calibre RVE for LVS. Certain settings are needed to enable various debug features.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Prerequisites

- A rule file for Calibre nmLVS.
- Case sensitivity is handled correctly; see "Case Sensitivity in Calibre RVE" in the *Calibre RVE User's Manual* for information.

## Procedure

1. Open Calibre Interactive nmLVS; see "Using Calibre Interactive to Perform LVS" on page 237.
2. Specify the output database (SVDB) as follows:
   a. Click **Outputs** on the left panel.
   b. Enable "Create SVDB database."
   c. Specify the "SVDB Directory" pathname in the text entry field.

   This adds the following statement to the Calibre Interactive control file:

   ```
   Mask SVDB Directory svdb QUERY
   ```

   where *svdb* is the path to your SVDB directory. The QUERY option is specified by default in Calibre Interactive. You can specify additional options on the Outputs pane, if desired.
3. (Optional) Enable short isolation and short verification as follows:
   a. Choose **Setup > LVS Options** to open the LVS Options pane.
   b. Enable "Run Short Isolation."
   c. Enable the following settings: "Output shorts by layer," "Output shorts by cell," "Isolate shorts in all cells," and "Between all names."

   See the section "Invoking Short Isolation" on page 278 for further details.
4. (Optional) Calibre RVE for LVS can display Design Fix Suggestions for certain types of common LVS problems. Do the following to enable Design Fix Suggestions in the results database:
   a. Choose **Setup > LVS Options** to open the LVS Options pane.
   b. Click the **Report** tab.
   c. For the LVS Report Options selection, enable "FX=Show detailed error analysis."

   **Tip**

   When using Calibre Interactive, the default GUI setting of "None" for LVS Report Options overrides your rule file settings unless you click **Load** on the Rules pane to load the rule file settings.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

If your SVDB (standard verification database) takes a long time to load into Calibre RVE, remove options A, AV, C, and CV from LVS Report Option if they are present. These options increase the size of the SVDB. Much of the information from these options is reported in the DETAILED INSTANCE CONNECTIONS section of the LVS Report. Also make sure that LVS Report Maximum ALL is not used.

For a complete description of setting up a Calibre nmLVS run, see the *Calibre Verification User's Manual*.

### Related Topics

Examining Connectivity Extraction Results [Calibre RVE User's Manual]

Interactive Short Isolation [Calibre RVE User's Manual]

Examining LVS Comparison Results [Calibre RVE User's Manual]

Using Design Fix Suggestions for LVS Comparison Results [Calibre RVE User's Manual]

# Running ERC Checks

Calibre can execute Electrical Rule Check (ERC) statements during a Calibre nmLVS run.

### Prerequisites

- A Calibre nmDRC license in addition to the Calibre nmLVS license. Refer to the *Calibre Administrator's Guide* for more details.

### Procedure

1. Click **LVS Options** on the left panel to display the LVS Options pane. If the **LVS Options** button is not available, choose **Setup > LVS Options**.
2. Click the ERC tab on the LVS Options pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Enable the “Run ERC” option.

4. Click the Select Checks button to select the ERC checks that are executed. Refer to the “Selecting ERC Checks” section for more information.
5. Specify the ERC Results File.
6. (Optional) Enable “Output ERC errors in cell space.” When enabled, this setting causes ERC results to be reported in cell space. This statement is necessary if you want to highlight ERC results in the cell they are reported in rather than the top cell, as described in the section “Viewing Results in Cell Space” in the *Calibre RVE User’s Manual*. This option is enabled by default. This option is required for Calibre Auto-Waivers.
7. (Optional) Enable “Report results from pseudocells in” and select either “Pseudocells” or “Top cell.” See the tooltip for descriptions.
8. Specify the ERC summary report filename.

## Related Topics

Electrical Rule Checks [Calibre Verification User's Manual]

LVS Execute ERC [Standard Verification Rule Format (SVRF) Manual]

ERC Results Database [Standard Verification Rule Format (SVRF) Manual]

ERC Summary Report [Standard Verification Rule Format (SVRF) Manual]

ERC Maximum Results [Standard Verification Rule Format (SVRF) Manual]

ERC Maximum Vertex [Standard Verification Rule Format (SVRF) Manual]

ERC Cell Name [Standard Verification Rule Format (SVRF) Manual]

Including Other Rule Files and SVRF Commands in Calibre Interactive



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Calibre Interactive nmLVS ERC Check Selection in Batch Mode

# Selecting ERC Checks

You use the Select Checks dialog box to specify which ERC checks to execute during your Calibre nmLVS run. You can also choose whether to abort an Calibre nmLVS run if an ERC check fails. Selections in the Select Checks dialog box are not saved to the runset and must be reset each time you open Calibre Interactive nmLVS.

You can also use a customization file to select ERC checks, as described in "Customization File to Select ERC Rule Checks" on page 651. If you are using a customization file, the customization settings initialize the settings in the Select Checks dialog box. Changes made in the Select Checks dialog box take precedence over the initial settings from the Customization GUI.

If you are running Calibre Interactive in batch mode, you must use a customization file to select the ERC checks or select the ERC checks in the rule file. The Select Checks dialog box is not available in batch mode.

## Prerequisites

- A Calibre nmDRC license. Refer to the *Calibre Administrator's Guide* for more details.
- ERC checks are enabled, as described in "Running ERC Checks" on page 282.

## Procedure

1. Click **LVS Options** on the left panel to display the Options pane. If the **LVS Options** button is not available, choose **Setup > LVS Options**.
2. Click the ERC tab on the LVS Options pane.
3. Click the Select Checks button. The Select Checks dialog box is displayed.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 7-5. Select Checks Dialog Box**

4. For each of the individual rules or rule groups, click the rule or group name checkbox to cycle between the following choices:
   - **Green Check Mark** — Execute this check during the Calibre nmLVS run.
   - **Orange Check Mark** — Abort the Calibre nmLVS run if this check fails.
   - **Red X** — Do not execute this check during the Calibre nmLVS run.
5. Choose **View > Close** to close the Select Checks dialog box.

## Related Topics

Electrical Rule Checks [Calibre Verification User's Manual]

ERC Select Check [Standard Verification Rule Format (SVRF) Manual]

Calibre Interactive nmLVS ERC Check Selection in Batch Mode



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Device Signature Creation with Calibre Interactive nmLVS

Calibre Interactive nmLVS allows you to easily generate unique signature strings for devices. Device signatures enable the matching of devices in the layout to a specific geometric pattern, rather than matching to just the layers present.

For information on device signatures, signature generation, and signature comparisons, refer to "Device Signatures" in the *Calibre Verification User's Manual* and the Device command in the *SVRF Manual*. It will help if you are familiar with device signature concepts as you fill out the Edit Parameters for Device Dialog Box.

**Generating Device Signatures for a Calibre nmLVS Rule File Using Calibre Interactive 286**

**Layers File for Device Signature Creation . . . . . 290**

**Edit Parameters for Device Dialog Box . . . . . 293**

## Generating Device Signatures for a Calibre nmLVS Rule File Using Calibre Interactive

You can use Calibre Interactive nmLVS to create a device signature file. Device signatures enable the matching of devices in the layout to a specific geometric pattern, rather than matching to just the layers present. The signature creation process from the GUI is a standalone step, separate from your LVS comparison or netlist extraction run. You can save the device commands with the generated device signatures to a new file or append to an existing file. An Include statement is typically used to include the Device statements with signatures in the rule file for your Calibre nmLVS run.

### Prerequisites

- A Calibre Advanced Device Properties (ADP) product license in addition to a Calibre nmLVS license.
- A layout file in a geometric format specified on the Inputs pane in the **Layout** tab; see "Layout Input in Calibre Interactive nmLVS" on page 244.
- A rule file specified on the Rules pane. See "Specifying a Rule File for a Calibre nmLVS Run" on page 239.
- Transcript output in Calibre Interactive must be enabled, which is the default behavior. Transcript output is controlled with the "Disable transcript output" setting (**Setup** > **Preferences**, **Disable** tab).
- (Optional) A Layers File. The Layers File specifies the device layers and the layers used to generate the device signature. The file is used to populate fields in the Edit Parameters



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

for Device dialog. See "Layers File for Device Signature Creation" on page 290 for the syntax of this file.

## Procedure

1. Click **Inputs** on the left panel to display the Inputs pane.
2. Click the **Signatures** tab, which is shown in Figure 7-6. There are two sub-tabs:
   - **Device Statements sub-tab** — Lists the parameters for the Device signature statements in table form, including the Device statement used for signature generation.
   - **Device Signatures sub-tab** — Lists the Device statements with the created signatures.

**Figure 7-6. Signatures Tab in LVS Inputs Pane**

3. (Optional) Specify and load the Layers File.

   If you do not have a Layers File, the parameters to use in the signature creation can be entered manually in the Edit Parameters for Device Dialog Box.

   a. Specify the Layers File in the text entry field. See "Layers File for Device Signature Creation" on page 290 for the syntax of this file.

   b. Click **Load** to load the Layers File data.

   The layers are available in dropdown lists in the appropriate sections of the Edit Parameters for Device Dialog Box.

   Note: A pop-up note displays if any layer parameters are missing in the Layers File; since all layers may be entered manually, this note is for information only and does not interfere with the signature creation process.

4. Click the **Add Device** button to add a new device, or right-click in the Device Signature area. The Edit Parameters for Device dialog box opens.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. Enter all required layers and options for signature creation. See "Edit Parameters for Device Dialog Box" on page 293 for a definition of the fields in the dialog.

6. Click **Save Device** or **Save As New Device** to update the entries in the Device Statements table.

   If you enabled "Use only seed layers with attached text label" in the Edit Parameters for Device dialog box, the Label column is filled in with the With Text statement that defines the labeled seed layer.

   Note: The new statement is *not* added if any of the following errors occur:

   - The device statement already exists in the Device Statements table.
   - There are one or more unspecified parameters for the device statement.

7. To add additional device statements, repeat Steps 4 to 6.

8. (Optional) If your rule file includes an LVS Summary Report statement with a Tcl script for custom report generation, it is required to generate the SVDB. Click the **Outputs** button and enable "Create SVDB Database."

9. Click **Create Signatures** to generate signatures for all of the device statements selected in the Device Statements table.

   **Tip**

   You can hold the Ctrl key and click the **Create Signatures** button to view the rule file that will be used in the signature creation run.

   If a duplicate device definition is detected between the created device statements from the GUI and the device statements in the original rule file, an error message displays and the run is aborted. Remove the duplicate statement from either the rule file or the Device Statements table, and click **Create Signatures** again.

   You can view the Device statements with the created signatures in the **Device Signatures** sub-tab. If more than one signature is possible for a device, separate Device statements for each signature are listed in the **Device Signatures** sub-tab.

10. Click **Save Signatures** to save the generated signature statements in a file. A dialog prompts you for the filename; you can choose to overwrite the file or append to it.

    If you enabled "Use only seed layers with attached text label" in the Edit Parameters for Device dialog box, the With Text statement that defines the labeled seed layer is also written to the saved file. The With Text statement can also be viewed in the Label column of the **Device Statements** sub-tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Results

At the end of the signature generation process, the Device statements with the generated signatures are saved in a signature file. To use the created signatures in your Calibre nmLVS run, you can do one of the following:

- Use the SVRF Include statement to include the signature file in your rule file.
- In Calibre Interactive, choose **Setup** > **LVS Options**, then the **Include** tab to include the signature file in your Calibre nmLVS run.

**Note**

A SVDB is created if the rule file includes an LVS Summary Report statement and “Create SVDB Database” is enabled.

## Related Topics

Device Signatures” [Calibre Verification User's Manual]

Device [Standard Verification Rule Format (SVRF) Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Layers File for Device Signature Creation

Input for: The **Signatures** tab on the Inputs pane in Calibre Interactive nmLVS.

The Layers File is used to define the device types, device layers, pin names and pin layers for signature creation. The entries in the Layers File populate the fields of the Edit Parameters for Device dialog box in Calibre Interactive. You create the Layers File with a text editor.

Table 7-6 defines the entries in the Layers File for device signature creation. An example is provided in the "Examples" section.

## Format

- ASCII text file.
- One entry per line.
- Comment lines start with a "//".
- Layer entries may have an optional *abbreviation* entry. If present, the layer *abbreviation* is used in the Edit Parameters for Device Dialog Box.
- For the CONTEXT_LAYER entry only, a list of layers enclosed by double quotes may be mapped to an abbreviation.

## Parameters

**Table 7-6. Layers File for Device Signature Creation Contents**

| Contents | Definition |
|---|---|
| *DEVICE_TYPES<br><device_type><br>… | List the device types you plan to create signatures for. The device types should be those permitted in the SVRF Device command. |
| *DEVICE_MODELS<br><device_model><br>… | List the device models you plan to create signatures for. The device models should be those permitted in the SVRF Device command. |
| *REFERENCE_LAYERS<br><reference_layer> [<abbreviation>]<br>… | List the layers that serve as the reference_layer parameter for signature creation in the Device statement. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 7-6. Layers File for Device Signature Creation Contents (cont.)**

| Contents | Definition |
|---|---|
| *CONTEXT_LAYERS<br><context_layer> [<abbreviation>]<br>…<br><br>or:<br>"<layer> <layer> …" [<list_name>] | List the layers that serve as the context_layer parameter for signature creation in the Device statement.<br><br>A list of context layers enclosed in double quotes may also be mapped to a layer list name. If the context layer list is selected in the Edit Parameters for Device Dialog Box, the list contents are enclosed in braces ({ }) when shown in the device statement; the braces are not included in the device statement written to the control file. |
| *PIN_LAYERS<br><pin_layer> [<abbreviation>]<br>… | List the layers that serve as the pin_layer parameter in the Device statement. |
| *PIN_NAMES<br><pin_name> [<abbreviation>]<br>… | List the layers that serve as the pin_name parameter in the Device statement. |
| *SEED_LAYERS<br><seed_layer> [<abbreviation>]<br>… | List the layers that serve as the seed layer (or device_layer parameter in the Device statement). |
| *TEXT_MODEL_LAYERS<br><model_text_layer> [<abbreviation>]<br>… | List the layers that serve as the text_layer parameter for the TEXT MODEL LAYER keyword in the Device statement. |
| *SEED_LABEL_LAYERS<br><seed_text_layer> [<abbreviation>]<br>… | If you are using seed shapes that are texted for the purpose of signature generation, list the layers that have text labels for the seed layer. See Table 7-7 for an explanation of how this parameter is used with the option "Use only seed layers with attached text label" in the Edit Parameters for Device Dialog Box. |

## Examples

```
// layers file for device signature creation
*DEVICE_TYPES
mp
C

*DEVICE_MODELS
PMOS_VTL
PMOS_VTH
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
*PIN_NAMES
G
S
D
pos
neg

*PIN_LAYERS
gate
psrcdrn
cappos
capneg

*SEED_LAYERS
pgate
hvtpgate
capbody

*REFERENCE_LAYERS
poly
fieldpoly fpoly
capbody

*CONTEXT_LAYERS
pimplant
metal2
via2
// create context layer list
"metal1 metal2 metal3" METAL_1_2_3
// the seed label layer entry is optional
// *SEED_LABEL_LAYERS
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Edit Parameters for Device Dialog Box

To access: **Edit Device** button on the **Signatures** tab on the Inputs pane of Calibre Interactive nmLVS,

Use the Edit Parameters dialog box to specify the layers, pin names, and options used for device signature creation in Calibre Interactive.

## Description

You can specify and load a Layers File to pre-populate the dropdown lists and tables in the dialog box; see "Layers File for Device Signature Creation" on page 290. However, you can type in any entry field or table row to enter an arbitrary parameter, so loading a Layers File is not necessary.

**Figure 7-7. Edit Parameters for Device Dialog Box**

## Objects

**Table 7-7. Device Parameter Fields for Create Device Signature**

| Device Parameter Field | Description |
|---|---|
| Device Name | The component name. |
| Seed Layer | The layer the device body is on; also known as the device layer. |
| Reference Layer | The reference layer defines the area around the seed shape used for deriving a signature.<br>Once a reference layer is used with a specific set of context layers, it may only be used with that set of context layers in the rule file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 7-7. Device Parameter Fields for Create Device Signature (cont.)**

| Device Parameter Field | Description |
|---|---|
| Pin Name<br>Pin Layer | The pin name and layer pairs necessary for the Device statement.<br>Use the right-click menu to select or unselect all pins and to toggle the pin selection. You can also Shift-click in the header of the Select column to select/unselect all rows. |
| Swappable With | Enter the pin names that are interchangeable with this pin.<br>If this entry is left blank, the default pin swappability rule applies: pins on the same layer are interchangeable. |
| Context Layers | Shapes on the context layer that intersect the reference layer are considered in the signature.<br>You can specify multiple context layers. The order of the context layers matters, since the generated signature reflects the given order. Change the order of context layers by dragging the layer entry.<br>Use the right-click menu to select or unselect all layers and to toggle the layer selection. You can also Shift-click in the header of the Select column to select/unselect all rows.<br>You can use the layers file to map a list of context layers to a list name; see the *CONTEXT_LAYER entry in "Layers File for Device Signature Creation" on page 290.<br>Note: The seed layer and pin layers are not part of the signature calculation unless the layers are included in the context layer list. |
| Use only seed layers with attached text label<br>Text Label<br>Text Layer | When enabled, only device seed layers with the specified text label on the given text layer are used in the Device statement. The text label string is appended to the seed layer name and a new layer is created which is used in the Device statement. These statements are used:<br>`<seed_layer>_<label> = <seed_layer> WITH TEXT`<br>`"label" <text_layer>`<br>`DEV <device> <seed_layer>_<label> ...` |
| Model Name | Use the specified model name (or component subtype). |
| Text Model Layer | Use the TEXT MODEL LAYER keyword and parameter, |
| Netlist Model | Use the NETLIST MODEL keyword and parameter |
| Netlist Element | Use the NETLIST ELEMENT keyword and parameter. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 8
# Using Calibre Interactive to Perform Parasitic Extraction

You can use Calibre Interactive to run Calibre parasitic extraction.

**Introduction to Calibre Interactive PEX** . . . . . 295
**Specifying a Rule File for PEX Run** . . . . . 299
**Inputs for a PEX Run** . . . . . 301
**Outputs for a PEX Run** . . . . . 312
**Calibre Interactive PEX Execution** . . . . . 320
**Run Control, Remote Execution and Licensing Options** . . . . . 322
**Setting PEX Options** . . . . . 324
**Setting LVS and Netlist Extraction Options in a PEX Run** . . . . . 348

## Introduction to Calibre Interactive PEX

The Calibre Interactive PEX Graphical User Interface (GUI) is an interface to the Calibre xRC, Calibre xACT, and Calibre xACT 3D batch parasitic extraction tools. The Calibre Interactive PEX GUI allows you to specify command line options and selected rule file statements that execute the desired Calibre PEX application.

Calibre Interactive facilitates performing the following types of extraction:

- **Transistor-level extraction** — Generates a flat netlist which includes the original intentional devices in your circuit along with the specified parasitic elements.
- **Gate-level extraction** — Generates a netlist which includes the top-level cells (logic gates) in your circuit, along with the specified parasitic elements associated with the interconnection of those cells.
- **Hierarchical extraction** — Generates a netlist which includes connectivity, intentional device and parasitic element information for all cells in the hierarchy of your design.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Input Files You Use With Calibre Interactive PEX

The following input files are required when using the Calibre Interactive PEX GUI:

**Table 8-1. Files Required by the Calibre Interactive PEX GUI**

| Name | Description |
|---|---|
| Layout database | The layout database, or design, is a physical representation of your IC. |
| SVRF rule file | A text input file you must create using Standard Verification Rule Format (SVRF) statements and operations. It controls the parasitic extraction tool's execution. |
| Source netlist (optional) | The source netlist, or schematic, is an electrical representation of your IC. Use it for input if you want source names rather than layout names in your output. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Calibre Interactive PEX Workflow

The following figure shows a typical workflow in Calibre Interactive PEX:



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 8-1. Generic PEX Flow**

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Using Calibre RVE for PEX [Calibre RVE User's Manual]

# Specifying a Rule File for PEX Run

You need to specify a rule file for any PEX run. You can specify a rule file automatically with a runset or you can specify one manually using this procedure.

### Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.

### Procedure

1. Click **Rules** on the left panel.

2. Specify the run directory name.
3. Specify the rule filename. You can use the **View** button to view or edit the rule file.
4. (Optional) Click **Load** to populate the GUI with the settings in the rule file.

   A rule file can also be specified in a runset. After loading that runset, the rule file is included in the control file but its settings are not be loaded into the GUI until you click **Load**.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

The setting "Read Inputs/Outputs fields while loading rules file" (**Setup** > **Preferences**, **Misc** tab) controls whether or not rule file settings populate the GUI options on the Inputs and Outputs panes when **Load** is clicked. This setting is enabled by default. See Table 3-20 on page 156 for more information.

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

5. (Optional) View the layer derivation tree at the bottom of the Rules pane; see "Layer Derivations Tree" on page 126 for information.

## Related Topics

About Rule Files

About Calibre Interactive Runsets

Including Other Rule Files and SVRF Commands in Calibre Interactive



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Inputs for a PEX Run

The Inputs pane includes the tabs for each type of input. Use these tabs to specify the inputs needed for your PEX run.

**Table 8-2. Inputs Pane for PEX**

| Tab | Purpose | Required |
|---|---|---|
| Layout | Specify the layout design database and format. See "Specifying the Layout Input in Calibre Interactive PEX" on page 301 and "Specifying Multiple Layouts for a PEX Run" on page 303. | Yes |
| Netlist | Specify the schematic netlist. This is used when you want the extracted netlist to use source schematic names, or you want to control cell extraction for an ADMS extraction run. See "Specifying Schematic Netlist Input" on page 304. | No |
| H-Cells | Specify cell lists for performing non-flat extraction. See "Defining Hcell and Xcell Input (Gate-Level, Hierarchical Extraction and ADMS Only)" on page 307. | No |
| Blocks | Specify blocks for ADMS extraction. See "Manipulating Cell Extraction Options Using the Blocks Tab" on page 308 and the *Calibre xRC User's Manual*. | No |
| Probes | Specify probe points. See "Specifying Probe Points in Calibre Interactive PEX" on page 310 and PEX Probe File in the *SVRF Manual*. | No |

# Specifying the Layout Input in Calibre Interactive PEX

A layout database is required for PEX runs. The source of your layout data can vary depending on whether your Calibre Interactive session is connected to a viewer or not.

**Tip**

You can set a preference for the setting of "Export from layout viewer" that depends on whether your Calibre Interactive session is connected to a layout viewer. By default, this preference is set so that" Export from layout viewer" is enabled if Calibre Interactive is started from a layout viewer, regardless of the saved runset value for the option. See "Export from Viewer Preferences" on page 152.

## Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Click **Inputs** on the left panel.
2. Click the Layout tab.
3. Select a layout format using the "Format" button dropdown list:
   - **GDSII** — GDS format
   - **OASIS** — OASIS format
   - **OPENACCESS** — see "Providing OpenAccess Input" on page 90

   For a full list of supported layout databases, refer to the "Layout Databases" section of the *Calibre Verification Users Manual*. Also see "Third Party Database Options" on page 80.
4. Specify the input source for the layout database, and the filename. There are two choices for the input source:
   - A connected layout viewer (GDS and OASIS format only) — Go to sub-step 4.a.
   - Layout database files — Go to sub-step 4.b.

   a. A connected layout viewer supplies the layout (GDS and OASIS format only).

      i. Enable "Export from layout viewer."

      This specifies the layout viewer as the input source.

      Batch runs: In addition, you must also disable the setting "Ignore layout and source export in batch mode" (**Setup** > **Preferences**, on the **Misc** tab). Also see "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104 for details on the layout viewer connection for a batch run.

      ii. Verify the filename for the extracted layout database in the "File" field.

      **Caution**

      When "Export from layout viewer" is enabled, the file specified in the "File" field is overwritten if it exists already. Do not enter a tapeout database in the "File" field, or use the extracted layout named in the "File" field for tapeout or any other use.

      When exporting a layout database from a connected viewer, the layout is extracted and saved to the file named in the "File" field. This filename is generated automatically based on the setting on the **Templates** > **Inputs** sub-tab of the Setup Preferences dialog box (**Setup** > **Preferences**).

      If the filename is red, the file does not exist and Calibre Interactive creates the file. If the filename is green, the file exists and will be overwritten. Enter another filename if you want to keep the existing file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

iii. Go to the next step, Step 5.

b. Layout database files are the source of the layout.

i. Disable the setting “Export from layout viewer,” if present.

This specifies a file as the input source, rather than a layout viewer.

ii. Specify the filename for the layout database in the “File” field.

You can specify the layout filename using any of the following methods:

- Enter it in the “File,” “Library,” or “Directory” field, depending on the database format.
- Specify it in the rule file, and load the rule file. Note: This assumes the “Read Inputs/Outputs fields when loading rules file” option is selected in the **Misc** tab of the Setup Preferences dialog box (**Setup** > **Preferences**).
- Load a saved runset that has the layout file specified.

5. (Optional) If you have multiple layout inputs for your PEX run, you can specify them using the **Database** > **Library** tab on the PEX Options pane. This is valid both for layout data from a file and exported from a layout viewer. See “Specifying Multiple Layouts for a PEX Run” on page 303.

6. Specify the top cell name.

If your layout is exported from a connected layout viewer, the top cell is determined from the current cell in the viewer.

### Related Topics

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Primary [Standard Verification Rule Format (SVRF) Manual]

Layout System [Standard Verification Rule Format (SVRF) Manual]

Export from Viewer Preferences

Setting Database and General Options in Calibre Interactive

## Specifying Multiple Layouts for a PEX Run

If your primary cell has references to other additional database files you may need to load multiple layouts. You can specify multiple layouts on the **Database** > **Library** sub-tab of the PEX Options pane (**Setup** > **PEX Options**).

Calibre Interactive checks the format of the additional database files and only includes files that match the format of the primary layout database.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Prerequisites

- Main layout database in GDSII or OASIS format.
- Additional layout files in GDSII or OASIS format. Files compressed with gzip may be used. The additional layout files must be the same format as the main layout database.
- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.

### Procedure

1. Click **Inputs** on the left panel, then click the **Layout** tab.
2. Enter a layout file in the "File" field of the **Layout** tab.
3. Select "GDSII" or "OASIS" for the "Format," as appropriate.
4. Enable the PEX Options pane by choosing **Setup > PEX Options**.
5. Click the **Database** tab; then click the **Library** sub-tab.
6. Use the buttons next to "Additional Layout Files" to add the required files.

   The additional files are added to the Layout Path statement. Only files that match the format of the primary database are added.
7. (Optional) In order to receive a warning if the additional layout files are not the same format as the main layout database, select **Setup > Preferences**, click the **Misc** tab, and enable "Warn when library layout file format differs from primary layout format." The warning is displayed when you click the **Run PEX** button.

   Additional layout files that are not the same format as the primary database are ignored.

### Related Topics

Specifying the Layout Input in Calibre Interactive PEX

## Specifying Schematic Netlist Input

Schematic netlist input is optional for a PEX run. A schematic netlist is needed if you want to use source schematic names instead of layout names in the extracted netlist. Also see the "Use Names From" setting on the Outputs pane.

Netlist data is also needed when using the **Blocks** tab on the Inputs pane to control cell extraction options for an ADMS run. See "Manipulating Cell Extraction Options Using the Blocks Tab" on page 308.

### Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Click the Netlist tab on the Inputs pane.

2. Select the input source for the schematic netlist with the "Export from schematic viewer" option as follows:

   - **Enabled** — A connected schematic viewer supplies the schematic source.

     Batch runs: In addition, you must also disable the setting "Ignore layout and source export in batch mode" at the **Setup** > **Preferences** on the **Misc** tab. Also see "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104 for details on the viewer connection for a batch run.

   - **Disabled** — Schematic netlist files provide the source schematic.

3. Specify the format of the source netlist using the "Format" button dropdown list.

   If Calibre Interactive is connected to a schematic viewer and "Export from schematic viewer" is enabled, specify the netlist format of the schematic that your viewer outputs.

   Select from these format options:

   - **SPICE** — The netlist is in SPICE format.

   - **VERILOG** — The netlist is in Verilog format. If you specify Verilog as the source format, Calibre Interactive calls a utility named v2lvs to translate the Verilog netlist to SPICE. Choose **Setup** > **Verilog Translator** to set up the translation parameters. For more information about setting up v2lvs, see the section "Setting Up the Verilog Translator (v2lvs)" on page 254 in this manual. The SPICE netlist name is taken from the "Files" field.

   - **MIXED** — Select the "Mixed" option to use a mixed mode of both SPICE and Verilog files. The application displays separate netlist entry fields for the SPICE and Verilog files. This option cannot be specified when "Export from schematic viewer" is enabled.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

4. Specify the netlist filename. The behavior depends on the setting for “Export from schematic viewer”:
   - "Export from schematic viewer" is enabled. Go to sub-step 4.a.
   - "Export from schematic viewer" is disabled. Go to sub-step 4.b.

   a. “Export from schematic viewer” is enabled.

      This specifies the schematic viewer as the input source for the schematic netlist.

      i. Verify the filename for the extracted schematic in the “Files” field. This file will be overwritten if it exists already.

         When exporting a source netlist from a connected schematic viewer, the schematic is extracted and saved to the file named in the “File” field. This filename is generated automatically based on the setting on the **Templates** > **Inputs** tab of the Setup Preferences dialog box (**Setup** > **Preferences**).

         If the filename is red, the file does not exist and Calibre Interactive creates the file. If the filename is green, the file exists and will be overwritten. Enter a different filename if you want to keep the existing file.

      ii. Go to the next step, Step 5.

   b. “Export from schematic viewer” is disabled.

      This specifies that a file is the input source for the schematic, rather than a schematic viewer.

      i. Specify the filename for the source netlist in the “Files” field.

      ii. Go to the next step.

5. Enter the top cell name in the “Top Cell” field. If you are connected to a viewer, this field is filled in automatically.

6. (Optional) Specify additional SPICE netlist files. This may be necessary to provide INCLUDE files for the source netlist.

   a. Click the **PEX Options** button or choose **Setup** > **PEX Options**.

   b. Click the **Database** tab and the **Library** sub-tab.

   c. Use the buttons next to “Additional SPICE Files” to specify the required SPICE files.

**Note**

Calibre Interactive verifies that the source netlist is complete before starting the run. You can turn off this check by disabling “Check source netlist is complete before running PEX” (**Setup** > **Preferences**, on the **Misc** tab). See “Miscellaneous Preferences” on page 156 for information on the source netlist check.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Results

The appropriate statements are written to the Calibre Interactive control file at runtime.

If you specified multiple source netlist files, Calibre Interactive creates a file named *_source.net_* with .INCLUDE statements for each netlist file; *_source.net_* is used as the source netlist for the Source Path statement in the control file. Files entered in the **Database** > **Library** tab on the PEX Options pane are listed before files in the **Netlist** tab on the Inputs pane.

## Defining Hcell and Xcell Input (Gate-Level, Hierarchical Extraction and ADMS Only)

The hcell list specifies the correspondence between layout and source cell names for hierarchical Calibre nmLVS operations. The xcell list is used during parasitic extraction and defines the cells that are preserved during extraction.

**Note**

You do not need to perform these steps for Transistor-level extraction.

### Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.

### Procedure

1. Click **Inputs** on the left panel of Calibre Interactive PEX.
2. Click the H-Cells tab.
3. (Optional) If you are using source names for the extracted netlist, you can enable "Match cells by name." This option adds the -automatch option to the "calibre -lvs" command line.

   This option specifies automatic correspondence by name for cells in Calibre nmLVS-H comparison, in addition to the cells listed in the hcell file. The -automatch option does not apply to netlist extraction.

   **Note**

   The "Match cells by name (LVS automatch)" option should only be used if the layout cells contain the same devices as the source subcircuits of the same name.

4. If using source names for the extracted netlist, enable "Use LVS H-Cells" file and specify the hcell filename. When performing netlist extraction using source names, the "calibre -lvs" step for netlist extraction and PHDB creation uses the specified hcell file.

   This option is not required if you specified "Use Names From" as "LAYOUT" on the Outputs pane in the **Netlist** tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. Specify the xcell filename in the field “PEX x-Cells file.” The “calibre -xrc” steps use the xcell file.

**Tip**

You can use an hcell file as the xcell file. However, an xcell list can have additional options that are not allowed in an hcell list.

### Related Topics

Hierarchy Control with Xcells [Calibre xRC User's Manual]

Hcells [Calibre Verification User's Manual]

Calibre nmLVS and Calibre nmLVS-H Command Line [Calibre Verification User's Manual]

Performing Hcell Analysis in Calibre nmLVS

## Manipulating Cell Extraction Options Using the Blocks Tab

The **Blocks** tab on the Inputs pane displays an alphabetical list of cells found in the netlist files specified in the **Netlist** tab. Each cell entry contains an expansion control and the number of placements of each subcell found within its parent cell.

You can select the extraction format, extraction mode, cross-annotate flag, and output netlist file for each extracted cell.

A P next to a cell name in the hierarchy tree indicates that the cell is listed in the xcell file as a primitive cell (the -P option), and the cell contents will not be extracted.

**Tip**

You can discard changes in the **Blocks** tab and refresh the view at any time by clicking the **Cell Options** button in the top right corner of the tab and choosing **Restore from Xcells File** and **Re-read Netlist Files**.

### Prerequisites

- The Extraction Type is Hierarchical or ADMS on the Outputs pane.
- Calibre Interactive PEX is open; see “Invoking Calibre Interactive Tools” on page 64.

### Procedure

1. Click the **Inputs** button on the left panel, then click the **Blocks** tab.
2. Locate the cell you want to specify extraction options for by expanding the tree view as needed. If needed, click **Cell Options** in the top right corner of the tab and choose **Show All Cells**.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Click the cell name to mark the cell for extraction; this is indicated with a ☑ next to the cell name.

4. Right-click in the cell name and choose **Format** to choose the extraction format.

   A light green box with extraction information is displayed to the right of the cell. Additional extraction options are available once the format is specified.

5. Right-click in the cell name again and specify additional options as desired from the following list:

   - **Mode** — Extraction mode.
   - **Cross-annotate** — Flag the cell for cross-annotation.
   - **Specify Output Netlist File** — The output netlist filename for the cell.
   - **Specify Top Verilog Module** — Only available for SDF format. Specify the top Verilog module for the cell. The module is passed to the timing tool chosen in the Setup Delay Calculation dialog box. For more information refer to "Setting Delay Calculation" on page 345.
   - **Filter X** — Only available for SDF format. Enable this checkbox to specify the FILTERX option for the PEX Netlist ADMS statement. The setting is disabled by default.

   Any changes in the **Blocks** tab cause an asterisk to appear at the top of the pane to the right of the "Netlist Blocks for ADMS/Hier Extraction" label. The asterisk indicates that cell information has been modified and may be different from what appears in the xcell file.

6. Click **Cell Options** in the top right corner of the tab and choose **Save to Xcells/Hcells Files** when done making changes. This writes your changes to the hcell and xcell files specified on the Inputs pane in the **H-Cells** tab.

   The asterisk to the right of the "Netlist Blocks for ADMS/Hier Extraction" label disappears when your changes have been saved to the xcell and hcell files.

   A view of the **Blocks** tab on the Inputs pane is shown in the following figure.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 8-2. Inputs > Blocks Tab in Calibre Interactive PEX**

### Related Topics

Hierarchy Control with Xcells [Calibre xRC User's Manual]

# Specifying Probe Points in Calibre Interactive PEX

You can specify probe points in your parasitic model using Calibre Interactive. Probe points are used to verify timing from specific points on each net.

### Prerequisites

- You are running Calibre Interactive PEX, and have specified the rule file, input layout, and input netlist, as described in the following sections:
  - "Specifying a Rule File for PEX Run" on page 299.
  - "Specifying the Layout Input in Calibre Interactive PEX" on page 301.
  - "Specifying Schematic Netlist Input" on page 304.

### Procedure

1. Click the **Inputs** button on the left panel.
2. Click the **Probes** tab.
3. Do one of the following, depending on whether you have an existing probe file:
   - Click the button to select and load an existing probe file.
   - Click the button to manually add a probe point.

   The interface should look similar to that shown in the following figure.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 8-3. PEX Probes Tab**

4. Click in each column to specify the required parameter. You can hover your mouse over each column header to view a tooltip for the parameter.

   To provide the probe location, click the button to choose a location in a connected layout design tool. To enter coordinates, click next to the button and enter a space-separated coordinate pair.

5. (Optional) Specify the options “Protect Probes” and “Replace Probes with Ports” below the Probe Points table.

   See PEX Probe File in the *SVRF Manual* for details.

6. When done, click the button to save the probe file.

## Related Topics

PEX Probe File [Standard Verification Rule Format (SVRF) Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Outputs for a PEX Run

The Outputs pane in Calibre Interactive PEX specifies the extraction mode and level and options for the extracted netlist, reports, and output database options.

**Specifying the Extraction Mode and Level for a Calibre Interactive PEX Run** . . . . . 312
**Specifying Extracted Netlist Options for a Calibre Interactive PEX Run** . . . . . 313
**Specifying Report Options for a Calibre Interactive PEX Run** . . . . . 315
**Specifying Database Output Options for a Calibre Interactive PEX Run** . . . . . 316
**Naming the Extracted Parasitic Netlist Based on the Extraction Type and Format** . . 317

## Specifying the Extraction Mode and Level for a Calibre Interactive PEX Run

The Outputs pane in Calibre Interactive PEX specifies the extraction mode and level.

### Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.

### Procedure

1. Click **Outputs** on the left panel.
2. Select the Extraction Mode as follows:
   - **xACT 3D** — Use the xACT 3D fieldsolver. See the *Calibre xACT User's Manual*.
   - **xRC** — Use model-based extraction. See the *Calibre xRC User's Manual*.
   - **xACT 3D/xRC Hybrid** — Select "xACT 3D," then enable the "Hybrid" checkbox. This runs hybrid xACT 3D/rule-based extraction (the -hybrid command line option). Choose **Setup** > **PEX Options** then the **xACT 3D** tab to select the nets for extraction.

   The following exceptions apply when "xACT 3D" or "xACT 3D/xRC Hybrid" is specified:

   - If "R" or "No R/C" is selected as the Extraction Type, the -3d command line option is not used and the "Accuracy" cannot be specified.
   - The Calibre-xRC CB license cannot be used; the -cb command line option is not used even if "Use Calibre-xRC CB license" is specified on the Run Control pane.
3. If the extraction mode is "xACT 3D" or "xACT 3D/xRC Hybrid," select the accuracy as follows:

   **200** — Less accurate, with faster runtime. This is the default.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**600** — More accurate, with slower runtime.

**MEMS** — Runs at the highest accuracy needed for MEMS designs.

4. Choose the extraction level (Transistor Level, Gate Level, Hierarchical, In-Context or ADMS) using the first button in the Extraction Type section.

   See Run Control, Remote Execution and Licensing Options for instructions on specifying the Calibre-xrc CB and Calibre-CB licenses for your extraction run.

5. Choose the type of parasitics to extract using the second button in the Extraction Type section. See Setting PEX Options for extracted netlist options.

6. Choose the type of parasitic inductance to extract using the third button in the Extraction Type section. Inductance extraction is controlled using the Inductance page in the PEX Options pane. See Controlling Inductance Extraction for more information.

   To extract inductance, you must use Transistor Level or Gate Level as the extraction type. In addition, you must also extract resistance, or set No R/C and enable incremental extraction ("Generate PDB incrementally" on the **Outputs** > **SVDB** tab).

**Note**

The second button in the Extraction Type section controls the information that is extracted into the parasitic database (PDB). To control the information from the PDB that is displayed in the netlist, choose **Setup** > **PEX Options**, then the **Misc** tab and select a setting for "Parasitics to output to RC netlist." This setting allows you to set up your netlist to display a subset of what you have extracted to the PDB. For more information, refer to "Specifying Which Parasitics to Include in the Netlist" on page 338.

# Specifying Extracted Netlist Options for a Calibre Interactive PEX Run

The **Netlist** and **Nets** tabs on the Outputs pane specify the netlist format and which nets to extract parasitics for.

## Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.

## Procedure

1. Click **Outputs** on the left panel.

2. Click the **Netlist** tab to specify the extracted netlist format and options.

   The settings on the Netlist tab control the parameters for the PEX Netlist command.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

a. Choose the output format from the “Format” dropdown list; available options are CALIBREVIEW, DSPF, ELDO, HSPICE, SPECTRE, and SPEF. Some formats have options that can be selected in the bottom half of the format dropdown list.

   If a format selection appears in red text, then the selection is not allowed with one of the other output selections, such as the extraction type or the “Use Names From” setting.

   For the CALIBREVIEW format, you can select the iCellMap syntax on the PEX Options pane one the **Netlist** > **Advanced** sub-tab.

   Also see “Creating a SPECTRE Netlist from the Calibre View Netlist” on page 790.

b. Choose the source for the net and instance names using the “Use Names From” dropdown list.

   - **SCHEMATIC** — Use names from the source schematic, but use the layout hierarchy. You must specify an LVS report name on the **Reports** tab and specify a schematic netlist on the Inputs pane on the **Netlist** tab. You can specify an optional “Device Info File” defining device parameters. (SOURCENAMES parameter)
   - **LAYOUT** — Use names from the layout. (LAYOUTNAMES parameter)
   - **SOURCEBASED** — Use the circuit pin order and cell hierarchy in the source netlist. (SOURCEBASED parameter)
   - **SCHEMATICONLY** — The parasitic netlist is based on LVS source hierarchy, with intentional device parameters extracted and backannotated from the layout. See the PEX Netlist command for details. You can specify an optional “Device Info File” defining device parameters. (SCHEMATICONLY parameter)

   The “SOURCEBASED” and “SCHEMATICONLY” options are most appropriate for Calibre View creation and other backannotation flows, since the parasitic netlist is based on the source schematic.

c. Enter the PEX netlist filename.

   If Calibre Interactive is connected to a design tool, you can use a template to name the output parasitic netlist based on the extraction type, format, and format options. See “Naming the Extracted Parasitic Netlist Based on the Extraction Type and Format” on page 317.

3. Click the **Nets** tab to select the nets to extract parasitics for and specify the parasitic model for specific nets. When the extraction mode is xACT 3D/xRC Hybrid, these controls specify the nets extracted using Calibre xRC model-based extraction.

   a. Select the setting for “Extract parasitics for”:

      - **All nets** — Parasitic extraction is done for all nets.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Specified nets** — Specify nets to include or exclude from extraction. For more information, refer to "Including/Excluding Nets" on page 325.

b. **Select netlist model for specific nets** — Select this setting and enter the net model filename to specify a parasitic model for specific nets. When this option is selected, the SVRF statement PEX Netlist Select File is added to the control file, and the -netmodel option is added to the Calibre xRC command line.

The option "Select netlist model for specific nets" is only available if the extraction type specified is "Transistor Level" or "Gate Level" with R+C+CC, R+C+CC+L, or R+C+CC+L+M. Other restrictions may apply; see the description for the PEX Netlist Select File command.

4. If the extraction mode is "xACT 3D/xRC Hybrid," then choose **Setup > PEX Options** then the **xACT3D** tab to specify the subset of nets extracted using the xACT 3D fieldsolver.

   Choose a setting for "Extract parasitics using xACT 3D for":

   - **All nets** — xACT 3D extraction is done for all nets.
   - **Specified nets** — Specify nets to include or exclude from xACT 3D extraction.

     For included nets, you can also specify the accuracy, as follows:

     **200** — Less accurate, with faster runtime. This is the default.

     **600** — More accurate, with slower runtime.

5. (Optional) If you selected "Specified nets" in Step 3 and "Include", you can specify to keep coupling capacitance separate from intrinsic capacitance with the following steps:

   a. Click the **PEX Options** button or choose **Setup > PEX Options**.

   b. Click the **Netlist** tab and the **Reduction and CC** sub-tab.

   c. Enable "Keep coupling capacitors separate."

   The -cselect command line option is used instead of -select if "Keep coupling capacitors separate" is enabled. This option may only be used with extraction types of C+CC, R+C+CC, and R+C+CC+L.

# Specifying Report Options for a Calibre Interactive PEX Run

You select report options on the **Reports** tab on the Outputs pane.

## Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Procedure

1. Click **Outputs** on the left panel.
2. Click the **Reports** tab and choose report options and specify filenames as appropriate.

   **PEX Report** — The results of your PEX run. The PEX Report is not generated if the extraction type is "No R/C."

   If you chose SOURCEBASED or SCHEMATICONLY in the dropdown list of the **Netlist** tab, click the dropdown list on the right to use names from the source (SCHEMATIC) or the layout (LAYOUT) in the PEX Report.

   **LVS Report** — You must specify an LVS report name if you chose SCHEMATIC in the "Use Names From" dropdown list of the **Netlist** tab.

   Select from the following sub-tabs to generate the corresponding reports:

   **Point to Point** — See PEX Report Point2Point in the *SVRF Manual* for the meaning of the available options.

   > The SOURCENAMES or LAYOUTNAMES keyword in the Report Point2Point statement is determined by the selection "Use Names From" on the **Outputs** pane, **Netlist** tab.

   **Coupling Capacitance** — See PEX Report Coupling Capacitance in the *SVRF Manual* for the meaning of the available options.

   **Net Summary** — See PEX Report Netsummary in the *SVRF Manual* for the meaning of the available options.

   Also see "Reports" in the *Calibre xRC User's Manual* for information on the "Resistance Point-to-Point," "Net-to-Net Coupling Capacitance," and "Net Summary" reports.

# Specifying Database Output Options for a Calibre Interactive PEX Run

You specify that results database name and options on the **Outputs** tab of the Outputs pane.

### Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.

### Procedure

1. Click **Outputs** on the left panel.
2. Click the **SVDB** tab to specify the standard verification database and its options. These options control the parameters for the Mask SVDB Directory command.

   **SVDB directory** — Required. Specify the SVDB directory name.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Start RVE after PEX** — Optional. Select this to start Calibre RVE after the PEX run completes. If you are creating a runset for a batch run, you may want to enable "Start RVE after batch run" (**Setup** > **Preferences**, **Misc** tab).

**Generate cross-reference data for RVE** — Recommended and on by default. Generates the binary cross-reference database (XDB) for Calibre RVE and the Query Server.

**Generate ASCII cross-reference files** — Generate ASCII cross-reference (ixf, nxf, lph, sph) files in the SVDB database.

**Generate Calibre Connectivity Interface data** — Generate Calibre Connectivity Interface (CCI) data in the SVD database; requires a Calibre Connectivity Interface (CI) license.

**Suppress pin-location information** — Add the NOPINLOC keyword to the Mask SVDB Directory statement. This option is only available when "Generate Calibre Connectivity Interface data" is selected.

3. Select the PDB generation method.

   By default, the PDB is recreated every time the PDB generation step is run.

   Enable "Generate PDB incrementally" when performing *iterative extraction*. This is often done when extracting a netlist with mixed parasitic networks. See the sections "Extracting a Netlist with Mixed Parasitic Networks" and "Mixing Parasitics from Calibre Interactive" in the *Calibre xRC User's Manual*, and "Controlling PEX Step Execution" on page 321 in this manual.

# Naming the Extracted Parasitic Netlist Based on the Extraction Type and Format

You can use a file naming template to name the extracted parasitic netlist based on the cell, extraction type, netlist format, and format options. For example, if the netlist format is DSPF, the file extension can be set to .dspf.

File naming templates are used to define file names based on properties of the open design. File naming templates are applied when Calibre Interactive is invoked from a design tool and a runset is initially loaded. The file naming template is not re-evaluated when options in the Calibre Interactive GUI change. Replaceable parameters can be used in the template file names. See "Templates for File Naming" on page 148 for complete information, including a table of replaceable parameters (Table 3-17).

All replaceable parameters may be used in the template for the extracted parasitic netlist filename. The following replaceable parameters are replaced with the GUI settings for the



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

extraction type, netlist format, and format options, and may only be used in extracted parasitic netlist filename:

**Table 8-3. Replaceable Parameters for Output PEX Netlist**

| Parameter | Definition |
|---|---|
| %X | The extraction type:<br>• R+C+CC — rcc<br>• R+C — rc<br>• R — r<br>• C+CC — cc<br>• No R/C — norc |
| %x | The output netlist format:<br>• Calibre View — cv<br>• DSPF — dspf<br>• Eldo — spi<br>• HSPICE — sp<br>• Spectre — scs<br>• SPEF — spef |
| %o | The netlist format option:<br>• PRIMETIME — pt<br>• HSIM — hsim<br>• ADITRAIL — ar<br>• Net Geometry — ng |

## Prerequisites

- Calibre Interactive PEX is open.

## Procedure

1. Select **Setup** > **Preferences** to open the Setup Preferences dialog box.
2. Select the **Templates** tab.
3. Select the **Outputs** sub-tab.
4. Enter the desired filename template in the “PEX Netlist” field. The default is %l.pex.netlist, where %l is replaced the name of the open layout cell.

   For example, to include the extraction type (%X), format options (%o), and netlist format (%x) in the file name, enter the following for the “PEX Netlist”:

   %l.pex.%X_%o.%x

5. Click **OK** to exit the Setup Preferences dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

6. Select **File** > **Save Runset** to save the runset.

## Results

When you invoke Calibre Interactive from a design tool and load the runset, the extracted parasitic netlist (Outputs pane, Netlist tab) is named according to the template.

For example, with the template given in Step 4, a cell name of opamp, extraction type of R+C+CC, and an output netlist format of HSPICE, the extracted netlist is named *opamp.pex.rcc_.sp*. If the netlist format is DSPF with the PRIMETIME option, the filename is *opamp.pex.rcc_pt.dspf*.

The extracted netlist filename is not updated if the GUI options change. For example, if you load a runset then change the netlist format in the GUI, the filename does not change. The filename is generated from the template when the runset is loaded.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Interactive PEX Execution

You click Run PEX to start your parasitic extraction run in Calibre Interactive PEX. You can run each step in the parasitic extraction process separately.

**Running Calibre Interactive PEX** . . . . . 320
**Controlling PEX Step Execution** . . . . . 321

## Running Calibre Interactive PEX

Use the following procedure to start a parasitic extraction run from Calibre Interactive—PEX.

### Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.

### Procedure

1. Enter all necessary information in the Rules, Inputs, Outputs, and Run Control panes. Options may also be specified on the PEX Options pane, if desired. All left panel buttons are be green when the necessary information is entered.

2. Choose **Setup** > **PEX Options** and select each of the option tabs to set additional options. All left panel buttons will be green when the necessary information is entered.

3. If you want to run each step in the parasitic extraction process separately, see "Controlling PEX Step Execution" on page 321 for instructions.

   **Tip**

   If you are using a TVF rule file, the TVF file is compiled and converted to SVRF format at each step. To save the converted rule file and re-use it for each execution step, enable "Save TVF Rules to SVRF before running Calibre" (**Setup** > **Preferences**, **Misc** tab).

4. If desired, choose **Setup** > **Preferences** then the **Triggers** tab to specify triggers. See "Trigger Functions in Calibre Interactive" on page 657.

5. Click **Run PEX** on the left panel of Calibre Interactive.

   If the run uses a source netlist, Calibre Interactive first verifies that the source netlist is complete; the source netlist check is case insensitive. If an internal pre-execution trigger is defined, this is done after the trigger executes. You can turn off the source netlist check by the choosing **Setup** > **Preferences**, clicking the **Misc** tab, and disabling the setting "Check source netlist is complete before running PEX." See "Miscellaneous Preferences" on page 156 for information on the source netlist check.

   When the run starts, the view switches to the Transcript pane, and any run output, including errors and warnings, is displayed there.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

6. View the transcript pane and examine the results of the run. See "Transcript Pane in Calibre Interactive" on page 99.

**Tip**

You can generate and view a control file without running Calibre by holding down the Ctrl key and clicking **Run PEX**. In addition, the calibre command line for the run is written to the transcript pane.

### Results

If an error occurs in Calibre Interactive before the Calibre run is started, the error is displayed in a pop-up window or in the Calibre Quality Agent for Calibre Interactive.

When the Calibre run starts, the view switches to the Transcript pane, and any run output, including errors and warnings, is displayed here. See "Transcript Pane in Calibre Interactive".

If "Start RVE after PEX" is checked, the results database is opened in Calibre RVE when the run is complete; see "Using Calibre RVE for PEX" in the *Calibre RVE User's Manual*. In addition, the Calibre Interactive GUI may open a report file in a text viewer when the run completes; see "Viewing and Editing Text Files in Calibre Interactive" on page 125.

### Related Topics

Specifying a Rule File for PEX Run

Inputs for a PEX Run

Outputs for a PEX Run

Setting PEX Options

Run Control, Remote Execution and Licensing Options

Setting Database and General Options in Calibre Interactive

Setting LVS Options for PEX

## Controlling PEX Step Execution

By default, the GUI executes all three PEX steps (Persistent Hierarchical Database creation, Parasitic Database creation and formatted pex netlist creation) each time you click Run PEX. You can select each step individually, if desired.

### Prerequisites

- Calibre Interactive PEX is open; see "Invoking Calibre Interactive Tools" on page 64.

### Procedure

1. Click **Run Control** on the left panel.
2. Click the **Advanced** tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Choose the desired PEX steps to execute, as shown in Figure 8-4.

**Figure 8-4. Selecting PEX Steps for Execution**

4. (Optional) Set options to control PDB and formatter message output:
   a. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
   b. Click the **Misc** tab on the PEX Options pane.
   c. Enable “Display formatter warnings” to add the -fmt_warnings option to the command line for the formatting stage.
   d. Enable “Display formatter information” to add the -fmt_info option to the command line for the formatting stage.
   e. Enable “Display PDB THRESHOLDING messages” to add the -pdb_info option to the command line for the PDB stage.

# Run Control, Remote Execution and Licensing Options

You can control how your Calibre PEX job is run by configuring options related to performance, remote and distributed computing, and license acquisition.

These topics are covered in detail in “Run Control Setup and Licensing” on page 681. Also see “Controlling PEX Step Execution” on page 321.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

You can select the Calibre Cell/Block license for Calibre xRC and Calibre nmLVS from the **Environment > Licensing** sub-tab of the Run Control pane. The following restrictions apply when using the Calibre CB license:

- The Calibre xRC CB license option is appropriate only for flat PEX runs on designs containing up to 100,000 transistors.
- The Calibre xRC CB license can only be used for flat transistor or gate level extraction.
- The Calibre Cell/Block license options for Calibre nmLVS and Calibre xRC should be selected if SOURCE names are requested in a PEX run of Calibre Interactive and you only have Calibre Cell/Block licenses.
- The Calibre Cell/Block license is used for Calibre xACT 3D or xACT 3D/Hybrid runs. The option is ignored if it is selected.

The -flatten command line option is used when Calibre nmLVS is invoked during a Calibre Interactive PEX run in the following cases:

- “Run hierarchical version of Calibre-LVS” is not enabled on the **Performance** tab of the Run Control pane.
- “Use Calibre-CB license for LVS” is enabled on the **Environment > Licensing** tab of the Run Control pane.

## Related Topics

Extracting a Block Using CB [Calibre xRC User's Manual]

Setting Licensing Options



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setting PEX Options

You can customize the extracted parasitic netlist by selecting appropriate options in Calibre Interactive. The settings in Calibre Interactive override rule file settings and defaults for the parasitic extraction run. Most option settings are on the PEX Options pane, while some are on the Outputs pane.

Do one of the following to access the PEX Options pane:

- Choose **Setup** > **PEX Options**. This action adds the PEX Options button to the left side of the Calibre Interactive PEX window and opens the tab.
- Click the **PEX Options** button on the left panel if it is present.

The following topics cover options related to LVS, netlist extraction, database and general options in a PEX run:

- "Setting LVS and Netlist Extraction Options in a PEX Run" on page 348
- "Setting Database and General Options in Calibre Interactive" on page 93

The following options are specific to parasitic extraction:

**Including/Excluding Nets** . . . . . 325
**Specifying Netlist Formatting Options** . . . . . 326
**Specifying Netlist Reduction and Coupling Capacitance Options** . . . . . 330
**Specifying Advanced Netlist Options** . . . . . 333
**Specifying Via Reduction** . . . . . 335
**Specifying the Top Level Pin Order** . . . . . 337
**Specifying Which Parasitics to Include in the Netlist** . . . . . 338
**Specifying Parameters for Resistance Calculations** . . . . . 338
**Specifying Temperature Parameters for Resistance Extraction** . . . . . 339
**Specifying Extracted Netlist Annotation of Net Geometries to Calibre View** . . . . . 339
**Adding the PRIMETIME command to a PEX NETLIST Statement** . . . . . 342
**Setting PEX Power and Ground Nets** . . . . . 342
**Generating a Driver/Receiver File** . . . . . 342
**Specifying a SPICE Instance Prefix** . . . . . 343
**Controlling Inductance Extraction** . . . . . 343
**Setting Delay Calculation** . . . . . 345
**Controlling Message Output** . . . . . 346
**Changing Trace Property Rules for a PEX Run** . . . . . 347



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Including/Excluding Nets

By default, Calibre extracts parasitics from all nets in your layout. You can include or exclude certain nets from extraction.

## Procedure

1. Click Outputs on the left panel.
2. Click the Nets tab.
3. Select the nets for parasitic extraction:
   - **All Nets** — Specify all nets for parasitic extraction.
   - **Specified Nets** — Select nets to include and exclude.

     You can specify top level and recursive nets to include or exclude from extraction.

     You can specify nets manually by listing them in the text field, or click the select button to select nets in a connected schematic viewer.

     - **Manual selection** — 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.
     - **Select in schematic** — The browse buttons are active only if SCHEMATIC is selected in the "Use Names From" dropdown list.

       Note: For Cadence Composer, you must press Ctrl-C on the keyboard to end the net selection process and enter the selected nets. (For Calibre 2012.3 and earlier releases, use the Esc key to end the net selection process.)
4. If the extraction mode is "xACT 3D/xRC Hybrid," then choose **Setup > PEX Options** and the **xACT3D** tab to specify the subset of nets extracted using the xACT 3D fieldsolver.

   Specify a setting for "Extract parasitics using xACT 3D for":

   - **All nets** — xACT 3D extraction is done for all nets.
   - **Specified nets** — Specify nets to include or exclude from xACT 3D extraction. See the discussion with Step 3 for directions on selecting nets.

     For included nets, you can also specify the accuracy, as follows:

     **200** — Less accurate, with faster runtime. This is the default.

     **600** — More accurate, with slower runtime.
5. (Optional) If you selected "Specified nets" and "Include" in Step 3, you can specify to keep coupling capacitance separate from intrinsic capacitance with the following steps:

   a. Click the **PEX Options** button or choose **Setup > PEX Options**.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

b. Click the **Netlist** tab and the **Reduction and CC** sub-tab.

c. Enable “Keep coupling capacitors separate.”

The -cselect command line option is used instead of -select if “Keep coupling capacitors separate” is enabled. This option may only be used with extraction types of C+CC, R+C+CC, and R+C+CC+L.

6. SVRF commands: PEX Extract Include and PEX Extract Exclude.

**Tip**

You can include PEX Extract Include and PEX Extract Exclude statements in your rule file. When you click **Load** on the Rules pane, Calibre Interactive populates the Exclude and Include fields with the nets specified in the statements. Since GUI settings override the rule file settings, you must also select “Specified Nets” and the desired “Exclude” or “Include” options for the settings to take effect.

Note: There is a limit of 16382 characters per line in the rule file for these SVRF statements. There is no character limit for entries in the corresponding GUI fields.

### Related Topics

Specifying Extracted Netlist Options for a Calibre Interactive PEX Run

## Specifying Netlist Formatting Options

The netlist formatting options specify how the netlist is constructed.

### Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the **Netlist** > **Format** sub-tab and make selections as desired. The following table describes the available options.

**Table 8-4. PEX Netlist Formatting Options**

| Selection (SVRF Command) | Comment |
|---|---|
| Ground node name<br>(GROUND keyword in PEX Netlist …) | Specify the name of the ground node in the PEX Netlist command. All intrinsic capacitors are grounded to this node. The default node name is 0. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 8-4. PEX Netlist Formatting Options (cont.)**

| Selection (SVRF Command) | Comment |
|---|---|
| Ground layer name<br>(PEX Ground Layer) | Specifies layers with shapes for ground regions. Only used in C+CC and R+C+CC extraction.<br>Enable the "Stray" checkbox to add the STRAY keyword; this eliminates all netlist output of intrinsic capacitance to substrate for the specified ground region. |
| Hierarchy separator<br>(SEPARATOR keyword in PEX Netlist …) | Use this option to change the character used to denote hierarchical separation in the extracted parasitic netlist. |
| Output intentional device locations<br>(LOCATION keyword in PEX Netlist …) | Device locations appear in the extracted parasitic netlist as comment-coded parameters ($X=x_value $Y=y_value) |
| Output values for intentional R,C devices<br>(RCNAMED keyword in PEX Netlist) | R and C values of intentional devices appear in the netlist for HSPICE, DSPF, and Eldo netlists. |
| Output multiple net ports from multiple text ports<br>(MULTITXTPORT keyword in PEX Netlist) | Generates multiple ports on the net from multiple text ports with the same name in the layout. Only used with the DSPF netlist format. |
| Enclose string properties in quotes in the netlist<br>(STRINGQUOTES keyword in PEX Netlist) | Specifies whether or not to enclose string properties in quotes (" ") in the netlist". By default, string properties are enclosed in quotes. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 8-4. PEX Netlist Formatting Options (cont.)**

| Selection (SVRF Command) | Comment |
|---|---|
| Output R<br>(adds keywords to PEX Netlist) | Choose this option to include comments in the extracted parasitic netlist which specify additional parasitic resistor information. Select from:<br>**locations** — keyword RLOCATION<br>comment: $x=x_coord $y=y_coord $x2=x2_coord $y2=y2_coord<br>**widths** — keyword RWIDTH<br>comment: $w=width<br>**minimum** — keyword MINIMUM<br>Specifies to report the minimum width for non-rectangular shapes, otherwise the average width is reported. Must be specified with “widths”; “minimum” and “actual” may not be used at the same time.<br>**lengths** — keyword RLENGTH<br>comment: $l=length<br>**layers** — keyword RLAYER<br>comment: $layer=layer_name<br>**thicknesses** — keyword RTHICKNESS<br>**actual** — keyword RACTUAL. When selected, actual parasitic resistor widths are reported with the keyword RWIDTH; otherwise the drawn width is reported.<br>**viacount** — keyword RVIACOUNT<br>comment: $count=number<br>**layermap** — keyword LAYERMAP, only available with DSPF format. When specified with the “locations” option, the resistor locations are reported as a bounding box with labels $llx, $lly, $urx, and $ury.<br><br>If you specify resistor reduction, choosing “layers” prevents combining resistors on different layers.<br>Note: Parasitic resistor information is not specified if you choose both “No R/C” and “No Inductance” in the Extraction Type selections on the Outputs pane. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 8-4. PEX Netlist Formatting Options (cont.)**

| Selection (SVRF Command) | Comment |
|---|---|
| Output C<br>(adds keywords to PEX Netlist) | Choose this option to include comments in the extracted parasitic netlist which specify additional parasitic capacitor information. Select from:<br>**locations** — keyword CLOCATION |
| Extract floating nets<br>(PEX Extract Floating Nets) | Specify how floating nets such as metal fill are handled during extraction. Select from:<br>**ALL**<br>**REDUCED**<br>**GROUNDED** — (default) A text entry field for the scale factor is available; enter a value greater than 0, or use the default value of 1.<br>**Ignore port text** — Ignore port text placed on floating nets during extraction. |
| Size multiplier for length units (in meters)<br>(*scale* parameter for output netlist format in PEX Netlist …) | Value must be a positive floating-point number.<br>This specifies the size multiplier (in units of meters) for the L, W, A, P, AS, AD, PS, and PD parameters of MOSFET, diode, and JFET devices. Set the size multiplier to 1e-6 meters to see lengths parameters in microns and areas in square microns. Default is 1. |
| Formatter map file:<br>(PEX BA Mapfile) | Specify a backannotation mapfile that contains the device, pin name, and model mappings used to map the extracted intentional devices in a Calibre View, DSPF or HSPICE netlist to their corresponding simulation models.<br>**Suppress LPE values from CalibreView netlist** — As described; adds the CALIBREVIEW keyword. |
| Enable corner extraction<br>(calibre -xrc -fmt -corner)<br>(calibre -xact -fmt -corner) | Select to output corners in the netlist. Select from:<br>**All** — (default)<br>**Specify** — A text entry box appears; enter the specified corner names in a space or comma-separated list. |

3. (Optional) Set options to control formatter message output:
   a. Click the **Misc** tab on the PEX Options pane.
   b. Enable “Display formatter warnings” to add the -fmt_warnings option to the command line for the formatting stage.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

c. Enable “Display formatter information” to add the -fmt_info option to the command line for the formatting stage.

# Specifying Netlist Reduction and Coupling Capacitance Options

The netlist reduction and coupling capacitance options provide additional control over the extracted parasitic netlist.

## Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present
2. Click the **Netlist** > **Reduction and CC** sub-tab and make the desired selections. Table 8-5 described the available options.

**Table 8-5. Netlist Reduction and CC Options**

| Selection (SVRF Command) | Comment |
|---|---|
| Enable TICER reduction below<br><br>(PEX Reduce TICER) | Enter the TICER cutoff frequency in Hertz. Scientific notation may be used.<br><br>Performs reduction of distributed RC extraction data in a way that has little impact on circuit characteristics such as delay up to a specified frequency.<br><br>Specify the “Port Merge” and “Max Degree” values to add the PORTMERGE and MAXDEG keywords. See the command reference for a description of these parameters and the required units. |
| Enable CC reduction<br><br>(PEX Reduce CC) | **Absolute** — Enter threshold (> 0). The threshold is scaled by the Unit Capacitance value.<br><br>**Ratio** — Enter ratio value between 0 and 1.<br><br>**Scale** — Optional parameter, with a default of 1.<br><br>Controls the reduction of coupled capacitance between nets based on total capacitance. See the description in the *SVRF Manual* for details. |
| Enable analog reduction:<br><br>(PEX Reduce Analog) | **Delay** — Enter the timing delay threshold in seconds to be used for parasitic reduction.<br><br>**Noise** — Enter the error threshold used for parasitic reduction; the value is a ratio between 0 and 1.<br><br>Specifies performing reduction of extracted data from analog designs. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 8-5. Netlist Reduction and CC Options (cont.)**

| Selection (SVRF Command) | Comment |
|---|---|
| Enable MinCap reduction<br><br>(PEX Reduce Mincap) | Enter a non-negative threshold in units specified by Unit Capacitance. If Unit Capacitance in not specified, picofarads is used.<br><br>Select this to combine or remove parasitic capacitors below a threshold value to reduce netlist size. |
| Enable MinRes reduction<br><br>(PEX Reduce Minres) | Select this to combine and short parasitic resistors below a threshold value to reduce netlist size.<br><br>• Select YES in the dropdown box to enable GUI controls.<br>Enter a non-negative threshold in units specified by Unit Resistance. These two commands are added to your control file:<br>`PEX Reduce Minres Combine` *`combine_threshold`*<br>`PEX Reduce Minres Short` *`short_threshold`*<br>If you select this option and do not change the threshold value from the default of zero, then the command uses zero for the threshold.<br>Select the LAYERWISE checkbox to specify the reduction of parasitic resistors for each layer and not across different layers.<br>• Select NO to disable MinRes reduction and override rule file settings. |
| Enable MinMutual reduction below<br><br>(PEX Reduce Minmutual) | Select this to remove parasitic mutual inductance below a threshold value to reduce netlist size. Enter a filtering threshold for small inductance coupling coefficient (K) values. The threshold value must be greater than zero and less than one (0 < K < 1). |
| Enable Parallel MOS reduction<br><br>(PEX Reduce Parallel MOS) | Controls whether or not mosfets that are wired in parallel, and are the same model type and subtype, are reduced to a single mosfet in the generated netlist.<br><br>**Types** — Enter ALL (the default) or specify one or more parallel mosfet or user-defined mosfet device type names to combine.<br><br>**R** <*threshold*> — The resistance threshold between parallel device pins.<br><br>**SAMEWL** — Enable to merge only parallel devices of the same width and length. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 8-5. Netlist Reduction and CC Options (cont.)**

| Selection (SVRF Command) | Comment |
|---|---|
| Ground all coupling capacitors | Convert each parasitic coupling capacitor between two nets into a capacitance from each net to ground in the extracted parasitic netlist. The value of each grounded capacitor equals the value of the original coupling capacitor.<br>Uses the command:<br>`calibre -xrc -fmt -g` |
| Keep coupling capacitors separate | Keeps coupling capacitance separate from intrinsic capacitance. Uses the -cselect option when specifying nets to include in the extracted netlist. See "Including/Excluding Nets" on page 325. |
| Keep parasitic net models marked with _noxref from netlists<br><br>(PEX Netlist Noxref Net Names) | **Enabled** — Nets in the design with "noxref" in the name still appear in the intentional circuit and are treated as ideal nets.<br>**Disabled** — _noxref nets are suppressed.<br>Default: Enabled. |
| Use current-controlled current sources to model mutual resistance<br><br>(PEX Netlist Mutual Resistance) | As described.<br>Default: Enabled. |
| Set global net names<br><br>(PEX Netlist Global Nets) | Enable to remove the named nets from the subcircuit and place them in the global name definition. Enter the net name or names in the text field.<br>Global nets must be excluded from the netlist, since you cannot netlist parasitics on global nets in hierarchical extraction. See "Including/Excluding Nets" on page 325. |
| Specify device resistance model<br><br>(PEX Netlist Device Resistance Model) | Select to specify which model to use for parasitic resistance calculation. Select from:<br>• SPICE — Use the internal HSPICE model.<br>• CALCULATE — Use the width (W) and length (L) from the Device statement in the rule file.<br>Default: SPICE |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Specifying Advanced Netlist Options

The advanced controls for the extracted parasitic netlist include options specific to the output netlist format.

## Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the **Netlist** > **Advanced** sub-tab and made the desired selections. Table 8-6 describes the available options.

**Table 8-6. Advanced Netlist Format Options**

| Selection | Comment |
|---|---|
| **DSPF** | |
| Eliminate instance section | As described.<br>NOINSTANCESECTION keyword for PEX Netlist |
| Suppress top-level subcircuit | As described.<br>NOTOPSUBCKT keyword for PEX Netlist |
| **SPEF** | |
| Generate name map | Enable to generate a list that maps instance names to numerical identifiers.<br>MAPNAMES keyword for PEX Netlist |
| **DSPF/SPEF** | |
| Set bus delimiter | Enter a string to set the BUS_DELIMITER variable in the DSPF or SPEF header. The default is brackets ( [] ).<br>“BUSDELIM *string*” keyword and value for PEX Netlist |
| **HSPICE** | |
| Redefine pin to subcircuit delimiter | _ (entry) (enter a character, _ is the default)<br>PINNAMESEP keyword for PEX Netlist |
| Use short pin names | Enable to cause the formatter to replace pin names with an integer when creating SPICE net names.<br>SHORTPINNAMES keyword for PEX Netlist |
| **CALIBREVIEW** | |
| Use iCellMap | Enable to use the iCellMap syntax for the CalibreView netlist.<br>ICELLMAP keyword for PEX Netlist |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 8-6. Advanced Netlist Format Options (cont.)**

| Selection | Comment |
|---|---|
| Cellmap File | Enable the checkbox and enter a Calibre View cellmap file in order to use it in the Calibre View flow.<br>If Calibre Interactive was invoked from Cadence Virtuoso, the cellmap file is entered into the "Cellmap File" field in the "Calibre View Setup Dialog Box" (**Calibre** > **Setup** > **Calibre View**) during the PEX run. However, if the Calibre View Setup dialog box is open at runtime, the setting in the Calibre View Setup dialog box takes precedence and it is not updated with the Calibre Interactive setting. |
| **General** | |
| Remove X in the instance names | Enable to cause the formatter to filter X's in HSPICE, DSPF, CALIBREVIEW, SPEF, and Spectre output.<br>NOINSTANCEX keyword for PEX Netlist |
| Retain X in flattened instance names | Enable to cause the formatter to retain X's in DSPF and CALIBREVIEW output.<br>LEADINGX keyword for PEX Netlist |
| Remove devices not attached to selected nets | Enable to remove all intentional devices not attached to the extracted nets.<br>PRUNE keyword for PEX Netlist |
| Scale parasitic resistor properties | Enter a *value* for the scaling factor for parasitic resistor properties.<br>RSCALE *value* keyword and value for PEX Netlist |
| Output temperature coefficients | As described. This option is ignored for CalibreView and SPEF netlist formats.<br>TCOEFF keyword for PEX Netlist |
| Combine transistor-level SPICE data | Enable to combine the transistor-level SPICE data, the subcircuit definitions of the parasitic net models for each extracted net, the parasitic resistors, the parasitic capacitors, and the instances of the net models into a single parasitic netlist output file. This option is only valid for HSPICE, ELDO, and Spectre netlist formats.<br>SINGLEFILE keyword for PEX Netlist |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 8-6. Advanced Netlist Format Options (cont.)**

| Selection | Comment |
|---|---|
| Create new instance names for smashed devices | **Enabled** — Create new instance names for smashed devices.<br>**Disabled** — Leave smashed device instance names unchanged.<br>Default: Enabled.<br>PEX Netlist Create Smashed Device Names |
| Upper case element names, HSPICE and DSPF keywords | As described.<br>Default: Disabled.<br>PEX Netlist Uppercase |
| Connect disjoint net model fragments | **Enabled** — Connect disjoint net model fragments created by Virtual Connect statements to the main net model using a resistor. This option is typically used for debug purposes.<br>**Disabled** — Leave net model fragments unconnected.<br>PEX Netlist Virtual Connect |
| Report contact/via area | The contact/via area that is used for extracting parasitic resistors is reported as a comment. This option applies to DSPF, SPEF, and HSPICE netlists.<br>RAREA keyword for PEX Netlist |

# Specifying Via Reduction

You can use the **Netlist** > **Vias** tab on the PEX Options pane to override the PEX Reduce Via Resistance statements in your rule file or add new statements.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 8-5. Netlist > Vias Tab in Calibre Interactive PEX**

## Prerequisites

- Calibre Interactive PEX is open.
- A rule file is specified on the Rules pane.

## Procedure

1. (Optional) On the Rules pane, click **Load** to load the rule file settings into the GUI.

   This loads the PEX Reduce Via Resistance statements in your rule file into the GUI, along with other rule file settings. This step is necessary if you want to change the rule file settings for via reduction.

2. Click the **PEX Options** button on the left panel to display the PEX Options pane. If the button is not available, choose **Setup** > **PEX Options**.

3. Click the **Netlist** tab.

4. Click the **Vias** sub-tab.

   If you loaded the rule file in Step 1, the via reduction statements in the rule file are loaded in table.

5. Select one of the following options:

   - **Use settings from rules** — Use PEX Reduce Via Resistance statements from the rule file. (default)
   - **Turn off via reduction** — Turn off via reduction; this adds the PEX Reduce Via Resistance OFF statement to the Calibre Interactive control file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Specify via reduction** — Use via reduction statements specified in the table. You can do the following:

| Action | Step |
|---|---|
| Disable or enable specific statements. | Use the checkboxes in the Include column to select the statements that are included in the run. |
| Edit the parameters for a statement. | Click in the Value column and edit the parameters. |
| Add a new statement. | Click to add a new statement, then fill in the parameters in the Value column. |

6. (Optional) Save the runset. Any via reduction statements in the table are saved to the runset and loaded into the table when you load the runset.

### Results

Statements are added to the Calibre Interactive control file depending on the settings in the GUI.

**Note**

Rule file compilation by Calibre Interactive may cause the order of [layer1 layer2] entries in the GUI to be different than that in the rule file.

## Specifying the Top Level Pin Order

You can specify the pin ordering for the netlist.

This functionality corresponds to the PEX Pin Order statement in the *SVRF Manual*.

### Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the **Misc** tab.
3. Enable “Create top level pin order.”
4. Choose LAYOUT, SOURCE, or FILE from the dropdown list to specify the pin ordering. The default is LAYOUT.

   Enter the file name if you selected FILE.
5. (Optional) If you specified SOURCE or FILE, you can enable “All Pins” to output all pins, including floating pins.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Specifying Which Parasitics to Include in the Netlist

By default, Calibre xRC includes both extracted resistors and capacitors in the extracted parasitic netlist for RC and RCC runs. You can limit the extracted parasitic netlist to just extracted capacitors or just extracted resistors.

## Procedure

1. Click the PEX Options button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the Misc tab.
3. Choose the desired option for the “Parasitics to output to RC netlist” setting.

**Note**

The “Parasitics to output to RC netlist” setting controls the information from the parasitic database (PDB) that the netlist displays. In other words, you can set up your netlist to display a subset of the PDB. To control which parasitics are extracted to the PDB, click **Outputs**, and select the second button in the “Extraction Type” section.

# Specifying Parameters for Resistance Calculations

You can specify layer-specific parameters for resistance calculations in Calibre Interactive PEX.

This corresponds to the SVRF statement PEX Resistance Parameters.

## Procedure

1. (Optional) On the Rules pane, click **Load**, and the parameters from PEX Resistance Parameters statements in the rule file are loaded into the dropdown list of resistance parameters.
2. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
3. Click the Misc tab.
4. Enable the selection “Resistance parameters.”
5. Enter the desired parameters for the PEX Resistance Parameters command in the text entry field. For example, to specify the TC1 and TC2 coefficients for the layer metal1, enter:

```
metal1 TC1 0.003 TC2 0.012
```

6. Click **Add**. This populates the dropdown list with the parameters you entered.
7. (Optional) Add additional commands by repeating Steps 5 and 6.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The text entry field lists the first parameter set from the dropdown list — simply overwrite the old text to enter a new parameter set. New parameter sets are added to the dropdown list if they are not duplicates of an existing entry, and previous entries are preserved.

8. (Optional) To delete a command, select the command from the dropdown list and click **Delete**.

### Results

If “Resistance parameters” is enabled, then the specified commands are used and PEX Resistance Parameters commands in the rule file are ignored.

## Specifying Temperature Parameters for Resistance Extraction

You can specify non-default temperature parameters for resistance calculations.

This setting corresponds to the PEX Extract Temperature SVRF statement.

### Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup > PEX Options** if the button is not present.
2. Click the Misc tab.
3. Enable “Extract Temperature.”
4. Specify the circuit temperature in degrees Celsius in the first entry field. The default is 27 C.
5. (Optional). Specify the nominal circuit temperature in degrees Celsius.The default is 27 C.
6. (Optional). Specify three temperatures separated by spaces in the “Range” field. These parameters are used in calculating TC1 and TC2. See PEX Extract Temperature for a discussion of how these parameters are used.

## Specifying Extracted Netlist Annotation of Net Geometries to Calibre View

You can choose to simplify the backannotation of net geometries into Calibre View by selecting the “Net Geometry” option for Calibre View and the “Show Parasitic Polygons” option in the Calibre View setup.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Backannotation of net geometries allows you to visualize the parasitics and their corresponding layout shapes, and also helps in identifying plot points for simulation from within the Cadence Analog Design Environment (ADE).

## Prerequisites

- A cellmap file with layer mapping statements. See the following sections:
  - "Creating the Cellmap File" on page 754
  - "Layer Mapping Syntax" on page 762
  - "Identifying Cadence Virtuoso Layers for Mapping" on page 771
- The following licenses:
  - A Calibre Connectivity Interface (CCI)
  - Calibre Interactive
  - Calibre RVE
  - Calibre nmLVS-H

## Procedure

1. Click the Outputs button on the left panel.
2. Click the Netlist tab.
   a. Click the "Format" dropdown button list and enable the "CALIBREVIEW" option.
   b. Click the "Format" dropdown button list again and enable the "Net Geometry" option. The "Format" button should now display "CALIBREVIEW+Net Geometry"
   c. Select "SCHEMATIC" for "Use Names From." Backannotation with the net geometries requires using names from the schematic netlist.

   The **Netlist** tab should look similar to the following image:

3. Click the **SVDB** tab.
4. Enable the "Generate Calibre Connectivity Interface data" option.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. Verify that the **PEX Options** button is displayed in the left pane; if it is not, choose the menu item **Setup** > **PEX Options** to display it.
6. Click the **Netlist** tab, then the **Format** sub-tab.
7. Enable the four Output parasitic resistor options: “locations,” “widths,” “lengths,” and “layers.”
8. In Cadence Virtuoso, choose **Calibre** > **Setup** > **Calibre View** to open the Calibre View dialog box.
   a. Enable “Always Show Dialog” at the bottom of the form.
   b. Click **Save** to close the dialog box. Do not click **OK**, as this creates the Calibre View.
9. Click **Run PEX** on the left panel of Calibre Interactive.
10. When the Calibre View Setup dialog box opens, make the following selections:.
    a. Set the “View” field to “masklayout.”
    b. Set the “Parasitic Placement” field to “Layout Location.”

       The “Show Parasitic Polygons” option should become active.
    c. Enable the “Show Parasitic Polygons” option.
    d. Click **Save** to close the dialog box. Do not click **OK**, as this starts the creation of the Calibre View.

## Results

Calibre places the intended devices and parasitics in the Cadence Virtuoso layout and annotates the original resistance polygons. Each polygon is split into two parts; one polygon connects to the resistor's PLUS pin, the other polygon connects to the resistor's MINUS pin, and the resistor is located in the middle of the two polygons. The polygons' layer is the same as the original resistance layer.

**Note**

The message “Calibre Error: Cannot create net objects in views of type schematic” is issued if creating a schematic view with Net Geometry enabled. This message is expected and the Calibre View is still created, although net objects are not created in the schematic view.

## Related Topics

Creating a Calibre View

Creating a Post-Parasitic Extraction Calibre View



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Adding the PRIMETIME command to a PEX NETLIST Statement

You can add the PRIMETIME keyword to your Pex Netlist statement if you are using either the SPEF or DSPF netlist formats in RC mode. The optional keyword PRIMETIME creates a more compact netlist that the PrimeTime® static timing analysis tool can parse.

## Procedure

1. Click Outputs button on the left panel.
2. Click the Netlist tab.
3. Choose SPEF or DSPF for the netlist format in the “Format” dropdown button list.
4. The PRIMETIME option is available at the bottom of the “Format” dropdown button list. Click the checkbox to enable it.

   Note: If the combination of netlist format and extraction type is not allowed the format indicator is printed in red.
5. See PEX Netlist in the *SVRF Manual* for complete details.

# Setting PEX Power and Ground Nets

You can configure the PEX Power and PEX Ground statements for specific nets using Calibre Interactive PEX.

Corresponding commands: PEX Power and PEX Ground. These settings are only used for Calibre xL.

## Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the **Misc** tab.
3. In the fields for “Power nets” and “Ground nets,” select “LAYOUT” or “SOURCE” as the source for the net names, then enter one or more net names; separate multiple net names with a space.

## Related Topics

Setting LVS Options for PEX

# Generating a Driver/Receiver File

You can use Calibre PEX to create a driver/receiver file for use during parasitic extraction.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the Misc tab.
3. Enable the “Generate driver/receiver file” checkbox and enter the name of the file and appropriate tags in the “Driver/Receiver File” and “Driver/Receiver Tags” fields.

   See PEX Generate Driver File Tag in the *SVRF Manual* for complete details.

## Specifying a SPICE Instance Prefix

You can specify the instance prefix used by your netlist file.

### Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the Misc tab.
3. Enter the appropriate prefix in the “Spice instance prefix” field.

## Controlling Inductance Extraction

You can control inductance extraction using Calibre Interactive.

### Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the **Inductance** tab.
3. Specify the setting for “Extract Mode,” corresponding to the PEX Inductance Mode statement.

   **LOOP** — Loop inductance extraction mode. This is the default.

   **PEEC** — Partial RLK extraction mode, which is based on the Partial Element Equivalent Circuit method. If selected, the GUI automatically sets “Extract parasitics for” to “Specified Nets” on the **Nets** tab of the Outputs pane, and enables “Include” on the Top-Level Nets sub-tab.

   Clear the checkbox for “Extract Mode” to use the rule file setting.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

4. Set the following options, depending on whether you want to use the broadband model:

| To do this … | Do the following … |
|---|---|
| Generate a broadband netlist | 1. Enable the "Broadband" checkbox.<br>2. Select one of the following options:<br>• **Operating Frequency** — Specify the maximum operating frequency. This option uses the PEX Inductance Frequency statement with the MAXIMUM keyword.<br>• **Switch Time** — Specify the switch time. This option uses the PEX Inductance Switch Time statement. |
| Do not use the broadband model in netlist generation | 1. Disable the "Broadband" checkbox.<br>2. Specify the operating frequency using the "Operating Frequency" option. This option uses the PEX Inductance Frequency statement. |

5. Use the "Net Filter" dropdown list to specify a value for the PEX Inductance Filter statement.

   If "File" is chosen from this list, two additional fields are enabled:

   **Driver/Receiver File** — Enter or browse for the file used by the PEX Inductance Driver File statement.

   **Driver/Receiver Tags** — Enter one of more optional tags; these are appended to the PEX Inductance Filter statement.

6. If desired, enable the "Use Victim Path File" option and enter the path to the victim file. See the PEX Inductance Victim SVRF statement for information about the victim file format.

7. Select the "Inductance Range" option to set the maximum distance used to determine inductance. "Cell Extent" is the default and results in the entire cell being used. You can also specify a maximum distance in microns. (SVRF statement: PEX Inductance Range)

8. Use the dropdown list next to the "Nets to include in return path" checkbox to specify return path nets. Select either Ground, Power, Ground and Power, Average or None. Selecting Average uses the average results of ground only and power only. Selecting None writes the PEX Inductance Returnpath NONE statement to the control file. Ground is the default.

   Clear the "Nets to include in return path" checkbox to omit the PEX Inductance Returnpath statement from the control file.

   This option is not available if "Extract Mode PEEC" is selected.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

9. If desired, enable “Same net mutual inductance” to enable the extraction of mutual inductance between wire segments within the same net. The default is off. (SVRF statement: PEX Inductance Same Net Mutual)
10. If needed, specify ground and power nets on the **Misc** tab of the PEX Options pane.
    a. Click the **Misc** tab on the PEX Options pane.
    b. In the fields for “Power nets” and “Ground nets,” select LAYOUT or SOURCE as the source for the net names, then enter one or more net names. Separate multiple net names with a space.

### Related Topics

Calibre xL Users Manual

Selecting Nets for Mutual Inductance [Calibre RVE User's Manual]

## Setting Delay Calculation

You can configure the delay calculation tool that runs on blocks marked for extraction.

Refer to Manipulating Cell Extraction Options Using the Blocks Tab for information on choosing blocks for extraction.

### Procedure

1. Click **Setup** > **Delay Calculation**.
2. Choose one or more technology library files to be used by the delay calculator for obtaining timing information for primitive cells.
3. Choose “Additional Verilog Files” in addition to those specified on the Inputs pane in the **Netlist** tab as necessary. These additional files are used to define the modules for delay calculation.
4. Specify a file containing SDC (Synopsys Design Constraint) commands as necessary.
5. Select either the TIME-IT or OTHER delay calculation tool. If OTHER is chosen, see Step 8.
6. Choose either the SPEF or DSPF parasitic extraction file format. Before the delay calculator is run to create a Standard Delay Calculator file (SDF) for the module, the SDF extraction is first used by Calibre-xRC in the chosen format. SPEF is strongly suggested if Time-It is chosen as the delay calculation tool.
7. Specify the MGC_HOME value for the Time-It delay calculation tool. This is the path for MGC_HOME or CALIBRE_HOME for your Calibre software tree.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

8. If OTHER is chosen as the delay calculator tool, the command line for that tool must be specified in the “Command to Invoke Delay Calculator.” The following parameters are recognized and substituted with the appropriate runtime values:

| Parameter | Substitution |
|---|---|
| %t | Technology files |
| %v | Verilog files (Include both files you would specify in the Netlist pane and Additional Verilog Files field) |
| %p | Top module in Verilog |
| %i | Input parasitic file |
| %f | Parasitic file format |
| %o | Output SDF file |
| %z | SDC constraints file |
| %m | Cell name |

9. If the ADMS flow is enabled, the command specified in the “Command to Invoke Delay Calculator” field is run upon completion of the formatting step of Calibre-xRC. This command is used to create SDF files for blocks specified to be extracted in SDF format.

# Controlling Message Output

You can control the level of output to the transcript for the PDB and formatter execution stages.

## Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the **Misc** tab on the PEX Options pane.
3. Enable “Display formatter warnings” to add the -fmt_warnings option to the command line for the formatting stage.
4. Enable “Display formatter information” to add the -fmt_info option to the command line for the formatting stage.
5. Enable “Display PDB THRESHOLDING messages” to add the -pdb_info option to the command line for the PDB stage.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Changing Trace Property Rules for a PEX Run

You can use the **Properties** tab on the PEX Options pane in Calibre Interactive to override the Trace Property statements in your rule file. You can add new statements, edit existing statements, and disable statements.

See "Changing Trace Property Rules" on page 274 for instructions on using the **Properties** tab on the PEX Options pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setting LVS and Netlist Extraction Options in a PEX Run

You can set options for gate recognition, gate reduction, and netlist extraction in a Calibre Interactive PEX run.

**Setting LVS Options for PEX** . . . . . 348
**Logic Gate Recognition Options in PEX** . . . . . 349
**Split Gate and Parallel MOS Reduction Options in PEX** . . . . . 349
**Filter Unused Device Options in PEX** . . . . . 349

## Setting LVS Options for PEX

You can specify LVS power and ground names and LVS report options on the **LVS Options** tab of the PEX Options pane.

### Procedure

1. Click the **PEX Options** button on the left panel, or choose **Setup** > **PEX Options** if the button is not present.
2. Click the **LVS Options** tab.
3. Enter the power and ground net names in the appropriate fields; use a space separated list to enter multiple nets.

   To enter power and ground net names using a file, see "Setting Power Supply Options" on page 267 for the file format.

   Corresponding rule file statements: LVS Ground Name and LVS Power Name.
4. Select the LVS Report Options using the dropdown list.

   For most runs, you should not specify any of these options, unless you actually need the information they provide. The default report options are usually all that is needed.

   a. Click the button to the right of "LVS Report Options".

   b. Select the desired report options. The button label changes to show the selected options.

      You can also disable report options that are specified in the rule file.

   Corresponding rule file statement: LVS Report Option.
5. Set other options as described in these sections:
   - "Logic Gate Recognition Options in PEX" on page 349
   - "Split Gate and Parallel MOS Reduction Options in PEX" on page 349



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- "Filter Unused Device Options in PEX" on page 349

# Logic Gate Recognition Options in PEX

You can control how Calibre performs logic gate recognition.

Choose **Setup** > **PEX Options**, click the **LVS Options** tab, then select the options in the "Gate Recognition" area. These controls are the same as for Calibre Interactive nmLVS except for the tab label.

### Related Topics

Controlling Logic Gate Recognition

# Split Gate and Parallel MOS Reduction Options in PEX

You can control how split gates are reduced using Calibre Interactive PEX.

Choose **Setup** > **PEX Options**, click the **LVS Options** tab, then select the options in the "Split Gate Reduction" area. These controls are the same as for Calibre Interactive nmLVS except for the tab label.

### Related Topics

Setting Split Gate and Parallel MOS Reduction Options

# Filter Unused Device Options in PEX

You can filter out specific unused devices in your layout and source using Calibre Interactive PEX.

Choose **Setup** > **PEX Options**, click the **LVS Options** tab, then select options in the "Filter Unused Device Options" table. The procedure is the same as that for Calibre Interactive nmLVS except for the tab label.

### Related Topics

Setting Filter Unused Device Options



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 9
# Using Calibre Interactive xACT

You can use Calibre Interactive to run the Calibre® xACT™ direct netlisting flow. The GUI includes the following features:

- A search option to easily locate controls
- The ability to set environment variables within the GUI
- GUI customization:
  - Design new panes, move the location of GUI controls, hide controls, and specify initial settings.
  - Attach a callback function to a signal from a GUI control. For example, a function to set netlist options in the GUI can be executed if the netlist format changes.
  - Specify #DEFINE, #UNDEFINE, and Variable statements in the Calibre Interactive control file.
- Support for the direct netlisting flow in Calibre xACT and Calibre xACT 3D.
- Integration with Calibre® DESIGNrev™, Cadence® Virtuoso®, Cadence® Encounter®, and Cadence® Innovus®.


**Requirements for Calibre Interactive xACT** . . . . . 352
**Invoking the Calibre Interactive xACT GUI** . . . . . 352
**Calibre Interactive xACT Command Line** . . . . . 355
**Setting Up and Starting a Calibre Interactive xACT Run**. . . . . 357
**Tips for Using the Calibre Interactive xACT GUI** . . . . . 359
**Populating the Load Runset Files Dialog Box and Recent Files Lists** . . . . . 360
**Using a Calibre Interactive PEX Runset with Calibre Interactive xACT** . . . . . 363
**Loading a Classic Tcl Customization File in Calibre Interactive xACT** . . . . . 365
**Providing OpenAccess Input with Calibre Interactive xACT**. . . . . 366
**Specifying Probe Points in Calibre Interactive xACT** . . . . . 368
**Setting Options in Calibre Interactive xACT** . . . . . 370
**Calibre Interactive xACT GUI Reference**. . . . . 375
**GUI Customization for Calibre Interactive xACT**. . . . . 406





Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Requirements for Calibre Interactive xACT

Several requirements must be met to use Calibre Interactive xACT.

- Using Calibre on a platform running the AOI executable.

  The AOI executable supports platforms running Red Hat Enterprise Linux 6 (RHEL 6), RHEL 7, SLES 11 Service Pack 2, or SLES 11 Service Pack 3. See the *Calibre Administrator's Guide* for more information.

- The Calibre Interactive xACT GUI uses the XKEYBOARD extension. If you notice that keyboard input to the GUI fields is not correct, make sure your interface to the Linux machine running Calibre uses the XKEYBOARD protocol extension.

- "Calibre Interactive System Requirements" on page 26

# Invoking the Calibre Interactive xACT GUI

You invoke Calibre Interactive xACT from a design tool or from the command line. Calibre DESIGNrev and Cadence Virtuoso are supported.

## Prerequisites

- "Requirements for Calibre Interactive xACT" on page 352

- (Optional) You can specify one or more runsets to load automatically by defining the environment variable MGC_CALIBRE_XACT_RUNSET_FILE. For example:

  ```
  setenv MGC_CALIBRE_XACT_RUNSET_FILE "runset1_path runset2_path"
  ```

  The specified runset(s) are loaded when Calibre Interactive xACT is invoked. The -runset command line option has precedence over the environment variable.

- (Optional) You can define the environment variable MGC_CALIBRE_XACT_CONFIG_FILE to specify one or more configuration files. The -config command line argument has precedence over the environment variable.

- (Optional) You can specify the environment variable MGC_CALIBRE_XACT_RUNSET_LIST to pre-populate the Recent Runsets and Recent Text Files lists in the **File** menu. When this environment variable is defined, the runsets specified by MGC_CALIBRE_XACT_RUNSET_FILE are selected. See "Populating the Load Runset Files Dialog Box and Recent Files Lists" on page 360.

## Procedure

1. Use one of the following methods to start the GUI:

| Invoke from … | Method |
|---|---|
| Calibre DESIGNrev | **Verification > Run xACT** |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Invoke from … | Method |
|---|---|
| Cadence tools | **Calibre > Run xACT** |
| Command line | calibre -gui -xact [-runset *runset*]… [-config *file*]… [-batch] |
| Calibre Interactive palette | calibre -gui |

The -runset and -batch command line options are the same as for other Calibre Interactive applications.

The -config *file* switch and argument option loads a custom configuration file; see "GUI Customization for Calibre Interactive xACT" on page 406. You can specify multiple configuration files.

**Note**

For consistency with other Calibre Interactive applications, the -runset switch is optional, but it is recommended in all cases. In particular, the -runset switch should be used when using "-config *config_file*" to specify a custom configuration file, to avoid parsing errors.

It is also possible to specify multiple runsets as a space-separated list of runsets. However, best practice is to specify the -runset switch with each runset.

2. Proceed to "Setting Up and Starting a Calibre Interactive xACT Run" on page 357.

## Results

If you invoke the GUI from a layout tool, the open design is automatically used as the layout input and the extracted netlist is named *<cell>.pex.netlist*, where *<cell>* is replaced with the name of the open design cell.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The initial view of the GUI is similar to that shown in the following figure. A ! indicates an incomplete or incorrect field. Some panes, such as the Inductance and FieldSolver panes, are only displayed in the left panel if the appropriate settings are enabled.

**Note**

The runset option naming convention for some options changed in the 2018.4 release. As of 2018.4, certain options are prefixed with “cmn” instead of “pex”. Runsets saved with Calibre 2018.4 or later may report errors if they are loaded with an earlier version of Calibre.

## Related Topics

Calibre Interactive xACT Command Line

Calibre Interactive xACT GUI Reference



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Interactive xACT Command Line

The Calibre Interactive xACT command line has options to run in interactive mode or batch mode. You can also specify one or more runsets to open.

## Usage

### Interactive operation:

**calibre -gui -xact** [-runset *runset*]… [-config *config_file*]… [-custom *custom_file*]

### Batch operation:

**calibre -gui -xact** { **-runset *runset*** | **-runset_options_display *runset***}…
[-config *config_file*]… [-customgui *custom_file*] **-batch**

## Arguments

- **-gui -xact**

  Specifies Calibre Interactive xACT.

- -runset *runset*

  An argument set specifying the pathname of a runset. When specified, the runset is automatically loaded when the GUI opens. A runset is required for batch operation (the -batch switch); it is optional for interactive operation.

  You can load multiple runsets by specifying the switch and argument multiple times; the runsets are loaded in the order specified and the settings loaded last take precedence.

  **Note**

  For consistency with other Calibre Interactive applications, the -runset switch is optional, but it is recommended in all cases. In particular, the -runset switch should be used when using "-config *config_file*" to specify a custom configuration file, to avoid parsing errors.

  It is also possible to specify multiple runsets as a space-separated list of runsets. However, best practice is to specify the -runset switch with each runset.

- -runset_options_display *runset*

  An argument set that specifies the pathname of a runset and also instruct the tool to display the runset options in the transcript. This option is only valid for batch operation (the -batch switch).

  **Note**

  Calibre Interactive xACT can read in classic Calibre Interactive PEX runsets or runsets saved by the Calibre Interactive GUI. When -runset_options_display is used, all the runsets specified on the command line must be of the same type—either all PEX runsets or all xACT runsets.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

You can combine the -runset_options_display and -runset switches. Runsets are loaded in the order given in the command line; options in runsets occurring later in the command line override options in runsets occurring earlier on the command line. Only options in runsets specified by -runset_options_display are echoed to the transcript. If an option in a -runset_options_display runset is overridden by a later runset specified with the -runset option, then the option is not echoed to the transcript.

- -config *config_file*

  An optional argument set that specifies a configuration file to load. See "GUI Customization for Calibre Interactive xACT" on page 406.

- [-custom *custom_file*]

  An optional argument set specifying the pathname of a classic Calibre Interactive Tcl customization file. The controls in the Tcl customization file are converted to equivalent configuration controls and placed on a page named Custom.

  This switch overrides the MGC_CALIBRE_CUSTOMIZATION_FILE environment variable and customization files set in the runset, if one is loaded.

  See "Loading a Classic Tcl Customization File in Calibre Interactive xACT" on page 365 for more information.

- [-customgui *custom_file*]

  An optional argument set specifying the pathname of a classic Calibre Interactive Tcl customization file. This option is only available in batch mode.

  The controls in the Tcl customization file are converted to equivalent configuration controls and placed on a page named Custom. When a batch run is started, a GUI with only the Custom page and the Run button is displayed. You can make changes on the Custom page. Click the Run button to proceed with the batch run using the settings on the Custom page.

  This switch overrides the MGC_CALIBRE_CUSTOMIZATION_FILE environment variable.

- **-batch**

  A required argument in batch invocation. Calibre Interactive runs in batch mode, without opening the GUI.

## Examples

Invoke with a runset:

```
calibre -gui -xact -runset runset_xact
```

Invoke with two runsets and a custom configuration file:

```
calibre -gui -xact -runset rs_xact -runset rs_xact_opt -config xact.cfg
```

Invoke in batch mode:

```
calibre -gui -xact -runset runset_xact -config xact.cfg -batch
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setting Up and Starting a Calibre Interactive xACT Run

To set up the GUI for a run, click each of the page names on the left panel of the GUI and fill in the required fields.

Errors and warnings in the GUI settings are indicated as follows; hover over the icon to view a message about the problem.

| | |
|---|---|
| ! | A missing or incorrect setting. A red exclamation point ( **!** ) is also displayed by the page name if any settings have an error. |
| ⚠ | A warning, such as for a conflicting setting. The run can proceed with warnings. The warning icon is not displayed next to the page name. |

## Prerequisites

- The Calibre Interactive xACT GUI is open. See "Invoking the Calibre Interactive xACT GUI" on page 352.

## Procedure

1. (Optional) Select **File** > **Load Runset** to load a saved runset.

   To load multiple runsets, click the + button and select another file.

   Click a checkbox to select the runset or a group of runsets. Check multiple individual runsets to load them as a group. When multiple runsets are loaded, the runsets are loaded in order, with options loaded last having precedence if there is a conflict.

**Note**

The runset option naming convention for some options changed in the 2018.4 release. As of 2018.4, certain options are prefixed with "cmn" instead of "pex". Runsets saved with Calibre 2018.4 or later may report errors if they are loaded with an earlier version of Calibre.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Click **Rules** on the left panel and specify the rule file and run directory.

   If desired, click the button to load the settings from the rule file into the GUI.

3. Click **Inputs** on the left panel and specify the required input.

   (Optional) Check “Export from layout viewer” and/or “Export from source viewer” in the Layout Path and Source Path areas to export a design from a connected viewer—the active cell in the design tool is used as the layout or source input.

   Note that other settings in the GUI can affect the controls displayed on the Inputs page. For example, setting the PEX Netlist option “Use Names From” to “Source Names” on the Outputs page causes the Source Path control to be displayed on the Inputs page.

   **Tip**

   To specify additional layout or schematic input files, click **Database** on the left panel and expand the Library area.

   If source input is used and the format is VERILOG or MIXED, provide Verilog translation options on the V2LVS page.

4. (Optional) If you are exporting a design from a connected design tool, click **Run Control** on the left panel and view the “Design Tool Settings.”

   If you invoked the GUI from a design tool, the design tool connection is made automatically. If you need to make a manual connection to a design tool, see “Connecting to a Design Tool in Calibre Interactive xACT” on page 370.

5. Click each of the other page names on the left panel and set options as needed.

   **Tip**

   To quickly find a specific option, click **Search** in the left panel and enter a search string. Changes made in the Search page are reflected when you view the control in its original page.

6. If any GUI pages have a ! by their name, provide the required input.

7. (Optional) Choose **File > Save Runset** to save the GUI settings.

8. Click the **Run xACT** button to start the run.

## Results

The run transcript is displayed in the Transcript page, with any errors and warnings listed in the bottom portion of the page. Use the search toolbar in the upper right to search the Transcript page.

## Related Topics

Calibre Interactive xACT GUI Reference



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Tips for Using the Calibre Interactive xACT GUI

A search page, rule file loading, and other features make the Calibre Interactive xACT GUI easy to use.

**Note**

If a page is not visible on the left panel of the GUI, choose **Settings** > **Show Pages** and select the desired page.

**Table 9-1. Tips for the Calibre Interactive xACT GUI**

| To do this … | Do the following: |
|---|---|
| Search the GUI | Enter a search term in the entry field at the top right of the GUI. Use Ctrl-f or Ctrl-F to change focus to the search entry box.<br>See "Search Toolbar in Calibre Interactive xACT" on page 399. |
| View the control file | Click the **Run xACT** button while pressing the Ctrl key, or select **File** > **View Control File**.<br>The control file is opened in the Files page. |
| Load the rule file | Click the button on the Rules page.<br>Note: The rule file settings overwrite current GUI settings, including the path to the input design and the top cell name. |
| Specify additional rule files and rule file statements | Click **Options** on the left panel, enable "Include Rules Files," and expand the control area to specify the additional rule files.<br>To specify additional rule file statements, enable "Include Rule Statements" on the Options page.<br>By default, additional rule files and statements are added before the main rule file. To change this, enable "Include additional rules files and statements after main rules file" on the Preferences page in the Rules area. |
| Specify additional layout or schematic input files | Click **Database** on the left panel and expand the Library area. |
| Start Calibre® RVE™ after the run | Click **Run Control** on the left panel, expand the "RVE Options" area, and enable "Show Results in RVE." |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-1. Tips for the Calibre Interactive xACT GUI (cont.)**

| To do this … | Do the following: |
|---|---|
| Load or save a runset | **File** > **Load Runset** or **File** > **Save Runset**, or use Ctrl-r to save a runset.<br>By default, only non-default options are saved to the runset. To save all runset options, set the environment variable MGC_CALIBRE_SAVE_ALL_RUNSET_VALUES to 1 before starting the GUI. |
| Change the font | **Settings** > **Change Font** |
| Hide and show pages | Select **Settings** > **Show Pages** |
| Use keyboard shortcuts | You can use a hotkey to go to some pages. See "Calibre Interactive xACT GUI Reference" on page 375. |

**Note**

The runset option naming convention for some options changed in the 2018.4 release. As of 2018.4, certain options are prefixed with "cmn" instead of "pex". Runsets saved with Calibre 2018.4 or later may report errors if they are loaded with an earlier version of Calibre.

### Related Topics

Calibre Interactive xACT GUI Reference

Invoking the Calibre Interactive xACT GUI

Setting Up and Starting a Calibre Interactive xACT Run

# Populating the Load Runset Files Dialog Box and Recent Files Lists

The environment variable MGC_CALIBRE_XACT_RUNSET_LIST populates the Load Runset Files dialog box and the recent files lists for runsets and text files. Used only in Calibre Interactive xACT.

The runset list file specified by MGC_CALIBRE_XACT_RUNSET_LIST uses the following formats. You cannot mix formats within the same file.

**Basic Format** — Used to specify single runsets:

```
runset1
runset2
runset3
...
```

**Advanced Format** — Used to specify single runsets, runset groups, runset type and info, and text files.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
XACT_RS paths: runset [runset ...] [runsetType: type] [runsetInfo: info]
XACT_RS paths: runset [runset ...] [runsetType: type] [runsetInfo: info]
...
TXT paths: text_file [text_file ...]
TXT paths: text_file [text_file ...]
...
```

where file entries are defined as follows:

- XACT_RS paths: *runset_file* … — Keyword and parameter specifying one or more runset files. If more than one runset file is specified, they are grouped together in the Load Runset Files dialog box.

  Include one line for each runset file or runset group you want to list in the Recent Runsets list.

- runsetType: *type* — Optional entry with "XACT_RS paths:" that specifies information to display in the Type column of the Load Runset Files dialog box.
- runsetInfo: *info* — Optional entry with "XACT_RS paths:" that specifies information to display in the Info column of the Load Runset Files dialog box.
- TXT paths: *text_file* … — Keyword and parameter specifying one or more text files. Text files are listed one per line in the Recent Text Files list.

Both relative and absolute filenames can be used. Relative filenames are evaluated relative to the Run Directory specified on the Rules page. Environment variables can be used in the filenames.

## Prerequisites

- "Requirements for Calibre Interactive xACT" on page 352.
- A text file in the basic or advanced format, described previously.

## Procedure

1. Specify the runset list file with the environment variable MGC_CALIBRE_XACT_RUNSET_LIST. For example (in csh):

   ```
   setenv MGC_CALIBRE_XACT_RUNSET_LIST xact_runset_list
   ```

   If the file referenced by MGC_CALIBRE_XACT_RUNSET_LIST is a relative pathname, the tool searches for it in the directory from which you invoke Calibre Interactive xACT.

   You can define the environment variable as a list of multiple files:

   ```
   setenv MGC_CALIBRE_XACT_RUNSET_LIST "xact_list1 xact_list2"
   ```

   Different files can have different formats (advanced or basic), but you cannot mix formats within the same file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. (Optional) Specify the default selected runset with the environment variable MGC_CALIBRE_XACT_RUNSET_FILE. When MGC_CALIBRE_XACT_RUNSET_LIST is defined, the runsets specified by MGC_CALIBRE_XACT_RUNSET_FILE are selected if they are in the list of predefined runsets.
3. Invoke Calibre Interactive xACT.

   See "Invoking the Calibre Interactive xACT GUI" on page 352.

## Results

When you choose **File > Load Runset**, the dialog box is populated with both the listed runsets and recent runsets. The Load Runset Files dialog box includes a dropdown selection to view "Predefined Runsets" (listed runsets) or "Recent Runsets."

When you choose **File > Recent Runsets**, the list is pre-populated with the listed runsets.

If you used the advanced format runset list and specified text files, when you choose **File > Recent Text Files**, the list is pre-populated with the listed text files.

If you open other runsets or text files during the Calibre Interactive session, those files are added to the appropriate recent files list.

## Examples

### Basic Runset List

For this basic runset list:

```
runset_rcc
runset_rccl
runset_A
runset_B
```

The Load Runset Files dialog box is populated with the listed runsets. To view recent runsets, click the header and choose "Recent Runsets."

When you choose **File > Recent Runsets**, that list is also pre-populated with the listed runsets.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Advanced Runset List

This is an example of an advanced format runset list file:

```
XACT_RS paths: runset_rcc runsetInfo: RCC
XACT_RS paths: runset_rccl runsetInfo: RCCL
XACT_RS paths: runset_A runset_B
TXT paths: $PROJECT/README_Project
```

This example specifies two single runsets, *runset_rcc* and *runset_rccl*, with relative filenames; these files must exist in the Run Directory specified on the Rules pane. The optional runsetInfo entry is provided for these two runsets.

The runsets *runset_A* and *runset_B* are specified as a group. They appear as a group in the Load Runset Files dialog box, and are both loaded if selected as a group. The two runsets are listed separately in the Recent Runsets list.

The text file, *$PROJECT/README_Project*, is specified with an absolute path and uses an environment variable for the directory path.

This is a view of the Load Runset Files dialog box:

## Related Topics

Runset Options for Calibre Interactive xACT

# Using a Calibre Interactive PEX Runset with Calibre Interactive xACT

Calibre Interactive xACT can read in runsets created for Calibre Interactive PEX. Supported runset options are automatically converted.

## Prerequisites

- A Calibre Interactive PEX runset



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Invoke Calibre Interactive xACT and specify the PEX runset to load.

   You can specify the runset on the command line with "-runset pex_runset" or load the runset from the GUI with **File > Load Runset**.

   If there were issues reading the PEX runset, the view automatically switches to the Transcript page, where warning, informational, and error messages are reported.

2. If the Transcript page contains messages related to runset options, handle them as follows:

| Message | Problem and Action |
|---|---|
| WARNING … <*runset_option*>: Value of <*value*> is not supported … | The option value is not supported. Supply a valid value in the PEX runset and reload the runset, or correct the problem in the GUI. |
| WARNING … Conversion for <*runset_option*> not found -- (skipping) | The PEX runset option does not have a conversion. The option may not be valid in the Calibre Interactive xACT GUI. Confirm the validity of the option and contact your Siemens representative if needed. |
| WARNING … Keyword <*runset_option*> not recognized -- (skipping) | The PEX runset option is not recognized. Confirm the validity of the option and contact your Siemens representative if needed. |

**Note**

A customization file used with Calibre Interactive PEX cannot be used in Calibre Interactive xACT.

3. Look for icons in the left panel of the GUI, indicating an incorrect or missing setting.

   Click the page name with the error, locate the problem option, hover over the to view a message about the problem, and correct as needed.

4. View each GUI page and search for the icon. The icon indicates a warning, such as an option setting that conflicts with another option. Hover over the icon to view a message about the problem and correct as needed. Warning icons are not displayed by the page name in the left panel of the GUI.

5. Choose **File > Save Runset As** to save the runset in the new format. You are not allowed to overwrite the PEX runset.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Loading a Classic Tcl Customization File in Calibre Interactive xACT

Calibre Interactive xACT can load a Tcl customization file that was developed for classic Calibre Interactive. A new Custom page is created within the GUI and the customization controls are placed on the Custom page.

There are three methods to load the customization file: 1) with a command-line switch, 2) using a classic runset that defines the customization file, or 3) with an environment variable. A customization file specified on the command line has the highest precedence, while a customization file specified with an environment variable has lowest precedence.

Customization commands with the -tool argument are automatically taken if set to PEX and ignored otherwise.

If you invoke Calibre Interactive from a design tool, you must specify the customization file with an environment variable or specify it in the runset.

## Prerequisites

- A Tcl customization file developed for classic Calibre Interactive.

## Procedure

Do one of the following.

**Table 9-2. Loading a Classic Tcl Customization File**

| Method | Steps |
|---|---|
| Command line specification<br>(highest precedence) | Specify the Tcl customization file with the -custom command line option. For batch operation, use the -customgui command line option.<br>For example, with interactive operation:<br>`calibre -gui -xact -runset runset -custom custom.tcl`<br>Batch operation:<br>`calibre -gui -xact -runset runset -customgui custom.tcl -batch` |
| Runset specification | You can load a classic runset that has a Tcl customization file defined in the runset. (In classic Calibre Interactive, customization files that are specified on the **Startup** tab of the Setup Preferences dialog box are saved to the runset.)<br>`calibre -gui -xact -runset runset_with_customfile` |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-2. Loading a Classic Tcl Customization File (cont.)**

| Method | Steps |
|---|---|
| Environment variable specification<br>(lowest precedence) | 1. Specify the Tcl customization file with the MGC_CALIBRE_CUSTOMIZATION_FILE environment variables before invoking the GUI<br>For example:<br>`setenv MGC_CALIBRE_CUSTOMIZATION_FILE custom.tcl`<br>2. Invoke Calibre Interactive.<br>See Invoking the Calibre Interactive xACT GUI. |

### Results

The customization controls are displayed on the new Custom page. If a runset is specified on the command line, it is loaded after the Custom page is created.

In batch operation (with the -customgui option), a GUI with only the Custom page and the Run button is displayed. When the Run button is clicked the run continues in batch mode using the settings on the Custom page and in the runset.

Problems in converting the Tcl customization file are noted on the Transcript page.

The CUSTOM::SETWINGEOM command sets the size of the Calibre Interactive GUI, because there is not a separate customization window, as in classic Calibre Interactive. However, the minimum dimensions of the GUI are different than for the customization window, which may cause differences in the expected GUI size.

# Providing OpenAccess Input with Calibre Interactive xACT

You provide OpenAccess database input on the Inputs page in Calibre Interactive xACT. You specify database conversion options on the OPENACCESS page.

**Caution**

Runs started from Calibre Interactive do not use the statement SVRF FDI Options to set third-party database read options in the rule file. Make sure database conversion options are set correctly in the GUI or with MGC_CALIBRE_DB_READ_OPTIONS or the database may not be converted correctly.

### Prerequisites

- For Cadence Virtuoso users, make sure CDS_INST_ROOT or CDS_INST_DIR is set. CDS_INST_ROOT takes precedence if both CDS_INST_ROOT and CDS_INST_DIR are defined.
- Make sure LD_LIBRARY_PATH includes the libstdc++.so.6.0.8 compiler libraries. This is true by default for Linux machines with RHEL 4u6 or later.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For example with Cadence Virtuoso 6.15.500:

```
setenv LD_LIBRARY_PATH ${CDS_INST_DIR}/share/oa/lib/
linux_rhel40_gcc44x_64/opt:
${CDS_INST_DIR}/tools/cdsSkillPcell/lib/64bit:
${CDS_INST_DIR}/tools/lib/64bit:${LD_LIBRARY_PATH}
```

The actual path depends on the tool version.

- Make sure environment variables for database read are set correctly. See "FDI Environment Variables" in the *Calibre Layout Comparison and Translation Guide*.

  In particular, if your design tool requires OpenAccess version 22.50, specify this setting with one of the following options:

  - In the rules file, include the SVRF statement LAYOUT SYSTEM OA "22.50".
  - Set the environment variable MGC_FDI_OA_VERSION to 22.50.

  Calibre Interactive does this automatically when connected to Cadence Virtuoso versions 6.17 or later. The setting specified by the SVRF statement overrides MGC_FDI_OA_VERSION.

  You can use the environment variable MGC_CALIBRE_DB_READ_OPTIONS to set database read options. The settings are displayed on the OPENACCESS page in Calibre Interactive and the GUI controls are made inactive.

## Procedure

1. Click Inputs on the left panel of Calibre Interactive xACT.
2. In the Layout Path area, set Format to OPENACCESS.
3. Enter the path to the library definitions file in the "Library Defs" field. This is typically *lib.defs* or *cds.lib* (if OA_HOME is set).
4. Enter the name of the design library in the "Library Name" field.

   **Tip**

   You can click the **Browse** button to open a library browser. The library browser enables you to select the library, cell, and view, then automatically fills in the correct GUI fields.

   The "Library Defs" field must be correctly defined for the library browser to work correctly.

5. Specify the Top Cell and View Name if these have not been filled in by the library browser.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

You can define a file naming template for the library and view. Click Templates on the left panel, expand the Database area, and specify templates in the "Layout Library" and "Layout View" fields. See "Templates for File Naming" on page 148 for general information.

6. Click OPENACCESS on the left panel to set database conversion options.

   This step and the following steps are not necessary if you set database conversion options with MGC_CALIBRE_DB_READ_OPTIONS.

7. For Cadence Virtuoso users, do the following to automatically set certain Cadence environment variables:

   a. Click OpenAccess Environment.

   b. Enable "Set OA_HOME when CDS_INST_ROOT or CDS_INST_DIR is set."

   c. Enable "Set CALIBRE_READDB_LD_LIBRARY_PATH when CDS_INST_ROOT or CDS_INST_DIR is set."

   d. Enable "Set PATH when CDS_INST_ROOT or CDS_INST_DIR is set."

   e. If you are using Express Pcells, enable "Set CDS_EXP_PCELL_DIR when CDS_INST_ROOT or CDS_INST_DIR is set."

   f. If you are using Express Pcells, enable "Set CDS_EXP_ALL_PARAMS when CDS_INST_ROOT or CDS_INST_DIR is set."

   g. If you are using Express Pcells, enable "Set CDS_ENABLE_EXP_PCELL when CDS_INST_ROOT or CDS_INST_DIR is set."

8. Set other database conversion options as needed on the OPENACCESS page. For complete details, see "OpenAccess Page in Calibre Interactive xACT" on page 385.

   These options control what is read from the OpenAccess database and sent to Calibre as input.

# Specifying Probe Points in Calibre Interactive xACT

You can specify probe points in your parasitic model using Calibre Interactive. Probe points are used to verify timing from specific points on each net.

## Prerequisites

- You are running Calibre Interactive xACT, and have specified the rule file, input layout, and input netlist, as described in "Setting Up and Starting a Calibre Interactive xACT Run" on page 357.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Click the **Inputs** button on the left panel.
2. Click the **Probes** tab.
3. Do one of the following, depending on whether you have an existing probe file:
   - Click the button to select and load an existing probe file.
   - Click the button to manually add a probe point.

   The interface should look similar to that shown in the following figure.

**Figure 9-1. xACT Probes Page**

4. Click in each column to specify the required parameter. You can hover your mouse over each column header to view a tooltip for the parameter.

   To provide the probe location, click the button to choose a location in a connected layout design tool. To enter coordinates, click next to the button and enter a space-separated coordinate pair.

5. When done, click the button to save the probe file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setting Options in Calibre Interactive xACT

Most option settings in the Calibre Interactive xACT GUI are easy to use, especially with the help of tooltips and knowledge of the corresponding SVRF statements. The option settings that benefit from more explanation are described in this section.

**Connecting to a Design Tool in Calibre Interactive xACT** . . . . . 370
**Adding and Changing LVS Box Statements in Calibre Interactive xACT** . . . . . 372

## Connecting to a Design Tool in Calibre Interactive xACT

The connection to a design tool is made automatically when you invoke Calibre Interactive xACT from the design tool. You can also make the connection to the design tool manually after the GUI is invoked from the command line.

| Connection Method | Directions |
|---|---|
| Automatic connection | Invoke GUI from:<br>• Calibre DESIGNrev: **Verification** > **Run xACT**<br>• Cadence Virtuoso: **Calibre** > **Run xACT** |
| Manual connection | See the following procedure. |

See the Results section after the procedure for notes on verifying the connection.

**Note**

You can set the socket port with the environment variables MGC_CALIBRE_LAYOUT_SERVER and MGC_CALIBRE_SCHEMATIC_SERVER. See "Setting the Socket Port in Calibre Interactive" on page 707.

### Prerequisites

- "Requirements for Calibre Interactive xACT" on page 352

### Procedure

1. Open your design tool to start the socket connection and note the socket number:
   - Calibre DESIGNrev

     Select **Verification** > **RVE/CI Setup**. In the Server area of the dialog box, make a note of the Socket Number and change it if needed. If the Status icon is not green, click the **Start Server** button.
   - Cadence Virtuoso



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Select **Calibre** > **Setup** > **Socket**. Make a note of the Socket Number and change it if needed.

2. Invoke the Calibre Interactive xACT GUI; see "Invoking the Calibre Interactive xACT GUI" on page 352.
3. Click **Run Control** on the left panel.
4. Expand the "Design Tool Settings" area.
5. In the "Layout Design Tool" area, enter the socket number from Step 1 and click the **Connect** button.
6. Make settings in the "Schematic Design Tool" area as follows:

| Schematic Tool Connection | "Use the same connection as the layout tool" setting | Further Action |
|---|---|---|
| No connection | Unchecked | None |
| Same as layout tool<br>(such as for Cadence Virtuoso) | Checked | None |
| Different than layout tool | Unchecked | Enter socket number for schematic tool and click **Connect**. |

7. (Optional) Specify a Highlight Data File. This file is used when sending layout highlight data to the layout viewer.
8. Click **Inputs** on the left panel and specify design export as follows:
   a. For layout export, expand "Layout Path" and enable "Export from layout viewer."
   b. For schematic export, expand "Source Path" and enable "Export from schematic viewer."

## Results

If the connection is successful, the Status is reported as "Connected" on the Run Control pane in the "Design Tool Settings" area. In addition, the "Tool Information" display appears below the Status line with the name of the connected design tool.

**Note**

If "Use the same connection as the layout tool" is enabled and a connection attempt is made to a layout design tool that does not include a schematic tool (such as Calibre DESIGNrev), a message is displayed that the schematic connection is not supported and the setting "Use the same connection as the layout tool" is disabled.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The following image shows a layout design tool connection to Calibre DESIGNrev/WORKbench and no connection to a schematic design tool.

# Adding and Changing LVS Box Statements in Calibre Interactive xACT

You can use Calibre Interactive to override the LVS Box related statements in your rule file; this is done on the LVS page with the LVS Box, LVS Black Box Port, and LVS Black Box Port Depth options. You can add new statements, edit existing statements, and disable statements.

Corresponding rule file statements: LVS Box, LVS Black Box Port, LVS Black Box Port Depth

## Prerequisites

- The Calibre Interactive xACT GUI is open.
- A rule file is specified on the Rules pane.

## Procedure

1. (Optional) Click **Rules** on the left panel and click the button to load the rules file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

This loads the LVS Box statements in the rule file into the GUI's table of LVS Box statements. This step is required if you want to change existing LVS Box statements.

A single LVS Box statement in the rule file may specify multiple cells and pin swap lists, and the combination LAYOUT SOURCE is used if neither of those keywords is specified; when loaded into the LVS Box table such statements are broken up into multiple LVS Box statements.

2. Click **LVS** on the left panel.
3. Check the "LVS Box" checkbox and expand the option area. If you loaded a rule file with LVS Box statements, this is already done.

   When "LVS Box" is checked, the run uses only LVS Box statements from the GUI.

"LVS Black Box Port" is displayed if "LVS Box" is checked, and "LVS Black Box Port Depth" is displayed if "LVS Black box Port" is checked.

4. Select, unselect, and edit statements in the "LVS Box" table as described in the following table:

| Action | Steps |
|---|---|
| Disable or enable specific statements. | Use the checkboxes in the "Use" column to select statements to include in the run. |
| Add a statement. | Click the + button to add a statement, then double-click in the Parameters area to enter the statement parameters. |
| Edit the parameters for a statement. | Double-click in the Parameters area. |
| Remove statements. | Click the button to remove all statements, or the × button to remove a single statement. |
| Disable all LVS Box statements. | Uncheck all statements with the "Use" column. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The GUI does not check the syntax of the parameter entries.

5. If needed, expand the “LVS Black Box Port” option area and supply (or edit) the LVS Black Box Port statement.

**Note**

The LVS Black Box Port statement is required if “LVS Box BLACK …” is specified. If you add a LVS Box BLACK statement, you must also add a LVS Black Box Port statement—the GUI does not enforce this.

6. If needed, check “LVS Black Box Port Depth” and select the hierarchy level.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Interactive xACT GUI Reference

The settings for a Calibre xACT run are made in several pages selected from the left panel of the GUI.

In most cases the name of the GUI control corresponds to an SVRF command or command option. Tooltips provide additional information. Choose **Settings** > **Show Pages** to hide and show certain pages. Some pages have a shortcut, or hotkey, for going to the page; this is indicated in parentheses.

**Table 9-3. Calibre Interactive xACT GUI Items**

| GUI Item | Description |
|---|---|
| Rules Page<br>(Alt+r / Alt+R) | Specify the rule file and the run directory on the Rules page.<br>Click the button to load rule file settings into the GUI. The rule file settings overwrite current GUI settings, including the path to the input design and the top cell name.<br>To specify additional rule files, click **Options** on the left panel, select “Include Rules Files,” and expand the control area to specify the additional rule files. |
| Inputs Page<br>(Alt+i / Alt+I) | Specify the input layout, schematic source (if used), and Xcell and Hcell files.<br>To specify additional layout or schematic input files, click **Database** on the left panel and expand the Library area. |
| Outputs Page<br>(Alt+o / Alt+O) | Specify parameters related to the output of the run, such as the extraction mode, process corner, PEX Netlist, Mask SVDB, and report options. |
| Options Page<br>(Alt+p / Alt+P) | Specify options related to extraction. |
| LVS Page | Specify LVS related options. You can specify the following SVRF statements: LVS Power Name, LVS Ground Name, LVS Recognize Gates, LVS Filter Unused Option, and LVS Report Option. |
| Inductance Page | Specify options related to inductance. This page is only displayed if inductance extraction is selected on the Outputs page or if “PEX Xact Control File” is checked on the Options page. |
| FieldSolver Page | Specify fieldsolver settings. You can specify the following SVRF statements: PEX Fieldsolver Mode … NETS, PEX Fieldsolver Mode … LAYERS, PEX Fieldsolver Boundary, and PEX Fieldsolver Cell_array.<br>This page is only displayed if the “xACT Mode” on the Outputs page is “xACT 3D” or “xACT 3DSelect.” |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-3. Calibre Interactive xACT GUI Items (cont.)**

| GUI Item | Description |
|---|---|
| Database Page | Specify the SVRF statements Resolution and Precision, and additional layout and schematic input files. |
| OPENACCESS Page | Specify database conversion options for third party databases.<br>The page is only displayed if "Layout Format" on the Inputs page is OPENACCESS.<br>Also see "Providing OpenAccess Input with Calibre Interactive xACT" on page 366. |
| Environment Page | View and specify environment variables for the run. |
| V2LVS Page | Specify options for the v2lvs translator, which translates VERILOG format source input to SPICE format.<br>The V2LVS page is available when VERILOG or MIXED is selected as the "Source Format" on the Inputs page. This page is similar to the Setup Verilog Translator dialog box in Calibre Interactive nmLVS—see "Setting Up the Verilog Translator (v2lvs)" on page 254. |
| Run Control Page<br>(Alt+c / Alt+C) | Specify the run mode, processor information, options for invoking Calibre RVE, whether to run Calibre nmLVS, and settings for connecting to a design tool.<br>See "Connecting to a Design Tool in Calibre Interactive xACT" on page 370. |
| Triggers Page | Specify trigger functions that execute before and after the run. |
| Templates Page | Specify settings for file-naming templates. |
| Preferences Page | Specify settings for loading and saving the runset, saving the transcript, and loading a custom configuration file. |
| Search Toolbar<br>(Ctrl+f / Ctrl+F) | Search the GUI for a specified string. |
| Transcript Page<br>(Alt+t / Alt+T) | Display the transcript of the run. Errors and warnings are listed in the bottom portion of the page.<br>Right-click in the text display for a menu. Also see Transcript preferences on the Preferences Page. |
| Run xACT button<br>(Alt+d / Alt+D) | Starts the Calibre xACT run.<br>To view the Calibre Interactive control file, click the **Run xACT** button while pressing the Ctrl key. |
| Start RVE button | Open Calibre RVE and load the Mask SVDB database specified on the Outputs page. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Errors and warnings are indicated as follows:



A missing or incorrect setting. Hover over the icon for a message about the error. A red exclamation point ( ! ) is also displayed by the page name if any settings have an error.

A warning, such as for a conflicting setting. The run can proceed with warnings. Hover over the icon for a message about the warning. The warning icon is not displayed next to the page name.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Outputs Page in Calibre Interactive xACT

To access: Click **Outputs** in the left panel of the Calibre Interactive xACT GUI.
(shortcut: Alt-o)

The Outputs page controls the extraction mode and statements related to the extracted netlist and results database.

## Objects

In most cases the options have an informative tooltip or correspond directly to a keyword in an SVRF statement. Refer to the *SVRF Manual* for keyword descriptions.

| Option | Description |
|---|---|
| xACT Mode | Select an extraction mode of xACT, xACT 3D, or xACT 3D Select. |
| Extraction Type | Choose the parasitics to extract. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Option | Description |
|---|---|
| PEX Xact Corner | Specify process corners with the PEX XACT Corner statement.<br>Do not specify this option in the GUI if your rule file includes the PEX XACT Corner statement. This option cannot be specified with the “Corner” option in the GUI.<br>The extracted netlist path in the PEX Netlist statement is determined automatically by the GUI based on the corner information and cannot be edited.<br>If the netlist format is CALIBREVIEW, an extracted netlist and Calibre View is created for each corner. |
| Corners | Specify the process corners to write out. This option uses the -corner command-line option.<br>This option cannot be specified with the “PEX Xact Corner” option in the GUI. |
| PEX Netlist | Specify the options for the PEX Netlist statement. Use the GUI tooltips or refer to the *SVRF Manual* for the definition of the various options.<br>This area includes the option “View Netlist after xACT finishes.” |
| Mask SVDB | Specify options for the Mask SVDB Directory statement. Enable the checkbox to use this statement.<br>Tip: Enable “Show Results in RVE” on the Run Control page in the RVE Options area to start Calibre RVE automatically after the run. |
| Reports | Specify the LVS Report and PEX Report.<br>This area includes the options “View LVS Report after run finishes” and “View PEX Report after run finishes.” “PEX Report” must be enabled in order for “View PEX Report after run finishes” to be displayed. |
| PEX Report Point2Point | See PEX Report Point2Point in the *SVRF Manual* for the meaning of the available options. |
| Formatter | Specify options to configure the Calibre xACT formatter. The formatter step is not run separately; the following command line options are added to the -xact step.<br>• Ground all coupling capacitors — Adds the -g option.<br>• Display formatter warnings — Adds the -fmt_warnings option.<br>• Display formatter information — Adds the -fmt_info option. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Related Topics

Calibre Interactive xACT GUI Reference

Setting Up and Starting a Calibre Interactive xACT Run



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# LVS Page in Calibre Interactive xACT

To access: Click **LVS** in the left panel of the Calibre Interactive xACT GUI

You can adjust LVS-related settings on the LVS page. These include power and ground net names, gate recognition, device filtering, LVS Report options, and LVS Box settings.

**Tip**

Settings in the GUI override those in the rule file. To populate the GUI with the rule file settings, click **Rules** on the left panel and click the button (to the right of the text entry field for the rule file).

## Objects

- Power Supply

  Specify the power and ground nets. (LVS Power Name and LVS Ground Name statements)

- LVS Recognize Gates

  Enable the checkbox next to “LVS Recognize Gates” to specify gate recognition options using the GUI. See LVS Recognize Gates in the *SVRF Manual* for details.

- LVS Filter Unused Option

  Enable the checkbox next to “LVS Filter Unused Option” to control the filtering of unused devices using the GUI. Do the following for each option:

  a. Enable or disable the checkbox in the “Use” column. Hover over the option keyword to see a description.

  b. (Optional) Double-click in the “Design Type” column to select whether the option applies to the layout, source, or both.

  c. (Optional) Double-click in the “Devices” column to specify which devices the option applies to. Specify devices as “type” or “type (sub-type)”, without the quotes. For example: MP or MN(NZ)

  See LVS Filter Unused Option for details.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- LVS Report Option

  Enable the checkbox next to “LVS Report Option” and select the desired keywords in the dropdown list.

  See LVS Report Option for details. For most Calibre runs, you should not specify any of these options, unless you actually need the information they provide. The default report is usually all that is needed for most runs.

- LVS Box

  Enable the checkbox next to “LVS Box” to control LVS Box related statements using the GUI. See “Adding and Changing LVS Box Statements in Calibre Interactive xACT” on page 372.

  The option “LVS Black Box Port” is displayed if “LVS Box” is checked, and “LVS Black Box Port Depth” is displayed if “LVS Black Box Port” is checked.

- LVS Execute ERC

  Enable the checkbox next to “LVS Execute ERC” to specify to execute ERC operations during the circuit extraction phase. See LVS Execute ERC for details.

  The option “ERC Results Database” is displayed if “LVS Execute ERC” is checked. Under that option, specify the path to the ERC results database, how to handle pseudocell results, and whether to output hierarchical ERC errors in their context cells. See ERC Results Database and ERC Cell Name for details.

- ERC Maximum Results

  Enable the checkbox next to “ERC Maximum Results” to specify a maximum result count for any given check. See ERC Maximum Results for details.

- ERC Maximum Vertex

  Enable the checkbox next to “ERC Maximum Vertex” to specify a maximum vertex count for any ERC result polygon written to the ERC results database. See ERC Maximum Vertex for details.

- ERC Summary Report

  Enable the checkbox next to “ERC Summary Report” to specify a report file name, the writing method, and whether to open the report after the run finishes. See ERC Summary Report for details.

- ERC Check Selection

  Select the ERC checks that are executed during the run. By default the checks selected in the rule file are executed. You can define a check selection recipe for ERC checks; the interface is similar to that used to define a check selection recipe in Calibre Interactive nmDRC.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Database Page in Calibre Interactive xACT

To access: Click **Database** in the left panel of Calibre Interactive xACT

The Database page has settings for the precision, resolution, and additional layout and source files.

## Objects

| Object | Description |
|---|---|
| **Units** | |
| Precision | Specifies the ratio of database units to user units.<br>Enter one floating-point value specifying the precision, or two positive integers, where *int_1*/*int_2* is the precision.<br>Rule file statement: Precision |
| Override layout value | Specifies whether to override the layout database precision. This option is only visible if the Precision option is checked.<br>Rule file statement: Layout Input Exception Severity PRECISION_RULE_FILE 1 |
| Resolution | Specifies the layout grid step-size (the number of database units on which original geometry is required to be aligned).<br>Enter a positive integer to specify an equal grid step size in the x and y directions.<br>Enter two positive integers to specify the x and y grid step size, respectively.<br>Rule file statement: Resolution |
| **Library** | |
| Additional Layout Files | Specify layout input files in addition to what is specified on the Inputs page. The additional layout files must have the same format as the layout specified on the Inputs page, otherwise they are ignored.<br>This setting is disabled when the Layout Format is set to OPENACCESS.<br>Rule file statement: Layout Path Layout Path |
| Additional Golden Layout Files | Specify additional layout files using the GOLDEN keyword in the Layout Path statement. The additional layout files must have the same format as the layout specified on the Inputs page, otherwise they are ignored.<br>This setting is disabled when the Layout Format is set to OPENACCESS. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Object | Description |
|---|---|
| Additional SPICE Files | Specify SPICE input files in addition to what is specified on the Inputs page.<br>Rule file statement: Source Path |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# OpenAccess Page in Calibre Interactive xACT

To access: Click OPENACCESS on the left panel of the GUI when the Layout Format on the Inputs page is set to OPENACCESS.

You can control database conversion options for third party databases using Calibre Interactive xACT.

**Note**

If the environment variable MGC_CALIBRE_DB_READ_OPTIONS is defined, the OPENACCESS page displays the options set with the environment variable and the GUI controls are not active.

**Caution**

Runs started from Calibre Interactive do not use the statement SVRF FDI Options to set third-party database read options in the rule file. When using Calibre Interactive, the FDI options must be set in the GUI or with MGC_CALIBRE_DB_READ_OPTIONS.

Make sure database conversion options are set correctly in the GUI or the database may not be converted correctly.

## Description

The options on the page are described in the following tables. See "Database Translation Utilities" in the *Calibre Layout Comparison and Translation Guide* for a more detailed description.

## Fields

**Table 9-4. Read Options on the OPENACCESS Page**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Read | Select objects to read in the dropdown list. The corresponding fdi2gds or fdi2oasis option is given in parentheses.<br>• Texts (-noText) — Read text objects. The default is to read text objects.<br>• Pins (-noPinShape) — Read pin shapes. The default is to read pin shapes.<br>• Fills (-outputFills) — This option is no longer used and has no effect.<br>• Blockages (-outputBlockages) — Read blockages for OpenAccess databases. The default is to not read blockages.<br>• Instance Pins (-instPinAsShape) — Read instance pin shapes. The default is to not read instance pin shapes. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-4. Read Options on the OPENACCESS Page (cont.)**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Read Net Names as Text<br>(-annotateNets TEXT [ALL \| TOP]) | Specifies that annotations of net names are added as text objects placed on a shape object.<br>The default is to add net name annotations in all cells. Specify TOP to add annotations in the top-level cell only.<br>This option is off by default. |
| Read Net Names as Property<br>(-annotateNets PROPERTY {*pname* \| *pnum*}) | Specifies that annotations of net names are added as properties[1] on the shape object. |
| Read Pin Names as Text<br>(-annotatePins TEXT) | Specifies that annotations of pin names are added as text objects placed on a pin object.<br>This option is off by default. |
| Read Pin Names as Property<br>(-annotatePins PROPERTY {*pname* \| *pnum*}) | Specifies that annotations of pin names are added as properties[1] on the pin shape. |
| Read Net Type as Property<br>(-annotateNetType) | Specifies to read net types and annotate them as a property.<br>This option is off by default. |
| Read Instance Name as Property<br>(-annotateInsts {*pname* \| *pnum*}) | Specifies to read instance names and annotate them as properties[1].<br>This option is off by default. |
| Abort on Reading Empty PCells<br>(-abortOnEmptyPCell) | Causes the Calibre application to exit when an empty PCell appears in an OpenAccess database.<br>This option is on by default in the GUI. |

[1]The *pname* and *pnum* parameters specify the property name. For fdi2gds the *pnum* is an integer. Property names are limited to integers in GDS. For fdi2oasis the *pname* can be any string.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-5. Mapping Files Options on the OPENACCESS Page**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Use Layer Map File (-layerMap) | Specifies the pathname of the layer mapping file to use when the database is read. The layer mapping file maps specific input layers to a user-specified list of output layers and datatypes. By default, the application maps objects with their corresponding layer number and a datatype of 0.<br>See the "Example: Setting the Layer Map, Cell Map, and Object Map for OpenAccess Databases" on page 85. |
| Use Object Map File (-objectMap) | Specifies the pathname of the object mapping file to use when the database is read. The object mapping file determines what blockage objects to convert, and how they are mapped.<br>See "Example: Setting the Layer Map, Cell Map, and Object Map for OpenAccess Databases" on page 85. |
| Use Net Properties File (-netProp) | Specifies the pathname of the net properties file to use when the database is read. |
| Use Cell Map File (-cellMap) | Specifies the pathname of the cell mapping file to use when the database is read. The cell mapping file maps input cell names to user-specified cell names. The contents of the cell mapping file are layout system dependent. By default, input cell names are used throughout the run.<br>See "Example: Setting the Layer Map, Cell Map, and Object Map for OpenAccess Databases" on page 85. |
| Use Input Exception File (-inputException) | Specifies the pathname to a file that gives exception levels for certain tool behaviors. |

**Table 9-6. Additional Files Options on the OPENACCESS Page**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Use Cell GDS Files (-cellGDS) | Specifies that lower-level cells and any instantiated hierarchy should be read from the specified files and directories. Only used for GDS output. |

**Table 9-7. Other Options on the OPENACCESS Page**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| OpenAccess View Lists (-viewList) | Specifies the order in which to open views for lower-level cells. By default the view list is empty. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-7. Other Options on the OPENACCESS Page (cont.)**

| Option (fdi2gds/fdi2oasis option) | Description |
|---|---|
| Write Log File (-logFile) | Specifies to write a runtime log file for the translation. |
| Write Map Info File (-mapInfo) | Specifies to write a output file with mapping information. This file shows what mappings the translator applied, including any user-specified mappings. |
| Write Layer Names File (-layerNames) | Specifies to write an output file with layer mapping information. |

**Table 9-8. OpenAccess Environment Options on the OPENACCESS Page**

| Option | Description |
|---|---|
| Set OA_HOME when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that OA_HOME is set to *$CDS_INST_DIR/share/oa* or *$CDS_INST_ROOT/share/oa* if CDS_INST_ROOT or CDS_INST_DIR is defined. CDS_INST_ROOT is used if both variables are set. |
| Set CALIBRE_READDB_LD_LIBRARY_PATH when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that *$CDS_INST_ROOT/tools/lib* and *$CDS_INST_ROOT/tools/lib/64bit* are added to CALIBRE_READDB_LD_LIBRARY_PATH if CDS_INST_ROOT is defined. The corresponding action is taken if CDS_INST_DIR is defined. CDS_INST_ROOT is used if both variables are set. |
| Set PATH when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that *$CDS_INST_ROOT/share/oa/bin*, *$CDS_INST_ROOT/tools/dfII/bin*, and *$CDS_INST_ROOT/tools/bin* are added to the PATH variable if CDS_INST_ROOT is defined. The corresponding action is taken if CDS_INST_DIR is defined. CDS_INST_ROOT is used if both variables are set. |
| Set CDS_EXP_PCELL_DIR when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that CDS_EXP_PCELL_DIR is set to */expressPcells* if CDS_INST_ROOT or CDS_INST_DIR is defined. |
| Set CDS_EXP_ALL_PARAMS when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that CDS_EXP_ALL_PARAMS is set to true if CDS_INST_ROOT or CDS_INST_DIR is defined. |
| Set CDS_ENABLE_EXP_PCELL when CDS_INST_ROOT or CDS_INST_DIR is set | Specifies that CDS_ENABLE_EXP_PCELL is set to TRUE if CDS_INST_ROOT or CDS_INST_DIR is defined. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Environment Page in Calibre Interactive xACT

To access: Click **Environment** in the left panel of the Calibre Interactive xACT GUI

The Environment page has controls to view and define environment variables for the Calibre Interactive xACT run.

## Description

You can view, edit, and add environment variables using the Environment page. Calibre Interactive maintains a distinction between rule file and runset environment variables, and between shell and runset values, as summarized here:

- **Rule file environment variable** — One referenced in the rule file, such as with a "#IFDEF $MY_VAR" statement.
- **Runset environment variable** — One created as part of your Calibre Interactive session.
- **Runset environment value** — The value defined in your Calibre Interactive session.
- **Shell environment value** — The value defined in your shell environment for the environment variable.

You can add an environment variable to have it used for the Calibre run. Such runset environment variables are saved when you save the runset. You can also override the value of a shell environment variable for the run—the new value is saved as a runset environment value and the variable is treated as a runset environment variable.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

You can also manipulate environment variables in the custom configuration file. See "envVars" on page 444.

## Objects

| Option | Description |
|---|---|
| Filter | Specify a wildcard filter to filter the environment variables displayed in the table. Only one filter string may be entered. |
| Show All<br>Show Runset Only<br>Show Rules File Only<br>(dropdown menu) | • **Show All** — Show shell, runset, and rule file environment variables.<br>• **Show Runset Only** — Show only runset environment variables. Runset environment variables are displayed in bold font.<br>• **Show Rules File Only** — Show only environment variables referenced in the rule file. |
| Edit | Click to change the variable name or value. If you change a shell environment variable it is stored as a runset variable for the run. You can also double click in the table cell to edit the entry. |
| Reset | Click to set the selected shell environment variable back to the value defined in the shell. |
| Unset | Click to unset an environment variable for the run. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Option | Description |
|---|---|
| Add | Click to add a runset environment variable. A new row is added to the table with the variable name ENV_VAR_<i>. Tip: Make sure the Filter entry is * so that the new variable name is not filtered out of the table. |
| Remove | Click to remove a runset environment variable. You cannot remove rule file environment variables. |
| Table | The table displays the environment variables for the run, subject to the Filter entry and the **Show …** dropdown selection. You can see their environment values and runset values, as well as choose which to set the variable to. |

## Related Topics

Calibre Interactive xACT GUI Reference



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Run Control Page in Calibre Interactive xACT

To access: Click **Run Control** in the left panel of the Calibre Interactive xACT GUI. (shortcut: Alt-c)

The Run Control Page in Calibre Interactive xACT has settings for the host processor and run options, connection to a design tool, and Calibre RVE invocation.

## Objects

| Category | Description |
|---|---|
| Run Calibre On | Settings for the host processor.<br>• **Local Host** — Calibre runs on the host currently running Calibre Interactive.<br>• **Remote Host** — Calibre runs on the specified remote host.<br>• **Cluster Command** — Calibre runs on a custom computer cluster. Enter the appropriate command in the Cluster Command field. You can use the replaceable parameters listed in the “Parameter Substitution Information” area.<br>**Synchronous checked** — Both steps (LVS and xACT) are run. Calibre RVE and reports are displayed when the run is finished if appropriate options are set.<br>**Synchronous not checked** — Only one step (LVS or xACT) is run. The run transcript, Calibre RVE, and reports are not displayed when the run finishes.<br>• **Platform LSF** — Calibre runs on the IBM® Spectrum™ LSF® computing system.<br>Also see the Calibre Licensing and Remote Calibre Environment options when Remote Host or Cluster Command is selected. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Category | Description |
|---|---|
| Run Calibre | Settings for run mode (single-threaded, multithreaded, or Calibre MTflex), number of processors (-turbo), and behavior if licenses are not available for all requested CPUs.<br>• **Single Thread** — Single-threaded execution.<br>• **Multiple Threads** — Multithreaded execution on a single host computer. Specify the number of CPUs to use in the entry field. (-turbo num_processors)<br>• **All Threads** — Multithreaded execution on a single host computer with all available processors. (-turbo)<br>• **Distributed (MTflex)** — Use your existing computer network as a distributed computing platform with Calibre® MTflex™.<br>Specify the remote hosts file, which is used as the parameter for the -remotefile command line argument.<br>If desired, specify Remote Data Server (RDS) and the number of servers. See "Remote Data Server" in the *Calibre Administrator's Guide*. |
| RVE Options | • **Show Results in RVE** — Automatically open the results database in Calibre RVE when the run finishes.<br>• **Start RVE after batch run** — Automatically open Calibre RVE after a batch run finishes.<br>• **Run RVE on local host** — When checked, Calibre RVE runs on the local host. When unchecked, Calibre RVE runs on the host specified for "Run Calibre On".<br>• **Close RVE before running Calibre xACT** — As described.<br>• **Close RVE when Calibre Interactive closes** — As described. |
| Design Tool Settings | Settings for connecting to a layout or schematic design tool, including the socket number.<br>See "Connecting to a Design Tool in Calibre Interactive xACT" on page 370. |
| xACT Steps | Includes the following checkbox:<br>• **Run LVS** — Run Calibre nmLVS if PEX Netlist specifies the use of source names. Also see "Check source netlist is complete before running xACT" on the Preferences page in the "Misc" area. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Category | Description |
|---|---|
| Calibre Licensing | The “Remote Calibre” section is available when “Run Calibre xACT on” is set to Remote Host, Cluster Command, or Platform LSF.<br>• **License File Name** — Select LM_LICENSE_FILE or MGLS_LICENSE_FILE as the environment variable used for licensing.<br>• **Use** — Specify whether to use the current license file value or to use the value specified in the License File Value field. |
| Calibre Interactive License Timeout | Setting the for license timeout.<br>The license timeout can also be set with the environment variable MGC_CALGUI_RELEASE_LICENSE_TIME; the default unit is hours. See “License Timeout in Calibre Interactive” on page 67 for instructions on using other units. |
| Remote Calibre Environment | This section is available when “Run Calibre xACT on” is set to Remote Host or Cluster Command.<br>Specify the value of MGC_HOME, the remote shell, and the remote username. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Triggers Page in Calibre Interactive xACT

To access: Click **Triggers** in the left panel of the Calibre Interactive xACT GUI

The Triggers page defines trigger functions that execute before and after the batch Calibre run. Trigger functions can run in the shell environment or in the design tool environment. Trigger definitions are saved in the runset. You can also specify trigger functions in the custom configuration file.

## Description

The transcript prints the following lines when a pre- or post-execution trigger is executed:

```
INFO: Running ... pre-trigger: <trigger_function>

INFO: Running ... post-trigger: <trigger function>
```

These lines are displayed in the warnings and errors section of the Transcript page as type “Info.”

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Objects

| Option | Control Type | Description |
|---|---|---|
| Pre-LVS | Tab | Trigger functions that run before Calibre LVS executes. |
| Post-LVS | Tab | Trigger functions that run after Calibre LVS finishes. |
| Pre-xACT | Tab | Trigger functions that run before Calibre xACT executes. |
| Post-xACT | Tab | Trigger functions that run after Calibre xACT finishes. |
| | Buttons | The buttons take the following actions:<br>• Add trigger at end of list<br>• Remove selected trigger<br>• Remove all triggers<br>• Move trigger up in list<br>• Move trigger down in list |
| Use | Table column | Execute or not execute the trigger function. |
| Type | Table column | • Process — Execute in the shell.<br>• Layout — Execute in the layout design tool.<br>• Schematic — Execute in the schematic design tool. |
| Trigger | Table column | The trigger function, including arguments. The arguments listed in Parameter Substitution may be used. |
| Parameter Substitution | GUI area | The listed parameters are used to pass various filenames and other values specified in the GUI to the trigger program. For example, %l is replaced with the name of the layout primary cell and passed to the trigger program. See "Trigger Parameters in Calibre Interactive xACT" on page 543. |

## Related Topics

Configuration File Functions for Trigger Functions

Configuration File Format for xACT GUI Customization

Calibre Interactive xACT GUI Reference



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Templates Page in Calibre Interactive xACT

To access: Click **Templates** in the left panel of the Calibre Interactive xACT GUI

The Templates page has settings for file-naming templates, which form the names that Calibre Interactive generates. For example, you can specify input and output filenames based on the open cell in a connected design tool.

**Note**

Templates are automatically applied if you open Calibre Interactive xACT from a design tool. The available templates depend on the selected options settings. For example, the "Source Path" template is not present if the run does not use the source netlist.

See "Templates for File Naming" on page 148 for complete information on file-naming templates.

## Objects

| Object | Object Type | Description |
|---|---|---|
| Rules | Expandable area | Templates for the rule file, run directory, and transcript file. |
| Inputs | Expandable area | Templates for input files.<br>See "Export from Viewer Preferences" on page 152 for an explanation of the "Export from layout viewer" and "Export from source viewer" template settings. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Object | Object Type | Description |
|---|---|---|
| Outputs | Expandable area | Templates for output files. |
| Database | Expandable area | Templates for the library and view names of the source and layout databases. |
| Parameter Substitution Information | Expandable area | List of replaceable parameters that can be used in templates. See Table 3-17 on page 148 in the topic "Templates for File Naming." |
| Apply Templates | Checkbox | Specifies whether to apply templates:<br>• **On** — Apply templates if a design tool is connected to the GUI session. Templates are not applied if there is no design tool connection.<br>• **Off** — Do not apply templates.<br>When "On" is selected and a runset is loaded, templates are applied at the end of the runset loading process. |
| Apply Templates Now | Button | Applies the templates to the current session. The templates are applied regardless of whether a design tool is connected. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Search Toolbar in Calibre Interactive xACT

To access: Enter a search term in the entry field in top right corner of the GUI. (shortcut: Ctrl-f)

The Search toolbar finds matches to a search string. The behavior depends on the GUI page that is active.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Description

| Active Page | Behavior |
|---|---|
| Page other than Files or Transcript | Finds matches to a search string and places the GUI control in the Search page. Matches are also made to text in tooltips, values, and keywords.<br>You can make changes to the settings, if desired. Settings made in the Search page are reflected when you view the control in its original page.<br>If a match is found in a sub-option, the top-level option and all its sub-options are displayed.<br>The search examines all pages except Transcript and Files. Only visible pages and options are searched. For example, if the Inductance page is not visible, matches to “inductance” on the Inductance page are not found. |
| Transcript or Files | Searches in only the active page and highlights matches. |

## Objects

| Object | Description |
|---|---|
| | Enter the search term. Click the dropdown button to select case-sensitive search. |
| Search page | Search results for pages other than the Files or Transcript pages are placed in the Search page. |

## Related Topics

Calibre Interactive xACT GUI Reference



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Preferences Page in Calibre Interactive xACT

To access: Click **Preferences** in the left panel of Calibre Interactive xACT

The Preferences page has settings for loading and saving the runset, saving the transcript, and loading a custom configuration file.

## Objects

| Object | Description |
|---|---|
| Startup | Specify the startup behavior for loading a runset. |
| Runset | See "Runset Preferences" on page 404. |
| Rules | See "Rules Preferences" on page 402. |
| Run | • **Stop parasitic extraction if LVS indicates a mismatch** — When enabled, the parasitic extraction run is not performed if Calibre nmLVS reports a mismatch between layout and source. |
| Transcript | Specify whether to save the transcript to file, the file name, and the save mode (replace or append).<br>If you do Save As (using right-click or Ctrl-s on the Transcript page), the new transcript file name and save mode are automatically saved as transcript preferences. |
| Errors and Warnings | Specify whether to save errors and warnings to file, the file name, and the save mode (replace or append). |
| Configuration | • **Configuration** — (checkbox) Specify whether to load configuration files on invocation.<br>• **Configuration files** — Specify one or more configuration files.<br>• **Load Configuration Files** — (button) Click to load the specified configuration file(s) in the current GUI session.<br>The configuration files are saved to the runset. If the "Configuration" checkbox is checked, the configuration files are loaded the next time you load the runset.<br>**Note:** Configuration files specified on the command line or with an environment variable have higher precedence. If either is specified, configuration files specified in the runset are not loaded and the **Load Configuration Files** is not present. |
| Misc | See "Miscellaneous Preferences in Calibre Interactive xACT" on page 405. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Rules Preferences

To access: Click **Preferences** on the left panel of Calibre Interactive xACT, then expand the Rules area. If **Preferences** is not visible in the left panel, choose **Settings > Show Pages > Preferences**.

The Rules preferences in Calibre Interactive xACT control behavior for reading, saving, and loading the rule file.

## Objects

- **Save TVF Rules to SVRF before running Calibre**

  Enables saving of TVF rules to an SVRF file. See "Calibre Interactive Operation with a TVF Rule File" on page 105.

- **Specify control file name**

  Specifies a control filename. This value is stored in the runset.

- **Write control file suitable for command line run**

  If enabled, the #IFDEF $MGC_CALIBRE_INTERACTIVE line is omitted before the DRC ICSTATION YES line in the control file, along with the corresponding #ENDIF. A control file saved with this option enabled can be used in a command line Calibre run. (Default: true)

  See "Running Batch Calibre with a Calibre Interactive Control File" on page 49.

  If the layout database format is OpenAccess, you also need to set database translation options with an environment variable; this is described in the referenced task.

- **Only compile the rules file when Run button is pressed**

  Prevents Calibre invocation when you click the **Run** button. Instead, only the rule file compiler is invoked and any compilation errors are displayed in a dialog box. If compilation succeeds, a message to that effect is printed in the Transcript page. This setting is saved to the runset.

  If "Save TVF Rules to SVRF before running Calibre" is checked, the SVRF rule file is generated.

- **Always compile the rules file when Run button is pressed**

  If enabled, the rule file is compiled in a separate step before starting the Calibre run. If compilation fails, the error is displayed in a pop-up message and the Calibre run is not started.

  By default, rule file compilation occurs as part of the Calibre run and compilation errors are displayed in the Transcript page. When this option is enabled, compilation occurs twice for a successful run.

- **Read Inputs/Outputs fields when loading rules file**

  If enabled, input and output statements from the rule file populate the Inputs and Outputs pages when a rule file is loaded with the button.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Display text of included rules files in transcript**

  Displays contents of included rule files in the transcript when Calibre is run.

  This setting takes precedence over the environment variable CALIBRE_ECHO_RULE_FILE.

- **Automatically load rules file**

  When “Automatically load rules file” is enabled, Calibre Interactive automatically loads the rule file at these times:

  - When you load a runset with a rule file definition
  - When you start a Calibre run and the rule file has been modified since it was last loaded

  **Show prompt** — Enable or disable displaying a confirmation dialog before automatically loading the rule file. If “Show prompt” is disabled, the setting for “Automatically load rules file” applies during Calibre Interactive batch operation.

- **Include additional rules files and statements after main rules file**

  If enabled, additional rules files and rule file statements are placed after the rule file specified on the Rules page. This is useful if the order of rule file statements matters, which can happen if the main rule file includes conditional compilation directives, such as #DEFINE, which can impact the compilation of statements found in an included file. This option is off by default.

  Additional rule files and rule statements are specified on the Options page with the options “Include Rules Files” and “Include Rules Statements.”



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Runset Preferences

To access: Click **Preferences** on the left panel of Calibre Interactive xACT, then expand the Runset area. If **Preferences** is not visible in the left panel, choose **Settings > Show Pages > Preferences**.

The runset preferences in Calibre Interactive xACT control runset open, close, and save behavior, and runset tagging information.

## Objects

- **Type**

  Specify a runset type.

- **Info**

  Specify descriptive information.

- **On Close Runset**
  - Prompt to save changes before closing runset — Display a prompt dialog when a modified runset is about to be closed.
  - Save changed settings before closing runset — Automatically save a modified runset before closing it.
  - Do not save changed settings — Do not save a modified runset and do not display a prompt dialog to save a modified runset.

- **Save runset each time Calibre is run**

  Save the runset automatically each time Calibre is run.

- **Save runset to run directory each time Calibre is run**

  When enabled, each time Calibre is run a copy of the runset is saved to the run directory specified on the Rules page.

  If you loaded multiple runsets and this setting is enabled, the runset is saved as *runset.autosaved*.

- **Run Calibre when runset opened**

  Start a run immediately after a runset is opened.

- **Process Local Links**

  When enabled, the file *.cgixactdb* is created in the runset directory and a symbolic link is created from *.cgixactdb* to the runset.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Miscellaneous Preferences in Calibre Interactive xACT

To access: Click **Preferences** on the left panel of Calibre Interactive xACT, then expand the Misc area. If **Preferences** is not visible in the left panel, choose **Settings** > **Show Pages** > **Preferences**.

The miscellaneous (Misc) preferences in Calibre Interactive xACT control warnings and prompts, automatic behaviors, and other options.

## Objects

- **Warn before overwriting layout database**

  Display a warning before overwriting the system- generated layout database. This occurs when the layout GDSII or OASIS database is extracted from the layout viewer. (Default: true)

- **Warn before overwriting source database**

  Display a warning before overwriting the source SPICE netlist. This occurs when the source SPICE netlist is extracted from the schematic viewer. (Default: true)

- **Check source netlist is complete before running xACT**

  Verify that the source netlist is complete before starting the run; the source netlist check is case insensitive.

  Only available when “Run LVS …” is checked on the Run Control page in the “Run Steps” area and “Use Names From” for PEX Netlist on the Outputs page is not “Layout Names.”

- **Warn when library layout file format differs from primary layout format**

  When enabled, Calibre Interactive displays a warning at runtime if additional layout files on the **Database** page in the “Library” options are not the same format as the layout file on the **Inputs** page.

- **Ignore layout and source export in batch mode**

  This setting is enabled by default, causing batch runs of Calibre Interactive to ignore the settings for “Export from layout viewer” and “Export from schematic viewer” on the Inputs page.

- **SPICE instance prefix**

  Specify the prefix that the netlister adds to each instance in the schematic. Calibre Interactive adds this prefix to instance names that are obtained by clicking in the schematic.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# GUI Customization for Calibre Interactive xACT

You can customize the GUI by loading a configuration file. You can specify controls as hidden or not editable and define custom GUI panes. You can also specify #DEFINE and Variable statements that are added to the Calibre Interactive control file.

**Tip**

You can change the font with **Settings** > **Change Font**.

**Loading a Configuration File in Calibre Interactive xACT** . . . . . 406
**Configuration File Reference** . . . . . 409
**Function Reference for GUI Customization in Calibre Interactive xACT** . . . . . 427
**Troubleshooting Configuration Files for the Calibre Interactive xACT GUI** . . . . . 556

# Loading a Configuration File in Calibre Interactive xACT

You can load a configuration file with a command line option, by specifying the configuration file with an environment variable, loading it from the GUI, or loading a runset with a configuration file specified.

Multiple configuration files can be loaded with all methods. Files are loaded in order (left to right as specified) and settings in the most recently loaded (rightmost) file take precedence if there is a conflict.

Configuration files specified on the command line have the highest precedence, while configuration files specified in a runset have the lowest precedence.

If you invoke Calibre Interactive from a design tool, you must specify the configuration file with an environment variable, specify it in the runset, or load it manually from the GUI.

## Prerequisites

- A configuration file. See "Configuration File Format for xACT GUI Customization" on page 410.

  These two videos demonstrate how to develop a configuration file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Video: Customizing the Calibre Interactive xACT GUI: How to Hide and Lock Options

Video: How To Create a Custom Page for the Calibre Interactive xACT GUI

## Procedure

Do one of the following. Assume you have configuration files named *cfgRC.cfg*, *cfgRCC.cfg*, and *cfgLM.cfg*.

**Table 9-9. Loading a Configuration File**

| Method | Steps |
|---|---|
| Command line specification<br>(highest precedence) | Specify the configuration file with the -config command line option. You may enter more than one configuration file on the command line.<br>For example:<br>`calibre -gui -xact -runset xact_runset -config cfgRC.cfg`<br>`calibre -gui -xact -config cfgRCC.cfg -config cfgLM.cfg` |
| Environment variable specification | 1. Specify the configuration file(s) with the MGC_CALIBRE_XACT_CONFIG_FILE environment variable before invoking the GUI. For example:<br>`setenv MGC_CALIBRE_XACT_CONFIG_FILE cfgRC.cfg`<br>Specify multiple configuration files as a space separated list enclosed in quotes.<br>2. Invoke Calibre Interactive.<br>See "Invoking the Calibre Interactive xACT GUI" on page 352. |
| GUI and runset specification<br>(lowest precedence) | You can specify a configuration file on the **Preferences** page in the GUI.<br>1. Click **Preferences** on the left panel.<br>2. Enable the "Configuration" checkbox.<br>3. Specify one or more configuration files. The configuration files are loaded in the order specified.<br>4. Click **Load Configuration Files**.<br>The configuration files are saved to the runset. When the "Configuration" checkbox is checked, the configuration files are loaded the next time you load the runset.<br>**Note:** Configuration files specified on the command line or with an environment variable have higher precedence. If either is specified, configuration files specified in the runset are not loaded and the **Load Configuration Files** button is not present. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Results

The configuration file(s) are loaded. If a runset is specified on the command line, it is loaded after the configuration file is initialized.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Configuration File Reference

The configuration file in Calibre Interactive contains commands and specifications to customize the appearance of the GUI.

**Configuration File Format for xACT GUI Customization** . . . . . 410
**Global Object Definitions** . . . . . 413
**Arrays and Maps** . . . . . 414
**Tool Properties** . . . . . 418
**Option Properties** . . . . . 421
**GUI Signals in Calibre Interactive xACT** . . . . . 422
**Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT** . 423
**Finding Option Names to Use in the Configuration File** . . . . . 425



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Configuration File Format for xACT GUI Customization

Input file for the Calibre Interactive xACT GUI

The configuration file specifies a custom GUI configuration. You can specify new pages, hide pages, control the visibility of options, and add controls for #DEFINE and Variable statements. You can also connect a function to a GUI action such as changing a value.

These topics are also relevant:


- "Loading a Configuration File in Calibre Interactive xACT" on page 406
- "Finding Option Names to Use in the Configuration File" on page 425


Video: Customizing the Calibre Interactive xACT GUI: How to Hide and Lock Options

Video: How To Create a Custom Page for the Calibre Interactive xACT GUI

## Format

### File Syntax

The configuration file has the following structure, where detailed information about the different elements is provided in the "Parameters" section.

```
Configuration {

   // Global object definitions

   // Configuration object definitions

   // Function definitions

   onInitialized: {
      // JavaScript commands

      // Configuration API calls

      // Function calls

      // Property assignments
   }

}
```

### Language and Formatting Rules

- Single line comments start with two slashes (//) and extend to the end of the line.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Multiple line comments start with /* and end with */. Comments cannot be nested.
- String values must be enclosed in quotes. This is generally shown in the command syntax.
- A newline within a string value is specified with “\n”. For example: “Test\nNewline”
- Whitespace, including newlines, is ignored. This means there is no need for a line continuation character.
- The semicolon (;) separates executable statements in JavaScript™[1]. While the semicolon is optional in some circumstances, it is good practice to include it at the end of executable statements.
- Commands are processed in the order they occur in the configuration file. As a consequence, for example, if the editable property for an object is set to false, a later command setting the value of the object has no effect.

## Parameters

- **Configuration { }**

  Required. It must contain the onInitialized section.

  The Configuration section can contain the following optional elements:

  - Global object definitions

    Objects defined in the Configuration section have a global scope. See “Global Object Definitions” on page 413 for the supported syntax.

  - Configuration object definitions

    The configuration objects FileIO and Command are defined in the Configuration section. See FileIO, launchCommand, and launchCommandReturnString.

  - Function definitions

    Function definitions are written in JavaScript with the following syntax:

    ```
    function functionName(arglist) {
       // JavaScript statements
    }
    ```

    The contents of the function definition follow the same rules as the onInitialized section.

1. JavaScript is a registered trademark of the Oracle Corporation.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **onInitialized: { }**

  Required. The contents of the onInitialized section are executed when the tool is initialized. If a runset is specified, it is loaded after the onInitialized section is executed. The onInitialized section can contain the following elements:

  - JavaScript commands

    Standard JavaScript constructs are allowed. See https://www.w3schools.com/js/ or one of the many other online references.

  - Configuration API calls

    The configuration API includes commands to add #DEFINE and Variable statements, add buttons, add GUI pages, print to the transcript, and more. See "Function Reference for GUI Customization in Calibre Interactive xACT" on page 427.

  - Function calls

    Functions defined in the Configuration section can be called from the onInitialized section or called from other functions.

    Functions can also be attached to a GUI signal with the connect() function. See "GUI Signals in Calibre Interactive xACT" on page 422.

    You can attach a callback function to the Run button with the tool property tool.runCallback. See "Example: Attach Callback Function to Run Button With tool.runCallback" on page 420

  - Property assignments

    Property assignments can be made for the tool in general or for a particular GUI object using this general syntax:

    ```
    tool.property = value
    tool.option.property = value
    tool.option.sub_option.property = value
    ```

    Tool properties specify overall properties for the GUI, such as the visible GUI pages. Option properties specify properties for a GUI control, such as the value, visibility, or label. See "Tool Properties" on page 418 and "Option Properties" on page 421.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Examples

```
Configuration {
   onInitialized: {
   // Create new combined "Rules and Input" page
   // Create new Netlist page and move Netlist options to it
   // Hide standard Rules, Inputs, Environment, and Run Control pages
   tool.pages = ["Rules and Inputs", "Outputs", "Netlist", "Options"]

   // Move rule file, run dir, and input settings to Rules and Inputs page
   tool.rulesFile.pages = ["Rules and Inputs"]
   tool.runDir.pages = ["Rules and Inputs"]
   tool.layout.pages = ["Rules and Inputs"]
   tool.source.pages = ["Rules and Inputs"]
   tool.hCells.pages = ["Rules and Inputs"]
   tool.xCells.pages = ["Rules and Inputs"]

   // Move Netlist options to Netlist page
   tool.pexNetlist.pages = ["Netlist"]

   //duplicate the Include and Exclude options on
   //Netlist and Options pages
   tool.pexExcludeNets.pages = ["Netlist", "Options"]
   tool.pexIncludeNets.pages = ["Netlist", "Options"]

   // Specify and lock out SVDB settings
   tool.svdb.specified = true
   tool.svdb.editable = false
   tool.svdb.dirPath.value = "svdb"
   tool.svdb.asciixref.value = true
   }
}
```

Also see the examples provided with the define, variable, and trigger command references, and the connect function.

## Global Object Definitions

Objects defined in the Configuration section of the Calibre Interactive xACT configuration file have a global scope. A global object is declared with the “property variant” keyword.

### Uninitialized Object

```
property variant myVal;
property variant myStr;
```

A value is assigned in the onInitialized section:

```
myVal = 25;
myStr = "top";
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Initialized Object

An object can be declared and initialized with a numeric value, a string, a boolean, an array, or a map of key-value pairs. A JavaScript expression can be used to initialize the value. An example is shown for each.

A numeric value:

```
property variant dist: 4.5;
property variant count: 1;
property variant count: 100 + 1;     // initialize with expression
```

A string value, which must be enclosed in quotes:

```
property variant companyName: "ACME";
```

A Boolean value, which must be true or false:

```
property variant myBool: false;
```

An array, which can have numeric, string, and Boolean values, or the objects returned by a configuration API call that creates a GUI control:

```
property variant arrShapes: ["Square", "Circle", "Triangle"];
property variant arrMixed: ["box", 5, 4]
```

A map:

```
property variant mapTest: {"name": "M1", "width": 100}
```

See “Arrays and Maps” on page 414 for important information on manipulating arrays and maps.

## Arrays and Maps

Arrays and key-value maps can be used in the Calibre Interactive xACT configuration file with some limitations.

The examples make use of the tool.transcript command, which prints a message to the Transcript page of Calibre Interactive.

### Array Syntax

An array has the following syntax:

```
[value0, value1, ...]
```

where the elements can be a mixture of numeric, string, and Boolean values, and objects returned by one of the configuration file API calls that create a GUI control. String values must



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

be enclosed in quotes. Boolean values must have the value true or false. Array indexing starts at zero.

## Map Syntax

A map is an array of key-value pairs, with the following syntax:

```
{key1: value1, key2: value2, ...}
```

where the keys can be numeric or string values and the values may be numeric, string, or Boolean values, or objects returned by one of the configuration file API calls that create a GUI control. Key values must be unique within the map. String values must be enclosed in quotes. Boolean values must have the value true or false.

## Declaring Arrays and Maps

The method for declaring an array or map depends on whether the declaration is in the Configuration or onInitialized section of the configuration file.

Object names must start with a lower-case letter [a-z].

- In the Configuration section

  An array or map is declared with the “property variant” keyword in the Configuration section and has a global scope. The following example declares and initializes the array arrNums and the map mapTest.

  ```
  property variant arrNums: [1, 2, 3, "four", "five"]
  property variant mapTest: { "shape": "square", "width": 20}
  ```

- In the onInitialized section

  An array or map is declared with the “var” keyword in the onInitialized section and has a local scope. The following example declares and initializes the array arrTemp and the map mapTemp.

  ```
  var arrTemp = [10, 9, 8, 7];
  var mapTemp = {"test": 1, 42: "forty-two"};
  ```

## Accessing Arrays and Maps

Array elements are accessed using the index number enclosed in brackets ([ ]), where the index starts at zero. For example:

```
tool.transcript("The second element of arrNums is " + arrNums[1]);
```

Map values are accessed using the key enclosed in brackets. For example, using the mapTemp object declared in the previous section:

```
tool.transcript("mapTemp test = " + mapTemp["test"] +
   "   mapTemp 42 = " + mapTemp[42]);
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Assigning Values to Arrays and Maps

The values in a globally declared array or map cannot be changed. To change a value, you can copy the global array or map object to a locally declared object, assign values, and copy the local array or map object back to the global object. This is demonstrated in the following example.

## Example

This example demonstrates global and local arrays and maps, value assignments, and the JavaScript for loop used to access the array and map values.

```
Configuration {
  property variant arrNums: [1, 2, 3, "four", "five"]
  property variant mapTest: { "shape": "square", "width": 20}

  onInitialized: {
  var i;
  var prop;
  var arrTemp = [10, 9, 8, 7];
  var mapTemp = {"test": 1, 42: "forty-two"};

   // uninitialized objects have the value undefined
   tool.transcript("prop = " + prop + "  index i = " + i);

   tool.transcript("The second element of arrNums is " + arrNums[1]);
   tool.transcript("mapTemp test = " + mapTemp["test"] +
         "   mapTemp 42 = " + mapTemp[42]);

   tool.transcript("\ninitial arrTemp");
   for (i=0; i<arrTemp.length; i++) {
      tool.transcript(arrTemp[i]);
   }
   tool.transcript("\ninitial mapTemp");
   for (prop in mapTemp) {
      tool.transcript(prop + " is " + mapTemp[prop]);
   }

   tool.transcript("\ninitial arrNums");
   for (i=0; i<arrNums.length; i++) {
      tool.transcript(arrNums[i]);
   }
   tool.transcript("\ninitial mapTest");
   for (prop in mapTest) {
      tool.transcript(prop + " is " + mapTest[prop]);
   }
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
   // change an array value
   arrTemp = arrNums;            // copy values to temp array
   arrTemp[2] = "three";         // change an array value
   arrNums = arrTemp;            // copy temp array back to global array
   tool.transcript("\nnew arrNums");
   for (i=0; i<arrNums.length; i++) {
      tool.transcript(arrNums[i]);
   }

   // add to the map and change a value
   mapTemp = mapTest;
   mapTemp["shape"] = "circle";  // change value for shape key
   mapTemp["width"] = 30;        //change value for width key
   mapTemp["metal"] = "copper";  // add a key and value
   mapTest = mapTemp;
   tool.transcript("\nnew mapTest");
   for (prop in mapTest) {
      tool.transcript(prop + " is " + mapTest[prop]);
   }
}
}
```

The following output is printed to the Transcript page:

```
prop = undefined  index i = undefined
The second element of arrNums is 2
mapTemp test = 1   mapTemp 42 = forty-two

initial arrTemp
10
9
8
7

initial mapTemp
42 is forty-two
test is 1

initial arrNums
1
2
3
four
five

initial mapTest
shape is square
width is 20

new arrNums
1
2
three
four
five
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
new mapTest
shape is circle
width is 30
metal is copper
```

## Tool Properties

Certain properties apply at the tool level to set overall behavior and appearance of the Calibre Interactive xACT GUI. For example, you can specify the pages that are visible in the GUI.

Tool properties are specified as follows, where the available properties are given in the table.

```
tool.property = value
```

**Table 9-10. Tool Properties**

| Property | Allowed Values | Description |
|---|---|---|
| blockRun | true \| false | Specifies whether to block the Calibre run when the **Run xACT** button is clicked and display a pop up dialog box. The pop up dialog box displays the string property blockReason.<br>See the example in getCalibreVersion. Note that the run is automatically blocked if a required option is missing information. |
| blockReason | string | The text displayed in the pop up dialog box that is displayed when the **Run xACT** button is clicked and blockRun is set to true. |
| defaultPages | [“*name*”, “*name*”, …] | Defines the default page or pages; if not defined, the default page is “Custom”.<br>If the pages property is not defined for a control defined in the configuration file, the control is placed on the default page(s). |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-10. Tool Properties (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| pages | [“*name*”, “*name*”, …] | Defines the visible pages, where *name* is the name that appears in the left panel of the GUI; *name* can be a standard page name or a custom page name. All standard pages are visible if this line is not included. Pages can also be controlled with the addPage and removePage functions. Pages with invalid objects are visible even when not defined by this property.<br>Example:<br>tool.pages = [“Inputs”, “Custom Page”] |
| runCallback | “*callbackFunction*” | Specifies the name of a callback function that is executed when the Run button is clicked. The function should return true or false. The callback function is automatically executed in a batch run.<br>The function must be defined in the configuration file, outside the onInitialized section.<br>If the return value is false, a pop-up warning is displayed and execution continues. Define tool.runCallbackError to include a user-defined message in the pop-up warning.<br>Calibre execution continues regardless of the return value. Use other means, such as tool.blockRun, to control execution.<br>See “Example: Attach Callback Function to Run Button With tool.runCallback” on page 420. |
| runCallbackError | string | Specifies an error message to display when *callbackFunction* returns false. When tool.runCallbackError is defined, the pop-up warning message includes a button “Show Details.” |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-10. Tool Properties (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| showSearchPage | true \| false | Specifies the visibility of the Search page. The default is true. The Search page is not controlled by the tool.pages property.<br>Example:<br>tool.showSearchPage = true |
| userRunLabelSuffix | string | Specifies a different label for the **Run xACT** button. “xACT” in the button label is replaced with the specified string.<br>For example:<br>tool.userRunLabelSuffix “3D RCC”<br>results in a button label of “Run 3D RCC”. |

## Example: Attach Callback Function to Run Button With tool.runCallback

This example uses tool.runCallback to attach a callback function to the Run button. If the callback function returns false, a warning is displayed and tool.blockRun is set to halt the run.

The callback function tests the value of an integer define control that is added to the GUI.

```
Configuration {
      property variant defvar;

onInitialized: {

   tool.addPage("Defines and Variables")

   defvar = tool.addIntegerDefine("INT_VAL_DEF")
   defvar.pages       = ["Defines and Variables"]
   defvar.label       = "Define INT_VAL_DEF"
   defvar.description = "SVRF Define for INT_VAL_DEF"
   defvar.value       = 2
   tool.runCallback = "defTest";

// end onInitialized
}
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
      function defTest() {
         tool.runCallbackError = "INT_VAL_DEF is 2";
         if (defvar.value == 2) {
            tool.blockRun = true;
            tool.blockReason = "INT_VAL_DEF cannot be 2, halting run";
            return false;
         }  else {
            tool.blockRun = false;
            return true;
         }
      }
   //end Configuration
   }
```

This simple example demonstrates the use of tool.runCallback. To display the warning message only, without halting run, remove the lines with tool.blockRun.

Alternatively, you can use the minValue and maxValue properties for addIntegerDefine to set limits on the possible values for the define control, rather than use tool.runCallback and tool.blockRun.

## Option Properties

Certain option properties are common to the all Calibre Interactive xACT GUI objects. For example, you can specify the tooltip, label, visibility, and the page on which an option is displayed. The objects created by the configuration API commands also have some of the same option properties.

See "Finding Option Names to Use in the Configuration File" on page 425 for instructions on finding the option name that corresponds to a standard GUI control (those not added with the configuration file).

With the exception of the "specified" property, the properties in the following table are available with all standard GUI control objects. The reference page for the configuration API commands that create a GUI object lists the properties available with the object; see "Function Reference for GUI Customization in Calibre Interactive xACT" on page 427.

**Table 9-11. Common Option Properties**

| Property | Allowed Values | Description |
|---|---|---|
| description | string | The tooltip. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-11. Common Option Properties (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| editable | true \| false | Specifies whether the value can be changed in the GUI.<br>This setting only affects whether the value can be changed in the GUI. The value can always be set with a runset option or the application of a template, even when editable=false. |
| expanded | true \| false | Specifies whether an option is expanded (true) or collapsed (false) when the GUI is opened. |
| label | string | The displayed label. |
| pages | list of strings | A list of the pages that the control appears on. For example: [“Outputs”, “Options”]<br>The pages property can only be set for a top-level option; it cannot be set for a sub-option. For example, you can set tool.layout.pages, but not tool.layout.format.pages. |
| promptText | string | The prompt text for a text entry box. The prompt text is only used if the value property is not specified. |
| specified | true \| false | Specifies the state of the checkbox for an optional control. The “specified” property is not used with Boolean controls (controls with *only* a checkbox). |
| value | See description column | The value for the control. For Boolean controls, which are displayed with only a checkbox, value is set to true or false. |
| visible | true \| false | Specifies whether to show or hide the control in the GUI.<br>If the control has a missing or incorrect setting, it is made visible even if this property value is set to false. |

## GUI Signals in Calibre Interactive xACT

You can connect functions to signals generated by the Calibre Interactive xACT GUI. Signals are generated when a property value changes and when a user-defined button is clicked.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-12. Signals from the Calibre Interactive GUI**

| GUI Signal | To connect a function … |
|---|---|
| Change in property value for a standard GUI control (those not added by the configuration file) | **tool.*option*.*property*Changed.connect(*cmd*)**<br>See connect. |
| Change in property value from an object added by the configuration file (for example, addDefine or addIntegerVariable) | ***ctrl_var*.*property*Changed.connect(*cmd*)**<br>See connect. |
| Button click from user-defined button | ***button_var*.activated.connect(*cmd*)**<br>See addAction. |
| New choice from a button with a dropdown list | ***button_var*.valueChanged.connect(*cmd*)**<br>See addChoicesAction. |

**Tip**

You can block a run from executing by setting the property tool.blockRun to false. See "Tool Properties" on page 418.

You can attach a callback function to the Run button with the tool property tool.runCallback. See "Example: Attach Callback Function to Run Button With tool.runCallback" on page 420.

# Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT

You can use the hyperlink option property to add a link to a custom control. The hyperlink property is supported for addHeader, addLabel, and the commands that add Variable, #DEFINE, and #UNDEFINE statements to the control file.

## Syntax

```
ctrl_var.hyperlink = "link_path[#anchor_tag]"
```

## Arguments

- ***ctrl_var*** — The variable name for the custom control.
- ***link_path*** — The path for the link. This can be a URL, an HTML file, a text file, or a PDF file. For files, some HTML browsers may require "file://" to precede the file path of the document.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- *#anchor_tag* — An optional parameter that specifies a named destination within the document at ***link_path***.

  Some typical anchor tags supported in PDF documents are #page=<page>, *#destination*, and #tag=*destination*. However, when viewing PDF documents in an HTML browser, support for the #page anchor tag depends on the browser and the browser settings.

## Description

The label for the control is displayed in blue underline font. The specified ***link_path*** is opened in your default web browser or default PDF viewer when you click on the label.

You can set a user-defined location for the web browser and PDF viewer executables with the environment variables MGC_RVE_HTML_BROWSER and MGC_RVE_PDF_VIEWER, respectively.

## Example 1

This is an example for the addLabel command:

```
var designDocLabel = tool.addLabel("Design documentation",
                                   "Opens the design document");
designDocLabel.pages = ["Custom"];
designDocLabel.hyperlink = "http://mySite.com/designDoc.html"
```

**Figure 9-2. hyperlink Property Example**

## Example 2

This example uses an anchor tag to open a PDF document at the tagged destination “Layout”. The PDF document is located in the working directory.

```
designDoc2Label.hyperlink = "designDoc.pdf#Layout"
```

## Related Topics

addHeader

addLabel

Configuration File Functions to Add #DEFINE and #UNDEFINE Statements

Configuration File Functions to Add Variable Statements



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Finding Option Names to Use in the Configuration File

The Calibre Interactive xACT configuration file uses the same option names as the runset file. You can determine option names by saving a runset in which only the options of interest have non-default settings.

The procedure gives a specific example that finds the option names corresponding to the PEX Report statement and the option to start Calibre RVE after the run is finished.

**Note**

As an alternative to the following method, you can set the environment variable MGC_CALIBRE_SAVE_ALL_RUNSET_VALUES to 1 before invoking the GUI and saving a runset. All runset options are saved, rather than just the non-default options. However, it is not always straightforward to determine the runset option that corresponds to a particular GUI option in the resulting runset file.

## Video

The video "Customizing the Calibre Interactive xACT GUI: How to Hide and Lock Options" includes a section on how to find option names.

## Procedure

1. Open Calibre Interactive xACT with default settings—that is, do not load a runset or a configuration file.

   ```
   calibre -gui -xact
   ```

2. Enter "PEX Report" in the Search text entry box at the top right corner of the GUI.

   The view automatically switches to the Search page, where matches are highlighted.

   Expand the Report area in the search results. The "PEX Report" option is highlighted, and the tooltip shows that this is the option corresponding to PEX Report.

   Change the setting for "PEX Report" so that the option no longer has its default setting.

3. Enter "RVE " in the Search text entry box. We are not sure of the name for the option to start Calibre RVE, so enter something general. Type at space the end of the search term to match complete words rather than the string "rve" within a word.

   On the Search page, expand the area for "RVE Options" and change the setting for "Show results in RVE".



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

4. Select **File** > **Save Runset** and save the runset as *xactoptions.runset*.
5. View the file *xactoptions.runset*:

```
pex.reports.pexReport.specified = true
pex.rve.autoStart.value = true
```

## Results

Only the non-default runset options are saved. For this example the option names are reports.pexReport and rve.autoStart, where the initial “pex.” is dropped. When used in a configuration file the options are preceded with “tool”, as in tool.rve.autoStart.

## Related Topics

Configuration File Format for xACT GUI Customization

Runset Options for Calibre Interactive xACT



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Function Reference for GUI Customization in Calibre Interactive xACT

The configuration file in Calibre Interactive xACT supports many commands for GUI customization.

**General Configuration File Functions** . . . . . 428
**Configuration Commands to Execute Functions and Commands** . . . . . 456
**Configuration File Functions to Add #DEFINE and #UNDEFINE Statements** . . . . . 467
**Configuration File Functions to Add Variable Statements** . . . . . 505
**Configuration File Functions for Trigger Functions** . . . . . 543
**Configuration File Functions to Add Buttons** . . . . . 548



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# General Configuration File Functions

The Calibre Interactive xACT configuration file has functions to add and remove pages, write to the transcript, add a separator line, and add text to the GUI. With the exception of the transcript function, these functions do not affect the Calibre run.

See "Configuration File Format for xACT GUI Customization" on page 410 for information on the file format and how to invoke Calibre Interactive xACT with a custom configuration file.

| Command | Description |
|---|---|
| addHeader | Adds a header label with a shaded background to the Calibre Interactive xACT GUI. |
| addLabel | Adds a text label to the Calibre Interactive xACT GUI. |
| addMaster | Specifies that the visibility of a control depends on the state of another control (the primary control) defined in the Calibre Interactive xACT configuration file. |
| addMasterSelect | Specifies that the visibility of a control depends on the state of another control (the primary control) defined in the Calibre Interactive xACT configuration file. The dependent control is displayed if the primary control is *not* selected. |
| addMastersWithExpr | Specifies that the visibility of a control depends on one or more other controls (the primary controls) that are defined in the Calibre Interactive xACT configuration file. The logic for displaying the dependent control is defined with an expression involving the primary controls. |
| addMastersWithFunction | Specifies that the visibility of a control depends on one or more other controls (the primary controls) that are defined in the Calibre Interactive xACT configuration file. The logic for displaying the dependent control is determined by a function defined in the configuration file. |
| addPage | Adds a custom page to the Calibre Interactive xACT GUI or changes the page order of an existing page. |
| addSeparator | Adds a separator line with an optional text label to the Calibre Interactive xACT GUI. |
| envVars | The tool.envVars object includes function calls to read and set environment variables from the Calibre Interactive xACT custom configuration file. |
| FileIO | The FileIO component and associated functions provide the means to perform file I/O with the Calibre Interactive xACT custom configuration file. |
| getCalibreVersion | Returns a string containing the Calibre version, for example v2017.3_0.74. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Command | Description |
|---|---|
| getProcessID | Returns the process ID for the Calibre Interactive GUI. |
| hasOwnProperty | Returns a Boolean value indicating whether the specified option name exists. |
| removePage | Removes a page from the Calibre Interactive xACT GUI. Pages with invalid objects are visible even when removed. |
| setItemOrder | Defines the order of controls on a page in Calibre Interactive. |
| transcript | Outputs text to the transcript. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addHeader

Function call for the custom configuration file for Calibre Interactive xACT

Adds a header label with a shaded background to the Calibre Interactive xACT GUI.

## Usage

### Function Syntax

***ctrl_var* = tool.addHeader("*label*"**[, "*tooltip*"]**)**

### Object Properties

*ctrl_var*.label = "*label*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.visible = true | false
*ctrl_var*.hyperlink = *link_path*

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***label***

  A required string specifying the header text.

- *tooltip*

  An optional string specifying the hover text (tooltip) for the header. If “tooltip” is not included in the function call it can be defined with the description property.

- Object Properties

  See “Option Properties” on page 421 for a description of these common properties: pages, description, and visible.

  You can specify the label and description (tooltip) properties in the command call or as property definitions.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

## Description

The function call tool.addHeader adds a header with label text to the GUI. The header has a shaded background.

The function addLabel adds a label without a shaded background.

## Examples

The following code adds a header and a define control to the default Custom page.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
var head = tool.addHeader("Corner Section",
                          "This section defines corner options");
var dCorner = tool.addDefine("USE_CORNER");
dCorner.value = true;
dCorner.label = "Use corners";
```

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addLabel

Function call for the custom configuration file for Calibre Interactive xACT

Adds a text label to the Calibre Interactive xACT GUI.

## Usage

### Function Syntax

***ctrl_var* = tool.addLabel("*label*"**[, "*tooltip*"]**)**

### Object Properties

*ctrl_var*.label = "*label*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.visible = true | false
*ctrl_var*.hyperlink = *link_path*

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***label***

  A required string specifying the text added to the GUI page.

- *tooltip*

  An optional string specifying the hover text (tooltip) for the label. If “tooltip” is not included in the function call it can be defined with the description property.

- Object Properties

  See “Option Properties” on page 421 for a description of these common properties: pages, description, and visible.

  You can specify the label and description (tooltip) properties in the command call or as property definitions.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

## Description

The function call tool.addLabel adds a text line to the GUI.

The function addHeader adds a label with a shaded background.

## Examples

The following code adds a text line and corresponding tooltip (hover text) to the default custom page.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
var lb = tool.addLabel("This page defines custom project settings",
                       "Most of these settings should not be changed");
```

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addMaster

Function call for the custom configuration file for Calibre Interactive xACT

Specifies that the visibility of a control depends on the state of another control (the primary control) defined in the Calibre Interactive xACT configuration file.

## Usage

***ctrl_var_dep*.addMaster(*ctrl_var_master*[, *value_list*])**

## Parameters

- ***ctrl_var_dep***

  The variable name for the dependent control.

- ***ctrl_var_master***

  The variable name for the primary control.

- *value_list*

  An optional list of primary control values to match.

## Description

addMaster designates that the state of a primary control (***ctrl_var_master***) determines the visibility of the dependent control (***ctrl_var_dep***). By default (without *value_list*) the dependent control is displayed only if the checkbox for the primary control is checked. If *value_list* is provided, the dependent control is displayed if the value of the primary control matches an item in *value_list*.

## Examples

### Example 1

A dependent control depends on the state of an addDefine control, which does not have a value.

```
// primary control
var dCornerMaster = tool.addDefine("USE_CORNER");
dCornerMaster.value = false;
dCornerMaster.label = "Use corners";

// #define for CORNERS_DEF, primary control is USE_CORNER
var dCornerDep = tool.addTextDefine("CORNERS_DEF", "typical")
dCornerDep.label       = "Define CORNERS_DEF"
dCornerDep.description = "Enter a space separated list of corners"
dCornerDep.allowSpaces = true;
dCornerDep.addMaster(dCornerMaster);
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The image on the left shows the initial GUI display. The image on the left shows the display after checking “Use corners.”

### Example 2

A dependent control depends on the value of a addChoicesDefine primary control. The dependent control is only displayed if the primary control is set to “adv” and the checkbox for the primary control is checked.

```
// primary for ADV_MODE define
var dModeMstr = tool.addChoicesDefine("MODE_DEFINE",
                  ["adv", "basic"],
                  ["advanced", "basic"],
                  ["advanced mode", "basic mode"]);
dModeMstr.label = "MODE_DEFINE control";
dModeMstr.optional = true;

// dependent on MODE_DEFINE value
var dModeAdv = tool.addTextDefine("ADV_MODE",
                  "p1",
                  "enter value for advanced mode");
dModeAdv.addMaster(dModeMstr, ["adv"]);
```

The following images show the different possible states of the primary and dependent control.

### Example 3

Three controls depend on the state of an addIntegerVariable control. One of the dependent controls depends on matching to a single value for the primary control, one matches to a list of values, and one depends on the state of the primary checkbox only.

```
var vIntMaster = tool.addIntegerVariable("VAR_RUN_LEVEL",
                        1, "controls run level");
vIntMaster.label = "VAR_RUN_LEVEL to control options";
vIntMaster.optional = true;
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
var vOptDep1 = tool.addTextVariable("VAR_OPTION1", "init1");
vOptDep1.label = "VAR_OPTION1 for VAR_RUN_LEVEL = 1";
vOptDep1.addMaster(vIntMaster, [1]);

var vOptDep23 = tool.addTextVariable("VAR_OPTION2_5", "init2_5");
vOptDep23.label = "VAR_OPTION23 for VAR_RUN_LEVEL 2-5";
vOptDep23.addMaster(vIntMaster, [2, 3, 4, 5]);

// this only shows up if master has optional =true, and checked
var vOptDepChk = tool.addTextVariable("VAR_OPTION_ANY", "initall");
vOptDepChk.label = "VAR_OPTION_ANY for any VAR_RUN_LEVEL";
vOptDepChk.addMaster(vIntMaster);
```

The following images show the different possible states of the primary and dependent controls.

## Related Topics

Configuration File Format for xACT GUI Customization

addMasterSelect

addMastersWithFunction

addMastersWithExpr



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addMasterSelect

Function call for the custom configuration file for Calibre Interactive xACT

Specifies that the visibility of a control depends on the state of another control (the primary control) defined in the Calibre Interactive xACT configuration file. The dependent control is displayed if the primary control is *not* selected.

## Usage

***ctrl_var_dep*.addMasterSelect(*ctrl_var_master*)**

## Parameters

- ***ctrl_var_dep***

  The variable name for the dependent control.

- ***ctrl_var_master***

  The variable name for the primary control.

## Description

addMasterSelect designates that the state of a primary control (***ctrl_var_master***) determines the visibility of the dependent control (***ctrl_var_dep***). The dependent control is displayed only if the checkbox for the primary control is *not* checked.

## Examples

A dependent control depends on the state of an addDefine control, which does not have a value.

```
// primary control
var dCornerMaster = tool.addDefine("USE_CORNER");
dCornerMaster.value = false;
dCornerMaster.label = "Use corners";

// info label when master USE_CORNER is not selected
var noCornerLb = tool.addLabel("Corners not used")
noCornerLb.addMasterSelect(dCornerMaster);
```

## Related Topics

Configuration File Format for xACT GUI Customization

addMaster

addMastersWithFunction

addMastersWithExpr



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addMastersWithExpr

Function call for the custom configuration file for Calibre Interactive xACT

Specifies that the visibility of a control depends on one or more other controls (the primary controls) that are defined in the Calibre Interactive xACT configuration file. The logic for displaying the dependent control is defined with an expression involving the primary controls.

## Usage

***ctrl_var_dep*.addMastersWithExpr(‘[’*master_list*‘]’, *"expression"*)**

## Parameters

- ***ctrl_var_dep***

  The variable name for the dependent control.

- ***master_list***

  A comma-separated list of one or more variable names for the primary controls that are used in the ***expression***.

- ***expression***

  The expression that controls whether the dependent control is displayed. The expression should return true or false and be within quotes.

## Description

addMastersWithExpr designates that the visibility of a dependent control (***ctrl_var_dep***) is controlled by an expression that involves one or more primary controls.

## Examples

The primary controls must be defined outside the onInitialized section as global variables.

```
Configuration {
    property variant primary1;
    property variant primary2;
    property variant dpndntAND;

    onInitialized: {
      primary1 = tool.addDefine("Primary1");
      primary2 = tool.addDefine("Primary1");

      dpndntAND = tool.addDefine("AND_Dependent");

      // use expression to show dpndntAND when master1 && master2 are on
      // value for addDefine is true/false for the checkbox state
      dpndntAND.addMastersWithExpr([primary1, primary2],
                                   "primary1.value && primary2.value");
    }
}
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Related Topics

Configuration File Format for xACT GUI Customization

addMaster

addMasterSelect

addMastersWithFunction



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addMastersWithFunction

Function call for the custom configuration file for Calibre Interactive xACT

Specifies that the visibility of a control depends on one or more other controls (the primary controls) that are defined in the Calibre Interactive xACT configuration file. The logic for displaying the dependent control is determined by a function defined in the configuration file.

## Usage

***ctrl_var_dep*.addMastersWithFunction(‘[’*master_list*‘]’, *function*)**

## Parameters

- ***ctrl_var_dep***

  The variable name for the dependent control.

- ***master_list***

  A comma-separated list of one or more variable names for the primary controls that are used in the ***function***.

- ***function***

  The name of the function that controls whether the dependent control is displayed. The function should return true or false.

## Description

addMastersWithFunction designates that the visibility of a dependent control (***ctrl_var_dep***) is controlled by a function that returns true or false. addMastersWithFunction is useful when the logic for determining the dependent control visibility is more complicated than can be easily specified by an expression in addMastersWithExpr.

## Examples

The primary controls must be defined outside the onInitialized section as global variables.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
Configuration {
    property variant primary1;
    property variant primary2;
    property variant dpdndntOR;

    onInitialized: {
      primary1 = tool.addDefine("Primary1");
      primary2 = tool.addDefine("Primary1");

      dpndntOR = tool.addDefine("OR_Dependent");

      // use function to show dpndntOR when primary1 || primary2 are on
      // value for addDefine is true/false for the checkbox state
      dpndntOR.addMastersWithFunction([primary1, primary2], control_or);
    }

   function control_or() {
      return (primary1.value || primary2.value);
   }
}
```

## Related Topics

Configuration File Format for xACT GUI Customization

addMaster

addMasterSelect

addMastersWithExpr



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addPage

Function call for the custom configuration file for Calibre Interactive xACT

Adds a custom page to the Calibre Interactive xACT GUI or changes the page order of an existing page.

## Usage

### Function Syntax

**tool.addPage("*page_name*"**[, "*before_page*"]**)**

## Arguments

- ***page_name***

  A required string specifying the name of a new or existing page.

- *before_page*

  An optional string specifying the name of an existing page.

## Description

If ***page_name*** does not exist, it is added. By default it is placed at the end of the page list on the left panel of the GUI, but before the pages “Search”, Transcript”, and “Files”. If *before_page* is specified, the new page is placed before the specified page.

If ***page_name*** exists, it is moved with the logic described in the preceding paragraph.

**Note**

You can also control page display with **Settings** > **Show Pages**.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addSeparator

Function call for the custom configuration file for Calibre Interactive xACT

Adds a separator line with an optional text label to the Calibre Interactive xACT GUI.

## Usage

### Function Syntax

***ctrl_var* = tool.addSeparator(**["*label*"]**)**

### Object Properties

*ctrl_var*.label = "*label*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.visible = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- *label*

  An optional string specifying a text label for the separator line. The label text can be specified in the function call or it can be defined with the label property.

- Object Properties

  See "Option Properties" on page 421 for a description of these common properties: pages, description, and visible.

  You can specify the optional *label* in the command call or as a property definition.

## Description

The function call tool.addSeparator adds a separator line to the GUI, with an optional text label.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# envVars

Object type for the custom configuration file for Calibre Interactive xACT

The tool.envVars object includes function calls to read and set environment variables from the Calibre Interactive xACT custom configuration file.

## Usage

**tool.envVars.isEnvSet("*name*")**

**tool.envVars.envValue("*name*")**

**tool.envVars.setEnv("*name*", "*value*")**

**tool.envVars.unsetEnv("*name*")**

## Parameters

### Object Functions

- **isEnvSet(“*name*”)**

  Returns a Boolean values indicating whether the environment variable ***name*** is set.

- **envValue(“*name*”)**

  Returns a string containing the value of the environment variable ***name***.

- **setEnv(“*name*”, “*value*”)**

  Sets the environment variable ***name*** to ***value***. The function can be used to change the value of an existing environment variable or to define a new environment variable. Specify ***value*** as an empty string to define the environment variable without an assigned value. The new settings are used within the current Calibre Interactive session. There is no return value.

- **unsetEnv(“*name*”)**

  Unsets the environment variable ***name*** within the current Calibre Interactive session. There is no return value.

## Examples

```
Configuration {
onInitialized: {

// set the inductance extraction mode depending on an env var
if (tool.envVars.isEnvSet("PEX_RUN_INDUCTANCE_LM")) {
   tool.transcript("PEX_RUN_INDUCTANCE_LM is set, setting LM extraction");
   tool.pexExtractionType.inductanceMode.value = "lm";
}
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
// set the netlist format based on the value of an env var
// note, this code does not have any error checking for the value,
// it assumes the env var is set to valid values for the runset option
if (tool.envVars.isEnvSet("PEX_NETLIST_FORMAT"))  {
   if (tool.envVars.envValue("PEX_NETLIST_FORMAT") != "") {
      tool.transcript("PEX_NETLIST_FORMAT is set to "
            + tool.envVars.envValue("PEX_NETLIST_FORMAT"));
      tool.pexNetlist.netlistFormat.value =
               tool.envVars.envValue("PEX_NETLIST_FORMAT");
   }
}

// end onInitialized
}
// end Configuration
}
```

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# FileIO

Component definition for the custom configuration file for Calibre Interactive xACT

The FileIO component and associated functions provide the means to perform file I/O with the Calibre Interactive xACT custom configuration file.

## Usage

### FileIO Component Definition

**FileIO {**
**id:** ***fileID***;
**fileName:** "***file_name***";
[accessMode: FileIO.*file_mode*;]
**}**

### FileIO Component Properties

*fileID*.fileName

### FileIO Component Functions

*fileID*.exists()
*fileID*.open()
*fileID*.close()
*fileID*.readLine()
*fileID*.writeLine("*string*")
*fileID*.endOfFile()
*fileID*.isDir()
*fileID*.isFile()
*fileID*.isReadable()
*fileID*.isWritable()
*fileID*.dirName()

## Parameters

- ***fileID***

  A string by which the file component is accessed. The ***fileID*** must be defined in the FileIO component definition. It must start with a lowercase letter ([a-z]).

- ***file_name***

  A string specifying the name of the file. The ***file_name*** must be defined in the FileIO component definition and can be accessed with the fileName component property.

- FileIO.*file_mode*

  Specifies the mode in which the file is opened.

  FileIO.Read — Open the file in read-only mode. This is the default mode if accessMode is not defined in the FileIO component definition.

  FileIO.Write — Open the file in write-only mode. The file is created if it does not exist. If the file exists, it is truncated to zero length on open.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

FileIO.Append — Open the file in append mode, with no read access. The file is created if it does not exist. If the file exists, the file offset marking the current position in the file is placed at the end of the file.

- FileIO Component Functions

**Table 9-13. Functions for FileIO Component**

| Function | Return Value |
|---|---|
| exists() | Boolean |
| open() | Boolean |
| close() | None |
| readLine() | String (the current line), or an empty string on failure |
| writeLine("*string*") | None |
| endOfFile() | Boolean |
| isDir() | Boolean |
| isFile() | Boolean |
| isReadable() | Boolean |
| isWritable() | Boolean |
| dirName() | String (the directory name in which the file exists), or an empty string on failure |

## Description

The FileIO component definition is placed before the onInitialized section of the configuration file. The FileIO component functions are used inside the onInitialized section.

## Examples

This example opens two files, one for reading and one for writing. It prints two lines from the read-access file, and writes the process id to the write-access file. Some basic file property checks are performed before opening the read-access file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
Configuration {

FileIO {
    id: fileRD1;
    fileName: "test_read";
}

FileIO {
    id: fileWr_log;
    fileName: "log_file";
    accessMode: FileIO.Write;
}

onInitialized: {

if (fileRD1.exists() && !fileRD1.isDir() && fileRD1.isReadable() ) {
   fileRD1.open();
   tool.transcript("Reading from "+ fileRD1.fileName +
      " in directory " + fileRD1.dirName() );
   tool.transcript("   line 1: " + fileRD1.readLine());
   tool.transcript("   line 2: " + fileRD1.readLine());
   fileRD1.close();
} else {
    if (!fileRD1.exists()) {
         tool.transcript(fileRD1.fileName + " doesn't exist");}
    if (fileRD1.isDir()) {
         tool.transcript(fileRD1.fileName + " is a directory");}
    if (!fileRD1.isReadable()) {
         tool.transcript(fileRD1.fileName + " is not readable");}
}

fileWr_log.open();
fileWr_log.writeLine("Log file, process id " + tool.getProcessID());
fileWr_log.close();

// end onInitialized
}

//end Configuration
}
```

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# getCalibreVersion

Function call for the custom configuration file for Calibre Interactive xACT

Returns a string containing the Calibre version, for example v2017.3_0.74.

## Usage

**tool.getCalibreVersion()**

## Arguments

None.

## Examples

This example parses the year portion of the Calibre version string. If the year is less than 2018 the run is blocked by setting the blockRun property to true.

```
var str_ver = tool.getCalibreVersion();
var str_yr = str_ver.substr(1,4);
if (str_yr < 2018) {
 tool.blockRun = true;
 tool.blockReason = "version year " + str_yr + " is less than 2018";
}
```

For Calibre versions earlier than 2018, the run is not started when the **Run xACT** button is clicked and the following dialog box is displayed:

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# getProcessID

Function call for the custom configuration file for Calibre Interactive xACT

Returns the process ID for the Calibre Interactive GUI.

## Usage

**tool.getProcessID()**

## Arguments

None.

## Examples

This example prints the process id to the transcript.

```
var gui_pid = tool.getProcessID();
tool.transcript("process id " + gui_pid);
```

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# hasOwnProperty

Function call for the custom configuration file for Calibre Interactive xACT

Returns a Boolean value indicating whether the specified option name exists.

## Usage

### Syntax 1

**tool.hasOwnProperty(*'optionName'*)**

### Syntax 2

**tool.*optionName*.hasOwnProperty('*subOptionName*')**

## Parameters

- ***optionName***

  An option name, such as rulesFile, runDir, layout, or source.

- ***subOptionName***

  A sub-option to ***optionName***. For example, layoutFile or format, where ***optionName*** is layout.

## Description

Returns a Boolean value indicating whether the specified option name exists. The function hasOwnProperty is useful in controlling the logic flow in the configuration file depending on the existence of an option.

Syntax 1 can be used with an option name that does not have sub-options (such as rulesFile and runDir) and with an option name that has sub-options (such as layout and source).

Syntax 2 is used to determine if a specific ***subOptionName*** exists for ***optionName***. The syntax can be extended as needed for multiple levels of sub-options.

## Examples

```
if (tool.hasOwnProperty('rulesFile')) {
        tool.transcript("rule file option exists");
    } else {
        tool.transcript("rule file option does not exist");
    }

if (tool.layout.hasOwnProperty('topCell')) {
         tool.transcript("layout.topCell option exists");
    } else {
         tool.transcript("layout.topCell option does not exist");
   }
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# removePage

Function call for the custom configuration file for Calibre Interactive xACT

Removes a page from the Calibre Interactive xACT GUI. Pages with invalid objects are visible even when removed.

**Note**

You can also control page display with **Settings** > **Show Pages**.

## Usage

### Function Syntax

**tool.removePage("*page_name*")**

## Arguments

- ***page_name***

  The name of an existing page.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# setItemOrder

Defines the order of controls on a page in Calibre Interactive.

## Usage

**tool.setItemOrder("*page_name*", *item_list*)**

## Arguments

- ***page_name***

  The name of the containing the items to be ordered.

- ***item_list***

  A list of items, or controls, on ***page_name***. Only top-level items can be listed. For custom controls, the control variable name is used. For built-in controls, the runset option name is used.

  Any items on ***page_name*** that are not included in ***item_list*** are placed after the items in ***item_list***, in the order they were created.

## Description

Defines the order of items, or controls, on the GUI page ***page_name***.

All controls in ***item_list*** must be defined before the setItemOrder command is used.

## Examples

This example creates two checkboxes on the Rules page to add #DEFINE statements to the control file. The order of one checkbox is specified with setItemOrder, while the order of the other checkbox is not defined.

```
Configuration {

onInitialized: {

// add define controls
var advDebug = tool.addDefine("DEBUG");
advDebug.description = "Include advanced debugging tools";
advDebug.pages = "Rules";

var defCorner = tool.addDefine("CORNER");
defCorner.description = "Include CORNER option";
defCorner.pages = "Rules";

tool.setItemOrder("Rules", [tool.rulesFile, advDebug]);

// onInitialized
}
//Configuration
}
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The advDebug control is placed after the Rules File control. All remaining controls are placed in order, so the custom control defCorner is placed after the built-in control for the Run Directory.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# transcript

Function call for the custom configuration file for Calibre Interactive xACT

Outputs text to the transcript.

## Usage

### Function Syntax

**tool.transcript("*text*")**

## Arguments

- ***text***

  The text to send to the transcript.

## Description

The specified text is sent to the transcript.

When viewed on the Transcript page, text that begins with “ERROR:”, “WARNING:”, and “INFO:” is color coded within the transcript and also displayed in a sorted list at the bottom of the Transcript page.

When multiple configurations are loaded, messages from each configuration file are appended to the transcript window.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Configuration Commands to Execute Functions and Commands

The custom configuration file in Calibre Interactive has commands available that call other commands or functions. The commands specify a callback function and provide a way to execute a shell command.

| Command | Description |
|---|---|
| connect | Specifies a function to execute on a signal from the Calibre Interactive xACT GUI. |
| launchCommand | The launchCommand function executes an external (shell) command. |
| launchCommandReturnString | The launchCommandReturnString function executes an external (shell) command that returns a string. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# connect

Function call for the custom configuration file in Calibre Interactive xACT

Specifies a function to execute on a signal from the Calibre Interactive xACT GUI.

## Usage

### Syntax 1: Connect to a Signal from an addAction Button

***button_var*.activated.connect(*cmd*)**

### Syntax 2: Connect to a Signal from an AddChoicesAction Button

***button_var*.valueChanged.connect(*cmd*)**

### Syntax 3: Connect to Change Signal from a Property

**tool.*option*.*property*Changed.connect(*cmd*)**

***ctrl_var*.*property*Changed.connect(*cmd*)**

## Parameters

- ***button_var***

  The variable name for an AddAction or AddChoicesAction button.

- ***cmd***

  A function name. The function must be defined in the Configuration section of the configuration file.

  When used in Syntax 1 and 3, the function cannot have arguments. In Syntax 2 (for the addChoicesAction button), the function must accept one argument corresponding to the selection in the dropdown list.

- ***option*.*property***
  - ***option*** — An option name for a standard GUI controls (those not added by the configuration file). See "Finding Option Names to Use in the Configuration File" on page 425. If the option has sub-options, the syntax can be extended as needed, for example: ***option*.*subOption*.*property***.
  - ***property*** — An option property. See "Option Properties" on page 421.

- ***ctrl_var*.*property***
  - ***ctrl_var*** — The variable name for an object added by the configuration file; for example, and addDefine or addIntegerVariable control object.
  - ***property*** — A property associated with ctrl_var object. See the relevant command reference for the list of available object properties.

## Description

The connect function specifies a function to call when a signal is received from the Calibre Interactive GUI.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Syntax 1: Connect to a Signal from an AddAction Button

The activated.connect(cmd) function specifies the function that is executed when an addAction button is clicked.

### Syntax 2: Connect to a Signal from an AddChoicesAction Button

The valueChanged.connect(cmd) function specifies the function that is executed when an a new selection is made in the dropdown list of an addChoicesAction button. The new control value (*ctrl_var*.value) is passed to the cmd function.

### Syntax 3: Connect to Change Signal from a Property

The connect functions specifies a function that is called when a property value changes. Any valid property may be used. The properties “specified” and “value” are typically used to call a function when the state of a checkbox or the option value changes.

**Note**

For “value” properties that are provided by means of a text entry field in the GUI, the callback ***cmd*** is called at each character entry.

See the addAction and addChoicesAction functions for examples that connect a function to a user-defined button in the GUI. To add a callback function to the Run button, use the tool property tool.runCallback, as described in “Example: Attach Callback Function to Run Button With tool.runCallback” on page 420.

## Examples

The following examples connect a function to a change signal from a property.

### Example 1: Callback Function to Set Options when Checkbox Enabled

This example sets options associated with the SVDB database if the checkbox state is changed to true.

```
Configuration {
   onInitialized: {
   // Attach setSVDBOptions() as a callback function
   // if the svdb checkbox is changed
   tool.svdb.specifiedChanged.connect(setSVDBOptions)
   }
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
      // define a function to set SVDB options if the
      // Mask SVDB checkbox is changed to true
      function setSVDBOptions() {
         var checkbox = tool.svdb.specified

         if (checkbox) {
            tool.svdb.query.value = true
            tool.svdb.asciixref.value = true
            tool.svdb.cci.value = true
            tool.svdb.genParasiticData.value = true
            tool.svdb.pinloc.value = "PINLOC"
            tool.svdb.pinloc.specified = true
         }
      }
   }
```

### Example 2: Callback Function to Set Parasitic Netlist File Name

This example defines a callback function to set the parasitic netlist file name based on the top cell name and the netlist format. The callback function is executed if the top cell name or netlist format changes. The example also places the PEX Netlist options on a new page named Netlist.

```
Configuration {
   onInitialized: {
   // Attach updateNetlistFile() as a callback function
   // if netlist format or topcell name changes
   tool.pexNetlist.netlistFormat.valueChanged.connect(updateNetlistFile)
   tool.layout.topCell.valueChanged.connect(updateNetlistFile)
   } //onInit
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
// define the function to construct the parasitic netlist name
function updateNetlistFile() {
  var topcell = tool.layout.topCell.value
  var format = tool.pexNetlist.netlistFormat.value

  switch (format) {
    case "HSPICE":
      tool.pexNetlist.netlistFile.value = topcell.concat(".hsp")
      break;
    case "DSPF":
      tool.pexNetlist.netlistFile.value = topcell.concat(".dspf")
      break;
    case "SPEF":
      tool.pexNetlist.netlistFile.value = topcell.concat(".spef")
      break;
    case "SPECTRE":
      tool.pexNetlist.netlistFile.value = topcell.concat(".spectre")
      break;
    case "ELDO":
      tool.pexNetlist.netlistFile.value = topcell.concat(".eldo")
      break;
    case "CALIBREVIEW":
      tool.pexNetlist.netlistFile.value = topcell.concat(".calibre")
      break;
    case "SPICE":
      tool.pexNetlist.netlistFile.value = topcell.concat(".sp")
      break;
    default:
      break;
  }
}

}//Configuration
```

### Example 3: Callback Function for AddDefine Control

```
Configuration {
// make control variable global
property variant def_DEBUG;

function printDebug() {
tool.transcript("debug on/off changed: " + def_DEBUG.value);
}

onInitialized: {
// add a new page named "Defines and Variables"
tool.addPage("Defines and Variables")

// #define for DEBUG_ON
def_DEBUG = tool.addDefine("DEBUG_ON")
def_DEBUG.pages       = ["Defines and Variables"]
def_DEBUG.label       = "Define DEBUG_ON for debug"
def_DEBUG.description = "Turn debug on or off"
def_DEBUG.value       = false;
def_DEBUG.valueChanged.connect(printDebug);

} //onInit
} //Config
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# launchCommand

Command definition and function call for the custom configuration file for Calibre Interactive xACT

The launchCommand function executes an external (shell) command.

## Usage

***cmdID*.launchCommand**

### Command Definition

**Command { id: *cmdID*;**
[commandString: "*cmd*"**;**]
**}**

## Arguments

- ***cmdID***

  The ID for the command. It must start with a lowercase letter ([a-z]).

- *cmd*

  The external command to execute, including any arguments. The command must be in your path.

  The *cmd* can be defined in the command definition or with the commandString property. The *cmd* string must be enclosed in double quotes.

- Object Properties

  commandString — The command to execute. See the *cmd* argument description.

## Return Values

The return value from the *cmd*.

## Description

The launchCommand function provides a way to call an external command, such as echo. You can call launchCommand from a function, call it directly in the onInitialized section of the configuration file, or use it as the connect command.

For example, if bSVDB is a addAction control variable, the following launches myCmd when the button is clicked:

```
bSVDB.activated.connect(myCmd.launchCommand);
```

## Examples

### Example 1

This example defines a button that executes the setSVDB callback function. The user-defined function echoVar prints statements to the console. The commandString property is redefined within the echoVar function so that the new option values are printed.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
Configuration {
   onInitialized: {
   // add button to turn on svdb options
   var bSVDB = tool.addAction("SVDB options"); //name is internal
   bSVDB.pages = ["Rules"];
   bSVDB.label = "SVDB options"; // the button label
   bSVDB.description = "turn on SVDB and some options"; // tooltip
   bSVDB.activated.connect(setSVDB);

   } // end onInitialized


// user-defined command echoVar
Command { id: echoVar;
      commandString: "echo var change: ";
}

// define function to set SVDB options if the Mask SVDB checkbox is set
function setSVDB() {
   echoVar.launchCommand();

   tool.svdb.query.value = true
   echoVar.commandString = "echo --svdb.query " + tool.svdb.query.value;
   echoVar.launchCommand();

   tool.svdb.asciixref.value = true
   echoVar.commandString = "echo --ascii " + tool.svdb.asciixref.value;
   echoVar.launchCommand();

}

} // end Configuration
```

### Example 2

This example calls the touch shell command. The return code is tested and a message printed to the transcript. The command string is defined in the command definition.

```
Configuration {

   Command { id: cmdTouch;
      commandString: "/bin/touch test.txt";
   }

   onInitialized: {

      if (cmdTouch.launchCommand() == 0) {
         tool.transcript("touch success\n");
      } else {
         tool.transcript("touch failed\n");
      }

   } // end onInitialized

} // end Configuration
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# launchCommandReturnString

Command definition and function call for the custom configuration file for Calibre Interactive xACT

The launchCommandReturnString function executes an external (shell) command that returns a string.

## Usage

***cmdID*.launchCommandReturnString**

### Command Definition

**Command { id: *cmdID*;**
[commandString: "*cmd*"**;**]
**}**

### Object Properties

*cmdID*.returnCode
*cmdID*.commandString

## Arguments

- ***cmdID***

  The ID for the command. It must start with a lowercase letter ([a-z]).

- *cmd*

  The external command to execute, including any arguments. The command must be in your path and should return a string value.

  The *cmd* can be defined in the command definition or with the commandString property. The *cmd* string must be enclosed in double quotes.

- Object Properties

**Table 9-14. Obejct Properties for launchCommandReturnString**

| Property | Description |
|---|---|
| returnCode | The return code from the command (*cmd*). |
| commandString | The command to execute. See the *cmd* argument description. |

## Return Values

The string value returned by the *cmd*.

## Description

The launchCommandReturnString function provides a way to call an external command, such as readlink, that returns a string. You can call launchCommandReturnString from a function,



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

call it directly in the onInitialized section of the configuration file, or use it as the connect command.

## Examples

This example uses the readlink command to print the full path for the file *README*. When successful, the example actually prints the full path twice: once when readlink is executed and once when the transcript command prints the variable readmePath containing return string.

```
Configuration {

   Command { id: cmdPrint;
      commandString: "/usr/bin/readlink -f ./README";
   }

   onInitialized: {

      var readmePath = cmdPrint.launchCommandReturnString();

      if (cmdPrint.returnCode === 0) {
           tool.transcript("\nREADME path:  " + readmePath);
      } else {
           tool.transcript("\ncmdPrint failed");
      }

   } // end onInitialized

} // end Configuration
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Configuration File Functions to Add #DEFINE and #UNDEFINE Statements

Several functions are available for the Calibre Interactive xACT custom configuration file that add a #DEFINE and #UNDEFINE to the Calibre Interactive control file.

See "Configuration File Format for xACT GUI Customization" on page 410 for information on the file format and how to invoke Calibre Interactive xACT with a custom configuration file.

| Command | Description |
|---|---|
| addChoicesDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with a value to the Calibre Interactive xACT GUI; the value is taken from an enumerated list. The statement is added to the control file. |
| addDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with no value to the Calibre Interactive xACT GUI. The statement is added to the control file. |
| addInputDirTextDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to a directory used for input. The statement is added to the control file. The control includes a text input field and a browser button for selecting the directory. |
| addInputFileTextDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to an input file. The statement is added to the control file. The control includes a text entry field, a file browser button for selecting the file, and a view file button. |
| addIntegerDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with an integer value to the Calibre Interactive xACT GUI. The statement is added to the control file. |
| addOutputDirTextDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to a directory used for output. The statement is added to the control file. The control includes a text input field and a browser button for selecting the directory. |
| addOutputFileTextDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to an output file. The statement is added to the control file. The control includes a text input field, a file browser button for selecting the file, and a view file button. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Command | Description |
|---|---|
| addRealDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with a floating-point value to the Calibre Interactive xACT GUI. The statement is added to the control file. |
| addTextBoxDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI. The GUI control is a multi-line text box. The statement is added to the control file. |
| addTextDefine | Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI. The statement is added to the control file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addChoicesDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with a value to the Calibre Interactive xACT GUI; the value is taken from an enumerated list. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addChoicesDefine("*name*", *value_list***[, *label_list*[, *tooltip_list*]]**)**

### Object Properties

*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*control_label*"
*ctrl_var*.description = "*control_tooltip*"
*ctrl_var*.value = *value*
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false
*ctrl_var*.choiceAt(*i*).label = "*choice_label*"
*ctrl_var*.choiceAt(*i*).value = "*choice_value*"
*ctrl_var*.choiceAt(*i*).description = "*choice_tooltip*"
*ctrl_var*.choiceAt(*i*).compatible = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The variable name for the define statement written to the Calibre Interactive control file.

- ***value_list***

  A required list of possible values for the define statement.

  For example: [1, 2, 3] or [on, off]

  Values are quoted by default in the statement written to the control file—set the valueQuoted property to false for integer or real values.

- *label_list*

  An optional list of labels for the control's dropdown list.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For example: [“one”, “two”, “three”]

If *label_list* is not included in the function call, the labels can be defined with the choiceAt(*i*).label property.

- *tooltip_list*

  An optional list of tooltips for the control’s dropdown list.

  For example: [“first choice”, “second choice”, “third choice”]

  If *tooltip_list* is not included in the function call, the tooltips can be defined with the choiceAt(*i*).description property.

- Object Properties

  The description and label properties for the choices list can be defined in the function call or with a separate property definition using choiceAt(*i*).

  See “Option Properties” on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addChoicesDefine are the following:

**Table 9-15. Object Properties for addChoicesDefine**

| Property | Allowed Values | Description |
|---|---|---|
| value | *value*<br>(string or number) | The assigned value for the #DEFINE statement. Set this property to one of the choice values to provide an initial value for the control. |
| valueQuoted | true \| false<br>(boolean) | The assigned value is enclosed in quotes by default. Set valueQuoted to false to not include the quotes, as for an integer or real value. |
| tvf | true \| false<br>(Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| label | *control_label*<br>(string) | The label displayed for the control.<br>If not specified, the default label is “Define ***name***”. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-15. Object Properties for addChoicesDefine (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| choiceAt(*i*).label | *choice_label*<br>(string) | The label for the $i^{th}$ choice in the dropdown list of the GUI control. Numbering starts at zero |
| choiceAt(*i*).value | *choice_value*<br>(string, integer, or real) | The assigned value for the $i^{th}$ choice. |
| choiceAt(*i*).description | *choice_tooltip*<br>(string) | The tooltip for the $i^{th}$ choice in the dropdown list of the GUI control. |
| choiceAt(*i*).compatible | true \| false<br>(Boolean) | Specifies whether the $i^{th}$ choice in the dropdown list of the GUI control is available (the default) or unavailable (dimmed). |
| optional | true \| false<br>(Boolean) | Places a checkbox next to the control. The default is false, which creates a required control. |
| specified | true \| false<br>(Boolean) | Sets the state of the checkbox. |
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |

## Description

The function tool.addChoicesDefine adds a define control with an assigned value to the GUI, where the assigned value is taken from a dropdown list of choices. The control causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file unless insertToControlFile is set to false.

When combined with appropriate logic, the choiceAt(*i*).compatible property can be used to set a choice as available or unavailable depending on other settings in the GUI.

The following statement is written to the control file if the control is required or if the optional checkbox is checked, where *value* is the selected choice from the dropdown list:

```
#DEFINE name "value"
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" "value"
 tvf::set_global_variable "name" "value"
```

The value is quoted by default; set valueQuoted to false to not include quotes for both the TVF and non-TVF case.

If the control is optional and the checkbox is unchecked the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
unset -nocomplain "name"
tvf::unset_global_variable "name"
```

## Examples

This example adds an optional define control with two possible choices. The assigned value is a number, so valueQuoted is set to false. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized.

The tool property “label” is not defined, so the default label of “Define ***name***” is used, where ***name*** is the variable name for the define statement, PROCESS_DEFINE in this case.

```
tool.addPage("Defines and Variables")
var d = tool.addChoicesDefine("PROCESS_DEFINE",
                              [22, 16],
                              ["22 nm", "16 nm"],
                              ["Set to 22", "Set to 16"])

d.pages       =  ["Defines and Variables"]
d.description = "Define the process"
d.optional    = true;
d.specified   = true;

// set initial value to second choice, and do not include quotes
d.value = d.choiceAt(1).value
d.valueQuoted = false;
```

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The following statement is added to the Calibre Interactive control file:

```
#DEFINE PROCESS_DEFINE 16
```

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with no value to the Calibre Interactive xACT GUI. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var* = tool.addDefine("*name*")**

### Object Properties

*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.value = true | false
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The variable name for the define statement that is written to the Calibre Interactive control file.

- Object Properties

  See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, and visible.

  See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addDefine are the following:

**Table 9-16. Object Properties for addDefine**

| Property | Allowed Values | Description |
|---|---|---|
| value | true \| false<br>(Boolean) | Specifies the state of the checkbox for the addDefine control; the default is false. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-16. Object Properties for addDefine (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| tvf | true \| false<br>(Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |

**Note**

addDefine automatically adds a checkbox for the control. The “optional” property, which specifies a checkbox, is not valid for addDefine and is ignored if used.

## Description

The function call tool.addDefine adds a define control with no value to the GUI, which causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file unless insertToControlFile is set to false. The control has only a checkbox with the specified label.

The following statement is written to the control file if the control if the checkbox is checked:

```
#DEFINE name
```

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" 1
 tvf::set_global_variable "name" 1
```

If the checkbox is unchecked the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
set "name" 0
 tvf::set_global_variable "name" 0
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Examples

The following code places these controls on the custom page named “Defines and Variables”. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized.

```
tool.pages = ["Rules", "Inputs", "Outputs", "Options", "Environment", "Run
Control", "Defines and Variables"]

// add an optional define for the debug run, initial value off
var debugDef = tool.addDefine("DEBUG")
debugDef.pages = ["Defines and Variables"]
debugDef.label = "Define DEBUG"
debugDef.description = "include #DEFINE for DEBUG"

// add an optional define for the verbose run, initial value on
var verboseDef = tool.addDefine("VERBOSE")
verboseDef.value = true
verboseDef.pages = ["Defines and Variables"]
verboseDef.label = "Define VERBOSE"
verboseDef.description = "include #DEFINE for VERBOSE"

// add required define for CORNER
var d = tool.addDefine("CORNER")
d.pages = ["Defines and Variables"]
d.label = "CORNER define"
d.description = "required #DEFINE for CORNER"
d.value = true
d.editable = false
```

The following lines are added to the Calibre Interactive control file:

```
#UNDEFINE DEBUG
#DEFINE VERBOSE
#DEFINE CORNER
```

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addInputDirTextDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to a directory used for input. The statement is added to the control file. The control includes a text input field and a browser button for selecting the directory.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addInputDirTextDefine("*name*"**[, "*dirName*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*dirName*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The variable name for the define statement that is written to the Calibre Interactive control file.

- *"dirName"*

  An optional string specifying the assigned text string for the define statement. If "*dirName*" is not included in the function call it can be defined with the value property.

- *"tooltip"*

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Object Properties

  The value (*dirName*) and description (“*tooltip*”) properties can be defined in the function call or with a separate property definition.

  See “Option Properties” on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addInputDirTextDefine are the following:

**Table 9-17. Object Properties for addInputDirTextDefine**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*dirName*" (string) | The assigned value for the define statement. |
| valueQuoted | true \| false (boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (boolean) | Allow spaces in strings; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (boolean) | Sets the state of the checkbox when optional = true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Description

The function call tool.addInputDirTextDefine adds a define control with a string value to the GUI, which causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file. The string value *dirName* is expected to correspond to a directory used for input. The control includes a text entry box and browser button. An error is displayed next to the control if the *dirName* is not readable or does not exist.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
#DEFINE name "dirName"
```

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" "dirName"
 tvf::set_global_variable "name" "dirName"
```

The value is quoted by default; set valueQuoted to false to not include quotes for both the TVF and non-TVF case.

If the control is optional and the checkbox is unchecked (optional = true and specified = false) the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
unset -nocomplain "name"
 tvf::unset_global_variable "name"
```

## Examples

See the example with addInputFileTextDefine, which has similar syntax.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addInputFileTextDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to an input file. The statement is added to the control file. The control includes a text entry field, a file browser button for selecting the file, and a view file button.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addInputFileTextDefine(**"***name***"[, "*fileName*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*fileName*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The variable name for the define statement that is written to the Calibre Interactive control file.

- "*fileName*"

  An optional string specifying the assigned text string for the define statement. If "*fileName*" is not included in the function call it can be defined with the value property.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Object Properties

  The value (*fileName*) and description (“*tooltip*”) properties can be defined in the function call or with a separate property definition.

  See “Option Properties” on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addInputFileTextDefine are the following:

**Table 9-18. Object Properties for addInputFileTextDefine**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*fileName*" (string) | The assigned value for the define statement. |
| valueQuoted | true \| false (boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (boolean) | Allow spaces in strings; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (boolean) | Sets the state of the checkbox when optional = true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Description

The function call tool.addInputFileTextDefine adds a define control with a string value to the GUI, which causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file. The string value *fileName* is expected to correspond to an input file. The control includes a text entry box, a file browser button, and a view file button. An error is displayed next to the control if the *fileName* is not readable or is not a file.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
#DEFINE name "fileName"
```

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" "fileName"
 tvf::set_global_variable "name" "fileName"
```

The value is quoted by default; set valueQuoted to false to not include quotes for both the TVF and non-TVF case.

If the control is optional and the checkbox is unchecked (optional = true and specified = false) the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
unset -nocomplain "name"
 tvf::unset_global_variable "name"
```

## Examples

The following code adds an optional control to provide a file name for the INPUT_FILE define statement. The label property is not specified, so a default label is used. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized.

```
// #define for INPUT_FILE
var tdf = tool.addInputFileTextDefine("INPUT_FILE");
tdf.optional = true;
tdf.specified = false;
tdf.description = "optional input file";
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The following control is added to the Custom page:

The default control adds the following statement to the Calibre Interactive control file:

```
#UNDEFINE INPUT_FILE
```

If the checkbox is clicked and a filename is provided, the following statement is added to the Calibre Interactive control file:

```
#DEFINE INPUT_FILE "my_input_file"
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addIntegerDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with an integer value to the Calibre Interactive xACT GUI. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addIntegerDefine("*name*"**[, *intValue*[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = *intValue*
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.minValue = *min*
*ctrl_var*.maxValue = *max*
*ctrl_var*.singleStep = *step*
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The variable name for the define statement that is written to the Calibre Interactive control file.

- *intValue*

  An optional integer value specifying the assigned value for the define statement. If "*intValue*" is not included in the function call it can be defined with the value property.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.

- Object Properties

  The value (*intValue*) and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

The properties specific to addIntegerDefine are the following:

**Table 9-19. Object Properties for addIntegerDefine**

| Property | Value (type) | Description |
|---|---|---|
| value | *intValue*<br>(integer) | The assigned value for the define statement. |
| minValue | *min*<br>(integer) | The minimum value allowed for the value property.<br>An error notification ( ) is displayed next to the control if the condition is not satisfied. |
| maxValue | *max*<br>(integer) | The maximum value allowed for the value property.<br>An error notification ( ) is displayed next to the control if the condition is not satisfied. |
| singleStep | *step*<br>(integer) | Specify a *step* value in order to use a spin control in the GUI to set the value property. The properties minValue and maxValue must also be specified.<br><br>When the spin control up or down arrow is pressed, the current value is incremented or decremented by *step*. It is still possible to enter a value directly into the entry field of a spin control, as long as *min* <= *intValue* <= *max*. |
| optional | true \| <u>false</u><br>(Boolean) | Set to true to add a checkbox for the control. The default is false, which creates a required control. |
| specified | <u>true</u> \| false<br>(Boolean) | Sets the state of the checkbox when optional = true. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-19. Object Properties for addIntegerDefine (cont.)**

| Property | Value (type) | Description |
|---|---|---|
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |

## Description

The function call tool.addIntegerDefine adds a define control with an integer value to the GUI, which causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file. The control includes a text entry box.

The properties minValue and maxValue specify minimum and maximum allowed values for *intValue* and may be specified with or without the singleStep property. Without singleStep, an error notification (!) is displayed next to the control if the condition is not satisfied. A spin control is included if singleStep is defined. The minValue and maxValue properties are required if singleStep is specified, and place limits on the spin control.

The following statement is written to the control file if the control is required or if the optional checkbox is checked, where intValue is replaced with the actual value:

```
#DEFINE name intValue
```

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" intValue
 tvf::set_global_variable "name" intValue
```

If the control is optional and the checkbox is unchecked the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
unset -nocomplain "name"
 tvf::unset_global_variable "name"
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Examples

This example code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized.

```
var d = tool.addIntegerDefine("INTEGER_DEFINE", 20)
d.pages       = ["Defines and Variables"]
d.label       = "Set INTEGER_DEFINE"
d.description = "tooltip definition"
d.minValue    = -100;
d.maxValue    = 200;
d.singleStep  = 10;
d.optional    = true
d.specified   = true
```

The following line is written to the control file:

```
#DEFINE INTEGER_DEFINE 20
```

## Related Topics

Configuration File Format for xACT GUI Customization

addRealDefine



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addOutputDirTextDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to a directory used for output. The statement is added to the control file. The control includes a text input field and a browser button for selecting the directory.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addOutputDirTextDefine("*name*"**[, "*dirName*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*dirName*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The variable name for the define statement that is written to the Calibre Interactive control file.

- *"dirName"*

  An optional string specifying the assigned text string for the define statement. If "*dirName*" is not included in the function call it can be defined with the value property.

- *"tooltip"*

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Object Properties

  The value (*dirName*) and description (“*tooltip*”) properties can be defined in the function call or with a separate property definition.

  See “Option Properties” on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addOutputDirTextDefine are the following:

**Table 9-20. Object Properties for addOutputDirTextDefine**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*dirName*" (string) | The assigned value for the define statement. |
| valueQuoted | true \| false (boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (boolean) | Allow spaces in strings; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (boolean) | Sets the state of the checkbox when optional = true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Description

The function call tool.addOutputDirTextDefine adds a define control with a string value to the GUI, which causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file. The string value *dirName* is expected to correspond to a directory used for output. The control includes a text entry box and browser button. An error is displayed next to the control if the *dirName* is not writable.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
#DEFINE name "dirName"
```

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" "dirName"
 tvf::set_global_variable "name" "dirName"
```

The value is quoted by default; set valueQuoted to false to not include quotes for both the TVF and non-TVF case.

If the control is optional and the checkbox is unchecked (optional = true and specified = false) the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
unset -nocomplain "name"
 tvf::unset_global_variable "name"
```

## Examples

See the example with addInputFileTextDefine, which has similar syntax.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addOutputFileTextDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to an output file. The statement is added to the control file. The control includes a text input field, a file browser button for selecting the file, and a view file button.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addOutputFileTextDefine("*name*"**[, "*fileName*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*fileName*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The variable name for the define statement that is written to the Calibre Interactive control file.

- *"fileName"*

  An optional string specifying the assigned text string for the define statement. If "*fileName*" is not included in the function call it can be defined with the value property.

- *"tooltip"*

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Object Properties

  The value (*fileName*) and description (“*tooltip*”) properties can be defined in the function call or with a separate property definition.

  See “Option Properties” on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addOutputFileTextDefine are the following:

**Table 9-21. Object Properties for addOutputFileTextDefine**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*fileName*" (string) | The assigned value for the define statement. |
| valueQuoted | true \| false (boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (boolean) | Allow spaces in strings; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (boolean) | Sets the state of the checkbox when optional = true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Description

The function call tool.addOutputFileTextDefine adds a define control with a string value to the GUI, which causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file. The string value *fileName* is expected to correspond to an output file. The control includes a text entry box, and file browser button, and a view file button. An error is displayed next to the control if the *fileName* is not writable.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
#DEFINE name "fileName"
```

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" "fileName"
 tvf::set_global_variable "name" "fileName"
```

The value is quoted by default; set valueQuoted to false to not include quotes for both the TVF and non-TVF case.

If the control is optional and the checkbox is unchecked (optional = true and specified = false) the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
unset -nocomplain "name"
 tvf::unset_global_variable "name"
```

## Examples

See the example with addInputFileTextDefine, which has similar syntax.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addRealDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with a floating-point value to the Calibre Interactive xACT GUI. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addRealDefine("*name*"**[, *realValue*[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = *realValue*
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.minValue = *min*
*ctrl_var*.maxValue = *max*
*ctrl_var*.singleStep = *step*
*ctrl_var*.includeMinEndpoint = true | false
*ctrl_var*.includeMaxEndpoint = true | false
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The variable name for the define statement that is written to the Calibre Interactive control file.

- *realValue*

  An optional floating-point value specifying the assigned value for the define statement. If "*realValue*" is not included in the function call it can be defined with the value property.

- *tooltip*

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Object Properties

  The value (*realValue*) and description (“*tooltip*”) properties can be defined in the function call or with a separate property definition.

  See “Option Properties” on page 421 for a description of these common properties: pages, label, description, value, editable, visible, and specified.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addRealDefine are the following:

**Table 9-22. Object Properties for addRealDefine**

| Property | Value (type) | Description |
|---|---|---|
| value | *realValue*<br>(floating-point) | The assigned value for the define statement. |
| minValue | *min*<br>(floating-point) | The minimum value allowed for the value property.<br>An error notification ( ) is displayed next to the control if the condition is not satisfied. |
| maxValue | *max*<br>(floating-point) | The maximum value allowed for the value property.<br>An error notification ( ) is displayed next to the control if the condition is not satisfied. |
| singleStep | *step*<br>(floating-point) | Specify a *step* value in order to use a spin control in the GUI to set the value property. The properties minValue and maxValue must also be specified.<br>When the spin control up or down arrow is pressed, the current value is incremented or decremented by *step*. It is still possible to enter a value directly into the entry field of a spin control, as long as *realValue* satisfies the conditions dictated by minValue, maxValue, includeMinEndpoint, and includeMaxEndpoint. |
| includeMinEndpoint | true \| false<br>(Boolean) | Specifies whether to include minValue in the range. |
| includeMaxEndpoint | true \| false<br>(Boolean) | Specifies whether to include maxValue in the range |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-22. Object Properties for addRealDefine (cont.)**

| Property | Value (type) | Description |
|---|---|---|
| optional | true \| false (Boolean) | Set to true to add a checkbox for the control. The default is false, which creates a required control. |
| specified | true \| false (Boolean) | Sets the state of the checkbox when optional = true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |

## Description

The function call tool.addRealDefine adds a define control with a floating-point value to the GUI, which causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file.

If the tvf property is set to true, then a Tcl variable is inserted in the Calibre Interactive control file, rather than a #DEFINE statement.

The properties minValue, maxValue, includeMinEndpoint, and includeMaxEndpoint specify minimum and maximum allowed values for *realValue* and may be specified with or without the singleStep property. Without singleStep, an error notification ( ) is displayed next to the control if the condition is not satisfied. A spin control is included if singleStep is defined. The minValue and maxValue properties are required if singleStep is specified, and place limits on the spin control.

The following statement is written to the control file if the control is required or if the optional checkbox is checked, where realValue is replaced with the actual value:

```
#DEFINE name realValue
```

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" realValue
 tvf::set_global_variable "name" realValue
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If the control is optional and the checkbox is unchecked the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
unset -nocomplain "name"
 tvf::unset_global_variable "name"
```

## Related Topics

Configuration File Format for xACT GUI Customization

addIntegerDefine



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addTextBoxDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI. The GUI control is a multi-line text box. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addTextBoxDefine("*name*"**[, "*textValue*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*textValue*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.emptyIsValid = true | false
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The variable name for the define statement that is written to the Calibre Interactive control file.

- "*textValue*"

  An optional string specifying the assigned text string for the define statement. If "*textValue*" is not included in the function call it can be defined with the value property.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Object Properties

  The value (*textValue*) and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.

  See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, and visible.

  See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addTextDefine are the following:

**Table 9-23. Object Properties for addTextDefine**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*textValue*" (string) | The assigned value for the define statement. |
| valueQuoted | true \| false (Boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (Boolean) | Allow spaces in strings; the default is false. |
| emptyIsValid | true \| false (Boolean) | Allow an empty *textValue* string; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (Boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (Boolean) | Sets the state of the checkbox when optional = true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-23. Object Properties for addTextDefine (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |

## Description

The function call tool.addTextBoxDefine adds a define control with a string value to the GUI, which causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file. The control includes a multi-line text entry box, in contrast to addTextDefine, which has a single line text entry box.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
#DEFINE name "textValue"
```

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" "textValue"
 tvf::set_global_variable "name" "textValue"
```

The value is quoted by default; set valueQuoted to false to not include quotes for both the TVF and non-TVF case.

If the control is optional and the checkbox is unchecked (optional = true and specified = false) the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
unset -nocomplain "name"
tvf::unset_global_variable "name"
```

## Examples

The following code adds a required control to provide a string value for the CORNERS_DEF variable. The initial value is “typical”. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized.

```
// #define for CORNERS_DEF
var d = tool.addTextDefine("CORNERS_DEF",
                           "typical hi lo testA testB processA processB",
                           "space separated list of corners")
d.pages       = ["Defines and Variables"]
d.allowSpaces = true
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The following statement is added to the Calibre Interactive control file:

```
#DEFINE CORNERS_DEF "typical hi lo testA testB processA processB"
```

The control looks like this in the GUI:

## Related Topics

Configuration File Format for xACT GUI Customization

addTextDefine



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addTextDefine

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a #DEFINE or #UNDEFINE statement with a string value to the Calibre Interactive xACT GUI. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addTextDefine("*name*"**[, "*textValue*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*textValue*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.emptyIsValid = true | false
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The variable name for the define statement that is written to the Calibre Interactive control file.

- *"textValue"*

  An optional string specifying the assigned text string for the define statement. If "*textValue*" is not included in the function call it can be defined with the value property.

- *"tooltip"*

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.

- Object Properties

  The value (*textValue*) and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, and visible.

See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

The properties specific to addTextDefine are the following:

**Table 9-24. Object Properties for addTextDefine**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*textValue*" (string) | The assigned value for the define statement. |
| valueQuoted | true \| false (Boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (Boolean) | Allow spaces in strings; the default is false. |
| emptyIsValid | true \| false (Boolean) | Allow an empty *textValue* string; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (Boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (Boolean) | Sets the state of the checkbox when optional = true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the #DEFINE or #UNDEFINE statement to the control file. The default is true; modify this property with caution. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Description

The function call tool.addTextDefine adds a define control with a string value to the GUI, which causes a #DEFINE or #UNDEFINE statement to be written to the Calibre Interactive control file. The control includes a text entry box.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
#DEFINE name "textValue"
```

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" "textValue"
 tvf::set_global_variable "name" "textValue"
```

The value is quoted by default; set valueQuoted to false to not include quotes for both the TVF and non-TVF case.

If the control is optional and the checkbox is unchecked (optional = true and specified = false) the following statement is written to the control file:

```
#UNDEFINE name
```

For the corresponding TVF case, the following statements are written to the control file:

```
unset -nocomplain "name"
tvf::unset_global_variable "name"
```

## Examples

The following code adds a required control to provide a string value for the CORNERS_DEF variable. The initial value is “typical”. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized.

```
// #define for CORNERS_DEF
var d = tool.addTextDefine("CORNERS_DEF", "typical")
d.pages       = ["Defines and Variables"]
d.label       = "Define CORNERS_DEF for corner settings"
d.description = "Enter a space separated list of corners"
d.allowSpaces = true
```

The following statement is added to the Calibre Interactive control file:

```
#DEFINE CORNERS_DEF "typical"
```

## Related Topics

Configuration File Format for xACT GUI Customization

addTextBoxDefine



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Configuration File Functions to Add Variable Statements

Several functions are available for the Calibre Interactive xACT custom configuration file that add a Variable statement to the Calibre Interactive control file.

See "Configuration File Format for xACT GUI Customization" on page 410 for information on the file format and how to invoke Calibre Interactive xACT with a custom configuration file.

| Command | Description |
|---|---|
| addChoicesVariable | Adds a control to specify a Variable statement with a value to the Calibre Interactive xACT GUI; the value is taken from an enumerated list. The statement is added to the control file. |
| addListVariable | Adds a control to specify a Variable statement to the Calibre Interactive xACT GUI, where the assigned value for the variable is a list. The statement is added to the control file. |
| addInputDirTextVariable | Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to a directory used for input. The Variable statement is added to the control file. The control includes a text entry field and a browser button for selecting the directory. |
| addInputFileTextVariable | Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to an input file. The Variable statement is added to the control file. The control includes a text entry field, a file browser button for selecting the file, and a view file button. |
| addIntegerVariable | Adds a control to specify a Variable statement with an integer value to the Calibre Interactive xACT GUI. The statement is added to the control file. |
| addMultiChoicesVariable | Adds a control to specify a Variable statement with a value to the Calibre Interactive xACT GUI. The value is taken from an enumerated list and multiple values can be selected. The statement is added to the control file. |
| addOutputDirTextVariable | Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to a directory used for output. The Variable statement is added to the control file. The control includes a text entry field and a browser button for selecting the directory. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Command | Description |
|---|---|
| addOutputFileTextVariable | Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to an output file. The Variable statement is added to the control file. The control includes a text entry field, a file browser button for selecting the file, and a view file button. |
| addRealVariable | Adds a control to specify a Variable statement with a real value to the Calibre Interactive xACT GUI. The statement is added to the control file. |
| addTextBoxVariable | Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI. The GUI control is a multi-line text box. The statement is added to the control file. |
| addTextVariable | Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI. The statement is added to the control file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addChoicesVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with a value to the Calibre Interactive xACT GUI; the value is taken from an enumerated list. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addChoicesVariable("*name*", *value_list***[, *label_list*[, *tooltip_list*]]**)**

### Object Properties

*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*control_label*"
*ctrl_var*.description = "*control_tooltip*"
*ctrl_var*.value = *value*
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false
*ctrl_var*.choiceAt(*i*).label = "*choice_label*"
*ctrl_var*.choiceAt(*i*).value = "*choice_value*"
*ctrl_var*.choiceAt(*i*).description = "*choice_tooltip*"
*ctrl_var*.choiceAt(*i*).compatible = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The name of the variable for the Variable statement written to the Calibre Interactive control file.

- ***value_list***

  A required list of possible values for the Variable statement. For example: [1, 2, 3]

  Values are quoted by default in the statement written to the control file—set the valueQuoted property to false for integer or real values.

- *label_list*

  An optional list of labels for the control's dropdown list.

  For example: ["one", "two", "three"]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If *label_list* is not included in the function call, the labels can be defined with the choiceAt(*i*).label property.

- *tooltip_list*

  An optional list of tooltips (hover text) for the control's dropdown list.

  For example: ["first choice", "second choice", "third choice"]

  If *tooltip_list* is not included in the function call, the tooltips can be defined with the choiceAt(*i*).description property.

- Object Properties

  The label (*label_list*) and description (*tooltip_list*) properties for the choices list can be defined in the function call or with a separate property definition using choiceAt(*i*).

  See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addChoicesVariable are the following:

**Table 9-25. Object Properties for addChoicesVariable**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*value*"<br>(string or number) | The assigned value for the Variable statement. Set this property to one of the choice values to provide an initial value for the control. |
| optional | true \| <u>false</u><br>(boolean) | Places a checkbox next to the control. The default is false, which creates a required control. |
| specified | <u>true</u> \| false<br>(boolean) | Sets the state of the checkbox when optional=true. |
| valueQuoted | <u>true</u> \| false<br>(boolean) | The assigned value is enclosed in quotes by default. Set valueQuoted to false to not include the quotes, as for an integer or real value. |
| tvf | true \| <u>false</u><br>(Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-25. Object Properties for addChoicesVariable (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |
| choiceAt(*i*).label | *choice_label*<br>(string) | The label for the $i^{th}$ choice in the dropdown list of the GUI control. Numbering starts at zero. |
| choiceAt(*i*).value | *choice_value*<br>(string, integer, or real) | The assigned value for the $i^{th}$ choice. |
| choiceAt(*i*).description | *choice_tooltip*<br>(string) | The tooltip for the $i^{th}$ choice in the dropdown list of the GUI control. |
| choiceAt(*i*).compatible | true \| false<br>(Boolean) | Specifies whether the $i^{th}$ choice in the dropdown list of the GUI control is available (the default) or unavailable (dimmed). |

## Description

The function call tool.addChoicesVariable adds a variable control with a dropdown list of choices to the GUI. The control causes a SVRF Variable statement to be written to the Calibre Interactive control file.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true), where “value” is one of the available choices:

```
VARIABLE name "value"
```

The value is quoted by default—set valueQuoted to false to not use quotes.

When combined with appropriate logic, the choiceAt(*i*).compatible property can be used to set a choice as available or unavailable depending on other settings in the GUI.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" {value}
 tvf::set_global_variable "name" {value}
```

The property valueQuoted is not used when the tvf property is set to true.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Examples

This example adds an optional control to include a Variable statement to set VAR_TEMP, where there are two possible choices for the assigned value. The property valueQuoted is set to false in order to define a numeric value. The label property is not defined so the default label of "Variable VAR_TEMP" is used for the control. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized.

```
var cv = tool.addChoicesVariable("VAR_TEMP",
                     [25, 27],
                     ["25", "27"],
                     ["25 degrees", "27 degrees"])
cv.pages       = ["Defines and Variables"]
cv.description = "Defines SVRF Variable VAR_TEMP for the temperature"
cv.optional    = true;
cv.specified   = true;
// necessary to get numeric value rather than a string
cv.valueQuoted = false;
// set initial value to one of the choices
cv.value = cv.choiceAt(1).value
```

The following statement is added to the Calibre Interactive control file:

```
VARIABLE VAR_TEMP 27
```

## Related Topics

Configuration File Format for xACT GUI Customization

addMultiChoicesVariable



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addListVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement to the Calibre Interactive xACT GUI, where the assigned value for the variable is a list. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addListVariable("*name*"**[, *value_list*[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = *value_list*
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The name of the variable for the Variable statement added to the Calibre Interactive control file.

- *value_list*

  An optional list of items specifying the assigned value for the Variable statement. For example: [“buf”, “inv”, “or”]

  If *value_list* is not included in the function call it can be defined with the value property.

- “*tooltip*”

  An optional string specifying the hover text (tooltip) for the control. If “*tooltip*” is not included in the function call it can be defined with the description property.

- Object Properties

  The value (*value_list*) and description (“*tooltip*”) properties can be defined in the function call or with a separate property definition.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

The properties specific to addListVariable are the following:

**Table 9-26. Object Properties for addListVariable**

| Property | Allowed Values | Description |
|---|---|---|
| value | *value_list*<br>(string or number list) | The assigned value for the Variable statement.<br>For example: ["buf", "inv", "or"] |
| valueQuoted | true \| false<br>(boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| promptText | "*prompt_string*"<br>(string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false<br>(boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false<br>(boolean) | Sets the state of the checkbox when optional=true. |
| tvf | true \| false<br>(Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |

## Description

The function call tool.addListVariable adds a variable control with a list value to the GUI, which causes a SVRF Variable statement to be written to the Calibre Interactive control file. The control includes a text entry box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If the value property is set to [item1, item2], the following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
VARIABLE name "item1" "item2"
```

The list items are enclosed in quotes by default; set valueQuoted to false to not use the quotes. Spaces are not allowed within strings in the Variable statement, therefore a string with a space is considered as two separate strings; for example, “aa bb” is considered as “aa” “bb”.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" {item1 item2}
 tvf::set_global_variable "name" {item1 item2}
```

The property valueQuoted is not used when the tvf property is set to true.

## Examples

This example defines a variable with a list of cell names. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized. The list value is included in the function call, while the tooltip is specified as a property definition. The label property is not specified and defaults to “Variable VAR_CELL_LIST”. The pages property is not specified, so the control appears on the default “Custom” page.

```
// VARIABLE VAR_CELL_LIST for list of strings for cell names
var v = tool.addListVariable("VAR_CELL_LIST",
                     ["inv", "buf", "cella cellb"])
v.description = "Adds SVRF Variable statement for VAR_CELL_LIST"
```

The following control is created:

This statement is added to the control file for Calibre Interactive:

```
VARIABLE VAR_CELL_LIST "inv" "buf" "cella" "cellb"
```

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addInputDirTextVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to a directory used for input. The Variable statement is added to the control file. The control includes a text entry field and a browser button for selecting the directory.

**Note**

Variables defined as a directories may be used in the tvf::svrf_var command, however, such variables cannot be used to specify directories for SVRF operations and statements. See the Variable statement.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addInputDirTextVariable("*name*"**[, "*dirName*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*dirName*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The name of the variable for the Variable statement written to the Calibre Interactive control file.

- "*dirName*"

  An optional string specifying the assigned text string for the Variable statement. If "*dirName*" is not included in the function call it can be defined with the value property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.

- Object Properties

  The value ("*dirName*") and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.

  See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addInputDirTextVariable are the following:

**Table 9-27. Object Properties for addInputDirTextVariable**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*dirName*" (string) | The assigned value for the Variable statement. |
| valueQuoted | true \| false (boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (boolean) | Allow spaces in strings; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (boolean) | Sets the state of the checkbox when optional=true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-27. Object Properties for addInputDirTextVariable (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |

## Description

The function call tool.addInputDirTextVariable adds a variable control with a string value to the GUI, which causes a Variable statement to be written to the Calibre Interactive control file. The string value *dirName* is expected to correspond to a directory used for input. The control includes a text entry box and a browser button. An error is displayed next to the control if the *dirName* is not readable or does not exist.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
VARIABLE name "dirName"
```

The value is quoted by default; set valueQuoted to false to not include quotes. Quotes are generally recommended.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" {dirName}
 tvf::set_global_variable "name" {dirName}
```

The property valueQuoted is not used when the tvf property is set to true.

## Examples

See the example with addInputFileTextVariable, which has similar syntax.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addInputFileTextVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to an input file. The Variable statement is added to the control file. The control includes a text entry field, a file browser button for selecting the file, and a view file button.

**Note**

Variables defined as a filenames may be used in the tvf::svrf_var command, however, such variables cannot be used to specify filenames for SVRF operations and statements. See the Variable statement.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addInputFileTextVariable("*name*"**[, "*fileName*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*fileName*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The name of the variable for the Variable statement written to the Calibre Interactive control file.

- "*fileName*"

  An optional string specifying the assigned text string for the Variable statement. If "*fileName*" is not included in the function call it can be defined with the value property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.

- Object Properties

  The value ("*fileName*") and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.

  See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addInputFileTextVariable are the following:

**Table 9-28. Object Properties for addInputFileTextVariable**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*fileName*" (string) | The assigned value for the Variable statement. |
| valueQuoted | true \| false (boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (boolean) | Allow spaces in strings; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (boolean) | Sets the state of the checkbox when optional=true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-28. Object Properties for addInputFileTextVariable (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |

## Description

The function call tool.addInputFileTextVariable adds a variable control with a string value to the GUI, which causes a Variable statement to be written to the Calibre Interactive control file. The string value *fileName* is expected to correspond to an input file. The control includes a text entry box, a file browser button, and a view file button. An error is displayed next to the control if the *fileName* is not readable or is not a file.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
VARIABLE name "fileName"
```

The value is quoted by default; set valueQuoted to false to not include quotes. Quotes are generally recommended.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" {fileName}
 tvf::set_global_variable "name" {fileName}
```

The property valueQuoted is not used when the tvf property is set to true.

## Examples

The following code adds a control to specify a Variable statement with a string value corresponding to a file name. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized. The variable name, initial setting for the file name, and tooltip are specified in the function call.

```
var var_f = tool.addInputFileTextVariable("VAR_INPUT_FILE",
                                          "test_input",
                                          "define Variable VAR_INPUT_FILE");
```

The following control is added to the Custom page:

The following statement is added to the control file for Calibre Interactive:

```
VARIABLE VAR_INPUT_FILE "test_input"
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addIntegerVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with an integer value to the Calibre Interactive xACT GUI. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addIntegerVariable("*name*"**[, *intValue*[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = *intValue*
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.minValue = *min*
*ctrl_var*.maxValue = *max*
*ctrl_var*.singleStep = *step*
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The name of the variable for the Variable statement written to the Calibre Interactive control file.

- *intValue*

  An optional integer value specifying the assigned value for the Variable statement. If "*intValue*" is not included in the function call it can be defined with the value property.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.

- Object Properties

  The value (*intValue*) and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

See "Option Properties" on page 421 for a description of these common properties: pages, label, description, value, editable, visible, and specified.

See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

The properties specific to addIntegerVariable are the following:

**Table 9-29. Object Properties for addIntegerVariable**

| Property | Value (type) | Description |
|---|---|---|
| value | *intValue*<br>(integer) | The assigned value for the Variable statement. |
| minValue | *min*<br>(integer) | The minimum value allowed for the value property.<br>An error notification ( ) is displayed next to the control if the condition is not satisfied. |
| maxValue | *max*<br>(integer) | The maximum value allowed for the value property.<br>An error notification ( ) is displayed next to the control if the condition is not satisfied. |
| singleStep | *step*<br>(integer) | Specify a *step* value in order to use a spin control in the GUI to set the value property. The properties minValue and maxValue must also be specified.<br>When the spin control up or down arrow is pressed, the current value is incremented or decremented by *step*. It is still possible to enter a value directly into the entry field of a spin control, as long as *min* <= *intValue* <= *max*. |
| optional | true \| <u>false</u><br>(Boolean) | Set to true to add a checkbox for the control. The default is false, which creates a required control. |
| specified | <u>true</u> \| false<br>(Boolean) | Sets the state of the checkbox when optional = true. |
| tvf | true \| <u>false</u><br>(Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-29. Object Properties for addIntegerVariable (cont.)**

| Property | Value (type) | Description |
|---|---|---|
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |

## Description

The function tool.addIntegerVariable adds a variable control with an integer value to the GUI, which causes a Variable statement to be written to the Calibre Interactive control file. The control includes a text entry box.

The properties minValue and maxValue specify minimum and maximum allowed values for *intValue* and may be specified with or without the singleStep property. Without singleStep, an error notification (!) is displayed next to the control if the condition is not satisfied. A spin control is included if singleStep is defined. The minValue and maxValue properties are required if singleStep is specified, and place limits on the spin control.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
VARIABLE name intValue
```

where *intValue* is replaced with the actual value.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" intValue
 tvf::set_global_variable "name" intValue
```

## Related Topics

Configuration File Format for xACT GUI Customization

addRealVariable



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addMultiChoicesVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with a value to the Calibre Interactive xACT GUI. The value is taken from an enumerated list and multiple values can be selected. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addMultiChoicesVariable("*name*", *value_list***[, *label_list*[, *tooltip_list*]]**)**

### Object Properties

*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*control_label*"
*ctrl_var*.description = "*control_tooltip*"
*ctrl_var*.value = *value*
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false
*ctrl_var*.choiceAt(*i*).label = "*choice_label*"
*ctrl_var*.choiceAt(*i*).value = "*choice_value*"
*ctrl_var*.choiceAt(*i*).description = "*choice_tooltip*"
*ctrl_var*.choiceAt(*i*).compatible = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The name of the variable for the Variable statement written to the Calibre Interactive control file.

- ***value_list***

  A required list of possible values for the Variable statement. For example: [1, 2, 3]

  Values are quoted by default in the statement written to the control file—set the valueQuoted property to false for integer or real values.

- *label_list*

  An optional list of labels for the control's dropdown list.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For example: [“one”, “two”, “three”]

If *label_list* is not included in the function call, the labels can be defined with the choiceAt(*i*).label property.

- *tooltip_list*

  An optional list of tooltips (hover text) for the control’s dropdown list.

  For example: [“first choice”, “second choice”, “third choice”]

  If *tooltip_list* is not included in the function call, the tooltips can be defined with the choiceAt(*i*).description property.

- Object Properties

  The label (*label_list*) and description (*tooltip_list*) properties for the choices list can be defined in the function call or with a separate property definition using choiceAt(*i*).

  See “Option Properties” on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addMultiChoicesVariable are the following:

**Table 9-30. Object Properties for addMultiChoicesVariable**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*value*"<br>(string or number) | The assigned value for the Variable statement. Set this property to one of the choice values to provide an initial value for the control. |
| optional | true \| <u>false</u><br>(boolean) | Places a checkbox next to the control. The default is false, which creates a required control. |
| specified | <u>true</u> \| false<br>(boolean) | Sets the state of the checkbox when optional=true. |
| valueQuoted | <u>true</u> \| false<br>(boolean) | The assigned value is enclosed in quotes by default. Set valueQuoted to false to not include the quotes, as for an integer or real value. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-30. Object Properties for addMultiChoicesVariable (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| tvf | true \| false<br>(Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |
| choiceAt(*i*).label | *choice_label*<br>(string) | The label for the $i^{th}$ choice in the dropdown list of the GUI control. Numbering starts at zero. |
| choiceAt(*i*).value | *choice_value*<br>(string, integer, or real) | The assigned value for the $i^{th}$ choice. |
| choiceAt(*i*).description | *choice_tooltip*<br>(string) | The tooltip for the $i^{th}$ choice in the dropdown list of the GUI control. |
| choiceAt(*i*).compatible | true \| false<br>(Boolean) | Specifies whether the $i^{th}$ choice in the dropdown list of the GUI control is available (the default) or unavailable (dimmed). |

## Description

The function call tool.addMultiChoicesVariable adds a variable control to the GUI. The control includes a dialog box in which one or more items from the ***value_list*** can be selected. The control causes a SVRF Variable statement to be written to the Calibre Interactive control file.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true) and two choice values are selected (value1 and “value2):

```
VARIABLE "name" "value1" "value2"
```

The values are quoted by default—set valueQuoted to false to not use quotes.

When combined with appropriate logic, the choiceAt(*i*).compatible property can be used to set a choice as available or unavailable depending on other settings in the GUI.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file when two choice values are selected:

```
set "name" {value1 value2}
 tvf::set_global_variable "name" {value1 value2}
```

The property valueQuoted is not used when the tvf property is set to true.

## Examples

This example adds an optional control to include a Variable statement to set VAR_NETS, where there are three possible choices for the assigned value. The label property is not defined so the default label of “Variable VAR_NETS” is used for the control. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized.

```
var cvm = tool.addMultiChoicesVariable("VAR_NETS",
                      ["a", "b", "rs1"],
                      ["a", "b", "rs1"],
                      ["net a", "net b", "net rs1"])
cvm.pages       = ["Defines and Variables"]
cvm.description = "Defines SVRF Variable VAR_NETS"
cvm.optional    = true;
cvm.specified   = true;
// set initial value to one of the choices
cvm.value = cvm.choiceAt(1).value
```

When the control is clicked, a dialog box opens in which you can select one or more of the values. The following statement is added to the Calibre Interactive control file if a and b are selected:

```
VARIABLE "VAR_NETS" "a" "b"
```

## Related Topics

Configuration File Format for xACT GUI Customization

addChoicesVariable



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addOutputDirTextVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to a directory used for output. The Variable statement is added to the control file. The control includes a text entry field and a browser button for selecting the directory.

**Note**

Variables defined as a directories may be used in the tvf::svrf_var command, however, such variables cannot be used to specify directories for SVRF operations and statements. See the Variable statement.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addOutputDirTextVariable("*name*"**[, "*dirName*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*dirName*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The name of the variable for the Variable statement written to the Calibre Interactive control file.

- "*dirName*"

  An optional string specifying the assigned text string for the Variable statement. If "*dirName*" is not included in the function call it can be defined with the value property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.

- Object Properties

  The value ("*dirName*") and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.

  See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addOutputDirTextVariable are the following:

**Table 9-31. Object Properties for addOutputDirTextVariable**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*dirName*" (string) | The assigned value for the Variable statement. |
| valueQuoted | true \| false (boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (boolean) | Allow spaces in strings; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (boolean) | Sets the state of the checkbox when optional=true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-31. Object Properties for addOutputDirTextVariable (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |

## Description

The function call tool.addOutputDirTextVariable adds a variable control with a string value to the GUI, which causes a Variable statement to be written to the Calibre Interactive control file. The string value *dirName* is expected to correspond to a directory used for output. The control includes a text entry box and a browser button. An error is displayed next to the control if the *dirName* is not writable.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
VARIABLE name "dirName"
```

The value is quoted by default; set valueQuoted to false to not include quotes. Quotes are generally recommended.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" {dirName}
 tvf::set_global_variable "name" {dirName}
```

The property valueQuoted is not used when the tvf property is set to true.

## Examples

See the example with addInputFileTextVariable, which has similar syntax.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addOutputFileTextVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI, where the string value is expected to correspond to an output file. The Variable statement is added to the control file. The control includes a text entry field, a file browser button for selecting the file, and a view file button.

**Note**

Variables defined as a filenames may be used in the tvf::svrf_var command, however, such variables cannot be used to specify filenames for SVRF operations and statements. See the Variable statement.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addOutputFileTextVariable("*name*"**[, "*fileName*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*fileName*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The name of the variable for the Variable statement written to the Calibre Interactive control file.

- "*fileName*"

  An optional string specifying the assigned text string for the Variable statement. If "*fileName*" is not included in the function call it can be defined with the value property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.

- Object Properties

  The value ("*fileName*") and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.

  See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, visible, and specified.

  See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addOutputFileTextVariable are the following:

**Table 9-32. Object Properties for addOutputFileTextVariable**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*fileName*" (string) | The assigned value for the Variable statement. |
| valueQuoted | true \| false (boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (boolean) | Allow spaces in strings; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (boolean) | Sets the state of the checkbox when optional=true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-32. Object Properties for addOutputFileTextVariable (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| insertToControlFile | true \| false<br>(Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |

## Description

The function call tool.addOutputFileTextVariable adds a variable control with a string value to the GUI, which causes a Variable statement to be written to the Calibre Interactive control file. The string value *fileName* is expected to correspond to an output file. The control includes a text entry box, a file browser button, and a view file button. An error is displayed next to the control if the *fileName* is not writable.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
VARIABLE name "fileName"
```

The value is quoted by default; set valueQuoted to false to not include quotes. Quotes are generally recommended.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" {fileName}
 tvf::set_global_variable "name" {fileName}
```

The property valueQuoted is not used when the tvf property is set to true.

## Examples

See the example with addInputFileTextVariable, which has similar syntax.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addRealVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with a real value to the Calibre Interactive xACT GUI. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addRealVariable("*name*"**[, *realValue*[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = *realValue*
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.minValue = *min*
*ctrl_var*.maxValue = *max*
*ctrl_var*.singleStep = *step*
*ctrl_var*.includeMinEndpoint = true | false
*ctrl_var*.includeMaxEndpoint = true | false
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***"name"***

  The name of the variable for the Variable statement that is written to the Calibre Interactive control file.

- *realValue*

  An optional real value specifying the assigned value for the Variable statement. If "*realValue*" is not included in the function call it can be defined with the value property.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Object Properties

  The value (*realValue*) and description (“*tooltip*”) properties can be defined in the function call or with a separate property definition.

  See “Option Properties” on page 421 for a description of these common properties: pages, label, description, value, editable, visible, and specified.

  See “Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT” on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

  The properties specific to addRealDefine are the following:

**Table 9-33. Object Properties for addRealVariable**

| Property | Value (type) | Description |
|---|---|---|
| value | *realValue*<br>(real) | The assigned value for the Variable statement. |
| minValue | *min*<br>(real) | The minimum value allowed for the value property.<br>An error notification ( ) is displayed next to the control if the condition is not satisfied. |
| maxValue | *max*<br>(real) | The maximum value allowed for the value property.<br>An error notification ( ) is displayed next to the control if the condition is not satisfied. |
| singleStep | *step*<br>(real) | Specify a *step* value in order to use a spin control in the GUI to set the value property. The properties minValue and maxValue must also be specified.<br>When the spin control up or down arrow is pressed, the current value is incremented or decremented by *step*. It is still possible to enter a value directly into the entry field of a spin control, as long as realValue satisfies the conditions dictated by minValue, maxValue, includeMinEndpoint, and includeMaxEndpoint. |
| includeMinEndpoint | true \| false<br>(Boolean) | Specifies whether to include minValue in the range. |
| includeMaxEndpoint | true \| false<br>(Boolean) | Specifies whether to include maxValue in the range |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-33. Object Properties for addRealVariable (cont.)**

| Property | Value (type) | Description |
|---|---|---|
| optional | true \| false (Boolean) | Set to true to add a checkbox for the control. The default is false, which creates a required control. |
| specified | true \| false (Boolean) | Sets the state of the checkbox when optional = true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |

## Description

The function call tool.addRealVariable adds a variable control with a real value to the GUI, which causes a Variable statement to be written to the Calibre Interactive control file. The control includes a text entry box.

The properties minValue, maxValue, includeMinEndpoint, and includeMaxEndpoint specify minimum and maximum allowed values for *realValue* and may be specified with or without the singleStep property. Without singleStep, an error notification ( ) is displayed next to the control if the condition is not satisfied. A spin control is included if singleStep is defined. The minValue and maxValue properties are required if singleStep is specified, and place limits on the spin control.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
VARIABLE name realValue
```

where *realValue* is replaced with the actual value.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" realValue
 tvf::set_global_variable "name" realValue
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Related Topics

Configuration File Format for xACT GUI Customization

addIntegerVariable



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addTextBoxVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI. The GUI control is a multi-line text box. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addTextBoxVariable("*name*"**[, "*textValue*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*textValue*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.emptyIsValid = true | false
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The name of the variable for the Variable statement written to the Calibre Interactive control file.

- "*textValue*"

  An optional string specifying the assigned text string for the Variable statement. If "*textValue*" is not included in the function call it can be defined with the value property.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.

- Object Properties

  The value ("*textValue*") and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, and visible.

See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

The properties specific to addTextVariable are the following:

**Table 9-34. Object Properties for addTextVariable**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*textValue*" (string) | The assigned value for the Variable statement. |
| valueQuoted | true \| false (Boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (Boolean) | Allow spaces in strings; the default is false. |
| emptyIsValid | true \| false (Boolean) | Allow an empty *textValue* string; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (Boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (Boolean) | Sets the state of the checkbox when optional=true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Description

The function call tool.addTextBoxVariable adds a variable control with a string value to the GUI, which causes a SVRF Variable statement to be written to the Calibre Interactive control file. The control includes a multi-line text entry box, in contrast to addTextVariable, which has a single line text entry box.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
VARIABLE name "textValue"
```

The value is quoted by default; set valueQuoted to false to not include quotes.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" {textValue}
 tvf::set_global_variable "name" {textValue}
```

The property valueQuoted is not used when the tvf property is set to true.

## Examples

The following code adds an optional control to define a SVRF Variable with a string value. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized.

```
var v = tool.addTextBoxVariable("VAR_METAL_TEXT",
         "net1 net2 netA netB net4 net5 netC net10 net12 net13 net15",
         "specify net labels")
v.pages = ["Defines and Variables"]
v.optional = true;
v.allowSpaces = true
```

The following statement is added to the control file for Calibre Interactive:

```
VARIABLE "VAR_METAL_TEXT" "net1 net2 netA netB net4 net5 netC net10 net12
net13 net15"
```

The control looks like this:

## Related Topics

Configuration File Format for xACT GUI Customization

addTextVariable



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addTextVariable

Function call for the custom configuration file for Calibre Interactive xACT

Adds a control to specify a Variable statement with a string value to the Calibre Interactive xACT GUI. The statement is added to the control file.

## Usage

### Function Syntax

***ctrl_var*** = **tool.addTextVariable("*name*"**[, "*textValue*"[, "*tooltip*"]]**)**

### Object Properties

*ctrl_var*.value = "*textValue*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*label*"
*ctrl_var*.promptText = "*prompt_string*"
*ctrl_var*.valueQuoted = true | false
*ctrl_var*.allowSpaces = true | false
*ctrl_var*.emptyIsValid = true | false
*ctrl_var*.editable = true | false
*ctrl_var*.visible = true | false
*ctrl_var*.optional = true | false
*ctrl_var*.specified = true | false
*ctrl_var*.tvf = true | false
*ctrl_var*.hyperlink = *link_path*
*ctrl_var*.insertToControlFile = true | false

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  The name of the variable for the Variable statement written to the Calibre Interactive control file.

- "*textValue*"

  An optional string specifying the assigned text string for the Variable statement. If "*textValue*" is not included in the function call it can be defined with the value property.

- "*tooltip*"

  An optional string specifying the hover text (tooltip) for the control. If "*tooltip*" is not included in the function call it can be defined with the description property.

- Object Properties

  The value ("*textValue*") and description ("*tooltip*") properties can be defined in the function call or with a separate property definition.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

See "Option Properties" on page 421 for a description of these common properties: pages, label, description, editable, and visible.

See "Hyperlink Property to Add a Link to Custom Controls in Calibre Interactive xACT" on page 423 for a description of the hyperlink property, which makes the label of the control into a link to the specified *link_path*.

The properties specific to addTextVariable are the following:

**Table 9-35. Object Properties for addTextVariable**

| Property | Allowed Values | Description |
|---|---|---|
| value | "*textValue*" (string) | The assigned value for the Variable statement. |
| valueQuoted | true \| false (Boolean) | Text strings are enclosed in quotes by default. Set valueQuoted to false to not include the quotes. |
| allowSpaces | true \| false (Boolean) | Allow spaces in strings; the default is false. |
| emptyIsValid | true \| false (Boolean) | Allow an empty *textValue* string; the default is false. |
| promptText | "*prompt_string*" (string) | The prompt string for the text entry box. The prompt is used only if an initial value is not set with the value property. |
| optional | true \| false (Boolean) | Places a checkbox next to the control. The default is false, resulting in a required control. |
| specified | true \| false (Boolean) | Sets the state of the checkbox when optional=true. |
| tvf | true \| false (Boolean) | Adds a Tcl variable to the Calibre Interactive control file when true is specified and a TVF rule file is used. The default is false if this property is not defined. If true is specified when not using a TVF rule file, the configuration command does not output any statement to the control file. |
| insertToControlFile | true \| false (Boolean) | Specifies whether to write the Variable statement to the control file. The default is true; modify this property with caution. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Description

The function call tool.addTextVariable adds a variable control with a string value to the GUI, which causes a SVRF Variable statement to be written to the Calibre Interactive control file. The control includes a text entry box.

The following statement is written to the control file if the control is required (optional = false) or if the optional checkbox is checked (optional = true and specified = true):

```
VARIABLE name "textValue"
```

The value is quoted by default; set valueQuoted to false to not include quotes.

For a TVF rule file with the tvf property set to true, the following statements are written to the control file:

```
set "name" {textValue}
 tvf::set_global_variable "name" {textValue}
```

The property valueQuoted is not used when the tvf property is set to true.

## Examples

The following code adds an optional control to define a SVRF Variable with a string value. The code is placed within the onInitialized section of the configuration file or within a user-defined function called from onInitialized. Only the variable name is specified in the function call—the value and tooltip are specified as property definitions.

```
var v = tool.addTextVariable("VAR_METAL_TEXT")
v.pages       = ["Defines and Variables"]
v.label       = "Set VAR_METAL_TEXT"
v.description = "SVRF Variable VAR_METAL_TEXT for metal text";
v.value       = "net1";
v.optional    = true;
```

The following statement is added to the control file for Calibre Interactive:

```
VARIABLE VAR_METAL_TEXT "net1"
```

## Related Topics

Configuration File Format for xACT GUI Customization

addTextBoxVariable



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Configuration File Functions for Trigger Functions

Several functions are available for the Calibre Interactive xACT custom configuration file to specify pre- and post-execution trigger functions. The trigger functions can run before or after the batch Calibre run, and can run in the shell environment or the design tool environment.

See "Configuration File Format for xACT GUI Customization" on page 410 for information on the file format and how to invoke Calibre Interactive xACT with a custom configuration file. The trigger functions added with the configuration file are displayed in the Triggers Pane.

- **Trigger Parameters —** Certain filenames and other values specified in the GUI can be passed to the trigger function using trigger parameters.
- **add…Trigger —** The functions to add trigger definitions specify the execution environment (shell or design tool) and timing for the trigger execution.

**Trigger Parameters in Calibre Interactive xACT** . . . . . 543
**add ... Trigger** . . . . . 545

## Trigger Parameters in Calibre Interactive xACT

Certain filenames and other parameters from the Calibre Interactive GUI can be passed to a trigger function using replaceable parameters. For example, the top cell name and the source netlist name can be passed to a trigger function using replaceable trigger parameters %L and %s, respectively.

The parameters listed in the table are recognized for triggers and substituted with the appropriate runtime values.

**Table 9-36. Replaceable Trigger Parameters**

| Runtime Replacement | Parameter |
|---|---|
| DRC or DFM RDB | %b |
| Hcell file | %h |
| Layout format<br>(GDSII, OASIS, OPENACCESS) | %F |
| Layout path | %l |
| Layout primary cell | %L |
| Layout library | %y |
| Layout view | %v |
| LVS Report file | %n |
| PEX netlist | %p |
| PEX netlist format | %f |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-36. Replaceable Trigger Parameters (cont.)**

| Runtime Replacement | Parameter |
|---|---|
| PEX Report file | %m |
| Rule file | %r |
| Run directory | %d |
| Runset file | %R |
| Source path for primary netlist source[1] | %s |
| Source primary cell | %S |
| Source path for all netlist sources[2] | %N |
| Termination status:<br>0 — Normal<br>1 — Aborted<br>2 and above — the Calibre exit code | %e |

[1] For SPICE netlist input, %s returns the netlist(s) listed on the **Inputs** page. In the case of VERILOG or MIXED source input, %s returns the SPICE file that is translated from the Verilog source input.

[2] In addition to the source paths returned by %s, the %N trigger parameter also returns the additional SPICE netlist files specified on the **Database** page in the Library area. In the case of MIXED source input, %N also includes the SPICE netlist files listed on the **Inputs** page.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# add ... Trigger

Function call for the custom configuration file for Calibre Interactive xACT

A set of function calls to add trigger definitions to the Triggers page in the Calibre Interactive xACT GUI. Trigger execution is relative to the Calibre nmLVS or Calibre xACT step. Triggers can execute in the shell environment or the design tool environment.

## Usage

### Function Syntax

***trig_var*** = **tool.addProcessPreTrigger("*run_mode*", "*name*")**

***trig_var*** = **tool.addLayoutPreTrigger("*run_mode*", "*name*")**

***trig_var*** = **tool.addSchematicPreTrigger("*run_mode*", "*name*")**

***trig_var*** = **tool.addProcessPostTrigger("*run_mode*", "*name*")**

***trig_var*** = **tool.addLayoutPostTrigger("*run_mode*", "*name*")**

***trig_var*** = **tool.addSchematicPostTrigger("*run_mode*", "*name*")**

### Object Properties

*trig_var*.command = "*trig_command_and_args*"
*trig_var*.specified = true | false

## Parameters

### Function Name

- **addProcessPreTrigger**

  Adds a pre-execution trigger function that runs in the shell environment.

- **addLayoutPreTrigger**

  Adds a pre-execution trigger function that runs in the layout design tool environment.

- **addSchematicPreTrigger**

  Adds a pre-execution trigger function that runs in the schematic design tool environment.

- **addProcessPostTrigger**

  Adds a post-execution trigger function that runs in the shell environment.

- **addLayoutPostTrigger**

  Adds a post-execution trigger function that runs in the layout design tool environment.

- **addSchematicPostTrigger**

  Adds a post-execution trigger function that runs in the schematic design tool environment.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Arguments

- ***trig_var***

  The variable name for the control, which must be unique within the configuration file. The variable name is used to set the object properties.

- ***"run_mode"***

  A required argument specifying the run mode for trigger function. The run mode is one of the following, where the quotes are required:

  "runLvs" — Execute the trigger relative to the Calibre nmLVS step.

  "runXact" — Execute the trigger relative to the Calibre xACT step.

  The choice of pre- or post-execution is determined by the function call.

- ***"name"***

  A required string used to identify the trigger, which must be unique within the configuration file and cannot include spaces. The ***"name"*** is used internally.

### Object Properties

| Property | Allowed Values | Description |
|---|---|---|
| command | "*trig_command_and_args*" (string) | The command and arguments (if any) to be executed for the trigger function; must be enclosed in quotes.<br>Replaceable parameters may be used in the command arguments; see "Trigger Parameters in Calibre Interactive xACT" on page 543.<br>Special characters must be escaped using standard JavaScript methods. |
| specified | true \| false (Boolean) | Specifies whether to execute the trigger function; the default is true. This property corresponds to the "Use" column in the triggers table on the Triggers page. |

## Description

The add…Trigger function calls are placed in the custom configuration file and add trigger definitions to the Triggers Pane; such trigger definitions cannot be edited in the GUI.

The choice of function call determines whether the specified trigger function executes before or after batch Calibre (add*PreTrigger and add*PostTrigger); the ***"run_mode"*** argument specifies whether the trigger function executes relative to the LVS or parasitic extraction step. The choice of function call also determines the execution environment for the trigger: shell (addProcess*Trigger), layout design tool (addLayout*Trigger), or schematic design tool (addSchematic*Trigger).



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The trigger function to be executed is defined with the command object property.

## Examples

### Process Trigger Example

This example adds two post-LVS triggers and one post-parasitic extraction trigger. The triggers use the Linux echo command to print information. The second post-LVS trigger uses the trigger parameter %n, which is replaced with the name of the LVS report. The post-parasitic extraction trigger prints the extracted parasitic netlist filename and format using the trigger parameters %p and %f. Quotes around the argument to the echo function are escaped.

```
// add lvs post-triggers
var trigLVSpost_1 = tool.addProcessPostTrigger("runLvs", "LVSpostTrig_1")
trigLVSpost_1.command = "echo finished LVS";

var trigLVSpost_2 = tool.addProcessPostTrigger("runLvs", "LVSpostTrig_2")
trigLVSpost_2.command = "echo \"LVS report %n\"";

// xact post-trigger
var trigXactPost_1 = tool.addProcessPostTrigger("runXact", "xactPost_1")
trigXactPost_1.command = "echo \"Done, PEX netlist %p, format %f\"";
```

### Runset Trigger Example

The trigger functions are also used to save trigger definitions to the runset. However, the tool name “pex” is used instead of “tool” when a function is used in the runset.

This is an example of a trigger definition in the runset, where the trigger was defined in the GUI.

```
var preTrigger_2_runXact_trig = pex.addProcessPreTrigger("runXact",
                                                         "preTrigger_2")
preTrigger_2_runXact_trig.command = "echo \"Running on cell %L\""
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Configuration File Functions to Add Buttons

You can add buttons to the Calibre Interactive xACT GUI and specify a callback function that is executed when the button is clicked.

| Command | Description |
|---|---|
| addAction | Adds a user-defined button with a callback function to the Calibre Interactive xACT GUI. |
| addChoicesAction | Adds a user-defined button with a callback function to the Calibre Interactive xACT GUI. The button has a dropdown list of choices. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addAction

Function call for the custom configuration file for Calibre Interactive xACT

Adds a user-defined button with a callback function to the Calibre Interactive xACT GUI.

## Usage

### Function Syntax

***ctrl_var* = tool.addAction("*name*")**

### Object Properties

*ctrl_var*.pages = ["*page_name*" …]
*ctrl_var*.label = "*label*"
*ctrl_var*.description = "*tooltip*"
*ctrl_var*.activated.connect(*cmd*)
*ctrl_var*.location = Location.Footer | Location.Header | Location.Body
| Location.HeaderAndFooter

### Control Functions

*ctrl_var*.setEnabled
*ctrl_var*.setDisabled

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  A required string used internally to identify the button. The ***name*** must be unique within the configuration file. The ***"name"*** is used internally.

- Object Properties

**Table 9-37. Object Properties for addAction**

| Property | Allowed Values | Description |
|---|---|---|
| pages | list of strings | A list of the pages that the control appears on. For example: `["Outputs", "Options"]` |
| label | string | The button label. |
| description | string | The tooltip. |
| activated.connect(*cmd*) | function name | The function that is executed when the button is clicked. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-37. Object Properties for addAction (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| location | Location.Footer \|Location.Header \| Location.Body \| \| Location.HeaderAndFooter | The location of the button; Location.Footer is the default. The header and footer locations are left justified. The body location is centered under the last control on the page. |

- Control Functions

  Control functions are built-in functions for a control.

**Table 9-38. Control Functions for addAction**

| Function | Description |
|---|---|
| setEnabled | Enable the button. Buttons created by addAction are enabled by default. |
| setDisabled | Disable the button. A disabled button is grayed out and not active. |

## Description

addAction adds a user-defined button to the Calibre Interactive xACT GUI. When the button is clicked, the function specified by the activated.connect property is executed. The activated.connect property can specify a function defined in the configuration file or a user-defined command called with launchCommand.

See the connect function to attach a function to a signal from the GUI. To add a callback function to the Run button, use the tool property tool.runCallback. See "Example: Attach Callback Function to Run Button With tool.runCallback" on page 420.

## Examples

The following example adds a button with the internal name "SVDBon" and label "SVDB turn on". The function setSVDBOn is called when the button is clicked.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
Configuration {
   onInitialized: {
   // add button to turn on svdb options
   var bSVDB = tool.addAction("SVDBon");   //name is internal
   bSVDB.pages = ["Rules"];
   bSVDB.label = "SVDB turn on"; // the button label
   bSVDB.description = "turn on SVDB and all the options"; // tooltip
   bSVDB.activated.connect(setSVDBOn);
   bSVDB.location = Location.Body      // default location is Footer
   } // end onInitialized


// define function to set SVDB options if SVDBon button is clicked
function setSVDBOn() {
      tool.svdb.query.value = true
      tool.svdb.asciixref.value = true
      tool.svdb.cci.value = true
      tool.svdb.genParasiticData.value = true
      tool.svdb.pinloc.value = "PINLOC"
      tool.svdb.pinloc.specified = true
}

} // end Configuration
```

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# addChoicesAction

Function call for the custom configuration file for Calibre Interactive xACT

Adds a user-defined button with a callback function to the Calibre Interactive xACT GUI. The button has a dropdown list of choices.

## Usage

### Function Syntax

***ctrl_var* = tool.addChoicesAction("*name*", *value_list*[, *label_list*[, *tooltip_list*]])**

### Object Properties

*ctrl_var*.pages = *page_list*
*ctrl_var*.label = "*control_label*"
*ctrl_var*.description = "*control_tooltip*"
*ctrl_var*.value = "*value*"
*ctrl_var*.valueChanged.connect(*cmd*)
*ctrl_var*.location = Location.Footer | Location.Header | Location.Body
| Location.HeaderAndFooter
*ctrl_var*.choiceAt(*i*).label = "*list_label*"
*ctrl_var*.choiceAt(*i*).value = "*list_value*"
*ctrl_var*.choiceAt(*i*).description = "*list_tooltip*"
*ctrl_var*.choiceAt(*i*).compatible = true | false

### Control Functions

*ctrl_var*.setEnabled
*ctrl_var*.setDisabled

## Arguments

- ***ctrl_var***

  The variable name for the control.

- ***name***

  A required string used internally to identify the button. The ***name*** must be unique within the configuration file. The ***"name"*** is used internally.

- ***value_list***

  A required list of possible assigned values for the control.

  For example: ["ONE", "TWO", "THREE"]

- *label_list*

  An optional list of labels for the control's dropdown list.

  For example: ["one", "two", "three"]

  If *label_list* is not included in the function call, the labels can be defined with the choiceAt(*i*).label property.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- *tooltip_list*

  An optional list of tooltips (hover text) for the control's dropdown list.

  For example: ["first choice", "second choice", "third choice"]

  If *tooltip_list* is not included in the function call, the tooltips can be defined with the choiceAt(*i*).description property.

- Object Properties

  The label (*label_list*) and description (*tooltip_list*) properties for the choices list can be defined in the function call or with a separate property definition using choiceAt(*i*).

**Table 9-39. Object Properties for addChoicesAction**

| Property | Allowed Values | Description |
|---|---|---|
| pages | list of strings | A list of the pages that the control appears on. For example: `["Outputs", "Options"]` |
| label | string | The control label. The control label is only used if location is set to Location.Body and is displayed to the left of the button. |
| description | string | The tooltip. |
| value | | The value for the control. This value is passed to the callback function for the button. Set this property to one of the choice values to provide an initial value for the control. |
| valueChanged.connect(*cmd*) | function name | *cmd* specifies the function that is executed when the a new choice is selected from the dropdown menu. The new control value (*ctrl_var*.value) is passed to the function. |
| location | Location.Footer \|Location.Header \| Location.Body \| \| Location.HeaderAndFooter | The location of the button; Location.Footer is the default. The header and footer locations are left justified. The body location is centered under the last control on the page. |
| choiceAt(*i*).label | string | The label for the $i^{th}$ choice in the dropdown list of the GUI control. Numbering starts at zero. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 9-39. Object Properties for addChoicesAction (cont.)**

| Property | Allowed Values | Description |
|---|---|---|
| choiceAt(*i*).value | | The assigned value for the $i^{th}$ choice. When a new choice is selected, the choices[*i*].value is assigned to *ctrl_var*.value. |
| choiceAt(*i*).description | string | The tooltip for the $i^{th}$ choice in the dropdown list of the GUI control. |
| choiceAt(*i*).compatible | true \| false<br>(Boolean) | Specifies whether the $i^{th}$ choice in the dropdown list of the GUI control is available (the default) or unavailable (dimmed). |

- Control Functions

  Control functions are built-in functions for a control.

**Table 9-40. Control Functions for addChoicesAction**

| Function | Description |
|---|---|
| setEnabled | Enable the button. Buttons created by addAction are enabled by default. |
| setDisabled | Disable the button. A disabled button is grayed out and not active. |

## Description

addChoicesAction adds a user-defined button to the Calibre Interactive xACT GUI. The button has a dropdown list of choices. When the selection is changed, the following actions occur:.

- The choices[*i*].value is assigned to *ctrl_var*.value, where *i* is the selected choice.
- The function specified by the valueChanged.connect property is executed, with *ctrl_var*.value as an argument.

When combined with appropriate logic, the choiceAt(*i*).compatible property can be used to set a choice as available or unavailable depending on other settings in the GUI.

See the connect function to attach a function to a signal from the GUI. To add a callback function to the Run button, use the tool property tool.runCallback. See "Example: Attach Callback Function to Run Button With tool.runCallback" on page 420.

## Examples

This example adds a button with a dropdown list of two choices.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
Configuration {
   onInitialized: {

      var d = tool.addChoicesAction("setupAB",
               ["Case_A", "Case_B"],
               ["Setup A", "Setup B"],
               ["hcell file", "layout case, extract temp, noinstancex"]);
      d.pages       =  ["Rules"];
      d.label       = "Setup";
      d.description = "Choose setup A or B";

      d.value = d.choiceAt(0).value;
      d.valueChanged.connect(setupChoice);
      // call setupChoice with the value of the initial setting
      setupChoice(d.value);

   } // end onInitialized

function setupChoice(arg) {

   switch (arg) {
      case "Case_A":
         tool.hCells.hCellsFile.value = "hcells"
         tool.hCells.hCellsFile.specified = true
         break;
      case "Case_B":
         tool.pexNetlist.noinstancex.value = true
         tool.pexExtractTemp.temperatures.specified = true
         tool.layoutCase.caseSensitive.specified = true
         break;
      default:
         break;

   } // switch
 } // setupChoice

} // end Configuration
```

## Related Topics

Configuration File Format for xACT GUI Customization



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Troubleshooting Configuration Files for the Calibre Interactive xACT GUI

There are some common errors in writing the configuration file for customization the Calibre Interactive xACT GUI. This section outlines methods for finding and fixing these errors.

**Note**

The actual error may be at the one plus the line number reported in the Transcript page.

## Problem: Control does not appear in the GUI

Cause: Check these common programming errors:

- Define or Variable name is not entered correctly throughout the code for the control.
- Other JavaScript entry error.

Solution: Inspect the code and correct the error.

## Problem: Quotes are around the assigned value but not wanted

Cause: Quotes are used by default for the text, choices, and addListVariable controls.

Solution: Set *ctrl_var*.valueQuoted = false.

## Problem: The text entry box does not allow spaces

Cause: Spaces are not allowed by default.

Solution: Set *ctrl_var*.allowSpaces = true.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 10
# Using Calibre Interactive to Perform PERC

The Calibre Interactive PERC GUI allows you to specify command line options and select rule file statements for a Calibre PERC run. You can save GUI settings to a Calibre Interactive runset and load them in future runs.

This chapter assumes you are familiar with the terms and concepts presented in the chapter "Running Calibre Interactive" on page 63.

You should also be familiar with the *Calibre PERC User's Manual*.

**Video**: Getting Started with Calibre Interactive PERC

**Introduction to Calibre Interactive PERC** . . . . . 557
**Specifying a Rule File for a Calibre PERC Run** . . . . . 560
**Specifying the PERC Run and Analysis Type** . . . . . 561
**Inputs for a Calibre Interactive PERC Run** . . . . . 563
**Specifying Waivers for a Calibre PERC Run** . . . . . 574
**Specifying the Calibre PERC Cell-Based Flow** . . . . . 575
**Creating a Calibre PERC Rule File with High Level Checks** . . . . . 576
**Specifying Outputs for a Calibre PERC Run** . . . . . 578
**Running Calibre Interactive PERC** . . . . . 579
**Run Control, Remote Execution, and Licensing Options** . . . . . 581
**Setting Calibre PERC Options** . . . . . 582

## Introduction to Calibre Interactive PERC

The Calibre Interactive PERC Graphical User Interface (GUI) allows you to specify command line options and selected rule file statements that control a Calibre PERC run.

A common scenario for performing Calibre PERC from Calibre Interactive assumes that a runset file containing all the necessary settings already exists. Your time is spent running Calibre PERC and correcting errors in an interactive verification loop, as illustrated in



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Figure 10-1. In this scenario, you spend very little time setting up the Calibre run because you can load the necessary settings from a runset. You can set up Calibre Interactive to load the runset automatically on invocation as described in "Loading Runsets on Invocation" on page 39.

Figure 10-1 shows a typical verification loop for Calibre PERC analysis of a layout database; in this case Calibre PERC extracts the layout netlist as part of the run. Calibre PERC can also analyze the netlist directly.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 10-1. Generic Calibre PERC Flow for Layout Analysis**

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Using Calibre RVE for PERC [Calibre RVE User's Manual]

# Specifying a Rule File for a Calibre PERC Run

You need to load a rule file for any Calibre PERC run. You can load a rule file automatically with a runset or you can specify one manually using this procedure.

To automatically generate a rule file for the High Level Checks flow, see "Creating a Calibre PERC Rule File with High Level Checks" on page 576.

### Prerequisites

- Calibre Interactive PERC is running; see "Invoking Calibre Interactive Tools" on page 64
- A rule file for Calibre PERC is available

### Procedure

1. Click the Rules button to display the Rules pane.

2. Specify the rule filename in the PERC Rules File field. You can use the **View** button to view and edit the rule file.

   Be aware that Calibre Interactive automatically recognizes a rule file with the literal name "rules" and displays the name in green. Because the name is shown in green, you may assume the file has been loaded but it has *not* been; the file must be explicitly loaded using the **Load** button.

3. Click **Load** to populate the GUI with the settings in the rule file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

A rule file can also be specified in a runset. After loading that runset, the rule file is included in the control file but its setting will not be loaded until you explicitly click the **Load** button.

**Tip**

The setting "Read Inputs/Outputs fields while loading rules file" on the **Misc** tab of the Setup Preferences dialog box controls whether or not rule file settings populate the GUI options on the Inputs and Outputs panes when **Load** is clicked. This setting is enabled by default.

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

4. Specify the run directory name in the PERC Run Directory field.
5. (Optional) View the layer derivation tree is at the bottom of the Rules pane; see "Layer Derivations Tree" on page 126 for information.

### Related Topics

About Rule Files

Including Other Rule Files and SVRF Commands in Calibre Interactive

# Specifying the PERC Run and Analysis Type

You can specify a hierarchical, flat, or Calibre PERC LDL run type, and choose to analyze the geometric layout, the layout netlist, or the source netlist.

### Prerequisites

- See these topics in the *Calibre PERC User's Manual* for special requirements for the Calibre PERC-LDL applications:

  "Current Density Checks" (PERC LDL CD)

  "Point-to-Point Resistance Checks" (PERC LDL P2P)

  "High-Level Checks and Topological DRC" (PERC LDL DRC)

### Procedure

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

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Choose the “Run” type as follows:
   - **PERC (Hierarchical)** — Run Calibre PERC hierarchically. This setting adds the -hier command line option.
   - **PERC (Flat)** — Run Calibre PERC flat.
   - **PERC-LDL** — Run the Calibre PERC logic driven layout (LDL) application. This setting uses the -ldl and -hier command line options.
3. Choose your analysis type in the Analyze section. The available settings depend on the Run type, as follows:
   - “PERC (Hierarchical)” and “PERC (Flat)” runs:
     - **Layout** — Run Calibre PERC on a geometric layout database. Calibre PERC extracts a layout SPICE netlist and runs on that netlist.

       Layout analysis inserts the PERC Netlist LAYOUT command in the rule file and uses the -spice option on the command line.
     - **Layout Netlist** — Run Calibre PERC on an existing layout netlist. The netlist must be in SPICE format, and can be a netlist extracted during a previous Calibre PERC Layout analysis.

       Layout netlist analysis inserts the PERC Netlist LAYOUT command in the rule file; the -spice option on the command line is not used.
     - **Source Netlist** — Run Calibre PERC on a source netlist.

       Source netlist analysis inserts the PERC Netlist SOURCE command in the rule file; the -spice option on the command line is not used.
   - “PERC-LDL” runs:
     - **Layout** — Run Calibre PERC LDL on a geometric layout database.
     - **Source-Based Flow** — Run Calibre PERC LDL DRC on a source netlist. The rule file must include the “ldl::export_perc -source” command.

## Related Topics

Inputs for a Calibre Interactive PERC Run

Specifying a Rule File for a Calibre PERC Run



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Inputs for a Calibre Interactive PERC Run

Inputs for a Calibre Interactive PERC run are specified on the Inputs pane, with some options available on the PERC Options pane. You specify the layout or source input, the run type, and Hcell data.

**Specifying Inputs for a Calibre PERC Run** . . . . . 563
**Input Sources for a Calibre PERC Run** . . . . . 564
**Specifying Calibre PERC Inputs for Layout Analysis** . . . . . 565
**Specifying Calibre PERC Inputs for Layout Netlist Analysis** . . . . . 568
**Specifying Calibre PERC Inputs for Source Netlist Analysis** . . . . . 569
**Supplying Hcell Data for a Calibre PERC Run** . . . . . 572
**Specifying a Calibre PERC LDL Probe Path** . . . . . 573
**Specifying Multiple Layouts for a Calibre PERC Run** . . . . . 573

## Specifying Inputs for a Calibre PERC Run

The inputs you specify for a Calibre PERC run depend on the type of analysis you choose. The basic procedure for specifying Calibre PERC inputs is outlined here; see the individual sections for detailed instructions.

### Procedure

1. Select the type of Calibre PERC run, as described in "Specifying the PERC Run and Analysis Type" on page 561.
2. Specify the inputs needed, depending on the analysis type:
   - "PERC (Hierarchical)" and "PERC (Flat)" runs:
     - **Layout** — Specifying Calibre PERC Inputs for Layout Analysis
     - **Layout Netlist** — Specifying Calibre PERC Inputs for Layout Netlist Analysis
     - **Source Netlist** — Specifying Calibre PERC Inputs for Source Netlist Analysis
   - "PERC-LDL" runs:
     - **Layout** — Provide layout information using instructions in "Specifying Calibre PERC Inputs for Layout Analysis" on page 565 as a guideline.
     - **Source-Based Flow** — Provide layout and source information, using instructions in "Specifying Calibre PERC Inputs for Layout Analysis" on page 565 and "Specifying Calibre PERC Inputs for Source Netlist Analysis" on page 569 as a guideline.

   The source of the required input can be an open layout or schematic viewer, or a file, as explained in "Input Sources for a Calibre PERC Run" on page 564.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Specify Hcells as described in "Supplying Hcell Data for a Calibre PERC Run" on page 572.
4. If you are doing a layout analysis, specify other options as needed, as described in "Specifying Multiple Layouts for a Calibre PERC Run" on page 573 and "Setting Database and General Options in Calibre Interactive" on page 93.
5. (Optional) Specify a Layout Text file as described in "Specifying a Layout Text File in Calibre Interactive" on page 93.

### Related Topics

Specifying the Calibre PERC Cell-Based Flow

# Input Sources for a Calibre PERC Run

The origin of your input is different depending on whether you run Calibre Interactive connected to a layout or schematic viewer or not.

The following table shows these differences.

**Table 10-1. Sources for Calibre PERC Input Data**

| Analysis Type | Viewer Connected | No Viewer Connected |
|---|---|---|
| Layout | One of the following:<br>a) The layout data comes from the currently-open layout viewer when the "Export from Layout Viewer" option is enabled.<br>b) The layout data comes from an existing set of layout files. | Layout data exists as one or more files used for input. |
| Layout Netlist | The netlist data comes from a file. | The netlist data comes from a file. |
| Source Netlist | One of the following:<br>a) The source netlist comes from the currently-open schematic viewer when the "Export from Schematic Viewer" option is enabled.<br>b) the source netlist comes from an existing set of files. | The source netlist comes from an existing set of files. |

### Related Topics

Running Calibre Interactive in Batch Mode

Export from Viewer Preferences



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Specifying Calibre PERC Inputs for Layout Analysis

When you analyze a geometric layout in a Calibre PERC run, the layout can be exported from an open layout viewer or come from database files. Calibre PERC extracts the layout netlist before doing the analysis.

**Reading Geometric Layout Input from an Open Design Tool in Calibre Interactive PERC** . . . . . 565

**Reading Geometric Layout Input from File in Calibre Interactive PERC** . . . . . 567

## Reading Geometric Layout Input from an Open Design Tool in Calibre Interactive PERC

Layout data for a Calibre Interactive PERC run is specified on the Inputs pane on the **Layout** tab. You can export the layout from an open design tool. The layout netlist is extracted from the geometric layout.

### Procedure

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

2. Make sure that “Layout” is selected at the top of the Inputs pane in the Analyze section.
3. Click the **Layout** tab.
4. Enable “Export from layout viewer” to specify that the layout is exported from a connected design tool; see “Communication with the Design Tool” on page 72 for more information.
5. Select the *exported* layout format using the “Format” dropdown button list. **OASIS**, **GDSII**, and **LEFDEF** formats are available if “Export from layout viewer” is selected.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

6. Specify the filename for the *exported* layout database in the "Layout File" or "DEF File" field. For LEFDEF format, also provide LEF input as described in "Providing LEF/DEF Input" on page 88.

   If you are performing a Calibre Interactive batch run, see "Specifying Layout and Schematic Export for a Calibre Interactive Batch Run" on page 104.

   **Caution**

   When "Export from layout viewer" is enabled, the file specified in the "File" field is overwritten if it exists already. Do not enter a tapeout database in the "File" field, or use the extracted layout named in the "File" field for tapeout or any other use.

   When exporting a layout database from a connected viewer, an intermediate file is extracted and saved to the file named in the "Layout File" field. This filename is generated automatically based on the setting on the **Templates** > **Inputs** tab of Setup Preferences dialog box (**Setup** > **Preferences**).

   If the filename is red, the file does not exist and Calibre Interactive creates the file. If the filename is green, the file exists and will be overwritten. Enter a different filename if you want to keep the existing file.

7. (Optional) If you have multiple layout inputs for your run, you can specify them on the **Database** > **Library** tab on the PERC Options pane. This is valid both for layout data from a file and exported from a layout viewer. See "Specifying Multiple Layouts for a Calibre PERC Run" on page 573.

8. Verify the top cell name in the "Top Cell" field. The top cell is determined from the current cell in the viewer.

9. Enter the name of the extracted layout netlist in the "Layout Netlist" field. This option is for hierarchical runs only; in addition, it is not present for Calibre PERC-LDL runs.

   When connected to a design tool, a system-generated SPICE netlist name appears in the "Layout Netlist" field by default. This name is taken from the "Extract to File" setting in the Setup Preferences dialog box on the **Templates** > **Outputs** tab. However, you can choose a different filename.

## Related Topics

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Primary [Standard Verification Rule Format (SVRF) Manual]

Layout System [Standard Verification Rule Format (SVRF) Manual]

Specifying the Calibre PERC Cell-Based Flow

Templates for File Naming



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Reading Geometric Layout Input from File in Calibre Interactive PERC

Layout data for a Calibre Interactive PERC run is specified on the Inputs pane on the **Layout** tab. You can read the layout data from design files saved on disk. The layout netlist is extracted from the geometric layout.

## Procedure

1. Click **Inputs** to display the Inputs pane.
2. Make sure that "Layout" is selected at the top of the Inputs pane in the Analyze section.
3. Click the **Layout** tab.
4. Uncheck "Export from layout viewer" to specify that the layout data is contained in the specified files.
5. Select a layout format using the "Format" dropdown button list.

   For a full list of supported layout databases, refer to the "Layout Databases" section of the *Calibre Verification User's Manual*. Also see "Third Party Database Options" on page 80.
6. Specify the filename for the layout database in the "Layout File" field for GDSII and OASIS format. For other formats, see the following topics for additional information:
   - **LEFDEF** — see "Providing LEF/DEF Input" on page 88
   - **OPENACCESS** — see "Providing OpenAccess Input" on page 90
7. Specify the top cell name in the "Top Cell" field.
8. Enter the name of the extracted layout netlist in the "Layout Netlist" field. This option is for hierarchical runs only; in addition, it is not present for Calibre PERC-LDL runs.
9. (Optional) If you have multiple layout inputs for your run, you can specify them using the **Database** > **Library** tab on the PERC Options pane.

## Related Topics

Layout Path [Standard Verification Rule Format (SVRF) Manual]

Layout Primary [Standard Verification Rule Format (SVRF) Manual]

Layout System [Standard Verification Rule Format (SVRF) Manual]

Specifying the Calibre PERC Cell-Based Flow

Setting Database and General Options in Calibre Interactive



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Specifying Calibre PERC Inputs for Layout Netlist Analysis

When you analyze a layout netlist in a Calibre PERC run, the layout netlist must already exist as a SPICE file.

## Procedure

1. Click **Inputs** to display the Inputs pane.
2. Make sure that “Layout Netlist” is selected at the top of the Inputs pane in the Analyze section.
3. Click the **Layout** tab.

   The input fields for the geometric layout database are not available for layout netlist runs.
4. Specify the top cell name in the “Top Cell” field.
5. Enter the name of the layout netlist in the “Layout Netlist” field. The layout netlist must be in SPICE format.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Specifying Calibre PERC Inputs for Source Netlist Analysis

When you analyze a source netlist in a Calibre PERC run, the source netlist can be exported from an open schematic viewer or come from an existing netlist file.

**Reading Source Netlist Data from a Schematic Viewer in Calibre Interactive PERC . 569**
**Reading Source Netlist Data from File in Calibre Interactive PERC. . . . . 571**

## Reading Source Netlist Data from a Schematic Viewer in Calibre Interactive PERC

You can read source netlist data from a connected schematic viewer in Calibre Interactive PERC.

Corresponding rule file statements: Source Path, Source Primary, Source System

### Procedure

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

2. Make sure that “Source Netlist” is selected in the “Analyze” dropdown button list.
3. Click the **Netlist** tab.
4. Enable “Export from schematic viewer” to specify that a connected schematic viewer supplies the schematic source netlist.

   Batch runs: In addition, you must also disable the setting “Ignore layout and source export in batch mode” in the Setup Preferences dialog box on the **Misc** tab. Also see “Specifying Layout and Schematic Export for a Calibre Interactive Batch Run” on page 104 for details on the viewer connection for a batch run.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. Specify the format of the *exported* source netlist using the “Format” dropdown button list.
   - **SPICE** — The exported netlist is in SPICE format.
   - **VERILOG** — The exported netlist is in Verilog format. If you specify Verilog as the source format, Calibre Interactive calls a utility named v2lvs to translate the Verilog netlist to SPICE. Choose **Setup > Verilog Translator** to set up the translation options; see the section “Setting Up the Verilog Translator (v2lvs)” on page 254. The translated SPICE netlist is then used for LVS comparison; the SPICE netlist name is taken from the “Files” field.

   The “MIXED” option is not available when “Export from schematic viewer” is enabled.
6. Specify the filename for the *exported* source netlist in the “Spice Files” or “Verilog Files” field; the exported netlist is written to this file. The filename is generated automatically based on the “Source Path” template setting on the **Templates > Inputs** tab of the Setup Preferences dialog box.
7. Verify the top cell name in the “Top Cell” field. The top cell is determined from the cell currently open in the viewer when “Export from schematic viewer” is enabled.
8. (Optional) Specify additional SPICE netlist files. This may be necessary to provide INCLUDE files for the source netlist.
   a. Click the **PERC Options** button or choose **Setup > PERC Options**.
   b. Click the **Database** tab and the **Library** sub-tab.
   c. Use the buttons next to “Additional SPICE Files” to specify the required SPICE files.

   **Note**

   Calibre Interactive verifies that the source netlist is complete before starting the run. You can turn off this check by disabling “Check source netlist is complete before running PERC” on the **Misc** tab of the Setup Preferences dialog box. See “Miscellaneous Preferences” on page 156 for information on the source netlist check.

## Results

The appropriate statements are written to the Calibre Interactive control file at runtime.

If you specified multiple source netlist files, Calibre Interactive creates a file named *_source.net_* with .INCLUDE statements for each netlist file; *_source.net_* is used as the source netlist for the Source Path statement in the control file. Files entered in the **Database > Library** tab on the PERC Options pane are listed before files in the **Netlist** tab on the Inputs pane.

## Related Topics

Templates for File Naming



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Communication with the Design Tool

# Reading Source Netlist Data from File in Calibre Interactive PERC

You can read source netlist data from file in Calibre Interactive PERC.

Corresponding rule file statements: Source Path, Source Primary, Source System

## Procedure

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

2. Make sure that “Source Netlist” is selected in the “Analyze” dropdown button list.
3. Click the **Netlist** tab.
4. Uncheck “Export from schematic viewer” to specify that schematic netlist files provide the source schematic.
5. Specify the format of the netlist using the “Format’ dropdown button list.
   - **SPICE** — The netlist is in SPICE format.
   - **VERILOG** — The netlist is in Verilog format. If you specify Verilog as the source format, Calibre Interactive calls a utility named v2lvs to translate the Verilog netlist to SPICE. Choose **Setup** > **Verilog Translator** to set up the translation options; see the section “Setting Up the Verilog Translator (v2lvs)” on page 254. The translated SPICE netlist is then used for Calibre PERC analysis.The SPICE netlist name is taken from the “Verilog Files” field.
   - **MIXED** — Select the Mixed option to use a mixed mode of both SPICE and Verilog files. Separate netlist entry fields are displayed for the SPICE and Verilog files. This option may not be specified when “Export from schematic viewer” is enabled.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

6. Specify the source netlist filename.
7. Enter the top cell name in the “Top Cell” field.
8. (Optional) Specify additional SPICE netlist files. This may be necessary to provide INCLUDE files for the source netlist.
   a. Click the **PERC Options** button or choose **Setup** > **PERC Options**.
   b. Click the **Database** tab and the **Library** sub-tab.
   c. Use the buttons next to “Additional SPICE Files” to specify the required SPICE files.

**Note**

Calibre Interactive verifies that the source netlist is complete before starting the run. You can turn off this check by disabling “Check source netlist is complete before running PERC” on the **Misc** tab of the Setup Preferences dialog box. See “Miscellaneous Preferences” on page 156 for information on the source netlist check.

### Results

The appropriate statements are written to the Calibre Interactive control file at runtime.

If you specified multiple source netlist files, Calibre Interactive creates a file named *_source.net_* with .INCLUDE statements for each netlist file; *_source.net_* is used as the source netlist for the Source Path statement in the control file. Files entered in the **Database** > **Library** tab on the PERC Options pane are listed before files in the **Netlist** tab on the Inputs pane.

### Related Topics

Specifying Calibre PERC Inputs for Source Netlist Analysis

## Supplying Hcell Data for a Calibre PERC Run

Hcells are used in a Calibre PERC run to specify subcircuits in the input that are preserved.

See the description for the -automatch and -hcell command line options in the “calibre -perc” section of the *Calibre PERC User’s Manual* for more information. Unlike a Calibre nmLVS run, Calibre PERC does not compare layout and source cells, so the hcell data does not specify correspondence between layout and source cells.

Related rule file statements: Hcell, LVS Exclude Hcell.

### Procedure

1. Click **Inputs** on the left panel.
2. Click the H-Cells tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Specify the method for hcell processing:
   - **Automatch** — All subcircuits in the input netlist are preserved during the Calibre PERC analysis, except for subcircuits listed in an LVS Exclude Hcell statement.

     Automatch is recommended when analyzing a netlist. The Automatch option is not used by the circuit extraction phase.

     Automatch is only available for hierarchical runs.
   - **Use H-Cells file** — Specify the hcell filename. Subcircuits in the hcell list are preserved and written to the extracted netlist, and preserved during Calibre PERC analysis. Subcircuits listed in an LVS Exclude Hcell statement are excluded from the hcell list. Calibre PERC only reads the layout or source column of the hcell list, depending upon which type of analysis you specified.

     This option is recommended when analyzing a geometric layout and performing netlist extraction. This option is only available for hierarchical runs and flat geometric layout analysis.

### Related Topics

Getting Started With Calibre PERC [Calibre PERC User's Manual]

# Specifying a Calibre PERC LDL Probe Path

You can specify the probe mapping file for a Calibre PERC LDL run using Calibre Interactive.

Related rule file statement: PERC LDL Probe Path

### Prerequisites

- The run type is “PERC-LDL” and the Analyze setting is “Layout” on the Inputs pane.

  The **Probes** tab on the Inputs pane is only visible when these settings are made.

### Procedure

1. Click **Inputs** on the left panel.
2. Click the **Probes** tab.
3. Check the “Use PERC LDL Probe File” checkbox and enter the path to the probe file.

# Specifying Multiple Layouts for a Calibre PERC Run

If your primary cell has references to other additional database files you may need to load multiple layouts. You can specify multiple layouts using the **Database** > **Library** sub-tab on the PERC Options pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Calibre Interactive checks the format of the additional database files and only includes files that match the format of the primary layout database.

### Prerequisites

- Main layout database in GDSII or OASIS format.
- Additional layout files in GDSII or OASIS format. Files compressed with gzip may be used. The additional layout files must be the same format as the main layout database.

### Procedure

1. Click**Inputs**, then click the **Layout** tab.
2. Enter a layout file in the “File” field of the **Layout** tab.
3. Select “GDSII” or “OASIS” for the “Format”, as appropriate.
4. Enable the PERC Options pane by choosing **Setup** > **PERC Options**.
5. Click the **Database** tab; then click the **Library** sub-tab.
6. Use the buttons next to “Additional Layout Files” to add the required files.

   The additional files are added to the Layout Path statement. Only files that match the format of the primary database are added.
7. (Optional) In order to receive a warning if the additional layout files are not the same format as the main layout database, select **Setup** > **Preferences**, click the **Misc** tab, and enable “Warn when library layout file format differs from primary layout format.” The warning is displayed when you click the **Run PERC** button.

   Additional layout files that are not the same format as the primary database are ignored.

### Related Topics

Specifying Calibre PERC Inputs for Layout Analysis

# Specifying Waivers for a Calibre PERC Run

You can specify a Calibre PERC Waiver description file in Calibre Interactive PERC. This causes the Calibre PERC Waiver flow to be used during the Calibre PERC run.

### Prerequisites

- A Calibre Auto-Waivers license in addition to the Calibre PERC license.
- A Calibre PERC waiver description file. See “Calibre PERC Waiver Flows” in the *Calibre PERC User’s Manual*.

### Procedure

1. Click the **Inputs** button on the left pane of Calibre Interactive.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Click the **Waivers** tab.
3. Enable "Use PERC Waiver Description File" and enter the path to the file.
4. (Optional) If the run type on the Inputs page is set to PERC-LDL, enable "Use PERC LDL Waiver Description File" and enter the path to the file. The waivers in this file apply to geometric (RDB) results.

### Results

The PERC Waiver Path statement is added to the control file for the run, and optionally the PERC LDL Waiver Path statement.

# Specifying the Calibre PERC Cell-Based Flow

You can specify the Calibre PERC cell-based flow in Calibre Interactive. This flow is part of the Calibre PERC LDL (Logic-Driven Layout) functionality. The flow assumes LEF/DEF design input instead of SPICE and performs topological checking without considering devices.

### Prerequisites

- A rule file set up for the Calibre PERC cell-based flow, as described in "Cell-Based Checks" in the *Calibre PERC User's Manual*.
- Layout is in LEF/DEF format.
- (Optional) A cell info file. This is needed when you are using GDS input for connectivity of certain cells. See "Cell Information File Format" in the *Calibre PERC User's Manual*.

### Procedure

1. Specify the rule file and run directory on the Rules pane.
2. Click **Inputs** on the left panel.
3. Choose a run type of "PERC-LDL. "
4. On the **Layout** tab, select "LEFDEF" in the "Format" dropdown button list.
5. Fill in the DEF and LEF input information.

   **Note**

   LEF/DEF direct read is used in the cell-based flow, so the database conversion options on the **Database** > **OpenAccess/LEFDEF** sub-tab of the PERC Options pane are not used.

6. Enable the "Enable Cell-Based flow" checkbox. This adds the -soc option to the command line.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

7. (Optional) Enable “Cell Info File” and specify a cell info file. This adds the option -cell_info_file and the specified file to the command line.

### Related Topics

Cell-Based Checks [Calibre PERC User's Manual]

Providing LEF/DEF Input

Specifying the PERC Run and Analysis Type

Specifying Inputs for a Calibre PERC Run

Specifying Outputs for a Calibre PERC Run

Running Calibre Interactive PERC

Setting Calibre PERC Options

# Creating a Calibre PERC Rule File with High Level Checks

You can generate a rule file for a Calibre PERC run with the Calibre PERC Rule Generator GUI. The rule checks that can be included are part of the high-level checks flow in Calibre PERC. You specify the various parameters and input files for each rule check using the rule generator GUI.

These sections in the *Calibre PERC User’s Manual* provide important information about the flow:

- “High-Level Checks”
- “High-Level Check Types”
- “Design Flow Selection”

### Prerequisites

- Calibre PERC and Calibre Interactive licenses.
- An LVS rule file.
  - (Optional) A parasitic resistance extraction rule file if you are running a check that requires it. These rules should be included in the LVS rules.
- Know the checks and design flow you want to run.
- Various setup files such as net voltages, layer lists, constraint tables, as required by the checks you plan to run.
- Calibre Interactive PERC is running; see “Invoking Calibre Interactive Tools” on page 64.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- The Rule File Generator GUI uses the XKEYBOARD extension. If you notice that keyboard input to the GUI fields is not correct, make sure your interface to the Linux machine running Calibre uses the XKEYBOARD protocol extension.

## Procedure

1. Click **Rules** on the left panel of Calibre Interactive and enter the desired name for the generated Calibre PERC rule file. This file should not exist; if it does exist, Calibre Interactive automatically assigns a new name so that your previous rule file is not overwritten.
2. Click **Inputs** on the left panel of Calibre Interactive and do the following:
   a. Specify the run and analysis type.
   b. Enter the corresponding input information. The format for the input design must be GDS, OASIS, LEFDEF, or SPICE.

   These values are automatically entered into the Calibre PERC Rule Generator GUI and cannot be changed. The design flow is determined by the run and analysis selections.
3. Click **Outputs** on the left panel of Calibre Interactive and do the following:
   a. Specify the “PERC Report File” name.
   b. Specify the SVDB directory.
4. (Optional) Specify the filename for the Calibre PERC Rule Generator setup file. The setup file saves the settings in the Calibre PERC Rule Generator GUI.
   a. Choose **Setup** > **PERC Options** to open the PERC Options pane.
   b. Click the **RuleGen** tab.
   c. Specify the binary “PERC Rule Generator” setup file (*.prg* file). The default setup file is *percrulegen.prg*.

      If the file specified in the “PERC Rule Generator File” field exists, the settings are automatically loaded when you start the Calibre PERC Rule Generator GUI.

      The updated settings in the Calibre PERC Rule Generator GUI are automatically saved to the specified setup file when the rule file is generated.
5. Choose **Setup** > **Generate PERC Rules** to open the Calibre PERC Rule Generator GUI.

   **Note**

   When invoking the Calibre PERC Rule Generator GUI from Calibre Interactive, settings on the Rules, Inputs, and Outputs pane of Calibre Interactive take precedence over the corresponding settings in the *.prg* file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

6. Choose **Inputs** in the left panel of the Calibre PERC Rule Generator GUI and enter your LVS rule file in the Rules File field.

   The Flow, Format, Layout File, and Top Cell entries are taken from the Calibre Interactive settings and cannot be changed.

7. Choose **Checks** in the left panel and add the desired checks.

   See Steps 6 through 8 in the procedure "Running the Rule File Generator GUI" in the *Calibre PERC User's Manual* for more information.

8. Choose **Outputs** in the left panel and verify the name of the output DFM Database.

   The "Generated Rule File," "SVDB Database," and "PERC Report" entries are read from the Calibre Interactive GUI and cannot be changed.

9. Click **Generate** to generate the rule file.

   The Calibre PERC Rule Generator GUI closes automatically. The Calibre Interactive "PERC Rules File" field on the Rules pane is automatically populated with the generated rule file name.

10. Click **Run PERC**.

### Results

A rule file of the name provided in the "Generated Rule File" field is written to the working directory.

The Calibre PERC Rule Generator GUI settings are automatically saved to the setup file specified on the **RuleGen** tab in the PERC Options pane (see Step 4.)

After the run completes, you can view the results in Calibre RVE for PERC.

# Specifying Outputs for a Calibre PERC Run

Outputs for a Calibre PERC run include the PERC report file and the SVDB. The output for a Calibre PERC-LDL run is a DFM Database, which is specified in the rule file.

Throughout this section, be aware that Calibre Interactive creates new directories if you specify new pathnames. Also, if the option "Read Inputs/Outputs fields when loading rules file" is selected on the **Misc** tab of the Setup Preferences dialog box, relevant rule file settings will populate the Outputs pane when you click the **Load** button on the Rules pane.

Corresponding rule file statements: PERC Report, Mask SVDB Directory



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Procedure

1. Click Outputs to display the Outputs pane, as shown here for a Calibre PERC run.

2. Specify the PERC Report filename.
3. (Optional) Enable the “View report after PERC finishes” and “View Summary Report after PERC finishes” options to automatically display the Calibre PERC reports in a text editor window at the completion of your Calibre PERC run.
4. If the run type is “PERC (Hierarchical)” or “PERC (Flat),” enable the “Create SVDB Database” option and specify the SVDB directory name, as desired. The SVDB database is required if you want to view results in Calibre RVE.

   You do not need to specify an output database if the run type is “PERC-LDL,” since Calibre PERC-LDL requires that the rule file include a DFM Database statement.
5. If desired, check “Start RVE after PERC finishes” to automatically start Calibre RVE at the completion of your Calibre PERC run. If you are creating a runset for a batch run, you may want to enable “Start RVE after batch run” on the **Misc** tab of the Setup Preferences dialog box.

# Running Calibre Interactive PERC

Use the following procedure to start a Calibre PERC run from Calibre Interactive PERC.

### Procedure

1. Enter all necessary information in the Rules, Inputs, and Outputs panes.
2. Choose **Setup** > **PERC Options** and select each of the option tabs to set additional options. All left panel buttons will be green when the necessary information is entered.
3. Specify options on the Run Control pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

In particular, you can specify "Number of hierarchical LVS licenses for PERC," which uses Calibre nmLVS-H licenses for the circuit extraction portion of the run. This adds the -lvs_supplement command line option.

4. If desired, choose **Setup** > **Preferences**, click the **Triggers** tab, and specify triggers. See "Trigger Functions in Calibre Interactive" on page 657.

5. Click **Run PERC** on the left panel of Calibre Interactive.

   If the run uses a source netlist, Calibre Interactive first verifies that the source netlist is complete; the source netlist check is case insensitive. If an internal pre-execution trigger is defined, this is done after the trigger executes. You can turn off the source netlist check by choosing **Setup** > **Preferences** clicking the **Misc** tab, and disabling the setting "Check source netlist is complete before running PERC." See "Miscellaneous Preferences" on page 156 for information on the source netlist check.

   When the run starts, the view switches to the Transcript pane, and any run output, including errors and warnings, is displayed there.

6. View the Transcript pane and examine the results of the run. See "Transcript Pane in Calibre Interactive" on page 99.

**Tip**

You can generate and view a control file without running Calibre by holding down the Ctrl key and clicking **Run PERC**. In addition, the calibre command line for the run is written to the Transcript pane.

## Results

If an error occurs in Calibre Interactive before the Calibre run is started, the error is displayed in a pop-up window or in the Calibre Quality Agent for Calibre Interactive.

When the Calibre run starts, the view switches to the Transcript pane, and any run output, including errors and warnings, is displayed here. See "Transcript Pane in Calibre Interactive".

If "Start RVE after PERC finishes" is checked, the results database is opened in Calibre RVE when the run is complete; see "Using Calibre RVE for PERC" in the *Calibre RVE User's Manual*. In addition, the Calibre Interactive GUI may open a report file in a text viewer when the run completes; see "Viewing and Editing Text Files in Calibre Interactive" on page 125.

## Related Topics

Specifying a Rule File for a Calibre PERC Run

Specifying the PERC Run and Analysis Type

Specifying Outputs for a Calibre PERC Run

Setting Calibre PERC Options

Setting Database and General Options in Calibre Interactive

Run Control, Remote Execution, and Licensing Options



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Run Control, Remote Execution, and Licensing Options

You can control how your Calibre PERC job is run by configuring options related to performance, remote and distributed computing, and license acquisition.

These topics are covered in detail the section "Run Control Setup and Licensing" on page 681.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setting Calibre PERC Options

You can specify options for your Calibre PERC run from the PERC Options panel. These settings generate SVRF specification statements in your control file. If the rule file you load has SVRF statements that correspond to these GUI options, the GUI is populated with the rule file settings. Whatever options you specify in the GUI override the rule file settings.

Some options are similar for all applications—see "Setting Database and General Options in Calibre Interactive" on page 93.

Choose **Setup > PERC Options** to display the PERC Options pane. This also adds the **PERC Options** button to the left panel.

The following options are described:

**Setting Power Supply Options** . . . . . 582
**Aborting Calibre PERC on Sconnect Errors** . . . . . 583
**Specifying PERC Report Options and Result Count** . . . . . 584
**Controlling Logic Gate Recognition in Calibre PERC** . . . . . 585
**Setting Split Gate and Parallel MOS Reduction Options in Calibre PERC** . . . . . 585
**Setting Filter Unused Device Options in Calibre PERC** . . . . . 585
**Changing Trace Property Rules for the Calibre PERC Run** . . . . . 586

## Setting Power Supply Options

You use the **Supply** tab on the PERC Options pane to specify power and ground net names.

Corresponding rule file statements: LVS Power Name, LVS Ground Name



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Procedure

1. Click PERC Options to display the Options pane. If the PERC Options button is not available, choose Setup > PERC Options.
2. Click the Supply tab.
3. Enter the power and ground net names using one of these methods:
   - Enter a space-separated list of net names in the text field.
   - Select "Load from file" and specify a file. The file should be an ASCII text file with the net names specified as a space-separated list or given on separate lines.

     For example, loading the following file:

     ```
     A B C D
     NET_IN1
     NET_OUT2
     ```

     results in the list "A B C D NET_IN1 NET_OUT2" being entered in the text field.

     **Tip**

     The power and ground net names are used to populate the Calibre PERC net type keywords. See the Table PERC Net Type Keywords in the *Calibre PERC User's Manual*.

     You must specify power and ground net names if you invoke LVS logic gate recognition.

## Aborting Calibre PERC on Sconnect Errors

If you use Sconnect statements in your rule file and encounter errors during your run, you can have Calibre PERC abort automatically on soft connection conflicts.

Corresponding rule file statement: LVS Abort On Softchk.

### Procedure

1. Click PERC Options to display the Options pane. If the PERC Options button is not available, choose Setup > PERC Options.
2. Click the Supply tab.
3. Enable "Abort PERC on Softchk errors."

### Related Topics

Sconnect [Standard Verification Rule Format (SVRF) Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Specifying PERC Report Options and Result Count

These settings control the verbosity of the Calibre PERC report and the maximum number of results reported.

Corresponding rule file statements: PERC Report Option and PERC Report Maximum.

## Procedure

1. Click PERC Options to display the Options pane. If the PERC Options button is not available, choose Setup > PERC Options.
2. Click the Report tab.
3. Enable the checkbox “PERC Report Options” to enable GUI control of the PERC Report Option settings. If this checkbox is not enabled, rule file settings are used and GUI settings are ignored.
4. Select the desired settings for PERC Report Options:
   - **Report**:
     - **All Net Types** — Report all net types and type sets carried by a net. The default is to report only the types specified in a given rule check. (option ALL_NET_TYPE)
     - **No Net Types** — Do not report net types or type sets carried by nets. The default is to report net types and type sets. (option NO_NET_TYPE)

     Note: The GUI only allows one of “All Net Types” or “No Net Types” to be selected. If a rule file is loaded which specifies both options, this will be reflected in the GUI; “No Net Types” takes precedence.
   - **Do not report passing cells** — Report only failed cells. The default is to report both passed and failed cells. (option SKIP_PASSED_CELL)
   - **Do not report waived results** — As described. (option SKIP_WAIVED_RESULT)
   - **Do not report pin information** — As described. (option NO_DEVICE_PIN)
   - **Do not report all path voltages** — Only report the minimum and maximum path voltages on a net. Printing of symbolic voltages is not affected. (option MIN_MAX_VOLTAGE_ONLY)
   - **Examine subtitles of all results** — Report results according to the subtitle for the rulecheck. (option TRACK_RESULT_SUBTITLE)
5. Select the maximum result count in “Max. discrepancies printed in report”:
   - **All** — Report all results.
   - **Count** — Enter the maximum allowed result count.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Controlling Logic Gate Recognition in Calibre PERC

These options instruct Calibre to build the layout netlist using either transistors or logic gates.

Corresponding rule file statement: LVS Recognize Gates.

### Procedure

1. Click PERC Options to display the Options pane. If the PERC Options button is not available, choose Setup > PERC Options.
2. Click the Gates tab on the PERC Options pane.
3. Choose the desired gate recognition option. Simple gates use single-level logic (NAND, for example); non-simple gates use multi-level logic (AOI, for example).
4. Decide whether to select the “Mix subtypes” option. If you do not select this option, transistors in series-parallel branches must all have the same subtype.

**Tip**

You must specify power and ground net names if you enable gate recognition. For more information on this topic, refer to “Setting Power Supply Options” on page 582.

# Setting Split Gate and Parallel MOS Reduction Options in Calibre PERC

You can control how split gates are reduced on the **Gates** tab of the Setup PERC Options pane. These controls are the same as for Calibre Interactive nmLVS.

See “Setting Split Gate and Parallel MOS Reduction Options” on page 271 for complete information.

# Setting Filter Unused Device Options in Calibre PERC

You can filter out specific unused devices in your layout and source using the Filter Unused Device Options table in the **Gates** tab of the PERC Options pane.

The procedure is the same as that for Calibre Interactive nmLVS; see “Setting Filter Unused Device Options” on page 272.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Changing Trace Property Rules for the Calibre PERC Run

You can use the **Properties** tab on the PERC Options pane in Calibre Interactive to override the Trace Property statements in your rule file. This may be useful when running the Calibre PERC LDL DRC source-based flow. You can add new statements, edit existing statements, and disable statements.

The Calibre PERC LDL DRC source-based flow is the only Calibre PERC flow in which the Trace Property statement is used in the same manner as for LVS comparison. You should not edit the Trace Property statements for other Calibre PERC flows.

See "Changing Trace Property Rules" on page 274 for instructions on using the **Properties** tab on the PERC Options pane.

## Related Topics

High-Level Checks and Topological DRC [Calibre PERC User's Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 11
# Customization Files

The Customization GUI in Calibre Interactive enables you program controls to customize various settings for your Calibre run. The customization file is a Tcl script with commands in the CUSTOM namespace.


**Customization Overview** . . . . . **588**

**Getting Started with Customization Files** . . . . . **589**

**Customization GUI Operation** . . . . . **591**
- Methods to Specify a Customization File . . . . . 591
- Methods to Open the Customization Settings Dialog Box . . . . . 593
- Customization Settings Dialog Box . . . . . 594
- Saving Customization Settings . . . . . 597
- Customization Overrides . . . . . 598
- Customization GUI Operation in Batch Mode . . . . . 602
- Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive . . . . . 604

**Customization Language** . . . . . **606**
- Tcl/Tk Basics . . . . . 606
- Tcl Lists . . . . . 607
- Tcl/Tk References . . . . . 608

**Customization Commands** . . . . . **610**
- Customization Control Summary . . . . . 610
- CUSTOM::CHECK . . . . . 613
- CUSTOM::DEFINE . . . . . 617
- CUSTOM::LABEL . . . . . 622
- CUSTOM::SEPARATOR . . . . . 623
- CUSTOM::SETWINGEOM . . . . . 624
- CUSTOM::VARIABLE . . . . . 625
- CUSTOM::getCalibreVersionString . . . . . 632
- CUSTOM::getRunsetVarValue . . . . . 633
- CUSTOM::runsetVarExists . . . . . 635
- CUSTOM::setRunsetVarValue . . . . . 636
- CUSTOM::setOKCallback . . . . . 638
- CUSTOM::setOKCallbackError . . . . . 640
- Primary-Dependent Commands . . . . . 640
- Input Type Checking for Customization GUI Entry Fields . . . . . 644
- Config and Cget Options . . . . . 646

**Customization File Examples** . . . . . **648**
- Customization File to Read and Set Calibre Interactive GUI Controls . . . . . 648
- Customization File to Select ERC Rule Checks . . . . . 651





Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

General Customization Example . . . . . 652
Customization File to Load Multiple Customization Files. . . . . 655

# Customization Overview

Customization files allow you to customize several settings that affect Calibre runs. The customization file creates a Customizations Settings dialog box, or Customization GUI, which you display with Calibre Interactive.

You can program controls to do the following:

- Write Variable statements to the Calibre Interactive control file.
- Write #DEFINE and #UNDEFINE pre-processor statements to the Calibre Interactive control file.
- Define Tcl variables in the control file if you are using a TVF rule file.
- Get and set values in the Calibre Interactive GUI using Runset Control Statements in the customization file.

  The customization file is processed after template settings are applied in Calibre Interactive, so the get commands return the GUI settings after template settings are made.
- Save customization settings to the runset, as described in "Saving Customization Settings" on page 597.
- Use CUSTOM::CHECK commands to select the ERC checks which are executed in a Calibre Interactive nmLVS run. ERC [Un]Select Check statements are written to the Calibre Interactive control file. The settings in the customization file initialize the settings in the Select Checks dialog box for Calibre Interactive nmLVS (see "Selecting ERC Checks" on page 284).

  Note: The Check Selection Recipe is the preferred method for selecting the rule checks that are evaluated in a Calibre Interactive nmDRC or Calibre Interactive DFM run; CUSTOM::CHECK controls are not recommended for these runs.

Customization files are written in Tcl/Tk and use commands in the CUSTOM namespace, as described in the sections "Customization Language" on page 606 and "Customization Commands" on page 610.

An example of a Customizations Settings dialog box is shown in Figure 11-1; the code for this example is in "General Customization Example" on page 652.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 11-1. Customization Settings Dialog Box**

# Getting Started with Customization Files

The customization file is specified in the **Startup** tab of the Setup Preferences dialog box.

## Prerequisites

- A customization file. See "Customization Language" on page 606, "Customization Commands" on page 610, and "Customization File Examples" on page 648 for information.

## Procedure

1. In Calibre Interactive, choose **Setup** > **Preferences** to open the Setup Preferences dialog box.

   a. Click the **Startup** tab.

   b. Enable the "Customization File" checkbox and specify the filename of the customization file.

   c. (Optional) Enable "Show customization dialog after opening runset" to have the Customization GUI open automatically after a runset is loaded.

   d. Click **OK** to close the Setup Preferences dialog box.

   The **Customization** button is now displayed in the left panel of Calibre Interactive.

2. Click the **Customization** button to open the Customization Settings dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Make changes as desired in the Customization GUI.

3. Click **OK** to apply the customization settings and close the Customization Settings dialog box.

   The customization settings are used in the next Calibre run unless overridden by later changes in the Calibre Interactive GUI.

4. Choose **File > Save Runset** to save the runset.

   If you made changes to the Customization GUI, the changes to the Customization GUI are saved to the runset as customization overrides. The override settings in the runset are applied the next time you load the runset and open the Customization GUI.

## Related Topics

Customization GUI Operation

Customization Settings Dialog Box

Customization GUI Operation in Batch Mode

Customization Overrides



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Customization GUI Operation

The topics in this section provide full details on each aspect of specifying and using the Customization GUI. Most actions can be performed in more than one way.

See "Getting Started with Customization Files" on page 589 for a simple procedure to load and use a customization file.

**Methods to Specify a Customization File** . . . . . 591
**Methods to Open the Customization Settings Dialog Box** . . . . . 593
**Customization Settings Dialog Box** . . . . . 594
**Saving Customization Settings** . . . . . 597
**Customization Overrides** . . . . . 598
**Customization GUI Operation in Batch Mode** . . . . . 602
**Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive** 604

# Methods to Specify a Customization File

The customization file can be specified using the Calibre Interactive GUI, with an environment variable, or on the command line.

## Requirements

- A customization file. See "Customization Language" on page 606 and "Customization Commands" on page 610.

### Methods

Choose one of the following methods to specify a customization file:

- Environment variable
- In the Calibre Interactive GUI
- Command line (interactive mode)
- Command line (Calibre Interactive batch mode)

## Environment variable

Set the MGC_CALIBRE_CUSTOMIZATION_FILE environment variable to the filename of the customization file.

## In the Calibre Interactive GUI

1. Choose **Setup** > **Preferences** to open the Setup Preferences dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Click the **Startup** tab.
3. Enable the "Customization File" checkbox and specify the filename of the customization file.
4. Choose **File** > **Save Runset** to save the settings to the runset.

   If this runset is used to run Calibre Interactive in batch mode, the initial settings in the customization file are applied to the Calibre run; the Customization GUI does *not* open. If the runset contains customization override settings, those take precedence over the settings in the customization file.

## Command line (interactive mode)

Specify a customization file from the command line with the -custom filename option. This option only works when you invoke a particular component of Calibre Interactive (for example, LVS), as shown in the following example:

```
calibre -gui -lvs -custom my_custom (correct)

   XX calibre -gui -custom my_custom (incorrect)
```

## Command line (Calibre Interactive batch mode)

Specify a customization file in Calibre Interactive batch mode with either the -customgui or -custom command line options, depending on whether you want the customization dialog box to open or not. See "Customization GUI Operation in Batch Mode" on page 602 for details.

**Note**

The customization file specified on the command line with -custom or -customgui overrides a customization file set with the environment variable MGC_CALIBRE_CUSTOMIZATION_FILE.

**Tip**

You can also use the **Load** button in an open Customization Settings dialog box to specify a new customization file. See "Specify and load a different customization file from the Customization GUI" on page 594.

## Results

When you invoke Calibre Interactive with a customization file, a **Customization** button appears in the left panel of Calibre Interactive. When you click the **Customization** button, the Customizations Settings dialog box is displayed with the custom GUI controls for that tool.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

Choose Setup > Preferences, click the Startup tab, then enable “Show customization dialog after opening runset” to have the customization dialog box open automatically after loading a runset.

To disable changes to the customization settings, choose **Setup** > **Preferences**, the **Disable** tab, and enable “Disable customization changes.”

### Related Topics

Customization Settings Dialog Box

Customization Control Summary

Customization File to Load Multiple Customization Files

Running Calibre Interactive in Batch Mode

# Methods to Open the Customization Settings Dialog Box

You can open the Customization Settings dialog box automatically when the runset is opened, manually with the Customization button, or during a Calibre Interactive batch run.

You can also load a different customization file with the **Load** button in the Customization GUI.

### Requirements

- A customization file is specified as described in “Methods to Specify a Customization File” on page 591.
- Calibre Interactive is open.

#### Methods

Choose one of the following methods to open the Customizations Setting dialog box:

- Automatically open the Customization GUI when the runset is loaded
- Manually open the Customization GUI
- Open the Customization GUI in Calibre Interactive batch mode
- Specify and load a different customization file from the Customization GUI

### Automatically open the Customization GUI when the runset is loaded

1. Choose **Setup** > **Preferences** from the main menu, then click the **Startup** tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Enable “Show customization dialog after opening runset.”
3. Save the runset with **File > Save Runset**.

   The next time you load the runset, the Customization GUI is opened automatically.

### Manually open the Customization GUI

- Click the **Customization** button on the left panel of Calibre Interactive.

  If the **Customization** button is not present, make sure a customization file is specified.

### Open the Customization GUI in Calibre Interactive batch mode

- Specify the Customization GUI on the command line with the options “-customgui *filename* -batch”. See “Customization GUI Operation in Batch Mode” on page 602 for details.

### Specify and load a different customization file from the Customization GUI

1. Click the **Customization** button on the left panel of Calibre Interactive.
2. Click the **Load** button at the bottom of the dialog box.
3. Specify a new customization file.
4. Click **OK**.

   The new customization file is loaded and the settings in the previously loaded Customization GUI are discarded. Customization override settings in the runset are *not* applied when the new customization file is opened using the **Load** button.

   The setting for “Customization File” on the **Startup** tab of the Setup Preferences dialog box is updated with the new customization file.

### Related Topics

Customization Settings Dialog Box

Methods to Specify a Customization File

Customization GUI Operation in Batch Mode

Customization Overrides

## Customization Settings Dialog Box

To access: **Customization** button, or see “Methods to Open the Customization Settings Dialog Box” on page 593 for methods to open the Customization Settings dialog box automatically or in batch mode.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The Customization Settings dialog box, or Customization GUI, contains the customization controls specified in the customization file. Several action buttons, such as Load and Reset, are also included.

## Description

The customization file is read the first time Customization GUI is opened and not read again unless the customization file changes or a new customization file is loaded with the **Load** button. In particular, -initval settings in customization commands are only processed the first time the customization file is read.

**Tip**

You can disable changes to the customization settings with “Disable customization changes” on the **Disable** tab of the Setup Preference dialog box (**Setup** > **Preferences**).

Table 11-1 describes the function of the action buttons in the Customization Settings dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Buttons

**Table 11-1. Action Buttons in Customization Settings Dialog Box**

| Action Button | Description |
|---|---|
| OK | Applies the current settings and closes the dialog box. The new settings are retained within the current Calibre Interactive session. The new settings are used in the next Calibre run unless overridden by later GUI actions.<br>If you make changes to the Customization GUI and then save a runset, the changes to the Customization GUI are saved to the runset as customization overrides. The override settings in the runset are applied the next time you load the runset and open the Customization GUI. See "Saving Customization Settings" on page 597 for complete details. |
| Cancel | Cancel any changes to the customization dialog box and close the dialog box. |
| Reset | Reload the current customization file. Changed settings are discarded and customization override values in the runset are *not* applied. The dialog box remains open. |
| Load | Opens a dialog box to load a new customization file. Customization override values in the runset are *not* applied when the new customization file is processed.<br>Loading a new customization file causes the current customization settings to be ignored. The setting for "Customization File" on the **Startup** tab of the Setup Preferences dialog box is updated with the new customization file. |
| Exit<br>(only present for batch runs) | Closes the Customization Settings dialog box without running Calibre Interactive. |

## Related Topics

Customization Overrides

Customization GUI Operation in Batch Mode

Saving Customization Settings

Methods to Specify a Customization File

Methods to Open the Customization Settings Dialog Box



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Saving Customization Settings

Settings in the Customization GUI are kept internally during the current Calibre Interactive session when you click **OK** in the Customization Settings dialog box. Customization settings are saved to the runset as customization overrides when you save a runset.

## Prerequisites

- A customization file is specified; see "Methods to Specify a Customization File" on page 591.

## Procedure

1. Click the **Customization** button on the left panel of Calibre Interactive to open the Customization Settings dialog box.
2. Adjust settings as desired.
3. Click **OK** to apply the settings and exit the dialog box.

   The new settings are used in the next Calibre run unless overridden by later GUI actions. The new settings are retained within the current Calibre Interactive session.
4. Choose **File > Save Runset** to save a runset with the new customization settings.

   The new customization settings are saved as customization overrides to the runset option cmnCustomFileOverrideValues. The next time you load the runset, the customization override settings are used for the initial settings of the customization controls.

## Related Topics

Saving Customization GUI Settings in Calibre Interactive Batch Mode Runs

Customization Overrides



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Customization Overrides

Customization overrides are the changed settings in the Customization GUI. The changed settings are saved in the runset option cmnCustomFileOverrideValues and then applied the next time you load the runset. Customization overrides let you save and automatically restore settings you make in a Customization GUI.

The syntax for the runset option is described in the section "cmnCustomFileOverrideValues Syntax" on page 599. However, it usually not necessary to know the syntax, since Calibre Interactive saves the runset with the correct value.

**cmnCustomFileOverrideValues Syntax** . . . . . **599**



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# cmnCustomFileOverrideValues Syntax

This Calibre Interactive runset option saves Customization GUI settings that are changed from the initial settings in the customization file. The new settings, or overrides, are applied the next time the runset is loaded. Only changed settings are saved to cmnCustomFileOverrideValues.

## Usage

***cmnCustomFileOverrideValues: {*controlType tool nameType name select value*}** …

## Parameters

- ***controlType***

  The type of customization control.

  Allowed values: DEFINE VARIABLE CHECK

- ***tool***

  The Calibre Interactive application that is running.

  Allowed values: DRC LVS PERC PEX DFM 3DSTACK

- ***nameType name***

  Parameter set that specifies the control name and naming method.

  - ***nameType*** — Specifies naming method for the control.

    Allowed values of ***nameType***: name prompt choices

  - ***name*** — Specifies the name.

  The values of ***nameType*** and ***name*** are determined with the following logic:

  1. If the control was created with the -name option, ***nameType*** is set to name and ***name*** is the value of the -name option.

  2. Else if the control was created with the -prompt option, ***nameType*** is set to prompt and ***name*** is the value of the -prompt option.

  3. Else the control was created with the -choices option, and ***nameType*** is set to choices and ***name*** is the value of the -choices option.

  For DEFINE and VARIABLE controls, name is the only possible value for ***nameType***.

- ***select***

  Gives the on/off (or select) state of the Boolean checkbox. If no checkbox is displayed for the control, the value is 1.

- ***value***

  The current setting of the control. This can be the selection from the -choices list or the entry in the text field. An empty string is indicated with braces ({}).



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Examples

### Example 1: Override for CUSTOM::DEFINE control

The following custom control:

```
CUSTOM::DEFINE -name process -prompt "Process" -choices {""} -initval 20
```

results in the control shown below:

1.a. Change value — If you change the assigned value for the control, then save the runset, the runset has this entry for the customization override option:

```
*cmnCustomFileOverrideValues: {DEFINE DRC name process 1 40}
```

1.b. Disable control — If you disable the checkbox, then save the runset, the runset has this entry for the customization override option:

```
*cmnCustomFileOverrideValues: {DEFINE DRC name process 0 20}
```

### Example 2: Override for CUSTOM::VARIABLE control with choices list

The following custom control:

```
CUSTOM::VARIABLE -name "var_constraint" \
   -choices { 1 2 3} -prompt "Variable for constraint"
```

results in the control shown below:

If you select a different choice from the list, then save the runset, the runset has this entry for the customization override option:

```
*cmnCustomFileOverrideValues: {VARIABLE DRC name var_constraint 1 3}
```

### Example 3: Override for CUSTOM::CHECK control with checkbox

The following custom control:

```
CUSTOM::CHECK -name all_erc -prompt "All ERC checks" -tool LVS
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

results in the control shown below:

If you unselect the control, then save the runset, the runset has this entry for the customization override option:

```
*cmnCustomFileOverrideValues: {CHECK LVS name all_erc 0 all_erc}
```

### Example 4: Override for CUSTOM::CHECK control with choices list

The following custom control:

```
CUSTOM::CHECK -choices {ERC_1 ERC_2 ERC_3} \
   -prompt "Single ERC checks" -tool LVS
```

results in the control shown below:

If you select a different choice from the list, then save the runset, the runset has this entry for the customization override option:

```
*cmnCustomFileOverrideValues: {CHECK LVS prompt {Single ERC checks}
   1 ERC_3}
```

where nameType is given the value prompt in cmnCustomFileOverrideValues.

Alternatively, if the control is created without the -prompt option, as follows:

```
CUSTOM::CHECK -choices {ERC_1 ERC_2 ERC_3} -tool LVS
```

Then nameType is given the value choices in cmnCustomFileOverrideValues, as shown here:

```
*cmnCustomFileOverrideValues: {CHECK LVS choices {ERC_1 ERC_2 ERC_3}
   1 ERC_3}
```

## Related Topics

Saving Customization Settings

Saving Customization GUI Settings in Calibre Interactive Batch Mode Runs



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Customization GUI Operation in Batch Mode

Customization files can be used in Calibre Interactive batch mode. The methods for using the Customization GUI differ from the usage in interactive mode.

**Methods to Specify the Customization File for Calibre Interactive Batch Mode . . . . . 602**
**Saving Customization GUI Settings in Calibre Interactive Batch Mode Runs . . . . . 603**

## Methods to Specify the Customization File for Calibre Interactive Batch Mode

You can specify the customization file for a batch mode run in the runset or on the command line.

### Requirements

- A runset for the batch mode run.
- A customization file. See "Customization Language" on page 606 and "Customization Commands" on page 610.

#### Methods

Choose one of the following methods to specify a customization file for a Calibre Interactive batch mode run:

- Runset specification (Customization GUI does not open)
- Command line specification with -customgui (Customization GUI opens)
- Command line specification with -custom (Customization GUI does not open)

### Runset specification (Customization GUI does not open)

1. In Calibre Interactive, choose **Setup** > **Preferences** from the main menu, then click the **Startup** tab.
2. Enable the "Customization File" checkbox and specify the filename of the customization file.
3. Save the runset with **File** > **Save Runset**.
4. Run Calibre Interactive in batch mode; for example:

   ```
   calibre -gui -lvs -runset my_runset -batch
   ```

   When the runset is used to run Calibre Interactive in batch mode, the initial settings in the customization file are applied to the Calibre run; the Customization GUI does *not* open. If the runset contains customization override settings, those take precedence over



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

the initial settings in the customization file. If there is a callback procedure specified with CUSTOM::setOKCallback, the callback procedure is executed automatically.

### Command line specification with -customgui (Customization GUI opens)

- Use the "-customgui filename -batch" options on the command line. For example:

  ```
  calibre -gui -lvs -runset my_runset -customgui my_custom -batch
  ```

  The Customization Settings dialog box appears at the beginning of the Calibre run. Once the customization selections are made and the dialog box is closed, the Calibre run continues in batch mode, without opening the Calibre Interactive GUI.

  Also see "Saving Customization GUI Settings in Calibre Interactive Batch Mode Runs" on page 603.

### Command line specification with -custom (Customization GUI does not open)

- Use the "-custom filename -batch" options on the command line. For example:

  ```
  calibre -gui -lvs -runset my_runset -custom my_custom -batch
  ```

  The initial settings in the customization file are applied and used in the batch run; the Customization Settings dialog box does *not* open. If the runset contains customization override settings, those take precedence over the settings in the customization file. If there is a callback procedure specified with CUSTOM::setOKCallback, the callback procedure is executed automatically.

## Saving Customization GUI Settings in Calibre Interactive Batch Mode Runs

The Customization GUI is displayed in Calibre Interactive batch mode runs invoked with the -customgui command-line option. You can control whether or not changes to the Customization GUI are saved to the runset. The settings on the **Misc** tab of the Setup Preferences dialog box for saving a runset also apply to Calibre Interactive batch runs.

### Prerequisites

- A runset for the batch mode run.
- A customization file. See "Customization Language" on page 606 and "Customization Commands" on page 610.

### Procedure

1. Load the runset for the batch mode run in Calibre Interactive.
2. Choose **Setup** > **Preferences** to open the Setup Preferences dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Click the **Misc** tab and make selections as follows:
   - Prompt to save a runset if customization settings change:

     Enable “Prompt to save changes before closing runset.”
   - Save the runset without prompting:

     Disable “Prompt to save changes before closing runset.”

     Enable “Save changed settings before closing runset.”
   - Do not save the runset and do not prompt:

     Disable “Prompt to save changes before closing runset.”

     Disable “Save changed settings before closing runset.”

### Results

If you make changes to the Customization GUI during a batch mode run, saving the runset is controlled by the settings listed in Step 3. If “Prompt to save changes before closing runset” is enabled and you make changes to the Customization GUI, the prompt to save the runset is displayed immediately after closing the Customization GUI.

## Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive

The CUSTOM::CHECK controls in the Customization GUI interact with the check selection methods for Calibre Interactive nmDRC, Calibre Interactive DFM, and Calibre Interactive nmLVS. It is important to know the precedence of the various check selection methods.

The precedence rules are described in the following two sections:

“Calibre Interactive nmDRC and Calibre Interactive DFM Check Selection” on page 604

“Calibre Interactive nmLVS Check Selection” on page 605

### Calibre Interactive nmDRC and Calibre Interactive DFM Check Selection

A Check Selection Recipe is used in Calibre Interactive nmDRC and Calibre Interactive DFM to select the checks that are run; this can potentially cause a conflict if CUSTOM::CHECK controls in a Customization GUI are also used.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Caution**

The Check Selection Recipe is the preferred method for selecting the rule checks that are evaluated in a Calibre Interactive nmDRC or Calibre Interactive DFM run.

If you do use CUSTOM::CHECK controls in a customization file, the default built-in check recipe "Checks selected in the rules file" should be used to ensure that the check selections in the Customization GUI are obeyed.

The following two statements govern the interaction of CUSTOM::CHECK controls in the Customization GUI with the Check Selection Recipe:

- Settings from CUSTOM::CHECK controls in a customization file *only* affect the evaluation of the "Checks selected in the rules file" expression in a Check Selection Recipe.
- Settings in the Check Selection Recipe take precedence over settings made by the Customization GUI if there is a conflict.

If a CUSTOM::CHECK control in the Customization GUI enables or disables a check in the rule file, this is reflected in the Check Selection Recipe Editor when the "Checks selected in the rules file" expression is displayed. All other recipe expressions ignore the CUSTOM::CHECK controls in the Customization GUI. Therefore, unexpected results may occur if a check recipe with expressions other than "Checks selected in the rules file" is used together with CUSTOM::CHECK controls in a Customization GUI.

## Calibre Interactive nmLVS Check Selection

Calibre Interactive nmLVS uses the Select Checks Dialog Box to select the ERC checks that are executed during a run. CUSTOM::CHECK controls in a Customization GUI initialize the selection status of checks in the Select Checks dialog box. Changes made in the Select Checks dialog box override the initial selections made by the CUSTOM::CHECK controls.

Check selections in the Select Checks Dialog Box are not saved to the runset, so selecting ERC checks with a customization file is a useful alternative for Calibre Interactive nmLVS runs. See "Customization File to Select ERC Rule Checks" on page 651 and "Rule Check Selection in Calibre Interactive Batch Runs" on page 103 for more information.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Customization Language

Customization files use Tcl/Tk, with API commands defined in the CUSTOM namespace. The CUSTOM commands control the elements in the customization dialog. The customization file is sourced by Calibre Interactive as a Tcl file; therefore all Tcl/Tk statements and constructs are available to the customization file developer.

Starting with Calibre version 2019.2, Calibre uses Tcl 8.6. See "Tcl Library in Calibre" in the *Calibre Administrator's Guide* for other general information.

You can write a customization file even if you do not already know Tcl/Tk. This section covers some basic concepts in Tcl and provides a list of references.

**Tcl/Tk Basics** . . . . . 606
**Tcl Lists** . . . . . 607
**Tcl/Tk References** . . . . . 608

## Tcl/Tk Basics

If you are not familiar with Tcl coding, this section described basics, such as command syntax and special characters.

Tcl characteristics:

- **Tcl is case sensitive** — The commands in the CUSTOM namespace must be used as shown.
- **Statements can span multiple lines** — The statement continuation character for multiple line statements is the backslash (\) character, which must terminate each continued line. There cannot be any whitespace characters after the backslash.
- **Many characters have special meanings in Tcl** — For a complete list of special characters, consult the "Tcl/Tk References" on page 608. The following are the special characters used in this chapter:

**Table 11-2. Tcl Special Characters**

| Character | Meaning |
|---|---|
| ; | The semicolon terminates the previous command, allowing you to place more than one command on the same line. |
| \ | The backslash causes backslash substitution, or, when within double quotes, disables substitution for the single character immediately following the backslash. Be careful not to have whitespace on the same line after a backslash that terminates a line, as this can cause Tcl interpretation errors. |
| \n | The backslash with the letter "n" creates a new line. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 11-2. Tcl Special Characters (cont.)**

| Character | Meaning |
|---|---|
| # | The pound sign indicates the start of a comment. |
| $ | The dollar sign in front of a variable name instructs the Tcl interpreter to access the value stored in the variable. |
| [ ] | Brackets cause command substitution, instructing the Tcl interpreter to treat everything within the brackets as the result of a command. Command substitution can be nested within words. |
| { } | Braces instruct the Tcl interpreter to treat the enclosed words as a single string. The Tcl interpreter accepts the string as is, without performing any variable substitution or evaluation. |
| " " | Quotes instruct the Tcl interpreter to treat the enclosed words as a single string. However, if the Tcl interpreter encounters variables or commands within the string in quotes, it first evaluates the variables and commands. |

**Note**

Braces and square brackets are used in both syntax definitions and Tcl constructs. If a syntax definition includes a Tcl construct with brackets or braces, the Tcl elements are enclosed in single quotes to indicate a literal element. For example, the following syntax indicates the square brackets are literal within a syntax definition:

'[' list a b c ']'

- **Comments are a type of command** — In Tcl, comments are indicated by “#”. It must be the first non-whitespace character of the command; nothing after it on the line is acted upon.
- **Tcl/Tk Troubleshooting** — Each CUSTOM command returns a Tk widget. You can use the standard Tk config and cget options with the returned command.

# Tcl Lists

The list is a basic data structure in Tcl, and is used in several customization commands. A Tcl list is an ordered collection of numbers, strings, or other lists.

A Tcl list of single elements can be specified in these equivalent ways:

```
"a b c d"
{a b c d}
[list a b c d]
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

A list of lists can be specified in these equivalent ways:

```
{{a1 a2} {b1 b2} {c1 c2} {d1 d2}}
[list [list a1 a2] [list b1 b2] [list c1 c2] [list d1 d2]]
```

# Tcl/Tk References

Siemens EDA recommends that you take advantage of the excellent books and websites on the language. The list that follows is provided to give you a place to start in your search for the reference material that works best for you. It is not an endorsement of any book or website. While every attempt has been made to ensure that the URLs listed here point to active websites, there is no guarantee of this.

## Books

**Practical Programming in Tcl and Tk (4th Edition)**

Brent B. Welch, Jeffrey Hobbs, Brent Welch, *Practical Programming in Tcl and Tk*, 4th Edition, Prentice Hall PTR, 2003

**Tcl/Tk in a Nutshell**

Paul Raines, Jeff Tranter, *Tcl/Tk in a Nutshell*, O'Reilly and Associates, Inc., 1999

**Tcl and the Tk Toolkit**

John K. Ousterhout, *Tcl and the Tk Toolkit*, Addison-Wesley Professional (1994)

**Tcl/Tk Tools**

Mark Harrison,*Tcl/Tk Tools*, O'Reilly (1997)

## Websites

Tcl Developer Xchange

http://www.tcl.tk/scripting/

Tcl Tutor

`http://www.msen.com/~clif/TclTutor.html`

Beginning Tcl

`http://wiki.tcl.tk/298`

The Tcler's Wiki

`http://wiki.tcl.tk`



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## ActiveTcl

http://www.activestate.com/Products/ActiveTcl/



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Customization Commands

The customization commands allow you to construct the Customization GUI.

**Customization Control Summary** . . . . . 610
**CUSTOM::CHECK** . . . . . 613
**CUSTOM::DEFINE** . . . . . 617
**CUSTOM::LABEL** . . . . . 622
**CUSTOM::SEPARATOR** . . . . . 623
**CUSTOM::SETWINGEOM** . . . . . 624
**CUSTOM::VARIABLE** . . . . . 625
**CUSTOM::getCalibreVersionString** . . . . . 632
**CUSTOM::getRunsetVarValue** . . . . . 633
**CUSTOM::runsetVarExists** . . . . . 635
**CUSTOM::setRunsetVarValue** . . . . . 636
**CUSTOM::setOKCallback** . . . . . 638
**CUSTOM::setOKCallbackError** . . . . . 640
**Primary-Dependent Commands** . . . . . 640
**Input Type Checking for Customization GUI Entry Fields** . . . . . 644
**Config and Cget Options** . . . . . 646

## Customization Control Summary

The customization commands are functionally divided into three groups: control file statements, runset control statements, and Customization GUI appearance statements.

**Table 11-3. Customization Command Types**

| Customization Command Type | Description |
|---|---|
| Control File Statements | Adds SVRF or TVF statements to the control file created by Calibre Interactive. |
| Runset Control Statements | Gets and sets the value of the fields in the Calibre Interactive GUI and defines a callback procedure. |
| Customization GUI Appearance Statements | Define labels, separators, and sets the size of the dialog box. |

The syntax conventions for the command descriptions are given in “Syntax Conventions” on page 29.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

Braces and square brackets are used in both syntax definitions and Tcl constructs. If a syntax definition includes a Tcl construct with brackets or braces, the Tcl elements are enclosed in single quotes to indicate a literal element. For example, the following syntax indicates the square brackets are literal within a syntax definition.

'[' list a b c ']'

## Control File Statements

The control file statements add SVRF or TVF statements to the control file created by Calibre Interactive. See "About Control Files" on page 46 for more information.

**Table 11-4. Control File Statements in the Customization File**

| Command | Description |
|---|---|
| CUSTOM::CHECK | Initializes the check selection status in the Select Checks Dialog Box for ERC checks in Calibre Interactive nmLVS and places ERC [Un]Select Check statements in the control file.<br>The Check Selection Recipe is the preferred method for selecting the rule checks that are evaluated in a Calibre Interactive nmDRC or Calibre Interactive DFM run. See "Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive" on page 604 for important information if you are using CUSTOM::CHECK controls with Calibre Interactive nmDRC or Calibre Interactive DFM. |
| CUSTOM::DEFINE | Places a #DEFINE or #UNDEFINE pre-processor directive in the control file for SVRF rule files or defines a Tcl variable in the control file when the -tvf argument is specified. |
| CUSTOM::VARIABLE | Places a Variable statement in the control file for SVRF rule files or defines a Tcl variable in the control file when the -tvf argument is specified. |

## Runset Control Statements

The runset control statements get and set the value of the fields in the Calibre Interactive GUI and define a callback procedure.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 11-5. Runset Control Statements in the Customization File**

| Command | Description |
|---|---|
| CUSTOM::getRunsetVarValue | Gets the Calibre Interactive GUI setting for the specified control. |
| CUSTOM::getCalibreVersionString | Returns the current Calibre version string, for example, v2012.4_0.28. |
| CUSTOM::runsetVarExists | Returns true or false depending on whether the specified control name corresponds to a valid Calibre Interactive GUI control. |
| CUSTOM::setRunsetVarValue | Sets the specified Calibre Interactive GUI control to the given value. |
| CUSTOM::setOKCallback | Specifies a Tcl proc that is executed when you click the **OK** button in the Customization GUI. |
| CUSTOM::setOKCallbackError | Specifies a message to display if the OKcallback function returns an error. |

## Customization GUI Appearance Statements

The Customization GUI Appearance statements affect the appearance of the Customization GUI.

**Table 11-6. Customization GUI Appearance Statements**

| Command | Description |
|---|---|
| CUSTOM::LABEL | Inserts a label in the customization dialog box. |
| CUSTOM::SEPARATOR | Inserts a separator in the customization dialog box. |
| CUSTOM::SETWINGEOM | Resizes and relocates the customization dialog box. |

You can specify a primary-dependent relationship between two CUSTOM commands; this is described in the section "Primary-Dependent Commands" on page 640.

The Customizations Settings dialog box includes control buttons such as **Load** and **Reset** by default. See the section "Customization Settings Dialog Box" on page 594 for a description of these control buttons.

The standard Tcl/Tk command options config and cget are available with the CUSTOM commands. These options are useful for troubleshooting. They are described the section "Config and Cget Options" on page 646.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::CHECK

Command type: Customization command for Calibre Interactive.

In Calibre Interactive nmLVS, the CUSTOM::CHECK command initializes the check selection status in the Select Checks dialog box for ERC checks in Calibre Interactive nmLVS.

**Caution**

In Calibre Interactive nmDRC or Calibre Interactive DFM, the Check Selection Recipe is the preferred method for selecting the rule checks that are evaluated.

## Usage

**CUSTOM::CHECK** {**-name** ***var_name*** | **-choices** ***choices_list*** [-initval *init_value*]}
[-prompt *prompt_text*] [-tooltip *tooltip_string*]
[-boolean {0|1}] [-select {0|1}] [-enable {0|1}] [-display {0|1}]
[-tool { DRC | LVS | DFM | *tool_list* }]
[ {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec* ]

## Arguments

- **-name** ***var_name***

  A required argument that specifies the name of the rule check. Either the **-name** or **-choices** keyword must be specified.

  A rule check name may be preceded by an exclamation point (!) to specify that the check is unselected. The exclamation point (!) is also called the negation character.

  A checkbox is displayed unless “-boolean 0” is specified.

- **-choices** ***choices_list***

  A required argument that specify a list of choices, where the parameter ***choices_list*** is a Tcl list of rule check names. Either the **-name** or **-choices** keyword must be specified.

  A rule check name may be preceded by an exclamation point (!) to specify that the check is unselected.

  A checkbox is displayed only if “-boolean 1” is specified.

  If a multi-element list is specified with the **-choices** switch, a button with a dropdown list is displayed. The values are displayed as mutually exclusive radio buttons.

  The ***choices_list*** can be specified as one of the following:

  - A Tcl list of single elements, where each element is the name of a rule check. For a list of single elements, the elements themselves are displayed as the label for the radio buttons.
  - A Tcl list of two-element sublists. The first element is displayed as a label for the radio button and the second element is the name of the rule check.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For example:

```
{{"Check A" A} {"Not check C" !C}}
```

where the braces are literal to indicate a Tcl list.

- -initval *init_value*

  An optional argument that specifies the default value from the -choices list.

  The *init_value* is read and applied the first time the Customization GUI is opened or if the customization file changes; however, *init_value* is overridden with the runset value if the runset option cmnCustomFileOverrideValues specifies a value for the control.

  The *init_value* is always applied if the Customization GUI is opened using the **Load** button in the Customization Settings dialog box; the runset value specified with the runset option cmnCustomFileOverrideValues is not used in this case.

- -prompt *prompt_text*

  Places the string *prompt_text* next to the button or checkbox.

- -tooltip *tooltip_string*

  Specifies a tooltip *tooltip_string* for the control. Line breaks in *tooltip_string* are preserved in the generated tooltip.

- -boolean {0|1}

  Controls the display of the checkbox, where 1 specifies to display the checkbox. See the **-name** and **-choices** options for the behavior if this option is not specified.

- -select {0|1}

  Controls the initial setting of the -boolean checkbox. The default value is 1 (true) if this option is not specified.

- -enable {0|1}

  Controls whether the -boolean checkbox is active (responds to user input). The default value is 1 (true) if this option is not specified.

- -display {0|1}

  Controls whether the control is displayed in the customization dialog box. The default value is 1 (true) if this option is not specified.

- -tool { DRC | LVS | DFM | *tool_list* }

  If included, the statement applies only to the specified tool(s). To specify more than one tool, provide a Tcl list; for example: {DRC DFM}

  If this option is not specified, the command applies to all supported tools.

- {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec*

  An optional keyword and parameter set that specifies the current control as a dependent control to another primary control in the customization file. In other words, this option



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

specifies that the appearance of this GUI object in the customization window depends on the value of another control referenced by *master_spec* or *expr_spec*.

See "Primary-Dependent Commands" on page 640 for a complete description of using the -master option.

## Description

In Calibre Interactive nmLVS, the CUSTOM::CHECK statement initializes the check selection status in the Select Checks Dialog Box for ERC checks. The CUSTOM::CHECK statement includes options to display a checkbox, a prompt, and a button with a dropdown list. The appropriate [Un]Select Check statement for the application is written to the Calibre Interactive control file.

**Caution**

When using Calibre Interactive nmDRC or Calibre Interactive DFM, the Check Selection Recipe is the preferred method for selecting the rule checks that are evaluated during the run.

However, if you do use CUSTOM::CHECK controls in a Customization GUI for Calibre Interactive nmDRC or DFM, make sure that the "Checks selected in the rules file" built-in check recipe is selected, as described in "Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive" on page 604.

### Rule Check Selection

Table 11-7 shows the selection status of a check for the possible combinations of checkbox state and the presence of the negation symbol (!) for the check.

**Table 11-7. Check State Table for CUSTOM::CHECK**

| -boolean | -selected | Check | Check state |
|---|---|---|---|
| 1 | 1 | A | Selected |
| 1 | 1 | !A | Unselected |
| 1 | 0 | A | Unselected |
| 1 | 0 | !A | Unselected |

## Examples

The following example selects the erc_check rule check. It places a checkbox in the Customization GUI for Calibre Interactive nmLVS and displays an "ERC Checks" prompt next to the checkbox.

```
CUSTOM::CHECK -name erc_check -boolean 1 -prompt "ERC Checks" \
      -tool LVS
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

This creates the following checkbox, with “ERC Checks” as the prompt:

## Related Topics

Customization File to Select ERC Rule Checks

Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::DEFINE

Command type: Customization command for Calibre Interactive.

Inserts a #DEFINE or #UNDEFINE pre-processor directive in the Calibre Interactive control file, or a Tcl variable in the case of TVF rule files.

## Usage

**CUSTOM::DEFINE -name *var_name***
[-choices {“” | *choices_list* }]
[-initval *init_value*]
[-prompt *prompt_text*] [-tooltip *tooltip_string*] [-multiline {0|1}]
[-boolean {0 | 1} [-select {0|1}] [-enable {0|1}]] [-display {0|1}]
[-tool { DRC | LVS | DFM | PEX | PERC | *tool_list*}] [-tvf {0|1}]
[-vtype *type*] [-vcmd ‘[’ list *v_proc* ‘]’]
[ {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec* ]

## Arguments

- **-name *var_name***

  A required option and parameter set, where ***var_name*** gives the name of the variable output to the control file.

- -choices {“” | *choices_list* }

  Specifies a value for the variable being defined. A text entry field or a dropdown list is displayed.

  - “” (empty string) — Display a text entry field for specifying the value of the #define variable.
  - *choices_list* — Display a control button with a dropdown list of choices for the value of the variable. The values are displayed as mutually exclusive radio buttons.

    The *choices_list* can be specified in the following ways:

    - A Tcl list of single elements. For a list of single elements, the values themselves are displayed in the dropdown list of the control button.
    - A Tcl list of two-element sublists. The first element is displayed in the fields of the dropdown list and the second element is the actual value. For example:

```
[list [list "Wide metal process" wmp] [list "Process 7lm" lm7]]
```

      where the brackets are required to indicate a Tcl list.

- -initval *init_value*

  A keyword and parameter set that specifies the default value as follows:

  - The default value from the *choices_list*, if present.
  - The initial value for the text entry field if -choices “” is specified.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

  - The value of the variable being defined if the -choices option is not specified. The assigned value cannot be changed in this case.

The *init_value* is read and applied the first time the Customization GUI is opened or if the customization file changes; however, *init_value* is overridden with the runset value if the runset option cmnCustomFileOverrideValues specifies a value for the control.

The *init_value* is always applied if the Customization GUI is opened using the **Load** button in the Customization Settings dialog box; the runset value specified with the runset option cmnCustomFileOverrideValues is not used.

- -prompt *prompt_text*

  Places the string *prompt_text* next to the control button or checkbox.

- -tooltip *tooltip_string*

  Specifies a tooltip *tooltip_string* for the control. Line breaks in *tooltip_string* are preserved in the generated tooltip.

- -multiline {0|1}

  Controls whether the text entry box is a single line (0) or multiple lines (1). The default is 0.

- -boolean {0|1}

  Controls the display of the checkbox, where 1 specifies to display the checkbox.

  The default value is 1 (display a checkbox) if -boolean is not specified. If the checkbox is selected (the default), a #define ***var_name*** statement is generated. If the checkbox is unselected, an #undefine ***var_name*** statement is generated.

- -select {0|1}

  Controls the initial setting of the -boolean checkbox. The default value is 1 (true) if this option is not specified.

- -enable {0|1}

  Controls whether the -boolean checkbox is active (responds to user input). The default value is 1 (true) if this option is not specified.

- -display {0|1}

  Controls whether any corresponding items are displayed in the customization dialog box. The default value is 1 (true) if this option is not specified.

- -tool {DRC | LVS | DFM | PEX | PERC | *tool_list*}

  If included, the statement applies only to the specified tool(s). To specify more than one tool, provide a Tcl list; for example: {DRC DFM}

  If -tool is not specified, the command applies to all tools.

- -tvf {0|1}

  Adds a Tcl variable to the Calibre Interactive control file when 1 is specified and a TVF rule file is being used. The default is 0 (false) if this option is not specified. If the option -tvf 1 is



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

specified when not using a TVF rule file, the customization command does not output any statement to the control file. See "Example 4."

- -vtype *type*

  Verify the type of the input if the control creates a text entry field. The parameter *type* is one of the following:

  - none — Any type is accepted. This is the default if the -vtype option is not specified.
  - nospaces — No spaces are allowed in the input.
  - integer — Integer input.
  - float — Floating point input. Scientific notation is accepted; for example: -54.3E6.
  - nonnegint — Non negative integer input.
  - nonnegreal — Non negative floating point input. Scientific notation is not accepted.

  Input that does not satisfy the specified type is ignored.

- -vcmd '[' list *v_proc* ']'

  Specifies a Tcl procedure *v_proc* to call in order to perform input verification for controls that create a text entry field. The brackets are required to cause command substitution. The procedure *v_proc* should return 1 if the input is acceptable and return 0 otherwise. See "Input Type Checking for Customization GUI Entry Fields" on page 644.

- {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec*

  An optional keyword and parameter set that specifies the current control as a dependent control to another primary control in the customization file. In other words, this option specifies that the appearance of this GUI object in the customization window depends on the value of another control referenced by *master_spec* or *expr_spec*.

  See "Primary-Dependent Commands" on page 640 for a complete description of using the -master option.

## Description

This command inserts a #DEFINE or #UNDEFINE pre-processor directive in the Calibre Interactive control file. The customization control is displayed with a checkbox by default. A #DEFINE statement is inserted if the checkbox is selected, and an #UNDEFINE statement if the checkbox is not selected. An optional text entry field or button with a dropdown list may be specified with the -choices argument; in this case a value is assigned to the variable.

The #DEFINE and #UNDEFINE directives are described under "Pre-Processor Directives" in the Standard Verification Rule Format (SVRF) Manual.

If the -tvf argument is included, then a Tcl variable is inserted in the Calibre Interactive control file, rather than a #DEFINE statement.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The -vtype and -vcmd switches are used to specify type checking for input fields. The input type checking performed with -vtype float and nonnegreal only checks that the input characters are valid for the specified type—checking for valid numbers is not performed. See "Input Type Checking for Customization GUI Entry Fields" on page 644 for more information and examples.

## Examples

### Example 1

This example creates a checkbox in the Customizations Settings dialog box with a "Use Marker Diode" prompt. The checkbox is selected by default.

```
CUSTOM::DEFINE -name "MARKER_DIODE" -boolean 1 -select 1 \
   -prompt "Define MARKER_DIODE"
```

This creates the following checkbox:

and places this command in the control file:

```
#define MARKER_DIODE
```

### Example 2

This example creates a text entry field and initializes the value as CLK.

```
CUSTOM::DEFINE -name NET_NAME -choices {""} -initval "CLK" \
   -prompt "Enter net name for analysis"
```

This creates the following checkbox with a text entry field:

### Example 3

This example creates a dropdown list of choices for the value of the variable being defined. The dropdown list displays specified text for each choice.

```
CUSTOM::DEFINE -name PROCESS -prompt "Process env var" \
   -choices [list [list "Wide metal process" wmp] \
                  [list "Process 7lm" lm7]]
```

This creates the following checkbox with a dropdown list of choices:

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Example 4

This example shows the use of the -tvf option. The following commands in a customization file:

```
CUSTOM::DEFINE -name df_varA -initval 3 -prompt "df_varA" -tvf 1
CUSTOM::DEFINE -name df_varB -prompt "df_varB" -select 0 -tvf 1
CUSTOM::DEFINE -name df_tvfC -choices {"poly" "metal" "all"} -tvf 1
```

produce this output by default at the beginning of the TVF control file:

```
## +++ TVF CUSTOMIZATION SETTINGS START +++

set df_varA 3
set df_varB 0
set df_tvfC "poly"

## +++ TVF CUSTOMIZATION SETTINGS END +++
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::LABEL

Command type: Customization command for Calibre Interactive.

Inserts a text label in the customization dialog box.

## Usage

**CUSTOM::LABEL -prompt *label_text*** [-tooltip *tooltip_string*]
[-tool {DRC | LVS | PEX | DFM | PERC | *tool_list*}]
[-wrapwidth *width*]
[ {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec* ]

## Arguments

- **-prompt *label_text***

  Prints the string *prompt_text* in the label box.

- -tooltip *tooltip_string*

  Specifies a tooltip *tooltip_string* for the control. Line breaks in *tooltip_string* are preserved in the generated tooltip.

- -tool {DRC | LVS | PEX | DFM | PERC | *tool_list*}

  If included, the statement applies only to the specified tool(s). To specify more than one tool, provide a Tcl list; for example: {DRC DFM}.

  If -tool is not specified, the command applies to all tools.

- -wrapwidth *width*

  Specifies that the label text wraps at a *width* pixels.

- {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec*

  An optional keyword and parameter set that specifies the current control as a dependent control to another primary control in the customization file. In other words, this option specifies that the appearance of this GUI object in the customization window depends on the value of another control referenced by *master_spec* or *expr_spec*.

  See “Primary-Dependent Commands” on page 640 for a complete description of using the -master option.

## Examples

CUSTOM::LABEL -prompt "Select checks by group "

Also see “Customization File to Select ERC Rule Checks” on page 651.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::SEPARATOR

Command type: Customization command for Calibre Interactive.

Inserts a horizontal separator line in the customization dialog box.

## Usage

**CUSTOM::SEPARATOR** [-tool { DRC | LVS | PEX | DFM | PERC | *tool_list* }]
[ {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec* ]

## Arguments

- -tool {DRC | LVS | PEX | DFM | PERC | *tool_list* }

  If included, the statement applies only to the specified tool(s). To specify more than one tool, provide a Tcl list; for example: {DRC DFM}.

  If -tool is not specified, the command applies to all tools.

- {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec*

  An optional keyword and parameter set that specifies the current control as a primary control to another primary control in the customization file. In other words, this option specifies that the appearance of this GUI object in the customization window depends on the value of another control referenced by *master_spec* or *expr_spec*.

  See "Primary-Dependent Commands" on page 640 for a complete description of using the -master option.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::SETWINGEOM

Command type: Customization command for Calibre Interactive.

Sets the size and location of the customization window.

## Usage

**CUSTOM::SETWINGEOM widthxheight**[±*x_coord*±*y_coord*]

## Arguments

- ***width***x***height***

  The parameters *width* and *height* are positive integers specifying the desired pixel dimensions of the window. The parameters *width* and *height* are separated by a lowercase 'x'.

  The geometry parameter has no spaces between any of the elements.

- ±*x_coord*±*y_coord*

  The optional parameters *x_coord* and *y_coord* specify the desired location of the window on the screen, in pixels.

  If *x_coord* is preceded by plus sign (+), it specifies the number of pixels between the left edge of the screen and the left edge of the window's border; if preceded by a minus sign (-) then *x_coord* specifies the number of pixels between the right edge of the screen and the right edge of the window's border.

  If *y_coord* is preceded by plus sign (+) then it specifies the number of pixels between the top of the screen and the top of the window's border; if *y_coord* is preceded by a minus sign (-) then it specifies the number of pixels between the bottom of the window's border and the bottom of the screen.

## Examples

The following example causes the window to display with its upper left corner at screen coordinate (10,20).

```
CUSTOM::SETWINGEOM 200x200+10+20
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::VARIABLE

Command type: Customization command for Calibre Interactive.

Inserts a Variable statement in the Calibre Interactive control file, or defines a Tcl variable in the case of TVF rule files.

## Usage

**CUSTOM::VARIABLE -name *var_name***
[-choices { *“”* | *choices_list* } [-multi_choice {0|1}] ] [-initval *init_value*]
[-vartype { auto | numeric | string [-file_browser {0|1}] | expression } ]
[-prompt *prompt_text*] [-tooltip *tooltip_string*] [-multiline {0|1}]
[-boolean {0 | 1} [-select {0|1}] [-enable {0|1}]] [-display {0|1}]
[-tool {DRC | LVS | PEX | DFM | PERC | *tool_list*}] [-tvf {0|1}]
[-vtype *type*] [-vcmd ‘[’ list *v_proc* ‘]’ ]
[ {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec* ]

## Arguments

- **-name *var_name***

  A required option and parameter that specifies the name of the variable output to the control file.

- -choices {“” | *choices_list* }

  Specifies whether a text entry field or a dropdown list with a list of choices is displayed for the control.

  - “” (empty string) — Display a text entry field for specifying the value of the variable. A text entry field is also displayed if the -choices argument is not included.
  - *choices_list* — Display a control button with a dropdown list of choices for the value of the variable. The values are displayed as mutually exclusive radio buttons.

    The *choices_list* can be specified in the following ways:

    - A Tcl list of single elements. For a list of single elements, the values themselves are displayed in the dropdown list of the control button.
    - A Tcl list of two-element sublists. The first element is displayed in the fields of the dropdown list and the second element is the actual value. For example:

      ```
      {{"poly layer" 7} {"metal layer" 5}}
      ```

      where the braces are required to indicate a Tcl list.

- -multi_choice {0|1}

  Specifies whether multiple items can be selected from the *choices_list.* This switch and value can only be used if “-choices *choices_list*” is specified. The default is 0, indicating that only one value can be selected.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If "-multi_choice 1" is specified the -initval parameter may specify a list. See "Example 3: CUSTOM::VARIABLE with Multiple Selections" on page 629.

- -initval *init_value*

  Specifies an initial value with one of the following choices:

  - The default value from the -choices list, if present.
  - The initial value of the variable. This value appears in the text entry field if the -choices option is not specified or if -choices "" is specified.

  The *init_value* may be a list if "-multi_choice 1" is used.

  The *init_value* is read and applied the first time the Customization GUI is opened or if the customization file changes; however, *init_value* is overridden with the runset value if the runset option cmnCustomFileOverrideValues specifies a value for the control.

  The *init_value* is always applied if the Customization GUI is opened using the **Load** button in the Customization Settings dialog box; the runset value specified with the runset option cmnCustomFileOverrideValues is not used.

- -vartype { auto | numeric | string [-file_browser {0|1}] | expression }

  Specifies how to interpret the input value for the variable and whether to surround the value with quotes.

  - auto — This is the default. If the input value is a floating point number, write the value without quotes; otherwise, surround the value with double quotes. If embedded spaces are present, the value is interpreted as a list and each list element is surrounded by double quotes.
  - numeric — Interpret the input value as a number and do not surround it with double quotes.
  - string [-file_browser {0|1}]

    string — Interpret the input value as a string and surround the value with double quotes. If embedded spaces are present, the value is interpreted as a list and each list element is surrounded by double quotes.

    -file_browser {0|1} — Specifies whether or not to includes a file browser button with the control. The -file_browser option is only valid with "-vartype string".
  - expression — Interpret the value as an expression and do not surround with double quotes. Embedded spaces are kept.

  Most of the possibilities for -vartype and the corresponding output in the control file are shown in "Example 5: CUSTOM::VARIABLE with Different Variable Types."

- -prompt *prompt_text*

  Places the string *prompt_text* next to the control button or checkbox.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- -tooltip *tooltip_string*

  Specifies a tooltip *tooltip_string* for the control. Line breaks in *tooltip_string* are preserved in the generated tooltip.

- -multiline {0|1}

  Controls whether the text entry box is a single line (0) or multiple lines (1). The default is 0.

- -boolean {0|1}

  Controls the display of the checkbox, where 1 specifies to display the checkbox. The default value is 0 (do not a checkbox) if -boolean is not specified.

  If a checkbox is used, the Variable statement is written to the control file only when the checkbox is selected.

- -select {0|1}

  Controls the initial setting of the -boolean checkbox. The default value is 1 (selected) if this option is not specified.

- -enable {0|1}

  Controls whether the -boolean checkbox is active (responds to user input). The default value is 1 (enabled) if this option is not specified.

- -display {0|1}

  Controls whether the control is displayed in the customization dialog box. The default value is 1 (true) if this option is not specified.

- -tool {DRC | LVS | PEX | DFM | PERC | *tool_list* }

  If included, the statement applies only to the specified tool(s). To specify more than one tool, provide a Tcl list; for example: {DRC DFM}

  If -tool is not specified, the command applies to all tools.

- -tvf {0|1}

  Adds a Tcl variable to the Calibre Interactive control file when 1 is specified and you are using a TVF rule file. The default is 0 (false) if this option is not specified. If the option -tvf 1 is specified when not using a TVF rule file, the customization command does not output any statement to the control file.

- -vtype *type*

  Verify the type of the input if the control creates a text entry field. The parameter *type* is one of the following:

  - none — Any type is accepted. This is the default if the -vtype option is not specified.
  - nospaces — No spaces are allowed in the input.
  - integer — Integer input.
  - float — Floating point input. Scientific notation is accepted; for example: -54.3E6.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

  - nonnegint — Non negative integer input.
  - nonnegreal — Non negative floating point input. Scientific notation is not accepted.

  Input that does not satisfy the specified type is ignored.

- -vcmd '[' list *v_proc* ']'

  Specifies a Tcl procedure *v_proc* to call in order to perform input verification for controls that create a text entry field. The brackets are required to cause command substitution. The procedure *v_proc* should return 1 if the input is acceptable and return 0 otherwise. See "Input Type Checking for Customization GUI Entry Fields" on page 644.

- {{-master *master_spec*} [-master_select {0 | 1}]} | -master_expr *expr_spec*

  An optional keyword and parameter set that specifies the current control as a dependent control to another primary control in the customization file. In other words, this option specifies that the appearance of this GUI object in the customization window depends on the value of another control referenced by *master_spec* or *expr_spec*.

  See "Primary-Dependent Commands" on page 640 for a complete description of using the -master option.

## Description

Inserts a Variable statement in the Calibre Interactive control file. It displays either a text entry field (the default) or a button with a dropdown list (-choices *choices_list*). Multiple items from the *choices_list* may be selected if "-multi_choice 1" is specified.

If the Variable value is a string (-vartype string), the option "-file_browser 1" can be specified to include a file browser button with the control.

The variable must be set to a value (not an empty string or white space) for the Variable statement to be written to the control file.

If "-tvf 1" is included, a Tcl variable is defined in the Calibre Interactive control file rather than adding a Variable statement. Two statements are added to the control file: a Tcl set command and a tvf::set_global_variable command.

The -vtype and -vcmd switches are used to specify type checking for input fields. The input type checking performed with -vtype float and nonnegreal only checks that the input characters are valid for the specified type—checking for valid numbers is not performed. See "Input Type Checking for Customization GUI Entry Fields" on page 644 for more information and examples.

## Examples

### Example 1: CUSTOM::VARIABLE

This example generates a "Variable "RES_TOL" 1" statement in the rule file for all Calibre runs. A text entry field with the prompt "var RES_TOL" is displayed in the Customization Settings dialog box for all tools.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
CUSTOM::VARIABLE -name "RES_TOL" -initval 1 -prompt "var RES_TOL"
```

### Example 2: CUSTOM::VARIABLE with Choices List

This example generates a default “Variable “DEPTH” ALL” statement in the control file for a Calibre Interactive nmLVS run. A “Text Depth:” prompt appears in the Customization Settings dialog box. A button with a dropdown list is displayed, where the selections in the list are ALL, PRIMARY, 1, 2, and NONE.

```
CUSTOM::VARIABLE -name "DEPTH" -choices {ALL PRIMARY 1 2 NONE} \
     -prompt "Text Depth: " -tool LVS
```

This displays the following control with a dropdown list of choices:

### Example 3: CUSTOM::VARIABLE with Multiple Selections

```
CUSTOM::VARIABLE -name "LISTVAR_J" -initval [list 1 3] \
   -choices {1 2 3 4} -multi_choice 1 \
   -prompt "List Variable J"
```

This displays the following control with a dialog to choose selections:

The following SVRF statement is added to the control file:

```
VARIABLE "LISTVAR_J" "1" "3"
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Example 4: CUSTOM::VARIABLE with Multiple Selections and TVF Output

If a TVF rule file is used and “-tvf 1” is included in the CUSTOM::VARIABLE statement for “Example 3: CUSTOM::VARIABLE with Multiple Selections”, the following Tcl code is added to the control file:

```
#!tvf

## +++ TVF CUSTOMIZATION SETTINGS START +++

set VAR_J "1 3"
tvf::set_global_variable VAR_J "1 3"

## +++ TVF CUSTOMIZATION SETTINGS END +++
```

### Example 5: CUSTOM::VARIABLE with Different Variable Types

The -vartype switch specifies how to interpret the input value. The following customization file demonstrates the use of the -vartype switch.

```
CUSTOM::VARIABLE -name "num_auto" -initval "2300" -vartype auto \
   -prompt "number, auto type"

CUSTOM::VARIABLE -name "num_string" -initval "2300" -vartype string \
   -prompt "number, string type"

CUSTOM::VARIABLE -name "num_numeric" -initval "2300" -vartype numeric \
   -prompt "number, numeric type"

CUSTOM::VARIABLE -name "mixed_auto" -initval "34xyz"  -vartype auto \
   -prompt "mixed value, auto type"

CUSTOM::VARIABLE -name "mixed_string" -initval "34xyz"  -vartype string \
   -prompt "mixed value, string type"

# control has a file browser to select a filepath for the filename string
CUSTOM::VARIABLE -name "map_filepath" -initval "map.txt"  \
   -vartype string -file_browser 1 -prompt "map file"

# This control causes an error since 34xyz is not a valid number
#CUSTOM::VARIABLE -name "mixed_numeric" -initval "34xyz"  \
#  -vartype numeric -prompt "mixed value, numeric type"

CUSTOM::VARIABLE -name "var_list" -initval "23 24" -vartype auto \
   -prompt "list, auto type"

# assign a value to var_num for use in the expression in the next control
CUSTOM::VARIABLE -name var_num -initval 10 \
   -prompt "enter var_num for expression"

CUSTOM::VARIABLE -name "var_expression" -initval "(var_num *2) + 5"  \
   -vartype expression -prompt "Var_num with expression"

CUSTOM::VARIABLE -name MGC_HOME -initval ENVIRONMENT \
   -vartype expression -display 0
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The above customization file results in the following output to the control file by default:

```
// +++ CUSTOMIZATION SETTINGS START +++

VARIABLE "num_auto" 2300
VARIABLE "num_string" "2300"
VARIABLE "num_numeric" 2300
VARIABLE "mixed_auto" "34xyz"
VARIABLE "mixed_string" "34xyz"
VARIABLE "map_filepath" "map.txt"
VARIABLE "var_list" "23" "24"
VARIABLE "var_num" 10
VARIABLE "var_expression" (var_num *2) + 5
VARIABLE "MGC_HOME" ENVIRONMENT

// +++ CUSTOMIZATION SETTINGS END +++
```

### Example 6: TVF Output for CUSTOM::VARIABLE

This example uses the -tvf option.

```
CUSTOM::VARIABLE -name var_tvfX -initval "procA" -tvf 1 -prompt "var_tvfX"
```

This code produces the following output in the TVF control file:

```
#!tvf

## +++ TVF CUSTOMIZATION SETTINGS START +++

set var_tvfX "procA"
tvf::set_global_variable var_tvfX "procA"

## +++ TVF CUSTOMIZATION SETTINGS END +++
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::getCalibreVersionString

Command type: Customization command for Calibre Interactive.

Gets the current Calibre version string.

## Usage

**CUSTOM::getCalibreVersionString**

## Arguments

None.

## Return Values

Returns a string with the Calibre version being run by the Calibre Interactive session. The string has the format v2012.4_0.28.

## Examples

```
set cal_version [CUSTOM::getCalibreVersionString]
```

See the command CUSTOM::setOKCallbackError for a extended example that uses the command CUSTOM::getCalibreVersionString.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::getRunsetVarValue

Command type: Customization command for Calibre Interactive.

Returns the setting of the Calibre Interactive GUI control given by the specified runset option name.

## Usage

**CUSTOM::getRunsetVarValue *name***

## Arguments

- ***name***

  The name of the GUI control. The name is given by the corresponding runset option variable name. See "Option Descriptions" on page 849 for a list of runset options.

## Return Values

Returns the setting of the specified GUI control.

## Description

Returns the setting of the Calibre Interactive GUI control given by *name*.

The customization file is processed after template settings are applied in Calibre Interactive, so the get commands return the GUI settings after template settings are made.

An error dialog is displayed if *name* is not a valid GUI control name. If the error occurs during reading of the customization file, the customization file is still processed and displayed. If the error occurs during execution of the CUSTOM::setOKCallback proc, then the setOKCallback proc does not complete and the customization dialog box is not updated or closed; you can use the **Cancel** button to close the dialog box.

## Examples

This example creates a CUSTOM::VARIABLE control initialized to the Calibre Interactive GUI setting for "Top Cell." The CUSTOM::VARIABLE command returns a Tk widget which is set to the Tcl variable topCellEntry, which is used later to get the setting of the customization control.

```
set topCellEntry [CUSTOM::VARIABLE -name "topCell_var" -choices {} \
   -initval [CUSTOM::getRunsetVarValue drcLayoutPrimary]  \
   -boolean 0 -select 1 -prompt "Top Cell (GUI setting)" ]
```

See CUSTOM::setRunsetVarValue for a complete example which uses a CUSTOM::setOKCallback proc and CUSTOM::setRunsetVarValue to set the "Top Cell" value in the GUI. Also see "Customization File to Read and Set Calibre Interactive GUI Controls" on page 648.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

When CUSTOM::getRunsetVarValue is used to initialize a control as in this example, CUSTOM::getRunsetVarValue is only evaluated the first time the customization file is read, unless the customization file changes. This means that changing the setting in Calibre Interactive after closing the Customization GUI does not result in a new initial value the next time the Customization GUI is opened.

In addition, -initval settings can be overridden by cmnCustomFileOverrideValues settings in the runset, as explained in "Saving Customization Settings" on page 597.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::runsetVarExists

Command type: Customization command for Calibre Interactive.

Determines if the input parameter is a valid GUI control name.

## Usage

**CUSTOM::runsetVarExists *name***

## Arguments

- ***name***

  The name of the GUI control. The name is given by the corresponding runset option variable name. See "Option Descriptions" on page 849 for a list of runset options.

## Return Values

Returns 1 if ***name*** is a valid GUI control name, 0 otherwise.

## Examples

This example determines if drcRulesFile is a valid GUI control name and prints a message based on the return value of CUSTOM::runsetVarExists.

```
if {[CUSTOM::runsetVarExists drcRulesFile]} {
   puts "found runsetVar drcRulesFile"
} else {
   puts "no such runsetVar drcRulesFile"
}
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::setRunsetVarValue

Command type: Customization command for Calibre Interactive.

Populates the setting of the Calibre Interactive GUI control given by *name* with the specified *value*.

## Usage

**CUSTOM::setRunsetVarValue *name value***

## Arguments

- ***name***

  The name of the Calibre Interactive GUI control. The name is given by the corresponding runset option name. See "Option Descriptions" on page 849 for a list of runset options.

- ***value***

  The value for the Calibre Interactive GUI control.

## Description

Populates the setting of the Calibre Interactive GUI control given by *name* with the specified *value*.

An error dialog is displayed if *name* is not a valid GUI control name. If the error occurs during reading of the customization file, the customization file is still processed and displayed. If the error occurs during execution of the CUSTOM::setOKCallback proc, then the setOKCallback proc does not complete and the customization dialog box is not updated or closed; you can click the **Cancel** button to close the dialog box.

## Examples

The following example can be used in a CUSTOM::setOKCallback proc to set "Top Cell" on the Calibre Interactive Inputs pane to the string "myTopCell".

```
CUSTOM::setRunsetVarValue LayoutPrimary "myTopCell"
```

The following extended example reads the "Top Cell" setting from the Calibre Interactive GUI and creates a CUSTOM::VARIABLE control initialized to the current GUI setting. You can change the setting for "Top Cell" in the Customization GUI, if desired. The CUSTOM::VARIABLE command returns a Tk widget which is set to the Tcl variable topCellEntry.

The callback proc myOKCallback reads the value of topCellEntry and uses CUSTOM::setRunsetVarValue to set the GUI control for "Top Cell" to the value currently set in the Customization GUI control.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
# customization file
# get CI value for top cell, and initialize control with value
set topCellEntry [CUSTOM::VARIABLE -name "topCell_var" -choices {""} \
   -initval [CUSTOM::getRunsetVarValue drcLayoutPrimary]  \
   -boolean 0 -select 1 -prompt "Top Cell (GUI setting)" ]

CUSTOM::setOKCallback myOKCallback
proc myOKCallback {} {
   global topCellEntry
   # get customization gui setting for topCellEntry widget
   set wv_topCell [$topCellEntry cget -cvar]
   # write setting to CI GUI control for top cell
   CUSTOM::setRunsetVarValue drcLayoutPrimary [set $wv_topCell]
}
```

When CUSTOM::getRunsetVarValue is used to initialize a control as in this example, CUSTOM::getRunsetVarValue is only evaluated the first time the customization file is read, unless the customization file changes. This means the initial value for the customization control does not change even if the Calibre Interactive GUI setting is changed sometime later.

In addition, -initval settings can be overridden by cmnCustomFileOverrideValues settings in the runset.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::setOKCallback

Command type: Customization command for Calibre Interactive.

Specifies a callback procedure that is executed when you click the **OK** button in the Customization GUI. If the Customization GUI is not displayed in a Calibre Interactive batch run, the callback procedure is automatically executed.

## Usage

**CUSTOM::setOKCallback *OKproc***

**CUSTOM::setOKCallback '[' list *OKproc arg1*** [*arg2* …] '**]**'

## Arguments

- ***OKproc***

  The name of the Tcl callback procedure.

- ***arg1*** [*arg2* …]

  One or more optional arguments to the callback procedure. If the callback procedure has arguments, they should be specified with ***OKproc*** as a Tcl list, as shown in the second syntax line.

## Description

The CUSTOM::setOKCallback command specifies a callback procedure that is executed when you click the **OK** button in the Customization GUI. The callback procedure is automatically executed if the Customization GUI is not displayed in a Calibre Interactive batch run—this is the case for batch runs without the -customgui command-line option. The CUSTOM::setRunsetVarValue command can be used in the callback procedure to set Calibre Interactive GUI controls based on Customization GUI input. The callback procedure should not include commands that create customization controls (the commands in Table 11-4 and Table 11-6). See "Customization File to Read and Set Calibre Interactive GUI Controls" on page 648 for a complete example.

If the callback procedure causes a Tcl error, a warning dialog appears containing the error message.

If the callback procedure ***OKproc*** returns -1, a warning dialog is displayed indicating there was a problem. The command CUSTOM::setOKCallbackError specifies a user-supplied error message for this warning dialog.

## Examples

See "CUSTOM::setRunsetVarValue" on page 636 for an extended example.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The following example specifies a callback procedure with no arguments:

```
proc myOKCallback {} {

# code ...
}

CUSTOM::setOKCallback myOKCallback
```

This example specifies a callback procedure with one argument:

```
proc myOKCallback {arg1} {
puts "starting myOKCallback with argument set to $arg1"
# code ...
}

CUSTOM::setOKCallback [list myOKCallback 5]
```

When you click the **OK** button in the Customization GUI, myOKCallback executes and prints the following line to stdout:

```
starting myOKCallback with argument set to 5
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CUSTOM::setOKCallbackError

Command type: Customization command for Calibre Interactive.

Specifies an error message that is displayed if the function specified by CUSTOM::setOKCallback returns -1.

## Usage

**CUSTOM::setOKCallbackError *"msg_text"***

## Arguments

- ***"msg_text"***

  The text of the error message, enclosed in quotes.

## Description

If the function specified by CUSTOM::setOKCallback returns -1, a warning dialog box is displayed that includes the text of ***msg_text.***

## Examples

The following example shows a callback procedure that checks the Calibre version. If the version is too early, the callback procedure sets the error message with CUSTOM::setOKCallbackError and returns -1.

```
# custom GUI controls ...

proc myOKCallback { } {

# check version and exit if not late enough
   set cal_version [CUSTOM::getCalibreVersionString]
   set version_year [string range $cal_version 1 4]

   if {$version_year < 2013} {
      CUSTOM::setOKCallbackError "Calibre $cal_version is too early"
      return -1
   }

# rest of callback procedure ...
# ...
}

CUSTOM::setOKCallback myOKCallback
```

# Primary-Dependent Commands

You can specify that the visibility of one control (the dependent control) depends on the state of one or more other controls (the primary) in the customization file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The following set of options are allowed for the customization commands that create GUI controls or elements:

{ {{-master *master_spec* } [-master_select {0 | 1}]} | -master_expr *expr_spec*}

- -master *master_spec* — An optional argument that specifies this control as a dependent to another primary control in the customization file. In other words, the appearance of this GUI object in the customization dialog box depends on the value of another control.

  The argument *master_spec* specifies the GUI control object that is the primary control and optional values to match. Suppose $*_master* references the primary control and *value1* and *value2* are two possible values for the control. The following syntax options are available, where the brackets and Tcl list command are literal:

  -master $*_master* — Show the dependent control if $*_master* is selected.

  -master [list $*_master value1*] — Show the dependent control if $*_master* is equal to *value1*.

  -master [list $*_master* [list *value1 value2* …] — Show the dependent control if $*_master* is equal to any value in the list *value1*, *value2* ….

  The optional matched value for the control is either from the -choices list or from a text entry field associated with the $*_master* control. Double quotes in the value must be escaped. See "Getting the Tcl control variable" on page 642 in the "Description" section for an example of how to reference the GUI control object.

- -master_select {0 | 1} — An optional argument specifying when the current (dependent) control is displayed. This option is only used with -master.

  0 — Display the dependent control when the primary control checkbox is *not* selected.

  1 — (default) Display the dependent control when the primary control checkbox is selected.

- -master_expr *expr_spec* — An optional argument that specifies this control as dependent to one or more primary controls in the customization file, where *expr_spec* is a Tcl expression that determines if the dependent control is displayed.

  Suppose $*_master1* and $*_master2* reference primary controls and *value1* and *value2* are two possible values for $*_master1*. The following syntax options demonstrate the types of expressions that are available, where the brackets and Tcl list command are literal:

  - -master_expr [list $*_master1* || $*_master2*] — Show the dependent control if either $*_master1* or $*_master2* is selected.
  - -master_expr [list $*_master1*&& $*_master2*] — Show the dependent control if both $*_master1* and $*_master2* are selected.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- -master_expr [list [list $_*master1* [list *value1 value2*]] || $_*master2*] — Show the dependent control if $_*master1* is equal to *value1* or *value2* or if $_*master2* is selected.
- -master_expr [list [list $_*master1 value1*] && $_*master2*] — Show the dependent control if $_*master1* is equal to *value1* and $_*master2* is selected.
- -master_expr [list ! $_*master1*] — Show the dependent control if $_*master1* is not selected or not visible.

The above list of expressions is not exhaustive; see "Expression Syntax for -master_expr" on page 642 in the "Description" section for a formal definition of the expression syntax.

## Description

The display of the control with the -master or -master_expr option is determined by the value of one of more other primary controls in the customization GUI. By default, the dependent control is shown if the checkbox for the primary control is enabled; this can be changed by specifying "-master_select 0" with the -master option to indicate that the dependent control is shown if the primary control checkbox is disabled. Use the -master_expr to specify a Tcl expression with multiple primary controls.

A dependent control can be a primary control for another control.

## Getting the Tcl control variable

Each CUSTOM command returns a Tcl/Tk widget. You can assign this widget to a Tcl variable using the Tcl set command, as shown in this example:

```
set _master [CUSTOM::DEFINE -name "change_defines" \
   -prompt "Change pre-processor defines"]
```

Here the CUSTOM command is enclosed in brackets to indicate command substitution. The Tcl variable is referenced using the dollar sign ($), as in $_*master*.

## Expression Syntax for -master_expr

The Tcl expression *expr_spec* in the option "-master_expr *expr_spec*" is defined as follows:

```
expr_spec = subexpr [op subexpr]....
subexpr = "(" subexpr ")" | control_expr [op {subexpr | control_expr}]....
control_expr = ["!"] control [value_list]
control = <value of Tcl var associated with customization dialog control>
value_list = <list of acceptable values for the given control>
op = "&&" | "||"
```

where control in the above expression is found as shown in the section "Getting the Tcl control variable" on page 642.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Examples

### Example 1

In this example, the control for the second CUSTOM::DEFINE command is only displayed if the checkbox for the control associated with the _masterDefines variable is enabled.

```
# Defines a primary control for the CUSTOM::DEFINE controls
set _masterDefines [CUSTOM::DEFINE -name "change_defines"  -select 1 \
     -prompt "Change pre-processor defines" -display 1 -tool DRC ]

CUSTOM::DEFINE -name "DRC_DENSITY" -boolean 1 -select 1  \
     -prompt "Define DRC_DENSITY" -tool DRC -master $_masterDefines
```

### Example 2

This example demonstrates several of the options for the -master option. Commands for three separate dependent controls are included to demonstrate matching a single value, matching a value in a list, and for displaying the dependent control if the primary control is not selected.

```
# Add a DEFINE for process type, it is a master for other controls
# process can have three possible values: wmp, 71m, or other
set _master_process [CUSTOM::DEFINE -name process \
     -choices {{"wide metal" wmp} {"process 7" 71m} {"other" other}} \
     -initval wmp -prompt "Add #DEFINE for process type" ]

# dependent control is displayed if process is set to wmp
# for _master_process control
CUSTOM::DEFINE -name "b_var" -boolean 1 -select 1  \
     -prompt "b_var, show if process wide metal (wmp)"  \
     -master [list $_master_process wmp]

# dependent control is displayed if process is set to wmp or 71m
# for _master_process control
CUSTOM::DEFINE -name "c_var" -boolean 1 -select 1  \
     -prompt "c_var, show if process wide metal or 7 (wmp or 71m)" \
     -master [list $_master_process [list wmp 71m]]

# dependent control (a label) is displayed if _master_process control
# is not enabled
CUSTOM::LABEL -prompt "process var not defined" \
   -master $_master_process -master_select 0
```

### Example 3

This example demonstrates several of the options for the -master_expr option. Commands for three separate dependent controls are included to demonstrate matching a single value, matching a value in a list, and for displaying the dependent control if the primary control is not selected.

```
# Add a DEFINE for process type as a primary control for other controls
# process can have three possible values: wmp, 71m, or other
set _master_process [CUSTOM::DEFINE -name process \
     -choices {{"wide metal" wmp} {"process 7" 71m} {"other" other}} \
     -initval wmp -prompt "Add #DEFINE for process type" ]
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
# Add a DEFINE for extra_checks, it is a primary control
# for other controls
set _master_checks [CUSTOM::DEFINE -name extra_checks \
     -initval wmp -prompt "run extra checks" ]

# dependent control is shown if either _master_process or _master checks
# control is enabled
CUSTOM::DEFINE -name "a_def" -boolean 1 -select 1  \
     -prompt "a_def, show if either process or extra checks defined"  \
     -master_expr [list $_master_process || $_master_checks]

# dependent control is shown if both _master_process and _master checks
# controls are enabled
CUSTOM::DEFINE -name "ab_def" -boolean 1 -select 1  \
     -prompt "ab_def, show if both process and extra checks defined"  \
     -master_expr [list $_master_process && $_master_checks]

# dependent control is shown if process
# is wmp or 71m (from _master_process)
# or if _master_checks control is enabled
CUSTOM::DEFINE -name "b_def" -boolean 1 -select 1  \
     -prompt "b_def, show if process is wmp or 71m, \
         or if extra checks is defined" \
     -master_expr [list [list $_master_process [list wmp 71m]] || \
         $_master_checks]

# dependent control is shown if process
# is wmp (from _master_process control)
# and if _master_checks control is enabled
CUSTOM::DEFINE -name "c_def" -boolean 1 -select 1  \
   -prompt "c_def, show if process is wmp and extra checks is defined" \
   -master_expr [list [list $_master_process wmp] && $_master_checks]

# dependent control (a label) is shown if _master_checks control \
# is not enabled
CUSTOM::LABEL -prompt "extra checks not defined" \
   -master_expr [ list !$_master_checks]
```

# Input Type Checking for Customization GUI Entry Fields

You can have the customization GUI verify that the input for a text entry field is of the correct type. The CUSTOM::DEFINE and CUSTOM::VARIABLE commands allow text entry fields and input type checking.

The following two optional parameter sets can be specified for the CUSTOM::DEFINE and CUSTOM::VARIABLE commands:

- -vtype *type* — Verify the type of the input if the control creates a text entry field. The parameter *type* is one of the following:
  - none — Any type is accepted. This is the default if the -vtype option is not specified.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

  - nospaces — No spaces are allowed in the input.
  - integer — Integer input.
  - float — Floating point input. Scientific notation is accepted; for example: -54.3E6.
  - nonnegint — Non negative integer input.
  - nonnegreal — Non negative real input. Scientific notation is not accepted.

  Input that does not satisfy the specified type is ignored.

- -vcmd ‘[’ list *v_proc* ‘]’ — Specifies a Tcl procedure *v_proc* to call in order to perform input verification for controls that create a text entry field. The brackets are required to cause command substitution. The procedure *v_proc* should return 1 if the input is acceptable and return 0 otherwise.

The -vtype and -vcmd options may be specified together. Input that is not of the correct type is ignored. If a control specifies an initial value that does not pass the input type checking, the incorrect entry is ignored.

The input type checking performed with -vtype float and -vtype nonnegreal only checks that the input characters are valid for the specified type—checking for valid numbers is not performed. For example, if -vtype float is specified, the entry 54.3e6E5 is considered valid because the individual characters are all allowed.

## Examples and Discussion

### Example 1: -vtype

This command creates a control with a text entry field, and outputs a Variable statement to the Calibre Interactive control file. The parameter set “-vtype integer” specifies that the input type is integer.

```
CUSTOM::VARIABLE -name "INT_VAR1" -choices {} -initval "8" \
   -prompt "INT_VAR1 (integer)" -vtype integer
```

Any input that is not consistent with an integer is ignored. For example, if “21.2” is typed in the text entry field, the “.2” is ignored and “21” is displayed in the text entry field. No notification is given when input is ignored in this manner.

### Example 2: -vcmd

The following code in a customization file creates two controls with text entry fields and provides a procedure to do the input type checking for each control (the -vcmd switch). Both type checking procedures allow for empty strings so that deleting existing input is allowed. Input type checking is also done with the -vtype switch.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
CUSTOM::DEFINE -name "def_lower" -choices {""} -boolean 0 \
   -prompt "def_lower (lower case, no spaces)" \
   -vtype nospaces -vcmd [list input_is_lower]

CUSTOM::VARIABLE -name "RES_TOL" -choices {""} -initval "8" \
   -prompt "RES_TOL  (1 < integer < 10)" \
   -vtype integer -vcmd [list check_range]

proc input_is_lower {args} {
   set newval $args
# allow empty string so backspacing and deleting is allowed
   if {$newval eq "" || [string is lower -strict $newval]} {
      return 1
   } else {
      return 0
   }
}

proc check_range {args} {
   set newval $args
# assumes -vtype option checks for integer value
# allow empty string so backspacing and deleting is allowed
      if {$newval eq "" } {
         return 1
   }  elseif { $newval > 1 && $newval < 10} {
         return 1
     } else {
         return 0
     }
}
```

The procedure input_is_lower() checks that the input is lower case, and returns 0 if not. Text input is ignored if input_is_lower() returns 0. For example, if “xyZ” is typed in the text entry field for the CUSTOM::DEFINE control, the “Z” is ignored and “xy” is displayed in the text entry field. The “-vtype nospaces” parameter set prevents the typing of spaces in the text entry field.

The procedure check_range() for the CUSTOM::VARIABLE control forces the input to be within a specified range. Input is ignored if check_range() returns 0. For example, if “21” is typed in the entry field, the “1” is ignored and “2” is displayed. The “-vtype integer” parameter set also forces the input to be of integer type.

# Config and Cget Options

Each of the CUSTOM commands returns a Tk widget. The Tk widget can be invoked with the standard Tk options config or cget.

## Config Option

```
widget config option value ?option value ...?
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

where the only valid options are -enable and -prompt. Other options have no effect.

If no options are specified, “widget config” returns a flat list composed of the name and value pairs of each parameter obtainable with cget.

### Cget Option

```
widget cget option
```

where the options that can be specified to cget are the same as those specified for the given CUSTOM command.

Three additional options are supported by cget:

- -type — Returns the type of the widget (DEFINE, VARIABLE, or CHECK).
- -bvar — Returns the name of the boolean variable that keeps track of the state of the checkbox.
- -cvar — Returns the name of the variable that keeps track of the choice control button.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Customization File Examples

Several customization file examples are included in the section.

**Customization File to Read and Set Calibre Interactive GUI Controls** . . . . . 648
**Customization File to Select ERC Rule Checks** . . . . . 651
**General Customization Example** . . . . . 652
**Customization File to Load Multiple Customization Files** . . . . . 655

## Customization File to Read and Set Calibre Interactive GUI Controls

You can read and set Calibre Interactive GUI controls using a customization file, as shown in this example.

In particular, this example customization file does the following:

- Creates text entry fields for the top cell and results database controls and initializes them to the Calibre Interactive GUI settings.
- Creates checkbox controls for the “Output cell error in cell space” and the “Show results in RVE” controls and initializes them to the Calibre Interactive GUI settings.
- Defines a callback procedure which does the following:
  a. Reads the setting for the text entry fields for top cell and results database in the Customization GUI.
  b. Reads the state of the checkbox for the “Output cell error in cell space” and the “Show results in RVE” controls in the Customization GUI.
  c. Writes the Customization GUI settings to the corresponding Calibre Interactive GUI controls.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Example

```
#customization file
CUSTOM::SETWINGEOM 500x250

# display control for top cell and initialize to current setting
set topCellEntry [CUSTOM::VARIABLE -name "topCell_var" -choices {""} \
   -initval [CUSTOM::getRunsetVarValue drcLayoutPrimary]  \
   -boolean 0 -select 1 -prompt "Top Cell (GUI setting)" ]
# display control for results DB and initialize to current setting
set resultsEntry [CUSTOM::VARIABLE -name "resultsFile_var" \
   -choices {""} -initval [CUSTOM::getRunsetVarValue drcResultsFile]  \
   -boolean 0 -select 1 -prompt "Results File (GUI setting)"  -tool DRC]

# display Boolean checkbox for Output in Cell Space and initialize to
# current value
set cellNameEntry [CUSTOM::DEFINE -name "cellName_var" \
   -select [CUSTOM::getRunsetVarValue drcCellName]  \
   -prompt "Output in cell space (GUI setting)" -tool DRC]

#display Boolean checkbox for Start RVE and initialize to current value
set startRVEEntry [CUSTOM::DEFINE -name "startRVE_var" \
   -select [CUSTOM::getRunsetVarValue drcStartRVE]  \
   -prompt "Start RVE (GUI setting)" ]
proc myOKCallback { } {
    global topCellEntry
    global resultsEntry
    global cellNameEntry
    global startRVEEntry

   # get settings from customization GUI control
   # use cget -cvar to get value, cget -bvar to get checkbox setting
    set wv_topCell [$topCellEntry cget -cvar]
    set wv_results [$resultsEntry cget -cvar]
    set wb_cellName [$cellNameEntry cget -bvar]
    set wb_startRVE [$startRVEEntry cget -bvar]
   # write setting to Calibre Interactive GUI
   CUSTOM::setRunsetVarValue drcLayoutPrimary [set $wv_topCell]
   CUSTOM::setRunsetVarValue drcResultsFile [set $wv_results]
   CUSTOM::setRunsetVarValue drcStartRVE [set $wb_startRVE]

   # test for existence of CI control drcCellName before writing
   if {[CUSTOM::runsetVarExists drcCellName]} {
      CUSTOM::setRunsetVarValue drcCellName [set $wb_cellName]
   } else {
      CUSTOM::setOKCallbackError "no such runset var: drcCellName"
      return -1
   }
}
CUSTOM::setOKCallback myOKCallback
```

Figure 11-2 show how the Customization GUI looks and the steps for changing and setting the Top Cell value in the Calibre Interactive GUI.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 11-2. Customization Example to Set Calibre Interactive GUI Controls**

When CUSTOM::getRunsetVarValue is used to initialize a control as in this example, CUSTOM::getRunsetVarValue is only evaluated the first time the customization file is read, unless the customization file changes. This means that changing the setting in Calibre Interactive after closing the Customization GUI does not result in a new initial value the next time the Customization GUI is opened.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Customization File to Select ERC Rule Checks

The customization file in this example selects the ERC rule checks that are executed in a Calibre Interactive nmLVS run.

The example defines a Tcl variable _select_by to determine whether all checks are run or if checks are selected by group name. The -master option with the _select_by variable is used to determine if the CUSTOM::CHECK controls are displayed; see "Primary-Dependent Commands" on page 640.

If you are also using the Select Checks dialog box in Calibre Interactive nmLVS to select ERC checks, then the selections in the Customization GUI initialize settings of the Select Checks dialog box. as described in "Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive" on page 604. See "Selecting ERC Checks" on page 284 for information on the Select Checks dialog box.

See "Rule Check Selection in Calibre Interactive Batch Runs" on page 103 for an example using a customization file with a Calibre Interactive batch run.

## Example

```
### Customization file to select ERC checks in LVS

# define var to select all checks, or select individually
# no commands needed for "All checks" since this is the default operation
set _select_by [CUSTOM::VARIABLE -name "select_by" \
   -choices {{"All checks" 0} {"individually" 1} } -initval {0} \
   -prompt "select ERC checks:" -display 1 -tool LVS ]

# select checks individually
CUSTOM::LABEL -prompt "Select ERC checks individually " \
   -tool LVS -master [list $_select_by 1]
CUSTOM::CHECK -name "ERC_Path" -prompt "ERC_Path" \
   -tool LVS -master [list $_select_by 1]
CUSTOM::CHECK -name "Floating" -prompt "Floating" \
   -tool LVS  -master [list $_select_by 1]
CUSTOM::CHECK -name "ERC_Check" -prompt "ERC_Check" \
   -tool LVS  -master [list $_select_by 1]
```

The following figure shows the Customization GUI for the above example.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 11-3. Customization GUI to Select Checks**

**Caution**

The Check Selection Recipe is the preferred method for selecting the rule checks that are evaluated in a Calibre Interactive nmDRC or Calibre Interactive DFM run.

If you are using CUSTOM::CHECK controls in a Customization GUI for Calibre Interactive nmDRC or DFM, make sure that the “Checks selected in the rules file” built-in check recipe is selected, as described in “Precedence of CUSTOM::CHECK Controls and Check Selection in Calibre Interactive” on page 604.

# General Customization Example

The following example contains many of the commands and options possible for customization files.

The control file for the run is shown after the customization file. The Customization Settings dialog box for a Calibre Interactive DRC session is shown in “Customization Overview” on page 588.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Example

```
##############################################################
#
#### Settings for all tools ##########
CUSTOM::SETWINGEOM 450x380

# Displays a GUI element with a text entry field for all tools.
# By default, it generates a Variable RES_TOL 1 statement in the
# control file.
CUSTOM::VARIABLE -name "RES_TOL" -initval 1 -prompt "Variable RES_TOL"
# Add define for process type
CUSTOM::DEFINE -name process \
     -choices {{"wide metal" wmp} {"process 7" 7lm}} \
     -initval wmp -prompt "Add #DEFINE for process type"
############ LVS settings ##############

CUSTOM::LABEL -prompt "LVS Run Settings" -tool LVS

# Displays a checkbox in the LVS GUI. The checkbox is selected by default
# and generates a #define MARKER_DIODE statement in the control file.
CUSTOM::DEFINE -name "MARKER_DIODE" -boolean 1 -select 1 -enable 1 \
     -prompt "Define MARKER_DIODE" -tool LVS

# Generates a Variable DEPTH ALL statement in the
# control file by default. The dropdown list includes
# ALL, PRIMARY, 1, 2, and NONE.
# ALL, PRIMARY, and NONE  must resolve to numeric values.
CUSTOM::VARIABLE -name "DEPTH" -choices {ALL PRIMARY 1 2 NONE} \
     -prompt "Text Depth: " -tool LVS
#

###### DRC settings ########
CUSTOM::LABEL -prompt "DRC GUI Settings" -tool DRC

# Controls for GUI runset settings, initialized to current setting.
# Need protection for runset var to prevent errors in LVS run
if {[CUSTOM::runsetVarExists drcCellName]} {

     set cellNameEntry [CUSTOM::DEFINE -name "cellName_var" \
   -select [CUSTOM::getRunsetVarValue drcCellName]  \
   -prompt "Output in cell space (GUI setting)" -tool DRC]
}
if {[CUSTOM::runsetVarExists drcStartRVE]} {
     set startRVEEntry [CUSTOM::DEFINE -name "startRVE_var" \
   -select [CUSTOM::getRunsetVarValue drcStartRVE]  \
   -prompt "Start RVE (GUI setting)" -tool DRC]
}
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
# Generates a Variable PAD_SIZE 10 statement in the control file.
#  No control is displayed in the Customization Settings dialog box.
CUSTOM::VARIABLE -name "PAD_SIZE" -initval 10 -display 0 -tool DRC
CUSTOM::SEPARATOR -tool DRC
# Defines a master control for the CUSTOM::DEFINE controls
set _masterDefines [CUSTOM::DEFINE -name "change_defines"  -select 1 \
     -prompt "Change pre-processor defines" -display 1 -tool DRC ]
CUSTOM::LABEL -prompt "Pre-Processor Defines" -tool DRC -master
$_masterDefines

CUSTOM::DEFINE -name "DRC_DENSITY" -boolean 1 -select 1  \
     -prompt "Define DRC_DENSITY" -tool DRC -master $_masterDefines
CUSTOM::DEFINE -name "DRC_NAR" -boolean 1 -select 0 \
     -prompt "Define DRC_NAR" -tool DRC -master $_masterDefines
##

## callback proc for OK button
proc myOKCallback { } {
    global cellNameEntry
    global startRVEEntry

    if {[CUSTOM::runsetVarExists drcCellName]} {
        set wb_cellName [$cellNameEntry cget -bvar]
   CUSTOM::setRunsetVarValue drcCellName [set $wb_cellName]
    }

    if {[CUSTOM::runsetVarExists drcStartRVE]} {
        set wb_startRVE [$startRVEEntry cget -bvar]
        CUSTOM::setRunsetVarValue drcStartRVE [set $wb_startRVE]
    }
}

CUSTOM::setOKCallback myOKCallback
```

Portions of the control file generated by a Calibre Interactive nmDRC session with the customization file shown in the preceding example are displayed below. “Customization Overview” on page 588 shows the settings that are in effect.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Control file excerpt:

```
//
//  Rule file generated on Tue Sep 04
//     by Calibre Interactive - DRC
//
//      *** PLEASE DO NOT MODIFY THIS FILE ***
//
// +++ CUSTOMIZATION SETTINGS START +++

VARIABLE RES_TOL 1
#define process wmp
#undefine cellName_var
#define startRVE_var
VARIABLE PAD_SIZE 10
#define change_defines
#define DRC_DENSITY
#undefine DRC_NAR

// +++ CUSTOMIZATION SETTINGS END +++

LAYOUT PATH "./layout.oas"
LAYOUT PRIMARY "TOPCELL"
LAYOUT SYSTEM OASIS
// more Calibre Interactive settings ...
// ...

DRC ICSTATION YES

INCLUDE "rules"
```

Variable and #define settings from the customization dialog box.

# Customization File to Load Multiple Customization Files

You can use the Tcl source command to create a customization file that loads multiple customization files; the following example shows such a file.

## Example

```
# customization file: custom_load_all
source custom_1_drc
source custom_2_drc
source custom_3_drc
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The customization file *custom_load_all* is specified in the same manner as any other customization file. For example, to specify the customization file when you invoke Calibre Interactive from the command line, use the following command:

```
% calibre -gui -drc -custom custom_load_all
```

Each customization file specified in *custom_load_all* is loaded sequentially into Calibre Interactive.

## Related Topics

Methods to Specify a Customization File



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 12
# Trigger Functions in Calibre Interactive

Calibre Interactive can execute trigger functions before and after Calibre Interactive execution and before and after the Calibre run.

**Trigger Overview** . . . . . 657
**Internal Triggers in Calibre Interactive** . . . . . 659
**External Triggers in Calibre Interactive** . . . . . 673

## Trigger Overview

Triggers are functions that execute at specific times in your work flow. Calibre Interactive has two types of triggers: external, which execute before and after Calibre Interactive, and internal, which execute before and after the Calibre run.

The triggers types are described in the following table.

**Table 12-1. Triggers in Calibre Interactive**

| Trigger Type | Characteristics |
|---|---|
| Internal Triggers in Calibre Interactive | Internal triggers are defined within Calibre Interactive and can be saved to a runset. Internal trigger functions run in the Linux shell or a supported design tool. |
| External Triggers in Calibre Interactive | External triggers operate when you start Calibre Interactive from a layout or schematic viewer that supports external triggers. They are written in the API for the supported design environment using the API language (Tcl, SKILL, or Scheme, for example). |

Both Internal and External triggers can be set up as pre-execution or post-execution triggers. The following figure illustrates the order in which the various triggers are executed during a Calibre Interactive job.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Figure 12-1. Trigger Pre- and Post-Execution Order

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Internal Triggers in Calibre Interactive

Calibre Interactive provides the ability to specify and run pre- and post-execution internal triggers. The trigger program is run immediately before or after batch Calibre. A trigger can be any program or script that can be executed from the Linux shell. Triggers can also run in supported design tools; in this case the trigger is written in the API language for the design tool.

Internal triggers are defined as follows:

- **Pre-execution internal trigger** — Performed before the Calibre job begins. If the internal pre-execution trigger returns an error code, the subsequent Calibre execution is canceled.

  If Calibre Interactive is exporting a database from an attached design tool, then Calibre Interactive checks for the existence of the exported file after the pre-execution trigger finishes executing.

  For Calibre nmLVS, Calibre PERC, and PEX runs, Calibre Interactive checks the source netlist for completeness after the pre-execution trigger finishes executing. (See "Check source netlist is complete before running LVS" in "Miscellaneous Preferences" on page 156).

- **Post-execution internal trigger** — Performed after the Calibre job ends. If the Calibre run itself fails, the internal post-execution trigger is not executed.

See "Trigger Overview" on page 657 for a diagram of the execution order of both internal and external trigger functions.

Calibre RVE for DRC offers additional options for setting triggers around database access events; see "Setup Database Filters and Triggers Options Pane" in the *Calibre RVE User's Manual*.

These topics are covered:

**Trigger Parameters for Internal Triggers in Calibre Interactive** . . . . . 659
**Setting Calibre Interactive Internal Triggers** . . . . . 661
**Trigger Access to the Design Tool Environment** . . . . . 665
**Trigger Options in the Runset** . . . . . 665
**Calibre Interactive Internal Trigger Examples** . . . . . 668

## Trigger Parameters for Internal Triggers in Calibre Interactive

Various filenames and other key values specified in the GUI can be passed as replaceable parameters that are expanded and passed to the internal trigger program.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The parameters listed in Table 12-2 are recognized for internal triggers and substituted with the appropriate runtime values. The column "Calibre Interactive Applications" lists the Calibre Interactive applications for which the parameter is valid. If you use a parameter not recognized in your run type, no substitution occurs. See "Calibre Interactive Internal Trigger Examples" on page 668 for examples using replaceable parameters in internal triggers functions. See "Setting Calibre Interactive Internal Triggers" on page 661 for instructions on defining internal triggers.

**Table 12-2. Internal Trigger Parameters**

| Runtime Replacement | Parameter | Calibre Interactive Applications |
|---|---|---|
| Circuit netlist | %c | LVS and PEX |
| DRC or DFM RDB | %b | DRC and DFM |
| DRC or DFM summary file | %m | DRC and DFM |
| Hcell file | %h | LVS, PERC, and PEX |
| Layout format<br>(GDSII, OASIS, LEFDEF, or OPENACCESS) | %F | All |
| Layout2 format<br>(GDSII, OASIS, LEFDEF, or OPENACCESS) | %F2 | DRC FastXOR |
| Layout path | %l | All |
| Layout2 path | %l2 | DRC FastXOR |
| Layout primary cell[1] | %L | All |
| Layout2 primary cell | %L2 | DRC FastXOR |
| Layout library | %y | All applications when started from Cadence Virtuoso |
| Layout2 library | %y2 | DRC FastXOR when started from Cadence Virtuoso |
| Layout view | %v | All |
| Layout2 view | %v2 | DRC FastXOR |
| LVS Report file | %m | LVS |
| LVS Report file | %n | PEX |
| PERC Report file | %m | PERC |
| PEX netlist | %p | PEX |
| PEX netlist format | %f | PEX |
| PEX Report file | %m | PEX |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 12-2. Internal Trigger Parameters (cont.)**

| Runtime Replacement | Parameter | Calibre Interactive Applications |
|---|---|---|
| Rule file | %r | All |
| Run directory | %d | All |
| Runset filename | %R | All |
| Source library | %w | LVS, PEX, and PERC |
| Source path for primary netlist source[2] | %s | LVS, PEX, and PERC |
| Source primary cell | %S | LVS, PEX, and PERC |
| Source path for all netlist sources[3] | %N | LVS, PEX, and PERC |
| Source view | %z | LVS, PEX, and PERC |
| Termination status:<br>0 — Normal<br>1 — Aborted<br>2 and above — the Calibre exit code | %e | All |

[1] Wildcards in the layout primary cell name are only expanded for a post-execution trigger run on the local host. In addition, the layout format must be GDSII, and the Calibre Interactive application must be DRC, DFM, or PERC.

[2] For SPICE netlist input, %s returns the netlist(s) listed on the **Netlist** tab of the Inputs pane. In the case of VERILOG or MIXED source input, %s returns the SPICE file that is translated from the Verilog source input.

[3] In addition to the source paths returned by %s, the %N trigger parameter also returns the additional SPICE netlist files specified on the **Database > Library** tab of the Options pane. In the case of MIXED source input, %N also includes the SPICE netlist files listed on the **Netlist** tab of the Inputs pane.

# Setting Calibre Interactive Internal Triggers

You can instruct Calibre Interactive to run internal triggers before and after the Calibre run. Multiple trigger functions may be specified for each stage. If you are connected to a supported design tool, you can also specify procedures that run in the design tool environment.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Prerequisites

- For triggers that run in the Linux shell:

  A trigger program written in the shell programming or scripting language of your choice (Awk, Sed, or Perl, for example). See "Processing a Netlist Before the Calibre Run" in the *Calibre Solutions for Physical Verification* manual for a simple script run as a trigger.

- For triggers that run in the design tool (Cadence Virtuoso, Cadence Encounter, Synopsys IC Compiler, and Calibre DESIGNrev):

  - A procedure written in SKILL or Tcl, depending on your design tool. The procedure should return the proper success or fail return code for the design environment. Calibre Interactive halts execution if the return value indicates failure.

  - Calibre Interactive was invoked from the design tool or the environment variable MGC_CALIBRE_ALLOW_VIEWER_TRIGGERS is set to "always". See "Trigger Access to the Design Tool Environment" on page 665.

  For an example, see "Internal Trigger Executed in Calibre DESIGNrev to Count Layout Cells" on page 670.

## Video

"How To Do Pre- and Post-Processing with Calibre Interactive Internal Triggers"

- Run a pre-processing script in the shell environment
- Run a SKILL procedure in Cadence Virtuoso before the Calibre run
- Use replaceable parameters as trigger arguments

## Procedure

1. Choose **Setup** > **Preferences**.
2. Click the **Triggers** tab.
3. Click the **Pre-Execution** tab.
4. Add trigger information to the table as shown in Figure 12-2. See the dialog box or "Trigger Parameters for Internal Triggers in Calibre Interactive" on page 659 for a list of available parameters.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 12-2. Calibre Interactive Internal Triggers**

Triggers are executed in the order shown; drag and drop a row to change the order.

The Type column specifies the where the trigger runs:

| Trigger Type | Execution environment for trigger |
|---|---|
| process | (default) Run as a Linux process. |
| layout | Run in the layout viewer.<br>(Cadence Virtuoso, Cadence Encounter, Synopsys IC Compiler, and Calibre DESIGNrev only) |
| schematic | Run in the schematic viewer.<br>(Cadence Virtuoso only) |

By default, layout and schematic triggers can only run in the design tool session that invoked Calibre Interactive. See "Trigger Access to the Design Tool Environment" on page 665 for further information.

5. Click the **Post-Execution** tab and enter trigger information as shown in Figure 12-2. The following options are also available for post-execution triggers:

   - **Run trigger in background** — Run the post-execution trigger as a background process.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Also run trigger if Calibre is terminated by user** — Run the post-execution triggers if the Calibre process is terminated by the user. By default the trigger is not run if the Calibre run is terminated.
- **Run on remote host** — Run the post-execution triggers on the same host as the main Calibre run. This setting only applies to triggers of type "process" (see Figure 12-2). By default, the post-execution trigger is run on the local host (the host running Calibre Interactive).

  The following statements apply when this option is enabled:

  - The post-execution trigger runs on the host specified by the "Run Calibre on" setting in Run Control pane on the **Performance** tab.
  - The post-execution trigger script is added to the control script for the Calibre run.
  - The control script preserves the Calibre exit code on return. The exit code of the post-execution trigger is not kept.
  - For Calibre MTflex runs, the control script is added to the primary host's script.
  - If using a Post-LVS-Extraction trigger during a Calibre Interactive PEX run, the Post-LVS-Extraction trigger is added to the PHDB control script.
  - If this setting is enabled, post-execution layout- and schematic type triggers are run in the design tool after the process-type triggers complete execution.

6. (Optional) If you are running Calibre Interactive PEX you can enter triggers on the **Calibre Post-LVS-Extraction** tab. These triggers run following the PHDB generation step of a PEX run. For more information on enabling PEX steps, see "Controlling PEX Step Execution" on page 321.
7. Click **OK**.

## Results

If the pre-execution trigger returns an error code, the subsequent Calibre execution is canceled. If the Calibre run itself fails, the post-trigger is not executed.

If a trigger function exits with an error code, the error code is printed to the transcript. The message has the following general format:

```
Trigger function <trigger_function> exited with return code <code>.
<error_message>
child process exited abnormally
```

## Related Topics

Calibre Interactive Internal Trigger Examples

Trigger Options in the Runset

External Triggers in Calibre Interactive



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Trigger Access to the Design Tool Environment

The **Triggers** tab on the Setup Preferences dialog box allows you to specify triggers that run in the design tool environment (the trigger Type of “layout” or “schematic”.) By default, a layout or schematic trigger can only run in the design tool session that launched the Calibre Interactive session.

For example, suppose you have two separate Cadence Virtuoso sessions open, one a layout view and one a schematic view. If you launch Calibre Interactive from the layout session, by default you can only run triggers in that session; triggers in the schematic session are not allowed to run.

The environment variable MGC_CALIBRE_ALLOW_VIEWER_TRIGGERS specifies the trigger access to the design tool environment.
MGC_CALIBRE_ALLOW_VIEWER_TRIGGERS can be defined as follows:

- **always** — Allow any process connected to the server socket to run trigger code in the design tool process. The value “always” is case-insensitive.
- **none** — Do not allow any process connected to the server socket to run trigger code in the design tool process. Any value other than “always” has the same effect.
- **undefined** — (default) A layout or schematic trigger can only run in the design tool session that launched the Calibre Interactive session if MGC_CALIBRE_ALLOW_VIEWER_TRIGGERS is undefined.

### Related Topics

Setting Calibre Interactive Internal Triggers

# Trigger Options in the Runset

The way in which Calibre Interactive saves trigger definitions in the runset differs according to the Calibre version.

This process is followed when reading trigger options in the runset:

- 2011.3 release and earlier

  Trigger definitions are saved in the runset with the *cmnPreTrigger and *cmnPostTrigger runset options; these are the only runset options recognized for defining triggers. The trigger definition applies to all Calibre Interactive applications.

- 2011.4 release through 2013.2

  When trigger definitions are saved in the runset, the runset option name reflects the Calibre Interactive application; for example: *drcPreTrigger and *drcPostTrigger. The trigger definition only applies to the specified Calibre Interactive application.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- 2013.3 and later

  Multiple triggers may be defined and the triggers may run in different environments. The runset options now have an “s” at the end of the option name. The runset options are a list of lists of this format:

  *drcPreTriggers: {name type enable} …

  - **name** — The program or procedure (and optional arguments) to execute before Calibre is run.
  - **type** — Type may have the values layout, schematic, or process.
  - **enable** — A Boolean value specifying whether to run the trigger.

  See “Setting Calibre Interactive Internal Triggers” on page 661 for complete information and the meaning of the different trigger types.

The following table lists the runset options available for defining triggers.

**Table 12-3. Runset Options for Trigger Definition**

| Pre-trigger, Post-trigger | Usage |
|---|---|
| *cmnPreTrigger, *cmnPostTrigger | • Calibre version 2011.3 and earlier — Applies to all Calibre Interactive applications.<br>• Calibre version 2011.4 and later — These trigger definitions are used only if the application-specific trigger definition is not present. |
| *drcPreTrigger, *drcPostTrigger<br>*lvsPreTrigger, *lvsPostTrigger<br>*dfmPreTrigger, *dfmPostTrigger<br>*percPreTrigger, *percPostTrigger<br>*pexPreTrigger, *pexPostTrigger | • Calibre version 2011.3 and earlier — Not recognized.<br>• Calibre version 2011.4 through 2013.2 — Trigger definitions apply only to the indicated application. The application-specific trigger definitions takes precedence over *cmnPreTrigger and *cmnPostTrigger. |
| *drcPreTriggers, *drcPostTriggers<br>*lvsPreTriggers, *lvsPostTriggers<br>*dfmPreTriggers, *dfmPostTriggers<br>*percPreTriggers, *percPostTriggers<br>*pexPreTriggers, *pexPostTriggers | • Calibre version 2013.2 and earlier — Not recognized.<br>• Calibre version 2013.3 and later — Trigger definitions are a list of lists. See “Setting Calibre Interactive Internal Triggers” on page 661 and “Runset Options” on page 847. |

The following rules are followed when saving a runset in Calibre versions 2011.4 through 2013.2:

- If *cmnPreTrigger or *cmnPostTrigger runset options are present in the runset, their definitions are not changed.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- When a runset is saved, trigger definitions are saved as the application-specific runset options.

  However, if a runset with *cmnPreTrigger or *cmnPostTrigger runset options is loaded, you will not be prompted to save the runset unless the triggers or other runset options have changed.

When saving a runset in Calibre version 2013.3 or later, the runset options are automatically updated to the new format.

See "Applying Triggers According to the Calibre Interactive Application" on page 671 for an example of using application-specific triggers.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Interactive Internal Trigger Examples

Several examples are provided for Calibre Interactive internal triggers.

See "Setting Calibre Interactive Internal Triggers" on page 661 for instructions on specifying the trigger in Calibre Interactive.

You can find more examples in the chapter "Pre- and Post-Processing for Calibre Runs" in the *Calibre Solutions for Physical Verification* manual.

**Internal Trigger for Calling a UNIX Utility** . . . . . 668
**Internal Trigger for Calling a Script** . . . . . 668
**Internal Trigger to Set an SVRF Variable** . . . . . 668
**Internal Trigger Executed in Calibre DESIGNrev to Count Layout Cells** . . . . . 670
**Applying Triggers According to the Calibre Interactive Application** . . . . . 671

## Internal Trigger for Calling a UNIX Utility

This example calls the "echo" utility with the trigger parameter %d as an argument. %d is replaced by the run directory at runtime.

Place the following text in the **Triggers** tab as a pre-execution trigger:

```
echo %d
```

The parameter %d is replaced with the run directory at runtime, and the result of the echo command is displayed in the Calibre transcript.

## Internal Trigger for Calling a Script

This example calls a custom script with the trigger parameter %r as an argument. %r is replaced by the rule file path at runtime.

Place the following text in the **Triggers** tab as a pre-execution trigger:

```
my_script %r
```

The script my_script is executed before the Calibre run starts, and is passed the path to the rule file.

## Internal Trigger to Set an SVRF Variable

In this example an internal pre-execution trigger parses the topcell parameter and creates a file that is used as an INCLUDE file with the rule file for the Calibre run. The created file contains a Variable statement.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For this example, assume the following:

- The top cell name of a design is “<cell_name>_TOP”, where <cell_name> varies from design to design.
- The cell, “<cell_name>_NEXT”, is always in the sub-hierarchy of <cell_name>_TOP. The <cell_name> arguments are identical for each.
- There are other cell names in the design that end in “_NEXT”.
- The following rule check is in the rule file:

  ```
  inside_cell_test {@layer L1 inside "NEXT" cell
  L1 inside cell CELLVAR
  }
  ```

  In this rule check, CELLVAR is a variable name and should specify “<cell_name>_NEXT”.

Note that you cannot use “*_NEXT” to define CELLVAR, because this would include other cells ending in “_NEXT” that you do not want to check.

Take the following steps to specify an internal trigger that creates a file with a Variable statement and to include the created file in the Calibre Interactive run. The trigger function uses the replaceable parameter %L to write the current top cell name to the Variable statement.

1. Create a script named *my.trigger* as a pre-execution trigger for Calibre Interactive:

   ```
   echo "VARIABLE CELLVAR '$1'" | sed "s/TOP/NEXT/" > cellvar.include
   ```

2. Create a file named *cellvar.include* with a Variable statement.

   ```
   echo "VARIABLE CELLVAR 'cell_TOP'" > cellvar.include
   ```

   This is done so that the include file exists and CELLVAR is defined when Calibre Interactive compiles the rule file for the first time; otherwise Calibre Interactive reports a compilation error. The *cellvar.include* file is overwritten with the correct cell name before Calibre execution begins.

3. In Calibre Interactive, choose **Setup > Preferences**, then the **Triggers** tab and specify “*my.trigger* %L” as a pre-execution trigger.

   The *my.trigger* script uses the top cell argument (%L), replaces TOP with NEXT, and writes the following statement to the file *cellvar.include*:

   ```
   VARIABLE CELLVAR '<cell_name>_NEXT'
   ```

4. Choose **Setup > *app* Options**, then the **Include** tab and specify *cellvar.include* as an include file. (***app*** indicates the Calibre Interactive application being run)

With this internal trigger and the created include file, Calibre Interactive uses the correct value for CELLVAR in the inside_cell_test rule check.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Setting Calibre Interactive Internal Triggers

## Internal Trigger Executed in Calibre DESIGNrev to Count Layout Cells

You can specify a Calibre Interactive internal trigger that runs in Calibre DESIGNrev and counts the number of cells in the layout.

### Procedure

1. Add the following procedure definition to your *$HOME/.calibre_workspace/wbinit.tcl* file:

   ```
   # called from trigger with %l as the layout file argument
   proc count_cells {layout} {
      puts "cells in $layout: [layout peek $layout -cellcount]";
      return 1;
   }
   ```

   Make sure your Tcl procedure returns 1 for success; other return values indicate failure and cause Calibre Interactive to halt execution.

2. Open your design in Calibre DESIGNrev. The procedures defined in the *wbinit.tcl* file are loaded automatically.

3. Start Calibre Interactive.

4. Choose **Setup** > **Preferences**.

5. Click the **Triggers** tab.

6. Click the **Add Trigger** button ( ) to add a trigger definition to the table.

7. Type the following in the Trigger column to specify the count_cells procedure as the trigger function with %l as the argument. The parameter %l is replaced with the name of the layout database specified on the **Layout** tab of the Inputs pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
count_cells %l
```

8. Click in the Type column and select “layout” from the dropdown list. The “layout” option specifies that the trigger executes in the connected layout viewer.
9. Click **OK** to close the dialog box.
10. Set up the remaining Calibre Interactive options for your run.
11. Click the **Run** button in Calibre Interactive.

### Results

The text “Running viewer pre-trigger” is printed at the top of the Transcript pane in Calibre Interactive.

The following is printed to the Calibre DESIGNrev shell window for a layout database *lab2.calibre.db* with 9 cells:

```
cells in lab2.calibre.db: 9
```

### Related Topics

Setting Calibre Interactive Internal Triggers

## Applying Triggers According to the Calibre Interactive Application

In some development environments, a system administrator may define a global or project-level runset that sets options for *all* Calibre Interactive applications. In this case, it can be useful for the same runset to define internal pre- and post-execution triggers that differ according to Calibre Interactive application; this is possible in 2011.4 and later releases.

The following example defines different internal pre-execution triggers for Calibre Interactive nmLVS and Calibre Interactive PERC within a runset that defines some global options for Calibre Interactive.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

1. Add the following lines to the runset used to define global options:

```
*lvsPreTriggers: {{lvs_netlist_script %s %S} process 1}
*percPreTriggers: {{perc_netlist_script %s %S} process 1}
```

   The trigger scripts take the source netlist (%s) and source primary cell (%S) as input. The parameter “process” indicates that the trigger runs as a Linux process.

2. Define an environment variable to load the multiple runsets for Calibre Interactive:

```
setenv MGC_CALIBRE_LVS_RUNSET_FILE "global_runset lvs_runset"
setenv MGC_CALIBRE_PERC_RUNSET_FILE "global_runset perc_runset"
```

   See “Loading Runsets on Invocation” on page 39 for other methods of loading multiple runsets.

## Related Topics

Setting Calibre Interactive Internal Triggers

Trigger Options in the Runset



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# External Triggers in Calibre Interactive

Calibre Interactive external triggers operate when you start Calibre Interactive from a layout or schematic viewer that supports external triggers. External triggers are written in the API for the supported design environment using the API language (Tcl, SKILL, or Scheme, for example). The triggers can be used to run custom routines that check or set environment variables or call setup and post-processing functions.

For example, Calibre Interactive runsets and customization files for the Calibre Interactive Customization GUI can be set by environment variables defined in an external pre-execution trigger. (Refer to "About Calibre Interactive Runsets" on page 33 for more information on Runsets, and to "Customization Files" on page 587 for more information on Customization files).

**External Trigger Types** . . . . . 673
**External Trigger Functions and File Locations** . . . . . 675
**Calibre Interactive External Trigger Examples** . . . . . 678

## External Trigger Types

External triggers execute in the design tool environment before and after Calibre Interactive execution. There are two types of external triggers: pre-execution and post-execution.

| Trigger Type | Definition |
|---|---|
| External pre-execution trigger | Executes after you choose a Calibre Interactive application from the **Verification** (Siemens viewers) or **Calibre** (other viewers) menu in your design tool. The trigger executes before Calibre Interactive starts.<br>function name: mgc_start_calibre_trigger |
| External post-execution trigger | Executes when you close the Calibre Interactive window.<br>function name: mgc_close_calibre_trigger |

"Trigger Overview" on page 657 describes the order in which the various triggers are executed in a Calibre Interactive session.

The location where you define the trigger function depends on the desired scope of the trigger, as explained here:

- **User environment trigger** — The trigger is defined in the user environment and applies to all Calibre sessions for that user.
- **Local trigger** — The trigger is defined in the run directory for the design tool or directly loaded into the design environment. Depending on the design tool, this might be done



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

with a command line argument or in a shell window for the design tool. This trigger definition takes precedence over the user environment trigger definition.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# External Trigger Functions and File Locations

The external trigger functions and the file locations depend on the design tool that is used.

External triggers are only supported for the following design tools:

**External Trigger Definitions in Calibre DESIGNrev** . . . . . 675
**External Trigger Definitions in Cadence Virtuoso** . . . . . 676
**External Trigger Definitions in Synopsys IC Compiler** . . . . . 677

## External Trigger Definitions in Calibre DESIGNrev

You can define Calibre Interactive external pre- and post-execution triggers in Calibre DESIGNrev. This section gives the function name and the location of the file containing the function definition.

```
proc mgc_start_calibre_trigger {run_type cell} {
   <Tcl code>
}
proc mgc_close_calibre_trigger {run_type cell} {
   <Tcl code>
}
```

where the procedures are called with the following arguments:

- run_type — The Calibre Interactive application (drc, lvs, pex, perc, or dfm)
- cell — The cell open in the viewer

The default external trigger definitions in Calibre DESIGNrev return 1.

You can define your own external trigger by providing a locally defined procedure that overrides the default definition. The following table gives the file locations for defining the external trigger procedures in Calibre DESIGNrev.

**Table 12-4. File Locations for External Trigger Definitions in Calibre DESIGNrev**

| Trigger Scope | File Location |
|---|---|
| User environment | *~/.calibrewb_workspace/wbinit.tcl*<br>See "wbinit.tcl File Format" in the *Calibre DESIGNrev Layout View User's Manual* for more information. |
| Local | *wbinit.tcl* file in a directory specified with the MGC_CWB_CONFIG_DIRS environment variable. See "Multilevel Configuration" in the *Calibre DESIGNrev Layout View User's Manual* for more information.<br>You can also load a Tcl script with the -s command line option to calibre DESIGNrev. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

External Trigger in Tcl

## External Trigger Definitions in Cadence Virtuoso

You can define Calibre Interactive external pre- and post-execution triggers in Cadence Virtuoso. This section gives the procedure name and the location of the file containing the procedure definition.

```
procedure( mgc_start_calibre_trigger(drc_lvs_pex win)
prog( ()
   printf( "Starting %s on top cell in window %L...\n"
      upperCase(drc_lvs_pex) win)
   return(t)
))
procedure( mgc_close_calibre_trigger(drc_lvs_pex win)
prog( ()
   printf( "Closing %s on top cell in window %L...\n"
      upperCase(drc_lvs_pex) win)
   return(t)
))
```

where the procedures are called with the following arguments:

- drc_lvs_pex — The Calibre Interactive application (drc, lvs, pex, perc, or dfm)
- win — The ID of the window that Calibre Interactive was started from

The trigger functions are expected to return t or nil (true or false), and execution proceeds only if “t” is returned by the function.

You can define your own external trigger by providing a locally defined procedure that overrides the default trigger. The following table gives the file locations for defining the external trigger procedures in Cadence Virtuoso.

**Table 12-5. File Locations for External Trigger Definitions in Cadence Virtuoso**

| Trigger Scope | File Location |
|---|---|
| User environment | *.cdsinit*<br>Add the definition of the trigger procedure after the Cadence interface has been loaded (see “Creating an Interface to Cadence Virtuoso” on page 725). |
| Local | Consult Cadence documentation. |

### Related Topics

External Trigger in SKILL for Cadence Virtuoso



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# External Trigger Definitions in Synopsys IC Compiler

You can define Calibre Interactive external pre- and post-execution triggers in Synopsys IC Compiler. This section gives the procedure name and the location of the file containing the procedure definition.

```
proc mgc_start_calibre_trigger {run_type cell} {
   <Tcl code>
}
proc mgc_close_calibre_trigger {run_type cell} {
   <Tcl code>
}
```

where the procedures are called with the following arguments:

- run_type — the Calibre Interactive application (drc, lvs, pex, perc, or dfm)
- cell — the cell open in the viewer

You can define your own external trigger by providing a locally defined procedure that overrides the default trigger. The following table gives the file locations for defining the external trigger procedures in Synopsys IC Compiler.

**Table 12-6. File Locations for External Trigger Definitions in Synopsys IC Compiler**

| Trigger Scope | File Location |
|---|---|
| User environment | ~/*.synopsys_icc_gui.tcl*<br>Add the definition of the trigger procedure after *icc_calibre.tcl* has been sourced (see "Loading the Calibre Interface to Synopsys IC Compiler" on page 824). |
| Local | Consult Synopsys IC Compiler documentation. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Interactive External Trigger Examples

Tcl and SKILL examples are provided for Calibre Interactive external triggers.

**External Trigger in Tcl** . . . . . 678
**External Trigger in SKILL for Cadence Virtuoso** . . . . . 678

## External Trigger in Tcl

The following code is an example of a pre-execution external trigger in Tcl.

If the run type is DRC, it sets environment variables for the customization file and the runset. For Calibre DESIGNrev, this code can be placed in the *wbinit.tcl* file; see "External Trigger Definitions in Calibre DESIGNrev" on page 675 for complete information.

```
proc mgc_start_calibre_trigger {app cell} {
puts "start trigger for CI-$app in cell $cell"

# if DRC run then set customization file and runset
if { $app eq "drc"} {
   global env
   set env(MGC_CALIBRE_CUSTOMIZATION_FILE) "custom_DRC"
   set env(MGC_CALIBRE_DRC_RUNSET_FILE) "runset_test"
   puts "runset: $env(MGC_CALIBRE_DRC_RUNSET_FILE)"
   puts "customization file: $env(MGC_CALIBRE_CUSTOMIZATION_FILE)"
   }
}
```

## External Trigger in SKILL for Cadence Virtuoso

The following example shows an external pre-execution trigger in the SKILL language for the Cadence Virtuoso design environment. This procedure can be defined in your *.cdsinit* file. The trigger defines a runset using an environment variable.

See "External Trigger Definitions in Cadence Virtuoso" on page 676 for information on defining the trigger.

```
procedure( mgc_start_calibre_trigger( run_type win)

   printf( "Starting %s run on top cell in %L\n" run_type win)

   ;Set Calibre Interactive Runset File
   setShellEnvVar("MGC_CALIBRE_DRC_RUNSET_FILE=/home/drc.runset")
   printf( "Using Calibre Interactive DRC Runset: %s/n" strcat(
      getShellEnvVar("MGC_CALIBRE_DRC_RUNSET_FILE")))

   ); end procedure
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

When the trigger is loaded, the following message appears in the CIW window:

```
Starting drc run on top cell in window: 2
Using Calibre Interactive DRC Runset: /home/drc.runset
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Chapter 13
# Run Control Setup and Licensing

The following sections in this chapter provide important information about setting up Calibre jobs and choosing a licensing behavior.

**Specifying the Host Processor and Run Options** . . . . . **681**

**Computer Cluster Options** . . . . . **686**
Using the Spectrum LSF Option. . . . . 686
Example Spectrum LSF Configurations . . . . . 690
Using the Univa Grid Engine Option . . . . . 691
Using the Other Cluster Option . . . . . 693
Replaceable Parameters for the Other Cluster Queue Command. . . . . 694

**Configuring for Distributed (MTflex) Execution** . . . . . **696**

**Configuring the Remote Environment** . . . . . **699**

**Setting Licensing Options** . . . . . **701**

**Reference for Run Control Options** . . . . . **703**
Hyperscaling Mode. . . . . 703
Hyperscale Compare Option in Calibre Interactive nmLVS . . . . . 703
Hyperscale Pathchk Option . . . . . 703
Remote Data Server (RDS) Option . . . . . 704
Hyperscale Remote Option. . . . . 704
Backup Data Option . . . . . 704

## Specifying the Host Processor and Run Options

Calibre Interactive allows you to run Calibre jobs on local, remote, and distributed resources. You can choose single threaded, multithreaded, or distributed execution with a choice of options such as number of CPUs and hyperscaling.

**Note**

Although Calibre Interactive DRC, DFM, LVS, PERC, and PEX offer nearly identical run control options, the Graphical User Interface (GUI) for the Run Control pane varies slightly with each tool.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Prerequisites

- For runs using remote hosts, configure your environment to allow rsh to execute your Calibre job remotely. This may include creating and editing a .rhosts file in your home directory.

## Procedure

1. Click the **Run Control** button on the left panel of the Calibre Interactive GUI.
2. Click the **Performance** tab.
3. Choose the host processor or cluster option in the "Run Calibre on" dropdown list from the options in the following table.

**Table 13-1. Run Calibre On Options in the Run Control Pane**

| Selection | Description |
|---|---|
| Local Host | Calibre runs on the computer currently running Calibre Interactive. |
| Remote Host | Calibre runs on a remote computer.<br>1. Type the name of the remote host in the text entry field and press Enter.<br>Basic information about the remote host is displayed in the Host Information field.<br>2. Configure the remote environment as described in "Configuring the Remote Environment" on page 699. |
| Platform LSF | Use the IBM® Spectrum™ LSF® (Load Sharing Facility) distributed computing solution. See "Using the Spectrum LSF Option" on page 686. |
| Univa Grid Engine | Use the Univa® Grid Engine® batch-queuing system. See "Using the Univa Grid Engine Option" on page 691. |
| Other Cluster | Use a custom computer cluster. See "Using the Other Cluster Option" on page 693. |

4. Select execution options in the "Run Calibre" section.

**Table 13-2. Run Calibre Execution Options in the Run Control Pane**

| Selection | Description |
|---|---|
| **Processing mode** (from dropdown list) | |
| Single-Threaded | Use single-threaded execution. This is the default. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 13-2. Run Calibre Execution Options in the Run Control Pane (cont.)**

| Selection | Description |
|---|---|
| Multi-Threaded | Use multithreaded execution on a single host computer. Select the number of CPUs to use in the entry fields. |
| Distributed (MTflex) | Use your existing computer network as a distributed computing platform with Calibre® MTflex™. Follow the setup procedure in "Configuring for Distributed (MTflex) Execution" on page 696. |
| **Execution options**<br>The available execution options depend on the choice of processing mode. | |
| Number of CPUs to use for <*app*> | Specifies the number of processors in multithreaded (MT) mode (the value for the -turbo command line option). |
| for LITHO<br>(Calibre nmDRC and DFM only) | Specifies the number of processors to use for RET and MDP applications in multithreaded (MT) mode (the value for the -turbo_litho command line option). |
| Synchronize Min CPUs in Host Select table<br>(Multithreaded Platform LSF and Univa Grid Engine runs only) | Forces the CPU specification in the host selection table to be the same as the number of CPUs specified for "Number of CPUs to use for <*app*>." For DRC and DFM runs, the value is set to the greater of the "Number of CPUs to use for <*app*>" and "for LITHO" values. |
| Halt <*app*> if licenses for number of CPUs cannot be obtained | As described. Adds the -turbo_all command line option. Only available for "Multi-Threaded" runs. |
| Number of hierarchical LVS licenses for PERC<br>(Calibre PERC only) | Specifies the "-lvs_supplement *num_processors*" command line option, which uses Calibre nmLVS-H licenses for the circuit extraction portion of the run.<br>This option is only available for multithreaded runs with "Halt PERC if licenses for number of CPUs cannot be obtained" unchecked. |
| Hyperscale | Use the -hyper command line switch. See "Hyperscaling Mode" on page 703. Recommended.<br>The hyperscale option is available for "Multi-Threaded" and "Distributed (MTflex)" runs. The hyperscale option is not available for PEX runs. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 13-2. Run Calibre Execution Options in the Run Control Pane (cont.)**

| Selection | Description |
|---|---|
| Hyperscale Compare<br>(Calibre nmLVS only) | Use the "-hyper cmp" command line switch in Calibre nmLVS-H. The "Hyperscale" option must also be enabled.<br>See "Hyperscale Compare Option in Calibre Interactive nmLVS" on page 703. |
| Hyperscale PathChk<br>(Calibre nmLVS only) | Use the "-hyper pathchk" command line switch in Calibre nmLVS-H. The "Hyperscale" option must also be enabled.<br>See "Hyperscale Pathchk Option" on page 703. |
| Remote Data Server (RDS)<br><br>(Calibre nmDRC, Calibre nmLVS, Calibre DFM, and Calibre PERC runs) | Use the -remotedata command line switch. See "Remote Data Server (RDS) Option" on page 704 for details.<br>"Remote Data Server" cannot be specified with FastXOR runs. |
| Backup Data | Use the -recoverremote command line switch to specify recovery from a remote host failure. This option is valid only when "Remote Data Server" is enabled. See "Backup Data Option" on page 704. |
| Hyperscale Remote | Use the "-hyper remote" command line switch and option to improve the performance and scaling of hyperscaling. This option is valid only when "Remote Data Server" is enabled. See "Hyperscale Remote Option" on page 704.<br>In Calibre nmLVS and Calibre PERC, the "Hyperscale Remote" option cannot be specified with "Hyperscale Compare." |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 13-2. Run Calibre Execution Options in the Run Control Pane (cont.)**

| Selection | Description |
|---|---|
| Connect to Remote Hosts<br>(Calibre nmDRC only) | Specify when to connect to remote hosts in HDB (hierarchical database) construction. Only available when the processing type is set to Distributed (MTFlex).<br>• At later stages of HDB construction — Uses the "-hdbflex" command line switch to specify that most of the HDB construction is to be performed by the local host, which does not connect to remotes until the later stages of HDB construction.<br>• In parallel with HDB construction — Uses the "-hdbflex_acquire" command line switch to specify to acquire resources in parallel with the HDB construction process. |

## Related Topics

Reference for Run Control Options



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Computer Cluster Options

Large Calibre jobs may benefit from the performance of a computer cluster (also known as load balancing distributed resources). Calibre Interactive offers an interface for setting up execution of Calibre jobs on IBM® Spectrum™ LSF® and Univa® Grid Engine®, along with a generic interface for other distributed computing solutions.

The most important consideration when configuring a computer cluster for running a Calibre job is to take care not to overload or overrun the primary or remote hosts. You can manage this concern by being familiar with the configuration options of the computer cluster. It is particularly important to consult both the documentation for your distributed solution and your system administrator prior to testing your Calibre Interactive computer cluster configuration.

Initial setup tasks and configurations for each type of cluster vary. In some cases you need to take configuration steps beyond the default setup for a satisfactory distributed run.

**Tip**

Environment variables set using the instructions in the section "Viewing and Editing Environment Variables" on page 130 are available to the cluster host in any computer cluster solution.

**Note**

You can Ctrl-left-click in any host table to display the shell script that will execute based on the current configuration.

**Using the Spectrum LSF Option** . . . . . 686
**Example Spectrum LSF Configurations** . . . . . 690
**Using the Univa Grid Engine Option** . . . . . 691
**Using the Other Cluster Option** . . . . . 693
**Replaceable Parameters for the Other Cluster Queue Command** . . . . . 694

# Using the Spectrum LSF Option

You can specify that your Calibre Interactive job runs on the IBM Spectrum LSF computing system.

Spectrum LSF (Load Sharing Facility) is a distributed computing solution and requires a commercial license. For complete details on configuring this environment, refer to the products documentation.

## Prerequisites

- Spectrum LSF must be installed on the host machine and on every machine that you may potentially query or submit jobs to.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- For runs using remote hosts, configure your environment to allow rsh to execute your Calibre job remotely. This may include creating and editing a .rhosts file in your home directory.

## Procedure

1. Click the **Run Control** button on the left panel of Calibre Interactive.
2. In the **Performance** tab, choose Platform LSF from the "Run Calibre on" dropdown list.
3. Choose the setting for "Interactive Job Submission":
   - **Enabled** — (default) Submit the Spectrum LSF job with the -I option

     ```
     bsub -I <options>
     ```

   - **Disabled** — Submit the Spectrum LSF job without the -I option

     ```
     bsub <options>
     ```

4. Specify the "Submission Host" if it is not the same host that Calibre Interactive is running on.

   If localhost is specified as the submission host, then remote submission does not take place, and the LSF job is submitted by localhost.

   The submission command obeys the remote shell and remote user name settings on the Run Control pane on the **Environment > Misc** sub-tab.

   If the environment variable LSF_ENVDIR is defined in the localhost environment or the Calibre Interactive session, then the following line is added to the script executed by Calibre Interactive:

   ```
   .$LSF_ENVDIR/profile.lsf
   ```

   so that LSF settings are available in the submission host.

5. Choose Run Calibre options for your job. For details refer to Table 13-2 in the topic "Specifying the Host Processor and Run Options" on page 681.

   In particular, for multithreaded runs, you can specify "Synchronize Min CPUs in Host Select table" to force Min CPUs in the host selection table to be the same as the number of CPUs specified for "Number of CPUs to use for <*app*>." For DRC and DFM runs, the Min CPUs value is set to the greater of the "Number of CPUs to use for <*app*>" and "for LITHO" values.

6. Specify host information.

   Calibre Interactive displays the Host Selection table as shown in the next figure for multithreaded execution. If you specify to run Calibre in "Distributed (MTflex)" mode, both a primary host selection table and a remote hosts selection table are displayed.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If you select Platform LSF and the LSF resource is unavailable, Calibre Interactive prints an error message to the shell and the Matching Hosts column displays 0.

**Figure 13-1. Spectrum LSF Dialog Box**

a. Use the Min CPUs, Type, and Memory columns in the Host Selection table, as shown in the following table, to define the filter criteria that you want used to determine the list of available hosts on your network. The table shows the columns displayed for multithreaded mode; a slightly different set of columns is displayed for MTflex mode.

**Table 13-3. Resource Option Variables for Spectrum LSF**

| Column Name | Filters on | Corresponding Substitution Variable |
|---|---|---|
| Min CPUs | Minimum number of available CPUs for each host.<br>This entry cannot be edited if “Synchronize Min CPUs in Host Select table” is checked. | %c |
| Type | Machine type. | %o |
| Memory | Minimum amount of memory available on each host in megabytes. | %m |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 13-3. Resource Option Variables for Spectrum LSF (cont.)**

| Column Name | Filters on | Corresponding Substitution Variable |
|---|---|---|
| N/A | Primary host name. (Only available as a resource option for a remote host.) | %M |

b. After you enter your criteria in these three columns, click in the Matching Hosts column (below the column title) to query for available hosts matching your criteria. The number of hosts that match your specified criteria display in the Matching Hosts column.

**Tip**

You can right-click in the Matching Hosts column (below the column title) to view additional information about available hosts.

c. Click in the Resource Options column (below the column title) and enter any additional custom resource options that you want in addition to those already specified in the first three columns of the table. These options represent arguments for the Spectrum LSF resource requirement switch, -R.

For examples of resource options specification statements, see "Example Spectrum LSF Configurations" on page 690. For additional information on LSF syntax, please refer to the Spectrum LSF production documentation.

If the Resource Options field is not specified, the host is selected based on the Min CPUs, Type, and Memory columns only; this is the default behavior.

d. Enter options unrelated to resource settings in the Submit Options column. For information on possible options, refer to the Spectrum LSF production documentation.

For multithreaded (MT) runs only, you can use the %b parameter in the Submit Options column. %b is substituted with the setting from "Number of CPUs to use for <*app*>" on the Run Control pane of Calibre Interactive; "All" is substituted for %b if "Number of CPUs to use for <*app*>" is "All". The %b parameter can be used so that the number of CPUs used for the job matches what is specified for the Calibre run with the -turbo option.

7. Click the **Environment** tab and configure the remote options appropriate for your run. For an example remote setup configuration, see "Configuring the Remote Environment" on page 699.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Results

When the number of CPUs is specified, the Spectrum LSF job is submitted using the -n option for both primary and remote hosts to obtain an even distribution of CPUs. The -n option should not be used in the Submit Options column.

The LAUNCH CLUSTER command is used to specify connection parameters; see "LAUNCH CLUSTER" in the *Calibre Administrator's Guide*.

### Related Topics

Computer Cluster Options

Configuring for Distributed (MTflex) Execution

Configuring the Remote Environment

Setting Licensing Options

Viewing and Editing Environment Variables

# Example Spectrum LSF Configurations

Custom options specified in the Resource Options column have priority over options specified in the individual table columns. The following examples of resource option statements illustrate the use of both custom filters and standard column filters.

### Example 1: Variables Inherit Column Filters

If you leave the Resource Options table cell completely blank, the following substitution occurs in which each of the values specified in the Min CPUs, Type, and Memory columns are substituted for their respective variable as described in Table 13-3 in the topic "Using the Spectrum LSF Option" on page 686:

```
-R "select[ncpus>=%c && type==%o && maxmem>=%m]"
```

#### Example 2: Literal Values Override Column Filters

If you enter an options statement in the Resource Options field that contains literal (non-variable) values, these literal values take priority over any values specified in the Min CPUs, Type, and Memory columns. In this case, only hosts with at least 4 CPUs and a minimum memory of 1024 MBytes are returned in the Matching Hosts field.

```
-R "select[ncpus>=4 && maxmem>=1024]"
```

#### Example 3: Variable Defines an Additional Filter

If you enter an options statement in the Resource Options field that contains both literal values and variables, each variable takes the value of its corresponding column value if there is one.

In this case, the Min CPUs, Type, and Memory column values are substituted into the variables. The remaining tmp option (Disk Space in */tmp* in Megabytes) retains its value of 1024.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
-R "select[ncpus>=%c && type==%o && maxmem>=%m && tmp>=1024]"
```

#### Example 4: Performance is Optimized for Remote Hosts

You can attempt to gain the best performance by configuring Spectrum LSF to use the -x Submit option; this option requires remote hosts on your distributed run to be exclusively dedicated to your Calibre job. However, with this option, if a host is currently running another job, it is not considered available as a remote host when queried.

An alternate method of optimizing the performance of a distributed run is to use the span option to force LSF to use all available remote hosts as shown in this example.

```
-R "select[ncpus>=%c && type==%o && maxmem>=%m && hname!=%M] &&
span[ptile=800]"
```

# Using the Univa Grid Engine Option

The Univa Grid Engine is an open source batch-queuing system. For complete details on configuring this environment, refer to the product's documentation.

## Prerequisites

- For runs using remote hosts, configure your environment to allow rsh to execute your Calibre job remotely. This may include creating and editing a .rhosts file in your home directory.

## Procedure

1. Click the **Run Control** button on the left panel of Calibre Interactive.
2. Click the **Performance** tab.
3. Select "Univa Grid Engine" from the "Run Calibre on" dropdown list.
4. Choose the setting for "Interactive Job Submission":
   - **Enabled** — (default) The job is submitted in interactive mode:

     ```
     qrsh -noshell <options>
     ```

   - **Disabled** — The job is submitted in non-interactive mode:

     ```
     qsub <options>
     ```

5. Select the run mode and options:
   a. Choose a run mode from the dropdown list for "Run Calibre": "Single-Threaded," "Multi-Threaded," or "Distributed (MTflex)."
   b. Select additional execution options. The available options depend on the setting selected for "Run Calibre."



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

In particular, for multithreaded runs, you can specify "Synchronize Min CPUs in Host Select table" to force CPU Number in the host selection table to be the same as the number of CPUs specified for "Number of CPUs to use for <*app*>." For DRC and DFM runs, the CPU Number value is set to the greater of the "Number of CPUs to use for <*app*>" and "for LITHO" values.

See "Configuring for Distributed (MTflex) Execution" on page 696, "Specifying the Host Processor and Run Options" on page 681, and "Reference for Run Control Options" on page 703.

6. Use the CPU Number, Architecture, and Memory columns to define filter criteria for hosts available on your network. Note that CPU Number filters on the *exact* number of available CPUs on each host. The Memory column filters on the minimum amount of memory available on each host. The grid systems use the unit bytes by default. Append a "M" to an integer value to specify megabytes.

7. Refresh the list matching the criteria defined in Step 6 by clicking in the Matching Hosts column below the column title. View additional information about available hosts by right-clicking in the in the Matching Hosts column.

8. Enter options in the Resource Options column as necessary. These options represent arguments for the grid resource requirement switch -l.

   Common resources are given their own columns on the left side of the Primary Host Selection table. The values can be substituted into the Resource Options cell using variables as follows:

   %c — Replace with the value of CPU Number

   %o — Replace with the value of Architecture

   %m — Replace with the value of Memory

   If you leave the Resource Options table cell completely blank, the following substitution occurs:

   ```
   -l "num_proc=%c,arch=%o,mem_total=%m"
   ```

9. Enter options unrelated to resource settings in the Submit Options column.

   For multithreaded (MT) runs only, you can use the %b parameter in the Submit Options column. %b is substituted with the setting from "Number of CPUs to use for <*app*>" on the Run Control pane of Calibre Interactive; "All" is substituted for %b if "Number of CPUs to use for <*app*>" is "All". The %b parameter can be used when specifying a parallel environment so that the number of CPUs used for the job matches what is specified for the Calibre run with the -turbo option.

10. Choose Run Calibre options for your job. For details refer to Table 13-2 in the topic "Run Calibre Execution Options in the Run Control Pane."



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

11. Click the **Environment** tab and configure the remote options appropriate for your run. For an example remote setup configuration, see Figure 13-4 on page 700 in the topic "Configuring the Remote Environment" on page 699.

## Examples

The following figure shows the **Performance** tab for Univa Grid Engine configuration for Distributed (MTflex) operation.

**Figure 13-2. Univa Grid Engine Configuration**

## Related Topics

Computer Cluster Options

Configuring for Distributed (MTflex) Execution

Configuring the Remote Environment

Setting Licensing Options

# Using the Other Cluster Option

The "Other Cluster" option provides a generic interface for configuring distributed computer solutions.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Prerequisites

- For runs using remote hosts, configure your environment to allow rsh to execute your Calibre job remotely. This may include creating and editing a .rhosts file in your home directory.

### Procedure

1. Click the **Run Control** button on the left panel of Calibre Interactive.
2. In the **Performance** tab, select "Other Cluster" from the "Run Calibre on" dropdown list.
3. Enter the appropriate command for your cluster in the Queue Command text field.

   See "Replaceable Parameters for the Other Cluster Queue Command" on page 694 for a list of the replaceable parameters that can be used in the queue command.
4. (Optional) Set the "Synchronous" option as follows:

| Synchronous | Behavior |
|---|---|
| On | Calibre Interactive waits for the Calibre run to complete. Calibre RVE and reports are displayed when the run is finished if appropriate options are set. |
| Off | Calibre Interactive executes the cluster command as a background process. The run transcript, Calibre RVE, and reports are *not* displayed when the run finishes. |

5. Choose Run Calibre options for your job. For details refer to Table 13-2 in the topic "Specifying the Host Processor and Run Options" on page 681.
6. Click the **Environment** tab and configure the remote options appropriate for your run. For an example remote setup configuration, see Figure 13-4 in the topic "Configuring the Remote Environment" on page 699.

### Related Topics

Computer Cluster Options

Setting Licensing Options

Configuring the Remote Environment

## Replaceable Parameters for the Other Cluster Queue Command

Several replaceable parameters are available for use in the Queue Command field for an "Other Cluster" run.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table 13-4. Parameters for Other Cluster Queue Command

| Parameter | Description | Run types valid for |
|---|---|---|
| **General parameters** | | |
| %d | Run directory | All |
| %u | User name<br>(Run Control pane, **Environment** > **Misc** sub-tab) | All |
| %m | Value of MGC_HOME<br>(Run Control pane, **Environment** > **MGC_HOME** sub-tab) | All |
| %L | Licensing variable name<br>(Run Control pane, **Environment** > **Licensing** sub-tab) | All |
| %l | Licensing variable value<br>(Run Control pane, **Environment** > **Licensing** sub-tab) | All |
| %E | Name-value pairs of environment variables | All |
| **Calibre command parameters** | | |
| %c | Calibre command (without options) | All |
| %o | Calibre command line options | All |
| %o1 | Command line options for PHDB generation in parasitic extraction | PEX only |
| %o2 | Command line options for parasitic database (PDB) generation in parasitic extraction | PEX only |
| %o3 | Command line options for the netlist formatting (-fmt) step in parasitic extraction | PEX only |
| %C | Calibre command with options (%c +%o) | All |
| %C1 | Calibre command with options for PHDB generation in parasitic extraction (%c + %o1) | PEX only |
| %C2 | Calibre command with options for parasitic database generation in parasitic extraction (%c + %o2) | PEX only |
| %C3 | Calibre command with options for netlist formatting step in parasitic extraction (%c + %o3) | PEX only |
| %k | Cluster control script filename | All |
| %k1 | Cluster control script filename for PHDB generation step in parasitic extraction | PEX only |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table 13-4. Parameters for Other Cluster Queue Command (cont.)**

| Parameter | Description | Run types valid for |
|---|---|---|
| %k2 | Cluster control script filename for parasitic database generation step in parasitic extraction | PEX only |
| %k3 | Cluster control script filename for netlist formatting step in parasitic extraction | PEX only |

# Configuring for Distributed (MTflex) Execution

Calibre® MTflex™ allows you to use your existing network as a low-cost distributed computing platform.

For more information about Calibre MTflex, see the *Calibre Administrator's Guide*.

## Prerequisites

- Configure your environment to allow rsh to execute your Calibre job remotely. This may include creating and editing a .rhosts file in your home directory.
- (Optional) If necessary, set one of the following environment variables to prevent "Word too long" errors from the remote host. This error is seen when the command line length exceeds the limit set in the shell environment. By default, Calibre Interactive uses the C shell (/bin/csh) to launch the MTflex primary script. You can specify the Bash shell instead, which allows a longer command line, by defining one of the following environment variables:
  - CALIBRE_CI_REMOTE_COMMAND_USE_BASH

    When set, use the Bash shell to launch the MTflex primary script.
  - CALIBRE_CI_REMOTE_COMMAND_LSF_USE_BASH

    When set, use the Bash shell to launch the MTflex primary script for Spectrum LSF runs only.
  - CALIBRE_CI_REMOTE_COMMAND_GRID_USE_BASH

    When set, use the Bash shell to launch the MTflex primary script for Univa Grid Engine runs only.

  In most cases the long command line is due to a long $path variable.

## Procedure

1. Click the **Run Control** button on the left panel of Calibre Interactive.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Select the host for Calibre execution in the "Run Calibre on" dropdown list (Local Host, Remote Host, Platform LSF, Univa Grid Engine, or Other Cluster). See "Specifying the Host Processor and Run Options" on page 681.
3. Select "Distributed (MTflex)" in the "Run Calibre on" dropdown list.
4. Configure the remote hosts with the GUI Remote Host table.

**Note**

If you are running on Local Host or Remote Host, you have the option of providing a configuration file to specify the Calibre MTflex remote hosts; the file is used as the parameter for the -remotefile command line argument.

Select "Specify File" in the dropdown list at the top of the remote hosts table, then specify the configuration file. (The default setting for the dropdown list is "Specify from GUI.") Go to Step 6.

Use the action buttons to add or remove hosts. You can also right-click in the Remote Hosts table to get a popup menu. These menu selections are available:

Add Host — Adds hosts to the parameter list for the calibre -remote command line switch.

Delete Host and Delete All Hosts — As indicated.

Load Hosts File — Opens a file browser to select the configuration file for the -remotefile command line switch.

Save Hosts File — Save a configuration file.

View Hosts File — View the configuration file (Ctrl-click in the host table).

For more information, see -remotefile and "About the Configuration File" in the *Calibre Administrator's Guide*.

See the following figure for an example Calibre MTflex configuration on a local host.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure 13-3. Example Calibre MTflex Configuration With Local Host**

See "Using the Univa Grid Engine Option" on page 691 for an example Calibre MTflex configuration on an Univa Grid Engine.

5. If you specified the remote hosts using the remote hosts table, you can view and edit additional parameters for each remote host by enabling the "Details" checkbox at the top right of the host table. The columns displayed depend on the selection for "Run Calibre on."

   - **Local Host** or **Remote Host** — The additional columns correspond to the parameters in the Remote Host statement; see "REMOTE HOST" in the *Calibre Administrator's Guide*.

     If there is no entry in the RSH column, then the remote shell specified on the **Environment > Misc** sub-tab is used for communication with the remote host. See "Configuring the Remote Environment" on page 699.

   - **Platform LSF** or **Univa Grid Engine** — The additional columns correspond to the parameters in the Launch Cluster statement; see "LAUNCH CLUSTER" in the *Calibre Administrator's Guide*.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

For Calibre MTflex runs, Calibre Interactive does not include the CPU count argument to the -turbo or -turbo_litho command line arguments. This is the recommended invocation and results in Calibre using the maximum number of CPUs available.

6. For Calibre nmDRC, Calibre nmLVS, and Calibre DFM runs, select “Hyperscale” (recommended).
7. For Calibre nmDRC runs, select “Remote Data Server” options. “Hyper Remote” is recommended for large designs. See the sections “Remote Data Server (RDS) Option” on page 704 and “Hyperscale Remote Option” on page 704.
8. Set other “Run Calibre” options. These options are described in “Specifying the Host Processor and Run Options” on page 681.

### Related Topics

Computer Cluster Options

Configuring the Remote Environment

Using the Univa Grid Engine Option

Tool-Specific Information for Distributed Calibre [Calibre Administrator's Guide]

# Configuring the Remote Environment

The selections in the **Environment** tab in Calibre Interactive allow you to set parameters such as executable paths and environment variables that control remote host execution.

Calibre Interactive makes the values of the remote user name, the MGC_HOME path, and the license file available as parameters for all Calibre MTflex operations.

### Prerequisites

- Configure your environment to allow rsh to execute your Calibre job remotely. This may include creating and editing a .rhosts file in your home directory.
- Install and configure Calibre to run on the remote computer.

### Procedure

1. Click **Run Control** on the left panel of Calibre Interactive.
2. Click the **Environment** tab.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

3. Click the **MGC_HOME** tab to specify MGC_HOME for the remote host. Specify one of the following:

   - **Local** — (Default) Use the same version of Calibre that is specified for Calibre runs on the local host.

   - **Specify** — Specify the path for MGC_HOME in the platform table. Click in the MGC_HOME column for the platform you want and enter the new setting in the text entry field, as shown in Figure 13-4.

**Figure 13-4. Remote MGC_HOME Table**

4. Click the **MGC_LIB_PATH** tab to specify MGC_LIB_PATH for the remote host. Specify one of the following:

   - **Current** — Use the current value.

   - **Specify** — Specify the path for MGC_LIB_PATH in the platform table. Click in the MGC_LIB_PATH column for the platform you want and enter the new setting in the text entry field, as shown in Figure 13-4 for the MGC_HOME table.

5. Click the **Licensing** tab.

   If needed, set “License File Name” and “License File Value” in the Remote Calibre section. See the section “Mentor Standard Licensing (MSL)” in the *Calibre Administrator’s Guide* for information on the license file.

   See “Setting Licensing Options” on page 701 for information on other licensing options.

6. Click the **Misc** tab.

   Specify the remote shell and remote user name, if needed. There are three options for the remote shell:

   - **Default —** Uses rsh, which may require a password or be otherwise restricted.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Secure —** Uses ssh, which uses host fingerprinting and key pairs. Creation of an ssh private key is recommended. When using a machine without rsh, ssh is automatically chosen.

  **Note**

  SSH_ASKPASS is used for ssh authentication. When using ssh, make sure the openssh-askpass package is installed; it is included with ssh. Without this package installed, host verification fails when the host is not in the known_hosts file and there is no ssh configuration file.

- **Specify —** Uses a user-defined shell. When selected, enter the path to the shell.

**Note**

For Calibre MTflex configurations with "Run Calibre on" specified as Local Host or Remote Host, the remote shell used for communication with each remote host is specified in the RSH column of the remote hosts table on the **Performance** tab. If the RSH column is not displayed, enable the "Details" checkbox to display all columns. If there is no entry in the RSH column, the remote shell specified on the **Environment** > **Misc** tab is used. See "Configuring for Distributed (MTflex) Execution" on page 696.

### Related Topics

REMOTE HOST [Calibre Administrator's Guide]

Configuring for Distributed (MTflex) Execution

Computer Cluster Options

Specifying the Host Processor and Run Options

# Setting Licensing Options

You can control licensing options using Calibre Interactive.

### Procedure

1. Select the Run Control pane.
2. Click the **Environment** tab and then the **Licensing** sub-tab.
3. Choose application-dependent license options:
   - Calibre nmDRC and Calibre nmLVS
     - **Acquire CB license for RVE when using Calibre CB**

       Select this option if you selected a run type of "Calibre-CB (flat)" on the Calibre Interactive nmDRC Inputs pane or "Calibre CB" on the Calibre Interactive nmLVS Inputs pane.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- PEX

  Use these options if you are performing a flat extraction run. The Calibre CB license options for Calibre nmLVS and Calibre xRC should be selected if SOURCE names are requested in a PEX run of Calibre Interactive and you only have the Calibre Cell/Block (CB) license.

  - **Use Calibre-xRC CB license** — Can only be used for flat transistor or gate-level extraction, and on designs containing up to 100,000 transistors.
  - **Use Calibre-CB license for LVS** — The -flatten command line option is used in the Calibre nmLVS invocation.
  - **Use Calibre-CB license for RVE**

  Also see "Extracting a Block Using CB" in the *Calibre xRC User's Manual*.

4. If needed, set licensing options for remote hosts in the "Remote Calibre" section.

   You can set the "License File Name" and "License File Value." See the section "Mentor Standard Licensing (MSL)" in the *Calibre Administrator's Guide* for information on the license file. See "Configuring the Remote Environment" on page 699 for other options related to setting up the remote environment.

5. Enable or disable licensing queue behavior. If enabled, select one of the following options:

   - **Disable Queuing** — (-nowait) The Calibre run is aborted if a license is not immediately available.
   - **Wait for License** — (-wait <time>) The run is placed into a queue if a license is not immediately available.
     - **Wait for ____ minutes** — Select this option to specify a time parameter for the -wait argument in the Calibre command line. If a wait time is not specified, Calibre waits forever in the queue for a license.
   - **Retry for License** — (-lmretry loop) Retry, or loop, until a license is available, subject to the following options:
     - Retry for: ____ minutes (MAXRETRY:<minutes>)
     - Wait for: ____ seconds between retries (INTERVAL:<seconds>)
     - Retry even when license errors occur (RETRY_ON_ERROR)

   When disabled, the Calibre command line does not include any licensing arguments.

## Related Topics

Calibre Licensing Command Line Options [Calibre Administrator's Guide]

Configuring the Remote Environment



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Reference for Run Control Options

This section provides detailed information on run control options such as hyperscaling, backup data, and remote data server (RDS).

**Hyperscaling Mode** . . . . . 703
**Hyperscale Compare Option in Calibre Interactive nmLVS** . . . . . 703
**Hyperscale Pathchk Option** . . . . . 703
**Remote Data Server (RDS) Option** . . . . . 704
**Hyperscale Remote Option** . . . . . 704
**Backup Data Option** . . . . . 704

## Hyperscaling Mode

Hyperscaling is a mode of multithreaded operations that improves performance and scalability of CPU usage. It is enabled through the "Hyperscale" option on the **Performance** tab of the run Control pane. It enables the same functionality as the -hyper command line switch for the Calibre batch tools.

Hyperscaling functions in both multithreaded and distributed (Calibre MTflex) modes. You should use hyperscaling unless you have good reason not to. It requires no extra licenses other than what is needed for MT or Calibre MTflex execution. Additional details are available in the *Calibre Administrator's Guide* and the *Calibre Verification User's Manual*.

The hyperscale option is not available for PEX runs in Calibre Interactive.

## Hyperscale Compare Option in Calibre Interactive nmLVS

When "Hyperscale Compare" is enabled, Calibre nmLVS-H runs certain portions of LVS comparison concurrently with LVS circuit extraction. This reduces the run time for a hierarchical LVS run that requires both extraction and comparison.

Performance tests with this feature have reduced run times by up to 20 percent, while requiring up to 100 percent more memory. See "LVS Comparison" and "Hierarchical Comparison with Hyperscaling Transcript" in the *Calibre Verification User's Manual* for more information.

## Hyperscale Pathchk Option

When "Hyperscale PathChk" is enabled, Calibre nmLVS-H runs certain portions of LVS comparison concurrently with Pathchk layer operations in hierarchical databases. This option is useful if Pathchk operations take the majority of time in a hyperscaling run. This option increases memory usage.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Remote Data Server (RDS) Option

The Calibre Remote Data Server (RDS) technology is designed to reduce local (primary) memory requirements and improve Calibre MTflex performance by distributing data across remote hosts.

Remote Data Server is only available for Calibre nmDRC, Calibre nmLVS, Calibre DFM, Calibre xACT, and Calibre PERC runs; it is not available for FastXOR runs. It uses the -remotedata command line option.

### Related Topics

Remote Data Server [Calibre Administrator's Guide]

Specifying the Host Processor and Run Options

Hyperscale Remote Option

# Hyperscale Remote Option

Hyperscale Remote is an option that improves the performance and scaling of hyperscaling. Hyperscale Remote is available in “Distributed (MTflex)” mode when the Remote Data Server (RDS) Option is enabled. These options are not available for Calibre Interactive PEX.

Hyperscale Remote mode is most useful for large designs that require more than one hour to complete. A minimum configuration of four remote hosts with 24 remote CPUs connected to a local host (primary) is required; scaling can continue beyond 100 CPUs. Additional details are available in the *Calibre Administrator’s Guide* and the *Calibre Verification User’s Manual*.

### Related Topics

Remote Data Server (RDS) Option

Specifying the Host Processor and Run Options

# Backup Data Option

The Backup Data option enables recovery in the event of a catastrophic remote host failure; it uses the -recoverremote command line option. Backup Data is available in “Distributed (MTflex)” mode when the Remote Data Server (RDS) Option is enabled. These options are not available for Calibre Interactive PEX.

### Related Topics

Remote Data Server (RDS) Option

Specifying the Host Processor and Run Options



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Appendix A
# Interfacing Calibre Interactive to Layout and Schematic Viewers

The ability to communicate with industry standard design tools is an important feature of Calibre Interactive. This chapter discusses how to use Calibre Interactive with some of the design tool integrations. It also explains the setup procedure for the design tools that require custom installation and discusses special operating considerations.

The interface to several of the design tools is made automatically and requires no special setup actions or instruction. Such tools are not discussed in this chapter. Some design tools may require you to turn on toolbar or menu visibility. The complete list of design tools with a Calibre integration is given in the section "Integration to Design Tools" on page 26. For most design tools, the Calibre integration enables the interface to both Calibre RVE and Calibre Interactive.

Siemens products have a Calibre or Verification menu by default, and you use it to access Calibre Interactive and Calibre RVE. Siemens layout viewers that are not mentioned explicitly in this chapter operate similarly to Calibre DESIGNrev.

The following topics and design tools are covered in this chapter:


**Basic Interface** . . . . . 706
**Custom Menu Items in Design Tools** . . . . . 709
**Calibre DESIGNrev and WORKbench** . . . . . 721
**IC Station/Pyxis** . . . . . 723
**Siemens Tanner** . . . . . 724
**Cadence Virtuoso** . . . . . 725
**Cadence Composer** . . . . . 816
**Cadence Encounter** . . . . . 817
**Cadence Innovus** . . . . . 820
**Synopsys IC Compiler and Synopsys IC Compiler II** . . . . . 822
**Silvaco Expert** . . . . . 834
**Other Viewer Integrations** . . . . . 835





Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Basic Interface

Some interface concepts and actions are common to all integrated layout and schematic viewers, as explained in these sections:

**Calibre Interactive** . . . . . 706
**Setting the Socket Port in Calibre Interactive** . . . . . 707

## Calibre Interactive

To start Calibre Interactive from an integrated viewer, choose the desired interface from the menu, depending on your viewer.

Typical menu selections are the following:

- Calibre DESIGNrev family of viewers — **Verification** menu.
- Other viewers — **Calibre** or **Calibre** > **Verification** menu.

Calibre Interactive communicates with the layout viewer to enable automatic export of GDSII layout databases. Calibre Interactive imports the Primary Cell according to the setting on the Inputs pane. By default, the currently open cell in the layout viewer is selected for import.

A communications socket is set up automatically when you invoke Calibre Interactive through the layout viewer. The default socket number is 9189. See Setting the Socket Port in Calibre Interactive for instructions on manually setting a socket number.

**Tip**

You can define the templates Calibre Interactive uses to initialize GUI field values when Calibre Interactive is started from a layout viewer. Template settings are saved as part of the runset. Refer to “Templates for File Naming” on page 148 for information on templates.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setting the Socket Port in Calibre Interactive

Calibre Interactive uses certain default port numbers for socket communication with a connected layout or schematic viewer. The port numbers can also be set manually, as described here. Individual layout and schematic viewers may have additional methods for setting the port numbers; see the section for the specific viewer.

**Setting the Calibre Interactive Socket Port for a Layout Viewer** . . . . . 707
**Setting the Calibre Interactive Socket Port for a Schematic Viewer** . . . . . 708

## Setting the Calibre Interactive Socket Port for a Layout Viewer

You can set the socket port manually using the Calibre Interactive GUI, or using an environment variable. By default, Calibre Interactive uses socket port 9189 to communicate with the layout viewer. The tool automatically searches for an available socket if it cannot obtain the default socket. It initializes the first available socket between the numbers 5000 and 9999 and reports the socket number through a dialog box.

Some layout viewers, such as Calibre DESIGNrev, also have a GUI interface to set the socket port — see the instructions for the specific viewer.

### Prerequisites

- Calibre Interactive must be open if you are setting the socket number using the GUI.

### Procedure

Set the layout socket port with one of the following methods:

| To set the layout socket port with … | Do this … |
|---|---|
| Calibre Interactive | 1. Choose **Setup** > **Layout Viewer** in Calibre Interactive to open the Setup Layout Viewer dialog box.<br>2. Enter the port number in the “Socket Number” field.<br>3. Click **Connect**. |
| Environment variable | Set the MGC_CALIBRE_LAYOUT_SERVER environment variable as shown here for the C-Shell:<br>`setenv MGC_CALIBRE_LAYOUT_SERVER [<host>:]<port>`<br>The <host> parameter is set to localhost by default, but can be set to a different host. The <port> parameter must be a numerical value.<br>To disable socket communication to the layout viewer, set MGC_CALIBRE_LAYOUT_SERVER to “ ” (blank). The layout server then does not attempt to open the communication socket. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Connection to a Design Tool [Calibre RVE User's Manual]

Setting the Calibre RVE Socket Port for a Layout Viewer [Calibre RVE User's Manual]

## Setting the Calibre Interactive Socket Port for a Schematic Viewer

You can set the socket port manually using the Calibre Interactive GUI, or using an environment variable. By default, Calibre Interactive uses socket port 9199 to communicate with the schematic viewer. The tool automatically searches for an available socket if it cannot obtain the default socket. It initializes the first available socket between the numbers 5000 and 9999 and reports the socket number through a dialog box.

### Prerequisites

- Calibre Interactive must be open if you are setting the socket number through the GUI.

### Procedure

Set the schematic viewer socket port with one of the following methods:

| To set the schematic socket port with … | Do this … |
|---|---|
| Calibre Interactive | 1. Choose **Setup > Schematic Viewer** in Calibre Interactive to open the Setup Schematic Viewer dialog box.<br>2. Enter the port number in the “Socket Number” field.<br>3. Click **Connect**. |
| Environment variable | Set the MGC_CALIBRE_SCHEMATIC_SERVER environment variable as shown here for the C-Shell:<br>`setenv MGC_CALIBRE_SCHEMATIC_SERVER [<host>:]<port>`<br>The <host> parameter is set to localhost by default, but can be set to a different host. The <port> parameter must be a numerical value.<br>To disable socket communication to the schematic viewer, set MGC_CALIBRE_SCHEMATIC_SERVER to “ ” (blank). The schematic server then does not attempt to open the communication socket. |

### Related Topics

Connection to a Design Tool [Calibre RVE User's Manual]

Setting the Calibre RVE Socket Port for a Schematic Viewer [Calibre RVE User's Manual]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Custom Menu Items in Design Tools

You can add custom, or user-defined, menu items to supported design tools. The custom menu items are added to the integration menu.

The supported design tools are given in the following table.

**Table A-1. Supported Design Tools for Custom Menu Items**

| Design Tool | Integration Menu |
|---|---|
| Calibre DESIGNrev | Verification |
| Cadence Virtuoso | Calibre |
| Cadence Encounter | Calibre |
| Synopsys IC Compiler | Calibre |

The standard items in the Calibre integration menu open Calibre Interactive, Calibre RVE, or the Calibre setup dialog box. With menu customization you can add menu items to do the following:

- Open Calibre Interactive with a particular runset
- Open Calibre RVE to a particular results database
- Start a batch Calibre run
- Run a custom script
- Specify environment variables that are in effect for the command that is executed with the menu selection

Templates for File Naming are applied when you invoke Calibre Interactive from a custom menu command.

A custom menu example is shown in Figure A-1 for Calibre DESIGNrev. The menu customization file for this example is given in “Menu Customization Example” on page 719.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure A-1. Custom Menu Example for Calibre DESIGNrev**

The menu customization file is written in the Tcl language and a set of menu customization commands is provided. The following topics describe how to specify the menu customization file, the file format, and the available commands:

**Creating Custom Menus in Design Tools** . . . . . 710
**Menu Customization File** . . . . . 711
**Menu Customization Commands** . . . . . 712
**Calibre Integration Menu Labels and Backward Compatibility for Menu Customization Files** . . . . . 718
**Menu Customization Example** . . . . . 719

# Creating Custom Menus in Design Tools

The following procedure describes how to add custom menu items to the supported design tools.

See “Custom Menu Items in Design Tools” on page 709 for a list of the supported design tools.

## Procedure

1. Write the menu customization file using the syntax and commands described in “Menu Customization File” on page 711 and “Menu Customization Commands” on page 712.

   The following simple example adds one menu item which starts a Calibre Interactive batch run:

```
# This menu customization file adds a menu item to
# start a Calibre Interactive DRC run in batch mode

mgc_calibre_add_menu_item -type command -after "Run nmDRC" \
   -command_type DRC -label "Run DRC GUI batch" \
   -command "calibre -gui -drc -runset drc_runset -batch"
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Set the environment variable MGC_CALIBRE_VIEWER_MENU_CMDS to the pathname for the menu customization file.
3. If desired, set MGC_CALIBRE_ECHO_VIEWER_MENU_CMDS to echo custom menu commands to the layout viewer's transcript.
4. If desired, set MGC_CALIBRE_VIEWER_MENU_CMDS_SINGLE_TOOL to make custom menu items behave the same as built-in menu items when selected multiple times. When set, custom menu items redirect you to a previous window instead of opening a new window.
5. Start your design tool.

### Results

The new menu items are included in the **Verification** or **Calibre** menu when the design tool starts up.

### Related Topics

Menu Customization File

Menu Customization Commands

Menu Customization Example

Running Calibre Interactive in Batch Mode

## Menu Customization File

Input for: Custom Menu Items in Design Tools

The menu customization file must be valid Tcl syntax.

The following syntax rules apply:

- Comment lines start with '#'.
- Characters special to Tcl which appear in string literals, such as '$', must be escaped.
- Blank lines are ignored.

The location of the menu customization file is specified with the environment variable MGC_CALIBRE_VIEWER_MENU_CMDS.

### Related Topics

Tcl Library in Calibre [Calibre Administrator's Guide]



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Menu Customization Commands

Several commands are available for creating custom menus in supported design tools.

The syntax conventions used in the command descriptions are given in Table 1-3 on page 29. The following menu customization commands are supported:

| Command | Description |
|---|---|
| mgc_calibre_get_viewer_name | Returns the name of the design tool reading the menu customization file. |
| mgc_calibre_add_menu_item | Adds a menu item to the indicated menu (by default, the root **Calibre** or **Verification** menu). Pre-execution code that executes in the design tool environment can also be specified for a command menu item. |
| mgc_calibre_delete_menu_item | Removes an item from the indicated menu (by default, the root **Calibre** or **Verification** menu). |

Also see "Calibre Integration Menu Labels and Backward Compatibility for Menu Customization Files" on page 718.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# mgc_calibre_get_viewer_name

Returns the name of the design tool reading the menu customization file.

## Usage

**mgc_calibre_get_viewer_name**

## Arguments

None.

## Description

The command returns the value of the MGC_CALIBRE_LAYOUT_SERVER_NAME environment variable set by the design tool. The possible values are the same as those allowed for the runset variable cmnLayoutServerType. The list of supported design tools for menu customization and the corresponding return value for mgc_calibre_get_viewer_name is given in the following table.

**Table A-2. Return Value for mgc_calibre_get_viewer_name**

| Design Tool | Return Value |
|---|---|
| Calibre DESIGNrev | Mentor:Calibre |
| Cadence Virtuoso | Cadence:Virtuoso |
| Cadence Encounter | Cadence:Encounter |
| Synopsys IC Compiler | Synopsys:ICCompiler |

## Examples

The command may be used to combine viewer-specific menu customization code within a single file in the following manner:

```
if {[mgc_calibre_get_viewer_name] eq "Mentor:Calibre"} {
# DesignRev-specific menu modifications
} else {
# modifications for some other viewer
}
```

## Related Topics

Common Runset Options



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# mgc_calibre_add_menu_item

Adds a menu item to the indicated menu (by default, the root **Calibre** or **Verification** menu). Pre-execution code that executes in the design tool environment can also be specified for a command menu item.

## Usage

Add menu item tied to a command

**mgc_calibre_add_menu_item -type command -label *"label"***
**-command *"command"***
**-command_type** { **DRC** | **LVS** | **DFM** | **PEX** | **PERC** | **XACT** | **RVE** | **CUSTOM** }
[-code "*viewer_code*"]
[-view_type {layout | schematic}]
[-after *"label"* | -before *"label"*] [-parent *"label"*]
[{-env "*name=value*" }…] [{-unsetenv "*name*"} …]

Add parent menu item

**mgc_calibre_add_menu_item -type menu -label *"label"***
[-view_type {layout | schematic}]
[-after *"label"* | -before *"label"*] [-parent *"label"*]

Add separator line

**mgc_calibre_add_menu_item -type separator**
[-view_type {layout | schematic}]
[-after *"label"* | -before *"label"*] [-parent *"label"*]

## Arguments

- **-type** {**command** | **menu** | **separator**}

  A required argument and parameter which specifies the type of the menu item, where the allowed types are described as follows:

  - **command** — Adds a menu item connected to a command.
  - **menu** — Adds a menu item, such as a parent menu, not connected to a command. The -command, -command_type, and -env parameters are not used.

    This option is not available for Synopsys IC Compiler. Use "-type command" with the -parent option to specify a parent menu; the parent menu is created if it does not exist.
  - **separator** — Adds a separator line.

- **-label *"label"***

  The -label parameter indicates the label for the new menu item.

  You may place the ampersand character (&) before a character in the menu label to indicate that it is a keyboard shortcut and should be underlined. The ampersand is not required when



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

referring to the label in the -before or -after option. This feature is not supported in Cadence Virtuoso versions prior to 6.0.

- **-command "*command*"**

  Specifies the path to a command to be invoked when the menu item is selected, including parameters.

  Any token beginning with '$' is taken to be the name of an environment variable, delimited by any of '/', '#', '.', or ' '. If it matches any token given as a name in a -env parameter, the value of the environment variable specified in the -env parameter will replace it when the command is invoked.

  By default, when custom menu items are selected multiple times, a new window appears instead of redirecting you to the previously opened window. If desired, set the environment variable MGC_CALIBRE_VIEWER_MENU_CMDS_SINGLE_TOOL to make custom menu items behave the same as built-in menu items.

  See the description section for important information about invoking Calibre RVE.

- -code "*viewer_code*"

  Optional argument and parameter that specifies code to execute in the viewer environment before invoking the command tied to the menu item. See "Example 3" on page 717.

  The -env and -unsetenv options are not applied during execution of *viewer_code*. If the ***command*** is an empty string (-command "") or -command is omitted, only the *viewer_code* is executed.

- **-command_type** { **DRC** | **LVS** | **DFM** | **PEX** | **PERC** | **RVE** | **XACT** | **CUSTOM** }

  Specifies the type of command being issued so that the correct Calibre Interactive cell, library, and view templates are used.

  - **DRC**, **LVS**, **DFM**, **PEX**, **PERC**, **XACT**, or **RVE** — Command is for the indicated Calibre Interactive application or for Calibre RVE.
  - **CUSTOM** — Command is of arbitrary type.

- -view_type {layout | schematic}

  Optional argument and parameter specifying the viewer type for Cadence Virtuoso. The default is layout if this option is not specified.

- {-after "*label*" | -before "*label*"}

  Optional argument and parameter which specifies the insertion method and insertion location for the new menu item. If neither argument is present, the new menu item is added at the end of the parent menu, or at the end of the root Calibre menu if -parent is not specified.

  The *label* parameter must match the label of an existing menu item; it is not necessary, however, to include the trailing ellipses (…). The *label* parameter is case-sensitive. If *label* is not found, the new menu item is added at the end of the parent menu, or at the end of the root Calibre menu if -parent is not specified.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

"Run DRC" and "Run LVS" are allowed abbreviations, as explained in "Calibre Integration Menu Labels and Backward Compatibility for Menu Customization Files" on page 718.

- -env "*name=value*"

  An optional argument and parameter which adds a name/value pair to the environment of the invoked command. It may be specified multiple times.

  If MGC_HOME is set with the -env argument, CALIBRE_HOME should also be set with -env. CALIBRE_HOME takes precedence if both MGC_HOME and CALIBRE_HOME exist in the environment in which the new process is invoked; therefore, if only MGC_HOME is set with the -env argument, the version of Calibre run in the new process may not be what is expected.

- -unsetenv "*name*"

  An optional argument and parameter which unsets the environment variable given by *name* in the environment of the invoked command. It may be specified multiple times.

  When used with Cadence Virtuoso, this option is only supported for version 6.1.5 and later.

- -parent "*label*"

  An optional argument and parameter that specify which menu the new item is added to. The *label* parameter can specify a hierarchy of labels separated by '/' and must exactly match the label of an existing menu item.

  If this argument is omitted, the new item is added to the root **Calibre** or **Verification** menu.

## Description

This command adds a menu item to the indicated menu. If the parent menu (-parent argument) and location (-before and -after arguments) are not specified, the menu item is added at the end of the root **Calibre** or **Verification** menu. If the -before or -after label is specified but not found, the menu item is added at the end of the **Calibre** or **Verification** menu, or the end of the parent menu.

When calling Calibre RVE from a custom menu, the syntax should be the following:

```
\$MGC_HOME/bin/calibre -gui -rve <other_options>
```

where the -gui option is needed when invoking Calibre RVE from another program.

Templates for File Naming are applied when you invoke Calibre Interactive from a custom menu command.

## Examples

### Example 1

This example adds a menu item that can be used as a parent menu.

```
mgc_calibre_add_menu_item -type menu -after "Run LVS" -label "LFD..."
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Example 2

This example adds a menu item that starts Calibre RVE and opens a specific database. The -gui switch is needed when invoking Calibre RVE from another application.

```
mgc_calibre_add_menu_item -type command \
   -parent "LFD..." -label "32nm" \
   -command_type RVE \
   -command "\$MGC_HOME/bin/calibre -gui -rve -dfm 32nmDB"
```

### Example 3

This example specifies code to execute in the Cadence Virtuoso environment before starting Calibre Interactive PEX.

```
# Add a menu item to start Calibre Interactive PEX with a runset.
# Use the -code option to call the SKILL command
# mgc_rve_default_setup_file before executing the command
# tied to menu item.

mgc_calibre_add_menu_item -type command -label "pex_with_code" \
   -command_type PEX -command "calibre -gui -pex -runset adder.runset" \
   -code "mgc_rve_default_setup_file()"
```

### Example 4

This example adds and deletes menu items from the schematic view in Cadence Virtuoso. The label for the menu item uses the ampersand (&) character to designate a keyboard shortcut.

```
mgc_calibre_delete_menu_item -label "Run LVS" -view_type schematic

mgc_calibre_add_menu_item -type command -label "PERC &source based" \
   -view_type schematic -after "Run PERC" \
   -command_type PERC \
   -command "calibre -gui -perc -runset perc.runset.source"
```

## Related Topics

Menu Customization Example

Calibre Integration Menu Labels and Backward Compatibility for Menu Customization Files



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# mgc_calibre_delete_menu_item

Removes an item from the indicated menu (by default, the root **Calibre** or **Verification** menu).

## Usage

**mgc_calibre_delete_menu_item -label *"label"***
[-view_type {layout | schematic}] [-parent "*label*"]

## Arguments

- **-label " *label* "**

  A required argument and parameter which specify the label of the menu item to be deleted.

- -view_type {layout | schematic}

  Optional argument and parameter specifying the viewer type for Cadence Virtuoso. The default is layout if this option is not specified.

- -parent "*label*"

  An optional argument and parameter which specifies the parent menu of the item to remove. The *label* parameter can specify a hierarchy of labels separated by '/' and must exactly match the label of an existing menu item.

  If the -parent argument is omitted, the item is deleted from the root **Calibre** or **Verification** menu.

## Examples

```
mgc_calibre_delete_menu_item -label "Run DFM..."
```

## Related Topics

Calibre Integration Menu Labels and Backward Compatibility for Menu Customization Files

# Calibre Integration Menu Labels and Backward Compatibility for Menu Customization Files

The menu labels for the Calibre integration changed in some design tools with the 2014.2 release. Menu customization files written for previous releases are still read correctly if they use the earlier menu label in the -before and -after options to specify the placement of a custom menu item.

Starting with the 2014.2 release, the **Calibre** or **Verification** menu items for the Cadence Virtuoso, Cadence Encounter, Synopsys IC Compiler, and Calibre DESIGNrev integrations are the same for each design tool. Menu items that were previously "Run DRC" or "Run LVS" are now "Run nmDRC…" or "Run nmLVS…", respectively. In addition, the order of the menu items is now the same in all of these integrations.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

It is not necessary to update menu customization files written prior to the 2014.2 release to correct the menu labels used in the -before and -after options to the mgc_calibre_add_menu_item and mgc_calibre_delete_menu_item commands. The menu customization process correctly resolves the labels used in earlier releases, as shown in the following table:

| These menu labels: | Match this in 2014.2 and later |
|---|---|
| Run DRC<br>Run DRC…<br>Run nmDRC | Run nmDRC… |
| Run LVS<br>Run LVS…<br>Run nmLVS | Run nmLVS… |

In addition, trailing ellipsis marks (…) are ignored when matching labels using the -before and -after options. For example, this command:

```
mgc_calibre_add_menu_item -type separator -before "Start RVE"
```

adds a separator line before “Start RVE…”. Trailing ellipsis marks are also ignored when matching custom menu items added with the mgc_calibre_add_menu_item command.

# Menu Customization Example

This customization file works with any supported design tool. It deletes the **DFM** and **3DSTACK** menu items, and adds two new menu items and a submenu.

The results are shown in Figure A-1.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
# menu customization file
#
mgc_calibre_delete_menu_item -label "Run DFM"
mgc_calibre_delete_menu_item -label "Run 3DSTACK"

# add the LFD... menu item to root menu
# use & to designate F as a keyboard shortcut
mgc_calibre_add_menu_item -type menu -after "Run nmLVS" -label "L&FD..."
mgc_calibre_add_menu_item -type command -parent "LFD..." -label "28nm" \
   -command_type DFM \
   -command "\$MGC_HOME/bin/calibre -gui -dfm -runset 28nmrunset" \
   -env "OA_HOME=/oa/v2"
mgc_calibre_add_menu_item -type command -parent "LFD..." -label "32nm"\
   -command_type DFM \
   -command "\$MGC_HOME/bin/calibre -gui -dfm -runset 32nmrunset" \
   -env "OA_HOME=/oa/v1"

# add RVE submenu to new LFD... menu
mgc_calibre_add_menu_item -type menu -label "RVE..." -parent "LFD..."
mgc_calibre_add_menu_item -type command \
   -parent "LFD.../RVE..." -label "28nm" \
   -command_type RVE \
   -command "\$MGC_HOME/bin/calibre -gui -rve -dfm 28nmDB"
mgc_calibre_add_menu_item -type command \
   -parent "LFD.../RVE..." -label "32nm" \
   -command_type RVE \
   -command "\$MGC_HOME/bin/calibre -gui -rve -dfm 32nmDB"

# add LFD Reports menu item to root menu
mgc_calibre_add_menu_item -type command \
   -label "LFD Reports" -after "LFD" \
   -command_type CUSTOM \
   -command "/local/bin/RunLFDReport /path_to_report_dir"
mgc_calibre_add_menu_item -type separator -before "LFD"
mgc_calibre_add_menu_item -type separator -after "LFD Reports"
```

**Tip**

You can omit trailing ellipsis marks (…) from labels specified in the -before and -after options.

## Related Topics

Creating a One-Click Custom Menu Command With Skill Pre-Processing [Calibre Solutions for Physical Verification]

Calibre Integration Menu Labels and Backward Compatibility for Menu Customization Files



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre DESIGNrev and WORKbench

Calibre DESIGNrev and Calibre WORKbench include a Verification menu for starting Calibre Interactive, Calibre RVE, and some Calibre Pattern Matching applications.

The Verification menu includes the following selections:

**Table A-3. Verification Menu in Calibre DESIGNrev and WORKbench**

| Item | Description |
|---|---|
| Run nmDRC<br>Run DFM<br>Run nmLVS<br>Run PERC<br>Run PEX<br>Run xACT<br>Run 3DSTACK | Opens the corresponding Calibre Interactive application. |
| Run PM Capture<br>Run Pattern Matching | Opens Calibre Pattern Matching applications. See these topics in the *Calibre Pattern Matching User's Manual*:<br>"Capturing Patterns From a Layout"<br>"Running Pattern Matching From a Calibre Layout Viewer" |
| Start RVE | Starts Calibre RVE. |
| RVE/CI Setup | Opens a dialog for setting Calibre RVE and Calibre Interactive setup parameters, such as socket number, highlighting layers, highlight line width, and cell mapping. |
| RealTime | See the *Calibre RealTime Custom User's Manual*. |

You can add custom menu items to the **Verification** menu; see "Custom Menu Items in Design Tools" on page 709.

**Setting Socket Connections with Calibre DESIGNrev or Calibre WORKbench . . . . . 721**
**Calibre Interactive with Calibre DESIGNrev or WORKbench . . . . . 722**

## Setting Socket Connections with Calibre DESIGNrev or Calibre WORKbench

By default, Calibre Interactive and Calibre RVE use socket port 9189 for communication. The tool automatically searches for an available socket if it cannot obtain the default socket.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

You can manually specify a port number using the environment variable MGC_CALIBRE_LAYOUT_SERVER, as described in "Setting the Calibre Interactive Socket Port for a Layout Viewer" on page 707.

You can also specify the socket port by choosing **Verification > RVE/CI Setup** from Calibre DESIGNrev or Calibre WORKbench. Enter the socket number and click **Start**.

# Calibre Interactive with Calibre DESIGNrev or WORKbench

To start Calibre Interactive from Calibre DESIGNrev or Calibre WORKbench, choose Verification and choose the desired interface.

Refer to "Invoking Calibre Interactive From a Layout or Schematic Viewer" on page 64 for more information on how to invoke Calibre Interactive from Calibre DESIGNrev.

You can also define a trigger for Calibre Interactive in Calibre DESIGNrev; see "External Trigger Definitions in Calibre DESIGNrev" on page 675.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# IC Station/Pyxis

The IC Station product family changed its name to Pyxis with the v10.0 release of Pyxis, and this manual therefore refers to the Pyxis product. Pyxis has a **Calibre** pulldown menu with selections to open Calibre Interactive and Calibre RVE.

The **Calibre** menu in Pyxis is similar to the **Verification** menu for Calibre DESIGNrev, which is shown in Table A-3.

**Setting the software tree** — Calibre Interactive or Calibre RVE is called from in Pyxis by setting the CALIBRE_HOME variable to the desired software path. Pyxis will search for environment variables as follows:

1. If CALIBRE_HOME is defined, it is searched for the Calibre tree. If there is no Calibre tree in CALIBRE_HOME, then the MGC_HOME defined for the Pyxis session is searched. If still no Calibre tree can be found, you should specify the path to Calibre.
2. If CALIBRE_HOME is not defined, then the MGC_HOME defined for the Pyxis session is searched. If there is no Calibre tree in MGC_HOME, then you should specify the path to Calibre.

**Setting Socket Connections with Pyxis** . . . . . 723
**Using Calibre Interactive with Pyxis** . . . . . 723

## Setting Socket Connections with Pyxis

By default, Calibre Interactive and Calibre RVE use socket port 9189 to communicate with Pyxis Layout. The tool automatically searches for an available socket if it cannot obtain the default socket.

You can manually specify a port number using the environment variable MGC_CALIBRE_LAYOUT_SERVER, as described in "Setting the Calibre Interactive Socket Port for a Layout Viewer" on page 707.

You can also use following command from within Pyxis Layout to specify a different port without resetting the MGC_CALIBRE_LAYOUT_SERVER environment variable and restarting Pyxis Layout:

```
mgc_rve_init_socket <port_number>
```

Type the command in Pyxis Layout to initialize the specified socket port for communication.

## Using Calibre Interactive with Pyxis

Select the desired option from the **Calibre** menu to open the corresponding Calibre Interactive application.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Siemens Tanner

Calibre Interactive is integrated to the viewers Siemens Tanner S-Edit, a schematic viewer, and L-Edit, a layout design tool.

For more information on using these integrations with Calibre, refer to *Tanner and Linux-Based Tools Setup* in the Tanner documentation. For details on specific integrations, see the *L-Edit User's Manual* or the *S-Edit User's Manual*.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Cadence Virtuoso

The following sections describe the Calibre interface to Cadence Virtuoso:

**Creating an Interface to Cadence Virtuoso** . . . . . 725
**The Calibre Menu in Cadence Virtuoso** . . . . . 727
**Layout Export in Cadence Virtuoso** . . . . . 730
**Direct Read of Cadence Virtuoso OpenAccess Database** . . . . . 738
**Netlist Export in Cadence Composer** . . . . . 740
**Using the Calibre - Virtuoso Interface** . . . . . 745
**Setting Socket Connections with Cadence Virtuoso** . . . . . 747
**Calling SKILL Functions from Calibre Interactive** . . . . . 748
**Creating a Calibre View** . . . . . 749
**Setting the Bus Delimiter for Schematic Netlists** . . . . . 808
**Mapping Schematic Nets to Calibre View Nets for Simulation and Plotting** . . . . . 810
**Backannotation of Parasitic Values to the Schematic in Cadence Virtuoso** . . . . . 813
**Automating Calibre Interactive Runsets in the Cadence Design Environment** . . . . . 814

# Creating an Interface to Cadence Virtuoso

Calibre Interactive and Calibre RVE support Cadence Virtuoso versions 5.1, 6.1, 6.18, 12.1, 12.2, 12.3, and 18.1.

Beginning with the 2013.3 release, Calibre supports OpenAccess version oa22.43p006 by default. OpenAccess version 22.50 can be set with an environment variable. See "Providing OpenAccess Input" on page 90 and "FDI Environment Variables" in the *Calibre Layout Comparison and Translation Guide*.

**Note**

Cadence Virtuoso 6.18 and 18.1 are supported for layout export and highlighting only. These versions create OpenAccess v22.60 databases, which cannot be read directly by Calibre.

## Procedure

1. Set the environment variable CALIBRE_HOME or MGC_HOME to point to the Calibre software tree; see "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide*. Setting CALIBRE_HOME is preferred.

   Instructions in this section assume that CALIBRE_HOME is set; substitute MGC_HOME for CALIBRE_HOME if that is how your software tree is defined.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Depending on whether you use CALIBRE_HOME or MGC_HOME to point to the Calibre software tree, make sure that $CALIBRE_HOME/tmp or $MGC_HOME/tmp exists. The tmp directory is required.

   If the tmp directory does not exist, define the environment variable MGC_TMPDIR to point to a directory that can be used by the installation process.

   The installation process searches first for $CALIBRE_HOME/tmp or $MGC_HOME/tmp, then searches for $MGC_TMPDIR if neither of those directories are found. If no tmp directory is found, a warning is issued and the installation process aborts.

3. Review the README file located at

   ```
   $CALIBRE_HOME/shared/pkgs/icv/tools/queryskl/skillREADME
   ```

   This file provides complete instructions on installing the *calibre.skl* file that resides at the same directory level. (Links to *calibre.skl* and *calibre.4.3.skl* are available at $CALIBRE_HOME/lib in the software tree.)

4. Edit your *.cdsinit* file as directed in the *skillREADME* file to automatically load the Calibre integration.

   **Note**

   Also see "Requirements and Setup for Calibre RealTime with Cadence Virtuoso" in the *Calibre RealTime User's Manual* if you are using Calibre RealTime.

   The **Calibre** menu is added by the SKILL Interface as a user menu trigger. Whenever a layout window (of type maskLayout, maskLayoutXL, maskLayoutGXL, maskLayoutEAD, or maskLayoutEXL) is opened, the user menu trigger is executed. The SKILL Interface then installs any other user menus that may have been defined (either by you or by other applications), and then installs the **Calibre** menu as the last menu.

   If other applications also install menus in layout windows, you should load calibre.skl after you load these other applications' SKILL files. If you want to load the **Calibre** menu before other applications' menus, you need to be sure that the other applications' SKILL code ensures that the **Calibre** menu also gets loaded.

   - You can also load the Calibre integration from the CIW as follows:

     Standard installation:

     ```
     load( strcat( getShellEnvVar("CALIBRE_HOME")
      "/shared/pkgs/icv/tools/queryskl/calibre.skl" ))
     ```

     Cadence Virtuoso 4.3 installation:

     ```
     load( strcat( getShellEnvVar("CALIBRE_HOME")
      "/shared/pkgs/icv/tools/queryskl/calibre.4.3.skl" ))
     ```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. (Optional) Define the window type for the **Calibre** menu if you are not using one of the supported layout views of maskLayout, maskLayoutXL, maskLayoutGXL, maskLayoutEAD, or maskLayoutEXL.

   Do this by setting the SKILL variable mgcCalibreMenuViewType to include the other view type before you load *calibre.skl*. This can be done in the *.cdsinit* file as shown in the following code section, where myLayout is the unsupported layout view:

   ```
   mgcCalibreMenuViewType = list("maskLayout" "maskLayoutXL"
    "maskLayoutGXL" "maskLayoutEAD" "maskLayoutEXL" "myLayout")
   load( strcat( getShellEnvVar("CALIBRE_HOME")
       "/shared/pkgs/icv/tools/queryskl/calibre.skl" ))
   ```

   In order to preserve support of the standard view types, define mgcCalibreMenuViewType as a list of supported view types, as shown. The **Calibre** menu is only displayed for the view types defined by mgcCalibreMenuViewType.

   **Note**

   The mgc_load_calibre_menu() SKILL command is also provided. This command installs the **Calibre** menu in the active window. You can tie this command to a bind key if desired.

### Related Topics

Basic Interface

Custom Menu Items in Design Tools

Using the Calibre - Virtuoso Interface

The Calibre Menu in Cadence Virtuoso

Cadence Virtuoso [Calibre RVE User's Manual]

# The Calibre Menu in Cadence Virtuoso

The **Calibre** menu in Cadence Virtuoso layout and schematic windows (windows of type maskLayout or schematic by default) has the selections to start Calibre Interactive, start Calibre RVE, set up options, and map schematic nets.

You can also add custom menu items to the Calibre integration menu; see “Custom Menu Items in Design Tools” on page 709.

The following tables describe the menu contents:

- Calibre Menu (Table A-4) — The menu items available from a layout or schematic window.
- Calibre Setup Menu (Table A-5) — The items in the **Calibre** > **Setup** submenu in a layout or schematic window.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

In addition, there is a Calibre menu in the Cadence Virtuoso Analog Design Environment (ADE) window; see "Mapping Schematic Nets to Calibre View Nets with Multiple Extracted Views for Simulation and Plotting" on page 811.

**Table A-4. Calibre Menu in Cadence Virtuoso**

| Menu Item | Description |
|---|---|
| Run nmDRC | Starts Calibre Interactive DRC. (not available in schematic view) |
| Run DFM | Starts Calibre Interactive DFM. (not available in schematic view) |
| Run nmLVS | Starts Calibre Interactive LVS. |
| Run PERC | Starts Calibre Interactive PERC. |
| Run PEX | Starts Calibre Interactive PEX. (not available in schematic view) |
| Start RVE | Starts Calibre RVE. |
| Clear Highlights | Clears Calibre RVE highlights in the current cell. |
| Setup | See Table A-5. |
| OPC Workbench | Contains a submenu for using Calibre WORKbench, if installed. |
| RealTime DRC | See the *Calibre RealTime Custom User's Manual*. |
| Map Schematic Net | Only available in the schematic view. Opens the dialog box "Map Schematic Nets to CalibreView Nets."<br>See the sections "Mapping Schematic Nets to Calibre View Nets for Simulation and Plotting" on page 810 and "Backannotation of Parasitic Values to the Schematic in Cadence Virtuoso" on page 813. |

The **Calibre** > **Setup** menu has the following items:

**Table A-5. Calibre Setup Menu in Cadence Virtuoso**

| Menu Item | Description |
|---|---|
| Layout Export | Displays a dialog that controls options used while exporting layout databases to Calibre. See "Layout Export in Cadence Virtuoso" on page 730. |
| Netlist Export | Controls options used while exporting Composer schematics to Calibre. See "Netlist Export in Cadence Composer" on page 740. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-5. Calibre Setup Menu in Cadence Virtuoso (cont.)**

| Menu Item | Description |
|---|---|
| Calibre View | Sets Calibre View options for Calibre parasitic extraction.<br>Environment variables and SKILL variables can be used as part of a filename. See "Calibre View Setup Dialog Box" on page 777. |
| RVE | Sets up your Cadence Library for Calibre RVE and your Highlight Layers. See "Setup RVE Dialog Box in Cadence Virtuoso" in the *Calibre RVE User's Manual*. |
| Socket | Specifies the socket that Calibre Interactive and Calibre RVE use to communicate with Cadence Virtuoso. The socket is not initialized until needed.<br>If you want to initialize the socket during the loading of calibre.skl, set the environment variable MGC_RVE_INIT_SOCKET_AT_STARTUP to any value before loading calibre.skl. The socket number is displayed in the CIW window.<br>The socket port number may be specified by the MGC_CALIBRE_LAYOUT_SERVER environment variable, as described in "Setting the Calibre Interactive Socket Port for a Layout Viewer" on page 707. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Layout Export in Cadence Virtuoso

You can control the layout export from Cadence Virtuoso with the Calibre Layout Export Setup dialog box and the Cadence Virtuoso template file. You can also use the Cadence Virtuoso streamOutKeys or oasisOutKeys variables.

The Calibre Layout Export Setup Dialog Box in Cadence Virtuoso (**Calibre > Setup > Layout Export**) does not provide control over all the various options available for layout export in Cadence Virtuoso. However, the export procedure utilizes the template file if one is specified. The recommended flow is to set up all the applicable options for export through the Cadence Virtuoso File > Export > Stream dialog box, then save these options in a template file. See "Using a Template File for Layout Export with Cadence Virtuoso" on page 734 for instructions on specifying the template file for use by Calibre Interactive.

You can load the template file in the **Calibre > Setup > Layout Export** dialog box or use the SKILL function mgc_rve_load_export_layout_template_file. The complete procedure is described in the section "Using the Calibre - Virtuoso Interface" on page 745.You can specify that the Calibre Layout Export Setup dialog box is opened before each export process by enabling "Show dialog before export" in the dialog. This is useful if you need to change layout export settings for each run.

When exporting a layout to GDSII format, pipo strmout or strmout is used, depending on whether you are using CDBA-based Cadence Virtuoso (version 5.1.4 and earlier) or the Open Access version (version 6.1 and later). Several options are available to control the layout export. These options are stored within Cadence Virtuoso as the SKILL variable streamOutKeys.

When exporting to OASIS format, the oasisout command is used. OASIS export is not supported in CDBA-based Virtuoso. The options for layout export are stored in the SKILL variable oasisOutKeys.

The streamOutKeys or oasisOutKeys variable may be loaded directly in the CIW, using a trigger function or a template file.

For pipo strmout with CDBA-based Cadence Virtuoso, Calibre Interactive determines whether the streamOutKeys variable has been set within the Cadence Virtuoso editor before instructing Cadence Virtuoso to export the layout. If the streamOutKeys variable is set, Calibre Interactive creates a text file containing the values. However, if the variable is not set, Calibre Interactive initializes the variable to default values. The created text file is saved in the Run Directory and is named *CGI.streamOut.<cell_name>*.

If you are using Cadence Virtuoso 6.1 and later, you can stream out the design from memory, rather than having to save the design to disk prior to running Calibre. Choose **Calibre > Setup > Layout Export** to open the Calibre Layout Export Setup dialog box, then enable "Export From Virtual Memory." All options in the Stream Out form are filled in by Calibre.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Lower and uppercase settings are supported. You can set the streamOutKeys->caseSensitivity or oasisOutKeys->case variable to upper, lower, or preserve before exporting the layout.

Calibre Interactive alters only two options set by the existing streamOutKeys or oasisOutKeys within Cadence Virtuoso—the case sensitivity and primary cell options. The case sensitivity option is set to preserve, and the primary cell option is set to the value specified in the Top Cell field on the Inputs pane.

The layout export process displays a dialog if the cell being exported has been modified. The dialog allows you to optionally save the cell before exporting it. Note that only the top-level cell is checked for modifications.

For export to GDS format, if the layout export process renames the top cell because another cell has the same name, the following warning is issued, and calibre is not run.

```
Duplicate cell name found in design. Renaming 'PATH:path' to 'path_2'.
### Calibre will not proceed due to above warning(s)!###
```

You can disable the check for duplicate cell names by setting the environment variable MGC_CALIBRE_NO_PIPO_WARNING_CHECK to any value. By default, MGC_CALIBRE_NO_PIPO_WARNING_CHECK is not set. This environment variable does not apply for OASIS export.

See "Cadence Virtuoso" on page 725 for general information about the Calibre Interactive interface to Cadence. You may also want to see these topics:

- "Automatically Saving the Express Pcell Cache" on page 738
- "Direct Read of Cadence Virtuoso OpenAccess Database" on page 738

See following topics for specific information related to layout export:

**Calibre Layout Export Setup Dialog Box in Cadence Virtuoso** . . . . . 731
**Using a Template File for Layout Export with Cadence Virtuoso** . . . . . 734
**SKILL Functions for Layout Export** . . . . . 736

## Calibre Layout Export Setup Dialog Box in Cadence Virtuoso

The Calibre Layout Export Setup dialog box sets options that control layout export from Cadence Virtuoso to Calibre. Environment variables and SKILL variables can be used in filenames. From Cadence, choose **Calibre** > **Setup** > **Layout Export** to open the layout export dialog box.

You can also have a similar dialog box, the Calibre Layout Export dialog box, open automatically before the export process starts; this enables you to change settings before the



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

run. To do this, enable “Show dialog before export” in the Calibre Layout Export Setup dialog box. The settings in the two dialog boxes are slightly different.

The following table lists the settings in both the Calibre Layout Export Setup and the Calibre Layout Export dialog box; a note is included if a setting is only present in one of the dialog boxes. For complete details on the settings and the related streamOutKeys and oasisOutKeys variables, consult the Cadence documentation.

**Table A-6. Calibre Layout Export Setup Dialog Box Contents**

| Setting | Description |
|---|---|
| Template File<br><br>(Layout Export Setup dialog box only) | Specify a template file. Click **Load** to load the settings into the dialog box.<br><br>Specifying a template file creates a streamOutKeys or oasisOutKeys variable and sets the SKILL variable mgc_calibre_export_layout_template_file (GDS) or mgc_calibre_export_layout_oasis_template_file (OASIS).<br><br>Also see “Using a Template File for Layout Export with Cadence Virtuoso” on page 734, which includes instructions for setting the template file using a SKILL variable. |
| Library Name<br><br>(Layout Export dialog box only) | The library name, taken from the “Library Name” field on the Inputs pane of Calibre Interactive. |
| Top Cell Name<br><br>(Layout Export dialog box only) | The top cell name, taken from the “Top Cell” field on the Inputs pane of Calibre Interactive. |
| View Name | Sets the Cadence view of the cell. For the Layout Export dialog box, this is taken from the “View Name” field on the Inputs pane of Calibre Interactive. |
| Stream File/OASIS File<br>(Layout Export dialog box only) | The filename for the exported layout file, taken from the “Layout File” field on the Inputs pane of Calibre Interactive. This setting cannot be changed. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-6. Calibre Layout Export Setup Dialog Box Contents (cont.)**

| Setting | Description |
|---|---|
| Run Directory | Specify the run directory used during layout export. Relative pathnames specified in the export dialog box are qualified with the run directory pathname. The default is the current directory.<br><br>Tip: The log file location can be controlled by setting the run directory and by specifying a relative pathname for the log file. |
| Export Log File<br><br>(Layout Export dialog box only) | Saves the Cadence log file for the export to the run directory with the specified filename (default: PIPO.LOG.<cellname> or OASIS.LOG.<cellname>) |
| Export Summary File<br><br>(Layout Export dialog box only) | Saves the Cadence summary file for the export to the run directory with the specified filename (default: PIPO.SUM.<cellname> or OASIS.SUM.<cellname>) |
| Layer Map File | Specify a layer map file to be used during layout export. |
| Object Map File | Specify a object mapping file to be used during layout export. If you do not specify object mapping, Calibre uses Cadence Virtuoso defaults.<br><br>This setting is only available if you are using an OpenAccess-based version of Cadence Virtuoso (5.1 or 6.1). |
| User Skill File | Specify a SKILL file to be used during layout export. |
| Export From Virtual Memory<br><br>(only available in OpenAccess version of Cadence Virtuoso) | Specify to stream out from virtual memory using the Cadence Virtuoso XStream Out or XOasis Out form instead of the strmout or oasisout command. All options in the form are filled in by Calibre. |
| Enable Coloring<br><br>(only available in Cadence Virtuoso IC 12) | Use the pre-coloring selections defined in the Cadence Virtuoso design. Coloring is handled by the Virtuoso streamout or oasisout command.<br><br>You can set the environment variable MGC_CI_EXPORT_ENABLE_COLORING to have this option set automatically. By default, coloring information is not exported. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-6. Calibre Layout Export Setup Dialog Box Contents (cont.)**

| Setting | Description |
|---|---|
| Options for Format | Specify GDSII or OASIS for the exported database format.<br>This setting cannot be changed from the Calibre Layout Export dialog box. |
| Show dialog before export<br><br>(Layout Export Setup dialog box only) | Display the Calibre Layout Export dialog box every time before export. |

# Using a Template File for Layout Export with Cadence Virtuoso

You can use a template file to control the layout export when using Calibre Interactive with Cadence Virtuoso. The template file is specified with the global variable mgc_calibre_export_layout_template_file (GDSII) or mgc_calibre_export_layout_oasis_template_file (OASIS) or using the GUI.

The template file is typically defined in your .cdsinit file or other startup routine. If specified, Calibre Interactive uses the template file as is and does not perform any error checking on the settings. Certain settings in the template file are overwritten by Calibre Interactive at runtime based on Calibre Interactive settings, the open cell view, and the settings in the Calibre Layout Export dialog box (**Calibre** > **Setup** > **Layout Export**). Default streamout settings are used if a template file is not specified.

If a template file is specified as part of the Cadence Virtuoso startup process, the settings are loaded into the Calibre Layout Export dialog box when the Calibre integration is loaded. You can also load the template file settings at any time using the mgc_rve_load_export_layout_template_file command or the Load button in the dialog box.

## Prerequisites

- A Cadence Virtuoso template file for layout streamout.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Procedure

Define the template file with one of these methods:

| To define using … | Do this … |
|---|---|
| SKILL variable: | Define one of the following variables with the path to the template file:<br>• GDS: mgc_calibre_export_layout_template_file<br>• OASIS: mgc_calibre_export_layout_oasis_template_file<br><br>(Optional) If you defined the template file manually rather than as part of a startup procedure, call mgc_rve_load_export_layout_template_file("GDSII") or mgc_rve_load_export_layout_template_file("OASIS") to load the settings into the Calibre Layout Export Setup dialog box. |
| GUI setting: | In Cadence Virtuoso, do the following:<br>1. Select **Calibre** > **Setup** > **Layout Export**.<br>2. In “Options for Format”, select GDSII or OASIS for the exported layout format.<br>3. Fill in “Template File” with the file path.<br>4. Click **Load** to read the settings into the Calibre Layout Export Setup dialog box.<br>5. Make other settings as desired.<br>6. Click **OK**. |

## Results

Calibre Interactive uses the template file to control layout export. Certain settings in the template file are overwritten by Calibre Interactive at runtime based on Calibre Interactive settings, the open cell view, and the settings in the Calibre Layout Export dialog box (**Calibre** > **Setup** > **Layout Export**).

If you used the GUI to specify the template file, this defines mgc_calibre_export_layout_template_file or mgc_calibre_export_layout_oasis_template_file, depending on whether GDS or OASIS export is selected.

## Examples

The following example specifies a layout template file for GDS export:

```
mgc_calibre_export_layout_template_file = "my_layout_template"
```

When placed in a .cdsinit file or other startup routine, the settings are automatically loaded into the Calibre Layout Export dialog box when the Calibre integration is loaded.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# SKILL Functions for Layout Export

Layout export functions are provided for Cadence Virtuoso.

The following functions are provided:

- mgc_rve_export_layout_cmd
- mgc_rve_load_export_layout_template_file

The commands mgc_rve_load_export_layout_template_file(“GDSII”) and mgc_rve_load_export_layout_template_file(“OASIS”) are executed upon loading the Calibre interface.

## mgc_rve_export_layout_cmd

Returns a string value consisting of the layout export command used by Calibre Interactive.

This command is provided for historical reasons for those wanting to customize the layout export process.

### Syntax

mgc_rve_export_layout_cmd(libName cellName viewName fileName @optional (format "GDSII") )

### Arguments

- libName — string; the library name.
- cellName — string; the top cell name.
- viewName —string; the cellview name.
- fileName — string; the layout database filename, taken from the File entry field on the Inputs pane of Calibre Interactive.
- format — An optional argument specifying the format of the exported layout; allowed values are the “GDSII” and “OASIS”. The default is “GDSII”.

### Return value

Returns a string value consisting of the layout export command.

### Related Topics

mgc_rve_load_export_layout_template_file

Layout Export in Cadence Virtuoso

Calibre Layout Export Setup Dialog Box in Cadence Virtuoso



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# mgc_rve_load_export_layout_template_file

SKILL command to load the layout template file specified by the SKILL variable mgc_calibre_export_layout_template_file or mgc_calibre_export_layout_oasis_template_file.

## Syntax

mgc_rve_load_export_layout_template_file(@optional (format “GDSII”))

### Arguments

- format — An optional argument specifying the format of the exported layout; allowed values are the following:

  GDSII — (default) Loads the file specified by the mgc_calibre_export_layout_template_file variable.

  OASIS — Loads the file specified by the mgc_calibre_export_layout_oasis_template_file variable.

### Return value

Returns “t” on success or an error message on failure.

### Examples

The following example loads a layout template file for GDS export:

```
mgc_calibre_export_layout_template_file = "my_layout_template"
mgc_rve_load_export_layout_template_file()
```

The following example loads a layout template file for OASIS export:

```
mgc_calibre_export_layout_oasis_template_file = "my_oasis_template"
mgc_rve_load_export_layout_template_file("OASIS")
```

**Note**

If a template file is specified as part of the Cadence Virtuoso startup process, the settings are loaded into the Calibre Layout Export dialog box when the Calibre integration is loaded. See “Using a Template File for Layout Export with Cadence Virtuoso” on page 734.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Direct Read of Cadence Virtuoso OpenAccess Database

Calibre can read the design data directly from the OpenAccess database. This is called Direct Read, in contrast to Layout Export, in which Cadence Virtuoso converts the design to GDS format and streams it out to Calibre Interactive.

See "Providing OpenAccess Input" on page 90 for instructions on specifying Direct Read of an OpenAccess database.

You can enable Calibre Interactive to save the Express Pcell cache, as described in "Automatically Saving the Express Pcell Cache" on page 738.

The following additional topics may be of interest:
"Layout Export in Cadence Virtuoso" on page 730
"Creating an Interface to Cadence Virtuoso" on page 725

**Automatically Saving the Express Pcell Cache . . . . . 738**

## Automatically Saving the Express Pcell Cache

Calibre Interactive automatically saves the Express Pcell cache when performing direct read of Cadence Virtuoso OpenAccess databases if the following conditions are met.

### Prerequisites

- Using the OpenAccess-based version of Cadence Virtuoso, and the environment variables CDS_INST_ROOT or CDS_INST_DIR, and PATH are set correctly. CDS_INST_ROOT takes precedence if both CDS_INST_ROOT and CDS_INST_DIR are defined.
- Make sure LD_LIBRARY_PATH includes the libstdc++.so.6.0.8 compiler libraries. This is true by default for Linux machines with RHEL 4u6 or later.

  Here is an example if you need to set LD_LIBRARY_PATH:

  ```
  setenv LD_LIBRARY_PATH
     ${CDS_INST_DIR}/share/oa/lib/linux_rhel40_gcc44x_64/opt:
     ${CDS_INST_DIR}/tools/cdsSkillPcell/lib/64bit:
     ${CDS_INST_DIR}/tools/lib/64bit:${LD_LIBRARY_PATH}
  ```

### Procedure

1. Make sure the pcell definition is loaded in the *.cdsinit* file. For example, your local *.cdsinit* file might look like this:

   ```
   ; local .cdsinit
   load("~/.cdsinit")
   load("mypcells.il")
   ; ... rest of code
   ```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Open Calibre Interactive.
3. Click the **Inputs** button on the left panel of Calibre Interactive, and set “Format” to “OPENACCESS.”
4. Choose **Setup** > **Options** from the menu bar, then click the **Database** > **OpenAccess/LEFDEF** tab and do the following:
   a. Choose the **Other Options** category on the **OpenAccess/LEFDEF** tab.
   b. Enable “Set OA_HOME when CDS_INST_ROOT or CDS_INST_DIR is set.”
   c. Enable “Set CALIBRE_READDB_LD_LIBRARY_PATH when CDS_INST_ROOT or CDS_INST_DIR is set.”
   d. Enable “Set PATH when CDS_INST_ROOT or CDS_INST_DIR is set.”
   e. If you are using Express Pcells, enable “Set CDS_EXP_PCELL_DIR when CDS_INST_ROOT or CDS_INST_DIR is set.”
   f. If you are using Express Pcells, enable “Set CDS_EXP_ALL_PARAMS when CDS_INST_ROOT or CDS_INST_DIR is set.”
   g. If you are using Express Pcells, enable “Set CDS_ENABLE_EXP_PCELL when CDS_INST_ROOT or CDS_INST_DIR is set.”

## Results

The Pcell cache is automatically saved in the directory specified by CDS_EXP_PCELL_DIR.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Netlist Export in Cadence Composer

Many aspects of netlist export from Cadence Composer are controlled with the Calibre Netlist Export Setup dialog box. The netlist export routines use the Simulation Interface system (si -batch) to translate the schematic to CDL when performing netlist export.

The Calibre Interactive GUI for LVS, PEX, and PERC supports automatic import of Composer netlists as the source database. You can specify CDL netlisting template parameters in the Calibre Netlist Export Setup Dialog Box in Cadence Composer. You can load the template file using the dialog box or the SKILL function mgc_rve_load_export_netlist_template_file.

The source database format is set to CDL when Calibre Interactive is invoked from Cadence Virtuoso. Consequently, the netlist export routines use the Simulation Interface system (si -batch) to translate the schematic to CDL. The si command can do one of three things: set up the environment, create a netlist, or run a simulation. In the case of Calibre Interactive, you are creating a netlist. The command is:

```
si -batch -command netlist
```

When using CDL views for netlisting, si reads the si.env file created by Calibre Interactive at runtime, generates a netlist, and performs as expected.

When using auCdl views, si once again reads the si.env file, but also checks the *.simrc* file to see what variables are set. If the variables cdlSimViewList or cdlSimStopList are set, they overwrite what you specified in the si.env file. If they are the same, there is no problem; however, if they are different, netlisting fails.

The settings in the Calibre Netlist Export Setup dialog box are similar to CDL export settings and you can set them through the cdlOutKeys global SKILL variable. This variable is found in the si.env file. You can load this variable in the CIW, in a trigger function, or by specifying the mgc_calibre_export_netlist_template_file SKILL variable to point to a template file.

The netlist export process displays a dialog if the cell being exported has been modified. The dialog allows you to optionally save the cell before exporting it. Note that only the top-level cell is checked for modifications.

During netlist export, the *si.log* file is created in the same directory as the netlist file. You can use the SKILL variable mgc_rve_globals->siLogFile to specify a different name for the file.

See "Cadence Virtuoso" on page 725 for general information about the Calibre Interactive interface to Cadence.

See following topics for specific information related to netlist export:

**Calibre Netlist Export Setup Dialog Box in Cadence Composer** . . . . . **741**
**SKILL Functions for Netlist Export** . . . . . **744**



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Netlist Export Setup Dialog Box in Cadence Composer

From Cadence, choose **Calibre > Setup > Netlist Export** to open the netlist export dialog box.

The dialog box sets options that control the export of Composer schematics to Calibre. Environment variables and SKILL variables can be used in filenames.

The Calibre Netlist Export Setup dialog has the settings shown in the following table. You may need to scroll down to see all of the entries. For complete details on the settings and the related cdlOutKeys variable, consult the Cadence documentation

**Table A-7. Calibre Netlist Export Setup Dialog Box Contents**

| Setting | Description |
|---|---|
| Template File | Specify a template file to load or save. Saving creates a cdlOutKeys variable.<br>You can also load the template file by specifying it with the SKILL variable mgc_calibre_export_netlist_template_file, then calling the SKILL function mgc_rve_load_export_netlist_template_file. |
| View name | Sets the Cadence view of the cell. |
| Simulator | Sets which tool does the netlisting (cdl or audCdl are typical choices). |
| View List | The view list (also call “view switch list”) controls hierarchy traversal during netlist export.<br>You can set the environment or SKILL variable CDS_Netlisting_Mode to pre-populate the View List and Stop List fields. |
| Stop List | The stop list (also call “stopping point view list”) controls when expansion stops during hierarchy traversal for netlist export.<br>You can set the environment or SKILL variable CDS_Netlisting_Mode to pre-populate the View List and Stop List fields. |
| Resistor Threshold Value | Sets the value for the *RESI command. |
| Resistor Model Name | Produces the *.RESI command in the netlist. |
| Equivalents | Produces *.EQUIV statement in netlist |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-7. Calibre Netlist Export Setup Dialog Box Contents (cont.)**

| Setting | Description |
|---|---|
| Connects | Produces *.CONNECT statement in netlist. Specify the nets with an equals sign (=) between them.<br>For example, specifying the following in the dialog entry field:<br>VDD=VDD! VCC=VCC!=VCC2<br>produces the following statements in the netlist:<br>*.CONNECT VDD VDD!<br>*.CONNECT VCC VCC! VCC2 |
| Include File | Produces *.INCLUDE statement in netlist |
| Check Resistors | When set, include resistor information in the output netlist. |
| Check Capacitors | When set, include capacitor information in the output netlist. |
| Check Diodes | When set, include diode information in the output netlist. |
| Scale | Select the scale for the output netlist; the default is meter. This produces a *SCALE METER or *SCALE MICRON command in the netlist. |
| Shrink Factor for width(w) and length(l) | Indicates a scaling percentage. |
| Check LDD | Produces *.LDD statement in netlist. |
| Display Pin Information | Produces *.PININFO statement in netlist |
| Map Bus Name From <> To [] | Control mapping of bus pins in netlist export.<br>Enabled — []<br>Disabled — <> (default)<br>Also see "Automatically Convert Bus Name Delimiter for Calibre Interactive and Calibre RVE" on page 808. |
| Global Power Signals | A list of global power signals, which are indicated with a :P in the netlist. |
| Global Ground Signals | A list of global ground signals, which are indicated with a :G in the netlist. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-7. Calibre Netlist Export Setup Dialog Box Contents (cont.)**

| Setting | Description |
|---|---|
| Analog Netlisting Type<br>(The “Simulator” option must be set to audCdl) | Selects how netlisting is done when the simulator is audCdl.<br>• Connection By Order<br>auCdlDefNetlistProc = "ansCdlSubcktCall"<br>(also known as implicit netlisting)<br>• Connection By Name<br>auCdlDefNetlistProc = "ansCdlHnlPrintInst"<br>(also known as explicit netlisting) |
| Show dialog before export | As described. |

.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# SKILL Functions for Netlist Export

Netlist export functions are provided for Cadence Virtuoso.

The following functions are provided:

- mgc_rve_export_netlist_cmd
- mgc_rve_load_export_netlist_template_file

The command mgc_rve_load_export_netlist_template_file() is executed upon loading the Calibre interface.

## mgc_rve_export_netlist_cmd

Returns a string value consisting of the netlist export command used by Calibre Interactive.

This command is provided for historical reasons for those wanting to customize the netlist export process.

### Syntax

mgc_rve_export_netlist_cmd(libName cellName viewName fileName)

### Arguments

- libName — string; the library name.
- cellName — string; the top cell name.
- viewName —string; the cellview name.
- fileName — string; the netlist filename, taken from the File entry field on the Inputs pane of Calibre Interactive.

### Return value

Returns a string value consisting of the netlist export command.

### Related Topics

mgc_rve_load_export_netlist_template_file

Netlist Export in Cadence Composer

Calibre Netlist Export Setup Dialog Box in Cadence Composer

## mgc_rve_load_export_netlist_template_file

Loads the netlist template file specified by the SKILL variable mgc_calibre_export_netlist_template_file.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Syntax

mgc_rve_load_export_netlist_template_file()

#### Example

```
mgc_calibre_export_netlist_template_file = "my_netlist_template"
mgc_rve_load_export_netlist_template_file
```

### Related Topics

Netlist Export in Cadence Composer

Calibre Netlist Export Setup Dialog Box in Cadence Composer

## Using the Calibre - Virtuoso Interface

The Calibre Interactive interface allows you to run Calibre on a Cadence Virtuoso cell. You can specify settings that control the layout export process.

The procedure below is written for the Calibre Interactive nmDRC application, but similar steps are used for other Calibre Interactive applications.

### Prerequisites

- The Calibre - Virtuoso interface is installed, as described in "Creating an Interface to Cadence Virtuoso" on page 725.
- (Recommended) A Cadence Virtuoso layout export template file is defined in the .cdsinit or other startup procedure. Use the SKILL variable mgc_calibre_export_layout_template_file (GDSII) or mgc_calibre_export_layout_oasis_template_file (OASIS), as explained in "Using a Template File for Layout Export with Cadence Virtuoso" on page 734.
- (Optional) If you are exporting a schematic, set up all the applicable options for netlist export using the Cadence Virtuoso File > Export > CDL dialog box, then save these options in a template file.
- Cadence Virtuoso is open to the desired cell, and the connection is made to Calibre Interactive. See Basic Interface - "Calibre Interactive" on page 706, "Setting the Socket Port in Calibre Interactive" on page 707, and "Setting Socket Connections with Cadence Virtuoso" on page 747.
- Calibre Interactive is open, and GUI settings are correct.

### Procedure

1. In Calibre Interactive, click **Inputs** on the left panel and verify that the "Export from layout viewer" option is selected; this directs Calibre Interactive to communicate with the SKILL Interface before the Calibre run.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If you are exporting a schematic, also verify that "Export from schematic viewer" is selected on the Inputs pane.

2. (Recommended) If you are using the Open Access version of Cadence Virtuoso and CDS_INST_ROOT or CDS_INST_DIR is set in your environment, you may want to set the following options in Calibre Interactive.

   a. Choose **Setup** > ***app*** **Options**.

   b. Click the **OpenAccess/LEFDEF** tab.

   c. Choose the **Other Options** category, and enable the following:

      - Set OA_HOME when CDS_INST_ROOT or CDS_INST_DIR is set
      - Set CALIBRE_READDB_LD_LIBRARY_PATH when CDS_INST_ROOT or CDS_INST_DIR is set
      - Set PATH when CDS_INST_ROOT or CDS_INST_DIR is set
      - Set CDS_EXP_PCELL_DIR when CDS_INST_ROOT or CDS_INST_DIR is set
      - Set CDS_EXP_ALL_PARAMS when CDS_INST_ROOT or CDS_INST_DIR is set
      - Set CDS_ENABLE_EXP_PCELL when CDS_INST_ROOT or CDS_INST_DIR is set

   The settings automatically set the indicated environment variable based on the value of CDS_INST_ROOT or CDS_INST_DIR. CDS_INST_ROOT takes precedence if both CDS_INST_ROOT and CDS_INST_DIR are defined. See the "Other Options" table in "OpenAccess/LEFDEF Options Tab" on page 82 for more information.

3. In Cadence Virtuoso, choose **Calibre > Setup > Layout Export** to open the Calibre Layout Export Setup dialog box and verify the settings are correct. See "Calibre Layout Export Setup Dialog Box in Cadence Virtuoso" on page 731.

   **Tip**

   You can enable the "Show dialog before export" option on the Calibre Layout Export Setup or Calibre Netlist Export Setup dialog box if you want to make changes to the options each time there is an export request from Calibre Interactive.

4. (Optional) If you are exporting a schematic, choose **Calibre > Setup > Netlist Export** to open the Calibre Netlist Export Setup dialog box.

   a. (Optional) Load a CDL template file.

   b. Adjust settings as needed, as described in "Calibre Netlist Export Setup Dialog Box in Cadence Composer" on page 741.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. In Calibre Interactive, click **Run DRC** on the left panel, or the appropriate button for your Calibre application.

   The SKILL Interface exports the layout cell in stream format using strmout, oasisout, or pipo strmout, as appropriate.

### Related Topics

Layout Export in Cadence Virtuoso

Netlist Export in Cadence Composer

Creating a Calibre View

Automating Calibre Interactive Runsets in the Cadence Design Environment

External Trigger Definitions in Cadence Virtuoso

# Setting Socket Connections with Cadence Virtuoso

The Calibre SKILL Interface uses a TCP socket to communicate with Calibre RVE and Calibre Interactive. When you load the calibre.skl file, the tool attempts to initialize a server socket at port 9189. If that port is being used, the tool searches for a free socket port in the range 5000 to 9999.

If the default port number is used, the following message is displayed in the CIW transcript:

```
//  Calibre layout-server initialized successfully at socket 9189.
```

If a different port is found after a search for free ports, the following messages are displayed in the CIW transcript:

```
//  Could not initialize Calibre layout-server socket at port 9189. Trying
to find free socket . . .
//  Calibre layout-server initialized successfully at socket 5000.
```

To find out what your socket is, enter the following in the CIW window:

```
mgc_rve_globals->socket_number
```

If necessary, you can control the port number used by setting the environment variable MGC_CALIBRE_LAYOUT_SERVER as described in "Setting the Calibre Interactive Socket Port for a Layout Viewer" on page 707.

You can also use following command from within Cadence Virtuoso to specify a different port without resetting the MGC_CALIBRE_LAYOUT_SERVER environment variable and restarting Cadence Virtuoso:

**mgc_rve_init_socket <port_number>**



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Type the command in Cadence Virtuoso to initialize the specified socket port for communication.

# Calling SKILL Functions from Calibre Interactive

It is often useful to call a SKILL procedure before or after your Calibre run.

The following topics discuss methods for doing this:

- "External Trigger Definitions in Cadence Virtuoso" on page 676
- "Setting Calibre Interactive Internal Triggers" on page 661, using a trigger type of "layout" or "schematic"
- "mgc_calibre_add_menu_item" on page 714, the -code option when adding a custom menu to a design tool



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Creating a Calibre View

A Calibre View of a cell is a schematic view of a cell you can simulate in the Cadence Analog Design Environment. This schematic view contains the connectivity and intentional devices the Calibre LVS tool extracts, and, optionally, parasitic devices the Calibre xRC tool extracts.

You can create the Calibre View using Calibre Interactive PEX or the Calibre Interactive nmLVS and Calibre RVE tools. You can access Calibre Interactive from the **Calibre** pulldown menu in Cadence Virtuoso. An OpenAccess database is required. When displaying the Calibre View, Cadence Virtuoso uses a cellmap file to map the layout device to the corresponding schematic symbol.

You can also create the Calibre View in batch mode, without Calibre Interactive. Use the SKILL functions mgc_rve_load_setup_file() or mgc_rve_create_cellview() to create the Calibre View. See "SKILL Functions for Creating a Calibre View" on page 798.

There are global variables and environment variables that affect Calibre View generation; see "Global Variables for Calibre View Generation" on page 800 and "Environment Variables for Calibre View" on page 843.

These topics are covered:

**What You Need Before Creating a Calibre View** . . . . . 750
**What a Calibre View Produces** . . . . . 751
**Automatic OpenAccess Version Detection in Calibre View** . . . . . 751
**Device Property Calculation in Calibre View Generation** . . . . . 752
**Handling of Nets With Inherited Connectivity** . . . . . 753
**Bus Delimiter in Calibre View** . . . . . 754
**Identifying Devices with a Cellmap File** . . . . . 754
**Creating the Cellmap File** . . . . . 754
**Syntax for Cellmap Statements** . . . . . 756
**Cellmap Examples** . . . . . 766
**Setting the Calibre View Cellmap File in Calibre Interactive** . . . . . 773
**Importing Schematic Properties for Calibre View** . . . . . 774
**Auto-Mapping Behavior, Control, and Debug in Calibre View Generation** . . . . . 775
**Calibre View Setup Dialog Box** . . . . . 777
**Creating a Post-LVS Calibre View** . . . . . 780
**Creating a Post-Parasitic Extraction Calibre View** . . . . . 782
**Creating a Gate-Level Calibre View** . . . . . 786
**Creating Calibre Views in a Multiple Corner Extraction Run** . . . . . 789
**Creating a SPECTRE Netlist from the Calibre View Netlist** . . . . . 790



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Creating a CalibreView Setup File** . . . . . 791
**Showing Parasitic Polygons in Calibre View** . . . . . 792
**CalibreView Setup File Format** . . . . . 794
**SKILL Functions for Creating a Calibre View** . . . . . 798
**Global Variables for Calibre View Generation** . . . . . 800
**Creating a Calibre View in Batch Mode** . . . . . 801
**Calibre View and IC Manage Revision Control** . . . . . 802
**Calibre View Generation Warnings** . . . . . 802

## What You Need Before Creating a Calibre View

Producing a Calibre View requires the correct product licenses, input files, and integration setup.

- Calibre SKILL Interface — For setup instructions, refer to the README file at *./shared/pkgs/icv/tools/queryskl/skillREADME* in your Siemens software tree.
- Rule File — A valid Standard Verification Rule Format (SVRF) rule file for the design. For parasitic extraction, the SVRF rule file must contain, at a minimum, valid calibrated SVRF Capacitance statements and, if required, SVRF Resistance statements.
- Cadence Virtuoso IC6 or IC12 — If you are using Cadence Virtuoso IC5, please contact your Siemens representative.
- Layout Database — An OpenAccess database.
- Layout Netlist — A CDL or Calibre Interactive tool-produced SPICE source netlist if you require source names in the output for debugging.
- Licenses — When creating a Calibre View, you must have the following license features:
  - A Calibre RVE license.
  - A Calibre Interactive license if the Calibre Interactive GUI is opened.
  - A Calibre nmLVS-H license if using source names or creating a post-LVS Calibre View.
  - A Calibre Connectivity Interface license if creating a post-LVS Calibre View.

  Refer to *Calibre Administrator's Guide* for complete licensing information.
- Cellmap File — A user-created ASCII file mapping the layout device the Calibre tool extracts to the corresponding schematic symbol. Multiple cellmap files may be specified. See “Identifying Devices with a Cellmap File” on page 754.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Library Definitions — If a *cds.lib* file exists in the working directory, it is used as the default library definition file. The *cds.lib* is converted to a temporary *lib.defs* file and saved in $MGC_TMPDIR. If both *cds.lib* and *lib.defs* exist in the working directory, *lib.defs* is ignored. The *lib.defs* file is used if there is not a *cds.lib* file in the working directory.

## What a Calibre View Produces

You can produce the Calibre Views after an LVS run or after a parasitic extraction run.

The different run methods have the following results:

- Post-LVS View — generates intentional devices for analysis. See "Creating a Post-LVS Calibre View" on page 780.
- Post-Parasitic Extraction View — generates intentional devices and parasitics for analysis. See "Creating a Post-Parasitic Extraction Calibre View" on page 782 and "Creating a Gate-Level Calibre View" on page 786.

This file is produced after the Calibre View is generated:

- *calibreview.setup* — A file in the CalibreView Setup File Format. This file can be loaded into the Calibre View Setup dialog box to populate the dialog box with settings for the run. Browse for the file or enter the filename into the "Calibreview Setup File" field and click the **Load** button. The file can also be used as input to the SKILL function mgc_rve_load_setup_file to create a Calibre View in batch mode.

The setup parameters used to create a Calibre View are saved with the cell view. You can view the Calibre View setup parameters using the following command:

```
mgc_eview_display_setup_settings("lib_name" "cell_name" "view_name")
```

## Automatic OpenAccess Version Detection in Calibre View

When Calibre View uses OpenAccess, it must be a compatible version to the version of Cadence Virtuoso being used. Calibre View automatically detects which OpenAccess version to use and warns the user if the version they specified does not match.

The OpenAccess version is automatically detected based on the version of Virtuoso being used and from the $CDS_ROOT and $CDS_HOME directory content. The version detected by the Virtuoso version is determined as follows:

| Virtuoso Version | OpenAccess Version |
|---|---|
| IC6.1.2*, IC6.12*, IC6.13*, IC6.14* | 22.04 |
| IC6.15* | 22.41 |
| IC6.16*, ICAD/12.1* | 22.43 |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| Virtuoso Version | OpenAccess Version |
|---|---|
| IC6.17*, ICAD/12.20.*, ICAD/12.30* | 22.50 |
| IC6.18*, ICADVM181/18.10.* | 22.60 |

If the version detected based on the Virtuoso version does not match the version detected based on the $CDS_ROOT and $CDS_HOME directories, then the following warning is issued to tell you which version is being used:

```
WARNING: The OA version "22.60" detected from the Virtuoso version is
different from the OA version "22.50" retrieved from the "CDS_ROOT"/
"CDS_HOME" environment variable. Using "22.50" or set the required version
using the "MGC_FDI_OA_VERSION" environment variable.
```

In the above case, the version specified by the $CDS_ROOT and $CDS_HOME directories was used.

You can also specify the OpenAccess version with the MGC_FDI_OA_VERSION environment variable. When this specified version does not match the automatically detected version, the following warning is issued to tell you which version is being used:

```
WARNING: The automatic detected OA version "22.43" is different from the
user-specified OA version "22.50". Using "22.50".
```

In the above case, the version specified with the MGC_FDI_OA_VERSION environment variable was used.

When the detected OpenAccess version is 22.04 or 22.41, the following warning is printed and version 22.43 is used:

```
WARNING: For "22.04" and "22.41" OA versions "22.43" OA version is used.
```

## Device Property Calculation in Calibre View Generation

Several internal steps are involved when device properties are calculated during Calibre View generation.

1. Layout properties are calculated and included in the Calibre View format netlist after running Calibre. The device property calculations are based on statements in the Calibre LVS rule file.

   This step depends on keywords in the PEX Netlist statement, such as SOURCENAMES, LAYOUTNAMES, SOURCEBASED, and SCHEMATICONLY.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Process properties that only exist in the schematic as follows:

   For each device instance, do one of the following:

   - If importSchemProp is specified for the device in the cellmap file, import the schematic property if importSchemProp is true and skip the property if importSchemProp is false.
   - If importSchemProp is *not* specified for the device in the cellmap file, import the schematic property if the global variable mgc_eview_globals->importSchematicProperties is set to t (the default), otherwise skip the property.

   **Note**

   Properties in the layout netlist always have precedence over schematic properties. Schematic properties are imported according to the logic stated in this step and only if the layout property does not exist.

3. Execute callbacks if mgc_eview_globals->executeCallbacks is set to t (for true). Callbacks are typically used for property calculations done with equations and may overwrite the device properties calculated in previous steps.

### Related Topics

Importing Schematic Properties for Calibre View

Global Variables for Calibre View Generation

## Handling of Nets With Inherited Connectivity

The Calibre View generated view does not show nets with inherited connectivity that do not exist in schematic.

Inherited connectivity is handled in this manner:

1. If there is a net expression in a netset property, it is only copied to the corresponding netset property in the generated view device. This behavior always occurs.
2. If the net expression in the netset property needs to be created in the view to form an inherited connection, enable the env var CALIBRE_FDIBA_ENABLE_NETEXPR_COPY. The net expression is not created by default.

   **Note**

   The environment variable CALIBRE_FDIBA_ENABLE_NETEXPR_COPY returns the behavior to that found in 2019.2 and earlier releases.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Bus Delimiter in Calibre View

The generated CalibreView netlist always uses <> as the bus delimiter to avoid conflicts with the standard Cadence schematic netlist, which uses <> as the bus delimiter. If you generate a CalibreView from a netlist which uses [] as the bus delimiter, pin and net names ending in [] are automatically converted to use <> as the bus delimiter.

# Identifying Devices with a Cellmap File

Before creating a Calibre View, you must create a cellmap file. A cellmap file is an ASCII mapping file used for mapping the layout device to the corresponding schematic symbol. Both intentional devices and parasitic devices can be mapped with the cellmap file.

Multiple cellmap files may specified as a space-separated list in the Calibre View Setup dialog box or the Calibre View setup file. Multiple files are concatenated in the order specified.

## Related Topics

Creating the Cellmap File

Calibre View Setup Dialog Box

Creating a CalibreView Setup File

# Creating the Cellmap File

The cellmap file is a text file that is used when creating a Calibre View.

## Procedure

1. In an ASCII text editor, create a unique entry for each intentional device.

   See “Cellmap Syntax for Intentional Devices” on page 756.

   Note: you may use an include file in the cellmap file.

2. Add entries for parasitic devices, if needed.

   See “Cellmap Syntax for Parasitic Devices” on page 763.

3. Add layer mapping entries, if needed. This step is generally recommended.

   See these topics for additional information:

   - “Layer Mapping Syntax” on page 762
   - “Layer Map Statements to Optimize the Cellmap File” on page 770
   - “Identifying Cadence Virtuoso Layers for Mapping” on page 771

4. Save the cellmap file in the directory containing the design.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Tip**

If you are importing schematic properties, see Importing Schematic Properties for Calibre View for needed modifications to the SVRF rule file.

## Related Topics

Syntax for Cellmap Statements

Full Syntax for Property Specification in the Cellmap File

Cellmap Examples



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Syntax for Cellmap Statements

The cellmap file is needed for creating a Calibre View.

See "Creating a Calibre View" on page 749 information on Calibre View. For examples of specific cellmap statements and comments on how to solve common mapping issues, see "Cellmap Examples" on page 766.

The syntax conventions used in the cellmap descriptions are given in Table 1-3 on page 29. You should enter literal text, which is not in italics, exactly as shown. Parentheses are literal in the syntax definitions. The semicolon (;) is the comment character for cellmap files. Text after the semicolon is not parsed.

This section includes syntax definitions for the following cellmap statements:

**Cellmap Syntax for Intentional Devices** . . . . . 756
**Cellmap Syntax for Cells**. . . . . 758
**Full Syntax for Property Specification in the Cellmap File**. . . . . 760
**Layer Mapping Syntax** . . . . . 762
**Cellmap Syntax for Parasitic Devices** . . . . . 763

## Cellmap Syntax for Intentional Devices

The cellmap statement for intentional devices instructs the Calibre View process on how to map the device.

### Syntax

```
(device_model_name_in_layout
     (CDN_library_name CDN_cell_name CDN_view_name)
     (
          ({ calibre_device_pin_name | nil } { CDN_cell_pin_name | nil } )
         ...
    )
    [(
         ( [nil] calibre_property_name CDN_property_name)
         ...
           [ (importSchemProp { t | nil })]
     )]
)
```

### Parameters

- ***device_model_name_in_layout***

  The Calibre device to be mapped. Black-boxed devices are handled automatically; it is not necessary to include them in the cellmap file.

- ***CDN_library_name CDN_cell_name CDN_view_name***



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Map device_model_name_in_layout to an instance of this cellview.

- **(** { ***calibre_device_pin_name*** | **nil** } { ***CDN_cell_pin_name*** | **nil** }**)**

  Maps device pins to cellview pins. At least one pin mapping instruction must be included.

  - **nil** — The parameter nil is used to indicate that the pin does not exist. See "Mapping with Fewer Device Pins" on page 767 for an example of mapping when the device has fewer pins than the cellview.

    When a device pin is mapped to a cellview terminal that doesn't exist, a property will be created on the cellview instance with its value set to CDN_cell_pin_name. See "Mapping Extra Device Pins as Nets Mapped to Cell Properties" on page 768 for an example of mapping a Calibre device to a cellview with fewer pins.

  Bus pins may be mapped as follows:

  **(** ***calibre_bus_name<start_pin:end_pin> CDN_cell_bus_name<start_pin:end_pin>*****)**

  The order and number of bus pins must be the same.

  Bus pins may also be mapped individually with the notation ***bus_name<pin>***.

- ([nil] *calibre_property_name CDN_property_name* ) …

  Maps a Calibre property to a Cadence cell view property. You can map a Calibre property to multiple Cadence cell view properties. This is done by specifying each property mapping separately.

  The property mapping section is optional. See the section "Full Syntax for Property Specification in the Cellmap File" on page 760 for the complete syntax of the property mapping section, including the use of expressions.

- (importSchemProp { t | nil })

  An optional parameter and value that specifies whether or not schematic properties are imported for this device

  t — import schematic properties

  nil, or any value other than t — do not import schematic properties

  If importSchemProp is specified for a device, it overrides the global flag mgc_eview_globals->importSchematicProperties. If importSchemProp is not specified for a device, the global flag is used. The default value of mgc_eview_globals->importSchematicProperties is t.

  See the section "Importing Schematic Properties for Calibre View" on page 774 for important information on case sensitivity related to importing schematic properties.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Usage Notes

You can have multiple devices with the same modelname, however each device should have different pin counts. No errors or warnings occur if you map a Calibre device to different Cadence cellviews with the same number of pins, but this is not recommended. See the example "Specifying Devices with the Identical Modelname" on page 766.

Black-boxed devices are handled automatically; it is not necessary to include them in the cellmap file. See "Auto-Mapping Behavior, Control, and Debug in Calibre View Generation" on page 775.

### Examples

This cellmap example specifies the mapping for device *n* with properties specified:

```
(n
     (gpdk nmos symbol)
     (
         (d D)
         (g G)
         (s S)
         (b B)
      )
     (
      (w W)
      (l  L)
      )
)
```

This example maps a bus.

```
(ElementB
    (libB ElementB symbol)
    (
        (VDD        VDD       )
        (test3<7:0> test3<7:0>)
        (BUS2<7:0>  BUS2<7:0> )
    )
)
```

### Related Topics

Full Syntax for Property Specification in the Cellmap File

Cellmap Syntax for Parasitic Devices

Cellmap Examples

## Cellmap Syntax for Cells

The cellmap statement instructs the Calibre View process on how to map the cell.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Syntax

```
(cell_name
     (CDN_library_name CDN_cell_name CDN_view_name)
     (
          ( { calibre_pin_name | nil } { CDN_cell_pin_name | nil } )
          …
     )
)
```

## Description

The cellmap entry for cells is used for a gate-level CalibreView; see "Creating a Gate-Level Calibre View" on page 786.

## Parameters

- ***cell_name***

  The cell name to be included in the gate-level CalibreView. Black-boxed devices are handled automatically; it is not necessary to include them in the cellmap file.

- ***CDN_library_name CDN_cell_name CDN_view_name***

  Map ***cell_name*** to an instance of this cellview.

- **( { *calibre_pin_name* | nil } { *CDN_cell_pin_name* | nil } )**

  Maps device pins to cellview pins. At least one pin mapping instruction must be included.

  - nil — The parameter nil is used to indicate that the pin does not exist. See Mapping with Fewer Device Pins for an example of mapping when the device has fewer pins than the cellview.

    When a pin is mapped to a cellview terminal that doesn't exist, a property will be created on the cellview instance with its value set to ***CDN_cell_pin_name***. See Mapping Extra Device Pins as Nets Mapped to Cell Properties for an example of mapping a Calibre device to a cellview with fewer pins.

## Examples

Suppose you have a cell named One_Bit_Adder with this source HSPICE definition:

```
.subckt One_Bit_Adder a b Sum Carry
   <circuit_details>
.ends
```

Suppose you have a symbol named "Adder1Bit" with pins A, B, SUM, and CARRY in the Cadence schematic library named "MyLib".



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

The cellmap entry for the cell One_Bit_Adder is written as follows:

```
(One_Bit_Adder
   (MyLib Adder1Bit symbol)
   (
      (a A)
      (b B)
      (Sum SUM)
      (Carry CARRY)
   )
)
```

## Full Syntax for Property Specification in the Cellmap File

You can add property specifications to the cellmap file that determine how property information is passed from Calibre to the Calibre View.

You can use the cellmap file to specify the following property behaviors:

- Specify expressions based on Calibre property values. Properties are evaluated in the order that they appear in the cellmap file.
- Enable and disable CDF callback functions.
- Ignore properties from the schematic when creating the Calibre View.
- Ignore properties from the layout when creating the Calibre View.
- Specify which property values have priority for properties common to both schematic and layout.

You can use the formats in the following table to specify property behaviors in the cellmap file. See the section "Importing Schematic Properties for Calibre View" on page 774 for important information on case sensitivity related to importing schematic properties.

In the table, *x* is a Calibre property name and *y* is a Cadence property name.

**Table A-8. Defining Properties in the Cellmap File**

| Property Syntax | Description | Example |
|---|---|---|
| (*x* "") | Calibre property *x* is ignored when Calibre View is created. | (w "") |
| (nil *x*) | Property *x* is not copied from the schematic cell view instance. | (nil w) |
| (nil *x y*) | *x* property is set to *y* in the Calibre View regardless of its value in the schematic. | (nil SUB gnd) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-8. Defining Properties in the Cellmap File (cont.)**

| Property Syntax | Description | Example |
|---|---|---|
| (*x y* (expression)) | Cadence property *y* is set to the value of the expression. The only parameter available in the expression is the Calibre device property *x*. | (w W (w * 1e-06))<br>Also see "Using Expressions to Specify Properties in the Cellmap File" on page 769. |
| ( (x y) … (importSchemProp t \| nil)) | importSchemProp is an optional parameter that specifies whether or not schematic properties are imported for the device.<br>If importSchemProp is specified for a device, it overrides the global flag *mgc_eview_globals->importSchematicProperties*. If importSchemProp is not specified for a device, the global flag is used. | See "Example Cellmap with Property Mapping Including Fingers" on page 768. |
| (*x*(*y* t \| nil)) | *x* is the Calibre property name<br>*y* is the Cadence property name<br>Second element of list indicates whether the CDF callback function will be executed when setting the value of property *y*.<br>t: (default) The CDF callback is executed.<br>nil: The CDF callback is not executed.<br>The following syntax is equivalent:<br>(*x* (*y* t))<br>(*x* (*y*))<br>(*x y*)<br>Note: Both the t option and the global control must be set in order to execute callbacks. For the global control, you can set *mgc_eview_globals->executeCallbacks* to t or check the option "Execute Callbacks" in the Calibre View Setup dialog box. The default for the global option is nil. | (w (w t)) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-8. Defining Properties in the Cellmap File (cont.)**

| Property Syntax | Description | Example |
|---|---|---|
| (*x* (*y* t \| nil) (expression)) | Cadence property *y* is set to the value of the expression, which is calculated based on the value of the Calibre *x* property; execution of the callback associated with property *y* (if a callback exists) is determined by rules stated in the previous row description. | (w (W t) (w * 1e-06)) |

You can map a Calibre property to multiple Cadence cell view properties. This is done by specifying each property mapping separately, for example:

```
(w WA) ;Map Calibre property w to Cadence cell view property WA
(w WB) ;Map Calibre property w to Cadence cell view property WB
```

See "Using Expressions to Specify Properties in the Cellmap File" on page 769 for a similar example using property expressions.

See the command PEX BA Mapfile for a description of the LPE parameter, which is included by default in Calibre View netlists.

### Related Topics

Cellmap Syntax for Intentional Devices

Cellmap Syntax for Parasitic Devices

Layer Mapping Syntax

Cellmap Examples

Global Variables for Calibre View Generation

## Layer Mapping Syntax

Each layer that is imported to a Calibre View requires a mapping statement to identify the corresponding Cadence layer name.

### Layer Mapping Syntax

```
mgc_layer_map( list(
     list( "calibre_layer_name" '(CDS_layer_name CDS_layer_purpose))
     ...
     )
)
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Example

```
mgc_layer_map( list(
   list( "NWL" '(NWELL drawing))
   list( "OX"  '(OXIDE drawing))
   list( "PO1" '(POLY1 drawing))
   list( "CO"  '(CONTACT drawing))
   list( "M1"  '(METAL1 drawing))
    )
   )
```

See "Layer Map Statements to Optimize the Cellmap File" on page 770 for information on what layers to map.

See "Identifying Cadence Virtuoso Layers for Mapping" on page 771 for information on how to match the Calibre layer to the Cadence layer.

## Cellmap Syntax for Parasitic Devices

This section describes the cellmap syntax for parasitic devices.

The following parasitic devices are included:

Parasitic Capacitor

Parasitic Resistor

Parasitic Self-inductor

Parasitic coupling-element

### Parasitic Capacitor

#### Syntax

```
((p cap capacitance_property_name)
     (library_name symbol_name symbol)
     (
     (symbol_pin)
     (symbol_pin)
     )
)
```

#### Example

```
((p cap c)
   (analog_lib cap symbol)
   (

      (PLUS)
      (MINUS)
   )

)
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Parasitic Resistor

### Syntax

```
((p res resistance_property_name)
     (library_name symbol_name symbol)
     (
     (symbol_pin)
     (symbol_pin)
     )
)
```

You can also specify extra cell pins—see the section "Mapping with Fewer Device Pins" on page 767.

### Example

```
((p res r)
   (analog_lib res symbol)
   (

      (PLUS)
      (MINUS)
   )

)
```

## Parasitic Self-inductor

### Syntax

```
((p ind inductance_property_name)
     (library_name symbol_name view_name)
     (
     (symbol_pin)
     (symbol_pin)
     )
)
```

### Example

```
((p ind l)
   (analogLib ind symbol)
      (
      (MINUS)
      (PLUS)
      )
)
```

**Note**

Use the pin order shown in the example.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Parasitic coupling-element

### Syntax

```
((p mutk coupling_element_property_name)
     (library_name symbol_nameview_name)
     (
     (parameter1)
     (parameter2)
     )
)
```

### Example

```
((p mutk k)
    (analogLib mind symbol)
      (
      (ind1)
      (ind2)
      )
)
```

In this example, ind1 and ind2 are parameters of the mutual inductor and CalibreView will treat them exclusively as parameters, not as pins.

## Related Topics

Cellmap Syntax for Intentional Devices

Full Syntax for Property Specification in the Cellmap File

Layer Mapping Syntax

Cellmap Examples



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Cellmap Examples

The following cellmap file examples are included in this section:

**Specifying Devices with the Identical Modelname** . . . . . 766
**Mapping with Fewer Device Pins** . . . . . 767
**Mapping Extra Device Pins as Nets Mapped to Cell Properties** . . . . . 768
**Example Cellmap with Property Mapping Including Fingers** . . . . . 768
**Using Expressions to Specify Properties in the Cellmap File** . . . . . 769
**Layer Map Statements to Optimize the Cellmap File** . . . . . 770
**Identifying Cadence Virtuoso Layers for Mapping** . . . . . 771

## Specifying Devices with the Identical Modelname

You can have multiple devices with the same modelname, however each device should have different pin counts.

For example, the two statements in the following example are valid mappings for device type mp.

No errors or warnings occur if you map a Calibre device to different Cadence cellviews with the same number of pins, but this is not recommended.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Example

```
(mp ; 4-terminal pfet
   ( sample pfet4 symbol )
   (
      (g G)
      (s S)
      (d D)
      (b B)
   )
   (
      (w WIDTH) (l LENGTH)
   )
)
(mp ; 6-terminal pfet
   ( sample pfet6 symbol )
   (
      (g G)
      (s S)
      (d D)
      (b B)
      (b0 B0)
      (b1 B1)
   )
   (
      (w WIDTH) (l LENGTH)
   )
)
```

## Mapping with Fewer Device Pins

You can map a Calibre device to a Cadence cell containing a greater number of pins. For example, the following cellmap entry would map a 2-pin Calibre device to a 3-pin device:

```
(cap
   ( sample cap3 symbol )
   (
      (plus PLUS)
      (minus MINUS)
      (nil BULK "GND")
   )
)
```

In the generated Calibre View, the BULK pin of the cap3 cell is connected to the GND net in any placement of the capacitor device extracted by the Calibre tool.

If pin BULK does not exist, property BULK is placed on the cellview instance with value “GND”.

Another option is possible, allowing you to connect Cadence pins together:

```
(nil SUB (D)) ; Connect Cadence pin SUB to net connected to pin D
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If pin SUB does not exist on the cellview, a property is created on the cellview instance. Note that pin D has to appear before pin SUB in the pin mapping specification.

## Mapping Extra Device Pins as Nets Mapped to Cell Properties

You can map nets connecting to a pin of a device the Calibre tool extracts to a property on a cell the Calibre View instantiates.

Instead of specifying a terminal of the cell in the cellmap file, you can specify the name of the property the Calibre tool will create on the cell. You assign the net name the value of this property. For example, you can map a 4-pin Calibre device to a 3-terminal cell with a property on the fourth pin's net.

The following cellmap file statement will map the net connected to pin b of the Calibre pfet device to the BULK property on the 3-terminal pmos cell instance.

```
;
; map 4-terminal pfet Calibre device to 3-terminal pmos cell instance
;
(pfet
   ( sample pmos symbol ) ; 3-terminals on pmos cell
   (
      (g G)
      (s S)
      (d D)
      (b BULK) ; net connected to b will attach with value of BULK
   )
)
```

## Example Cellmap with Property Mapping Including Fingers

The schematic property "fingers" is set to the number of fingers of the device. Since each individual finger in the layout needs fingers=1, we need to set this in the cellmap file, or the schematic property value will be applied to each finger of the device.

Use this syntax to set the property "fingers" in the schematic:

```
(nil fingers 1) ;Property "fingers" is not in the layout
                ;but is in the schematic. Want to set fingers
                ;to a specific value
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For example:

```
(n
   (tsmc13lg nmos1v symbol)
   (
      (b B)
      (d D)
      (g G)
      (s S)
   )
   (
      (w fw) ;  map layout property "w" to schematic property "fw"
      (nil fingers 1) ; Property "fingers" is not in the layout
                      ; so set its value in the schematic
      (importSchemProp t) ; import schematic properties for this device
   )
)
(p
   (tsmc13lg pmos1v symbol)
   (
      (b B)
      (d D)
      (g G)
      (s S)
   )
   (
      (w fw) ;  map layout property "w" to schematic property "fw"
      (nil fingers 1) ; Property "fingers" is not in the layout
                      ; so set its value in the schematic
      (importSchemProp t) ; import schematic properties for this device
   )
)
```

Please note that if you attempt to set the “fw” property to the value of the “w” property for all devices by using the Reset Properties field in the Calibre View Setup dialog box (**Calibre** > **Setup** > **Calibre View**), it will not work properly because the Reset Properties field performs simple assignments. In performing these assignments the Reset properties field is not doing any variable or property de-referencing or expression evaluation. To get the value of a layout property assigned to schematic property for all devices, you should use the property mapping statement in each device, as shown in this cellmap example.

## Using Expressions to Specify Properties in the Cellmap File

This example uses expressions to map a Calibre property to Cadence properties.

The Calibre *w* property value is mapped to two different Cadence properties (*wscale1* and *wscale2*) using an expression. A conditional expression is used to map the Calibre *lpe* property. See the command PEX BA Mapfile for a description of the LPE parameter, which is included by default in Calibre View netlists.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
(p
   ( mgc_p4 pmos4 symbol)
   ( (b B) (d D) (g G) (s S) )
   ( (l l) (w w)  (m m)
      (w wscale1 (w*17))
      (w wscale2 (w*125))
      (lpe lpe_a (if lpe==0 || lpe==1 then lpe_a=1 else lpe_a=0))
   )
)
```

## Layer Map Statements to Optimize the Cellmap File

To create an optimized cellmap file, you should map the Connect statement layers and Device statement layers.

Inspect your LVS rule file for the following layers:

- Connect statement layers

  For example, if you have the following statement in your LVS rule file, you will need to map the metal1, metal2, and via1 layers to their corresponding Cadence layers:

  ```
  Connect metal1 metal2 by via1
  ```

- Device layers

  You need to include both seed layers and pin layers because the seed layer is the layer that enables you to recognize the device and the pin layers will most likely be included in the connect statements.

  For example, if you have the following statement in your LVS rule file, you will need to map the ngate, poly, nsd and psub layers to their corresponding Cadence Layers.

  ```
  DEVICE MN(nch) ngate poly(G) nsd(S) nsd(D) psub(B)
  ```

By mapping these two types of layers, you can generate a mask-Layout Calibre View as output of both Calibre LVS and Calibre xRC.

**Tip**

Layer mapping statements are required in the cellmap file when backannotating net geometries to the Calibre View. See the section "Specifying Extracted Netlist Annotation of Net Geometries to Calibre View" on page 339.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

Calibre LVS and xRC write layers used for either device definition or to define circuit connectivity to the standard verification database (SVDB). Some original or intermediate derived layers may not be available to import since they were never written to the SVDB.

If a layer is in the SVDB and has a cell mapping statement, but does not have a Cadence layer purpose pair (LPP) defined, Calibre will attempt to create that LPP and import the geometries.

### Related Topics

Creating the Cellmap File

Layer Mapping Syntax

Identifying Cadence Virtuoso Layers for Mapping

## Identifying Cadence Virtuoso Layers for Mapping

Calibre layers can be mapped Cadence drawn layers in the cellmap file, and you need to identify the corresponding Calibre layer and Cadence drawn layer.

As explained in Layer Map Statements to Optimize the Cellmap File, certain Calibre layers should be mapped to Cadence drawn layers in the cellmap file. These steps illustrate how to match the Calibre layers to the Cadence layers. See “Layer Mapping Syntax” on page 762 for a definition of the layer mapping statement.

### Procedure

1. In Cadence Virtuoso, choose **Tools > Technology File Manager** to display the Technology File Toolbox.
2. Click **Dump**. The Dump Technology File dialog box displays.
3. Select a “Technology Library” name from the dropdown menu; enable the “layerRules” options; enter a name for the file into the ASCII Technology File field; click **OK**.

   A Cadence technology file is output containing the layer names and their GDS numbers as shown in the following figure.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure A-2. Cadence Technology File**

```
; Technology File sphinx
; Generated on Aug 29 17:33:33 2006
;     with @(#)$CDS: icfb.exe version 5.1.0 06/24/2004 17:30 (intelibm5.Cadence.
COM) $


;*********************************
; LAYER RULES
;*********************************
layerRules(

 streamLayers(
 ;( layer       streamNumber    dataType        translate )
 ;( ------       ------------    --------        --------- )
  ( ("diff_ment" "P0")  2              0               t        )
  ( ("NP_ment"   "P0")   3              0               t        )
  ( ("PP_ment"   "P0")   4              0               t        )
  ( ("NWELL_ment" "P0") 1               0               t        )
  ( ("DNW" "P0")        5               0               t        )
  ( ("DG" "P0") 6               0               t        )
  ( ("GT" "P0") 30              0               t        )
  ( ("HRP" "P0")        39              0               t        )
  ( ("SN" "P0") 7               0               t        )
  ( ("SP" "P0") 8               0               t        )
  ( ("MVP" "P0")        44              0               t        )
  ( ("MVN" "P0")        45              0               t        )
  ( ("SAB" "P0")        48              0               t        )
  ( ("CT" "P0") 9               0               t        )
  ( ("MIM" "P0")        58              0               t        )
  ( ("M1_ment" "P0")    10              0               t        )
  ( ("M2_ment" "P0")    11              0               t        )
  ( ("M3_ment" "P0")    12              0               t        )
  ( ("M4_ment" "P0")    13              0               t        )
  ( ("M5_ment" "P0")    14              0               t        )
  ( ("M6_ment" "P0")    15              0               t        )
  ( ("Via1" "annotate") 16              0               t        )
  ( ("Via2" "annotate") 17              0               t        )
  ( ("Via3" "annotate") 18              0               t        )
  ( ("Via4" "annotate") 19              0               t        )
  ( ("Via5" "annotate") 20              0               t        )
  ( ("NAN" "annotate")  85              0               t        )
  ( ("HRPDUM" "annotate")       92              0               t        )
  ( ("Res_NW" "annotate")       95              0               t        )
  ( ("Res_P1" "annotate")       96              0               t        )
  ( ("Res_AA" "annotate")       97              0               t        )
  ( ("SUBD" "annotate") 300             0               t        )
  ( ("HAP" "annotate")  131             1               t        )
  ( ("BJTDM" "annotate")        134             0               t        )
  ( ("CAPBP" "annotate")        137             0               t        )
  ( ("DSTR" "annotate") 138             0               t        )
  ( ("M1Res_ment" "annotate")   171             0               t        )
  ( ("M2Res_ment" "annotate")   172             0               t        )
  ( ("M3Res_ment" "annotate")   173             0               t        )
  ( ("M4Res_ment" "annotate")   174             0               t        )
                                                                1,1           Top
```

4. Identify the device you want to map.

5. Determine the layers that device is composed of using the LVS rule file. The following excerpt shows the layers that make up the MN(nch) device.

```
// LVS rule file

Layer active 1
Layer nplus 2
Layer poly 3                <----- poly has a GDS number of 3

// NMOS Device
ndiff = active and nplus
ngate = poly and ndiff      <----- device contains poly
nsd = ndiff not ngate

DEVICE MN(nch) ngate poly(G) nsd(S) nsd(D)
```

6. Determine the mapping of device layers to drawn layers.

   For example, the layer ngate in the above rule file maps to the drawn layer poly, which has GDS layer number 3.

7. Check the Cadence technology file and find the Cadence layer names that corresponds to each GDS layer number you plan to map.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

In the example rule file from Step 5, we want to find the Cadence layer name for layer 3. From Figure A-2 (or the following excerpt), you can see that Cadence layer name PO has a GDS number of 3.

```
streamLayers(
;( Layer                streamNumber dataType translate )
;( -----                ------------ -------- --------- )
( ("act" "drawing")          1           0        t     )
( ("PO" "drawing")           3           0        t     )
```

8. In the cellmap file, map the Calibre layers to the Cadence layers you identified in the technology file.

   In this example, derived Calibre layer ngate corresponds to drawn layer poly, which is mapped to the Cadence layer PO.

```
;Layer mapping part of cell map file

mgc_layer_map( list(
   list( "ngate" '(PO drawing))
   list( "poly" '(PO drawing))
   list( "nsd" '(act drawing))
   )
)
```

9. Map the remaining device layers using the steps in this procedure.

   In the cellmap except shown with Step 8, Calibre layer poly is mapped to Cadence layer PO, and Calibre derived layer nsd is mapped to Cadence layer act.

## Setting the Calibre View Cellmap File in Calibre Interactive

You can specify the cellmap file for Calibre View generation using Calibre Interactive PEX or Calibre Interactive xACT. At run time, the Calibre View cellmap file is also entered in the Cadence Virtuoso “Calibre View Setup” dialog box.

**Note**

If the “Calibre View Setup” dialog box is already open at run time, the cellmap setting in the dialog box takes precedence over the setting in Calibre Interactive—the setting in the Calibre View Setup dialog box is used during the run and the dialog box setting is not updated with the Calibre Interactive setting.

In addition, the cellmap file defined in the CIW with mgc_eview_globals->cellMapFile takes precedence over the cellmap file defined in Calibre Interactive.

### Prerequisites

- A design is open in Cadence Virtuoso.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- The Calibre View Setup Dialog Box is not open.
- Calibre Interactive PEX or Calibre Interactive xACT is open and connected to the Cadence Virtuoso session.
- Calibre Interactive is set up to generate a Calibre View netlist.

  For Calibre Interactive PEX, see "Creating a Calibre View" on page 749. For an example, see "Creating a Post-Parasitic Extraction Calibre View" on page 782.

### Procedure

Do one of the following, depending on the Calibre Interactive application:

| For … | Do this … |
|---|---|
| Calibre Interactive PEX | 1. Click **PEX Options** on the left panel of Calibre Interactive to open the options pane. If the button is not present, choose **Setup** > **PEX Options** from the main menu.<br>2. Select the **Netlist** > **Advanced** tab.<br>3. In the CALIBREVIEW option area, check the "Cellmap File" checkbox and enter one or more Calibre View cellmap files. |
| Calibre Interactive XACT | 1. Click **Options** on the left panel of Calibre Interactive.<br>2. Check the "CALIBREVIEW Cellmap Files" checkbox.<br>3. Enter one or more cellmap files. |

### Results

The specified Calibre View cellmap file is used for Calibre View generation. If the Calibre View Setup dialog box pops up during the run, the Calibre Interactive setting for the cellmap file is present in the dialog box.

If multiple cellmap files are specified the files are concatenated in the order specified. If there are two cellmap statements for the same device, the last statement is used.

## Importing Schematic Properties for Calibre View

Because the Cadence Analog Design framework treats net and instance names in schematics case sensitively, you must add the Source Case statement to your SVRF rule file when creating a Calibre View.

For example:

```
Source Case YES
```

Adding this statement enables highlighting by the Calibre RVE tool in the Cadence schematics and the importation of schematic properties by the Calibre View generator.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

You may also need to use the SVRF statements Layout Case and LVS Compare Case, depending on your design.

The global variable mgc_eview_globals->importSchematicProperties controls whether schematic properties are imported; the default is t (or true). You can specify importSchemProp for a particular device in the cell map file in order to override the global setting. See these relevant topics:

- "Global Variables for Calibre View Generation" on page 800
- "Cellmap Syntax for Intentional Devices" on page 756
- "Device Property Calculation in Calibre View Generation" on page 752

You can specify mgc_eview_globals->debug_instance_properties_sources to print information to the log file about whether a parameter is from the schematic or layout.

When searching for a schematic instance in order to import properties, if an exact match by name is not found, then alternative schematic instances names are searched for. In particular, if no hierarchical schematic instance can be retrieved, properties are copied from the parent instance if it exists. For example, if the "I0/p0/_MP" instance is not placed in schematic, then properties are retrieved from "I0/p0", if it exists.

Also see "Creating a Calibre View" on page 749.

# Auto-Mapping Behavior, Control, and Debug in Calibre View Generation

Intentional devices and cells without mapping information in the cellmap file are automatically mapped (auto-mapped) and instantiated in the generated Calibre View. By default, the mapping uses case-sensitive matching of pin names followed by order-based mapping. You can specify a different auto-map behavior with the environment variable MGC_FDI_PIN_AUTOMAP_METHOD. You can also specify that information about the auto-map process be printed to the transcript with the environment variable MGC_FDI_ENABLE_AUTOMAP_LOGGING.

Auto-mapping often takes place in the gate-level extraction flow using LVS Box. The contents of the boxed cells are not extracted and do not appear in the Calibre View netlist. However, the boxed cells appear in the netlist as a device.

## Auto-Map Control

Auto-mapping is controlled with the environment variable MGC_FDI_PIN_AUTOMAP_METHOD. The allowed values are described in the following table.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-9. MGC_FDI_PIN_AUTOMAP_METHOD Values**

| Value | Auto-Map Behavior |
|---|---|
| 0 (default) | Apply case-sensitive pin name matching first, then apply order-based mapping. |
| 1 | Apply case-insensitive pin name matching first, then apply order-based mapping. |
| 2 | Apply case-sensitive pin name matching only. |
| 3 | Apply case-insensitive pin name matching only. |
| 4 | Apply order-based pin mapping only. |

## Auto-Map Behavior

If auto-mapping fails for a device, the device placement is skipped. Auto-mapping fails in these situations:

- The pin count in the Calibre View netlist is greater than the pin count in the master for the matching schematic instance. (Warning FDI3046)
- More than one Calibre View netlist pin maps to the same pin in the Cadence schematic view. (Warning FDI3047)
- A mapping cannot be determined. (Warning FDI3044)

  For example, if case-sensitive name matching only is specified and the Calibre View netlist pins are (“A” “B” “C” “D”), while the Cadence pins are (“E” “B” “A” “D”), auto-mapping fails.

Devices are auto-mapping with a warning in this case:

- The pin count in the Calibre View netlist is less than the pin count in the schematic. The extra pins are connected to the noConn device. (Warning FDI3045)

## Debug Output

Define the environment variable MGC_FDI_ENABLE_AUTOMAP_LOGGING to enable informational messages about the auto-mapping process. These messages are printed to the Calibre View log file (*calview.log* by default).

The following messages are possible:

- INFO: Auto mapping cell <*var*> successfully done according to the netlist.

  This message is printed when all pins are mapped by name.
- INFO: Auto mapping cell <*var*>, the following mapping is done: <pin mapping>



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

This message is printed when mapping only by name does not succeed and order-based mapping is used.

### Related Topics

Calibre View Generation Warnings

## Calibre View Setup Dialog Box

Open with: **Calibre** > **Setup** > **Calibre View** from Cadence Virtuoso

The Calibre View Setup dialog box has settings for several aspects of the Calibre View creation process.

**Table A-10. Calibre View Setup Dialog Box Contents**

| Option | Description |
|---|---|
| CalibreView Setup File | Specifies a CalibreView Setup File. Click **View** to view the file and **Load** to load the parameters into the Calibre View Setup dialog box. The parameters are not used unless you click **Load**.<br>This filename is also used as the name of the setup file when you click the **Save** button.<br>See "Creating a CalibreView Setup File" on page 791 for information on the setup file. |
| CalibreView Netlist File | The pathname of an existing CalibreView netlist file. This field is filled in by the tool when the dialog box is opened automatically during a Calibre Interactive run. |
| Output Library | The library for the design. |
| Schematic Library | The library for the schematic. |
| Cellmap File | The cellmap or iCellMap filename. See "Identifying Devices with a Cellmap File" on page 754 and "Syntax for Cellmap Statements" on page 756.<br>Multiple cellmap files may be specified; enter multiple files as a space-separated list. Files are concatenated in the order specified. If there are two cellmap statements for the same device, the last statement is used.<br>**Tip:** You can enter the cellmap file in Calibre Interactive PEX. See "Setting the Calibre View Cellmap File in Calibre Interactive" on page 773. Only one cellmap file may be specified in Calibre Interactive. |
| Log File | The name of the log file for Calibre View creation. |
| Calibre View Name | The view name for the Calibre View. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-10. Calibre View Setup Dialog Box Contents (cont.)**

| Option | Description |
|---|---|
| Calibre View Type | The view type, which can be “maskLayout” or “schematic.”<br>A Calibre View Type of “maskLayout” permits both geometries and schematic symbols to be drawn together. |
| Create Terminals | Selects the mode for creating terminals:<br>• When matching terminal exists on symbol<br>• Create all terminals |
| Preserve Device Case | Specifies whether to preserve device case in netlist files.<br>Select this option to ensure that mixed case netlist files have their capitalization maintained throughout Calibre processing. The case of pin names is always ignored, regardless of this setting. |
| Execute Callbacks | Specifies whether to execute CDF callback functions.<br>Also see “Full Syntax for Property Specification in the Cellmap File” on page 760 and mgc_eview_globals->executeCallbacks in “Global Variables for Calibre View Generation” on page 800. |
| Suppress Notes | Specifies whether informational notes are output to the transcript. |
| Reset Properties | Specify properties to reset to a specific value after being copied from the schematic. Only simple assignment is done; you cannot assign to a variable or another property value. |
| Magnify Instances By[1] | Magnification factor for intentional devices and parasitics.<br>The magnification factor is useful to prevent overlapping of instances when the Calibre View is generated in maskLayout view. Specify a magnification factor less than one to reduce overlap. |
| Device Placement | Specifies method for device placement:<br>• Layout Location<br>• Arrayed |
| Parasitic Placement | Specifies method for placement of parasitic elements:<br>• Layout Location<br>• Arrayed<br>Parasitic placement is only performed if “Calibre View Type” is set to “maskLayout.” “Parasitic Placement” should be set to “Layout Location” if net geometries are included in the Calibre View. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-10. Calibre View Setup Dialog Box Contents (cont.)**

| Option | Description |
|---|---|
| Show Parasitic Polygons | Specifies whether to show polygons for the parasitic elements.<br>If this setting is enabled, then “Parasitic Placement” must be set to “Layout Location” and “Calibre View Type” must be set to “maskLayout.” “Show Parasitic Polygons” should be enabled if net geometries are included in the Calibre View. |
| Open Calibre CellView | Specifies the mode for opening the Calibre View:<br>• Read-mode<br>• Edit-mode<br>• Don’t Open |
| Generate SPECTRE Netlist | Specifies whether to generate a SPECTRE netlist in addition to the Calibre View netlist. See “Creating a SPECTRE Netlist from the Calibre View Netlist” on page 790. |
| Always Show Dialog | Specifies whether to automatically open the Calibre View Setup dialog box during Calibre View creation. |
| **Buttons** | |
| OK | Create a CalibreView using the settings in the dialog box. |
| Cancel | Discard any modifications made in the dialog box and close it. |
| Save | Save the parameters specified in the dialog box to a setup file and close the dialog box. If a filename is specified in the “CalibreView Setup File” field, that filename is used, otherwise the default filename of *calibreview.setup* is used.<br>In addition, the parameters are used the next time you open the Calibre View Setup dialog box, and when mgc_rve_create_cellview is called. |

[1] For Cadence Virtuoso CDBA-based databases (Cadence Virtuoso version 5.1.4 and earlier) the dialog box has the entries “Magnify Devices By” and “Magnify Parasitics By”.

## Related Topics

Creating a CalibreView Setup File

Showing Parasitic Polygons in Calibre View

Specifying Extracted Netlist Annotation of Net Geometries to Calibre View



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Creating a Post-LVS Calibre View

Calibre Interactive and Calibre RVE generate a Calibre View for the top-level cell, regardless of the cell context set in the tools. The tools generate net and device names in the Calibre View from the source (or schematic) name space, unless nets and devices were unmatched during LVS comparison. In the latter case, the tool prefixes layout names with a "_Layout".

## Prerequisites

- Ensure you have the necessary input files. See "What You Need Before Creating a Calibre View" on page 750.
- If importing schematic properties, make sure Source Case, Layout Case, and LVS Compare Case SVRF statements are in your SVRF rule file.

  Because the Cadence Analog Design framework treats net and instance names in schematics case sensitively, you must use the following statement in your SVRF rule file:

  Source Case YES

  Adding this statement enables highlighting by the Calibre RVE tool in the Cadence schematics and the importation of schematic properties by the Calibre View generator. Also see "Importing Schematic Properties for Calibre View" on page 774.
- Create the cellmap file. See "Creating the Cellmap File" on page 754.
- (Optional) Set the environment variable CALIBRE_DISABLE_PPAR_WARNINGS to disable pPar evaluation warnings.

## Procedure

1. Open the layout in Cadence Virtuoso.
2. Choose **Calibre** > **Run nmLVS**.

   This invokes Calibre Interactive nmLVS.

   **Note**

   Leave the "Export from layout viewer" button checked on the **Layouts** tab of the Inputs pane.

3. Click the **Outputs** button.
4. In the **Report/SVDB** tab, do the following:
   a. Enable "Create SVDB Database."
   b. Enable "Start RVE after LVS finishes."
   c. Enter the "SVDB Directory" name.
   d. Enable the "Generate Calibre Connectivity Interface Data" option.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

5. Set all other required parameters for the Rules, Inputs, Outputs and LVS Options, as described in "Using Calibre Interactive to Perform LVS" on page 237.

6. Click **Run LVS**.

   Upon completion of the Calibre nmLVS-H run, Calibre RVE automatically opens with the results of the run.

7. In Calibre RVE, choose **Tools** > **Export CalibreView**.

   **Tip**

   If you did *not* open Calibre RVE from Calibre Interactive, choose **Highlight** > **Setup Cell Map** before creating the Calibre View and confirm that the cell mapping is correct. See "Map Calibre RVE Cells to Different Cadence Library for Highlighting" in the *Calibre RVE User's Manual*.

8. Click **OK** in the Export Calibre View dialog box.

9. Make selections in the Calibre View Setup dialog box. If the Calibre View Setup dialog box does not open automatically, choose **Calibre** > **Setup** > **Calibre View** and enable "Always Show Dialog." Most dialog settings are self-explanatory; refer to "Calibre View Setup Dialog Box" on page 777 or see below for tips on setting some of the fields:

   - **CalibreView Setup File** — Enter a CalibreView setup file then click **Load** to populate the dialog box with the settings. If the setup file includes all the necessary settings, proceed to Step 10. See "Creating a CalibreView Setup File" on page 791.
   - **CalibreView Netlist File** — Enter the name of an existing CalibreView netlist file. This field is filled in automatically when the dialog box is opened automatically during a Calibre Interactive run.
   - **Output Library** — Enter the library name for the design.
   - **Cellmap File** — Enter the cellmap filename. See "Cellmap Syntax for Intentional Devices" on page 756 for cellmap creation procedures. Multiple cellmap files may be specified; enter multiple files as a space-separated list. Files are concatenated in the order specified. You may also specify an iCellMap file.
   - **Calibre View Type** — Set the Calibre View Type to "maskLayout." This view permits both geometries and schematic symbols to be drawn together.
   - **Preserve Device Case** — Select this option to ensure that mixed case netlist files have their capitalization maintained throughout Calibre processing. The case of pin names is always ignored, regardless of this setting.
   - **Reset Properties** — Specify properties you want reset to a specific value after being copied from the schematic. Only simple assignment is done; you cannot assign to a variable or another property value.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Magnify Instances By** — Enter the magnification value the tool applies to intentional and parasitic devices. Specify a value less than one to prevent overlap of instances when generating a maskLayout Calibre View.
- **Generate SPECTRE Netlist** — Generate a SPECTRE netlist in addition to the Calibre View netlist. See "Creating a SPECTRE Netlist from the Calibre View Netlist" on page 790.

10. Click **OK** to create the CalibreView and save the settings in the dialog box.

**Note**

Wire stubs with net name labels are available in maskLayout view type. Similar to Schematics view type, the net names associated with the wires should also be created.

To make the wires selectable in CalibreView, choose **Options** > **Display**. In the Options dialog box, set both the X Snap Spacing and Y Snap Spacing fields to smaller values (e.g. 0.001); verify that the layer purpose pair ("wire" "drawing") is both visible and selectable in the Layer Selection Window.

### Results

The Calibre View netlist is created.

In addition, the file *calibreview.setup* is created in the run directory with the current Calibre View settings. This file is in the CalibreView Setup File Format. The file can be loaded into the Calibre View Setup dialog box to populate the dialog box with settings for the run. Browse for the file or enter the filename into the "Calibreview Setup File" field and click the **Load** button. It can also be used as input to the SKILL function mgc_rve_load_setup_file to create the Calibre View in batch mode.

### Related Topics

Mapping Schematic Nets to Calibre View Nets for Simulation and Plotting

Backannotation of Parasitic Values to the Schematic in Cadence Virtuoso

Showing Parasitic Polygons in Calibre View

Creating a Calibre View in Batch Mode

## Creating a Post-Parasitic Extraction Calibre View

The post-parasitic extraction Calibre View includes parasitic elements as well as intentional devices.

### Prerequisites

- Ensure you have the necessary input files. See "What You Need Before Creating a Calibre View" on page 750.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- If importing schematic properties, make sure Source Case, Layout Case, and LVS Compare Case SVRF statements are in your SVRF rule file.

  Because the Cadence Analog Design framework treats net and instance names in schematics with case sensitivity, you must use the following statement in your SVRF rule file:

  Source Case YES

  This statement enables highlighting by Calibre RVE in the Cadence schematics and the importation of schematic properties by the Calibre View generator. Also see "Importing Schematic Properties for Calibre View" on page 774.

- Create the cellmap file. See "Creating the Cellmap File" on page 754.
- (Optional) Set the environment variable CALIBRE_DISABLE_PPAR_WARNINGS to disable pPar evaluation warnings.

## Procedure

1. Open the layout in Cadence Virtuoso.
2. Choose **Calibre** > **Run Pex**.

   **Note**

   Leave "Export from layout viewer" enabled on the **Layout** tab of the Inputs pane.

3. Click the **Rules** button and enter the filename for the SVRF rule file. Also see "Specifying a Rule File for PEX Run" on page 299.
4. Click the **Load** button to load the rule file.
5. Click the **Inputs** button. Fill in the fields as described in "Inputs for a PEX Run" on page 301.
6. Click the **Outputs** button.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

7. Click the **Netlist** tab. For Format, select CALIBREVIEW from the dropdown button list, as shown.

8. Select SCHEMATIC for "Use Names From" on the **Netlist** tab of the Outputs pane.

   This specifies SOURCENAMES in PEX Netlist. See "Outputs for a PEX Run" on page 312 for details.

9. Set all other required parameters for the Outputs pane and for PEX Options, as described in "Outputs for a PEX Run" on page 312 and "Setting PEX Options" on page 324.

   Also see "Showing Parasitic Polygons in Calibre View" on page 792 for instructions on setting parameters to show parasitic polygons.

10. Click **Run PEX**.

    The tool creates the post-parasitic extraction view and automatically begins importing the Calibre View into the Cadence Environment.

11. Make selections in the Calibre View Setup dialog box. If the Calibre View Setup dialog box does not open automatically, choose **Calibre > Setup > Calibre View** from Cadence Virtuoso and enable "Always Show Dialog." Most dialog settings are self-explanatory; refer to "Calibre View Setup Dialog Box" on page 777 or see below for tips on setting some of the fields:

    - **CalibreView Setup File** — Enter a CalibreView setup file then click **Load** to populate the dialog box with the settings. If the setup file includes all the necessary settings, proceed to Step 12. See "Creating a CalibreView Setup File" on page 791.
    - **CalibreView Netlist File** — Enter the name of an existing CalibreView netlist file. This field is filled in automatically when the dialog box is opened during a Calibre Interactive run.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- **Output Library** — Enter the library name for the design.
- **Cellmap File** — Enter the cellmap filename. See "Cellmap Syntax for Intentional Devices" on page 756 for cellmap creation procedures. Multiple cellmap files may be specified; enter multiple files as a space-separated list. Files are concatenated in the order specified. You may also specify an iCellMap file.

  You can also specify the cellmap file in Calibre Interactive; see "Setting the Calibre View Cellmap File in Calibre Interactive" on page 773.
- **Calibre View Type** — Set the Calibre View Type to "maskLayout." This view permits both geometries and schematic symbols to be drawn together.
- **Preserve Device Case** — Select this option to ensure that mixed case netlist files have their capitalization maintained throughout Calibre processing. The case of pin names is always ignored, regardless of this setting.
- **Reset Properties** — Specify properties you want reset to a specific value after being copied from the schematic. Only simple assignment is done; you cannot assign to a variable or another property value.
- **Magnify Instances By** — Enter the magnification value the tool applies to intentional and parasitic devices. Specify a value less than one to prevent overlap of instances when generating a maskLayout Calibre View.
- **Generate SPECTRE Netlist** — Generate a SPECTRE netlist in addition to the Calibre View netlist. See "Creating a SPECTRE Netlist from the Calibre View Netlist" on page 790.

  **Tip**

  Symbols are generally smaller when viewed on top of a layout, so magnifying them is recommended.

12. Click **Save** to save the settings and exit the Calibre View Setup dialog box. The CalibreView is created during the Calibre run.

    The **OK** button creates the Calibre View immediately.

## Results

The CalibreView netlist is created.

In addition, the file *calibreview.setup* is created in the run directory with the current Calibre View settings. This file is in the CalibreView Setup File Format. The file can be loaded into the Calibre View Setup dialog box to populate the dialog box with settings for the run. Browse for the file or enter the filename into the "Calibreview Setup File" field and click the **Load** button. It can also be used as input to the SKILL function mgc_rve_load_setup_file to create the Calibre View in batch mode.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Mapping Schematic Nets to Calibre View Nets for Simulation and Plotting

Backannotation of Parasitic Values to the Schematic in Cadence Virtuoso

Showing Parasitic Polygons in Calibre View

Creating a Calibre View in Batch Mode

## Creating a Gate-Level Calibre View

A gate-level Calibre View includes parasitics from the top-level routing plus symbols for the cells.

### Prerequisites

- Ensure you have the necessary input files. See "What You Need Before Creating a Calibre View" on page 750.
- If importing schematic properties, make sure Source Case, Layout Case, and LVS Compare Case SVRF statements are in your SVRF rule file.

  Because the Cadence Analog Design framework treats net and instance names in schematics with case sensitivity, use the following statement in your SVRF rule file:

  Source Case YES

  This statement enables highlighting by Calibre RVE in the Cadence schematics and the importation of schematic properties by the Calibre View generator. Also see "Importing Schematic Properties for Calibre View" on page 774.
- Create the cellmap file. See "Creating the Cellmap File" on page 754.

  For a gate-level Calibre View, the cellmap file needs to include entries for each cell; these entries follow the syntax given in "Cellmap Syntax for Cells" on page 758.
- (Optional) Set the environment variable CALIBRE_DISABLE_PPAR_WARNINGS to disable pPar evaluation warnings.

### Procedure

1. Open the layout in Cadence Virtuoso.
2. Choose **Calibre > Run PEX**.

   **Note**

   Leave the "Export from layout viewer" option enabled on the **Layout** tab of the Inputs pane.

3. Click the **Rules** button and enter the filename for the SVRF rule file. Also see "Specifying a Rule File for PEX Run" on page 299.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

4. Click the **Load** button to load the rule file.
5. Click the **Inputs** button. Fill in the fields as described in "Inputs for a PEX Run" on page 301.
6. Click the **Outputs** button.
7. Click the **Netlist** tab.
   a. For "Extraction Type," specify "Gate Level."
   b. For "Format," select "CALIBREVIEW" from the dropdown button list, as shown.

   c. Select SCHEMATIC for "Use Names From" on the **Netlist** tab of the Outputs pane.

      This specifies SOURCENAMES in PEX Netlist. See "Outputs for a PEX Run" on page 312 for details.

8. Set all other required parameters for the Outputs pane and for PEX Options, as described in "Outputs for a PEX Run" on page 312 and "Setting PEX Options" on page 324.

   Also see "Showing Parasitic Polygons in Calibre View" on page 792 for instructions in setting parameters to show parasitic polygons.

9. Click **Run PEX**.

   The tool creates the post-parasitic extraction view and automatically begins importing the Calibre View into the Cadence environment.

10. Make selections in the Calibre View Setup dialog box. If the Calibre View Setup dialog box does not open automatically, choose **Calibre > Setup > Calibre View** from Cadence Virtuoso and enable "Always Show Dialog." Most dialog settings are self-



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

explanatory; refer to "Calibre View Setup Dialog Box" on page 777 or see below for tips on setting some of the fields:

- **CalibreView Setup File** — Enter a CalibreView setup file then click **Load** to populate the dialog box with the settings. If the setup file includes all the necessary settings, proceed to Step 11. See "Creating a CalibreView Setup File" on page 791.
- **CalibreView Netlist File** — Enter the name of an existing CalibreView netlist file. This field is filled in automatically when the dialog box is opened during a Calibre Interactive run.
- **Output Library** — Enter the library name for the design.
- **Cellmap File** — Enter the cellmap filename. See "Cellmap Syntax for Intentional Devices" on page 756 for cellmap creation procedures. Multiple cellmap files may be specified; enter multiple files as a space-separated list. Files are concatenated in the order specified. You may also specify an iCellMap file.

  You can also specify the cellmap file in Calibre Interactive; see "Setting the Calibre View Cellmap File in Calibre Interactive" on page 773.
- **Calibre View Type** — Set the Calibre View Type to "maskLayout." This view permits both geometries and schematic symbols to be drawn together.
- **Preserve Device Case** — Select this option to ensure that mixed case netlist files have their capitalization maintained throughout Calibre processing. The case of pin names is always ignored, regardless of this setting.
- **Reset Properties** — Specify properties you want reset to a specific value after being copied from the schematic. Only simple assignment is done; you cannot assign to a variable or another property value.
- **Magnify Instances by** — Enter the magnification value the tool applies to intentional and parasitic devices. Specify a value less than one to prevent overlap of instances when generating a maskLayout Calibre View.
- **Generate SPECTRE Netlist** — Generate a SPECTRE netlist in addition to the Calibre View netlist. See "Creating a SPECTRE Netlist from the Calibre View Netlist" on page 790.

  **Tip**

  Symbols are generally smaller when viewed on top of a layout, so magnifying them is recommended.

11. Click **Save** to save the settings and exit the Calibre View Setup dialog box. The CalibreView is created during the Calibre run.

    Note: The **OK** button creates the Calibre View immediately.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Results

The CalibreView netlist is created.

In addition, the file *calibreview.setup* is created in the run directory with the current Calibre View settings. This file is in the CalibreView Setup File Format. The file can be loaded into the Calibre View Setup dialog box to populate the dialog box with settings for the run. Browse for the file or enter the filename into the "Calibreview Setup File" field and click the **Load** button. It can also be used as input to the SKILL function mgc_rve_load_setup_file to create the CalibreView in batch mode.

### Related Topics

Mapping Schematic Nets to Calibre View Nets for Simulation and Plotting

Backannotation of Parasitic Values to the Schematic in Cadence Virtuoso

Showing Parasitic Polygons in Calibre View

Creating a Calibre View in Batch Mode

## Creating Calibre Views in a Multiple Corner Extraction Run

When you generate a Calibre View with multiple corners selected for extraction, a Calibre View is created for each corner.

### Prerequisites

- Ensure you have the necessary input files. See "What You Need Before Creating a Calibre View" on page 750.
- Calibre Interactive PEX is open.

### Procedure

1. Specify the corners to extract:
   a. Choose **Setup** > **PEX Options** to open the PEX Options pane.
   b. Click the **Netlist** > **Format** sub-tab.
   c. Check "Enable corner extraction" and choose the corners to extract.
2. Set up a Calibre Interactive run for Calibre View generation:
   - "Creating a Post-Parasitic Extraction Calibre View" on page 782
   - "Creating a Gate-Level Calibre View" on page 786
3. Click **Run PEX**.

   The tool creates the post-parasitic extraction view and automatically begins importing the Calibre View into the Cadence environment.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

4. Make selections in the Calibre View Setup dialog box. See the procedures noted in Step 2 for details on the dialog box entries.

   A dialog box opens for each corner that is extracted. The view name and netlist file are named automatically as follows, where the examples assume corners are named typical, MIN, and MAX.

   Calibre View name: calibre_<*corner*>

   Example: calibre_typical, calibre_MIN, calibre_MAX

   Calibre View netlist name:

   - For netlists with a period in the name, insert "_<*corner*>" before the last period, however, do not change the netlist name for a corner named typical.

     Example, for an original netlist name of *opamp.pex.netlist*: *opamp.pex.netlist*, *opamp.pex_MIN.netlist*, *opamp.pex_MAX.netlist*

   - For netlist without a period in the name, the Calibre View netlist name is <*netlist*>_<*corner*>.<*netlist_minus_last_character*>, however , do not change the netlist name for a corner named typical.

     Example, for an original netlist name of *opamp_netlist*: *opamp_netlist*, *opamp_netlist_MIN.opamp_netlis*, *opamp_netlist_MAX.opamp_netlis*

   Click **OK** to close each Calibre View Setup dialog box.

# Creating a SPECTRE Netlist from the Calibre View Netlist

You can generate a SPECTRE netlist from the Calibre View netlist using the Cadence Analog Design Environment (ADE). Cellmap information is used in the netlist generation, unlike the case when generating a SPECTRE netlist from Calibre xRC, in which source names are used.

## Prerequisites

- Ensure you have the necessary input files. See "What You Need Before Creating a Calibre View" on page 750.

- If importing schematic properties, then manually add Source Case, Layout Case, and LVS Compare Case SVRF statements to your SVRF rule file.

  Because the Cadence Analog Design framework treats net and instance names in schematics case sensitively, you must add the following statement to your SVRF rule file:

  Source Case YES

  Adding this statement enables highlighting by the Calibre RVE tool in the Cadence schematics and the importation of schematic properties by the Calibre View generator. Also see "Importing Schematic Properties for Calibre View" on page 774.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Create the cellmap file. See "Creating the Cellmap File" on page 754.
- (Optional) Set the environment variable CALIBRE_DISABLE_PPAR_WARNINGS to disable pPar evaluation warnings.

### Procedure

1. Set up to produce a Calibre View netlist as described in "Creating a Post-LVS Calibre View" on page 780 or "Creating a Post-Parasitic Extraction Calibre View" on page 782.
2. In the Calibre View Setup dialog box, enable "Generate SPECTRE Netlist."

   If the Calibre View Setup dialog box does not open automatically, choose **Calibre** > **Setup** > **Calibre View** from Cadence Virtuoso.
3. Click **Save** to save the settings and close the Calibre View Setup dialog box.

   The Map Calibre Device dialog box opens automatically.
4. Make appropriate selections in the Map Calibre Device dialog box, then close it.

   The Cadence Virtuoso Analog Design Environment window opens, and then a window with the SPECTRE netlist opens.
5. In the window with the SPECTRE netlist, choose **File** > **Save As** to save the netlist.

   **Tip**

   Once a Calibre View netlist has been created, you can create the SPECTRE netlist from the CIW with the following command:

   mgc_eview_run_netlister("lib_name" "cell_name" "view_name")

## Creating a CalibreView Setup File

The CalibreView setup file contains all the settings necessary to generate a CalibreView. You can use the file as input to the SKILL function mgc_rve_load_setup_file to create the CalibreView in batch mode, without opening the Calibre View Setup dialog box. The file can also be loaded into the Calibre View Setup dialog box in order to populate the dialog box with saved settings.

**Note**

The file *calibreview.setup* is automatically created in the run directory after Calibre View generation. The file contains the Calibre View settings and is in the CalibreView Setup File Format. You can edit this file to create a setup file with different parameters.

You can also create a setup file using the Calibre View Setup dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Procedure

1. Do one of the following, depending on whether you have generated a Calibre View:

| If … | Do this … |
|---|---|
| You have generated a Calibre View and want to use most of the same parameters | Copy *calibreview.setup* to another filename, such as *newcalibreview.setup*. |
| You have *not* generated a Calibre View and want a default setup file | 1. Execute the SKILL function:<br>`mgc_rve_default_setup_file`<br>This creates the file *.calibreview_default_setup_file* in the directory you started Cadence Virtuoso from.<br>2. Copy *.calibreview_default_setup_file* to another filename, such as*newcalibreview.setup*. |
| You want to enter parameters for the setup file from the Calibre View Setup dialog box | 1. From Cadence Virtuoso, select **Calibre** > **Setup** > **Calibre View**.<br>2. Enter the filename for the setup file in the "CalibreView Setup File" field, or leave it blank to use the default filename of *calibreview.setup*.<br>3. Enter parameters in the dialog box.<br>4. Click the **Save** button at the bottom of the dialog box. |

2. Edit your CalibreView setup file (*newcalibreview.setup* for example) and provide required file names, library settings, and other parameters. See "CalibreView Setup File Format" on page 794 for details.

   This step is not necessary if you created the setup file using the Calibre View Setup dialog box and entered all the desired parameters.

3. Save and exit the file.

**Note**

See "Creating a Calibre View in Batch Mode" on page 801 for instructions on creating the CalibreView in batch mode using the CalibreView setup file.

You can also load the file into the Calibre View Setup dialog box—browse to the file and click **Load** to populate the dialog box with the settings.

## Showing Parasitic Polygons in Calibre View

You can use the display and annotation of parasitic polygons to more easily analyze design parasitics.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

See "Specifying Extracted Netlist Annotation of Net Geometries to Calibre View" on page 339.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# CalibreView Setup File Format

Input for: The SKILL function mgc_rve_load_setup_file and the Calibre View Setup dialog box

The CalibreView setup file allows you to specify the parameters for CalibreView creation without opening the Calibre View Setup dialog box.

See "Creating a CalibreView Setup File" on page 791 for instructions on creating a setup file. The file *calibreview.setup* is automatically created in the run directory after Calibre View generation. The file contains the Calibre View settings.

## Format

- ASCII text file. Blank lines are ignored, and white space at the beginning of a line is ignored.
- Comments are not allowed.
- Keywords and values are not case-sensitive.
- Only one keyword/value pair is allowed on each line.

### File Contents

Each line has the following format:

```
Keyword: value
```

where the allowed keywords and values are given in the next section.

## Parameters

The allowed keywords and values for the CalibreView setup file are given in the following table.

**Table A-11. CalibreView Setup File Keywords and Values**

| Keyword | Required or Optional | Allowed Values | Default Value |
|---|---|---|---|
| calibre_view_netlist_file | Required | A valid pathname for a CalibreView netlist. | None |
| output_library | Required | A valid Cadence library. | None |
| schematic_library | Required | A valid Cadence library. | None |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-11. CalibreView Setup File Keywords and Values (cont.)**

| Keyword | Required or Optional | Allowed Values | Default Value |
|---|---|---|---|
| cell_name | Optional | The cell name of the generated CalibreView.<br>This field is filled in by the tool when the *calibreview.setup* file is created during a run. The field is not required for successful batch generation of the Calibre View. | None |
| cellmap_file | Required | A valid CalibreView cellmap file.<br>Multiple cellmap files may be specified; enter multiple files as a space-separated list. Files are concatenated in the order specified. | None |
| calibreview_log_file | Optional | A valid path to the log file. | ./calview.log |
| calibreview_name | Optional | A valid name. | calibre |
| calibreview_type | Optional | • maskLayout<br>• schematic<br>calibreview_type must be set to maskLayout to include net geometries in the CalibreView. | maskLayout |
| create_terminals | Optional | • if_matching<br>• all | if_matching |
| preserve_device_case | Optional | • on<br>• off | off |
| execute_callbacks | Optional | • on<br>• off | off |
| suppress_notes | Optional | • on<br>• off | off |
| reset_properties | Optional | A valid list of properties in this format:<br>(prop=value,prop=value,…) | (m=1) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-11. CalibreView Setup File Keywords and Values (cont.)**

| Keyword | Required or Optional | Allowed Values | Default Value |
|---|---|---|---|
| magnify_devices_by | Optional | A valid positive value.<br>For OpenAccess databases[1], this keyword affects magnification of both intentional and parasitic devices, and corresponds to “Magnify Instances By” in the Calibre View Setup dialog box.<br>The magnification factor is useful to prevent overlapping of instances when the Calibre View is generated in maskLayout view. Specify a magnification factor less than one to prevent overlap. | 1 |
| magnify_parasitics_by | Optional | A valid positive value.<br>Not used with OpenAccess databases[1]. | 1 |
| device_placement | Optional | • layout_location<br>• arrayed | layout_location |
| parasitic_placement | Optional | • layout_location<br>• arrayed<br>Parasitic placement is only performed if calibreview_type is set to maskLayout.<br>parasitic_placement should be set to layout_location if net geometries are included in the CalibreView. | arrayed |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-11. CalibreView Setup File Keywords and Values (cont.)**

| Keyword | Required or Optional | Allowed Values | Default Value |
|---|---|---|---|
| show_parasitic_polygons | Optional | • on<br>• off<br>If this value is on, then parasitic_placement must be set to layout_location and calibreview_type must be set to maskLayout.<br>show_parasitic_polygons should be set to on if net geometries are included in the CalibreView. | off |
| open_calibreview | Optional | • read_mode<br>• edit_mode<br>• don't_open | don't_open |
| generate_spectre_netlist | Optional | • on<br>• off | off |

[1]For CDBA-based databases, the keywords magnify_devices_by and magnify_parasitics_by correspond to “Magnify Devices by” and “Magnify Parasitics by” in the Calibre View Setup dialog box.

## Examples

The following is an example of a CalibreView setup file:

```
calibre_view_netlist_file : ./adder_4.pex.netlist
output_library : ADDER
schematic_library : ADDER
cell_name : adder_4
cellmap_file : ./calview.cellmap
calibreview_log_file : ./calview.log
calibreview_name : calibre
calibreview_type : maskLayout
create_terminals : if_matching
preserve_device_case : off
execute_callbacks : off
reset_properties : (m=1)
magnify_devices_by : 1
magnify_parasitics_by : 1
device_placement : layout_location
parasitic_placement : arrayed
show_parasitic_polygons : off
open_calibreview : don't_open
generate_spectre_netlist : off
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Related Topics

Creating a CalibreView Setup File

Identifying Devices with a Cellmap File

## SKILL Functions for Creating a Calibre View

Several functions are provided that allow you to create a Calibre View from the command interpreter without opening the Calibre Interactive GUI or Calibre RVE.

mgc_rve_default_setup_file
mgc_eview_display_setup_settings
mgc_eview_export_from_layout
mgc_eview_export_from_schematic
mgc_eview_post_execution_trigger
mgc_rve_load_setup_file
mgc_eview_run_netlister
mgc_rve_create_cellview
mgc_rve_run_pex

### mgc_rve_default_setup_file

**mgc_rve_default_setup_file**

Output a file named *.calibreview_default_setup_file* to the directory Cadence Virtuoso was invoked from. The file *.calibreview_setup_file* is in the correct format for the CalibreView setup file and has default settings for most parameters. See the function mgc_rve_load_setup_file() and "Creating a CalibreView Setup File" on page 791.

### mgc_eview_display_setup_settings

**mgc_eview_display_setup_settings("*lib_name*" "*cell_name*" "*view_name*")**

Display the Calibre View setup settings that were used to create the specified cell view.

### mgc_eview_export_from_layout

**mgc_eview_export_from_layout( "*libName*" "*cellName*" "*viewName*")**

Export the netlist from the layout using the specified library name, cell name, and view name.

### mgc_eview_export_from_schematic

**mgc_eview_export_from_schematic( "*libName*" "*cellName*" "*viewName*")**

Export the netlist from the schematic using the specified library name, cell name, and view name.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## mgc_eview_post_execution_trigger

**mgc_eview_post_execution_trigger**("***layLibName***" "***layCellName***" "***layViewName***" "***schLibName***" "***schCellName***" "***schViewName***" "***calView***")

A trigger function that is called after the view is created and saved for the Calibre View, and after the SPECTRE netlist is generated. The trigger function is called with the indicated arguments, where *calView* is the handle to the new view. Define this function in your Cadence environment in order to use it.

## mgc_rve_load_setup_file

**mgc_rve_load_setup_file**("***calibreview_setup_file***")

Create a CalibreView using the parameters in the *calibreview_setup_file*. You can use mgc_rve_default_setup_file() to create a default file with the correct format. See "CalibreView Setup File Format" on page 794.

## mgc_eview_run_netlister

**mgc_eview_run_netlister("*lib_name*" "*cell_name*" "*view_name*")**

Generate a SPECTRE netlist from the Calibre View netlist with the specified library, cell, and view. Also see "Creating a SPECTRE Netlist from the Calibre View Netlist" on page 790.

## mgc_rve_create_cellview

**mgc_rve_create_cellview**(@optional "*calibreview_netlist*")

Create the Calibre View using the specified netlist. The Calibre View Setup dialog box opens in order to provide necessary Calibre View generation information. The netlist name is filled in if provided.

## mgc_rve_run_pex

**mgc_rve_run_pex("*batch_script*")**

Executes the shell commands in the file *batch_script*.

**Tip**

Also see "Global Variables for Calibre View Generation" on page 800.

## Related Topics

Creating a Calibre View

Creating a Calibre View in Batch Mode

Creating a CalibreView Setup File



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Global Variables for Calibre View Generation

Several global variables are provided that are used in the creation of a Calibre View.

**Table A-12. Global Variables for Calibre View Generation**

| Variable | Definition |
|---|---|
| mgc_eview_globals->cellMapFile | Path to cell map file |
| mgc_eview_globals->debug_instance_properties_sources | Specifies whether to print information to the log file about whether a parameter is from the schematic or layout. Allowed values are t or nil. |
| mgc_eview_globals->executeCallbacks | Specifies whether to execute callbacks. The default value is nil. Set this variable to t in order for callbacks to be executed. |
| mgc_eview_globals->extViewName | Specifies the extracted view name to search in when mapping schematic nets.<br>See "Mapping Schematic Nets to Calibre View Nets for Simulation and Plotting" on page 810. |
| mgc_eview_globals->extViewType | View type |
| mgc_eview_globals->importSchematicProperties | Specifies whether to import schematic properties. Allowed values are t or nil. The default is t.<br>The global setting can be overridden for a particular device with the importSchemProp setting in the cellmap file; see "Cellmap Syntax for Intentional Devices" on page 756.<br>Also see "Device Property Calculation in Calibre View Generation" on page 752. |
| mgc_eview_globals->outputLibrary | Output library |
| mgc_eview_globals->pathSeparator | The path separator. |
| mgc_eview_globals->schematicLibrary | Schematic library. The default setting is outputLibrary. |
| mgc_eview_globals->showCalviewDlg | Specifies whether or not to automatically open the Calibre View Setup dialog box. Set to t or nil. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Creating a Calibre View in Batch Mode

The SKILL function mgc_rve_load_setup_file() uses the parameters in the CalibreView setup file to create a CalibreView without opening the Calibre View Setup dialog box.

## Prerequisites

- "What You Need Before Creating a Calibre View" on page 750

## Procedure

1. Run Calibre, either from the command interpreter or as a batch run from the shell.

   To run calibre from the command interpreter, use the commands defined in the section "SKILL Functions for Creating a Calibre View" on page 798.

2. Create the CalibreView setup file as described in "Creating a CalibreView Setup File" on page 791.

   If the CalibreView includes net geometries make sure the CalibreView setup File parameters are set accordingly and that the following conditions are met:

   - The cellmap file includes layer mapping
   - The query_results file generated by Calibre RVE exist in the run directory

3. Create the Calibre View by calling this function from the command interpreter or a SKILL file:

   ```
   mgc_rve_load_setup_file("calibreview_setup")
   ```

   where *calibreview_setup* is the CalibreView setup file created in Step 2.

## Results

The following behavior takes place if errors are encountered while reading the CalibreView setup file, depending on whether the parameter is required or optional (see the parameter definitions in "CalibreView Setup File Format" on page 794):

- **Required keyword** — Exit with an error message.
- **Optional keyword** — Use the default value, issue a warning, and continue.

The Calibre View is written to the file specified in the CalibreView setup file if no errors are encountered.

The termination message from Calibre View is printed to the CDS.LOG file as follows:

```
Calibre View generation completed with <num> WARNINGs and <num> ERRORs.
Please consult the CIW transcript for messages.
```

## Related Topics

Importing Schematic Properties for Calibre View

Showing Parasitic Polygons in Calibre View



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Specifying Extracted Netlist Annotation of Net Geometries to Calibre View

## Calibre View and IC Manage Revision Control

Calibre View creation is designed to work with the IC Manage integration to Cadence Virtuoso.

New or updated Calibre Views are created in the add mode and not checked in automatically; checkin must be done by the user.

- **New Calibre View** — The new view is created in the "Open for Add" state. The view must be checked in to save the view; the checkin is not done automatically.
- **Existing Calibre View** — If the Calibre View already exists in IC Manage the view is checked out and a new version is created in "Open for Edit" mode. The view must be checked in to save the view; the checkin is not done automatically.

See "Creating a Calibre View" on page 749 for further information.

## Calibre View Generation Warnings

This section describes warnings from Calibre View generation.

The number of warnings printed to the transcript can be controlled with the environment variable FDI_DBDIFF_SUPPRESS_VALUES. See "FDI Environment Variables" in the *Calibre Layout Comparison and Translation Guide*. For example, to limit the number of warnings for FDI3034 to 20, set FDI_DBDIFF_SUPPRESS_VALUES to "FDI3034 20".

**Table A-13. Calibre View Warnings**

| Warning # | Warning Text (Explanation) |
|---|---|
| FDI3009 | WARNING: [FDI3009] Unsupported parasitic type <*type*> received.<br>(Occurs when an unsupported parasitic type is specified in netlist file.) |
| FDI3013 | WARNING: [FDI3013] Unable to connect instance <*inst*> to net <*net*>.<br>(Occurs when it is not possible to connect an instance to the net.) |
| FDI3014 | WARNING: [FDI3014] Could not find cell mapping for device <*dev*>. Ignoring instance <*inst*>.<br>(Occurs when there is no cell mapping for a device in the cellMap file.) |
| FDI3017 | WARNING: [FDI3017] Could not find schematic view of cell <*cell*> to copy properties from.<br>(Occurs when it is not possible to open the schematic view of a cell.) |
| FDI3018 | WARNING: [FDI3018] Library <*lib*> not found for device <*dev*>. Ignoring instance <*inst*>.<br>(Occurs when the library is not found for a device.) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-13. Calibre View Warnings (cont.)**

| Warning # | Warning Text (Explanation) |
|---|---|
| FDI3020 | WARNING: [FDI3020] Could not find symbol or schematic for cell <*cell*>. All terminals in view <*view*> will be of type inputOutput.<br>(Occurs when it is not possible to open symbol or schematic views for a cell.) |
| FDI3021 | WARNING: [FDI3021] Creating an inputOutput terminal <*term*> in (<*cell*> <*view*>) since matching terminal was not found on <*symView_or_schView*>.<br>(Occurs when -createUnmatchedTerminals option is specified.) |
| FDI3022 | WARNING: [FDI3022] Not creating terminal <*term*> in (<*cell*> <*view*>) since matching terminal was not found on <*symView_or_schView*>.<br>(Occurs when the corresponding terminal does not exist in the symbol view and the -createUnmatchedTerminals option is not set.) |
| FDI3023 | WARNING: [FDI3023] Additional entry for parasitic in cellmap. ignoring parameter <*param*>.<br>(Occurs when an additional entry is defined for the parasitic devices in cellmap file.) |
| FDI3024 | WARNING: [FDI3024] Can't find layermap entry for layer <*layer*> in cellmap file.<br>(This warning occurs when a layer from netlist file does not exist in the cellmap file.) |
| FDI3025 | WARNING: [FDI3025] Invalid layer-purpose pair <*layer*> <*purpose*>.<br>(Occurs when an invalid layer-purpose pair specified in the cellmap file.) |
| FDI3026 | WARNING: [FDI3026] lpe construction <*expr*> is not supported.<br>(Occurs when the syntax of lpe construction is not correct, if its value is not numeric.) |
| FDI3027 | WARNING: [FDI3027] lpe value for device <*dev*> is not specified in netlist file.<br>(Occurs when the lpe value is not specified for the device in the Calibre View netlist file.) |
| FDI3028 | WARNING: [FDI3028] Failed to open cellview <*lib*>/<*cell*>/<*view*> because cellView database does not exist.<br>(Occurs when the cellvew database does not exist.) |
| FDI3029 | WARNING: [FDI3029] Failed to open schematic cellview <*lib*>/<*cell*>/ <*view*> from SKILL.<br>(Occurs when it is not possible to open a schematic cellview from SKILL.) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-13. Calibre View Warnings (cont.)**

| Warning # | Warning Text (Explanation) |
|---|---|
| FDI3030 | WARNING: [FDI3030] Could not evaluate property <*prop*> for the instance: <*inst*>.<br>(Occurs when a property value is not specified for the instance.) |
| FDI3031 | WARNING: [FDI3031] Cellview <*cell*> (<*view*>) does not exist in library <*lib*>.<br>(Occurs when the cellview does not exist in the library.) |
| FDI3032 | WARNING: [FDI3032] Could not create instances of parasitic device "<*dev*>": Cell mapping information not specified.<br>(Occurs when the cell mapping information is not specified for an instance of a parasitic device.) |
| FDI3033 | WARNING: [FDI3033] Schematic instance <*inst*> not found.<br>(Occurs when a schematic instance from the netlist file is not found.) |
| FDI3034 | WARNING: [FDI3034] Schematic instance <*inst*> not found, use found instance <*inst*> instead.<br>(Occurs when the schematic instance is not found.) |
| FDI3035 | WARNING: [FDI3035] Terminal <*term*> does not exist on cells: (<*cells*>). Nets will be added as <*prop*> property values.<br>(Occurs f the terminal does not exist in the cell but does exists in the cdf properties list.) |
| FDI3036 | WARNING: [FDI3036] Terminal <*term*> does not exist on cells: (<*cells*>).<br>(Occurs when the cell does not contain the required terminal for connection.) |
| FDI3037 | WARNING: [FDI3037] Copying properties from schematic instance of <*schLib:schCell:schView*> to instance of <*lib:cell:view*>.<br>(Occurs when the cellviews of the matched schematic instance and the generating instance are different.) |
| FDI3038 | WARNING: [FDI3038] Wrong cellmap syntax detected. Block is partly ignored.<br>(Occurs when unknown cellmap syntax is detected. In this case the entire cell's block is ignored.) |
| FDI3039 | WARNING: [FDI3039] Failed to evaluate expression <*expr*>. Leaving value <*val*>.<br>(Occurs when an expression evaluation fails and the value from the netlist file is mapped as is.) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-13. Calibre View Warnings (cont.)**

| Warning # | Warning Text (Explanation) |
|---|---|
| FDI3044 | WARNING: [FDI3044] Failed to create mapping for device <*deviceName*>. Placement is skipped.<br>(Occurs when the auto-mapping process cannot determine a mapping. For example, if case-sensitive name mapping only is specified and the Calibre View netlist pins are (“A” “B” “C” “D”), while the Cadence pins are (“E” “B” “A” “D”), auto-mapping fails.) |
| FDI3045 | WARNING: [FDI3045] Netlist for <*deviceName*> has fewer pins than schematic view. The following pins are missing and will be connected to noConn device: <*pin_list*>.<br>(Occurs during auto-mapping fails if the pin count in the Calibre View netlist is less than the pin count in the schematic. The extra pins are connected to the noConn device.) |
| FDI3046 | WARNING: [FDI3046] Failed to create mapping for device <*deviceName*>. Netlist for <*deviceName*> has more pins than schematic view.<br>(Occurs when auto-mapping fails because the pin count in the Calibre View netlist is greater than the pin count in the master for the matching schematic instance.) |
| FDI3047 | WARNING: [FDI3047] Failed to create mapping for device <*deviceName*>. More than one netlist pins mapped to one pin from schematic view.<br>(Occurs when auto-mapping fails because more than one Calibre View netlist pin maps to the same pin in the Cadence schematic view.) |
| FDI3048 | WARNING: [FDI3048] Incorrect mapping of bus pins detected in cellmap file. Block is ignored.<br>(Occurs if the order of bus pins in the cellmap file are different or the number of bus pins are not the same.) |
| FDI3050 | WARNING: [FDI3050] NET GEOMETRY: <*file_name_line_number*>: Invalid statement <*statement_name*>.<br>(Occurs when the *query_results* file has an invalid specification statement.) |
| FDI3051 | WARNING: [FDI3051] NET GEOMETRY: The attached technology library was not found.<br>(Occurs when the design has no attached technology file.) |
| FDI3052 | WARNING: [FDI3052] NET GEOMETRY: Could not find layer <*layer_name*>. Skipping objects on Calibre layer <*calibre_layer_name*> in the generated layout in Calibre View.<br>(Occurs when the mapped layer is missing from the technology library.) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-13. Calibre View Warnings (cont.)**

| Warning # | Warning Text (Explanation) |
|---|---|
| FDI3053 | WARNING: [FDI3053] NET GEOMETRY: Calibre layer <*layer_name*> is mapped to nil. Skipping objects on that layer.<br>(Occurs when the Calibre layer is mapped to nil in the cellmap file.) |
| FDI3054 | WARNING: [FDI3054] NET GEOMETRY: No default purpose set. Skipping objects on Calibre layer <*layer_name*>.<br>(Occurs when the technology library does not have default purpose.) |
| FDI3056 | WARNING: [FDI3056] NET GEOMETRY: Could not find layer-purpose <*purpose_name*>. Skipping objects on Calibre layer <*layer_name*>.<br>(Occurs when the Calibre layer's purpose from the cellmap file does not exist in the technology library.) |
| FDI3059 | WARNING: [FDI3059] NET GEOMETRY: Could not find net <*net_name*>.<br>(Occurs when the net name in the mgc_rve_net statement does not exist in the generated view.) |
| FDI3060 | WARNING: [FDI3060] NET GEOMETRY: Could not find “query_results” file.<br>(Occurs when running MaskLayout with Net Geometry flow if the *query_results* file has not been created within one minute of invoking the flow.) |
| FDI3061 | WARNING: [FDI3061] File <*file_name*> specified by “mgc_point_to_point” statement could not be opened for reading.<br>(Occurs when the file specified by a “mgc_point_to_point” statement does not exist or is not readable.) |
| FDI3062 | WARNING: [FDI3062] The device <*device_name*> with *pins_count* pins is already defined in the cellmap file. Ignoring the previous definitions.<br>(Occurs when the callmap contains multiple definitions of the device with the same pin count.) |
| FDI3063 | WARNING: [FDI3063] Terminal with name “<*terminal_name*>” already exists in the design.<br>(Occurs when a “mgc_rve_cell_start” statement has more than one terminal with the same name.) |
| FDI3064 | WARNING: [FDI3064] Failed to open cellview *lib/cell/view* from SKILL.<br>(Occurs when the generating view fails to open from SKILL.) |
| FDI3065 | WARNING: [FDI3065] Failed to close cellview from SKILL.<br>(Occurs when the generating view fails to close from SKILL.) |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-13. Calibre View Warnings (cont.)**

| Warning # | Warning Text (Explanation) |
|---|---|
| FDI3066 | WARNING: [FDI3066] Failed to create pCell instance on SKILL side.<br>(Occurs when creating pCell instances using SKILL procedures fails.) |
| FDI3067 | WARNING: [FDI3067] Failed to get pCell schematic instance <*instance_name*> from SKILL.<br>(Occurs when retrieving the pCell schematic instance using Skill procedures fails.) |
| FDI3068 | WARNING: [FDI3068] <*file_name_line_number*> Invalid Point2Point statement: <*statement_text*><br>(Occurs when a RESISTANCE statement is not valid.) |

## Related Topics

Auto-Mapping Behavior, Control, and Debug in Calibre View Generation



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Setting the Bus Delimiter for Schematic Netlists

The delimiter for bus names may be <> or [], depending on the convention used in the schematic netlist. You can convert the bus delimiter, if needed.

**Convert Bus Name Delimiter During Schematic Netlist Export** . . . . . 808
**Automatically Convert Bus Name Delimiter for Calibre Interactive and Calibre RVE** 808

## Convert Bus Name Delimiter During Schematic Netlist Export

You can instruct the tool to convert the bus name delimiter from <> to [] when the schematic netlist is exported for a Calibre Interactive run.

### Procedure

1. Open the schematic in Cadence Composer.
2. Choose **Calibre > Setup > Netlist Export**.
3. Enable “Map Bus Name From <> to [].” You may need to scroll down in the dialog box to see the selection.

## Automatically Convert Bus Name Delimiter for Calibre Interactive and Calibre RVE

In some cases Calibre Interactive or Calibre RVE is using [] as the bus name delimiter but the tool is interfacing with a Cadence schematic view that uses <> as the bus name delimiter. You can convert the bus name delimiter so that net selection and highlighting is handled correctly.

### Procedure

1. Do one of the following to instruct the tool to automatically convert between the two conventions for the bus name delimiter:
   - Define the environment variable MGC_CALIBRE_MAP_BUS_NAME in the shell environment.
   - Set MGC_CALIBRE_MAP_BUS_NAME to a non-nil value in the SKILL environment.
2. Start Calibre Interactive and Calibre RVE.

### Results

If MGC_CALIBRE_MAP_BUS_NAME is set, Calibre Interactive and Calibre RVE have the following behavior:

- Calibre Interactive Behavior



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

If you select a net in an attached schematic to populate a GUI field, the bus name delimiter is converted from <> to [].

For instance, in Calibre Interactive PEX you can enter nets to be excluded by selecting nets in the attached Cadence schematic view. If the net is named IN<2> in the schematic view, it will appear as IN[2] in the Calibre Interactive field if MGC_CALIBRE_MAP_BUS_NAME is set.

- Calibre RVE Behavior

  If you are using Calibre RVE to examine an SVDB in which [] is used for the bus name delimiter, it will automatically convert to <> for the bus name delimiter when highlighting nets in an attached Cadence schematic view if MGC_CALIBRE_MAP_BUS_NAME is set.

## Related Topics

Including/Excluding Nets



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Mapping Schematic Nets to Calibre View Nets for Simulation and Plotting

A typical step after parasitic extraction is the simulation of an extracted Calibre View in the Cadence Analog Design Environment. The method you use depends on whether you have multiple extracted Calibre Views or just one.

**Mapping Schematic Nets to Calibre View Nets with a Single Extracted View for Simulation and Plotting** . . . . . 810

**Mapping Schematic Nets to Calibre View Nets with Multiple Extracted Views for Simulation and Plotting** . . . . . 811

**Annotating Point to Point Resistance in Calibre View** . . . . . 812

## Mapping Schematic Nets to Calibre View Nets with a Single Extracted View for Simulation and Plotting

A typical step after parasitic extraction is the simulation of an extracted Calibre View in the Cadence Analog Design Environment. You can select nets to plot using the selection **Calibre > Map Schematic Net** from the Cadence Virtuoso Schematic Editor.

### Prerequisites

- A Calibre View; see "Creating a Post-Parasitic Extraction Calibre View" on page 782.
- Simulation in the Cadence environment is set up to use the extracted Calibre View. Consult the Cadence documentation for details.
- A single extracted CalibreView is used.

### Procedure

1. Open your schematic in Cadence Composer.
2. Choose **Calibre > Map Schematic Net** to open the Map Schematic Nets to CalibreView Nets dialog box.
3. If necessary, use the **Clear Selection** button to clear the net selection.
4. In the CalibreView Selection area, select the library, cell, and view name for the extracted Calibre View.
5. Click the **Select Schematic Net** button, then click in a net in the Composer schematic.

   The following changes occur in the Map Schematic Nets to CalibreView Nets dialog box:

   - The schematic net name is entered into the "Schematic Net" field at the top of the dialog box.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- The CalibreView net name is entered in the “CalibreView Net field.
- Values are displayed in the “Net R,C Totals” field.

If the CalibreView net name is not displayed and no values appear in “Net R,C Totals,” make sure the entries for library, cell, and view are correct.

6. (Optional) If you selected a bus, the “CalibreView Net” field includes a dropdown list in which you can select a specific net in the bus.
7. Click the **Select** button next to “Plot Selected Nets” and select an item from the dropdown list. This sends the net to the Outputs pane in the Cadence Analog Design Environment. Actions are logged to the CIW.
8. If desired, click the **Select** button next to “Calculator Expression” and select an expression. This opens the calculator with the selected net. For some expressions, you may need to select a pin in the Net Pins area. Actions are logged to the CIW.
9. Click **Apply** and continue with simulation.

### Related Topics

Mapping Schematic Nets to Calibre View Nets with Multiple Extracted Views for Simulation and Plotting

## Mapping Schematic Nets to Calibre View Nets with Multiple Extracted Views for Simulation and Plotting

A typical step after parasitic extraction is the simulation of an extracted Calibre View in the Cadence Analog Design Environment (ADE). After you select nets to plot in the Cadence Analog Design Environment, you can convert between CalibreView names and schematic names with the selections **Calibre > Map Schematic Nets to CalibreView Nets** and **Calibre > Map CalibreView Nets to Schematic Nets**. The text file *extviews.txt* provides the mapping between the extracted views and the cells.

### Prerequisites

- A Calibre View; see “Creating a Post-Parasitic Extraction Calibre View” on page 782.
- Simulation in the Cadence environment is set up to use the extracted Calibre View. Consult the Cadence documentation for details.
- The text file *extviews.txt* must exist in the directory containing the CalibreView netlist. Each line in *extviews.txt* should have one of the following formats:

```
extracted_viewName cellName
library extracted_viewName cellName
```

  Use the second format when the extracted Calibre View is not in the same library as the schematic, and specify the library name for the Calibre View.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For multiple extracted views within the same design, mapping is done by specifying all view names for every cell in the *extviews.txt* file. Calibre View searches within these views and performs the mapping. When mapping is done, hierarchical nets such as A1⁄IO⁄net2 are renamed to A1_IO_net2 in the CalibreView. If no extracted view is found, the net name does not change.

For example:

```
calibreRCC     buffer2
ref_calibreRCC buffer2
libCV calview2 buf3
```

### Procedure

1. Open your schematic in Cadence Composer.
2. Launch Cadence Virtuoso ADE.
3. Populate the Outputs table in Cadence Virtuoso ADE with the nets to plot.
4. Switch between CalibreView and schematic net naming with the following steps:
   a. Choose **Calibre > Map Schematic Nets to CalibreView Nets** or **Calibre > Map CalibreView Nets to Schematic Nets**.
   b. Click the **Netlist and Run** button.

   The net names in the Outputs table are converted as requested. Net names are not mapped until you click **Netlist and Run**.
5. Continue with the simulation.

### Results

Net names are converted between schematic and CalibreView naming conventions as requested, using the mapping information in the *extviews.txt* file.

For instance, suppose you are looking at cellA and descend into instance I1 which has net midA. When you select the net to plot it is entered in the Outputs table with schematic naming as I1⁄midA. After converting to CalibreView naming the net appears as I1_midA in the Outputs table.

### Related Topics

Mapping Schematic Nets to Calibre View Nets with a Single Extracted View for Simulation and Plotting

## Annotating Point to Point Resistance in Calibre View

When generating a Point to Point report to be viewed in the Map Schematic Nets to CalibreView Nets dialog box, you can include details on the resistances for specified nets.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

When enabled, information on each individual resistor is displayed, and highlighting it displays the resistor.

This feature can only be used if the netlist output format is CalibreView. To use this feature, perform one of the following steps:

- In the Calibre Interactive GUI, navigate to **Outputs** > **Reports** > **Point to Point** and enable the CalibreView checkbox.
- When using the PEX Report Point2Point statement, include the CALIBREVIEW keyword with SOURCENAMES.

The resistance information is annotated in the Net R,C Totals field in the Map Schematic Nets to CalibreView Nets dialog box.

When the resistance value is missing for a selected net, the following warning is printed:

```
WARNING: No Point2Point value is found for "net" net.
```

# Backannotation of Parasitic Values to the Schematic in Cadence Virtuoso

When debugging parasitics in the design it is sometimes useful to backannotate the schematic with the extracted parasitic values. This can be done using the dialog box "Map Schematic Nets to CalibreView Nets". Backannotation can also be deleted.

## Prerequisites

- A Calibre View; see "Creating a Post-Parasitic Extraction Calibre View" on page 782.

## Procedure

1. Open your schematic in Cadence Composer.
2. (Optional) If you are mapping hierarchical nets, specify the extracted view name to search in when doing the mapping. This step automates the handling of hierarchical nets and is useful if many nets are being mapped.

   Create a text file *extviews.txt* in the directory containing the CalibreView netlist. Each line in *extviews.txt* should have one of the following formats:

   ```
   extracted_viewName cellName
   library extracted_viewName cellName
   ```

   Use the second format when the extracted Calibre View is not in the same library as the schematic, and specify the library name for the Calibre View.

   For multiple extracted views within the same design, mapping is done by specifying all view names for every cell in the *extviews.txt* file. Calibre View searches within these views and performs the mapping. When mapping is done, hierarchical nets such as



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

A1/IO/net2 are renamed to A1_IO_net2 in the CalibreView. If no extracted view is found, the net name does not change.

3. Choose **Calibre** > **Map Schematic Nets to CalibreView Nets** to open the Map Schematic Nets to CalibreView Nets dialog box.
4. In the Calibreview Selection area, select the library, cell, and view name for the extracted Calibre View.
5. Enable the checkbox “Show/Hide Annotation for Net selected in Schematic.”
6. In the Composer schematic, click each net that you want to backannotate. If necessary, use the **Clear Selection** button to clear the net selection.

   The total resistance, lumped capacitance, and total cross-coupled capacitance values are annotated into the schematic.

   If you click in a net with existing annotation, the annotation is deleted.
7. Uncheck the “Show/Hide Annotation for Net selected in Schematic” checkbox when done annotating.

# Automating Calibre Interactive Runsets in the Cadence Design Environment

Environment variables can be set by custom scripts, or with Calibre Interactive pre-execution triggers for use within the Cadence Design Environment. The following script shows how Calibre Interactive execution triggers can be used within the Cadence Design Environment to automatically specify a Calibre Interactive runset based on the technology library attached to the design.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

For example:

```
procedure(mgc_start_calibre_trigger(drc_or_lvs win)
prog( (
   topCell
   designlibId
   techlibId
   techlibName
)

   printf( "Starting %s on top cell in window %L...\n"
      upperCase(drc_or_lvs)
      win )
   topCell=win->topCellView
   designlibId=topCell->lib
   techlibId=techGetTechFile(designlibId)
   techlibName=techlibId->libName

case( upperCase(drc_or_lvs)
         ("DRC"
            if( techlibName=="tech1"
               then
                  setShellEnvVar(
                     "MGC_CALIBRE_DRC_RUNSET_FILE=tech1.drc.runset")
                  println( "Calibre Interactive runset tech1.drc.runset")
               else
                  csh("unsetenv MGC_CALIBRE_DRC_RUNSET_FILE")
                  println( "No runset file set for this technology..." )
               )
         )

         ("LVS"
            if( techlibName=="tech1"
               then
                  setShellEnvVar(
                     "MGC_CALIBRE_LVS_RUNSET_FILE=tech1.lvs.runset")
                  println( "Calibre Interactive runset tech1.lvs.runset")
               else
                  csh("unsetenv MGC_CALIBRE_LVS_RUNSET_FILE")
                  println( "No runset file set for this technology..." )
               )
         )
         ("PEX"
            if( techlibName=="tech1"
               then
                  setShellEnvVar(
                     "MGC_CALIBRE_PEX_RUNSET_FILE=tech1.pex.runset")
                  println( "Calibre Interactive runset tech1.pex.runset")
               else
                  csh("unsetenv MGC_CALIBRE_PEX_RUNSET_FILE")
                  println( "No runset file set for this technology..." )
               )
         )
   )
return(t)
))
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Cadence Composer

The Cadence Composer integration is similar to that for Cadence Virtuoso.

See the section "Cadence Virtuoso" on page 725 for general information about the Calibre Interface to Cadence tools.

In Cadence Composer, after selecting a net or multiple nets in the schematic viewer for entry into a Calibre Interactive or Calibre RVE GUI field, you must enter Ctrl-C on the keyboard to end the net selection process and enter the selected nets.

See "Map Cadence Netlist Names to Schematic Names for Highlighting from Calibre RVE" in the *Calibre RVE User's Manual* for specific information about highlighting schematic elements.

**Note**

For Calibre 2012.3 and earlier releases, use the Esc key to end the net selection process. For Calibre 2012.4 and later releases, use Ctrl-C to end the net selection process.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Cadence Encounter

Calibre includes software that enables seamless integration of Calibre Interactive and Calibre RVE within the Cadence Encounter platform. A **Calibre** menu is activated when the supplied software is loaded into the Encounter environment, usually through the cal_enc.tcl file.

Calibre Interactive and Calibre RVE supports Cadence Encounter up to version 14.1.

You can add custom menu items to the Calibre integration menu; see "Custom Menu Items in Design Tools" on page 709.

**Creating an Interface to Cadence Encounter** . . . . . 817
**Importing saveNetlist Parameters for Verilog Export in Cadence Encounter** . . . . . 818

## Creating an Interface to Cadence Encounter

The following script can be inserted into the enc.tcl file to load the Calibre software:

```
proc load_calibre {} {
   global env
   if {![info exists env(CALIBRE_HOME)] || $env(CALIBRE_HOME)==""} {
      if {![info exists env(MGC_HOME)] || $env(MGC_HOME)==""} {
         puts "Environment variable CALIBRE_HOME not set."
         puts "Calibre interface NOT loaded."
         return
      } else {
         puts "Environment variable CALIBRE_HOME not set, using MGC_HOME."
         set env(CALIBRE_HOME) $env(MGC_HOME)
      }
   }

   set etclf [file join $env(CALIBRE_HOME) lib cal_enc.tcl]
   if {![file readable $etclf]} {
      puts "Could not find Calibre initialization files."
      puts "Calibre interface not loaded."
      return
   }
   if {[catch {source $etclf} msg]} {
      puts "ERROR while loading Calibre interface: $msg"
   }
}
load_calibre
```

This script first checks to see if the CALIBRE_HOME environment variable is set. If so, it loads the cal_enc.tcl file located in the lib subdirectory of the CALIBRE_HOME directory. The rest of the files accessed by cal_enc.tcl are located in the $CALIBRE_HOME/shared/pkgs/icv/tools/queryenc subdirectory. Collectively, these files are referred to as the Calibre Encounter Interface. The Calibre Encounter Interface adds a **Calibre** menu to the Encounter application's banner menu.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

You can also use MGC_HOME instead of CALIBRE_HOME to define the location of your software tree; see "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide*.

The following menu items are available in the **Calibre** menu:

**Table A-14. Calibre Menu in Cadence Encounter**

| Menu Item | Description |
|---|---|
| Run nmDRC | Starts Calibre Interactive DRC on the current cell |
| Run DFM | Starts Calibre Interactive DFM on the current cell |
| Run nmLVS | Starts Calibre Interactive LVS on the current cell |
| Run PERC | Starts Calibre Interactive PERC on the current cell |
| Run PEX | Starts Calibre Interactive PEX on the current cell |
| Start RVE | Starts Calibre RVE |
| Clear Highlights | Clears Calibre RVE highlights in the current cell |
| Setup | Open a dialog box to set parameters for the following:<br>• GDS Export<br>• DEF Export<br>• Verilog Export (also see "Importing saveNetlist Parameters for Verilog Export in Cadence Encounter" on page 818)<br>• RVE<br>• Socket |
| About | Displays information about the Calibre Encounter Interface |

For general information on the socket communication between a layout viewer and Calibre Interactive and Calibre RVE, see "Setting the Socket Port in Calibre Interactive" on page 707.

# Importing saveNetlist Parameters for Verilog Export in Cadence Encounter

You can populate the **Calibre** > **Setup** > **Verilog Export** dialog box with the parameters used by the saveNetlist command in Cadence Encounter.

## Procedure

1. Create a text file that contains the desired saveNetlist parameters. For example:

```
% cat exportParamsFile
-excludeLeafCell -flat -includeBumpCell
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

2. Define the environment variable MGC_CALIBRE_VERILOG_EXPORT_PARAM_FILE to point to the location of the file created in Step 1. For example:

   ```
   setenv MGC_CALIBRE_VERILOG_EXPORT_PARAM_FILE exportParamsFile
   ```

3. Start Cadence Encounter.

   The saveNetlist parameters specified in the file pointed to by MGC_CALIBRE_VERILOG_EXPORT_PARAM_FILE are used to populate the **Calibre** > **Setup** > **Verilog Export** dialog box and are used when the saveNetlist command is invoked.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Cadence Innovus

Calibre Interactive and Calibre RVE are integrated with the Cadence® Innovus™ Implementation System. The Calibre menu is added when the integration code is loaded.

The integration is supported for Cadence Innovus versions 16.22, 16.23, and 16.24.

**Creating an Interface to Cadence Innovus** . . . . . 820

# Creating an Interface to Cadence Innovus

Integration code must be loaded to enable the Calibre integration to Cadence Innovus. The integration is supported for Calibre RVE and all Calibre Interactive applications except Calibre Interactive 3DSTACK.

The integration is supported for Cadence Innovus versions 16.22, 16.23, 16.24, and 17.11.

The Tcl integration script is found at the following location:

```
$CALIBRE_HOME/lib/cal_enc.tcl
```

## Procedure

Load the integration script with one of the following methods:

- At the Cadence Innovus command prompt

  Invoke Cadence Innovus as usual, then source the integration script at the command prompt. For example:

  **innovus 1> source $::env(CALIBRE_HOME)/lib/cal_enc.tcl**

- At invocation with the -files argument

  You can load the integration script using the -files command line option when invoking Cadence Innovus. For example, at the shell prompt:

  **innovus -files "$CALIBRE_HOME/lib/cal_enc.tcl enc_setup.tcl" -win -log innovus.log**

- With automatic loading

  Consult the Cadence documentation regarding automatically loaded initialization files such as *enc.tcl* or *innovus.tcl*. You can add the Tcl integration code to such a file, or create a symbolic link to *$CALIBRE_HOME/lib/cal_enc.tcl*.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Results

Cadence Innovus is invoked and the Calibre menu is added with the following items:

**Table A-15. Calibre Menu in Cadence Innovus**

| Menu Item | Description |
|---|---|
| Run nmDRC | Starts Calibre Interactive nmDRC on the current cell |
| Run DFM | Starts Calibre Interactive DFM on the current cell |
| Run nmLVS | Starts Calibre Interactive nmLVS on the current cell |
| Run PERC | Starts Calibre Interactive PERC on the current cell |
| Run PEX | Starts Calibre Interactive PEX on the current cell |
| Run xACT | Starts Calibre Interactive xACT on the current cell |
| Start RVE | Starts Calibre RVE |
| Clear Highlights | Clears Calibre RVE highlights in the current cell |
| Setup | Opens a dialog box to set parameters for the following:<br>• GDS Export<br>• DEF Export<br>• Verilog Export (also see "Importing saveNetlist Parameters for Verilog Export in Cadence Encounter" on page 818)<br>• RVE<br>• Socket |
| RealTime DRC | • **Run DRC** — Starts a DRC run using Calibre RealTime Digital.<br>• **Options** — Opens the Calibre RealTime Options dialog box.<br>See "Calibre RealTime Digital in Cadence Innovus" in the *Calibre RealTime Digital User's Manual*. |
| About | Displays the document "About the Calibre Encounter/Innovus Interface." The interface to Calibre is the same for Cadence Encounter and Cadence Innovus. |

For general information on the socket communication between a layout viewer and Calibre Interactive and Calibre RVE, see "Setting the Socket Port in Calibre Interactive" on page 707.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Synopsys IC Compiler and Synopsys IC Compiler II

Calibre provides an integration of Calibre Interactive and Calibre RVE to Synopsys IC Compiler and Synopsys IC Compiler II. The integration makes it easy to run Calibre flows in the Synopsys IC Compiler environment and debug the results.

Code shipped in the Calibre software tree enables the integration, which adds a **Calibre** menu to the ICC and ICC2 GUI (see Figure A-3).

**Supported version** — The Calibre—IC Compiler interface supports Synopsys IC Compiler versions through 2014.9 and Synopsys IC Compiler II version M-2016.12-SP2.

You can do the following using the Calibre interface to Synopsys IC Compiler:

- Run Calibre Interactive nmDRC, nmLVS, PERC, PEX, and DFM:
  - Use any Calibre rule file.
  - Adjust rule file settings using the Calibre Interactive GUI.
  - Select DRC checks using the Calibre Interactive GUI.
  - Run Calibre LFD, FastXOR, Calibre PERC LDL, dummy fill, Calibre Pattern Matching, Calibre DP, and other applications.
- View and debug Calibre results using Calibre RVE (Results Viewing Environment):
  - View DRC, LVS, PERC, PEX, and DFM results.
  - Highlight results in layout and schematic.
  - Group, sort, and filter results.
  - Perform interactive short isolation using Calibre RVE for LVS.
- Add custom menu items to the **Calibre** menu in Synopsys IC Compiler. See "Custom Menu Items in Design Tools" on page 709.
- Convert results from an ASCII format Calibre results database (RDB) to error markers and route guides in Synopsys IC Compiler. This allows you to use familiar tools when debugging errors and correcting the design. See "Calibre RDB Conversion: Working with Calibre Results in Place and Route Tools" in the *Calibre RVE User's Manual*.

**Calibre Flow with Synopsys IC Compiler** . . . . . 823
**Calibre Interface to Synopsys IC Compiler** . . . . . 824
**Running Calibre Interactive with Synopsys IC Compiler** . . . . . 828
**Setting Socket Connections with Synopsys IC Compiler and IC Compiler II** . . . . . 831
**Troubleshooting the Calibre Interface with Synopsys IC Compiler** . . . . . 832



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Flow with Synopsys IC Compiler

The interactive flow with Calibre Interactive, Calibre RVE, and Synopsys IC Compiler allows you to easily run Calibre and debug results. The flow with Calibre nmDRC is illustrated.

**Figure A-3. Interactive Calibre Flow with Synopsys IC Compiler**

Although the figure shows Calibre nmDRC, the Calibre nmLVS, Calibre PERC, DFM, and PEX applications are also supported with Synopsys IC Compiler.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Calibre Interface to Synopsys IC Compiler

The Calibre interface to Synopsys IC Compiler supports the Calibre nmDRC, Calibre nmLVS, Calibre PERC, PEX, and DFM applications for both Calibre Interactive and Calibre RVE. The **Calibre** menu is available after the interface is loaded. Synopsys IC Compiler II is also supported.

**Loading the Calibre Interface to Synopsys IC Compiler** . . . . . 824
**Loading the Calibre Interface to Synopsys IC Compiler II** . . . . . 825
**README File for the Calibre-IC Compiler Integration** . . . . . 827
**Calibre Menu in Synopsys IC Compiler** . . . . . 827

## Loading the Calibre Interface to Synopsys IC Compiler

The Calibre interface can be loaded automatically each time you start Synopsys IC Compiler, or you can manually install the interface. The Calibre—IC Compiler interface supports Synopsys IC Compiler versions through 2014.9. Synopsys IC Compiler II version M-2016.12-SP2 is supported.

### Prerequisites

- The environment variable CALIBRE_HOME or MGC_HOME points to the Calibre software tree. See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide*.

### Procedure

1. Copy the following code into a file *load_calibre_interface.tcl*:

```
if {![info exists ::env(CALIBRE_HOME)] && [info exists ::env(MGC_HOME)]} {
   puts "Environment variable CALIBRE_HOME not set, using MGC_HOME"
   set ::env(CALIBRE_HOME) $::env(MGC_HOME)
}
if {![info exists ::env(CALIBRE_HOME)]} {
   puts "ERROR, environment variable CALIBRE_HOME or MGC_HOME not set"
   puts "Calibre interface not loaded"
} else {
   source [file join $::env(CALIBRE_HOME) lib icc_calibre.tcl]
}
```

2. Use one of the following methods to install the interface:

   - **Installation with a startup script and -f command-line option**

     i. Source the installation code from a startup command script which is specified with the -f command-line option. For example, place this line in your startup command script *startup_cmd.tcl*:

```
source load_calibre_interface.tcl
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

where *load_calibre_interface.tcl* is the file created in Step 1. Alternatively, you can copy the installation code directly into the startup script.

ii. Start Synopsys IC Compiler as follows:

```
% $SYNOPSYS/bin/icc_shell -gui -f startup_cmd.tcl
```

- **Automatic installation with *.synopsys_icc_gui.tcl* file**

  Source the installation code in your *$HOME/.synopsys_icc_gui.tcl* file. The interface is loaded each time you start Synopsys IC Compiler in gui mode.

- **Manual Installation at the icc_shell command prompt**

  Type the following command at the command prompt for Synopsys IC Compiler:

```
icc_shell> source load_calibre_interface.tcl
```

### Results

When the Calibre interface to Synopsys IC Compiler is successfully loaded, the **Calibre** menu is added to the ICC menu bar. The **Calibre** menu is described in the section "Calibre Menu in Synopsys IC Compiler" on page 827.

In addition, a banner similar to the one below is displayed:

```
//
//             Calibre IC Compiler Interface * (v2014.3_16.15) *
//
//                        Copyright Siemens 1996 - 2020
//                             All Rights Reserved.
//         THIS WORK CONTAINS TRADE SECRET AND PROPRIETARY INFORMATION
//            WHICH IS THE PROPERTY OF MENTOR GRAPHICS CORPORATION
//              OR ITS LICENSORS AND IS SUBJECT TO LICENSE TERMS.
//
```

## Loading the Calibre Interface to Synopsys IC Compiler II

The Calibre interface can be loaded automatically each time you start Synopsys IC Compiler II, or you can manually install the interface. Synopsys IC Compiler II version M-2016.12-SP2 is supported.

### Prerequisites

- The environment variable CALIBRE_HOME or MGC_HOME points to the Calibre software tree. See "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide*.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Do one of the following, depending on whether you want automatic installation:
  - **For automatic installation with ~/*.synopsys_icc2.setup* file:**

    Copy the following code into the file *$HOME/.synopsys_icc2.setup*:

```
if {![info exists ::env(CALIBRE_HOME)] && [info exists ::env(MGC_HOME)]} {
      puts "Environment variable CALIBRE_HOME not set, using MGC_HOME"
      set ::env(CALIBRE_HOME) $::env(MGC_HOME)
}
if {![info exists ::env(CALIBRE_HOME)]} {
      puts "ERROR, environment variable CALIBRE_HOME or MGC_HOME not set"
      puts "Calibre interface not loaded"
} else {
      set_app_options -name gui.custom_setup_files \
         -value [file join $::env(CALIBRE_HOME) lib icc_calibre.tcl]
}
```

  - **For non-automatic installation:**

    Copy the following code into the file *load_calibre_interface.tcl*:

```
if {![info exists ::env(CALIBRE_HOME)] && [info exists ::env(MGC_HOME)]} {
   puts "Environment variable CALIBRE_HOME not set, using MGC_HOME"
   set ::env(CALIBRE_HOME) $::env(MGC_HOME)
}
if {![info exists ::env(CALIBRE_HOME)]} {
   puts "ERROR, environment variable CALIBRE_HOME or MGC_HOME not set"
   puts "Calibre interface not loaded"
} else {
   source [file join $::env(CALIBRE_HOME) lib icc_calibre.tcl]
}
```

## Procedure

Use one of the following methods to install the interface:

- **Automatic installation with *$HOME/.synopsys_icc2.setup* file**

  The interface is automatically loaded each time you start Synopsys IC Compiler II in gui mode. For example:

  **% $SYNOPSYS/bin/icc2_shell -gui <*options*>**

- **Installation with a startup script and the -f command-line option**

  i. Source the installation code from a startup command script which is specified with the -f command-line option. For example, place this line in your startup command script *startup_cmd.tcl*:

```
source load_calibre_interface.tcl
```

  where *load_calibre_interface.tcl* is the file created in the Prerequisites section. Alternatively, you can copy the installation code directly into the startup script.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

ii. Start Synopsys IC Compiler II as follows:

**% $SYNOPSYS/bin/icc2_shell -gui -f startup_cmd.tcl**

- **Manual installation at the icc_shell command prompt**

i. Start Synopsys IC Compiler II:

**% $SYNOPSYS/bin/icc2_shell -gui <*options*>**

ii. Type the following at the command prompt for Synopsys IC Compiler II:

**icc_shell> source load_calibre_interface.tcl**

### Results

When the Calibre interface to Synopsys IC Compiler II is successfully loaded, the **Calibre** menu is added to the menu bar. The **Calibre** menu is described in the section "Calibre Menu in Synopsys IC Compiler" on page 827.

In addition, a banner similar to the one below is displayed:

```
//
//             Calibre IC Compiler Interface * (v2019.3) *
//
//                        Copyright Siemens 1996 - 2020
//                             All Rights Reserved.
//         THIS WORK CONTAINS TRADE SECRET AND PROPRIETARY INFORMATION
//            WHICH IS THE PROPERTY OF MENTOR GRAPHICS CORPORATION
//              OR ITS LICENSORS AND IS SUBJECT TO LICENSE TERMS.
//
```

## README File for the Calibre-IC Compiler Integration

The README file for the Calibre -IC Compiler integration is at the following location:

*$CALIBRE_HOME/shared/pkgs/icv/tools/querytcl/ICCompilerREADME*

The file includes these instructions for installing the **Calibre** menu in Synopsys IC Compiler and further details about the Calibre—IC Compiler Interface.

## Calibre Menu in Synopsys IC Compiler

The **Calibre** menu is available after Calibre interface is loaded. The menu has selections to start Calibre Interactive, Calibre RVE, and to selection setup options.

The **Calibre** menu in Synopsys IC Compiler has the selections shown in Figure A-4. See "Loading the Calibre Interface to Synopsys IC Compiler" on page 824 if you do not see the **Calibre** menu in the ICC GUI.

You can also add custom menu items to the Calibre integration menu; see "Custom Menu Items in Design Tools" on page 709.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Figure A-4. Calibre Menu Contents in Synopsys IC Compiler**

# Running Calibre Interactive with Synopsys IC Compiler

All Calibre Interactive applications are supported: Calibre nmDRC, Calibre nmLVS, DFM, Calibre PERC, and PEX.

You can use Calibre Interactive to easily change options for your Calibre run, since Calibre Interactive settings override settings in the rule file. You save all your Calibre Interactive settings in a runset for later use.

## Prerequisites

- "Loading the Calibre Interface to Synopsys IC Compiler" on page 824 has been completed.

## Procedure

1. Launch Synopsys IC Compiler:

   ```
   icc_shell -gui
   ```

   You should see the Calibre banner displayed in the ICC command window; if not, see "Loading the Calibre Interface to Synopsys IC Compiler" on page 824.

2. Open your library



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

```
icc_shell> open_mw_lib my_milkyway_lib
icc_shell> open_mw_cel top_cell
```

3. Choose **Calibre** > **Run nmDRC**, or the desired Calibre Interactive application.

**Figure A-5. Calibre Menu in Synopsys IC Compiler**

Calibre Interactive opens and displays the Inputs pane with the current cell selected.

**Note**

Calibre Pattern Matching, Calibre DP, and dummy fill are run using the **Run DRC** selection. Calibre SmartFill is run using the **Run DFM** selection.

4. (Optional) Load a runset with **File** > **Load Runset**; a runset includes all Calibre Interactive settings. See "About Calibre Interactive Runsets" on page 33 for information.

   If your runset includes all the desired options, you may be ready to start a run; if so, skip to Step 13.

5. Specify the input source as "Layout Export" and the desired format of the exported design.

   These steps instruct Synopsys IC Compiler to export the open design to Calibre. Synopsys IC Compiler controls the conversion to the output format.

   a. Set "Format" to GDSII, OASIS, or LEFDEF.

      See "Export DEF Layout from Design Tool" on page 89 for additional details if exporting DEF format.

   b. Enable "Export from layout viewer."

   c. Verify the filename for the exported layout database in the "File" field.

      An intermediate file is extracted and saved to the file named in the "File" field during layout export. This filename is generated automatically based on the setting



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

on the **Templates** > **Inputs** sub-tab of the Setup Preferences dialog box (**Setup** > **Preferences**); see "Templates for File Naming" on page 148.

d. Set streamout options in Synopsys IC Compiler. Consult the Synopsys IC Compiler documentation, or view **Calibre** > **Layout Export** and **Calibre** > **Netlist Export** in Synopsys IC Compiler for information.

6. View the Rules and Outputs panes and fill in the required settings.

7. (Optional) You can select the checks that are run in Calibre Interactive nmDRC, DFM, and Calibre nmLVS (for ERC checks). You can select checks by groups, by layer, and other criteria.

   See the following topics for more information:

   - "Specifying the Checks to Execute in a Calibre Interactive nmDRC Run" on page 165
   - "Specifying the Checks to Execute in a Calibre Interactive DFM Run" on page 218
   - "Selecting ERC Checks" on page 284

8. (Optional) Calibre Interactive supports trigger functions which can execute at several points in your flow. Trigger functions are useful for customizing your flow and doing pre- and post-processing of design input and result output.

   Choose **Setup** > **Preferences**, then click the **Triggers** tab.

   See the following topics for information:

   - "Trigger Functions in Calibre Interactive" on page 657 — Complete information.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- "External Trigger Definitions in Synopsys IC Compiler" on page 677 — External triggers, which execute before and after Calibre Interactive execution.
- "Internal Triggers in Calibre Interactive" on page 659 — Internal triggers, which execute before and after the Calibre run.

9. Click **Run Control** on the left panel and review the settings. Performance on large designs can be improved by enabling the multithreaded (MT) and hyperscaling processing modes. See "Run Control Setup and Licensing" on page 681 for more information.
10. (Optional) You can run Calibre on a selected region of the design when using Calibre Interactive nmDRC and DFM.

    See "Performing Area DRC" on page 189 for instructions.
11. (Optional) View the control file for the Calibre Interactive run by clicking Ctrl-**Run DRC**. The control file contains the settings that control the Calibre run; control file settings take precedence over rule file settings if there is a conflict. See "About Control Files" on page 46 for more information.
12. Choose **File** > **Save Runset** to save your Calibre Interactive settings; see "About Calibre Interactive Runsets" on page 33 for information.
13. Click **Run DRC** (or your selected application) on the left panel to start the Calibre run.

    Calibre Interactive switches the view to the Transcript pane, where you can view the progress of the Calibre run and any run-time errors and warnings. See "Transcript Pane in Calibre Interactive" on page 99 for more information.
14. If Calibre RVE did not open automatically, choose **Calibre** > **Start RVE** in Synopsys IC Compiler to view the Calibre results. See "Calibre RVE with Synopsys IC Compiler" in the *Calibre RVE User's Manual* for more information.

### Related Topics

Troubleshooting the Calibre Interface with Synopsys IC Compiler

## Setting Socket Connections with Synopsys IC Compiler and IC Compiler II

Calibre Interactive and Calibre RVE communicate with Synopsys IC Compiler through a socket port. Socket communication is established automatically when you start Calibre Interactive or Calibre RVE from the **Calibre** menu in Synopsys IC Compiler. By default, Calibre Interactive and Calibre RVE use socket port 9189 for communication, and the tool automatically searches for an available socket if it cannot obtain the default socket.

Synopsys IC Compiler II is also supported.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

You can also manually configure the socket, as described below, although this is not needed in most situations.

### Prerequisites

- "Loading the Calibre Interface to Synopsys IC Compiler" on page 824

### Procedure

Do one of the following:

| To set the socket using … | Do this … |
|---|---|
| Synopsys IC Compiler | Choose **Calibre** > **Setup** > **Socket** in Synopsys IC Compiler. |
| Environment variable | Set the environment variable MGC_CALIBRE_LAYOUT_SERVER, as described in "Setting the Calibre Interactive Socket Port for a Layout Viewer" on page 707. |

### Results

The socket is initialized when needed.

**Tip**

You can also specify that the socket is started when Synopsys IC Compiler starts. Set the environment variable MGC_RVE_INIT_SOCKET_AT_STARTUP to any value; when the Calibre integration code is read into Synopsys IC Compiler, the server socket will be started.

## Troubleshooting the Calibre Interface with Synopsys IC Compiler

Several common problem may occur in the Calibre Interface with Synopsys IC Compiler.

The following issues are discussed:

- Problem: Some rule file settings are not used in the Calibre Interactive run
- Tip: Debugging Calibre Interactive operation

### Problem: Some rule file settings are not used in the Calibre Interactive run

Cause: Calibre Interactive settings override rule file settings.

Solution: Do one of the following, depending on your situation.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Enable the desired setting in the GUI. You can save the GUI settings to a runset with **File > Save Runset**; then load the runset later with **File > Load Runset**. Also see "Loading Runsets on Invocation" on page 39.
- If you want to load all rule file settings into the GUI, click **Load** after specifying the rule file on the Rules pane. Any GUI changes that you make after loading the rule file override the corresponding setting in the rule file. Also see "Automatically Loading a Rule File" on page 76.

**Tip**

You can view the control file for the Calibre Interactive run by holding the Ctrl key while clicking **Run *app*** where ***app*** is the Calibre Interactive application you are running. The control file contains the settings that control the Calibre run; control file settings take precedence over rule file settings if there is a conflict. See "About Control Files" on page 46 and "How Rule Files Relate to Control Files" on page 53 for more information.

## Tip: Debugging Calibre Interactive operation

- Save the Calibre Interactive transcript by choosing **Transcript > Echo to File**. This makes it easier to compare the transcript output to other runs, or to the transcript from a command line run.
- View the control file for the Calibre Interactive run by holding the Ctrl key while clicking **Run *app***. The control file contains the settings that control the Calibre run; control file settings take precedence over rule file settings if there is a conflict. See "About Control Files" on page 46 and "How Rule Files Relate to Control Files" on page 53 for more information.
- Perform a command line Calibre run and compare transcript and results output to that from the Calibre Interactive run.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Silvaco Expert

The Silvaco Expert design tool includes integrations to Calibre Interactive nmDRC, Calibre Interactive nmLVS, Calibre RVE, and a custom interface to launch a batch Calibre nmDRC job. You can use these interfaces to launch a Calibre verification run from Silvaco Expert, then use Calibre RVE to highlight the errors in the open design.

Also see "Silvaco" on the Calibre Interfaces web page for additional information.

**Creating the Interface to Silvaco Expert** . . . . . 834
**Setting Socket Connections with Silvaco Expert** . . . . . 834
**Calibre Menu in Silvaco Expert** . . . . . 834

## Creating the Interface to Silvaco Expert

Silvaco Expert requires the MGC_HOME environment variable to be set and pointing to the location of the Calibre installation. It is good practice to also define CALIBRE_HOME in the same manner.

No additional setup is needed to enable the Calibre integration. The integration automatically adds the **Verification** > **Calibre** menu items to Silvaco Expert; these menu selections are used to start a Calibre job. See "Calibre Menu in Silvaco Expert" on page 834 for more information.

### Related Topics

Setting the CALIBRE_HOME Environment Variable [Calibre Administrator's Guide]

## Setting Socket Connections with Silvaco Expert

Calibre Interactive and Calibre RVE use certain default port numbers for socket communication with a connected design tool. By default, Calibre Interactive and Calibre RVE use socket port 9189 to communicate with a layout editor. The tool automatically searches for an available socket if it cannot obtain the default socket. When Silvaco Expert successfully opens a socket, it reports the socket number to the log file.

You can manually set the socket number with the MGC_CALIBRE_LAYOUT_SERVER environment variable. See "Setting the Calibre Interactive Socket Port for a Layout Viewer" on page 707 for additional information.

## Calibre Menu in Silvaco Expert

The **Calibre** menu in Silvaco Expert is at **Verification** > **Calibre**.

The following table describes the menu selections and provides basic instructions for running Calibre.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table A-16. Calibre Menu in Silvaco Expert**

| Menu Selection | Description |
|---|---|
| Verification > Calibre > Run DRC | Opens Calibre Interactive nmDRC.<br>You can enable "Show results in RVE" on the Outputs pane to have Calibre RVE open automatically when the run is finished, then highlight DRC results in Expert.<br>See "Using Calibre Interactive to Perform DRC" on page 161 and "Using Calibre RVE for DRC" in the *Calibre RVE User's Manual* for more information. |
| Verification > Calibre > Run LVS | Opens Calibre Interactive nmLVS.<br>You can enable "Start RVE after LVS finishes" on the Outputs pane to have Calibre RVE open automatically when the run is finished, then highlight errors in Expert.<br>See "Using Calibre Interactive to Perform LVS" on page 237 and "Using Calibre RVE for LVS" in the *Calibre RVE User's Manual* for more information. |
| Verification > Calibre > Batch DRC | Opens a dialog box to start a batch Calibre nmDRC run.<br>Fill in parameters, then click **OK** to start the Calibre run.<br>To inspect the errors after the run finishes, choose **Verification** > **DRC** > **Errors** > **Load Errors** in Expert. |

# Other Viewer Integrations

Some viewer integrations are not discussed here because the integration is straightforward and does not require any special setup actions or custom commands for operation.

The full list of design tools with a Calibre integration is given in "Integration to Design Tools" on page 26. If the design tool is listed but not discussed in detail in this appendix, then no special setup is needed.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Appendix B
# Environment Variables

Calibre Interactive and Calibre View generation read a number of environment variables that specify run-time behavior. Some environment variables apply to more than one application.

The *Calibre Administrator's Guide* lists general and application specific Calibre environment variables in the appendix "Calibre Environment Variables".

**Environment Variables for Calibre Interactive** . . . . . **837**
**Environment Variables for Calibre View** . . . . . **843**

## Environment Variables for Calibre Interactive

Several environment variables are available to control the behavior of Calibre Interactive.

The following table lists environment variables used in Calibre Interactive. The Type column indicates whether the environment variable is available in the shell environment (env) or in SKILL scripts.

**Table B-1. Calibre Interactive Environment Variables**

| Variable | Type | Description |
|---|---|---|
| CALIBRE_CI_REMOTE_COMMAND_USE_BASH | env | When set, use the Bash shell to launch the MTflex master script.<br>See "Configuring for Distributed (MTflex) Execution" on page 696 for this and the next two environment variables. |
| CALIBRE_CI_REMOTE_COMMAND_LSF_USE_BASH | env | When set, use the Bash shell to launch the MTflex master script for Spectrum LSF runs only. |
| CALIBRE_CI_REMOTE_COMMAND_GRID_USE_BASH | env | When set, use the Bash shell to launch the MTflex master script for Univa Grid Engine runs only. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table B-1. Calibre Interactive Environment Variables (cont.)**

| Variable | Type | Description |
|---|---|---|
| CALIBRE_HOME | env | Specifies the software tree invoked when Calibre Interactive or Calibre RVE is called from Pyxis (see "Using Calibre Interactive with Pyxis" on page 723).<br>Also sets the location of the software tree for Calibre tools; see "Setting the CALIBRE_HOME Environment Variable" in the *Calibre Administrator's Guide*. |
| CDS_NETLISTING_MODE | env or SKILL | Sets netlisting export mode in Cadence Virtuoso ("Netlist Export in Cadence Composer" on page 740). |
| MGC_CALGUI_RELEASE_LICENSE_TIME | env | Sets the time after which the Calibre Interactive license is released if there is no activity ("License Timeout in Calibre Interactive" on page 67). The default time unit is hours. |
| MGC_CALIBRE_ALLOW_VIEWER_TRIGGERS | env or SKILL | Specifies access level for triggers running in the design tool. See "Trigger Access to the Design Tool Environment" on page 665. |
| MGC_CALIBRE_CGIDB_DIR | env | Sets location of preference files ("Startup Preferences" on page 144). |
| MGC_CALIBRE_CUSTOMIZATION_FILE | env | Sets location of customization file; see "Customization Files" on page 587. |
| MGC_CALIBRE_DB_READ_OPTIONS | env | Sets database translation options for OpenAccess and LEF/DEF database. This environment variable must be set when using a Calibre Interactive control file to perform a command line Calibre run if the layout database format is OpenAccess or LEF/DEF. ("MGC_CALIBRE_DB_READ_OPTIONS Environment Variable for Database Options" on page 81) |
| MGC_CALIBRE_DFM_RUNSET_FILE | env | Sets the Calibre Interactive DFM runset ("Loading Runsets on Invocation" on page 39). |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table B-1. Calibre Interactive Environment Variables (cont.)**

| Variable | Type | Description |
|---|---|---|
| MGC_CALIBRE_DFM_RUNSET_LIST | env | Populates Calibre Interactive DFM runsets in the Load Runset File dialog box and opens the Load Runset dialog at startup (“Loading Runsets on Invocation” on page 39). |
| MGC_CALIBRE_DRC_RUNSET_FILE | env | Sets the Calibre Interactive DRC runset (“Loading Runsets on Invocation” on page 39). |
| MGC_CALIBRE_DRC_RUNSET_LIST | env | Populates Calibre Interactive DRC runsets in the Load Runset File dialog box and opens the Load Runset dialog at startup (“Loading Runsets on Invocation” on page 39). |
| MGC_CALIBRE_EXPORT_LAYOUT_TEMPLATE_FILE | SKILL | Sets layout template file location in Cadence Virtuoso (“Calibre Layout Export Setup Dialog Box in Cadence Virtuoso” on page 731). |
| MGC_CALIBRE_EXPORT_NETLIST_TEMPLATE_FILE | SKILL | Sets netlist template file location in Cadence Virtuoso (“Calibre Netlist Export Setup Dialog Box in Cadence Composer” on page 741). |
| MGC_CALIBRE_LAYOUT_SERVER | env | Specifies a hostname and communications socket port number for Calibre Interactive and Calibre RVE to use for communication with a connected layout viewer (“Setting the Calibre Interactive Socket Port for a Layout Viewer” on page 707). |
| MGC_CALIBRE_LAYOUT_SERVER_NAME | env | This environment variable is set automatically by the tool to the type of layout viewer. The allowed values are the same as those for the cmnLayoutServerType runset variable. |
| MGC_CALIBRE_LAYOUT_TMP_FILE | env | Sets the name of the intermediate file used in layout editor communication. |
| MGC_CALIBRE_LVS_RUNSET_FILE | env | Sets the Calibre Interactive LVS runset (“Loading Runsets on Invocation” on page 39). |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table B-1. Calibre Interactive Environment Variables (cont.)**

| Variable | Type | Description |
|---|---|---|
| MGC_CALIBRE_LVS_RUNSET_LIST | env | Populates Calibre Interactive LVS runsets in the Load Runset File dialog box and opens the Load Runset dialog at startup (“Loading Runsets on Invocation” on page 39). |
| MGC_CALIBRE_MAP_BUS_NAME | env or SKILL | See Automatically Convert Bus Name Delimiter for Calibre Interactive and Calibre RVE.<br>Note: this environment variable does not affect netlist export; see Convert Bus Name Delimiter During Schematic Netlist Export. |
| MGC_CALIBRE_MRU_SIZE | env | Sets the list length of the “most recently used” lists in Calibre Interactive, Calibre Interactive xACT, and Calibre RVE. The environment variable also controls the list length for the Recent Runsets and Recent Text Files menu selections.<br>The default list length is 20. The maximum length is 50. |
| MGC_CALIBRE_PERC_RUNSET_FILE | env | Sets the Calibre Interactive PERC runset (“Loading Runsets on Invocation” on page 39). |
| MGC_CALIBRE_PERC_RUNSET_LIST | env | Populates Calibre Interactive PERC runsets in the Load Runset File dialog box and opens the Load Runset dialog at startup (“Loading Runsets on Invocation” on page 39). |
| MGC_CALIBRE_PEX_RUNSET_FILE | env | Sets the Calibre Interactive PEX runset (“Loading Runsets on Invocation” on page 39). |
| MGC_CALIBRE_PEX_RUNSET_LIST | env | Populates Calibre Interactive PEX runsets in the Load Runset File dialog box and opens the Load Runset dialog at startup (“Loading Runsets on Invocation” on page 39). |
| MGC_CALIBRE_SAVE_ALL_RUNSET_VALUES | env | Controls whether you save both default and non-default data to a runset file (“Save All GUI Options to the Runset File” on page 44). |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table B-1. Calibre Interactive Environment Variables (cont.)**

| Variable | Type | Description |
|---|---|---|
| MGC_CALIBRE_SCHEMATIC_SERVER | env | Specifies a hostname and communications socket port number Calibre Interactive and Calibre RVE to use for communication with a connected schematic viewer (“Setting the Calibre Interactive Socket Port for a Schematic Viewer” on page 708). |
| MGC_CALIBRE_SCHEMATIC_SERVER_NAME | env | This environment variable is set automatically by the tool to the type of schematic viewer. The allowed values are the same as those for the cmnSchematicServerType runset variable. |
| MGC_CALIBRE_VERILOG_EXPORT_PARAM_FILE | env | Specifies the path to a file with saveNetlist parameters; see “Importing saveNetlist Parameters for Verilog Export in Cadence Encounter” on page 818. |
| MGC_CALIBRE_VIEWER_MENU_CMDS_SINGLE_TOOL | env | Specifies custom menu items to behave the same way as standard menu items when selected multiple times.<br>See “Creating Custom Menus in Design Tools” on page 710. |
| MGC_CALIBRE_XACT_RUNSET_FILE | env | Used in Calibre Interactive xACT to pre-populate the Recent Runsets and Recent Text Files lists in the File menu, and the Load Runset Files dialog box.<br>See “Populating the Load Runset Files Dialog Box and Recent Files Lists” on page 360. |
| MGC_CI_EXPORT_ENABLE_COLORING | env | When set, specifies to export coloring information during streamout from Cadence Virtuoso. The option “Enable coloring” is checked in the Calibre Layout Export Setup dialog box. By default, coloring information is not exported. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table B-1. Calibre Interactive Environment Variables (cont.)**

| Variable | Type | Description |
|---|---|---|
| MGC_FDI_OA_VERSION | env | Sets the Open Access version for FDI and DBdiff utilities. The allowed values are 22.43 and 22.50. The default is 22.43. The SVRF statement LAYOUT SYSTEM OA overrides this variable.<br>If you specify the OA version in the fdi2gds or fdi2oasis utilities, the environment variable is ignored.<br>Also see "Providing OpenAccess Input" on page 90 and "FDI Environment Variables" in the *Calibre Layout Comparison and Translation Guide*. |
| MGC_RVE_INIT_SOCKET_AT_STARTUP | env | Sets socket initiation preference for Cadence Virtuoso ("Calibre Setup Menu in Cadence Virtuoso" on page 728) and for Synopsys IC Compiler and IC Compiler II ("Setting Socket Connections with Synopsys IC Compiler and IC Compiler II" on page 831). Used for both Calibre Interactive and Calibre RVE. |
| MGC_RVE_RELEASE_LICENSE_TIME | env | Sets the time in hours after which the Calibre RVE license is released if there is no activity (License Timeout in Calibre RVE" in the *Calibre RVE User's Manual*).<br>If the environment variable MGC_CALGUI_RELEASE_LICENSE_TIME is not set, then MGC_RVE_RELEASE_LICENSE_TIME also sets the license timeout for Calibre Interactive. ("License Timeout in Calibre Interactive" on page 67). |
| MGCCALIBREMENUVIEWTYPE | SKILL | Specifies window type in which to load a **Calibre** menu ("Creating an Interface to Cadence Virtuoso" on page 725). |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table B-1. Calibre Interactive Environment Variables (cont.)**

| Variable | Type | Description |
|---|---|---|
| OA_HOME[1] | env | Calibre uses the OpenAccess library at $CALIBRE_HOME/shared/pkgs/icv_o. If additional plug-ins are needed, use the OA_HOME environment variable to specify the path to an OA library that contains the plug-ins. Note, that the library specified by the OA_HOME variable is not used. Calibre always uses the default OpenAccess library included with the Calibre installation. (See "Specifying Third Party Database Options in Calibre Interactive" on page 94)<br>Note: As of the 2013.3 release, the supported OpenAccess version in Calibre is oa22.43p006. Also see footnote. |
| PEX_FMT_HP_PORT_MAP_MODE | env | Specifies the use of templates in Calibre Interactive PEX. When this environment variable is defined, the templates directory in the SVDB is not deleted before a run. See "Templates" in the *Calibre xRC User's Manual*. |

1. If this variable is set, then the specified library must be version oa22.43p006 or later. For Linux platforms, libstdc++.so.6.0.8 must be available from the host (RHEL 4u6 and later have it by default) or through the LD_LIBRARY_PATH environment variable.

# Environment Variables for Calibre View

Several environment variables are available to control Calibre View generation.

The following table lists environment variables used in Calibre View generation. The Type column indicates whether the environment variable is available in the shell environment (env) or in SKILL scripts.

Also see "Global Variables for Calibre View Generation" on page 800.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table B-2. Calibre View Environment Variables**

| Variable | Type | Description |
|---|---|---|
| CALIBRE_HOME | env | Specifies the software tree invoked when Calibre Interactive or Calibre RVE is called from Pyxis (“Using Calibre Interactive with Pyxis” on page 723).<br>Also sets the location of the software tree for Calibre tools; see “Setting the CALIBRE_HOME Environment Variable” in the *Calibre Administrator’s Guide*. |
| CALIBRE_DISABLE_PPAR_WARNINGS | env | Set this environment variable to disable pPar evaluation warnings. |
| CALIBRE_FDIBA_ENABLE_NETEXPR_COPY | env | See “Handling of Nets With Inherited Connectivity” on page 753. |
| MGC_RVE_INIT_SOCKET_AT_STARTUP | env | Sets socket initiation preference for Cadence Virtuoso (“Calibre Setup Menu in Cadence Virtuoso” on page 728) and Synopsys IC Compiler (“Setting Socket Connections with Synopsys IC Compiler and IC Compiler II” on page 831). |
| MGC_RVE_RELEASE_LICENSE_TIME | env | Sets the time in hours after which the Calibre RVE license is released if there is no activity (License Timeout in Calibre RVE” in the *Calibre RVE User’s Manual*).<br>If the environment variable MGC_CALGUI_RELEASE_LICENSE_TIME is not set, then MGC_RVE_RELEASE_LICENSE_TIME also sets the license timeout for Calibre Interactive. (“License Timeout in Calibre Interactive” on page 67). |
| MGCCALIBREMENUVIEWTYPE | SKILL | Specifies window type in which to load a **Calibre** menu (“Creating an Interface to Cadence Virtuoso” on page 725). |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table B-2. Calibre View Environment Variables (cont.)**

| Variable | Type | Description |
|---|---|---|
| OA_HOME[1] | env | Sets OpenAccess library version to one other than the default. The OpenAccess default library that ships with Calibre is located at ./shared/pkgs/icv_o in your Calibre software tree. (“Specifying Third Party Database Options in Calibre Interactive” on page 94)<br>Note: As of the 2013.3 release, the supported OpenAccess version in Calibre is oa22.43p006. Also see footnote. |

1. If this variable is set, then the specified library must be version oa22.43p006 or later. For Linux platforms, libstdc++.so.6.0.8 must be available from the host (RHEL 4u6 and later have it by default) or through the LD_LIBRARY_PATH environment variable.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Appendix C
# Runset Options

A Calibre Interactive runset is a text file created by Calibre Interactive to store the settings you specify in the GUI. You can load a runset into Calibre Interactive to restore the GUI interface to the saved settings. This appendix gives the runset file format and option names.


**Calibre Interactive Runset File Format** . . . . . **847**

**Creating a Calibre Interactive Runset** . . . . . **848**

**Option Descriptions** . . . . . **849**

DRC Runset Options . . . . . 849
DFM Runset Options . . . . . 859
LVS Runset Options . . . . . 875
PERC Runset Options . . . . . 887
PEX Runset Options . . . . . 895
Common Runset Options . . . . . 918
Waiver Setup Runset Options . . . . . 944
3DSTACK Runset Options . . . . . 949
Runset Options for Template Parameters . . . . . 960


## Calibre Interactive Runset File Format

Input for: Calibre Interactive

Created by: Calibre Interactive

Calibre Interactive runset files are built automatically using the Calibre Interactive GUI. Although not recommended, it is possible to manually edit the options in a runset file.

The majority of runset options can be set using the Calibre Interactive GUI. For example, the Calibre nmDRC runset option, drcLayoutSystem, can be set by choosing the Inputs pane and selecting an option from the “Format” dropdown list. In most cases finding the GUI element corresponding to each runset option is straightforward.

### Format

The Calibre runset file is made up of runset options and their corresponding values. The syntax is as follows:

```
*OPTION: value
```




Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Parameters

See the section "Option Descriptions" on page 849 for a listing of the OPTION and value pairs for each Calibre Interactive application.

Environment variables may be used for runset values that correspond to Calibre Interactive fields that accept a pathname, such as the rule file name or the run directory.

### Related Topics

About Calibre Interactive Runsets

Option Descriptions

Runset Options for Template Parameters

Creating a Calibre Interactive Runset

# Creating a Calibre Interactive Runset

You can create a Calibre runset file from the Calibre Interactive GUI.

### Procedure

1. (Optional) To save all runset options, define the environment variable MGC_CALIBRE_SAVE_ALL_RUNSET_VALUES in your shell environment.

   By default, only runset options that differ from the default setting are saved to the runset file.

2. Start Calibre Interactive.
3. Make settings for the run in the Calibre Interactive GUI.
4. (Optional) Choose **Setup** > **Preferences**, then the **Info** tab.

   Provide the Runset Type and Runset Info information. If a runset has been tagged with this information, it is displayed in the Load Runset File dialog box.

5. Choose **File** > **Save Runset** or **File** > **Save Runset As** to save the GUI settings to a runset.

### Related Topics

About Calibre Interactive Runsets

Option Descriptions



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Option Descriptions

Each Calibre Interactive runset option has a corresponding value made up of a string of characters, one or more of a list of keywords, or a number.

Environment variables may be used for runset values that correspond to Calibre Interactive fields that accept a pathname, such as the rule file name or the run directory.

The following tables describe the runset options:

**DRC Runset Options** . . . . . 849
**DFM Runset Options** . . . . . 859
**LVS Runset Options** . . . . . 875
**PERC Runset Options** . . . . . 887
**PEX Runset Options** . . . . . 895
**Common Runset Options** . . . . . 918
**Waiver Setup Runset Options** . . . . . 944
**3DSTACK Runset Options** . . . . . 949
**Runset Options for Template Parameters** . . . . . 960

## DRC Runset Options

DRC runset options are specific to Calibre Interactive nmDRC.

**Table C-1. DRC Runset Options**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcActiveRecipe | Currently selected check recipe that is shown in the Recipe Editor dialog box. | [] () {} |
| drcAdditionalHCells | List of additional cells to be marked as cells to preserve. | [] () {} |
| drcAnalyzeMode | Boolean that controls whether to run in DRC Analyze mode. | [ 0/1 ] (Boolean) (0) |
| drcAppendSummary | Boolean that controls whether the summary file is overwritten (0) or appended to (1). | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcAutoHalo | Boolean that controls whether the tool should use an automatically calculated halo size for area DRC. This size is the larger dimension of the area DRC window. | [ 1/0 ] (Boolean) {1} |
| drcCellName | Boolean controlling whether cell names are written to the ASCII results database. | [ 0/1 ] (Boolean) {1} |
| drcClipToArea | Boolean controlling whether area DRC results are clipped to the window so that results outside the window are removed from the results database. Without clipping, all results from the run are retained, which may cause spurious results due to edge effects. | [ 1/0 ] (Boolean) {1} |
| drcConfigureDFMDefaults | Boolean which controls whether to configure the DFM Defaults RDB statement. | [ 0/1 ] (Boolean) {0} |
| drcCreateCTOFile | Boolean controlling whether a DRC CTO file is written from the DRC run. | [ 0/1 ] (Boolean) {0} |
| drcCreateDRCReport | Boolean controlling whether a DRC HTML report is written after the DRC run is complete. | [ 0/1 ] (Boolean) {0} |
| drcCreateXORRule | Boolean controlling whether to run DBdiff to create the XOR rules file before running FastXOR. | [ 0/1 ] (Boolean) {0} |
| drcDFMDefaultsResultsCellPrefix | Prefix for cells written to output GDS or OASIS databases for the DFM Defaults RDB statement. | [] () {} |
| drcDFMDefaultsResultsCellSuffix | Suffix for cells written to output GDS or OASIS databases for the DFM Defaults RDB statement. | [] () {} |
| drcDFMDefaultsResultsFile | File path of the output DRC database for the DFM Defaults RDB statement. | [] () {} |
| drcDFMDefaultsResultsFormat | Format specification for the output DRC database for the DFM Defaults RDB statement. | [GDS OASIS] () {GDS} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcDFMDefaultsResultsPseudoCells | Boolean controlling whether pseudo cells are written to output GDS or OASIS databases for the DFM Defaults RDB statement. | [ 0/1 ] (Boolean) {0} |
| drcDontOpenEmptyMainRDB | Boolean controlling whether Calibre RVE opens an empty DRC results file after a run in which side RDB files have been created. | [ 0/1 ] (Boolean) {0} |
| drcDRCAreaCoords | Two user-unit coordinate pairs for the bottom-left and top-right corners of the area DRC window. | [] (list of two coordinate pairs) {} |
| drcDRCCheckArea | Boolean controlling whether area DRC is to be run. | [ 0/1 ] (Boolean) {0} |
| drcDRCMaxResultsAll | Boolean controlling whether all results are written to the DRC results database (1), or whether the result count per check is limited to the DRCMaxResultsCount option (0). | [ 0/1 ] (Boolean) {0} |
| drcDRCMaxResultsCount | Maximum number of results (on a per check basis) to be written to the DRC results database. | [] (positive integer) {1000} |
| drcDRCMaxVertexAll | Boolean controlling whether a limit is specified for the maximum vertex count for polygons written to the DRC results database. | [ 0/1 ] (Boolean) {0} |
| drcDRCMaxVertexCount | Maximum number of vertices for any polygon output to the results database. Polygons with vertices greater than this number are segmented into 4 point polygons. | [] (positive integer) {4096} |
| drcDRCReportConfigFile | Path to the custom DRC HTML report configuration file. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcDRCReportConfigTemplate | Name of a predefined report configuration template. The allowed values are:<br>• Cell-Check<br>• Check-Cell<br>• Check-SC<br>• Check-Cell-SC<br>• Custom Template<br>If Custom Template is specified, drcDRCReportConfigFile specifies the path to the custom configuration file. See "Creating a DRC HTML Report from Calibre Interactive nmDRC" on page 185. | [see Description column for allowed values] () {Check-Cell-SC} |
| drcDRCReportDirectory | Path of the directory where the DRC HTML report is created | [] () {report} |
| drcDRCReportFormat | Output format for the DRC HTML report. | [HTML MHTML] () {HTML} |
| drcDynamicRVE | Boolean controlling whether Calibre RVE is started as soon as DRC results are available when DRC is being run. | [ 1/0 ] (Boolean) {1} |
| drcEnvNamePatterns | Filter for shell environment variables to display in Set Environment Variables dialog box. | [] () {MGC_* *CALIBRE* *DRC*} |
| drcEnvVars | {*name value use*} triplets of environment variable to apply to the DRC run. *use* should be one of:<br>• **Runset** — use the value from the triplet.<br>• **Env** — use the value from the environment<br>• **Unset** — unset the environment variable | [] () {} |
| drcExtraLayoutPaths | File paths of extra input GDS/OASIS databases. | [] (environment variable is valid) {} |
| drcGoldenLayoutPaths | File paths of any input GDS/OASIS databases to be marked "GOLDEN". | [] (environment variable valid) {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcGroupCategory | Boolean controlling if Group Categories are included in the generated DRC CTO file. | [ 0/1 ] (Boolean) {0} |
| drcHaloSize | Size in user-units of the halo region used for area DRC. Used when the drcAutoHalo setting is set to 0. | [] (non-negative real) {0.0} |
| drcHCellsFile | Name of hcells file used for DRC waivers. | [] (environment variable valid) {} |
| drcIncludeCmdsType | Specifies the type of commands to insert into the control file. | [none SVRF TVF] () {none} |
| drcIncludeFiles | List of paths of included rule files. | [] (environment variable valid) {} |
| drcLayerCategory | Boolean controlling if Layer Categories are included in the generated DRC CTO file. | [ 0/1 ] (Boolean) {0} |
| drcLayoutGetFromViewer | Boolean that controls the import from layout viewer option. This option has dependencies on cmnTemplate_IL; see "Export from Viewer Preferences" on page 152. | [ 0/1 ] (Boolean) {0} |
| drcLayoutGetFromViewer2 | Boolean that controls the import from layout viewer option for the second layout used for FastXOR applications. | [ 0/1 ] (Boolean) {0} |
| drcLayoutLibrary2 | The library name of the reference layout topcell to run DRC FastXOR on; only valid for Cadence Virtuoso. | [] () {} |
| drcLayoutLibrary | The library name of the topcell to run DRC on; only valid for Cadence Virtuoso. | [] () {} |
| drcLayoutPaths2 | File path of the reference input database for a FastXOR run. | [] () {} |
| drcLayoutPaths | File paths of the input GDS/OASIS databases. | [] (environment variable valid) {} |
| drcLayoutPrimary2 | The topcell name for the reference input database in a FastXOR run. | [] () {} |
| drcLayoutPrimary | The name of the topcell to run DRC on. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcLayoutSystem2 | Format of the reference input layout database for a FastXOR run. | [ GDSII OASIS OPENACCESS ] () {GDSII} |
| drcLayoutSystem | Format of the input layout database. | [ GDSII OASIS LEFDEF OPENACCESS ] () {GDSII} |
| drcLayoutView2 | The cell view name of the reference layout topcell to run DRC FastXOR on; only valid for Cadence Virtuoso. | [] () {} |
| drcLayoutView | The cell view name of the topcell to run DRC on; only valid for Cadence Virtuoso. | [] () {} |
| drcPostTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute after Calibre nmDRC is run. Type may have the values layout, schematic, or process, and enable is a boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| drcPreTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute before Calibre nmDRC is run. Type may have the values layout, schematic, or process, and enable is a boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| drcRecipeCategory | Boolean controlling if Recipe Categories are included in the generated DRC CTO file. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcReconMode | Controls the Calibre nmDRC Reconnaissance mode:<br>• 0 — Do not use Calibre nmDRC Recon mode<br>• 1 — Use Calibre nmDRC Recon mode<br>• 2 — Use Calibre nmDRC Recon Inverse mode | [ 0 1 2] (Boolean) {0} |
| drcResultsCellPrefix | Prefix for cells written to output GDS or OASIS databases. | [] () {} |
| drcResultsCellSuffix | Suffix for cells written to output GDS or OASIS databases. | [] () {} |
| drcResultsCheckText | Boolean controlling whether to output rule check text statements. | [ 0/1 ] (Boolean) {0} |
| drcResultsCheckTextUseFile | Boolean controlling whether to use a check text file containing the rule checks selected for the run and the check text specified by the keyword options. | [ 0/1 ] (Boolean) {0} |
| drcResultsCheckTextFile | The name of the check text file. | [] () {} |
| drcResultsCheckTextValue | The value for the DRC Check Text statement if DRCResultsCheckText is enabled. | [COMMENTSRFI COMMENTS ALLRFI ALL NONE] () {COMMENTSRFI} |
| drcResultsFile | File path of the output DRC database. | [] (environment variable valid) {drc.results} |
| drcResultsFormat | Format specification for the output DRC database. | [ASCII<br>GDSII<br>OASIS] () {ASCII} |
| drcResultsOasisCblock | Boolean controlling whether to embed compressed data in an OASIS results database (the CBLOCK keyword for the DRC Results Database statement). | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcResultsOasisIndex | Boolean controlling whether to index an OASIS results database; uses the INDEX keyword to DRC Results Database. | [ 0/1 ] (Boolean) {0} |
| drcResultsOasisNoview | Boolean controlling whether to optimize OASIS results data for direct input to Calibre MDP functions; uses the NOVIEW keyword to DRC Results Database. | [ 0/1 ] (Boolean) {0} |
| drcResultsOasisStrict | Boolean controlling whether to create strict-mode output for all fields in the table-offsets structure of an OASIS database (the STRICT keyword for the DRC Results database statement). | [ 0/1 ] (Boolean) {0} |
| drcResultsPseudoCells | Boolean controlling whether pseudo cells are written to output GDS or OASIS databases. | [ 0/1 ] (Boolean) {0} |
| drcResultsPseudoCellsValue | Controls pseudo cells handling in Calibre nmDRC-H for GDS or OASIS databases. | [PSEUDO USER_MERGED USER TOP COMBINE] () {PSEUDO} |
| drcRHDBFile | The name of the reusable hierarchical database (RHDB) file, generated from all input files in the control file. | [] () {} |
| drcRHDBMode | Controls whether to save or restore from the reusable hierarchical database (RHDB) file. | [SAVE RESTORE] () {SAVE} |
| drcRulesFile | The file path of the rule file to be associated with this runset. | [] (environment variable valid) {} |
| drcRunDir | Path of the run directory. | [] (environment variable valid) {.} |
| drcRunsetInfo | User-defined one line text specifying information about the runset. See “Runset Type and Runset Info” on page 45. | [] () {} |
| drcRunsetType | User-defined runset type. See “Runset Type and Runset Info” on page 45. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcScaleAreaCoords | Boolean controlling whether area DRC coordinates are scaled to take LAYOUT MAGNIFY and DRC RESULTS DATABASE PRECISION settings into account. | [ 1/0 ] (Boolean) {1} |
| drcShowChecksAlpha | Boolean controlling whether the checks in the check selection dialog are displayed alphabetically. | [ 0/1 ] (Boolean) {0} |
| drcShowGroupsAlpha | Boolean controlling whether the groups in the check selection dialog are displayed alphabetically. | [ 0/1 ] (Boolean) {0} |
| drcShowGroupsHier | Boolean controlling whether the groups in the check selection dialog are displayed hierarchically. | [ 0/1 ] (Boolean) {0} |
| drcShowGroupsTop | Boolean controlling whether the groups are displayed from the top down. | [ 0/1 ] (Boolean) {0} |
| drcSideRDBOpenList | List of names of side RDBs to open after the Calibre run, if StartRVE is true. | [] (list of RDB Names) {} |
| drcSpecifyUserRecipesFiles | Boolean controlling whether to import check recipe files at startup. (Setup>Preferences>Startup tab) | [ 0/1 ] (Boolean) {0} |
| drcStartRVE | Boolean controlling whether Calibre RVE is started after the DRC run, either at the end or as soon as results are available. | [ 1/0 ] (Boolean) {1} |
| drcSummaryFile | File path of the DRC summary file. | [] (environment variable valid) {drc.summary} |
| drcSVRFCmds | Additional SVRF commands to be inserted into the control file. | [] () {} |
| drcTemplate_DD[1] | DFM Defaults results database template. %l is substituted with the name of the top cell. | [] () {%l.dfmDefaults.db} |
| drcTemplate_RD[1] | DRC results database template. %l is substituted with the name of the top cell. | [] () {%l.drc.results} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcTemplate_RH[1] | Reusable hierarchical databse (RHDB) template. %l is substituted with the name of the top cell. | [] () {%l.rhdb} |
| drcTemplate_SF[1] | DRC summary file template. %l is substituted with the name of the top cell. | [] (environment variable valid) {%l.drc.summary} |
| drcTVFCmds | Additional TVF commands to be inserted into the control file. | [] () {} |
| drcUseAdditionalHCells | Boolean controlling whether additional cells to preserve are to be individually specified. | [ 0/1 ] (Boolean) {0} |
| drcUseHCellsWaiverFiles | Boolean controlling whether waiver files are to be read to obtain a list of cells to preserve. | [ 0/1 ] (Boolean) {0} |
| drcUseRHDB | Boolean controlling whether to use a reusable hierarchical database (RHDB) file. | [ 0/1 ] (Boolean) {0} |
| drcUserRecipesFiles | List of user recipe files to import at Calibre Interactive startup when drcSpecifyUserRecipesFiles is enabled. (Setup > Preferences > Startup tab) | [] () {} |
| drcUserRecipes | List of user-defined check recipes. | [] () {} |
| drcViewDRCReport | Boolean controlling whether a DRC HTML report is opened after it is created. | [ 0/1 ] (Boolean) {1} |
| drcViewerLayermapFile | The CTO File will be populated with layer visibility control statements (RVE Show Layers) using layer names from this file. | [] () {} |
| drcViewSummary | Boolean controlling whether the summary file is opened after DRC finishes | [ 1/0 ] (Boolean) {1} |
| drcWaiverHCellFiles | List of waiver files read to obtain list of cells to preserve. | [] (environment variable valid) {} |
| drcWarnCTOOverwrite | Boolean controlling whether a dialog is shown when the CTO file is about to be overwritten. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-1. DRC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| drcWriteSummary | Boolean controlling whether a DRC summary file is written from the DRC run. | [ 1/0 ] (Boolean) {1} |

1. The *Template_* runset options set templates for file naming.

**Note**

See "Templates for File Naming" on page 148 and "Runset Options for Template Parameters" on page 960 for more information on file naming templates.

# DFM Runset Options

DFM runset options are specific to Calibre Interactive DFM.

**Table C-2. DFM Runset Options**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmActiveRecipe | Currently selected check recipe that is shown in Recipe Editor dialog box. | [] () {} |
| dfmAlphaNegativeBinominal | Non-negative real number controlling the alpha parameter of the Negative Binomial yield model. | [] (Non-negative real) {0.5} |
| dfmAppendSummary | Boolean controlling whether the summary file is overwritten (0) or appended to (1). | [ 0/1 ] (Boolean) {0} |
| dfmArrayCells | Control list of array cells. | [] () {} |
| dfmArrayCellsSelMode | Boolean controlling selection of "Large Arrays" check box. | [ 0/1 ] (Boolean) {0} |
| dfmAutoHalo | Boolean controlling whether the tool should use an automatically calculated halo size for area DRC. This size is the larger dimension of the area DRC window. | [ 1/0 ] (Boolean) {1} |
| dfmBackupWrapBttn | Boolean controlling the "Backup and wrap flow" checkbox. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmBackWrapMode | Boolean controlling whether Backup (1) or Wrap (0) mode is enabled. | [ 1/0 ] (Boolean) {1} |
| dfmCAANormalizeArea | Sets a chip area to be normalized to in um^2. | [] () {} |
| dfmCAOutputFirstN | Number of reports for “First n” defect sizes in CAA analysis. | [] (positive integer) {3} |
| dfmCAOutputSelMode | Controls defining of results of output.<br>A — “None”<br>B — “All”<br>D — “First n” | [ A B D ] () {A} |
| dfmCellName | Boolean controlling whether cell names are written to the ASCII results database. | [ 0/1 ] (Boolean) {1} |
| dfmCellSelMode | Boolean controlling selection of the “By Cell” checkbox. | [ 0/1 ] (Boolean) {0} |
| dfmCellUsageFile | Path to the cell usage file. | [] () {} |
| dfmCheckButtonLSThresh | Boolean that controls the “Large Size Threshold” checkbox in CAA. | [ 0/1 ] (Boolean) {0} |
| dfmCheckFillExtent | Boolean controlling whether FILL_EXTENT should be derived or not. | [ 0/1 ] (Boolean) {0} |
| dfmClipToArea | Boolean controlling whether area run results are clipped to the window so that results outside the window are removed from the results database. Without clipping, all results from the run are retained, which may cause spurious results due to edge effects. This only has effect if the output database is ASCII. | [ 1/0 ] (Boolean) {1} |
| dfmCMPAddTAFStatements | Boolean controlling whether TAF connectivity statements are included in the rule file. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmCmpaSummaryReportConfiguration | Settings for producing the CMPA summary report. Five parameters are expected, corresponding to the options on the CMP tab on the Outputs pane:<br>• write_summary (Boolean)<br>• replace_or_append (w/a)<br>• filename (string)<br>• view_report (Boolean)<br>• view_prop_statistics (Boolean)<br>The parameter replace_or_append can take the values w or a. The other parameters are Booleans, with the exception of the filename. | [] () {} |
| dfmCMPExtraOption | Extra option for MGC CMP simulator geometry extraction | 0 (positive real) |
| dfmCMPMultiLayerProcessing | Boolean controlling whether the Calibre should run extraction and simulation in parallel or not. | [ 0/1 ] (Boolean) {0} |
| dfmCMPPreDefinedLayers | SVRF file for predefined layers. | [] () {} |
| dfmCMPRecipeFile | Recipe file for MGC CMP Analysis. | [] () {} |
| dfmCMPSaveGrid. | Boolean controlling whether the grid saving is allowed or not. | [ 0/1 ] (Boolean) {0} |
| dfmCMPSaveVectorProperties | Boolean controlling whether to save the data for each property as a vector on a single-geometry CMP layer. | [ 0/1] (Boolean) {0} |
| dfmCMPUDHSFile | UDHS definition file for CMP Analysis. | [] () {} |
| dfmDDKVendor | Format of DDK Vendor. | [ Mentor SEC UMC ] () {Mentor} |
| dfmDefectStatPaths | File path of the DDK vendor file specific to the chosen DDK type. | [] () {} |
| dfmDFMCAABaseLayers | List that shows states of base layer table. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmDFMCAADbSaveLayers | List of layer names and boolean values which control corresponding layer options for writing layers to the DFM CAA database. | [] () {Analyze 0<br>Devices 0<br>Measure 0<br>Original 0<br>Property 0<br>Density 0} |
| dfmDFMCAALayers | List of { Boolean “Custom Layer Name” } pairs for the layer table of CAA. The boolean parameter controls inclusion of the corresponding layer to CAA analysis. | [] () {} |
| dfmDFMCalcMode | Boolean that controls the DFM calculation mode.<br>0 — DFM CRITICAL AREA<br>1 — DFM CAF | [ 0/1 ] (Boolean) {0} |
| dfmDFMDbAnnotations | List of lists. Each sublist has 2 elements: name and value pair. | [] () {} |
| dfmDFMDbAnnotationSource | Specifies whether to use annotations from the rule file (R) or from the GUI (G). | [R G] () {R} |
| dfmDFMDbSaveConnect | Boolean controlling whether Connect layers are automatically saved to the DFM Database. | [ 0/1 ] (Boolean) {0} |
| dfmDFMDbSaveLayers | List of layer names and boolean values that control corresponding layer options for writing layers to DFM database. | [] () {Analyze 1<br>Devices 1<br>Measure 1<br>Original 1<br>Property 1<br>Density 1} |
| dfmDFMResultsDB | Directory path of the output DFM database. | [] () {dfmdb} |
| dfmDFMUseYSScript | Boolean that controls the “Use YieldServer post processing file” setting. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmDFMYSScript | Path of the YieldServer post processing file. | [] () {} |
| dfmDFMYSScriptArgs | List of arguments to the -ys command. | [] () {} |
| dfmDFMYSScriptMode | Specifies whether to run the YieldServer script in autostart mode or run as a YieldServer post processing script file. | [ AUTOSTART RUN_FROM_FILE] () {RUN_FROM_FILE } |
| dfmDontOpenEmptyMainRDB | Boolean controlling whether Calibre RVE opens an empty DFM results database after a run in which side RDB files have been created. | [ 0/1 ] (Boolean) {0} |
| dfmDRCAreaCoords | Two user-unit coordinate pairs giving the bottom-left and top-right corners of the area run window. | [] (two coordinate pairs) {} |
| dfmDRCCheckArea | Boolean controlling whether area Calibre is to be run. | [ 0/1 ] (Boolean) {0} |
| dfmDRCMaxResultsAll | Boolean controlling whether all results are written to the DRC results database (1), or whether the result count per check is limited to the dfmDRCMaxResultsCount option (0). | [ 0/1 ] (Boolean) {0} |
| dfmDRCMaxResultsCount | Maximum number of results (on a per check basis) to be written to the DRC results database. | [] (positive integer) {1000} |
| dfmDRCMaxVertexAll | Boolean controlling whether a limit is specified for the maximum vertex count for polygons written to the DRC results database. | [ 0/1 ] (Boolean) {0} |
| dfmDRCMaxVertexCount | Maximum number of vertices for any polygon output to the results database. Polygons with vertices greater than this number are segmented into 4-point polygons. | [] (positive integer) {4096} |
| dfmDynamicRVE | Boolean controlling whether Calibre RVE is started as soon as DRC results are available when DRC is being run. | [ 1/0 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmEnableNegativeFlow | Boolean controlling whether the slotted negative layers should be optimized or not. | [ 0/1 ] (Boolean) {1} |
| dfmEnableNetScoresHighlighting | Boolean controlling selection of “Score Net Shapes” control. This control is visible only for DFM CAF analysis mode. | [ 0/1 ] (Boolean) {0} |
| dfmEnvNamePatterns | Filter for shell environment variables to display in Set Environment Variables dialog box. | [] () {MGC_* *CALIBRE* *DFM*} |
| dfmEnvVars | {Name Value Use} triplets of environment variable to apply to the DFM run. Use should be one of:<br>• **Runset** — use the value from the triplet.<br>• **Env** — use the value from the environment<br>• **Unset** — unset the environment variable | [] (see description) {} |
| dfmExcludeCells | Control list of cells excluded from reports. | [] () {} |
| dfmExcludeCellsFromAnalysis | Control list of cells excluded from analysis. | [] () {} |
| dfmExponentialYieldModel | Boolean controlling selection of “Exponential” yield model. | [ 0/1 ] (Boolean) {0} |
| dfmExtentLayersBttn | Boolean controlling whether to use customer specified extent layer or not. | [ 0/1 ] (Boolean) {0} |
| dfmExtentLayersEntry | Customer specified extent layer number/number.datatype. | [] (positive real) {} |
| dfmExtraLayoutPaths | File paths of extra input GDS/OASIS databases. | [] () {} |
| dfmFillExtentLayer | FILL_EXTENT GDS layer and datatype | [] () {} |
| dfmGoldenLayoutPaths | File paths of any input GDS/OASIS databases to be marked “GOLDEN”. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmGoodDiePerWaferHandlingStateAndWaferParameters | Good Die Per Wafer handling state and wafer parameters. | [] () {} |
| dfmHaloSize | Size in user-units of the halo region used for area DRC. Used when the drcAutoHalo setting above is set to 0. | [] (non-negative real) {0.0} |
| dfmIncludeCmdsType | Specifies the type of commands to insert into the control file. | [none SVRF TVF] () {none} |
| dfmIncludeFiles | List of paths of included rule files. | [] () {} |
| dfmInterLayerShort | Boolean that controls “Inter Layer Short” defect type selection in CAA analysis. | [ 0/1 ] (Boolean) {0} |
| dfmLayerMapFile | Path to the CAA Layers File mapping technology layer names to Siemens standard layer names. | [] () {./caa_layer.txt} |
| dfmLayoutGetFromViewer | Boolean that controls the import from layout viewer option. This option has dependencies on cmnTemplate_IL; see “Export from Viewer Preferences” on page 152. | [ 0/1 ] (Boolean) {0} |
| dfmLayoutLibrary | The library name of the topcell to run DFM on, only valid for Cadence Virtuoso. | [] () {} |
| dfmLayoutPaths | File path of the main input GDS/OASIS database. | [] () {} |
| dfmLayoutPrecision | DFM layout precision value. | [] (integer) {1000} |
| dfmLayoutPrimary | The name of the topcell to run DFM on. | [] () {} |
| dfmLayoutShrink | Boolean controlling whether a layout shrink factor should be used (dfmShrinkValue). | [ 0/1 ] (Boolean) {1} |
| dfmLayoutSystem | Format of the input layout database. | [GDSII OASIS LEFDEF OPENACCESS] () {GDSII} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmLayoutView | The cell view name of the topcell to run DFM on, only valid for Cadence Virtuoso. | [] () {} |
| dfmLibraryPlacementGrid | Placement grid of 1x3 and 3x3 library flow on the **CAA** > **Analysis** sub-tab on the Inputs pane. All placements should belong to {N S FN FS} or {E W FE FW}. See the *Calibre YieldAnalyzer and YieldEnhancer User's and Reference Manual* for details about this option. | [placement1 placement2 placement3 placement1 placement2 placement3 placement4 placement5 placement6 placement7 placement8 placement9] () {} |
| dfmLibrarySelMode | Indicates the library flow mode. | [ 1x1 1x3 3x3 ] () {1x1} |
| dfmLSThresholdEntry | Positive value for "Large Size Threshold" in CAA. | [] (positive value) {100} |
| dfmMemoryConfigFile | Path of the memory configuration file. | [] () {} |
| dfmMetalLayerCountState | The number of metal layers in your layout. Use this with "Update" button to pre-select the layers from layer table. | [ 0 1 2 3 4 5 6 7 8 9 10 Custom ] () {9} |
| dfmMGCBucketWidthRange | Sets the bucketing with percentage (limit) used by MGC CMP simulator to calculate buckets width ranges for each tile. Number should be in [0,1] range. | (positive real) |
| dfmMGCCMPMetalLayerCount | Count of metal layers for CMP Analysis. | [1 2 3 4 5 6 7 8 9 10] () {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmMGCCMPMetalLayerList | Column-based list for specifying custom thresholds for metals in the MGC CMP layer table.<br>See "dfmMGCCMPMetalLayerList Description" on page 874 for a definition of the list parameters. | [] (){<br>{ltype {{Metal 1}}}<br>{lname M1}<br>{lnum 31.0}<br>{preetch {{}}}<br>{deposel {{}}}<br>{ebs {{}}}<br>{ebl {{}}}<br>{dmxnum {{}}}<br>{vnum {{}}}<br>{opc {{}}}<br>{nonopc {{}}}<br>{rthick 1}<br>{wins 20}<br>{arr 0}<br>{lst 0}<br>{dish {{}}}<br>{erosion {{}}}<br>{dof {{}}}<br>{antidish {{}}}<br>} |
| dfmMGCLastSimulatedLayer | Sets the last layer number, which should be simulated. | [] (positive integer) {} |
| dfmMGCOpenWireCASelmode | Boolean controlling selection of "Wire" (value is B) and "UWire" (value is A) checkboxes for MGC. | [ A B ] () {A} |
| dfmMGCShortCAOverlap | Boolean that controls the "CA Overlap" checkbox for MGC. | [ 0/1 ] (Boolean) {0} |
| dfmMGCSimCPUCount | Sets CPU count for MGC CMP simulator flow. | [] (positive integer) {4} |
| dfmMGCWindowSize | Sets tile size for MGC CMP simulator flow. | [] (real) {20} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmMurphyYieldModel | Boolean controlling selection of the “Murphy” yield model. | [ 0/1 ] (Boolean) {0} |
| dfmNDefects | The number of defect particles to investigate. | [] (positive integer) {0} |
| dfmNegativeBinominalYieldModel | Boolean controlling selection of the “Negative Binomial” yield model. | [ 0/1 ] (Boolean) {0} |
| dfmOctagonalMetric | Boolean that controls the “Use METRIC OCTAGONAL” checkbox if DFM CAF is selected. | [ 0/1 ] (Boolean) {0} |
| dfmOctagonalRegion | Boolean that controls the “Use REGION OCTAGONAL” checkbox if DFM CAF is not selected. | [ 0/1 ] (Boolean) {0} |
| dfmOpenCADepth | Positive value that shows maximum depth of open defect type. | [] (positive value) {0} |
| dfmOverrideUMCAlphaNegativeBinominal | Boolean controlling the Alpha Negative Binomial overriding mechanism (UMC DDK only).<br>**1** — AlphaNegativeBinominal Overrides all Alphas.<br>**0** — Overrides Alphas with “N” value. | [ 0/1 ] (Boolean) {0} |
| dfmPartitionSelMode | Controls selection of partition mode.<br>0 — “By Library”<br>1 — “By Chip” analysis (default)<br>2 — “By Net” | [ 0/1/2] () {1} |
| dfmPfxFilePaths | Path to a valid pfx file. Applicable to UMC DDK only. | [] () {} |
| dfmPfxPassword | Specification of the pfx password. Applicable to UMC DDK only. | [] () {} |
| dfmPoissonYieldModel | Boolean controlling selection of the “Poisson” yield model. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmPostTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute after Calibre DFM is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| dfmPreTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute before Calibre DFM is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| dfmResultsCellPrefix | Prefix for cells written to output GDS or OASIS databases. | [] () {} |
| dfmResultsCellSuffix | Suffix for cells written to output GDS or OASIS databases. | [] () {} |
| dfmResultsCheckText | Boolean controlling whether to output rule check text statements. | [ 0/1 ] (Boolean) {0} |
| dfmResultsCheckTextValue | The value for the DRC Check Text statement if dfmResultsCheckText is enabled. | [COMMENTSRFI COMMENTS ALLRFI ALL NONE] () {COMMENTSRFI} |
| dfmResultsFile | File path of the output results database. | [] () {dfm.results } |
| dfmResultsFormat | Format of the output database. | [DFMDB ASCII GDSII OASIS] () {DFMDB} |
| dfmResultsOasisCblock | Boolean controlling whether to embed compressed data in an OASIS results database (the CBLOCK keyword for the DRC Results database statement). | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmResultsOasisIndex | Boolean controlling whether to index an OASIS results database; uses the INDEX keyword to DRC Results Database. | [ 0/1 ] (Boolean) {0} |
| dfmResultsOasisNoview | Boolean controlling whether to optimize OASIS results data for direct input to Calibre MDP functions; uses the NOVIEW keyword to DRC Results Database. | [ 0/1 ] (Boolean) {0} |
| dfmResultsOasisStrict | Boolean controlling whether to create strict-mode output for all fields in the table-offsets structure of an OASIS database (the STRICT keyword for the DRC Results database statement). | [ 0/1 ] (Boolean) {0} |
| dfmResultsPseudoCells | Boolean controlling the handling of psuedocells in DRC results databases. | [ 0/1 ] (Boolean) {0} |
| dfmResultsPseudoCellsValue | Specifies pseudocell handling for DRC results databases. | [PSEUDO USER_MERGED USER TOP COMBINE] () {PSEUDO} |
| dfmRulesFile | File path of the rule file. | [] () {rules} |
| dfmRunDir | Path of the run directory. The default is the current directory (.). | [] () {.} |
| dfmRunMode | Mode in which to run Calibre DFM. | [CFA CAA LFD CMP CreateWaivers] () {CFA} |
| dfmRunsetInfo | One line of text for the runset info field. See “Runset Type and Runset Info” on page 45. | [] () {} |
| dfmRunsetType | One line of text for the runset type field. See “Runset Type and Runset Info” on page 45. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmScaleAreaCoords | Boolean controlling whether area coordinates are scaled to take LAYOUT MAGNIFY and DRC RESULTS DATABASE PRECISION settings into account. This only has effect if the output database is ASCII. | [ 1/0 ] (Boolean) {1} |
| dfmSeedsYieldModel | Boolean controlling selection of the “Seeds” yield model. | [ 0/1 ] (Boolean) {0} |
| dfmSelectedCells | Control list of selected cells. | [] () {} |
| dfmSelectedNets | Control list of selected nets. | [] () {} |
| dfmShortCADepth | Positive value that shows maximum depth of short defect type. | [] (positive integer) {0} |
| dfmShortCAExcludeEdge | Boolean that controls the “Exclude Gate Edge” checkbox. | [ 1/0 ] (Boolean) {1} |
| dfmShowChecksAlpha | Boolean controlling whether the checks in the check selection dialog are displayed alphabetically. | [ 0/1 ] (Boolean) {0} |
| dfmShowGroupsAlpha | Boolean controlling whether the groups in the check selection dialog are displayed alphabetically. | [ 0/1 ] (Boolean) {0} |
| dfmShowGroupsHier | Boolean controlling whether the groups in the check selection dialog are displayed hierarchically. | [ 0/1 ] (Boolean) {0} |
| dfmShowGroupsTop | Boolean controlling whether the groups are displayed from the top down. | [ 0/1 ] (Boolean) {0} |
| dfmShrinkValue | The shrink factor for dfmLayoutShrink. | [] (positive real) {1} |
| dfmSideRDBOpenList | List of names of side RDBs to open after the Calibre run, if StartRVE is true. | [] (list of RDB Names) {} |
| dfmSpecifyUserRecipesFiles | Boolean controlling whether to import check recipe files at startup.The check recipe file specifies the rule checks that are executed during the run. (Setup>Preferences>Startup tab) | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmStartRVE | Boolean controlling whether Calibre RVE is started after the Calibre run, either at the end or as soon as results are available. | [ 1/0 ] (Boolean) {1} |
| dfmSummaryFile | File path of the DFM summary file. | [] () {dfm.summary} |
| dfmSVRFCmds | Additional SVRF commands to be inserted into the control file. | [] () {} |
| dfmTemplate_RD[1] | DRC results database template. %l is substituted with the name of the top cell. | [] () {%l.dfm.results} |
| dfmTemplate_SF[1] | DRC summary file template. %l is substituted with the name of the top cell. | [] () {%l.dfm.summary} |
| dfmTSMCOpenWireCASelmode | Boolean controlling selection of “Wire” (value is B) and “UWire” (value is A) checkboxes for TSMC. | [ A B ] () {B} |
| dfmTSMCShortCAOverlap | Boolean that controls “CA Overlap” checkbox for TSMC. | [ 1/0 ] (Boolean) {1} |
| dfmTVFCmds | Additional TVF commands to be inserted into the control file. | [] () {} |
| dfmUMC10xM_layer | Number of 10xM metal layers for UMC DDK format. | [] (positive integer) {0} |
| dfmUMC1dot4xM_layer | Number of 1.4xM metal layers for UMC DDK format. | [] (positive integer) {0} |
| dfmUMC2xM_layer | Number of 2xM metal layers for UMC DDK format. | [] (positive integer) {0} |
| dfmUMC4xM_layer | Number of 4xM metal layers for UMC DDK format. | [] (positive integer) {0} |
| dfmUMC6xM_layer | Number of 6xM metal layers for UMC DDK format. | [] (positive integer) {0} |
| dfmUMC8xM_layer | Number of 8xM metal layers for UMC DDK format. | [] (positive integer) {0} |
| dfmUMCOpenWireCASelmode | Boolean controlling selection of “Wire” (value is B) and “UWire” (value is A) checkboxes for UMC. | [ A B ] () {A} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmUMCShortCAOverlap | Boolean that controls “CA Overlap” checkbox for UMC. | [ 0/1 ] (Boolean) {0} |
| dfmUMCUxM_layer | Number of UxM metal layers for UMC DDK format. | [] (positive integer) {0} |
| dfmUseCustomRuleDeck | Boolean controlling custom rule deck file usage. | [ 0/1 ] (Boolean) {0} |
| dfmUseExcludeLayers | Boolean controlling existence of the exclude layers in layers table (the “Use Exclude Layers” option on the **CAA** > **Settings** sub-tab.) | [ 0/1 ] (Boolean) {0} |
| dfmUseMarkerLayers | Boolean controlling the existence of marker layers in layers table (the “Use Marker Layers” option on the **CAA** > **Settings** tab.) | [ 0/1 ] (Boolean) {0} |
| dfmUseMGCNumberOfDefectsDefaults | Boolean that controls whether custom or default number of defects is selected for MGC DDK Vendor. | [ 1/0 ] (Boolean) {1} |
| dfmUserRecipes | List of user-defined check recipes. | [] () {} |
| dfmUserRecipesFiles | List of user recipe files to import at Calibre Interactive startup when dfmSpecifyUserRecipesFiles is enabled. The check recipe file specifies the rule checks that are executed during the run. (Setup > Preferences > Startup tab) | [] () {} |
| dfmViaCADepth | Positive value that shows maximum depth of “Via Failure” defect type. | [] (positive integer) {0} |
| dfmViaDefectType | Boolean that controls “Via” defect type selection in CAA analysis. | [ 1/0 ] (Boolean) {1} |
| dfmViaFailureRate | Controls the Via Failure’s Analysis type.<br>A — “Single”<br>B — “Open”<br>E — “Electrical” | [ A B E] () {A} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-2. DFM Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| dfmViewSummary | Boolean controlling whether the summary file is opened after DRC finishes. | [ 1/0 ] (Boolean) {1} |
| dfmWindowSelMode | Boolean controlling selection of the “By Window” checkbox. | [ 0/1] (Boolean) {0} |
| dfmWinSize | Shows windows size for window partition mode. | [] (positive integer) {50} |
| dfmWireOpenDefectType | Boolean that controls “Open” defect type selection in CAA analysis. | [ 1/0 ] (Boolean) {1} |
| dfmWireShortDefectType | Boolean that controls “Short” defect type selection in CAA analysis. | [ 1/0 ] (Boolean) {1} |
| dfmWrapGap | The value for the gap size between the original and copied shapes when in Wrap mode. | [] (positive real) {} |
| dfmWriteSummary | Boolean controlling whether a DFM summary file is written from the DFM run. | [ 0/1] (Boolean) {0} |

1. The *Template_* runset options set templates for file naming.

**Note**

See “Templates for File Naming” on page 148 and “Runset Options for Template Parameters” on page 960 for more information on file naming templates.

## dfmMGCCMPMetalLayerList Description

- ltype — The type and description of each layer used (Metal 1, Metal 2, …)
- lname — The name of each metal layer, which is saved in the dfmdb (M1, M2, …)
- lnum — Defines metal layers with either the GDS layer numbers or predefined layer names
- preetch — Pre-etch layers
- deposel — Deposit selective layers
- ebs — Dtch-back selective layers
- ebl — Dtch-back layers
- dmxnum — Obsolete. Dummy metal exclusion layer



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- vnum — Obsolete. Via layers
- opc — Obsolete. opc fill shape layers
- nonopc — Obsolete. nonopc fill shape layers
- rthick — Model index
- wins — Window size per layer (beta variable CMP_FAST_ENABLE)
- arr — Array recognition parameter per layer (beta variable CMP_FAST_ENABLE)
- lst — Layer stack topography
- dish — Threshold of “Dishing” hotspot rule
- erosion — Threshold of “Erosion” hotspot rule
- dof — Threshold of “Depth of Focus” hotspot rule
- antidish — Deprecated. TSMC specific parameter

# LVS Runset Options

LVS runset options are specific to Calibre Interactive nmLVS.

**Table C-3. LVS Runset Options**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsAbortOnSoftchk | Boolean that controls whether LVS should abort if soft connection errors are discovered in the extraction phase. | [ 0/1 ] (Boolean) {0} |
| lvsAbortOnSupplyError | Boolean that controls whether LVS should abort if power/ground errors are discovered in the extraction phase. | [ 1/0 ] (Boolean) {1} |
| lvsAutoMatch | Boolean that controls the “Match cells by name” (automatch) hcells setting. | [ 0/1 ] (Boolean) {0} |
| lvsConfigureSplitGates | Boolean that controls whether the GUI has control over the split gate reduction options:<br>• 0 — Use settings from rule file (default) and ignore GUI settings.<br>• 1 — Use GUI settings and ignore rule file settings. | [ 0/1 ] (Boolean) {0} |
| lvsCreateSVDB | Boolean that controls whether a SVDB directory is created from the LVS run. | [ 1/0 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsDeviceFilterOptionsDevicesList | Specifies Device type/subtype for each of LVS Filter Unused Option setting. | [] () {} |
| lvsDeviceFilterOptionsEnabled | Boolean that controls whether GUI settings for Filter Unused Device Options are used:<br>• 0 — GUI settings are ignored; rule file settings are used.<br>• 1 — GUI settings are used (default), and override the rule file settings.<br>See "Setting Filter Unused Device Options" on page 272. | [ 1/0 ] (Boolean) {1} |
| lvsDeviceLabelList | List of device seed layer label statements. | [] () {} |
| lvsDeviceSignatureFile | Output file from the Create Device Signature dialog. The file contains the device signature statements. | [] () {} |
| lvsDeviceSignatureFileAppend | Boolean controlling whether the signature file is overwritten (0, default) or appended to (1). | [ 0/1 ] (Boolean) {0} |
| lvsDeviceStatementList | List of device statements composed by the user. | [] () {} |
| lvsDeviceStatementSelect | Selected device statements. | [] () {} |
| lvsEnvNamePatterns | Filter for shell environment variables to display in Set Environment Variables dialog box. | [] () {MGC_*<br>*CALIBRE*<br>*LVS*} |
| lvsEnvVars | {Name value Use} triplets of environment variable to apply to the LVS run. Use should be one of:<br>• **Runset** — use the value from the triplet.<br>• **Env** — use the value from the environment<br>• **Unset** — unset the environment variable | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsERCAppendSummary | Boolean that controls whether the ERC summary report file is overwritten (0) or appended to (1). | [ 0/1 ] (Boolean) {0} |
| lvsERCCellName | Boolean controlling whether cell names are written to the ERC results database. | [ 0/1 ] (Boolean) {1} |
| lvsERCDatabase | Pathname of the ERC results database file. | [] () {erc.results} |
| lvsERCDatabasePseudoCells | Boolean to control the handling of pseudocells in ERC results databases. | [ 0/1 ] (Boolean) {0} |
| lvsERCDatabasePseudoCellsValue | Specifies pseudo cells handling in ERC results databases. | [PSEUDO TOP] () {PSEUDO} |
| lvsERCMaxResultsAll | Boolean that controls whether all results are written to the ERC results database (1), or are subject to a maximum count (0). | [ 0/1 ] (Boolean) {0} |
| lvsERCMaxResultsCount | Maximum number of results to write to the ERC results database file. | [] (positive integer) {1000} |
| lvsERCMaxVertexAll | Boolean that controls whether polygons written to the ERC database are segmented if their vertex count exceeds a maximum vertex count. | [ 0/1 ] (Boolean) {0} |
| lvsERCMaxVertexCount | The maximum vertex count for polygons written to the ERC results database. | [] (positive integer) {4096} |
| lvsERCSummaryFile | Pathname of the ERC summary report file. | [] () {erc.summary} |
| lvsExpandOnError | Boolean that controls whether to examine LVS discrepancies and, in certain cases, to automatically perform additional comparison runs with fewer hcells. | [ 0/1 ] (Boolean) {0} |
| lvsExpandOnErrorValue | Value for the lvsExpandOnError option. | [NO YES] () {NO} |
| lvsExtraLayoutPaths | File paths of extra input GDS/OASIS databases. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsExtraSourcePaths | File paths of extra input SPICE source netlist files. | [] () {} |
| lvsGeneratedSignatures | List of generated device signatures. | [] () {} |
| lvsGenHCells | Boolean that controls whether to automatically generate an hcell list. | [ 0/1 ] (Boolean) {0} |
| lvsGenHCellsSelect | Boolean that controls whether to use the default thresholds as the hcell selection criteria. | [ 0/1 ] (Boolean) {0} |
| lvsGroundNames | Names of ground net(s). | [] () {} |
| lvsHCellsFile | Path to the hcells file. | [] () {hcells} |
| lvsIgnorePorts | Boolean that controls the “Ignore layout and source ports during comparison” setting. | [ 0/1 ] (Boolean) {0} |
| lvsIncludeCmdsType | Specifies the type of commands to insert into the control file. | [none SVRF TVF] () {none} |
| lvsIncludeFiles | Pathnames of rule files to include in the control file. | [] () {} |
| lvsIsolateShorts | Boolean that controls whether short isolation is run. | [ 0/1 ] (Boolean) {0} |
| lvsIsolateShortsValue | Specifies the type of short isolation run for the “Run short isolation” setting. | [ YES NO POWERSHORTS GROUNDSHORTS PWRGNDSHORTS IOSHORTS ] () {YES} |
| lvsIsolateShortsAccumulate | Boolean that controls the “Identify common paths” setting to add the Accumulate keyword. | [ 0/1 ] (Boolean) {0} |
| lvsIsolateShortsAllNames | Boolean that controls the “Isolate shorts between all names” setting: isolate shorts between all names (1) or only between specified names (0). | [ 1/0 ] (Boolean) {1} |
| lvsIsolateShortsByCell | Boolean that controls the “Output shorts by Cell” setting. | [ 0/1 ] (Boolean) {0} |
| lvsIsolateShortsByLayer | Boolean that controls the “Output shorts by Layer” setting. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsIsolateShortsFlat | Boolean that controls the “Run flat short-isolation” setting. | [ 0/1 ] (Boolean) {0} |
| lvsIsolateShortsInTopCell | Boolean that controls the “Isolate shorts in top-cell only” setting (1), or the “Isolate shorts in all cells” (0) setting. | [ 1/0 ] (Boolean) {1} |
| lvsIsolateShortsNames | List of specific text points to isolate shorts between. | [] () {} |
| lvsIsolateShortsNoContacts | Boolean that controls the “Exclude contact polygons” setting. | [ 0/1 ] (Boolean) {0} |
| lvsIsolateShortsOp | Controls the AND or OR selection on the short-isolation controls page. The operation chosen is used to combine cell settings with name settings for short isolation. | [AND OR] () {AND} |
| IsolateShortsNegOp | Controls the NOT selection on the short-isolation controls page. | [ 0/1 ] (Boolean) {0} |
| lvsIsolateShortsMaxResults | Boolean that controls the “Maximum Short Results” setting. | [0/1] (Boolean) {0} |
| lvsIsolateShortsMaxResultsAll | Boolean that controls the “All” setting (1) or the “Count” setting (0) for the “Maximum Short Results” setting. | [0/1] (Boolean) {0} |
| lvsIsolateShortsMaxResultsCount | Specifies the value for when the “Maximum Short Results” setting is enabled and set to “Count”. | [] (integer) {50} |
| lvsLayersFile | Text file where the device layer information is listed for the Create Device Signature dialog. | [] () {} |
| lvsLayoutDeviceFilterOptions | Specifies LVS Filter Unused Option for the LAYOUT setting. Any combination of the allowed options may be specified. | [AB AC AD AE AF AH AG AI B C D E EG F FF FY G GF GY H I IH INV J K KF L M MG N NP O P Q RB RC RD RE RF RG S T U V YB YC ZB ZC ZD ZE] (any combination of allowed options) {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsLayoutGetFromViewer | Boolean that controls the import from layout viewer option. This option has dependencies on cmnTemplate_IL; see "Export from Viewer Preferences" on page 152. | [ 0/1 ] (Boolean) {0} |
| lvsLayoutLibrary | The library name of the layout topcell to run LVS on, only valid for Cadence Virtuoso. | [] () {} |
| lvsLayoutPaths | File paths of the input GDS/OASIS databases. | [] () {} |
| lvsLayoutPrimary | The name of the topcell for the Calibre nmLVS run. | [] () {} |
| lvsLayoutSystem | Format of the input layout database. | [GDSII OASIS LEFDEF OPENACCESS] () {GDSII} |
| lvsLayoutView | The cell view name of the layout topcell to run LVS on, only valid for Cadence Virtuoso. | [] () {} |
| lvsPlacementMatch | Boolean that controls the "Match cells by number of placements" hcell analysis setting. | [ 0/1 ] (Boolean) {0} |
| lvsPostTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute after Calibre nmLVS is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| lvsPowerNames | Names of power net(s). | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsPreTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute before Calibre nmLVS is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| lvsQsTclFile | File path of hcells Query Server Tcl script to be executed during the run. Valid only if lvsGenHCells and lvsUseQsTclFile are enabled. | [] () {qs_tcl_file} |
| lvsRecognizeGates | Controls the gate recognition option settings. | [ALL<br>SIMPLE<br>NONE] () {ALL} |
| lvsRecognizeGatesMixSubtypes | Boolean that controls the "Mix subtypes during gate recognition" setting. | [ 0/1 ] (Boolean) {0} |
| lvsReconSIMode | Controls the type of shorts to locate when running Calibre nmLVS Reconnaissance Short Isolation.<br>• **ALL** — Perform both Power Ground and IO short isolation<br>• **PG** — Power Ground short isolation only<br>• **OP** — IO short isolation only | [ ALL PG OP ] () {ALL} |
| lvsReduceParallelMOS | Boolean that controls whether to perform parallel MOS reduction.<br>If lvsReduceSplitGates is set to 1, then lvsReduceParallelMOS is automatically set to 1. To disable lvsReduceParallelMOS, both lvsReduceParallelMOS and lvsReduceSplitGates must be set to 0. See "Setting Split Gate and Parallel MOS Reduction Options" on page 271. | [ 1/0 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsReduceSameOrderSplitGates | Boolean that controls whether to reduce split gates only when the input order is the same. | [ 0/1 ] (Boolean) {0} |
| lvsReduceSemiSplitGates | Boolean that controls whether to reduce semi-split gates. | [ 0/1 ] (Boolean) {0} |
| lvsReduceSeriesParallelSplitGates | Boolean that controls whether to perform series-parallel split gate reduction. | [ 0/1 ] (Boolean) {0} |
| lvsReduceSplitGates | Boolean that controls whether to reduce split gates. Also see lvsReduceParallelMOS. | [ 1/0] (Boolean) {1} |
| lvsReduceSplitGatesMixTypes | Boolean that controls whether to allow mixed types during split gate reduction. | [ 0/1 ] (Boolean) {0} |
| lvsReduceSplitGatesWithinTolerance | Boolean that controls whether to reduce split gates only when within tolerance. | [ 0/1 ] (Boolean) {0} |
| lvsReportFile | Filename of the LVS report file. | [] () {lvs.report} |
| lvsReportMaximumAll | Boolean that controls whether LVS should report all discrepancies or a specific max count. | [ 0/1 ] (Boolean) {0} |
| lvsReportMaximumCount | Maximum number of discrepancies to report per hcell. | [] (integer) {50} |
| lvsReportOptions | Specifies LVS REPORT OPTION settings. Any combination of LVS Report options is allowed. | [A AV B BV BX C CV D E E1 EB EC EO ES F FX G H I LPE MC N NCA NE NOK NP O P R RA RD S SP SPE V W X XR] () {} |
| lvsRulesFile | File path of the rule file. | [] () {rules} |
| lvsRunDir | Path of the run directory. | [] () {. (current directory)} |
| lvsRunERC | Boolean that controls the “Run ERC” setting. | [ 1/0 ] (Boolean) {1} |
| lvsRunsetInfo | One line of text for the runset info field. See “Runset Type and Runset Info” on page 45. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsRunsetType | One line of text for the runset type field. See "Runset Type and Runset Info" on page 45. | [] () {} |
| lvsRunWhat | Controls the LVS functionality specified on the Inputs pane.<br>• LVN = Layout vs Netlist<br>• NVN = Netlist vs Netlist<br>• HNE = Hierarchical Netlist Extraction | [LVN NVN HNE] () {LVN} |
| lvsShortEquivalentNodes | Boolean that controls whether to short equipotential nodes together during split gate reduction. | [ 0/1] (Boolean) {0} |
| lvsShowChecksAlpha | Boolean that controls whether checks are sorted alphabetically in the ERC Check selector. | [ 0/1] (Boolean) {0} |
| lvsShowGroupsAlpha | Boolean that controls whether groups are sorted alphabetically in the ERC Check selector. | [ 0/1 ] (Boolean) {0} |
| lvsShowGroupsHier | Boolean that controls whether groups are shown hierarchically in the ERC Check selector. | [ 0/1 ] (Boolean) {0} |
| lvsShowGroupsTop | Boolean that controls whether groups are listed top-down in the ERC Check selector dialog. | [ 0/1] (Boolean) {0} |
| lvsShowSeedPromotions | Boolean that controls whether a seed promotions report is created. | [ 0/1 ] (Boolean) {0} |
| lvsShowSeedPromotionsMax | Maximum number of seed promotions reported to the seed promotions report. | [] (integer) {50} |
| lvsSignatureGUIGeom | Signature dialog initial position and size. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsSourceDeviceFilterOptions | Specifies LVS Filter Unused Option for the SOURCE setting.<br><br>Any combination of the allowed options may be specified. | [AB AC AD AE AF AH AG AI B C D E EG F FF FY G GF GY H I IH INV J K KF L M MG N NP O P Q RB RC RD RE RF RG S T U V YB YC ZB ZC ZD ZE] (any combination of allowed options) {} |
| lvsSourceGetFromViewer2 | Boolean that controls the import from schematic viewer option for the layout netlist in netlist vs netlist comparison runs. | [ 0/1 ] (Boolean) {0} |
| lvsSourceGetFromViewer | Boolean that controls the import from schematic viewer option for the source netlist. This option has dependencies on lvsTemplate_IS; see "Export from Viewer Preferences" on page 152. | [ 0/1 ] (Boolean) {0} |
| lvsSourceLibrary2 | The library name of the layout netlist topcell to run LVS netlist vs netlist comparison on, only valid for Cadence Virtuoso. | [] () {} |
| lvsSourceLibrary | The library name of the source topcell to run LVS on, only valid for Cadence Virtuoso. | [] () {} |
| lvsSourcePath | File paths of the input source netlist files. | [] () {} |
| lvsSourcePrimary | The name of the topcell in the source netlist. | [] () {} |
| lvsSourceSystem2 | Format of the input layout netlist file for netlist vs netlist comparison runs. | [SPICE] () {SPICE} |
| lvsSourceSystem | Format of the input source netlist file. | [SPICE VERILOG MIXED] () {SPICE} |
| lvsSourceView2 | The cell view name of the layout netlist topcell to run LVS netlist vs netlist comparison on, only valid for Cadence Virtuoso. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsSourceView | The cell view name of the source topcell to run LVS on, only valid for Cadence Virtuoso. | [] () {} |
| lvsSpiceFile | Path to the SPICE netlist extracted from the layout. | [] () {lay.net} |
| lvsStartRVE | Boolean that controls whether Calibre RVE is started after the LVS run. | [ 1/0 ] (Boolean) {1} |
| lvsSVDBcci | Boolean that controls the “Generate Calibre Connectivity Interface data” setting in the Outputs pane. | [ 0/1 ] (Boolean) {0} |
| lvsSVDBDir | The directory path of the SVDB directory (relative to the run directory). | [] () {svdb} |
| lvsSVDBnopinloc | Boolean that controls whether pin location generation is suppressed if Calibre Connectivity Interface (CCI) is enabled. | [ 0/1 ] (Boolean) {0} |
| lvsSVDBtxref | Boolean that controls the “Generate ASCII cross-reference files” setting in the Outputs pane. | [ 0/1 ] (Boolean) {0} |
| lvsSVDBxcal | Boolean that controls the “Generate data from Calibre-xRC” setting in the Outputs pane. | [ 0/1 ] (Boolean) {0} |
| lvsSVRFCmds | Additional SVRF commands to be inserted into the control file. | [] () {} |
| lvsTemplate_ED[1] | Template for the ERC results filename. | [] () {%l.erc.results} |
| lvsTemplate_ES[1] | Template for the extracted spice netlist file. | [] () {%l.sp} |
| lvsTemplate_EU[1] | Template for the ERC summary filename. | [] () {%l.erc.summary} |
| lvsTemplate_HF[1] | Template for the hcells filename | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-3. LVS Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| lvsTemplate_IS[1] | Controls the template preference "Export netlist from schematic viewer" setting on the **Templates** > **Inputs** sub-tab in the Setup Preferences dialog box.<br>1 — Yes, export from viewer<br>0 — Do not export from viewer<br>-1 — Use runset value.<br>See "Export from Viewer Preferences" on page 152. | [1 0 -1] () {-1} |
| lvsTemplate_LR[1] | The LVS report filename template. | [] () {%l.lvs.report} |
| lvsTemplate_LS[1] | Template for the source library name for database formats. | [] () {%S} |
| lvsTemplate_SP[1] | The Source Path filename template. | [] () {%s.src.net} |
| lvsTemplate_SV[1] | Template for the SVDB database directory name. | [] () {svdb} |
| lvsTemplate_VS[1] | Template for the source view name for database formats. | [] () {%W} |
| lvsTVFCmds | Additional TVF commands to be inserted into the control file. | [] () {} |
| lvsUseHCells | Boolean that controls the "Use H-Cells File" setting. | [ 0/1 ] (Boolean) {0} |
| lvsUseQsTclFile | Boolean that controls the "Use H-Cells Query Server Tcl script" setting. | [ 0/1 ] (Boolean) {0} |
| lvsViewExtractionReport | Boolean that controls whether the Extraction Report file is displayed after the run. | [ 0/1 ] (Boolean) {0} |
| lvsViewLVSSummaryFile | Boolean that controls whether the LVS Summary Report file is displayed after the run. | [ 0/1 ] (Boolean) {0} |
| lvsViewReport | Boolean that controls whether the LVS Report file is displayed after the run. | [ 1/0 ] (Boolean) {1} |
| lvsVlogSourcePath | File paths of the input Verilog source netlist files in the mixed mode. | [] () {} |

1. The *Template_* runset options set templates for file naming.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Note**

See "Templates for File Naming" on page 148 and "Runset Options for Template Parameters" on page 960 for more information on file naming templates.

# PERC Runset Options

PERC runset options are specific to Calibre Interactive PERC.

**Table C-4. PERC Runset Options**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| percAbortOnSoftchk | Boolean that controls whether PERC should abort if soft connection errors are discovered in the extraction phase. | [ 0/1 ] (Boolean) {0} |
| percAutoMatch | Boolean that controls the "Match cells by name" (automatch) hcells setting. | [ 0/1 ] (Boolean) {0} |
| percCellInfoFile | File path of the Cell Info file in the Calibre PERC Cell-based flow. | [] () {} |
| percConfigureSplitGates | Boolean that controls whether the GUI has control over the split gate reduction options:<br>• 0 — Use settings from rule file (default) and ignore GUI settings.<br>• 1 — Use GUI settings and ignore rule file settings. | [ 0/1 ] (Boolean) {0} |
| percCreateSVDB | Boolean that controls whether a SVDB directory is created from the run. | [ 1/0 ] (Boolean) {1} |
| percDeviceFilterOptionsDevicesList | Specifies Device type/subtype for each LVS Filter Unused Option setting. | [] () {} |
| percDeviceFilterOptionsEnabled | Boolean that controls whether GUI settings for Filter Unused Device Options are used:<br>• 0 — GUI settings are ignored; rule file settings are used.<br>• 1 — GUI settings are used (default), and override the rule file settings.<br>See "Setting Filter Unused Device Options" on page 272. | [ 1/0 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-4. PERC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| percEnableLEFDEFDirectRead | Boolean that controls whether to enable the Calibre PERC Cell-based flow. | [ 0/1 ] (Boolean) {0} |
| percEnableLvsSupplement | Boolean controlling whether to allow Hierarchical LVS license pairs to be used when -turbo option is also specified. Adds the "-lvs_supplement num_processors" command line option. | [ 0/1 ] (Boolean) {0} |
| percEnvNamePatterns | Filter for shell environment variables to display in Set Environment Variables dialog box. | [] () {MGC_* *CALIBRE* *LVS*} |
| percEnvVars | {Name value Use} triplets of environment variable to apply to the run. Use should be one of:<br>• **Runset** — Use the value from the triplet.<br>• **Env** — Use the value from the environment.<br>• **Unset** — Unset the environment variable. | [] () {} |
| percExtraLayoutPaths | File paths of extra input GDS/OASIS databases. | [] () {} |
| percExtraSourcePaths | File paths of additional input SPICE source netlist files. | [] () {} |
| percGroundNames | Names of ground net(s). | [] () {} |
| percHCellsFile | Path to the hcells file. | [] () {hcells} |
| percIncludeCmdsType | Specifies the type of commands to insert into the control file. | [none SVRF TVF] () {none} |
| percIncludeFiles | Pathnames of rule files to include in the control file. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-4. PERC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| percLayoutDeviceFilterOptions | Specifies LVS Filter Unused Option for the LAYOUT setting. Any combination of the allowed options may be specified. | [AB AC AD AE AF AH AG AI B C D E EG F FF FY G GF GY H I IH INV J K KF L M MG N NP O P Q RB RC RD RE RF RG S T U V YB YC ZB ZC ZD ZE] (any combination of allowed options) {} |
| percLayoutGetFromViewer | Boolean that controls the import from layout viewer option. This option has dependencies on cmnTemplate_IL; see "Export from Viewer Preferences" on page 152. | [ 0/1 ] (Boolean) {0} |
| percLayoutLibrary | The library name of the topcell to run PERC on, only valid for Cadence Virtuoso. | [] () {} |
| percLayoutPaths | File paths of the input GDS/OASIS databases. | [] () {} |
| percLayoutPrimary | The name of the topcell for the PERC run. | [] () {} |
| percLayoutSystem | Format of the input layout database. | [GDSII OASIS LEFDEF OPENACCESS] () {GDSII} |
| percLayoutView | The cell view name of the topcell to run PERC on; only valid for Cadence Virtuoso. | [] () {} |
| percNumLvsSupplement | Number of Hierarchical LVS supplement license pairs to be used as -turbo reserves from available PERC licenses. Used with percEnableLvsSupplement. | ["All" or positive integer] () {All} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-4. PERC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| percPercInput | Controls the PERC functionality specified on the Input pane.<br>LAYOUT = Input as Layout,<br>LAYOUTNETLIST = Input as Layout Netlist,<br>SOURCENETLIST = Input as Source Netlist | [ LAYOUT LAYOUTNETLIST SOURCENETLIST ] () {LAYOUT} |
| percPERCLDLProbePath | File path of the PERC LDL Probe Path file. | [] () {} |
| percPERCRuleGenFile | File path of the PERC Rule Gen file. | [] () {percrulegen.prg} |
| percPERCWaiverPath | Path for the file specified in the PERC Waiver Path statement. | [] () {PERC.waivers} |
| percPostTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute after Calibre PERC is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| percPowerNames | Names of power net(s). | [] () {} |
| percPreTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute before Calibre PERC is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| percRecognizeGates | Controls the gate recognition option settings. | [ALL<br>SIMPLE<br>NONE] () {ALL} |
| percRecognizeGatesMixSubtypes | Boolean that controls the "Mix subtypes during gate recognition" setting. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-4. PERC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| percReduceParallelMOS | Boolean that controls whether to perform parallel MOS reduction.<br><br>If percReduceSplitGates is set to 1, then percReduceParallelMOS is automatically set to 1. To disable percReduceParallelMOS, both percReduceParallelMOS and percReduceSplitGates must be set to 0. See "Setting Split Gate and Parallel MOS Reduction Options" on page 271. | [ 1/0] (Boolean) {1} |
| percReduceSameOrderSplitGates | Boolean that controls whether to reduce split gates only when the input order is the same. | [ 0/1 ] (Boolean) {0} |
| percReduceSemiSplitGates | Boolean that controls whether to reduce semi-split gates. | [ 0/1 ] (Boolean) {0} |
| percReduceSeriesParallelSplitGates | Boolean that controls whether to perform series-parallel split gate reduction. | [ 0/1 ] (Boolean) {0} |
| percReduceSplitGates | Boolean that controls whether to reduce split gates. Also see percReduceParallelMOS. | [ 1/0] (Boolean) {1} |
| percReduceSplitGatesMixTypes | Boolean that controls whether to allow mixed types during split gate reduction. | [ 0/1 ] (Boolean) {0} |
| percReduceSplitGatesWithinTolerance | Boolean that controls whether to reduce split gates only when within tolerance. | [ 0/1 ] (Boolean) {0} |
| percReportFile | Filename of the PERC report file. | [] () {perc.report} |
| percReportMaximumAll | Boolean that controls whether LVS should report all discrepancies or a specific max count. | [ 0/1 ] (Boolean) {0} |
| percReportMaximumCount | Maximum number of discrepancies to report per hcell. | [] (integer) {50} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-4. PERC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| percReportOptions | Specifies PERC Report Option settings. Can be any combination of the available options. | [ALL_NET_TYPE NO_NET_TYPE NO_DEVICE_PIN SKIP_PASSED_CEL SKIP_WAIVED_RESULT TRACK_RESULT_SUBTITLE MIN_MAX_VOLTAGE_ONLY)] () {}) |
| percReportOptionsEnabled | Boolean that controls whether the GUI has control over the PERC Report options:<br>• 0 — Use settings from rule file (and ignore GUI settings.<br>• 1 — Use GUI settings (default) and ignore rule file settings. | [ 1/0] (Boolean) {1} |
| percRulesFile | File path of the rule file. | [] () {rules} |
| percRunDir | Path of the run directory. | [] () {. (current directory)} |
| percRunsetInfo | One line of text for the runset info field. See "Runset Type and Runset Info" on page 45. | [] () {} |
| percRunsetType | One line of text for the runset type field. See "Runset Type and Runset Info" on page 45. | [] () {} |
| percShortEquivalentNodes | Boolean that controls whether to short equipotential nodes together during split gate reduction. | [ 0/1 ] (Boolean) {0} |
| percShowSeedPromotions | Boolean that controls whether a seed promotions report is created. | [ 0/1 ] (Boolean) {0} |
| percShowSeedPromotionsMax | Maximum number of seed promotions reported to the seed promotions report. | [] (integer) {50} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-4. PERC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| percSourceDeviceFilterOptions | Specifies LVS Filter Unused Option for the SOURCE setting.<br><br>Any combination of the allowed options may be specified. | [AB AC AD AE AF AH AG AI B C D E EG F FF FY G GFGY H I IH INV J K L KF M MG N NP O P Q RB RC RD RE RF RG S T U V YB YC ZB ZC ZD ZE] (any combination of allowed options) {} |
| percSourceGetFromViewer | Boolean that controls the import from schematic viewer option. This option has dependencies on percTemplate_IS; see "Export from Viewer Preferences" on page 152. | [ 0/1 ] (Boolean) {0} |
| percSourceLibrary | The library name of the source topcell to run PERC on; only valid for Cadence Virtuoso. | [] () {} |
| percSourcePath | File paths of the input source netlist files. | [] () {} |
| percSourcePrimary | The name of the topcell in the source netlist. | [] () {} |
| percSourceSystem | Format of the input source netlist file. | [SPICE VERILOG MIXED] () {} |
| percSourceView | The cell view name of the source topcell to run PERC on; only valid for Cadence Virtuoso. | [] () {} |
| percSpecifyCellInfoFile | Boolean that controls whether a Cell Info File is specified in the Calibre PERC Cell-based flow. | [ 0/1 ] (Boolean) {0} |
| percSpiceFile | Path to the SPICE netlist extracted from the layout. | [] () {lay.net} |
| percStartRVE | Boolean that controls whether Calibre RVE is started after the LVS run. | [ 1/0 ] (Boolean) {1} |
| percSVDBDir | The directory path of the SVDB directory (relative to the run directory). | [] () {svdb} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-4. PERC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| percSVDBmuwdb | Boolean controlling whether multi-user mode is enabled for the PERC waiver database. | [ 0/1 ] (Boolean) {0} |
| percSVRFCmds | Additional SVRF commands to be inserted into the control file. | [] () {} |
| percTemplate_ES[1] | Template for the extracted spice netlist file. | [] () {%l.sp} |
| percTemplate_HF[1] | Template for the hcells filename. | [] () {} |
| percTemplate_IS[1] | Controls the template preference "Export netlist from schematic viewer" setting on the **Templates** > **Inputs** sub-tab in the Setup Preferences dialog box.<br>1 — Yes, export from viewer<br>0 — Do not export from viewer<br>-1 — Use runset value.<br>See "Export from Viewer Preferences" on page 152. | [1 0 -1] () {-1} |
| percTemplate_LR[1] | The PERC report filename template. | [] () {%l.perc.report} |
| percTemplate_LS[1] | Template for the source library name for database formats. | [] () {%S} |
| percTemplate_SP[1] | The source path filename template. | [] () {%s.src.net} |
| percTemplate_SV[1] | Template for the SVDB database directory name. | [] () {svdb} |
| percTemplate_VS[1] | Template for the source view name for database formats. | [] () {%W} |
| percTVFCmds | Additional TVF commands to be inserted into the control file. | [] () {} |
| percUseHCells | Boolean that controls the "Use H-Cells File" setting. | [ 0/1 ] (Boolean) {0} |
| percUsePERCLDLProbePath | Boolean controlling whether to use the PERC LDL Probe Path file. | [ 0/1 ] (Boolean) {0} |
| percUsePERCWaiverPath | Boolean controlling whether to use the PERC Waiver Path file. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-4. PERC Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| percViewPERCSummaryFile | Boolean controlling whether the PERC Summary Report file is displayed after the run. | [ 0/1 ] (Boolean) {0} |
| percViewReport | Boolean that controls whether the PERC Report file is displayed after the run. | [ 1/0 ] (Boolean) {1} |
| percVlogSourcePath | File paths of the input verilog source netlist files in the mixed mode. | [] () {} |

1. The *Template_* runset options set templates for file naming.

**Note**

See "Templates for File Naming" on page 148 and "Runset Options for Template Parameters" on page 960 for more information on file naming templates.

# PEX Runset Options

PEX runset options are specific to Calibre Interactive PEX.

**Table C-5. PEX Runset Options**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex3DExtractSelectedNets | Boolean controlling whether to extract parasitics using xACT 3D for only specific nets. | [ 0/1 ] (Boolean) {0} |
| pex3DFieldSolverAccuracy | Sets the accuracy mode for the xACT 3D fieldsolver. | [200 600] () {200} |
| pex3DRecursiveExcludeNets | Boolean controlling whether to exclude specific recursive nets for xACT 3D. | [ 0/1 ] (Boolean) {0} |
| pex3DRecursiveExcludeNets Value | Names of recursive nets for xACT 3D to specifically exclude. | [] () {} |
| pex3DRecursiveIncludeNets | Boolean controlling whether to include specific recursive nets for xACT 3D. | [ 0/1 ] (Boolean) {0} |
| pex3DRecursiveIncludeNetsA ccuracy | xACT 3D field solver accuracy mode for recursive include nets. | [200 600] () {200} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex3DRecursiveIncludeNetsValue | Names of recursive nets for xACT 3D to specifically include. | [] () {} |
| pex3DTopLevelExcludeNets | Boolean controlling whether to exclude specific top level nets for xACT 3D. | [ 0/1 ] (Boolean) {0} |
| pex3DTopLevelExcludeNetsValue | Names of top level nets for xACT 3D to specifically exclude. | [] () {} |
| pex3DTopLevelIncludeNets | Boolean controlling whether to include specific top level nets for xACT 3D. | [ 0/1 ] (Boolean) {0} |
| pex3DTopLevelIncludeNetsAccuracy | xACT 3D field solver accuracy mode for top level include nets. | [200 600] () {200} |
| pex3DTopLevelIncludeNetsValue | Names of top level nets to specifically include. | [] () {} |
| pexADMSNetlistOutputFiles | List of {block output_file_names} for SPICE modules. | [] () {} |
| pexADMSNetlistOutputVerilogModules | List of {block output_file_names} for Verilog modules. | [] () {} |
| pexAutoMatch | Boolean that controls the “Match cells by name” (automatch) hcells setting. | [ 0/1 ] (Boolean) {0} |
| pexBlockHWView | Sets viewing option for **Blocks** tab of the Inputs pane. | [hier list] () {hier} |
| pexBlocksForSeparateExtraction | List of blocks for separate extraction in the ADMS flow.<br>List of {blockname format extraction_type annot} lists, where the entries may have following values:<br>• format — DSPF, SPEF, HSPICE, SPEF, or SDF<br>• extraction_type — RCC, RC, or R<br>• annot — indicates whether cross-annotation will be applied. | [] (see description) {} |
| pexCalibreViewCellMapFile | File path(s) of the cellmap file(s) for Calibre View generation. | [] () {calview.cellmap} |
| pexCalibreViewCellMapFileEnabled | Boolean controlling whether to set the CalibreView cellmap file(s). | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexConfigureSplitGates | Boolean that controls whether the GUI has control over the split gate reduction options:<br>• 0 — Use settings from rule file (default) and ignore GUI settings.<br>• 1 — Use GUI settings and ignore rule file settings. | [ 0/1 ] (Boolean) {0} |
| pexDCFormat | Format of netlist file extracted for input to the delay calculation tool. | [SPEF DSPF] () {SPEF} |
| pexDCOtherToolCmd | Invocation line for delay calculation tool other than Time-It. | [] () {} |
| pexDCSDCFile | SDC constraints input file for the delay calculation tool. | [] () {} |
| pexDCTechLibFiles | Technology library files for the delay calculation tool. | [] () {} |
| pexDCTimeItMgcHome | MGC_HOME value to use for Time-It. | [] () {} |
| pexDCTool | Delay Calculation tool for the ADMS flow. | [TIME-IT OTHER] () {TIME-IT} |
| pexDCVerilogFiles | Additional Verilog input files for the delay calculation tool. | [] () {} |
| pexDefLayerMapFile | File path to DEF layer map file. | [] () {} |
| pexDeviceFilterOptionsDevicesList | Specifies Device type/subtype for each of LVS Filter Unused Option setting. | [] () {} |
| pexDeviceFilterOptionsEnabled | Boolean that controls whether GUI settings for Filter Unused Device Options are used:<br>• 0 — GUI settings are ignored; rule file settings are used.<br>• 1 — GUI settings are used (default), and override the rule file settings.<br>See "Setting Filter Unused Device Options" on page 272. | [ 1/0 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexEnvNamePatterns | Filter for shell environment variables to display in Set Environment Variables dialog box. | [] () {MGC_* *CALIBRE* *PEX*} |
| pexEnvVars | {Name value Use} triplets of environment variable to apply to the PEX run. "Use" should be one of:<br>• **Runset** — Use the value from the triplet.<br>• **Env** — Use the value from the environment.<br>• **Unset** — Unset the environment variable. | [] () {} |
| pexPexExtractionMode | Sets the parasitic extraction mode:<br>• xRC — model-based extraction<br>• 3D — xACT 3D fieldsolver | [xRC 3D] () {xRC} |
| pexExtraLayoutPaths | File paths of extra input GDS/OASIS databases. | [] () {} |
| pexExtraSourcePaths | File paths of additional input SPICE source netlist files. | [] () {} |
| pexFilterX | Boolean that controls whether to apply FILTERX in ADMS format netlist. See "Manipulating Cell Extraction Options Using the Blocks Tab" on page 308. | [ 0/1 ] (Boolean) {0} |
| pexFmtInfo | Boolean that controls whether the -fmt_info switch of the formatter is enabled. | [ 0/1 ] (Boolean) {0} |
| pexFmtWarnings | Boolean that controls whether the -fmt_warnings switch of the formatter is enabled. | [ 0/1 ] (Boolean) {0} |
| pexGroundNames | Names of ground net(s) for LVS operations. | [] () {} |
| pexHCellsFile | File path of hcells file to use for LVS. | [] () {hcells} |
| pexIncludeCmdsType | Specifies the type of commands to insert into the control file. | [none SVRF TVF] () {none} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexIncludeFiles | Pathnames of rule files to include in the control file. | [] () {} |
| pexInductBroadband | Boolean that controls whether broadband frequency is used for inductance extraction. | [ 0/1 ] (Boolean) {0} |
| pexInductFilterTags | Values specifying how to generate a driver/receiver file for inductance extraction. | [] () {} |
| pexInductFilterType | Specifies type of net filtering to apply during inductance extraction. | [APPROX OFF FILE] () {APPROX} |
| pexInductFreqGHz | Global maximum operating frequency (GHz). | [] (positive real) {10} |
| pexInductOpFreqOrSwitch | Specifies whether a global operating frequency (freq) or a global switch time (switch) will be used during inductance extraction. | [freq switch] () {freq} |
| pexInductRange | Inductance range value. | [] (positive real) {} |
| pexInductRetPath | Specifies ground or power nets to use in the return path. GANDPAVG indicates an average of ground and power, and GANDP indicates both. | [GROUND GANDPAVG POWER GANDP NONE] () {GROUND} |
| pexInductSameNetMutual | Boolean that controls whether to enable the extraction of the mutual inductance between wire segments within the same net. | [ 0/1 ] (Boolean) {0} |
| pexInductSwitchTime | Global switch time for inductance extraction. | [] (positive real) {} |
| pexInductUseDefaultRange | Boolean that controls whether the cell extent should be used as inductance range. | [ 1/0 ] (Boolean) {1} |
| pexInductUseDRTFile | Boolean that controls whether a driver/receiver file will be used in inductance analysis. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexInductUseRetPath | Boolean that controls whether to use ground or power nets in the return path. Controls the "Nets to include in return path" checkbox; setting to 0 omits the PEX Inductance Returnpath statement from the rule file. | [ 1/0 ] (Boolean) {1} |
| pexInductUseVpathFile | Boolean that controls whether a victim file will be used during inductance extraction. | [ 0/1 ] (Boolean) {0} |
| pexInductVpathFile | File path for the inductance victim file. | [] () {victim.xl} |
| pexLayoutDeviceFilterOptions | Specifies LVS Filter Unused Option for the LAYOUT setting. Any combination of the allowed options can be used. | [AB AC AD AE AF AH AG AI B C D E EG F FF FY G GF GY H I IH INV J K KF L M MG N NP O P Q RB RC RD RE RF RG S T U V YB YC ZB ZC ZD ZE] (any combination of allowed options) {} |
| pexLayoutGetFromViewer | Boolean that controls the "Export from layout viewer" setting on the Inputs pane in the **Layout** tab. This option has dependencies on cmnTemplate_IL; see "Export from Viewer Preferences" on page 152. | [ 0/1 ] (Boolean) {0} |
| pexLayoutGetLEFFromViewer | Boolean that controls the import from layout viewer option for the LEF file. | [ 0/1 ] (Boolean) {0} |
| pexLayoutLibrary | The library name of the layout topcell to run PEX on; only valid for Cadence Virtuoso. | [] () {} |
| pexLayoutPaths | File path of the main input GDS/OASIS/DEF database. | [] () {} |
| pexLayoutPrimary | The name of the topcell to run PEX on. | [] () {} |
| pexLayoutSystem | Format of the input layout database. | [GDSII OASIS OPENACCESS] () {GDSII} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexLayoutView | The cell view name of the layout topcell to run PEX on; only valid for Cadence Virtuoso. | [] () {} |
| pexLeadingX | Boolean controlling whether to retain the leading X's on flattened instance names in DSPF and CALIBREVIEW formats. | [ 0/1 ] (Boolean) {0} |
| pexPdbInfo | Boolean which controls whether the -pdb_info switch of the PDB stage is enabled. | [ 0/1 ] (Boolean) {0} |
| pexPexBusDelim | Boolean controlling whether to use the BUS_DELIMITER variable in the DSPF or SPEF header. | [ 0/1 ] (Boolean) {0} |
| pexPexBusDelimValue | SPEF or DSPF output format that specifies an optional string to set the BUS_DELIMITER variable in the SPEF or DSPF header. | [] (string) {} |
| pexPexClocation | Boolean controlling whether capacitor locations are output. | [ 0/1 ] (Boolean) {0} |
| pexPexCmpMode | Boolean that controls whether to specify the type of CMP mode used to handle layer thickness variation during parasitic extraction. | [ 0/1 ] (Boolean) {0} |
| pexPexCornerNames | A space or comma-separated string of specified corner names. | [] () {} |
| pexPexCornerType | Selects the type of corner extraction. | [All Specify] () {All} |
| pexPexCSelect | Boolean that controls whether to keep coupling capacitors distinct from intrinsic capacitors. | [ 0/1 ] (Boolean) {0} |
| pexPEXCreateDRTFile | Boolean that controls whether a driver/ receiver file will be generated. | [ 0/1 ] (Boolean) {0} |
| pexPEXDRTFile | Name of the generated driver/receiver file. | [] () {driver.xl} |
| pexPEXDRTFileTags | Values specifying how to generate the driver/receiver file. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexEnableCorner | Boolean that controls whether the -corner option will be applied to formatter stage. | [ 0/1 ] (Boolean) {0} |
| pexPexEnableTICER | Boolean that controls whether a threshold for TICER reduction will be applied. | [ 0/1 ] (Boolean) {0} |
| pexPexExtractFloatingNets | Boolean that controls whether to specify extract floating nets. | [ 0/1 ] (Boolean) {0} |
| pexPexExtractFloatingNetsIgnoreText | Boolean that controls whether to specify ignore port text placed on floating nets. | [ 0/1 ] (Boolean) {0} |
| pexPexExtractFloatingNetsScale | Capacitance scaling value for GROUNDED extract floating nets. | [] (positive real) {1} |
| pexPexExtractFloatingNetsType | Type of extract floating nets. | [GROUNDED ALL REDUCE] () {GROUNDED} |
| pexPexExtractSelectedNets | Boolean that controls whether to extract parasitics only for specific nets. | [ 0/1 ] (Boolean) {0} |
| pexPexExtractTemperature | Boolean that controls whether to specify extract temperature. | [ 0/1 ] (Boolean) {0} |
| pexPexExtractTemperatureCircuit | Circuit temperature in degrees Celsius. | [] (float) {27} |
| pexPexExtractTemperatureNominal | Nominal circuit temperature in degrees Celsius. | [] (float) {} |
| pexPexExtractTemperatureRange | Three temperatures used in calculating TC1 and TC2. | [] (three floating-point numbers) {} |
| pexPexExtractType | Extraction type to perform:<br>T — transistor<br>G — gate<br>H — hierarchical<br>A — ADMS | [T G H A] () {T} |
| pexPexExtractionModeHybrid | Boolean controlling whether the xACT 3D/xRC hybrid node will be applied | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexFMTAllowGroundCoupling | Boolean that controls whether to ground coupling capacitors when writing the parasitic netlist. | [ 0/1 ] (Boolean) {0} |
| pexPexFMTOutputElem | Parasitics to put to the RC netlist:<br>all — all<br>c — capacitances<br>r — resistors | [all c r] () {all} |
| pexPexFormatAditRail | Boolean controlling whether to format the netlist for ADiT Rail | [ 0/1 ] (Boolean) {0} |
| pexPexFormatCalibreViewICellMap | Boolean controlling whether to use the iCellMap format for the CalibreView netlist. | [ 0/1 ] (Boolean) {0} |
| pexPexFormatCalibreViewWithNetGeometry | Boolean that controls whether to annotate net geometries into the extracted Calibre View netlist. | [ 0/1 ] (Boolean) {0} |
| pexPexFormatHsim | Boolean controlling whether to format netlist for HSIM. | [ 0/1 ] (Boolean) {0} |
| pexPexFormatPrimeTime | Boolean that controls whether to format the netlist for PrimeTime. | [ 0/1 ] (Boolean) {0} |
| pexPexFormatterMapFileCalibreView | Boolean which controls whether to suppress the automatically generated LPE values from the CalibreView netlist output. | [ 0/1 ] (Boolean) {0} |
| pexPexFormatterMapFile | File path of the input back annotation mapfile. | [] () {} |
| pexPexGroundLayer | Boolean that controls whether the ground layer name will be defined. | [ 0/1 ] (Boolean) {0} |
| pexPexGroundLayerStray | Boolean controlling whether to eliminate all netlist output of intrinsic capacitance to substrate for the specified ground region. | [ 0/1 ] (Boolean) {0} |
| pexPexGroundLayerValue | Names of the ground layers. | [] () {} |
| pexPexGroundName | Boolean that controls whether the ground node name will be defined. | [ 0/1 ] (Boolean) {0} |
| pexPexGroundNameValue | Name of the ground node. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPEXGroundNetNameType | Type of net name for PEX ground nets. | [LAYOUT SOURCE] () {LAYOUT} |
| pexPEXGroundNetNames | Name(s) of power nets for PEX operations. | [] () {} |
| pexPexIncrementalPDB | Boolean that controls whether the PDB is generated incrementally. | [ 0/1 ] (Boolean) {0} |
| pexPexInductance | Type of inductance to extract:<br>N — none<br>L — self inductance<br>LM — self and mutual inductance | [N L LM] () {N} |
| pexPexLayerMap | Boolean controlling whether to report parasitic resistor layers as comments for the EM/IR analysis flow. | [ 0/1 ] (Boolean) {0} |
| pexPexLeadingX | Boolean that controls whether to retain leading Xs on flattened instance names in DSPF and CALIBREVIEW formats. | [ 0/1 ] (Boolean) {0} |
| pexPexLocation | Boolean that controls whether intentional device locations will be output. | [ 0/1 ] (Boolean) {0} |
| pexPexMapNames | Boolean controlling whether to generate name map for SPEF. | [ 0/1 ] (Boolean) {0} |
| pexPexMultiTxtPort | Boolean controlling whether multiple net ports from text ports with the same name will be output. | [ 0/1 ] (Boolean) {0} |
| pexPexNetlistCreateSmashedDeviceNames | Boolean that controls whether to create new instance names for smashed devices in DSPF and SPEF formats. | [ 1/0 ] (Boolean) {1} |
| pexPexNetlistDeviceResistanceModel | Boolean that controls the model used for parasitic resistance calculation.<br>**0** — (default) Use the default model (SPICE).<br>**1** — Use the model specified by the pexPexNetlistDeviceResistanceModelValue parameter. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexNetlistDeviceResistanceModelValue | Specifies the model for parasitic resistance calculation if pexPexNetlistDeviceResistanceModel is set to 1.<br>• **SPICE** — Use the internal calculations from the HSPICE model<br>• **CALCULATE** — Use measured W and L as calculated by a DEVICE specification statement in the rule file. | [ SPICE CALCULATE ] () {SPICE} |
| pexPexNetlistFile | File path of the generated netlist file. | [] () {} |
| pexPexNetlistFormat | Format of the output netlist. | [ELDO DSPF SPEF CALIBREVIEW HSPICE SPECTRE] () {ELDO} |
| pexPexNetlistGlobalNets | Boolean that controls whether to remove named nets from a subcircuit and place them in a global name definition. The nets are listed in the parameter pexPexNetlistGlobalNetsValue. | [ 0/1 ] (Boolean) {0} |
| pexPexNetlistGlobalNetsValue | A list of named nets that are removed from a subcircuit and placed in a global name definition when pexPexNetlistGlobalNets is set to 1. | [] () {} |
| pexPexNetlistMutualResistance | Boolean that controls whether the Calibre xL tool uses current-controlled current sources to model mutual resistance. | [ 1/0 ] (Boolean) {1} |
| pexPexNetlistNameSource | Type of names to use for nets and instances in the generated netlist. | [SOURCENAMES LAYOUTNAMES SOURCEBASED SCHEMATICONLY ] () {SOURCENAMES} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexNetlistNetmodelFile | The netmodel filename used in the SVRF command “PEX NETLIST SELECT file” when the GUI option for pexPexNetlistSelectedNets is true. | [] () {} |
| pexPexNetlistNoxrefNetNames | Boolean that controls whether to keep parasitic net models marked with _noxref from netlists. | [ 1/0 ] (Boolean) {1} |
| pexPexNetlistSchematicOnlyFileName | Filename for name source as SCHEMATICONLY or SOURCENAMES. | [] () {} |
| pexPexNetlistSelectedNets | Boolean controlling whether to netlist parasitics using the netmodel file specified with pexPexNetlistNetmodelFile. Connected to GUI option “Select netlist model for specific nets” | [ 0/1 ] (Boolean) {0} |
| pexPexNetlistType | Type of parasitics to extract. | [RCC RC R C NoRC] () {RCC} |
| pexPexNetlistUpperCaseKeywords | Boolean that controls whether to upper case element names, HSPICE and DSPF keywords in an output netlist. | [ 0/1 ] (Boolean) {0} |
| pexPexNetlistVirtualConnect | Boolean that controls whether to leave disjoint net model fragments unconnected. | [ 0/1 ] (Boolean) {0} |
| pexPexNoInstanceSection | Boolean controlling whether to eliminate instance section in DSPF. | [ 0/1 ] (Boolean) {0} |
| pexPexNoInstanceX | Boolean controlling whether to filter out the Xs in HSPICE, DSPF, CALIBREVIEW, SPEF and Spectre formats. | [ 0/1 ] (Boolean) {0} |
| pexPexNoTopSubCkt | Boolean controlling whether to suppress top-level subcircuit in DSPF. | [ 0/1 ] (Boolean) {0} |
| pexPexP2PInputFile | Filename of the PEX point-to-point input file. | [] () {} |
| pexPexP2POutputFile | Filename of the PEX point to point output file. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexP2PSamePort | Boolean controlling whether or not to short ports with the same net name. | [ 0/1 ] (Boolean) {0} |
| pexPexP2PSamePortValue | Specifies how resistance for shorted ports is reported. | [SHORT OPEN] () {SHORT} |
| pexPexPinNameSep | Boolean controlling whether to redefine pin to subcircuit delimiter. | [ 0/1 ] (Boolean) {0} |
| pexPexPinNameSepValue | Pin to subcircuit delimiter. | [] (string) {_} |
| pexPexPinOrder | Boolean that controls whether to specify the netlist pin order. | [ 0/1 ] (Boolean) {0} |
| pexPexPinOrderAllPins | Boolean controlling whether to add the ALLPINS keyword to the PEX Pin Order statement. | [ 0/1 ] (Boolean) {0} |
| pexPexPinOrderFile | File path to pin order file. | [] () {} |
| pexPexPinOrderValue | Type of pin order. | [LAYOUT SOURCE FILE] () {LAYOUT} |
| pexPEXPowerNetNameType | Type of net name for PEX power nets. | [LAYOUT SOURCE] () {LAYOUT} |
| pexPEXPowerNetNames | Name(s) of power nets for PEX operations. | [] () {} |
| pexPexPrune | Boolean controlling whether to remove devices not attached to selected nets. | [ 0/1 ] (Boolean) {0} |
| pexPexRactual | Boolean controlling whether actual parasitic resistor width will be output. | [ 0/1 ] (Boolean) {0} |
| pexPexRarea | Boolean controlling whether to report via/contact area. | [ 0/1 ] (Boolean) {0} |
| pexPexRecursiveExcludeNets | Boolean that controls whether to exclude specific recursive nets. | [ 0/1 ] (Boolean) {0} |
| pexPexRecursiveExcludeNets Value | Names of recursive nets to specifically exclude. | [] () {} |
| pexPexRecursiveIncludeNets | Boolean that controls whether to include specific recursive nets. | [ 0/1 ] (Boolean) {0} |
| pexPexRecursiveIncludeNets Value | Names of recursive nets to specifically include. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexReduceAnalog | Boolean that controls whether analog reduction is enabled. | [ 0/1 ] (Boolean) {0} |
| pexPexReduceAnalogDelay | Timing delay threshold used for parasitic reduction. The delay units are seconds. The default is 0.5e-12 seconds (0.5 picoseconds). | [] (positive real) {} |
| pexPexReduceAnalogNoise | Error threshold ratio, a unitless floating number. The default is 0.01, or 1%. | [] (real between 0 and 1) {} |
| pexPexReduceCC | Boolean that controls whether CC reduction will be enabled. | [ 0/1 ] (Boolean) {0} |
| pexPexReduceCCAbsolute | Threshold for absolute CC reduction. | [] (positive real) {} |
| pexPexReduceCCRatio | Threshold ratio for CC reduction. | [0-1] (real) {} |
| pexPexReduceCCScale | Scaling factor applied to reduced capacitances. | [] (positive real) {1} |
| pexPexReduceConfigureVRs | Integer that controls how to configure PEX REDUCE VIA RESISTANCE statements:<br>-1 — Turn off via reduction<br>0 — Use rule file settings (default)<br>1 — Override via reduction | [-1 0 1] (integer) {0} |
| pexPexReduceMinCap | Boolean that controls whether to combine or remove extracted capacitors based on a user-defined threshold value. | [ 0/1 ] (Boolean) {0} |
| pexPexReduceMinCapCombine | Threshold for combine extracted capacitors. | [>0] (real) {} |
| pexPexReduceMinCapNoRemove | Boolean that controls whether to turn off the default reduction of small coupling capacitance between pairs of nets that are less than 2e-19f. | [ 0/1 ] (Boolean) [0] |
| pexPexReduceMinCapRemove | Threshold for remove extracted capacitors. | [>0] (real) {} |
| pexPexReduceMinMutual | Boolean that controls whether a threshold for mutual inductance reduction is applied. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexReduceMinMutualThreshold | Threshold for mutual inductance reduction, given as a K value, where 0 < K < 1. | [0-1] (positive real) {} |
| pexPexReduceMinRes | Boolean that controls whether to combine or short extracted resistors based on a user-defined threshold value. | [ 0/1 ] (Boolean) {0} |
| pexPexReduceMinResCombine | Threshold for combine extracted resistors. | [>=0] (real) {0} |
| pexPexReduceMinResLayerWise | Boolean that controls whether to specify the reduction of parasitic resistors for each layer, and not across different layers. | [ 0/1 ] (Boolean) {0} |
| pexPexReduceMinResShort | Threshold for short extracted resistors. | [>=0] (real) {0} |
| pexPexReduceMinResYesNo | Specifies whether to turn on or off PEX REDUCE MINRES. | [ YES NO ] ({YES} |
| pexPexReduceParallelMos | Boolean that controls whether mosfets that are wired in parallel are reduced. | [ 0/1 ] (Boolean) {0} |
| pexPexReduceParallelMosRThreshold | Resistance threshold between parallel device pins | [>=0] (real) {0} |
| pexPexReduceParallelMosSameWL | Boolean that controls whether to merge parallel devices with the same width and length. | [ 0/1 ] (Boolean) {0} |
| pexPexReduceParallelMosTypes | Specifies the types of parallel mosfets to combine. Specify ALL to combine all mosfet devices or provide a list of mosfet device type names. | [ALL \| mos_type …] () {ALL} |
| pexPexReduceViaResistances | List of PEX REDUCE VIA RESISTANCE Statement values. | [] () {} |
| pexPexReport | Boolean that controls whether a PEX Report file is generated. | [ 0/1 ] (Boolean) {0} |
| pexPexReportCC | Boolean controlling whether the PEX CC Report file is generated. | [ 0/1 ] (Boolean) {0} |
| pexPexReportCCFile | Filename of the PEX CC report output file. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexReportCCNameSource | Specifies using net names from schematic or layout for coupling capacitance report. | [SOURCE LAYOUT] () {SOURCE} |
| pexPexReportCCNumber | Maximum number of coupled capacitors to include in the report. | [] (positive integer) {} |
| pexPexReportCCSplitNet | Boolean controlling whether the PEX CC Report SPLIT_NET is selected. | [ 0/1 ] (Boolean) {0} |
| pexPexReportCCThreshold | Threshold value for coupling capacitance report. | [] (positive value) {} |
| pexPexReportFile | Filename of the PEX Report file. | [] () {} |
| pexPexReportNameSource | Type of names to use for nets and instances in the generated PEX report. | [SOURCENAMES LAYOUTNAMES] () {SOURCENAMES} |
| pexPexReportNetSummary | Boolean controlling whether the PEX NetSummary Report file is generated. | [ 0/1 ] (Boolean) {0} |
| pexPexReportNetSummaryCell | Cell name for PEX REPORT NETSUMMARY, only nets within this cell will be reported. | [] () {} |
| pexPexReportNetSummaryColumns | Output control for PEX REPORT NETSUMMARY. | [ BASIC ADVANCED ] () {BASIC} |
| pexPexReportNetSummaryFile | Filename of the PEX REPORT NETSUMMARY output file. | [] () {} |
| pexPexReportNetSummaryNetFile | Filename of the PEX REPORT NETSUMMARY containing net names. | [] () {} |
| pexPexReportNetSummaryNetNames | Net names to be reported in PEX REPORT NETSUMMARY. | [] () {} |
| pexPexReportNetSummaryQuery | Specifies whether the nets' capacitances are itemized by sections or only reported for the whole net. | [ DETAIL SUMMARY ] () {DETAIL} |
| pexPexReportNetSummaryScale | Floating-point number that is multiplied with the capacitive values reported. | [] (positive real) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexReportNetSummaryScope | The scope of PEX NetSummary Report. | [ FULL LOCAL ALL ] () {FULL} |
| pexPexReportNetSummarySystem | Specify the net names based on LAYOUT, SOURCE, ALL or NETFILE. | [ ALL LAYOUT SOURCE NETFILE ] () {ALL} |
| pexPexReportP2P | Boolean that controls whether the PEX point to point report file is generated. | [ 0/1 ] (Boolean) {0} |
| pexPexReportP2PCalibreView | Boolean that controls whether to add the point to point report file to the Calibre View netlist. | [ 0/1 ] (Boolean) {0} |
| pexPexReportP2PUseDbu | Use database unit for COORD location parameter in P2P input file. | [ 0/1 ] (Boolean) {0} |
| pexPexRlayer | Boolean that controls whether resistor layers will be output. | [ 0/1 ] (Boolean) {0} |
| pexPexRlength | Boolean that controls whether resistor lengths will be output. | [ 0/1 ] (Boolean) {0} |
| pexPexRlocation | Boolean that controls whether resistor locations will be output. | [ 0/1 ] (Boolean) {0} |
| pexPexRscale | Boolean controlling whether to apply a scaling factor to parasitic resistor properties. | [ 0/1 ] (Boolean) {0} |
| pexPexRscaleValue | Scaling factor to parasitic resistor properties. | [] (positive real) {1} |
| pexPexRthickness | Boolean controlling whether resistor thickness are output. | [ 0/1 ] (Boolean) {0} |
| pexPexRviaCount | Boolean controlling whether to report the number of vias/contacts represented by the netlisted via resistor value. | [ 0/1 ] (Boolean) {0} |
| pexPexRwidth | Boolean that controls whether resistor widths are output. | [ 0/1 ] (Boolean) {0} |
| pexPexRwidthMinimum | Boolean controlling whether minimum resistor widths for non-rectangular shapes will be output. | [ 0/1 ] (Boolean) {0 |
| pexPexSeparator | Boolean that controls whether the hierarchy separator character are specified. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexSeparatorValue | Hierarchy separator character. | [] (character) {/} |
| pexPexShortPinNames | Boolean controlling whether to use short pin names. | [ 0/1 ] (Boolean) {0} |
| pexPexSingleFile | Boolean controlling whether to combine transistor-level SPICE data. | [ 0/1 ] (Boolean) {0} |
| pexPexSizeMultiplier | Scaling factor for length units in the parasitic netlist. | [] (positive real) {1} |
| pexPexSpecifyStringQuotes | Boolean controlling whether Calibre Interactive includes the STRINGQUOTES parameter in the PEX Netlist statement written to the control file. | [ 0/1 ] (Boolean) {0} |
| pexPexStringQuotes | Boolean controlling whether or not to enclose string properties in quotes in the netlist. (The yes/no selection for the STRINGQUOTES parameter). | [ 1/0 ] (Boolean) {1} |
| pexPexTCoeff | Boolean controlling whether to output temperature coefficients. | [ 0/1 ] (Boolean) {0} |
| pexPexTICERMaxDeg | Maximum degree for TICER reduction. | [2-12] (positive integer) {} |
| pexPexTICERPortMerge | Port merge range for TICER reduction. | [] (positive real) {} |
| pexPexTICERThreshold | Threshold for TICER reduction. | [] (positive real) {} |
| pexPexTopLevelExcludeNets | Boolean that controls whether to exclude specific top level nets. | [ 0/1 ] (Boolean) {0} |
| pexPexTopLevelExcludeNets Value | Names of top level nets to specifically exclude. | [] () {} |
| pexPexTopLevelIncludeNets | Boolean that controls whether to include specific top level nets. | [ 0/1 ] (Boolean) {0} |
| pexPexTopLevelIncludeNetsV alue | Names of top level nets to specifically include. | [] () {} |
| pexPexValue | Boolean that controls whether intentional device values will be output. | [ 0/1 ] (Boolean) {0} |
| pexPexViewPexNetlist | Boolean that controls whether the netlist file is displayed after the run. | [ 1/0 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexPexViewPexReport | Boolean that controls whether the PEX Report file is displayed after the run. | [ 0/1 ] (Boolean) {0} |
| pexPostLVSTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute after LVS is run and before the parasitic extraction phase is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| pexPostTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute after Calibre PEX is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| pexPowerNames | Names of power net(s) for LVS operations. | [] () {} |
| pexPreTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute before Calibre PEX is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| pexProbeList | List of probe point column lists; each column list contains settings for rows in the probe table.<br>{SELECTED {0/1 …}}<br>{NAME {probe_name …}}<br>{CELL {cell_name …}}<br>{LOCATION {{x y} …}}<br>{LAYER {layer_name …}} | [] (see description) {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexProbePort | Boolean that controls whether to replace probes on the net represented as subnodes (*\|S) with ports (*\|P) when extracting a DSPF netlist. | [ 0/1 ] (Boolean) {0} |
| pexProbeProtect | Boolean that controls whether to prevent probe points from being removed when a probe point is at the same node as a pin or port. | [ 0/1 ] (Boolean) {0} |
| pexRecognizeGates | Specifies the type of gates for the extractor to recognize. | [ALL SIMPLE NONE] () {ALL} |
| pexRecognizeGatesMixSubtypes | Boolean that controls whether subtype mixing in gate extraction occurs. | [ 0/1 ] (Boolean) {0} |
| pexReduceParallelMOS | Boolean that controls whether to perform parallel MOS reduction.<br><br>If pexReduceSplitGates is set to 1, then pexReduceParallelMOS is automatically set to 1. To disable pexReduceParallelMOS, both pexReduceParallelMOS and pexReduceSplitGates must be set to 0. See "Setting Split Gate and Parallel MOS Reduction Options" on page 271. | [ 1/0 ] (Boolean) {1} |
| pexReduceSameOrderSplitGates | Boolean that controls whether to reduce split gates only when the input order is the same. | [ 0/1 ] (Boolean) {0} |
| pexReduceSemiSplitGates | Boolean that controls whether to reduce semi-split gates. | [ 0/1 ] (Boolean) {0} |
| pexReduceSeriesParallelSplitGates | Boolean that controls whether to perform series-parallel split gate reduction. | [ 0/1 ] (Boolean) {0} |
| pexReduceSplitGates | Boolean that controls whether to reduce split gates. Also see pexReduceParallelMOS. | [ 1/0 ] (Boolean) {1} |
| pexReduceSplitGatesMixTypes | Boolean that controls whether to allow mixed types during split gate reduction. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexReduceSplitGatesWithinTolerance | Boolean that controls whether to reduce split gates only when within tolerance. | [ 0/1 ] (Boolean) {0} |
| pexReportFile | Filename of the LVS report file. | [] () {lvs.report} |
| pexReportOptions | Specifies LVS REPORT OPTIONS settings. Any combination of the allowed options may be used. | [A AV B BV BX C CV D E E1 F FX G H I LPE MC N NCA NE NOK O P R RA RD S SP SPE V W X XR] (any combination of allowed options) {} |
| pexResistanceParameters | Boolean controlling whether the resistance parameters will be defined. | [ 0/1 ] (Boolean) {0} |
| pexResistanceParametersValue | List of options for resistance parameters. | [] () {} |
| pexRulesFile | File path of the rule file. | [] () {rules} |
| pexRunDir | Path of the run directory. | [] () {.} |
| pexRunFMTStep | Boolean that controls whether the FMT step will be run. | [ 1/0 ] (Boolean) {1} |
| pexRunHierLVS | Boolean that controls whether the hierarchical version of LVS will be run. | [ 1/0 ] (Boolean) {1} |
| pexRunPDBStep | Boolean that controls whether the PDB step will be run. | [ 1/0 ] (Boolean) {1} |
| pexRunPHDBStep | Boolean that controls whether the PHDB step will be run. | [ 1/0 ] (Boolean) {1} |
| pexRunsetInfo | One line of text for the runset info field. See “Runset Type and Runset Info” on page 45. | [] () {} |
| pexRunsetType | One line of text for the runset type field. See “Runset Type and Runset Info” on page 45. | [] () {} |
| pexShortEquivalentNodes | Boolean that controls whether to short equipotential nodes together during split gate reduction. | [ 0/1 ] (Boolean) {0} |
| pexShowPEXSteps | Boolean that controls whether to show checkbuttons for individual PEX steps. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexSourceDeviceFilterOptions | Specifies LVS Filter Unused Option for the SOURCE setting. Any combination of allowed options may be used. | [AB AC AD AE AF AH AG AI BC D E EG F FF FY G GF GY H I IH INV J K KF L M MG N NP O P Q RB RC RD RE RF RG S T U V YB YC ZB ZC ZD ZE] (any combination of allowed options) {} |
| pexSourceGetFromViewer | Boolean that controls the "Export from schematic viewer" setting on the Inputs pane in the **Netlist** tab. This option has dependencies on pexTemplate_IS; see "Export from Viewer Preferences" on page 152. | [ 0/1 ] (Boolean) {0} |
| pexSourceLibrary | The library name of the source topcell to run PEX on; only valid for Cadence Virtuoso. | [] () {} |
| pexSourcePath | File paths of the input source netlist files. | [] () {} |
| pexSourcePrimary | The name of the topcell in the source netlist. | [] () {} |
| pexSourceSystem | Format of the input source netlist file. | [SPICE VERILOG MIXED] () {SPICE} |
| pexSourceView | The cell view name of the source topcell to run PEX on; only valid for Cadence Virtuoso. | [] () {} |
| pexStartRVE | Boolean that controls whether to start Calibre RVE after the run completes. | [ 0/1 ] (Boolean) {0} |
| pexStopAfterLVSMismatch | Boolean that controls whether to stop the XRC run if the LVS step fails. | [ 0/1 ] (Boolean) {0} |
| pexSVDBcci | Boolean that controls whether Calibre Connectivity Interface data is generated. | [ 0/1 ] (Boolean) {0} |
| pexSVDBDir | Pathname to the svdb directory. The path is relative to the run directory. | [] () {svdb} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexSVDBnopinloc | Boolean that controls whether pin location generation is suppressed if Calibre Connectivity Interface (CCI) is enabled. | [ 0/1 ] (Boolean) {0} |
| pexSVDBquery | Boolean that controls whether cross-reference data is generated for Calibre RVE. | [ 1/0 ] (Boolean) {1} |
| pexSVDBtxref | Boolean that controls whether ASCII cross-reference files are generated. | [ 0/1 ] (Boolean) {0} |
| pexSVRFCmds | Additional SVRF commands to be inserted into the control file. | [] () {} |
| pexTemplate_HF[1] | Template for the hcells filename. | [] () {} |
| pexTemplate_IS[1] | Controls the template preference "Export netlist from schematic viewer" setting in the Setup Preferences dialog box in the **Templates** > **Inputs** sub-tab.<br>1 — Yes, export from viewer<br>0 — Do not export from viewer<br>-1 — Use runset value.<br>See "Export from Viewer Preferences" on page 152. | [1 0 -1] () {-1} |
| pexTemplate_LR[1] | The LVS report filename template. | [] () {%l.lvs.report} |
| pexTemplate_LS[1] | Template for the source library name for database formats. | [] () {%S} |
| pexTemplate_PN[1] | Template for the extracted netlist file. | [] () {%l.pex.netlist} |
| pexTemplate_PR[1] | The PEX report filename template. | [] () {%l.pex.report} |
| pexTemplate_SP[1] | The source path filename template. | [] () {%s.src.net} |
| pexTemplate_SV[1] | Template for the SVDB database directory name. | [] () {svdb} |
| pexTemplate_VS[1] | Template for the source view name for database formats. | [] () {%W} |
| pexTemplate_XF[1] | Template for the Xcells filename. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-5. PEX Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pexTVFCmds | Additional TVF commands to be inserted into the control file. | [] () {} |
| pexUseCBforLVS | Boolean that controls whether the Calibre CB license will be used for the Calibre nmLVS run. | [ 0/1 ] (Boolean) {0} |
| pexUseCBforPEX | Boolean that controls whether the Calibre CB license will be used for the PEX run. | [ 0/1 ] (Boolean) {0} |
| pexUseCBforRVE | Boolean that controls whether the Calibre CB license will be used for Calibre RVE. | [ 0/1 ] (Boolean) {0} |
| pexUseHCells | Boolean that controls whether to use an hcells file for LVS. | [ 0/1 ] (Boolean) {0} |
| pexViewLVSReport | Boolean that controls whether the LVS Report file is displayed after the run. | [ 0/1 ] (Boolean) {0} |
| pexVlogSourcePath | File paths of the input verilog source netlist files in the mixed mode. | [] () {} |
| pexxCellsFile | Pathname of xcells file to use during PEX. | [] () {hcells} |

1. The *Template_* runset options set templates for file naming.

**Note**

See "Templates for File Naming" on page 148 and "Runset Options for Template Parameters" on page 960 for more information on file naming templates.

# Common Runset Options

The common runset options are used by all Calibre Interactive applications.

Also see "Templates for File Naming" on page 148 and "Runset Options for Template Parameters" on page 960 for more information on file naming templates.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table C-6. Common Runset Options

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnApplyTemplates | Boolean that controls whether templates are applied when Calibre Interactive is started from a design tool. | [ 1/0 ] (Boolean) {1} |
| cmnBackupRDS | Boolean that controls whether to backup processing host data when using the Remote Data System. | [ 0/1 ] (Boolean) {0} |
| cmnBatchModeCompatibleControlFile | Boolean that controls whether the control file will be written out so that it may be used as is by command line Calibre runs. | [ 0/1 ] (Boolean) {1} |
| cmnCheckForModifiedRules | Boolean that controls the “Automatically load rules file” setting in the Setup Preferences dialog box in the **Misc** tab. | [ 0/1 ] (Boolean) {0} |
| cmnCloseRVEonExit | Boolean that controls the “Close RVE when Calibre Interactive closes” setting. | [ 1/0 ] (Boolean) {1} |
| cmnCloseRVEonRun | Boolean that controls the “Close RVE before running DRC/DFM/LVS/PEX” setting. | [ 1/0 ] (Boolean) {1} |
| cmnCompileOnly | Boolean that controls whether to compile the rule file without starting a Calibre run when the user clicks the **Run** button. | [ 0/1 ] (Boolean) {0} |
| cmnConfigureLVSBox | Boolean that controls whether to configure LVS Box statements. | [ 0/1 ] (Boolean) {0} |
| cmnConfigureTPs | Boolean that controls whether to configure Trace Property statements in the **Properties** tab on the Options pane. Available for Calibre nmLVS and Calibre PERC. | [ 0/1 ] (Boolean) {0} |
| cmnControlFile | Control file name. | [] () {} |
| cmnCustomFileName | Filename of the customization file. | [] () {} |
| cmnCustomFileOverrideValues | List of control/value pairs which override initial settings in the customization file. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnDisableCustomEdit | Boolean that controls the “Disable customization changes” setting. | [ 0/1 ] (Boolean) {0} |
| cmnDisableEditRecipes | Boolean that controls the “Disable check selection recipe editing” setting. | [ 0/1 ] (Boolean) {0} |
| cmnDisableEnvEdit | Boolean that controls the “Disable environment changes” setting. | [ 0/1 ] (Boolean) {0} |
| cmnDisableERCChanges | Boolean that controls the “Disable ERC changes” setting for LVS. | [ 0/1 ] (Boolean) {0} |
| cmnDisableFastXORMaxDiffCount | Boolean controlling the FastXOR “Max Diff” count control. The control is not displayed when this option is set to 1 (the default). | [ 0/1 ] (Boolean) {1} |
| cmnDisableFastXOROptions | Boolean controlling the FastXOR Options tab. | [ 0/1 ] (Boolean) {1} |
| cmnDisableFastXORRulesFileCustomName | Boolean controlling the FastXOR option to specify a custom rules file name for the generated rule file. | [ 0/1 ] (Boolean) {1} |
| cmnDisableOptions | Boolean that controls the “Disable DRC/DFM/LVS/PERC/PEX Options” setting. | [ 0/1 ] (Boolean) {0} |
| cmnDisableSelectChecks | Boolean that controls the “Disable check selection” setting. | [ 0/1 ] (Boolean) {0} |
| cmnDisableSignatureCreation | Boolean that controls the “Disable device signature creation” setting for LVS. | [ 0/1 ] (Boolean) {0} |
| cmnDisableTranscriptOutput | Boolean that controls whether to display output to transcript window. | [ 0/1 ] (Boolean) {0} |
| cmnDisableWaivers | Boolean that controls the “Disable Waivers tab” setting. | [ 0/1 ] (Boolean) {0} |
| cmnDisableWXRLayerMapAndBaseLayersCombinedUsage | Boolean that controls the “Disable Fast XOR layer map and base layers combined usage” on the **Setup** > **Preferences** > **Disable** tab for Calibre Interactive nmDRC. | [ 1/0 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnDontWaitForLicense | Integer that controls license queuing.<br>• 0 — Queue when license is unavailable<br>• 1 — Do not queue<br>• 2 — Retry (loop) until license is available | [ 0 1 2] (integer) {1} |
| cmnEnableEditingForTextViewer | Boolean that controls whether the text file viewer allows editing when it first opens a file. | [ 0/1 ] (Boolean) {0} |
| cmnEnableLicenseModes | Boolean that controls whether license arguments are specified on the Calibre command line. When disabled, no command line argument is used.<br>"Licensing Modes" checkbox on the Run Control pane, **Environment** > **Licensing** tab. | [ 1/0 ] (Boolean) {1} |
| cmnErrorFile | The name of the transcript errors and warnings file. | [] () {} |
| cmnFDIAbortOnPCellRead | Boolean that controls whether to abort upon encountering empty PCells when reading an OpenAccess library. (Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Read Options** category) | [ 1/0 ] (Boolean) {1} |
| cmnFDICellGDSDirectories | Cell GDS directory list used in database conversion on the Options pane, **Database** > **OpenAccess/ LEFDEF** tab, **Additional Files** category. | [] () {} |
| cmnFDICellGDSFiles | Cell GDS file list used in database conversion on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Additional Files** category. | [] () {} |
| cmnFDICellMapFile | Name of the cell map file on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Mapping Files** category. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnFDIDEFDirectories2 | Name of the directory or directories containing DEF files for LEF/DEF databases for the Layout2 input of a FastXOR run. | [] () {} |
| cmnFDIDEFDirectories | Name of the directory or directories containing DEF files for LEF/DEF databases. | [] () {} |
| cmnFDIDEFLayoutPath2 | Name of the DEF file or files for LEF/DEF databases for the Layout2 input of a FastXOR run. | [] () {} |
| cmnFDIDEFLayoutPath | Name of the DEF file or files for LEF/DEF databases. | [] () {} |
| cmnFDIInputExceptionFile | Name of the input exception file on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Mapping Files** category. | [] () {} |
| cmnFDILayerMapFile | Name of the layer map file. | [] () {} |
| cmnFDILayoutLibrary | Name of the layout library for OpenAccess databases. | [] () {} |
| cmnFDILayoutLibrary2 | Name of the reference input layout library for OpenAccess databases. | [] () {} |
| cmnFDILayoutLibraryPath | Pathname of the layout library for OpenAccess databases. | [] () {} |
| cmnFDILayoutLibraryPath2 | Pathname of the reference layout library for OpenAccess databases. | [] () {} |
| cmnFDILayoutView | Name of the layout view for OpenAccess databases | [] () {} |
| cmnFDILayoutView2 | Name of the reference input layout view for OpenAccess databases | [] () {} |
| cmnFDILEFDirectories2 | Name of directory or directories containing LEF files for LEF/DEF databases for the Layout2 input of a FastXOR run. | [] () {} |
| cmnFDILEFDirectories | Name of directory or directories containing LEF files for LEF/DEF databases. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnFDILEFLayoutPaths2 | Names of LEF files for LEF/DEF databases for the Layout2 input of a FastXOR run. | [] () {} |
| cmnFDILEFLayoutPaths | Names of LEF files for LEF/DEF databases. | [] () {} |
| cmnFDILEFTechFile2 | Name of the LEF technology file for LEF/DEF databases for the Layout2 input of a FastXOR run. | [] () {} |
| cmnFDILEFTechFile | Name of the LEF technology file for LEF/DEF databases. | [] () {} |
| cmnFDINetPropFile | Name of the net properties file on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Mapping Files** category. | [] () {} |
| cmnFDIObjectMapFile | Name of the object map file on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Mapping Files** category. | [] () {} |
| cmnFDIReadBlockages | Boolean that controls the “Read: Blockages” setting on the Options pane, **Database** > **OpenAccess/ LEFDEF** tab, **Read Options** category. | [ 1/0 ] (Boolean) {0} |
| cmnFDIReadInstNameAsProp | Value of the property name used for reading instance names as properties. | [] () {} |
| cmnFDIReadInstPinShapes | Boolean that controls the “Read: Instance Pins” setting on the Options pane, **Database** > **OpenAccess/ LEFDEF** tab, **Read Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIReadNetTypeAsProp | Value of the property name used for reading net types as properties. | [] () {} |
| cmnFDIReadNetsAsPropVal | Value of the property name used for reading nets as properties. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnFDIReadNetsAsText | Controls the "Read net names as: All Cells/Top Cells" setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Read Options** category. | [ALL TOP] () {ALL} |
| cmnFDIReadPinShape | Boolean that controls the "Read: Pins" setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Read Options** category. | [ 1/0 ] (Boolean) {1} |
| cmnFDIReadPinsAsPropVal | Value of the property name used for reading nets as properties. | [] () {} |
| cmnFDIReadText | Boolean that controls the "Read: Texts" setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Read Options** category. | [ 1/0 ] (Boolean) {1} |
| cmnFDISetEnableOaExpPcell | Boolean that controls the "Set CDS_ENABLE_EXP_PCELL …" setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDISetOaExpAllParams | Boolean that controls the "Set CDS_EXP_ALL_PARAMS …" setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDISetOaExpPcellDir | Boolean that controls the "Set CDS_EXP_PCELL_DIR …" setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDISetOaHome | Boolean that controls the "Set OA_HOME when CDS_INST_ROOT or CDS_INST_DIR is set" setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, in the **Other Options** category. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnFDISetOaLdl | Boolean that controls the "Set CALIBRE_READDB_LD_LIBRARY_PATH when CDS_INST_ROOT or CDS_INST_DIR is set" setting in the on the Options pane, **Database** > **OpenAccess/LEFDEF** tab in the **Other Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDISetOaPath | Boolean that controls the "Set PATH when CDS_INST_ROOT or CDS_INST_DIR is set" setting in the on the Options pane, **Database** > **OpenAccess/LEFDEF** tab in the **Other Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseCellGDS | Boolean that controls the "Use Cell GDS Files" setting in the on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Additional Files** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseCellMap | Boolean that controls the "Use Cell Map File" setting in the on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Mapping Files** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseInputException | Boolean that controls the "Use Input Exception File" setting on the on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Mapping Files** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseLayerMap | Boolean that controls the "Use Layer Map File" setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Mapping Files** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseLayerNames | Boolean that controls the "Write LayerNames File" setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnFDIUseLogFile | Boolean that controls the “Write Log File” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseMapInfo | Boolean that controls the “Write Map Info File” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseNetProp | Boolean that controls the “Use Net Properties File” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Mapping Files** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseObjectMap | Boolean that controls the “Use Object Map File” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Mapping Files** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseReadInstName | Boolean that controls the “Read Instance Name As” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Read Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseReadNetType | Boolean that controls the “Read Net Type As” setting on the Options pane, **Database** > **OpenAccess/ LEFDEF** tab, **Read Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseReadNetsAsProperty | Boolean that controls the “Read Net Names As: Property” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Read Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseReadNetsAsText | Boolean that controls the “Read Net Names As: Text” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Read Options** category. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnFDIUseReadPinsAsProperty | Boolean that controls the “Read Pin Names As: Property” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Read Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseReadPinsAsText | Boolean that controls the “Read Pin Names As: Text” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Read Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIUseViewListOA | Boolean that controls the “OpenAccess View List” setting on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnFDIViewListOA | View list for OpenAccess databases on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [] () {} |
| cmnFDIWriteLayerNamesFile | Name of the Layer Names file on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [] () {} |
| cmnFDIWriteLogFile | Name of the log file on the Options pane, **Database** > **OpenAccess/ LEFDEF** tab, **Other Options** category. | [] () {} |
| cmnFDIWriteMapInfoFile | Name of the Map Info file on the Options pane, **Database** > **OpenAccess/LEFDEF** tab, **Other Options** category. | [] () {} |
| cmnGridInteractiveSub | Boolean controlling the “Interactive Job Submission” setting for Sun Grid Engine jobs. | [ 0/1 ] (Boolean) {1} |
| cmnGridMasterTbl | Settings for Calibre MTflex Sun Grid Engine runs for the primary process. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnGridMasterTblViewDetails | Boolean that controls the “Show Details” setting for Sun Grid Engine remote hosts. | [ 0/1 ] (Boolean) {0} |
| cmnGridMtTbl | Settings for multithreaded Sun Grid Engine runs. | [] () {} |
| cmnGridSlaveTbl | Settings for Calibre MTflex Sun Grid Engine runs for the remote processes. | [] () {} |
| cmnGridStTbl | Settings for single-threaded Sun Grid Engine runs. | [] () {} |
| cmnHDBFlex | Boolean controlling the “Connect to Remote Hosts” setting for Distributed (MTflex) runs. | [ 0/1] (Boolean) {0} |
| cmnHDFBFlexMode | The mode determining whether to connect to remote stages at a later stage of or in parallel with HDB construction. | [LATERHDB ACQUIREHDB] () {LATERHDB} |
| cmnHostName | The name of the host running the layout viewers. | [] () {localhost} |
| cmnHostPort | The port number of the layout viewer. | [] (positive integer) {9189} |
| cmnIgnoreExportInBatch | Boolean that controls whether to ignore or honor export settings when in batch mode. By default, export settings are ignored when in batch mode. | [1/0] (Boolean) {1} |
| cmnIncludeRulesFilesAfterMain | Boolean that controls whether included rules files are included in the control file after the main rules file. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnLayoutServerType | Type of layout viewer. | [Mentor:Calibre Mentor:ICgraph Mentor:Sierra Mentor:L-Edit Cadence:Virtuoso Cadence:Encounter Synopsys:Laker Synopsys:ICCompiler Synopsys:ICCompilerII Synopsys:CustomCompiler Other:Generic None] () {} |
| cmnLayoutTextFile | Path to the file containing layout text statements. | [] () {} |
| cmnLayoutTmpFile | Name of the temporary file used in layout viewer communication. | [] () {query_results} |
| cmnLicenseLoopIntervalTime | Time in seconds to wait between loop iterations for a license. | [] (non-negative integer) {60} |
| cmnLicenseLoopTime | Time in minutes to loop for a license | [] (non-negative integer) {180} |
| cmnLicenseReleaseTime | Number of hours of inactivity after which to release the license. | [] (positive real) {3} |
| cmnLicenseWaitTime | Time in minutes to queue for a license (maximum 45000). | [max 45000] (non-negative integer) {0} |
| cmnLsfHostName | The name of the LSF submission host. | [] () {localhost} |
| cmnLSFInteractiveSub | Boolean controlling the “Interactive Job Submission” setting for LSF jobs. | [ 0/1 ] (Boolean) {1} |
| cmnLSFMasterTbl | Settings for Calibre MTflex LSF runs for the primary process. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnLSFMasterTblViewDetails | Boolean that controls the “Show Details” setting for LSF remote hosts. | [ 0/1 ] (Boolean) {0} |
| cmnLSFMtTbl | Settings for multithreaded LSF runs. | [] () {} |
| cmnLSFSlaveTbl | Settings for Calibre MTflex LSF runs for the remote processes. | [] () {} |
| cmnLSFStTbl | Settings for single-threaded LSF runs. | [] () {} |
| cmnLVSBlackBoxPortDepthNumber | Hierarchical level from which ports are recognized in LVS Box BLACK cells. | [] (non-negative integer) {0} |
| cmnLVSBlackBoxPortDepth | Controls the hierarchical levels from which ports are recognized in LVS Box BLACK cells. | [PRIMARY ALL NUMBER] () {PRIMARY} |
| cmnLVSBlackBoxPorts | List of LVS Black Box Port Statement values. | [] () {} |
| cmnLVSBoxes | List of LVS Box Statement values. | [] () {} |
| cmnMTFlexHostFile | Calibre MTflex remote host filename. This option is used when “Run Calibre on” is set to Local Host or Remote Host and “Remote Hosts” is set to Specify File. (cmnRunHostType set to localhost (default) or remotehost and cmnUseMTFlexHostFile set to 1). | [] () {} |
| cmnNumTurbo | Number of threads to start for Calibre operations. | [All <number>] (All or positive integer) {All} |
| cmnNumTurboLitho | Number of threads to start for LITHO operations. | [All <number>] (All or positive integer) {All} |
| cmnOverrideLayoutPrecision | Boolean that determines whether the Precision setting from the control file will override the layout precision setting without producing a fatal Calibre error. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnPrecision | Database Precision setting. | [] (positive real) {1000} |
| cmnProcessLocalLinks | Boolean that controls whether links are created in the directory of the runset. | [ 1/0 ] (Boolean) {1} |
| cmnPromptSaveRunset | Boolean that controls whether a dialog is shown prompting the user to save modified runsets. | [ 1/0 ] (Boolean) {1} |
| cmnRDSCount | Number of Remote Data Servers allocated to each remote host.<br>The default value is empty, meaning that Calibre allocates the Remote Data Servers. It is strongly recommended that you use this default value. | [] (positive integer) {} |
| cmnReadIOFromRules | Boolean that controls whether input and output settings are read from the rule file when the rule file is loaded. | [ 1/0 ] (Boolean) {1} |
| cmnReleaseLicense | Boolean that controls the timed license release feature. | [ 0/1 ] (Boolean) {0} |
| cmnRemoteClusterQueueCmd | Name of the program and its arguments to use to launch Calibre on a remote cluster. | [] () {} |
| cmnRemoteClusterSynch | Boolean that controls whether the remote cluster is synchronous. | [ 0/1 ] (Boolean) {0} |
| cmnRemoteHostName | Name of the remote host; used when cmnRunHostType is set to remotehost. | [] () {} |
| cmnRemoteLICENSEFILEName | Environment variable to use for licensing on the remote host for remote runs. | [] () {LM_LICENSE_FILE} |
| cmnRemoteLICENSEFILEUseCurrent | Boolean that controls whether to use the current licensing environment variable and value for remote runs. | [ 1/0 ] (Boolean) {1} |
| cmnRemoteLICENSEFILEValue | Value of the remote licensing environment variable. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnRemoteMGCHOME | Path of the remote MGC_HOME. | [] () {} |
| cmnRemoteMGCHOME_aoi | Path of the remote MGC_HOME on the AOI remote hosts.[1] | [] () {} |
| cmnRemoteMGCHOME_aoj | Path of the remote MGC_HOME on AOJ remote hosts.[1] | [] () {} |
| cmnRemoteMGCHOME_ixl | Path of the remote MGC_HOME on IXL remote hosts.[1] | [] () {} |
| cmnRemoteMGCHOMEUseCurrent | Boolean that controls whether to use the current MGC_HOME for remote runs. | [ 1/0 ] (Boolean) {1} |
| cmnRemoteMGCLIBPATH_aoi | MGC_LIB_PATH path on AOI remote hosts.[1] | [] () {} |
| cmnRemoteMGCLIBPATH_aoj | MGC_LIB_PATH path on AOJ remote hosts.[1] | [] () {} |
| cmnRemoteMGCLIBPATH_ixl | MGC_LIB_PATH path on IXL remote hosts.[1] | [] () {} |
| cmnRemoteMGCLIBPATHUseCurrent | Boolean that controls whether to use the current MGC_LIB_PATH for remote runs. | [] () {} |
| cmnRemoteShellDefault | Boolean that controls which remote shell program to use.<br>• 0 — Specify. The shell program is specified with cmnRemoteShellName.<br>• 1 — Default. Use the default rsh shell.<br>• 2 — Secure. Use the ssh shell. | [0 1 2] () {1} |
| cmnRemoteShellName | Name of the user-specified remote shell program to use. | [] () {} |
| cmnRemoteUserIsCurrentUser | Boolean that controls whether to use the current user name for remote runs. | [ 1/0 ] (Boolean) {1} |
| cmnRemoteUserName | Name of the remote user. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnResolution | Layout grid step size in database units. | [] (positive integer) {1} |
| cmnRunHier | Specifies the run type for DRC, LVS, and PERC runs:<br>-2 — Flat and Calibre CB<br>(-flatten mode in LVS)<br>-1 — Calibre CB (DRC)<br>0 — Flat (DRC, PERC)<br>1 — Hierarchical run<br>(DRC, LVS, PERC)<br>2 — LDL (PERC)<br>4 — Fast XOR (DRC)<br>6 — Autowaiver Creation<br>7 — Calibre Reconnaissance nmLVS Short Isolation (LVS)<br>(DRC/LVS/DFM)<br>Note: The value 3 is obsolete and the option is set to 1 if 3 is specified.<br>If the value 0 is set for an LVS run, the value is changed to -2. | [-2 -1 0 1 2 4 6] () {1} |
| cmnRunHostType | Specifies the host for the Calibre run (the “Run Calibre on” setting). | [localhost remotehost remoteuniva remotelsf remotecluster] () {localhost} |
| cmnRunHyper | Boolean that controls whether to turn on hyperscaling for the primary host. | [ 0/1 ] (Boolean) {0} |
| cmnRunHyperCompare | Boolean controlling whether to turn on hyperscaling for LVS comparison. | [ 0/1 ] (Boolean) {0} |
| cmnRunHyperSlave | Boolean that controls whether to turn on hyperscaling for the processing hosts. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnRunMT | Run Calibre single-threaded (0), multithreaded (1), or Calibre MTflex (2). | [0 1 2] () {0} |
| cmnRunOnOpen | Boolean that controls whether a Calibre run will start immediately when a runset is loaded. | [ 0/1 ] (Boolean) {0} |
| cmnRunPostTriggerInBG | Boolean that controls whether the post-trigger is run in background mode. | [ 0/1 ] (Boolean) {0} |
| cmnRunPostTriggerInRemote | Boolean controlling whether to run post-trigger on localhost or remote host. | [ 0/1 ] (Boolean) {0} |
| cmnRunPostTriggerOnAbort | Boolean that controls whether the post-trigger is run if the user aborts the Calibre run. | [ 0/1 ] (Boolean) {0} |
| cmnRunRVEOnLocalHost | Boolean that controls whether Calibre RVE is run on localhost. | [ 1/0 ] (Boolean) {1} |
| cmnSaveRunsetChanges | Boolean that controls the “Save changed settings before closing runset” setting. | [ 1/0 ] (Boolean) {1} |
| cmnSaveRunsetOnRun | Boolean that controls whether the runset is saved each time Calibre is run. | [ 0/1] (Boolean) {0} |
| cmnSaveRunsetToCurDirOnRun | Boolean that controls whether a copy of the runset is saved to the current directory each time Calibre is run. | [ 0/1] (Boolean) {0} |
| cmnSaveTVFRulesToSVRF | Boolean that controls whether the SVRF translation of a TVF rule file is saved to a file each time Calibre runs. | [ 0/1 ] (Boolean) {0} |
| cmnSchematicServerType | Type of schematic viewer. | [Mentor:ICgraph Mentor:S-Edit Cadence:Composer Synopsys:Laker Synopsys:CustomCompiler Other:Generic None] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnSHostName | Name of the host running the schematic viewer. | [] () {localhost} |
| cmnSHostPort | Port number of the schematic viewer. | [] () {9199} |
| cmnShowCustomDialog | Boolean that controls whether the customization dialog is shown when a runset is opened. | [ 1/0 ] (Boolean) {1} |
| cmnShowModifiedRulesPrompt | Boolean controlling whether a prompt is displayed when the rule file is automatically loaded. Only in effect if cmnCheckForModifiedRules is enabled. | [ 1/0 ] (Boolean) {1} |
| cmnShowOptions | Boolean that controls the Options checkbox in the **Setup** menu. | [ 0/1 ] (Boolean) {0} |
| cmnShowToolTips | Boolean that controls the “Show Tooltips” setting. | [ 1/0 ] (Boolean) {1} |
| cmnSlaveHosts | List of remote hosts and their settings for a MTflex localhost or remotehost run (cmnRunHostType set to localhost (default) or remotehost). | [] () {} |
| cmnSlaveHostsViewDetails | Boolean that controls the “Show Details” setting for Calibre MTflex remote hosts. | [ 0/1 ] (Boolean) {0} |
| cmnSpecifyControlFile | Boolean that controls whether the user has specified a control file name (this should not be confused with the template name when started from the layout editor, which has precedence). | [ 0/1 ] (Boolean) {0} |
| cmnSpecifyLicenseLoopIntervalTime | Boolean controlling whether a time between loop iterations is specified when loop (retry) licensing is in effect. | [ 0/1 ] (Boolean) {0} |
| cmnSpecifyLicenseLoopRetryOnError | Boolean controlling whether to continue retrying for a license when license errors are detected. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnSpecifyLicenseLoopTime | Boolean controlling whether a time to loop is specified when loop (retry) licensing is in effect. | [ 0/1 ] (Boolean) {0} |
| cmnSpecifyLicenseWaitTime | Boolean that controls whether a time to wait will be specified when license queueing is in effect. | [ 0/1 ] (Boolean) {0} |
| cmnSpecifyWaiverSetupFile | Boolean controlling whether the user has specified a waiver setup filename. | [ 0/1 ] (Boolean) {0} |
| cmnSpiceInstPrefix | Prefix added to each instance in the schematic by the netlister. | [] () {X} |
| cmnStartRVEAfterBatch | Boolean controlling whether Calibre RVE is started after a Calibre Interactive batch run. | [ 0/1 ] (Boolean) {0} |
| cmnSUseLayoutConn | Boolean that controls if the schematic viewer is also listening on the same port as the layout viewer. | [ 1/0 ] (Boolean) {1} |
| cmnSyncCPUNum | Boolean controlling whether to synchronize the Min CPUs in the host selection table. | [ 0/1 ] (Boolean) {0} |
| cmnTemplate_CF[2] | Template for the control file name. | [] () {} |
| cmnTemplate_DF[2] | Template for the DEF filename. | [] () {%l.def} |
| cmnTemplate_IL[2] | Controls the template preference "Export layout from layout viewer" setting on the Setup Preferences dialog box, **Templates** > **Inputs** tab.<br>1 — Yes, export from viewer<br>0 — Do not export from viewer<br>-1 — Use runset value.<br>See "Export from Viewer Preferences" on page 152. | [1 0 -1] () {1} |
| cmnTemplate_LI[2] | Template for the layout library name for database formats. | [] () {%L} |
| cmnTemplate_LP[2] | Template for the layout path filename. | [] () {%l.calibre.db} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnTemplate_RL[2] | Template for the rule file name. | [] () {} |
| cmnTemplate_RN[2] | Template for the run directory pathname. | [] () {} |
| cmnTemplate_TF[2] | Template for the transcript echo filename. | [] () {} |
| cmnTemplate_VI[2] | Template for the layout view name for database formats. | [] () {%V} |
| cmnTraceProperties | List of Trace Property statement values. Each entry is a list of lists with the format { *enable* {*arglist*}}, where arglist is a list of the arguments to the Trace Property statement. | [] () {} |
| cmnTranscriptEchoToFile | Boolean that controls whether the transcript is automatically echoed to a file. | [ 0/1 ] (Boolean) {0} |
| cmnTranscriptFile | The name of the transcript echo file. | [] () {} |
| cmnTranscriptFileAppend | Boolean that controls whether the transcript file is overwritten or appended to. | [ 0/1 ] (Boolean) {0} |
| cmnTranscriptIncludedFiles | Boolean that controls whether included files are expanded in the transcript window. | [ 0/1 ] (Boolean) {0} |
| cmnTurboAll | Boolean controlling whether to halt Calibre when licenses for requested number of CPUs cannot be obtained (the -turbo_all option) | [ 0/1 ] (Boolean) {0} |
| cmnUseCBforRVE | Boolean that controls the “Acquire CB license for RVE when using Calibre-CB” setting. | [ 1/0 ] (Boolean) {1} |
| cmnUseCustomFile | Boolean that controls the “Customization file” setting. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnUseMTFlexHostFile | Boolean controlling whether to specify a Calibre MTflex host file. This option is available if “Run Calibre on” is set to Local Host or Remote Host (cmnRunHostType set to localhost (default) or remotehost). | [ 0/1 ] (Boolean) {0} |
| cmnUsePrecision | Boolean that determines whether the Precision setting will have effect. | [ 0/1 ] (Boolean) {0} |
| cmnUseRDS | Boolean that controls whether to use the Remote Data System to store data on processing hosts. | [ 0/1 ] (Boolean) {0} |
| cmnUseResolution | Boolean that determines whether the Resolution setting will have effect. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_Achars | Array delimiter characters. | [] () {} |
| cmnV2LVS_AddPinNames | Names of pins to be added to each subckt. | [] () {} |
| cmnV2LVS_AddPins | Boolean controlling the “Add pins to each subckt” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_Cchars | Illegal SPICE character substitution characters. | [] () {} |
| cmnV2LVS_ConnectUPins | Boolean that controls the “Create numbered nets for unconnected pins” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_DontConnectPower | Boolean that controls the “Don't connect supply0/supply1 nets to global supply” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_GenerateTemplate | Boolean that controls the “Generate xRC source template file” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_KeepBackSlash | Boolean that controls the “Preserve backslash in escaped identifiers” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_LastTranslation | Time of last V2LVS translation. | [] () {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnV2LVS_PositionalPins | Boolean that controls the “Use positional pin order” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_Pprefix | Prefix for gate level primitives. | [] () {} |
| cmnV2LVS_s0Net | Name of net to connect to pins with 0 value. | [] () {} |
| cmnV2LVS_s1Net | Name of net to connect to pins with 1 value. | [] () {} |
| cmnV2LVS_SPICEIncFiles | Filenames of the SPICE include files used by the V2LVS Verilog translator. | [] () {} |
| cmnV2LVS_SPICELibFiles | Filenames of the SPICE library files used by the V2LVS Verilog translator. | [] () {} |
| cmnV2LVS_TranslateAlways | Boolean that controls the “Always translate before running Calibre” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_Uprefix | Prefix for unnamed pins and gates. | [] () {} |
| cmnV2LVS_UseCBLicense | Boolean that controls the “Prefer Calibre-CB license during license search” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_UseRangeMode | Boolean that controls the “Use aggregate mode for bus bits” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |
| cmnV2LVS_UserOptions | Specification of V2LVS options not provided in the GUI. | [] () {} |
| cmnV2LVS_VLOGLibFiles | Filenames of the Verilog library files. | [] () {} |
| cmnV2LVS_WarningLevel | Warning level for the V2LVS translator. See V2LVS documentation for value interpretation. | [0 1 2 3 4] () {2} |
| cmnV2LVS_WarningsAreErrors | Boolean that controls the “Treat warnings as errors” setting of the V2LVS translator. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnVConnectColon | Boolean that controls the “Connect nets with colon (:)” setting. | [ 0/1 ] (Boolean) {0} |
| cmnVConnectNames | The texts to connect-by-name if the SOME setting is used for cmnVConnectNamesState. | [] () {} |
| cmnVConnectNamesState | Controls the connect by names options:<br>NONE — Do not connect nets by name<br>ALL — Connect all nets by name<br>SOME — Connect nets named … | [NONE ALL SOME] () {NONE} |
| cmnVConnectReport | Boolean that controls the “Report connections made by name” setting (the Virtual Connect Report statement). | [ 0/1 ] (Boolean) {0} |
| cmnVConnectReportMaximumAll | Boolean that controls whether Calibre nmLVS, Calibre PERC, and PEX runs should report all virtual connect messages during the run. | [ 0/1 ] (Boolean) {1} |
| cmnVConnectReportMaximumCount | Maximum number of virtual connect messages to report during the run. | [ ] (non-negative integer) {0} |
| cmnVConnectReportUnSatisfied | Boolean that controls the “Don’t report connections that are physically connected” setting. When enabled, this option adds the UNSATISFIED keyword to the Virtual Connect Report statement. | [ 0/1 ] (Boolean) {0} |
| cmnVerifyCheckSelected | Boolean controlling whether to verify that at least one check is selected before running DRC. | [ 1/0 ] (Boolean) {1} |
| cmnVerifySourceNetlist | Boolean that controls whether to check the source netlist for errors before running Calibre. | [ 1/0 ] (Boolean) {1} |
| cmnWarnLayoutFormatsDiff | Boolean which controls whether to warn when a mixture of GDS and OASIS layout databases are detected before running Calibre. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnWarnLayoutOverwrite | Boolean that controls whether a dialog is shown when the layout database file is about to be overwritten. | [ 1/0 ] (Boolean) {1} |
| cmnWarnNoCustomization | Boolean that controls whether a dialog is shown to warn that the customization file does not contain any tool (DRC/DFM/LVS/PEX) specific settings. | [ 0/1 ] (Boolean) {0} |
| cmnWarnRunsetNotWritable | Boolean that controls whether a warning dialog is shown when the runset file is not writable. | [ 1/0 ] (Boolean) {1} |
| cmnWarnSourceOverwrite | Boolean that controls whether a dialog is shown when the schematic netlist file is about to be overwritten. | [ 1/0 ] (Boolean) {1} |
| cmnWXRBaseLayers | The list of base layers in FastXOR. Used when cmnWXRLayerMapOrBaseLayersUsage is set to “base”. | [] () {} |
| cmnWXRCommonOnly | Boolean that controls whether the XOR rule generation is for only layers common to both designs. | [ 0/1 ] (Boolean) {0} |
| cmnWXREnhanced | Boolean that controls the FastXOR setting “Add New and Missing shapes rules” on the **Rule File Generation** tab, **Format Options** category. | [ 0/1 ] (Boolean) {0} |
| cmnWXRExcludeLayersFile | Name of the exclude layers file on the **Rule File Generation** tab, **Mapping Files** category for FastXOR. | [] () {} |
| cmnWXRIncludeLayersFile | Name of the include layers file on the **Rule File Generation** tab, **Mapping Files** category for FastXOR. | [] () {} |
| cmnWXRIncludeRulesFile | Name of the include rules file on the **Rule File Generation** tab, **Mapping Files** category for FastXOR. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnWXRInternalFilePrefix | String for the FastXOR setting "Output Files Prefix" on the **Rule File Generation** tab, **Format Options** category. | [] () {rules.fxor} |
| cmnWXRLayerMapFile | Name of the layer map file on the **Rule File Generation** tab, **Mapping Files** category for FastXOR. | [] () {} |
| cmnWXRLayerMapOrBaseLayersUsage | Controls the state of the mutually exclusive "Use Layer Map File" and "Use Base Layers" settings on the **Rule File Generation** tab, **Mapping Files** category for FastXOR. Allowed values are:<br>• <empty string> — Do not use a layer map file or base layers (default).<br>• map — Use a layer map file.<br>• base — Use base layers. | [ <empty string> map base ] () {} |
| cmnWXRLayersFileUsage | Controls the state of the mutually exclusive "Use Include Layers File" and "Use Exclude Layers File" settings on the **Rule File Generation** tab, **Mapping Files** category for FastXOR. Allowed values are:<br>• <empty string> — Do not use an include or exclude layers file (default).<br>• include — Use an include layers file.<br>• exclude — Use an exclude layers file. | [ <empty string> include exclude ] () {} |
| cmnWXRResultFormatASCII | Boolean that controls the "Result Format: ASCII" setting on the **Rule File Generation** tab, **Format Options** category for FastXOR. | [ 1/0 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnWXRResultFormatGDS | Boolean that controls the “Result Format: GDS” setting in the **Rule File Generation** tab, **Format Options** category for FastXOR. | [ 1/0 ] (Boolean) {1} |
| cmnWXRResultFormatOASIS | Boolean that controls the “Result Format: OASIS” setting in the **Rule File Generation** tab, **Format Options** category for FastXOR. | [ 0/1 ] (Boolean) {0} |
| cmnWXRRulesFile | FastXOR generated rules file name. | [ ] () {rules.fxor} |
| cmnWXRUseIncludeRulesFile | Boolean that controls the “Use Include Rules File” setting on the **Rule File Generation** tab, **Mapping Files** category for FastXOR. | [ 0/1 ] (Boolean) {0} |
| cmnXRCompareAllPlacedCells | Boolean that controls the “Compare All Placed Cells” option on the FastXOR Options tab. (DBdiff option -compareallplacedcells) | [ 0/1 ] (Boolean) {0} |
| cmnXRCompareText | Boolean that controls the “Compare Text” option on the FastXOR Options tab. (DBdiff option -comparetext) | [ 0/1 ] (Boolean) {0} |
| cmnXREnablePassThroughLayers | Boolean that controls the “Enable Pass-Through Layers” option on the FastXOR Options tab. (CALIBRE_FX_PASS_THROUGH_LAYER_ENABLE) | [ 0/1 ] (Boolean) {0} |
| cmnXRMaxDiffCount | Threshold option for switching FastXOR to traditional XOR when the difference count found by DBdiff pre-processing exceeds the value. | [ ] (positive integer or empty string) {0} |
| cmnXRRDBFile | The name of the FastXOR RDB output file with optional filters.<br>*filename* [FC \| FP \| FT] [-sortlayer]<br>(DBdiff option -rdb) | [ ] () {} |
| cmnXRReportFile | The name of the FastXOR report file. (DBdiff option -report) | [ ] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-6. Common Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnXRTemplateFile | The name of the FastXOR template file. (DBdiff option -template) | [ ] () {$MGC_HOME/shared/pkgs/fdi/templates/fastxor.template} |

[1] See "Supported Operating Systems" in the *Calibre Administrator's Guide* for the definition of the AOI, AOJ, and IXL platforms.

[2] The cmnTemplate_* runset options set templates for file naming. See "Templates for File Naming" on page 148 and "Runset Options for Template Parameters" on page 960 for more information on file naming templates.

# Waiver Setup Runset Options

The waiver runset options apply to DRC, LVS, and DFM-CFA applications. With the exception of cmnSpecifyWaiverSetupFile, all control settings are on the Inputs pane in the **Waivers** tab of Calibre Interactive.

See "Calibre Auto-Waivers in Calibre Interactive" on page 201 for more information.

**Table C-7. Waiver Setup Runset Options**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnRunOrCreateWaivers | Use waivers during a Calibre run. | [ 0/1 ] (Boolean) {0} |
| cmnSpecifyWaiverSetupFile | Boolean controlling whether to use a user-specified waiver setup filename for the automatically generated waiver setup file. See "Specify waiver setup filename" in the Setup Preferences dialog box on the **Misc** tab. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverAddWaiverHierarchy | Boolean controlling whether to write waiver cell placements into an intermediate container cell. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverCalibreVersionTextLayer | Layer number for Calibre version text objects. | [ ] (non-negative integer) {1234} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-7. Waiver Setup Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnWaiverCalibreVersionTextType | Texttype number for Calibre version text objects. | [ ] (non-negative integer) {7} |
| cmnWaiverCalibreWaiverNumber | Layer used internally by Calibre during waiver run. | [ ] (non-negative integer) {2000} |
| cmnWaiverCellsFile | Name of the waiver flow cells file. | [] () {} |
| cmnWaiverChecksumTextLayer | Layer number for checksum text objects. | [ ] (non-negative integer) {1234} |
| cmnWaiverChecksumTextType | Texttype number for checksum text objects. | [ ] (non-negative integer) {1} |
| cmnWaiverCombineResults | Boolean that controls whether to generate a combined waivers results database with both the used and unused waiver shapes. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverCommentTextLayer | Layer number for comment text objects. | [ ] (non-negative integer) {1234} |
| cmnWaiverCommentTextType | Texttype number for comment text objects. | [ ] (non-negative integer) {5} |
| cmnWaiverCriteriaFile | Name of the waiver flow criteria file. | [ ] () {} |
| cmnWaiverCriteriaTextLayer | Layer number for criteria text objects. | [ ] (non-negative integer) {1234} |
| cmnWaiverCriteriaTextType | Texttype number for criteria text objects. | [ ] (non-negative integer) {2} |
| cmnWaiverDatabase | Name of file containing waiver cells. | [ ] () {} |
| cmnWaiverDatatypeNumber | Datatype number for waiver shapes. | [ ] (non-negative integer) {5678} |
| cmnWaiverDateTextLayer | Layer number for date text objects. | [ ] (non-negative integer) {1234} |
| cmnWaiverDateTextType | Texttype number for date text objects. | [ ] (non-negative integer) {4} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-7. Waiver Setup Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnWaiverDensityRDBOption | Boolean that controls whether to generate the *density_waived.rdb* file. This option corresponds to DENSITY_WAIVER_RDB in the waiver setup file. | [YES NO] () {YES} |
| cmnWaiverDescriptionFile | (Obsolete) Name of the waiver flow description file. This option is obsolete and should be replaced by WaiverCriteriaFile and WaiverCellFile. | [ ] () {waiver_desc} |
| cmnWaiverDescriptionMode | Mode used for waiver criteria file creation during waiver runs. In the obsolete usage with a waiver description file, this option applied to the waiver description file. | [ NONE EXTRACT APPEND] () {NONE} |
| cmnWaiverEnableDynamicResultsReporting | Boolean controlling whether to enable dynamic results reporting for Calibre RVE. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverEnableWaiverCellReportingt | Boolean controlling whether to annotate waiver cell name property in RDB files. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverIgnoreDensity | Boolean controlling whether to write waivers to waive density rule checks | [ 0/1 ] (Boolean) {0} |
| cmnWaiverIgnoreOps | List of layer operations to be ignored by waiver_flow. | [ ] () {} |
| cmnWaiverInternalLayerAuto | Boolean controlling whether to use Auto for the Calibre waiver number. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverIpmatchTextType | Texttype number for IP_MATCH text objects. | [ ] (non-negative integer) {6} |
| cmnWaiverLayerNumber | Layer number for waiver shapes. | [ ] (non-negative integer) {1234} |
| cmnWaiverMagnifyTextSpacing | Boolean that controls whether to apply the magnification factor for spacing between text. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-7. Waiver Setup Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnWaiverMerge | Boolean that controls the merge feature in the waiver flow. Set to 0/1 to disable or enable merge. Set to 2 to ignore waivers embedded in design when applying waivers; this option is only supported for DRC. | [ 0 1 2 ] () {0} |
| cmnWaiverNotIgnoreOps | List of layer operations to not be ignored by waiver_flow. | [ ] () {} |
| cmnWaiverOptimizeNoneWaiverStatistics | Boolean controlling whether to calculate and report waived, used, or unused waiver RDB with NONE. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverPrecisionConversion | Boolean controlling "Use rule file precision" (PRECISION_CONVERSION) setting. See "Waiver Cell Precision and Magnification Considerations" in the *Calibre Auto-Waivers User's and Reference Manual* for details. | [ 0/1 ] (Boolean) {1} |
| cmnWaiverRDBOption | Specifies options for generating the *waived.rdb* file. This option corresponds to WAIVER_RDB in the waiver setup file. | [ ALL PROP USER GEOM NONE] () {ALL} |
| cmnWaiverRetainResultType | Boolean controlling whether to retain edge and error results during waiver runs. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverRuleChecksumTextLayer | Layer number for rule checksum text objects. | [ ] (non-negative integer) {1234} |
| cmnWaiverRuleChecksumTextType | Texttype number for rule checksum text objects. | [ ] (non-negative integer) {9} |
| cmnWaiverRunCalibreVersionChecksum | Boolean controlling whether to verify the Calibre version during waiver runs. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverRunDFMGlobal | Boolean controlling application of waivers in a DFM waiver run. | [ 0/1 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-7. Waiver Setup Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnWaiverRunIPMatchChecksum | Boolean controlling whether to verify IP_MATCH checksums during waiver runs. | [ 0/1 ] (Boolean) {1} |
| cmnWaiverRunLayoutChecksum | Boolean controlling whether to run checksum verification of waiver cells. | [ 0/1 ] (Boolean) {1} |
| cmnWaiverRunRuleChecksum | Boolean controlling whether to verify rule checksums during the Calibre Auto-Waivers run. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverSetIPMatchForAllCells | Boolean controlling whether to set IP_MATCH checksums for waiver cells. | [ 0/1 ] (Boolean) {0} |
| cmnWaiverSetupFile | Waiver setup filename. This value is used when cmnSpecifyWaiverSetupFile is set to 1. | [ ] () {} |
| cmnWaiverSummaryFile | Name of the summary report file written during waiver_flow execution. | [ ] () {} |
| cmnWaiverTempDirectory | Location for temporary files written during waiver operations. | [ ] () {} |
| cmnWaiverTextMagnification | Magnification factor for text written to GDS waiver file during waiver_flow. | [ ] (positive double) {0.005} |
| cmnWaiverUnusedRDBOption | Specifies options for generating the *unused_waiver.rdb* file. This options corresponds to UNUSED_WAIVER_RDB in the waiver setup file. | [ ALL PROP USER GEOM NONE] () {ALL} |
| cmnWaiverUsedRDBOption | Specifies options for generating the *used_waiver.rdb* file. USED_WAIVER_RDB This options corresponds to USED_WAIVER_RDB in the waiver setup file. | [ ALL PROP USER GEOM NONE] () {ALL} |
| cmnWaiverUsernameTextLayer | Layer number for username text objects. | [ ] (non-negative integer) {1234} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-7. Waiver Setup Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmnWaiverUsernameTextType | Texttype number for username text objects. | [ ] (non-negative integer) {3} |

# 3DSTACK Runset Options

The 3DSTACK runset options are specific to Calibre Interactive 3DSTACK.

**Table C-8. Calibre 3DSTACK Runset Options**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackAnchorsTableLists | A list of list with the entries for the Anchors table on the Inputs pane in the **Placements** tab. The entries for each sublist are {use name chipName xOrigin yOrigin}. | [ ] (list of lists) {} |
| 3dstackAnchorsTableState | Boolean controlling whether 3DSTACK anchor commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackAssemblyFile | Pathname of the assembly file. | [ ] () {./3dstack_assembly} |
| 3dstackAssemblyFileFormat | Format of the assembly file. | [GDSII OASIS ] () {GDSII} |
| 3dstackAssemblyRunMode | Controls the generation and usage of assembly files.<br>**0** — Create and Use (default)<br>**1** — Use Generated<br>**2** — Create Only | [ 0 1 2] () {0} |
| 3dstackAttachTextsTableLists | List of lists which has the following elements: "use" "Chip/Placement Name" "Attached Placement Name" "Text Depth". | [ ] () {} |
| 3dstackAttachTextsTableState | Boolean controlling whether 3DSTACK attach texts commands are generated. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackAutomaticallyShowLayersInRVE | Boolean that controls whether Calibre RVE shows layers automatically. | [ 0/1 ] (Boolean) {0} |
| 3dstackCheckSources | Boolean controlling the "Check Sources" option on the Rules pane. | [ 0/1 ] (Boolean) {0} |
| 3dstackChipNameComboBoxValue | Chip Name which is exported from layout viewer. | [] () {} |
| 3dstackColumnOrder | The order of spreadsheet file columns. This is equivalent to the -order option for commands that read CSV formatted netlist. See "Spreadsheet Converters" in the *Calibre 3DSTACK User's Manual* for details. | [] () {} |
| 3dstackConnectivityTableLists | A list of lists with the entries for the Export Connectivity Information table on the Outputs pane in the **Export Data** tab. The entries for each sublist are {Use FileName Format Hierarchical No_Top Package}, where Format can be VERILOG, MGC, or SPICE. | [] (list of lists) {} |
| 3dstackConnectivityTableState | Boolean controlling whether 3DSTACK export_connectivity commands are generated. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackCreateAssembly | NOTE: This option and 3dstackRunMode are obsolete as of the 2013.1 release; the option 3dstackAssemblyRunMode is now used.<br>Boolean controlling generation of assembly files:<br>• **1**: Create the assembly file only<br>• **0**: Use an already generated assembly file<br>Note: This setting is only used when 3dstackRunMode is set to 1. | [ 0/1 ] (Boolean) {1} |
| 3dstackEnableChildRdbs | Boolean that controls whether a Calibre 3DSTACK run creates child RDBs. | [ 0/1 ] (Boolean) {0} |
| 3dstackEnvNamePatterns | Filter for shell environment variables to display in Set Environment Variables dialog. | [] () {MGC_*<br>*CALIBRE*<br>*3DSTACK*} |
| 3dstackEnvVars | {*name value use*} triplets of environment variable to apply to the DRC run. *use* should be one of:<br>• **Runset** — Use the value from the triplet.<br>• **Env** — use the value from the environment<br>• **Unset** — Unset the environment variable. | [ ] () {} |
| 3dstackExportConnectivity<br>This option is replaced by 3dstackConnectivityTableState, but is available for backward compatibility. | Boolean controlling whether 3DSTACK export_connectivity commands are generated. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackExportFromLayout | Boolean controlling whether the input layout is exported from the layout viewer. | [ 0/1 ] (Boolean) {0} |
| 3dstackExportLayoutTableLists | A list of lists for the Export Layout table. The sublists have the format { use filename property renameTexts text flat}. | [ ] (list of lists) {} |
| 3dstackExportLayoutTableState | Boolean controlling whether 3DSTACK export_layout commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackExportTemplate<br>This option is replaced by 3dstackTemplateTableState, but is available for backward compatibility. | Boolean controlling whether 3DSTACK export_template commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackGenerateNetlistFile | Boolean controlling whether to generate the 3D Netlist file through NLB. | [ 0/1 ] (Boolean) {0} |
| 3dstackGenerateNetlistOnly | Boolean controlling whether only Netlist file generation will be done during the Calibre 3DSTACK run. | [ 0/1 ] (Boolean) {0} |
| 3dstackIgnorePinsTableLists | A list of lists with the entries for the Ignore Pins table on the Inputs pane in the **Misc** tab. The entries for each sublist are {use layerPlacementName regexp}. | [ ] (list of lists) {} |
| 3dstackIgnorePinsTableState | Boolean controlling whether 3DSTACK ignore_pin commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackImportNetMapsTableLists | A list of lists for the import net maps table. The sublists have the format { use filename}. | [ ] (list of lists) {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackImportNetMapsTableState | Boolean controlling whether 3DSTACK import net map commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackImportPinMapsTableLists | List of lists for the Import Pin Maps table, which has following format: {use filename} | [ ] (list of lists) {} |
| 3dstackImportPinMapsTableState | Boolean controlling whether 3DSTACK import pin map commands are generated | [ 0/1 ] (Boolean) {0} |
| 3dstackImportTextLabelsTableLists | A list of list with the entries for the Import Text Labels table on the Inputs pane in the **Texts** tab. The entries for each sublist are {use fileName chipName}. | [ ] (list of lists) {} |
| 3dstackImportTextLabelsTableState | Boolean controlling whether 3DSTACK import_text_labels commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackLayersTableLists | A list of lists with the entries for the Layers table on the Inputs pane in the **Layout** tab. The entries for each sublist are {layerName chipName layerNum.datatype, …, SVRF, levelNumber}. | [ ] (list of lists) {} |
| 3dstackLayoutGetFromViewer | Boolean that controls the import from layout viewer option. This option has dependencies on cmnTemplate_IL; see "Export from Viewer Preferences" on page 152. | [ 0/1 ] (Boolean) {0} |
| 3dstackLayoutPaths | File path of the GDS/OASIS database which is imported from the layout viewer. | [] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackLayoutPexMapTableLists | A list of lists for the Layout PEX Map table. The entries for the sublists are {use placementLayerName calibratedLayerName} | [ ] (list of lists) {} |
| 3dstackLayoutPexMapTableState | Boolean controlling whether 3DSTACK pex_map commands are generated for export_layout. | [ 0/1 ] (Boolean) {0} |
| 3dstackLayoutsTableLists<br>(3dstackLayoutTableLists also works for backward compatibility) | List of Layout Table entries, where each entry is a list with the following items:{chip_name primary_cell layout_path layout_format precision} | [] () {} |
| 3dstackMapPlacementsTableLists | List of lists with the following format : {use layout_placement source_placement} | [] () {} |
| 3dstackMapPlacementsTableState | Boolean controlling whether 3DSTACK map_placement commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackLayoutSystem | Format of the input layout database. | [GDSII OASIS] () {GDSII} |
| 3dstackMaxResults | Boolean that controls how many results a 3DSTACK summary file includes:<br>• 0 — All results<br>• 1 — (default) A specified maximum results count. | [ 0/1 ] (Boolean) {1} |
| 3dstackMaxResultsValue | Maximum count of 3DSTACK summary file results. | [] (positive integer) {50} |
| 3dstackNetlistFileCase | Boolean controlling whether the source netlist data processing and connectivity checking is done in a case-sensitive manner. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackNetlistFile | Pathname of 3DSTACK source netlist file. | [] () {} |
| 3dstackNetlistFileFormat | Format of the 3DSTACK source netlist. | [SPICE MGC TSMC CSV XPI AIF VERILOG] () {SPICE} |
| 3dstackNetlistFileHier | Boolean controlling whether the source netlist data processing and connectivity checking should be done in a hierarchical manner. | [ 0/1 ] (Boolean) {0} |
| 3dstackNetlistFileWrap | The wrapped subckt name in source netlist. | [] () {} |
| 3dstackNetlistPlacementsTableLists | List of lists which has the following format: “use” “Chip Name” “Primary Cell” “Placements” “Netlist File Path” “Format” | [] () {} |
| 3dstackNetlistPlacementsTableState | Boolean controlling whether to generate the 3D Netlist file using the System Netlister. | [ 0/1 ] (Boolean) {0} |
| 3dstackNLBConfigurationFile | File path of System Netlister configuration file. | [] () {} |
| 3dstackPexMapTableLists | List of lists for the PEX Map table. The sublists have the form {Use CalibratedLayerName OriginalLayerName} | [ ] (list of lists) {} |
| 3dstackPexMapTableState | Boolean controlling whether 3DSTACK pex_map commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackPinSwapTableLists | List of lists for the Pin Swap table, which has following format: {use chip swappable_pin_list} | [ ] (list of lists) {} |
| 3dstackPinSwapTableState | Boolean controlling whether 3DSTACK pin_swap commands are generated. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackPlacementsTableLists | A list of list with the entries for the Placements table on the Inputs pane in the **Placements** tab. The entries for each sublist are {Name Type ChipLayerName PlacementType xOrigin yOrigin AnchorPlacementsName AnchorFromName AnchorToName Rotate Magnify Flip}. | [ ] (list of lists) {} |
| 3dstackPostTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute after Calibre 3DSTACK is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [] () {} |
| 3dstackPreTriggers | List of {name type enable} of the programs or procedures (and their arguments) to execute before Calibre 3DSTACK is run. The parameter type is layout, schematic, or process, and enable is boolean. See "Setting Calibre Interactive Internal Triggers" on page 661. | [ ] () {} |
| 3dstackRenameTextsTableLists | A list of list with the entries for the Rename Texts table on the Inputs pane in the **Texts** tab. The entries for each sublist are {use type ChipPlacementName pattern}. | [ ] (list of lists) {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackRenameTextsTableState | Boolean controlling whether 3DSTACK rename_text commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackReportIgnoredPins | Boolean controlling whether the 3DSTACK report ignores pins. | [ 0/1 ] (Boolean) {0} |
| 3dstackResultsDir | Directory path of the output 3DSTACK run. | [ ] () {.} |
| 3dstackRulesFile | Pathname of the rules file. | [ ] () {} |
| 3dstackRunDir | Pathname of the run directory. | [ ] () {.} |
| 3dstackRunMode | NOTE: This option and 3dstackCreateAssembly are obsolete as of the 2013.1 release; the option 3dstackAssemblyRunMode is now used.<br>Boolean controlling whether 3DSTACK<br>runs in default mode:<br>• **0** — Run in default mode<br>• **1** — Run mode is determined by 3dstackCreateAssembly | [ 0/1 ] (Boolean) {0} |
| 3dstackRunsetInfo | User-defined one line text specifying information about the runset. See "Runset Type and Runset Info" on page 45. | [ ] () {} |
| 3dstackRunsetType | User-defined runset type. See "Runset Type and Runset Info" on page 45. | [ ] () {} |
| 3dstackRVECTOFile | File path to the 3DSTACK RVE CTO file. | [ ] () {} |
| 3dstackSelectableRDBList | List of RDBs created by Calibre 3DSTACK. | [ ] () {} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackSelectedRDBList | List of RDBs which are opened in Calibre RVE after the Calibre 3DSTACK run completes. | [ ] (list of Booleans) {} |
| 3dstackSelectedRDBs | Obsolete. This runset option has been replaced by 3dstackSelectableRDBList and 3dstackSelectedRDBList, but is included for backward compatibility.<br>List of boolean values of selected RDBs to open in Calibre RVE. The list has the same number of row entries as the Layouts Table and the same order. For example, if the Layouts table has two chips. base and stack, a list of {1 0} specifies to open base.rdb. See 3dstackLayoutsTableLists for the entries in the Layouts Table. | [ ] (list of Booleans) {} |
| 3dstackShowChecksAlpha | Boolean controlling whether the checks in the check selection dialog are displayed alphabetically. | [ 0/1 ] (Boolean) {0} |
| 3dstackShowGroupsAlpha | Boolean controlling whether the groups in the check selection dialog are displayed alphabetically. | [ 0/1 ] (Boolean) {0} |
| 3dstackShowGroupsHier | Boolean controlling whether the groups in the check selection dialog are displayed hierarchically. | [ 0/1 ] (Boolean) {1} |
| 3dstackShowGroupsTop | Boolean controlling whether the groups are displayed from the top down. | [ 0/1 ] (Boolean) {1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackSourceFilterTableLists | List of lists with the following format:{ use chip_name {Subckt or Device} Subckt_Name Short_Device Short_Pins Constraint} | [ ] () {} |
| 3dstackSourceFilterTableState | Boolean controlling whether 3DSTACK source_filter commands are generated. | [ 0/1 ] (Boolean) {0} |
| 3dstackSourceNetlist | Boolean that controls whether a 3DSTACK source_netlist command is used. | [ 0/1 ] (Boolean) {0} |
| 3dstackStartRVE | Boolean controlling whether RVE is started after the Calibre run, either at the end or as soon as results are available. | [ 0/1 ] (Boolean) {1} |
| 3dstackSubcktPins | The 3DSTACK source spreadsheet file subckt pin. This is equivalent to the -subckt_pins option in the 3dstack::ss2spice and 3dstack::xsi2spice commands. | [NET_NAME PIN_NUMBER PIN_NAME] () {NET_NAME} |
| 3dstackSummaryFile | File path of the 3DSTACK summary file. | [ ] () {3dstack.summary} |
| 3dstackTemplateTableLists | A list of list with the entries for the Export Templates table on the Outputs pane in the **Export Data** tab. The entries for each sublist are {Use FileName Format ChipName}, where Format can be GDSII or OASIS, and ChipName can be stack or base. | [ ] (list of lists) {} |
| 3dstackTemplateTableState | Boolean controlling whether 3DSTACK export_template commands are generated. | [ 0/1 ] (Boolean) {0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-8. Calibre 3DSTACK Runset Options (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| 3dstackTopCell | The name of the top cell. | [ ] () {TOPCELL_3DI} |
| 3dstackVerilogVersion | The 3DSTACK source VERILOG file version. | [2001] () {2001 1995} |
| 3dstackViewSummary | Boolean controlling whether the summary file is opened after 3DSTACK finishes. | [ 1/0 ] (Boolean) {1} |
| 3dstackWriteSummary | Boolean controlling whether a 3DSTACK summary file is written from the DRC run. | [ 1/0 ] (Boolean) {1} |

# Runset Options for Template Parameters

Template parameters allow you to define certain Calibre Interactive GUI settings based on the layout or schematic database open in your design tool. These template parameters are saved in the runset.

The runset options that define template parameters are listed in Table C-9. The default template, if there is one, is given in the first column. See the section "Templates for File Naming" on page 148 for an explanation of file naming templates.

Template parameters that begin with the prefix "cmn" apply to all Calibre Interactive applications unless otherwise noted; templates parameters that apply to a specific application are prefixed accordingly.

**Table C-9. Runset Parameters for Template Settings**

| Template Parameter in the Runset | Corresponding Runset Option for Resolved Template | Definition |
|---|---|---|
| **Templates for all Calibre Interactive applications** | | |
| cmnTemplate_CF<br>*no default* | cmnControlFile | The control file name. |
| cmnTemplate_DF<br>default: %l.def | cmnFDIDEFLayoutPath | The DEF file for LEF/DEF databases. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-9. Runset Parameters for Template Settings (cont.)**

| Template Parameter in the Runset | Corresponding Runset Option for Resolved Template | Definition |
|---|---|---|
| cmnTemplate_IL<br>default: 1 | <*app*>LayoutGetFromViewer<br>where <*app*> is one of drc, dfm, lvs, perc, 3dstack, or pex | Determines the value for the "Export from layout viewer" setting when Calibre Interactive is opened from a layout viewer. See "Export from Viewer Preferences" on page 152. |
| cmnTemplate_LI<br>default: %L | cmnFDILayoutLibrary | Name of the layout library for GDS, OASIS, and OpenAccess databases. |
| cmnTemplate_LP<br>default: %l.calibre.db | <*app*>LayoutPaths, where <*app*> is one of drc, dfm, lvs, perc, or pex | Path to the input GDS or OASIS database. |
| cmnTemplate_RL<br>*no default* | <*app*>RulesFile, where <*app*> is one of drc, dfm, lvs, perc, or pex | Path to the rule file. |
| cmnTemplate_RN<br>*no default* | <*app*>RunDir, where <*app*> is one of drc, dfm, lvs, perc, or pex | Path to the run directory. |
| cmnTemplate_TF<br>*no default* | cmnTranscriptFile | Name of the transcript echo file. |
| cmnTemplate_VI<br>default: %V | cmnFDILayoutView | Name of the layout view for GDS, OASIS, and OpenAccess databases. |
| **Template parameters specific to Calibre Interactive nmDRC** | | |
| drcTemplate_DD<br>default: %l.dfmDefaults.db | drcDFMDefaultsResultsFile | File path of the output DRC database for the DFM Defaults RDB statement. |
| drcTemplate_RD<br>default: %l.drc.results | drcResultsFile | Path for the output DRC database. |
| drcTemplate_RH<br>default: %l.rhdb | drcRHDBFile | File path for the reusable hierarchical database (RHDB) file. |
| drcTemplate_SF<br>default: %l.drc.summary | drcSummaryFile | Path for the output DRC summary file. |
| **Template parameters specific to Calibre Interactive nmLVS** | | |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-9. Runset Parameters for Template Settings (cont.)**

| Template Parameter in the Runset | Corresponding Runset Option for Resolved Template | Definition |
|---|---|---|
| lvsTemplate_ED<br>default: %l.erc.results | lvsERCDatabase | Pathname of the ERC results database file. |
| lvsTemplate_ES<br>default: %l.sp | lvsSpiceFile | Path to the SPICE netlist extracted from the layout. |
| lvsTemplate_EU<br>default: %l.erc.summary | lvsERCSummaryFile | Pathname of the ERC summary report file. |
| lvsTemplate_HF<br>*no default* | lvsHCellsFile | Path to the hcells file. |
| lvsTemplate_IS<br>default: -1 | lvsSourceGetFromViewer | Determines the value for the "Export from schematic viewer" setting when Calibre Interactive is opened from a design tool. See "Export from Viewer Preferences" on page 152. |
| lvsTemplate_LR<br>default: %l.lvs.report | lvsReportFile | Filename of the LVS report file. |
| lvsTemplate_LS<br>default: %S | lvsSourceLibrary | The library name of the source topcell to run LVS on, only valid for Cadence Virtuoso. |
| lvsTemplate_SP<br>default: %s.src.net | lvsSourcePath | File paths of the input source netlist files. |
| lvsTemplate_SV<br>default: svdb | lvsSVDBDir | The directory path of the SVDB directory (relative to the run directory). |
| lvsTemplate_VS<br>default: %W | lvsSourceView | The cell view name of the source topcell to run Calibre nmLVS on, only valid for Cadence Virtuoso. |
| **Template parameters specific to Calibre Interactive DFM** | | |
| dfmTemplate_RD<br>default: %l.dfm.results | dfmResultsFile | File path of the output results database. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-9. Runset Parameters for Template Settings (cont.)**

| Template Parameter in the Runset | Corresponding Runset Option for Resolved Template | Definition |
|---|---|---|
| dfmTemplate_SF<br>default: %l.dfm.summary | dfmSummaryFile | File path of the DFM summary file. |
| **Template parameters specific to Calibre Interactive PERC** | | |
| percTemplate_ES<br>default: %l.sp | percSpiceFile | Path to the SPICE netlist extracted from the layout. |
| percTemplate_HF<br>*no default* | percHCellsFile | Path to the hcells file. |
| percTemplate_IS<br>default: -1 | percSourceGetFromViewer | Determines the value for the "Export from schematic viewer" setting when Calibre Interactive is opened from a design tool. See "Export from Viewer Preferences" on page 152. |
| percTemplate_LR<br>default: %l.perc.report | percReportFile | Filename of the PERC report file. |
| percTemplate_LS<br>default: %S | percSourceLibrary | The library name of the source topcell to run Calibre PERC on, only valid for Cadence Virtuoso. |
| percTemplate_SP<br>default: %s.src.net | percSourcePath | File paths of the input source netlist files. |
| percTemplate_SV<br>default: svdb | percSVDBDir | The directory path of the SVDB directory (relative to the run directory). |
| percTemplate_VS<br>default: %W | percSourceView | The cell view name of the source topcell to run Calibre PERC on, only valid for Cadence Virtuoso. |
| **Template parameters specific to Calibre Interactive PEX** | | |
| pexTemplate_HF<br>*no default* | pexHCellsFile | File path of hcells file to use for LVS. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table C-9. Runset Parameters for Template Settings (cont.)**

| Template Parameter in the Runset | Corresponding Runset Option for Resolved Template | Definition |
|---|---|---|
| pexTemplate_IS<br>default: -1 | pexSourceGetFromViewer | Determines the value for the "Export from schematic viewer" setting when Calibre Interactive is opened from a design tool. See "Export from Viewer Preferences" on page 152. |
| pexTemplate_LR<br>default: %l.lvs.report | pexReportFile | Filename of the LVS report file. |
| pexTemplate_LS<br>default: %S | pexSourceLibrary | The library name of the source topcell to run parasitic extraction on, only valid for Cadence Virtuoso. |
| pexTemplate_PN<br>default: %l.pex.netlist | pexPexNetlistFile | File path of the generated netlist file. |
| pexTemplate_PR<br>default: %l.pex.report | pexPexReportFile | Filename of the PEX Report file. |
| pexTemplate_SP<br>default: %s.src.net | pexSourcePath | File paths of the input source netlist files. |
| pexTemplate_SV<br>default: svdb | pexSVDBDir | Pathname to the svdb directory. The path is relative to the run directory. |
| pexTemplate_VS<br>default: %W | pexSourceView | The cell view name of the source topcell to run PEX on, only valid for Cadence Virtuoso. |
| pexTemplate_XF<br>**no default** | pexxCellsFile | Pathname of xcells file to use during PEX. |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Appendix D
# Runset Options for Calibre Interactive xACT

A Calibre Interactive runset is a text file created by Calibre Interactive to store the settings you specify in the GUI. You can load a runset into Calibre Interactive xACT to restore the GUI interface to the saved settings.

**Calibre Interactive xACT Runset File Format** . . . . . 965
**General Runset Options for Calibre Interactive xACT** . . . . . 966
**xACT Runset Options** . . . . . 990
**Template Definitions in the Calibre Interactive xACT Runset** . . . . . 1069
**Trigger Functions for the Calibre Interactive xACT Runset** . . . . . 1073

## Calibre Interactive xACT Runset File Format

Input for: Calibre Interactive xACT

Created by: Calibre Interactive xACT

Calibre Interactive runset files are built automatically using the Calibre Interactive GUI. Although not recommended, it is possible to manually edit the options in a runset file.

### Format

Most runset options are set by assigning the “value” or “specified” property. For example:

```
pex.rulesFile.value = "rules"
pex.runDir.value = "project"
cmn.applyTemplates.value = true
cmn.turboCommand.runHow.value = "mtflex"

cmn.layoutTextFile.specified = true
cmn.layoutTextFile.value = "layoutText.svrf"

cmn.showOptionPages.value = [ "Options", "Preferences”]
```

- Option names are prefixed by either cmn or pex.
- String values are quoted, although the quotes may not be included in the list of “Available Values” in the runset option tables. The values true and false are not quoted.
- Some settings have both a value and a checkbox that enables use of the setting. To use the option, the “specified” property must be set to true, as in the preceding example for the layoutTextFile runset option.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

- Some values are lists, which are enclosed in [] brackets.
- Some options are set by defining the “parameters” property (not shown in the example). The lists of runset options give the property name.

Some runset entries are functions, such as the following, which defines an environment variable to use during the run:

```
pex.envVars.setEnv(“name”, “value”)
```

Template definitions are also saved to the runset as functions. Trigger definitions are saved to the runset as an object declaration and command definition.

### Parameters

The entries for the runset file are described in these sections:

- “General Runset Options for Calibre Interactive xACT” on page 966
- “xACT Runset Options” on page 990

  This section includes the pex.envVars.setEnv and pex.envVars.unsetEnv runset options for setting and unsetting environment variables. Also see the envVars configuration file command.
- “Template Definitions in the Calibre Interactive xACT Runset” on page 1069
- “Trigger Functions for the Calibre Interactive xACT Runset” on page 1073

# General Runset Options for Calibre Interactive xACT

The general runset options have the prefix cmn.

Calibre Interactive xACT-specific runset options are prefixed with pex and are described in “xACT Runset Options” on page 990. See “Calibre Interactive xACT Runset File Format” on page 965 for important information.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table D-1. General Runset Options for Calibre Interactive xACT

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.applyTemplates.value | Boolean that controls whether templates are applied when Calibre Interactive is started from a design tool.<br>**Templates** page | [true false]<br>(Boolean)<br>{true} |
| cmn.cmnRunControl.clusterCommand.value | The command used to submit a job to the remote cluster. Used when cmn.cmnRunControl.runWhere = “cluster”.<br>**Run Control** page, “Run Calibre On” section, “Cluster Command”. | []<br>()<br>{} |
| cmn.cmnRunControl.remoteClusterSynch.value | Checkbox for “Synchronous” in Calibre runs on a remote cluster (cmn.cmnRunControl.runWhere = “cluster”).<br>• true — Both steps (LVS and xACT) are run. Calibre RVE and reports are displayed when the run is finished if appropriate options are set.<br>• false — Only one step (LVS or xACT) is run. The run transcript, Calibre RVE, and reports are not displayed when the run finishes.<br>**Run Control** page, “Run Calibre On” section, “Synchronous”. | [false true]<br>(Boolean)<br>{false} |
| cmn.cmnRunControl.remoteHostName.value | The name of the remote host for the Calibre run. Used when cmn.cmnRunControl.runWhere = “remote”.<br>**Run Control** page, “Run Calibre On” section, “Host Name”. | []<br>()<br>{localhost} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.cmnRunControl.runWhere.value | Specifies the host processor for the Calibre run.<br>• local — Run on the host that is running Calibre Interactive.<br>• remote — Run on the remote host specified with cmn.cmnRunControl.remoteHostName.value.<br>• cluster — Run on a computer cluster. The launch command is specified with cmn.cmnRunControl.clusterCommand.value.<br>**Run Control** page, “Run Calibre On” section, “Run Calibre xACT on” | |
| cmn.ipcSettings.layoutHostName.value | The name of the host running the layout viewer.<br>**Run Control** page, “Design Tool Settings” section | []<br>()<br>{localhost} |
| cmn.ipcSettings.layoutSocketNumber.value | The socket port number of the layout viewer. Enter a value from 1024 to 65535.<br>**Run Control** page, “Design Tool Settings” section | []<br>(integer)<br>{9189} |
| cmn.ipcSettings.layoutTmpFile.specified | Boolean controlling cmn.ipcSettings.layoutTmpFile.<br>**Run Control**, “Design Tool Settings” section | [false true]<br>(Boolean)<br>{false} |
| cmn.ipcSettings.layoutTmpFile.value | The name of the intermediate file to use for sending layout highlight data to the layout tool.<br>This setting is used if cmn.ipcSettings.layoutTmpFile.specified = true.<br>**Run Control**, “Design Tool Settings” section | []<br>()<br>{query_results} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.ipcSettings.schematicHostName.value | The name of the host running the schematic viewer.<br>This setting is used if cmn.ipcSettings.schematicUseLayoutIPC.value = false.<br>**Run Control** page, "Design Tool Settings" section | []<br>()<br>{localhost} |
| cmn.ipcSettings.schematicSocketNumber.value | The socket port number of the schematic viewer. Enter a value from 1024 to 65535.<br>This setting is used if cmn.ipcSettings.schematicUseLayoutIPC.value = false.<br>**Run Control** page, "Design Tool Settings" section | []<br>(integer)<br>{9199} |
| cmn.ipcSettings.schematicUseLayoutIPC.value | Boolean that specifies if the schematic viewer uses the same socket port as the layout viewer.<br>**Run Control** page, "Design Tool Settings" section | [true false]<br>(Boolean)<br>{true} |
| cmn.layout.layoutLibraryPathFDI.value | The name of the OPENACCESS library definitions file. This could be the path to either the *cds.lib* or *lib.defs* library definitions file.<br>Used only when the layout format is OPENACCESS.<br>**Inputs** page, Layout Path | []<br>()<br>{} |
| cmn.layout.topCellLibraryFDI.value | The library name for the top cell in an OPENACCESS database.<br>Used only when the layout format is OPENACCESS.<br>**Inputs** page, Layout Path | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.layout.topCellViewFDI.value | The view name for the top cell in an OPENACCESS database.<br>Used only when the layout format is OPENACCESS.<br>**Inputs** page, Layout Path | []<br>()<br>{} |
| cmn.layoutTextFile.specified | Boolean controlling cmn.layoutTextFile<br>**Options** page, “Include Layout Text File” | [false true]<br>(Boolean)<br>{false} |
| cmn.layoutTextFile.filePath.value | Specifies a file that contains Layout Text objects.<br>**Options** page, “Include Layout Text File” section | []<br>()<br>{} |
| cmn.licenseTimeout.specified | Boolean controlling cmn.licenseTimeout<br>**Run Control** page, “Calibre Interactive License Timeout” | [false true]<br>(Boolean)<br>{false} |
| cmn.licenseTimeout.timeout.value | The number of hours of inactivity after which to release the license.<br>**Run Control** page, “Calibre Interactive License Timeout” section | []<br>()<br>{1} |
| cmn.licensing.remoteLICENSEFILE Name.value | For runs on remote hosts or computer clusters, the name of the licensing environment variable to use on the remote host.<br>**Run Control** page, “Calibre Licensing” section, “License File Name” | [LM_LICENSE _FILE MGLS_LICENS E_FILE]<br>()<br>{LM_LICENSE _FILE} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.licensing.remoteLICENSEFILE UseCurrent.value | Specifies what license file value to use on remote hosts:<br>• CURRENT — Use the current value.<br>• SPECIFY — Specify the license file value (set with cmn.licensing.remoteLICENSEFI LEValue.value).<br>**Run Control** page, “Calibre Licensing” section, “Use” | [CURRENT SPECIFY]<br>()<br>{CURRENT} |
| cmn.licensing.remoteLICENSEFILE Value.value | The value of the remote licensing environment variable to use when cmn.licensing.remoteLICENSEFILE UseCurrent.value is set to SPECIFY.<br>**Run Control** page, “Calibre Licensing” section, “License File Value” | []<br>()<br>{} |
| cmn.miscPrefs.ignoreExportInBatch. value | Boolean that specifies how to handle layout and schematic export settings in a batch run.<br>• true — Ignore export settings<br>• false — Use export settings<br>By default, export settings are ignored in batch runs.<br>**Preferences** page, “Misc” section | [true false]<br>(Boolean)<br>{true} |
| cmn.miscPrefs.overwriteLayoutPref. value | Boolean that controls whether a dialog is shown when the layout database file is about to be overwritten during layout export.<br>**Preferences** page, “Misc” section | [true false]<br>(Boolean)<br>{true} |
| cmn.miscPrefs.overwriteSourcePref. value | Boolean that controls whether a dialog is shown when the schematic source file is about to be overwritten during schematic export.<br>**Preferences** page, “Misc” section | [true false]<br>(Boolean)<br>{true} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.miscPrefs.spiceInstPrefix.value | The prefix that the netlister adds to each instance in the schematic. Calibre Interactive adds this prefix to instance names obtained by clicking in the schematic.<br>**Preferences** page, “Misc” section | []<br>()<br>{X} |
| cmn.miscPrefs.verifySourceNetlist.value | Boolean that controls whether to check the source netlist for errors before running Calibre xACT.<br>**Preferences** page, “Misc” section | [true false]<br>(Boolean)<br>{true} |
| cmn.miscPrefs.warnLayoutFormatsDiff.value | Boolean that controls whether to warn when the format of a library layout file differs from the primary layout format. This setting applies only when the primary layout format is GDS or OASIS.<br>**Preferences** page, “Misc” section | [true false]<br>(Boolean)<br>{false} |
| cmn.oaEnvOptions.setEnableExpPCell.value | Boolean that controls the “Set CDS_ENABLE_EXP_PCELL …” option.<br>**OPENACCESS** page, “OpenAccess Environment” section | [true false]<br>(Boolean)<br>{false} |
| cmn.oaEnvOptions.setExpAllParams.value | Boolean that controls the “Set CDS_EXP_ALL_PARAMS …” option.<br>**OPENACCESS** page, “OpenAccess Environment” section | [true false]<br>(Boolean)<br>{false} |
| cmn.oaEnvOptions.setExpPCellDir.value | Boolean that controls the “Set CDS_EXP_PCELL_DIR …” option.<br>**OPENACCESS** page, “OpenAccess Environment” section | [true false]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.oaEnvOptions.setLdLibPath.value | Boolean that controls the "Set CALIBRE_READDB_LD_LIBRARY_PATH …" option.<br>**OPENACCESS** page, "OpenAccess Environment" section | [true false]<br>(Boolean)<br>{false} |
| cmn.oaEnvOptions.setOaHome.value | Boolean that controls the "Set OA_HOME when …" option.<br>**OPENACCESS** page, "OpenAccess Environment" section | [true false]<br>(Boolean)<br>{false} |
| cmn.oaEnvOptions.setPath.value | Boolean that controls the "Set PATH when …" option.<br>**OPENACCESS** page, "OpenAccess Environment" section | [true false]<br>(Boolean)<br>{false} |
| cmn.oaLefDefAdditionalFiles.cellGDSDirectories.value | Cell GDS directory list used in database conversion. Used when cmn.oaLefDefAdditionalFiles.useCellGDSFiles.value = true.<br>**OPENACCESS** page, "Additional Files" section | []<br>()<br>{} |
| cmn.oaLefDefAdditionalFiles.cellGDSFiles.value | Cell GDS file list used in database conversion. Used when cmn.oaLefDefAdditionalFiles.useCellGDSFiles.value = true.<br>**OPENACCESS** page, "Additional Files" section | []<br>()<br>{} |
| cmn.oaLefDefAdditionalFiles.useCellGDSFiles.value | Boolean that controls the "Use Cell GDS Files" checkbox.<br>**OPENACCESS** page, "Additional Files" section | [true false]<br>(Boolean)<br>{false} |
| cmn.oaLefDefMappingFiles.cellMapFile.specified | Boolean controlling cmn.oaLefDefMappingFiles.cellMapFile<br>**OPENACCESS** page, "Mapping Files" section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.oaLefDefMappingFiles.cellMapFile.value | The cell map file used in database conversion.<br>**OPENACCESS** page, “Mapping Files” section | []<br>()<br>{} |
| cmn.oaLefDefMappingFiles.inputExceptionFile.specified | Boolean controlling cmn.oaLefDefMappingFiles.inputExceptionFile<br>**OPENACCESS** page, “Mapping Files” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefMappingFiles.inputExceptionFile.value | The input exception file used in database conversion.<br>**OPENACCESS** page, “Mapping Files” section | []<br>()<br>{} |
| cmn.oaLefDefMappingFiles.layerMapFile.specified | Boolean controlling cmn.oaLefDefMappingFiles.layerMapFile<br>**OPENACCESS** page, “Mapping Files” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefMappingFiles.layerMapFile.value | The layer map file used in layout database conversion.<br>**OPENACCESS** page, “Mapping Files” section | []<br>()<br>{} |
| cmn.oaLefDefMappingFiles.netPropertiesFile.specified | Boolean controlling cmn.oaLefDefMappingFiles.netPropertiesFile<br>**OPENACCESS** page, “Mapping Files” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefMappingFiles.netPropertiesFile.value | The net properties file used in layout database conversion.<br>**OPENACCESS** page, “Mapping Files” section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.oaLefDefMappingFiles.objectMapFile.specified | Boolean controlling cmn.oaLefDefMappingFiles.objectMapFile<br>**OPENACCESS** page, “Mapping Files” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefMappingFiles.objectMapFile.value | The object map file used in layout database conversion.<br>**OPENACCESS** page, “Mapping Files” section | []<br>()<br>{} |
| cmn.oaLefDefOtherOptions.oaViewLists.value | The view list for OpenAccess libraries,<br>**OPENACCESS** page, “Other Options” section | []<br>()<br>{} |
| cmn.oaLefDefOtherOptions.writeLayerNamesFile.specified | Boolean controlling cmn.oaLefDefOtherOptions.writeLayerNamesFile<br>**OPENACCESS** page, “Other Options” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefOtherOptions.writeLayerNamesFile.value | The output layer name file for “Write Layer Names File”. Used in layout database conversion.<br>**OPENACCESS** page, “Other Options” section | []<br>()<br>{} |
| cmn.oaLefDefOtherOptions.writeLogFile.specified | Boolean controlling cmn.oaLefDefOtherOptions.writeLogFile<br>**OPENACCESS** page, “Other Options” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefOtherOptions.writeLogFile.value | The output log file for “Write Log File”. Used in layout database conversion.<br>**OPENACCESS** page, “Other Options” section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.oaLefDefOtherOptions.writeMapInfoFile.specified | Boolean controlling cmn.oaLefDefOtherOptions.writeMapInfoFile<br>**OPENACCESS** page, “Other Options” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefOtherOptions.writeMapInfoFile.value | The output map info file for “Write Map InfoFile” Used in layout database conversion.<br>**OPENACCESS** page, “Other Options” section | []<br>()<br>{} |
| cmn.oaLefDefReadOptions.abortOnReadEmptyPCells.value | Boolean that controls whether to abort upon encountering empty PCells when reading an OpenAccess library.<br>**OPENACCESS** page, “Read Options” section | [true false]<br>(Boolean)<br>{true} |
| cmn.oaLefDefReadOptions.instanceNamesAsProperty.specified | Boolean controlling cmn.oaLefDefReadOptions.instanceNamesAsProperty<br>**OPENACCESS** page, “Read Options” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefReadOptions.instanceNamesAsProperty.value | Value of the property name used for “Read Instance Name as Property”.<br>**OPENACCESS** page, “Read Options” section | []<br>()<br>{} |
| cmn.oaLefDefReadOptions.netNamesAsProperty.specified | Boolean controlling cmn.oaLefDefReadOptions.netNamesAsProperty<br>**OPENACCESS** page, “Read Options” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefReadOptions.netNamesAsProperty.value | Value of the property name used for “Read Net Names as Property”.<br>**OPENACCESS** page, “Read Options” section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.oaLefDefReadOptions.netNamesAsText.specified | Boolean controlling cmn.oaLefDefReadOptions.netNamesAsText<br>**OPENACCESS** page, “Read Options” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefReadOptions.netNamesAsText.value | Value set for “Read Net Names as Text”.<br>• ALL — (default) Add net name annotations in all cells<br>• TOP — Add net name annotations in the top cell only<br>This option is off by default.<br>**OPENACCESS** page, “Read Options” section | [TOP ALL]<br>()<br>{ALL} |
| cmn.oaLefDefReadOptions.netTypeAsProperty.specified | Boolean controlling cmn.oaLefDefReadOptions.netTypeAsProperty<br>**OPENACCESS** page, “Read Options” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefReadOptions.netTypeAsProperty.value | Value of the property name used for “Read Net Type as Property”.<br>**OPENACCESS** page, “Read Options” section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.oaLefDefReadOptions.objects.value | Specifies the objects that are read when converting a third-party database to GDS or OASIS. Specify a comma-separated list enclosed in brackets.<br>• TEXTS — Read text objects<br>• PINS — Read pin shapes.<br>• BLOCKAGES — Read blockages for OpenAccess databases.<br>• INSTANCEPINS — Read instance pin shapes.<br>Default: [ “TEXTS”, “PINS”]<br>**OPENACCESS** page, “Read Options” section | [TEXTS PINS BLOCKAGES INSTANCEPINS]<br>(list)<br>{} |
| cmn.oaLefDefReadOptions.pinNamesAsProperty.specified | Boolean controlling cmn.oaLefDefReadOptions.pinNamesAsProperty<br>**OPENACCESS** page, “Read Options” section | [false true]<br>(Boolean)<br>{false} |
| cmn.oaLefDefReadOptions.pinNamesAsProperty.value | Value of the property name used for “Read Pin Names as Property”.<br>**OPENACCESS** page, “Read Options” section | []<br>()<br>{} |
| cmn.oaLefDefReadOptions.pinNamesAsText.value | Boolean controlling the “Read Pin Names as Text” checkbox.<br>**OPENACCESS** page, “Read Options” section | [true false]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.remoteEnv.remoteMGCHOME UseCurrent.value | Specifies what value of MGC_HOME to use for launching Calibre on the remote host.<br>• LOCAL — Use the localhost value of MGC_HOME<br>• SPECIFY — Specify the value of MGC_HOME<br>**Run Control** page, Remote Calibre Environment section | [LOCAL SPECIFY]<br>()<br>{LOCAL} |
| cmn.remoteEnv.remoteMGCHOME _aoi.value | The value of MGC_HOME to use for launching Calibre on an aoi remote host. Used when cmn.remoteEnv.remoteMGCHOME UseCurrent=SPECIFY.<br>**Run Control** page, Remote Calibre Environment section | []<br>()<br>{} |
| cmn.remoteEnv.remoteMGCHOME _aoj.value | The value of MGC_HOME to use for launching Calibre on an aoj remote host. Used when cmn.remoteEnv.remoteMGCHOME UseCurrent=SPECIFY.<br>**Run Control** page, Remote Calibre Environment section | []<br>()<br>{} |
| cmn.remoteEnv.remoteMGCHOME _ixl.value | The value of MGC_HOME to use for launching Calibre on an ixl remote host. Used when cmn.remoteEnv.remoteMGCHOME UseCurrent=SPECIFY.<br>**Run Control** page, Remote Calibre Environment section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.remoteEnv.remoteMGCLIBPATHUseCurrent.value | Specifies what value of MGC_LIB_PATH to use for Calibre on the remote host.<br>• LOCAL — Use the localhost value of MGC_LIB_PATH<br>• SPECIFY — Specify the value of MGC_LIB_PATH<br>**Run Control** page, Remote Calibre Environment section | [LOCAL SPECIFY]<br>()<br>{LOCAL} |
| cmn.remoteEnv.remoteMGCLIBPATH_aoi.value | The value of MGC_LIB_PATH to use for Calibre on an aoi remote host. Used when cmn.remoteEnv.remoteMGCLIBPATHUseCurrent=SPECIFY.<br>**Run Control** page, Remote Calibre Environment section | []<br>()<br>{} |
| cmn.remoteEnv.remoteMGCLIBPATH_aoj.value | The value of MGC_LIB_PATH to use for Calibre on an aoj remote host. Used when cmn.remoteEnv.remoteMGCLIBPATHUseCurrent=SPECIFY.<br>**Run Control** page, Remote Calibre Environment section | []<br>()<br>{} |
| cmn.remoteEnv.remoteMGCLIBPATH_ixl.value | The value of MGC_LIB_PATH to use for Calibre on an ixl remote host. Used when cmn.remoteEnv.remoteMGCLIBPATHUseCurrent=SPECIFY.<br>**Run Control** page, Remote Calibre Environment section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.remoteEnv.remoteShellDefault.value | Specifies which shell to use for launching Calibre on the remote host.<br>• DEFAULT — Use the default platform-specific shell<br>• SECURE — Use the secure shell. Password authentication is disabled and public key authentication is recommended.<br>• SPECIFY — Specify the path to the shell executable.<br>**Run Control** page, Remote Calibre Environment section | [DEFAULT SECURE SPECIFY]<br>()<br>{DEFAULT} |
| cmn.remoteEnv.remoteShellName.value | The path to the shell to use for launching Calibre on the remote host. Used when cmn.remoteEnv.remoteShellDefault=SPECIFY.<br>**Run Control** page, Remote Calibre Environment section | []<br>()<br>{} |
| cmn.remoteEnv.remoteUserIsCurrentUser.value | Specify what user name to use for the remote Calibre run.<br>• CURRENT — Use the current user name<br>• SPECIFY — Specify the user name<br>**Run Control** page, Remote Calibre Environment section | [CURRENT SPECIFY]<br>()<br>{CURRENT} |
| cmn.remoteEnv.remoteUserName.value | The user name for the remote Calibre run. Used when cmn.remoteEnv.remoteUserIsCurrentUser=SPECIFY.<br>**Run Control** page, Remote Calibre Environment section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.rulesPrefs.alwaysCompile.value | Boolean that controls the option "Always compile the rules file when Run button is pressed".<br>**Preferences** page, Rules section | [true false]<br>(Boolean)<br>{false} |
| cmn.rulesPrefs.batchModeCompatibleControlFile.value | Boolean that controls the option "Write control file suitable for command line run".<br>**Preferences** page, Rules section | [true false]<br>(Boolean)<br>{true} |
| cmn.rulesPrefs.checkForModifiedRules.value | Boolean that controls the option "Automatically load rules file".<br>When enabled, the rule file is automatically loaded 1) when the runset is loaded, and 2) at runtime if the rules file has been changed.<br>**Preferences** page, Rules section | [true false]<br>(Boolean)<br>{false} |
| cmn.rulesPrefs.compileOnly.value | Boolean that controls the option "Only compile the rules file when Run button is pressed"<br>**Preferences** page, Rules section | [true false]<br>(Boolean)<br>{false} |
| cmn.rulesPrefs.controlFile.specified | Boolean controlling cmn.rulesPrefs.controlFile<br>**Preferences** page, Rules section | [false true]<br>(Boolean)<br>{false} |
| cmn.rulesPrefs.controlFile.value | The name of the Calibre Interactive control file.<br>This setting is not used if Calibre Interactive is invoked from a design tool and a template for the control file is specified.<br>**Preferences** page, Rules section | []<br>()<br>{} |
| cmn.rulesPrefs.readIOFromRules.value | Boolean that controls the option "Read Inputs/Outputs fields when loading rules file".<br>**Preferences** page, Rules section | [true false]<br>(Boolean)<br>{true} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.rulesPrefs.rulesAfterMain.value | Boolean that controls the option "Include additional rules files and statements after main rules file".<br>**Preferences** page, Rules section | [true false]<br>(Boolean)<br>{false} |
| cmn.rulesPrefs.saveTvfToSvrf.value | Boolean that controls the option "Save TVF Rules to SVRF before running Calibre".<br>The generated SVRF is named *rules.svrf*, where *rules* is the name of the TVF rule file on the **Rules** page. This adds the -E option to the calibre command line.<br>**Preferences** page, Rules section | [true false]<br>(Boolean)<br>{false} |
| cmn.rulesPrefs.showModifiedRulesPrompt.value | Boolean that controls the option "Show prompt". Enable this to display a prompt before loading the rule file.<br>This option takes effect only if cmn.rulesPrefs.checkForModifiedRules.value is set to true.<br>**Preferences** page, Rules section | [true false]<br>(Boolean)<br>{true} |
| cmn.rulesPrefs.transcriptIncludedFiles.value | Select this setting to display contents of included rules files in the transcript when Calibre is run.<br>Boolean that controls the option "Display text of included rules files in transcript".<br>**Preferences** page, Rules section | [true false]<br>(Boolean)<br>{false} |
| cmn.runsetPrefs.processLocalLinks.value | Boolean that controls whether links are created in the directory of the runset.<br>**Preferences** page, Runset section | [true false]<br>(Boolean)<br>{true} |
| cmn.runsetPrefs.runOnOpen.value | Boolean that controls the option "Run Calibre when runset is opened".<br>**Preferences** page, Runset section | [true false]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.runsetPrefs.saveOnClose.value | Setting for the option "On Close Runset," which determines behavior when the GUI is closed:<br>• PROMPT — Prompt to save the runset if it has changed.<br>• AUTOSAVE — Automatically save the runset.<br>• NOSAVE — Do not save the runset and do not prompt.<br>**Preferences** page, Runset section | [PROMPT AUTOSAVE NOSAVE]<br>()<br>{PROMPT} |
| cmn.runsetPrefs.saveOnRun.value | Boolean that controls the option "Save runset each time Calibre is run".<br>**Preferences** page, Runset section | [true false]<br>(Boolean)<br>{false} |
| cmn.runsetPrefs.saveRunsetToCurDirOnRun.value | Boolean that controls the option "Save runset to run directory each time Calibre is run".<br>**Preferences** page, Runset section. | [true false]<br>(Boolean)<br>{false} |
| cmn.rve.autoClose.value | Boolean that controls the option "Close RVE before running Calibre *<app>*".<br>**Run Control** page, RVE Options section | [true false]<br>(Boolean)<br>{true} |
| cmn.rve.closeOnExit.value | Boolean that controls the option "Close RVE when Calibre Interactive closes".<br>**Run Control** page, RVE Options section | [true false]<br>(Boolean)<br>{true} |
| cmn.rve.runOnLocalHost.value | Boolean that controls the option "Run RVE on local host". If set to false, RVE is run using the settings in the "Run Calibre On" option.<br>**Run Control** page, RVE Options section | [true false]<br>(Boolean)<br>{true} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.showOptionPages.value | The list of pages to show. The default is to show all optional pages.<br>Default: [ “Database”, “Environment”, “Options”, “Preferences”, “Templates”, “Triggers” ]<br>**Settings** > **Show Pages** menu item | []<br>(list)<br>{} |
| cmn.transcriptIssuesPrefs.specified | Checkbox for “Errors and Warnings” on the **Preferences** page. Enable this to automatically save the errors and warnings.<br>Controls the cmn.transcriptIssuesPrefs group of options. | [true false]<br>(Boolean)<br>{false} |
| cmn.transcriptIssuesPrefs.appendOrReplace.value | Selection for the “Mode” option. Specifies whether to replace or add to an existing errors and warnings file when cmn.transcriptIssuesPrefs.specified = true.<br>**Preferences** page, Errors and Warnings section | [REPLACE APPEND]<br>()<br>{REPLACE} |
| cmn.transcriptIssuesPrefs.errorEchoFile.value | The name of the saved errors and warnings file when cmn.transcriptIssuesPrefs.specified = true.<br>**Preferences** page, Errors and Warnings section, “File Path” | []<br>()<br>{} |
| cmn.transcriptPrefs.specified | Checkbox for “Transcript” on the **Preferences** page. Enable this to automatically save the transcript.<br>Controls the cmn.transcriptPrefs group of options. | [true false]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.transcriptPrefs.appendOrReplace.value | Selection for the "Mode" option. Specifies whether to replace or add an existing transcript file when cmn.transcriptPrefs.specified = true.<br>**Preferences** page, Transcript section | [REPLACE APPEND]<br>()<br>{REPLACE} |
| cmn.transcriptPrefs.transcriptEchoFile.value | The name of the saved transcript when cmn.transcriptPrefs.specified = true.<br>**Preferences** page, Transcript section, "File Path" | []<br>()<br>{} |
| cmn.turboCommand.mTFlexHostFile.value | The path to remote host configuration file for Calibre MTflex runs. (-remotefile)<br>**Run Control** page, Run Calibre section, "Remote Host File" | []<br>()<br>{} |
| cmn.turboCommand.numProcessors.value | The number of processors to use for the Calibre run in multithreaded runs. (-turbo)<br>**Run Control** page, Run Calibre section, "Number of CPUs to use" | []<br>(integer)<br>{2} |
| cmn.turboCommand.remoteHostsHow.value | Select how to specify the remote hosts for a Calibre MTflex run.<br>**Run Control** page, Run Calibre section, "Remote Hosts" | |
| cmn.turboCommand.runHow.value | Specify the Calibre execution mode:<br>• single — Single-threaded (-turbo 1)<br>• multi — Multithreaded (-turbo *num*)<br>• all — Multithreaded with all available processors (-turbo)<br>• mtflex — Calibre MTflex (-turbo -remotefile *file*)<br>**Run Control** page, Run Calibre section, "Run Calibre xACT using" | [single multi all mtflex]<br>()<br>{multi} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.turboCommand.turboAll.value | Boolean control for "Halt Calibre <*app*> if licenses cannot be obtained for number of CPUs specified". (-turbo_all)<br>**Run Control** page, Run Calibre section | [true false]<br>(Boolean)<br>{false} |
| cmn.units.overrideLayoutPrecision.value | Checkbox control for "Override layout value". The control is active if "Precision" is specified.<br>If not enabled, a compiler error results if the database precision does not match the rule file precision.<br>**Database** page, Units section | [true false]<br>(Boolean)<br>{false} |
| cmn.units.precision.specified | Boolean controlling cmn.units.precision<br>**Database** page, Units section | [false true]<br>(Boolean)<br>{false} |
| cmn.units.precision.value | Specifies the precision, which is the ratio of database units to user units.<br>This setting provides the value for the Precision statement. It can be a single integer value, or a list of two integers.<br>**Database** page, Units section | []<br>()<br>{1000} |
| cmn.units.resolution.specified | Boolean controlling cmn.units.resolution<br>**Database** page, Units section | [false true]<br>(Boolean)<br>{false} |
| cmn.units.resolution.value | Specifies the layout (or user) grid step size in database units.<br>This setting provides the value for the Resolution statement. It can be a single integer value, or a list of two integers to provide the step size in the x and y directions separately.<br>**Database** page, Units section | []<br>()<br>{1} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.virtualConnect.colon.specified | Boolean controlling cmn.virtualConnect.colon<br>**Options** page, Virtual Connect section | [false true]<br>(Boolean)<br>{false} |
| cmn.virtualConnect.colon.value | Value for the “Virtual Connect Colon” option. Specifies whether to virtually connect physically disjoint nets whose names agree up to the first colon character found in the names.<br>**Options** page, Virtual Connect section | [NO YES]<br>()<br>{NO} |
| cmn.virtualConnect.netNames.value | The list of net names to connect by name for the Virtual Connect Name statement. The wildcard character “?” matches zero or more characters.<br>This option is used when cmn.virtualConnect.vconnect.value= “SPECIFIEDNAMES” (the “Virtual Connect Name” option).<br>**Options** page, Virtual Connect section | []<br>(list)<br>{} |
| cmn.virtualConnect.report.specified | Boolean controlling cmn.virtualConnect.report<br>**Options** page, Virtual Connect section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table D-1. General Runset Options for Calibre Interactive xACT (cont.)

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.virtualConnect.report.value | The setting for the “Virtual Connect Report” option, which controls warnings produced when a virtual connection is made.<br>• NO — Do not report connections.<br>• YES — Report connections.<br>• YES UNSATISFIED — Only report virtual connections that are not physically connected higher in the hierarchy.<br>**Options** page, Virtual Connect section | [“NO” “YES” “YES UNSATISFIED”]<br>()<br>{NO} |
| cmn.virtualConnect.reportMaximum.specified | Boolean controlling cmn.virtualConnect.reportMaximum<br>**Options** page, Virtual Connect section | [false true]<br>(Boolean)<br>{false} |
| cmn.virtualConnect.reportMaximum.value | Specifies whether to limit the maximum number of virtual connect warnings issued when virtual connect reporting is enabled.<br>• MAXIMUMALL — Report all<br>• MAXIMUMCOUNT — Limit reporting to the number specified by cmn.virtualConnect.reportMaximumCount.value.<br>**Options** page, Virtual Connect section | [MAXIMUMALL MAXIMUMCOUNT]<br>()<br>{MAXIMUMALL} |
| cmn.virtualConnect.reportMaximumCount.value | Specifies the upper limit on the number of virtual connect messages. Used when cmn.virtualConnect.reportMaximum = MAXIMUMCOUNT.<br>**Options** page, Virtual Connect section | []<br>(positive integer)<br>{0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-1. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| cmn.virtualConnect.vconnect.specified | Boolean controlling cmn.virtualConnect.vconnect.<br>This option controls the "Virtual Connect Name" checkbox, which determines whether to use the statement Virtual Connect Name.<br>• false — Do not use. Disjoint nets with the same name are treated as separate nets.<br>• true — Use. Disjoint nets are connected by name with the setting specified by cmn.virtualConnect.vconnect.value.<br>**Options** page, Virtual Connect section | [false true]<br>(Boolean)<br>{false} |
| cmn.virtualConnect.vconnect.value | Specifies how to form virtual connections of named nets.<br>• ALLNAMES — Virtually connect disjoint nets with identical names.<br>• SPECIFIEDNAMES — Virtually connect disjoint nets that are in a specified list. The list is defined by cmn.virtualConnect.netNames.value.<br>**Options** page, Virtual Connect section | [ALLNAMES SPECIFIEDNAMES]<br>()<br>{ALLNAMES} |

# xACT Runset Options

The Calibre Interactive xACT-specific runset options have the prefix pex.

General runset options are prefixed with cmn and are described in "General Runset Options for Calibre Interactive xACT" on page 966. See "Calibre Interactive xACT Runset File Format" on page 965 for important information.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table D-2. General Runset Options for Calibre Interactive xACT

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.configPrefs.specified | Checkbox for "Configuration," which enables loading configuration files from the GUI.<br>**Preferences** page, Configuration section | [false true]<br>(Boolean)<br>{false} |
| pex.configPrefs.configFiles.value | The file names of the configuration files.<br>Used when pex.configPrefs.specified = true.<br>**Preferences** page, Configuration section | []<br>()<br>{} |
| pex.envVars.setEnvNamePattern | Specifies the filter field when searching for environment variables.<br>pex.envVars.setEnvNamePattern("*pattern*")<br>**Environment** page | []<br>()<br>{} |
| pex.envVars.setEnv | Specifies an environment variable that is set in the runset. The format is:<br>pex.envVars.setEnv("*name*", "*value*")<br>See the envVars configuration file command.<br>**Environment** page | []<br>()<br>{} |
| pex.envVars.unsetEnv | Specifies an environment variable that is unset in the runset. The format is:<br>pex.envVars.unsetEnv("*name*")<br>See the envVars configuration file command.<br>**Environment** page | []<br>()<br>{} |
| pex.ercdb.resultsCellName.value | Specifies to output hierarchical ERC errors in their context cells.<br>**LVS** page | [true false]<br>(Boolean)<br>{true} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.ercdb.resultsFile.value | Specifies the name of the ERC results database.<br>**LVS** page | []<br>()<br>{erc.results} |
| pex.ercdb.resultsPseudoCells.specified | Boolean controlling pex.ercdb.resultsPsuedoCells.<br>**LVS** page | [false true]<br>(Boolean)<br>{true} |
| pex.ercdb.results.PseudoCells.value | Specifies the handling of pseudocells in the ERC database.<br>**LVS** page | [PSEUDO TOP]<br>()<br>{PSEUDO} |
| pex.erc.execute.specified | Boolean controlling pex.erc.execute.<br>**LVS** page | [false true]<br>()<br>{false} |
| pex.erc.execute.value | Specifies whether to execute ERC operations during the circuit extraction phase.<br>**LVS** page | [YES NO]<br>()<br>{YES} |
| pex.ercMaximumResults.count.value | Specifies the maximum result count for an individual rule check in ERC. | []<br>(positive integer)<br>{1000} |
| pex.ercMaximumResults.specified | Boolean controlling pex.ercMaximumResults.<br>**LVS** page | [false true]<br>()<br>{false] |
| pex.ercMaximumResults.upperLimit.value | Specifies the maximum result count for an individual rule check in ERC.<br>**LVS** page | [MAXIMUMCOUNT MAXIMUMALL]<br>()<br>{MAXIMUMCOUNT} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.ercMaximumVertex.count.value | Specifies the maximum result count for an individual rule check in ERC. Must be greater than or equal to 4.<br>**LVS** page | []<br>(integer)<br>{4096} |
| pex.ercMaximumVertex.specified | Boolean controlling pex.ercMaximumVertex.<br>**LVS** page | [false true]<br>()<br>{false] |
| pex.ercMaximumVertex.upperLimit.value | Specifies the maximum vertex count of any result polygon that ERC writes to the results database.<br>**LVS** page | [MAXIMUMCOUNT MAXIMUMALL]<br>()<br>{MAXIMUMCOUNT} |
| pex.ercSummaryReport.reportAcces.specified | Boolean controlling pex.ercSummaryReport.reportAccess.<br>**LVS** page | [false true]<br>()<br>{false} |
| pex.ercSummaryReport.reportAccess.value | Specifies whether to overwrite or append to previous contents of the summary report.<br>**LVS** page | [REPLACE APPEND]<br>()<br>{REPLACE} |
| pex.ercSummaryReport.reportKeywords.specified | Boolean controlling pex.ercSummaryReport.reportKeyWords.<br>**LVS** page | [false true]<br>()<br>{false} |
| pex.ercSummaryReport.reportKeywords.value | Specifies additional keywords for the ERC Summary Report statement.<br>**LVS** page | [HIER]<br>()<br>{} |
| pex.ercSummaryReport.report.specified | Boolean controlling pex.ercSummaryReport.report.<br>**LVS** page | [false true]<br>()<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.ercSummaryReport.report.value | Specifies the path to a file containing the summary of a ERC run.<br>**LVS** page | []<br>()<br>{erc.summary} |
| pex.ercSummaryReport.reportView AfterRun.value | Specifies whether to automatically view the ERC Summary Report after the run completes.<br>**LVS** page | [false true]<br>()<br>{false} |
| pex.fmtMode.options.fmtAllowGrou ndCoupling | Boolean that specifies to ground all parasitic coupling capacitors.<br>**Outputs** page, Formatter section | [false true]<br>(Boolean)<br>{false} |
| pex.fmtMode.options.fmtInfo | Boolean that specifies to display informational messages from the Calibre xACT formatter.<br>**Outputs** page, Formatter section | [false true]<br>(Boolean)<br>{false} |
| pex.fmtMode.options.fmtWarnings | Boolean that specifies to display warnings from the Calibre xACT formatter.<br>**Outputs** page, Formatter section | [false true]<br>(Boolean)<br>{false} |
| pex.hCells.automatch.value | Boolean that controls the “Match cells by name” checkbox. Adds -automatch to the Calibre nmLVS command line.<br>**Inputs** page, H-Cells section | [false true]<br>(Boolean)<br>{false} |
| pex.hCells.hCellsFile.specified | Boolean controlling pex.hCells.hCellsFile.<br>Adds “-hcell hcell_file”to the Calibre nmLVS command line.<br>**Inputs** page, H-Cells section | [false true]<br>(Boolean)<br>{false} |
| pex.hCells.hCellsFile.value | The hcells file.<br>**Inputs** page, H-Cells section | []<br>()<br>{hcells} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.includes.specified | Boolean controlling pex.includes<br>**Options** page, Include Rules Files section | [false true]<br>(Boolean)<br>{false} |
| pex.includes.ruleFiles.value | The file names of the additional rule files.<br>**Options** page, Include Rules Files section | []<br>(list)<br>{} |
| pex.layout.exportFromLayoutViewer.value | Boolean that controls the “Export from layout viewer” checkbox.<br>**Inputs** page, Layout Path section | [false true]<br>()<br>{false} |
| pex.layout.extraGoldenLayoutFiles.specified | Boolean controlling pex.layout.extraGoldenLayoutFiles<br>**Database** page, Library section | [false true]<br>(Boolean)<br>{false} |
| pex.layout.extraGoldenLayoutFiles.value | The file names of the additional golden layout files.<br>**Database** page, Library section | []<br>(list)<br>{} |
| pex.layout.extraLayoutFiles.specified | Boolean controlling pex.layout.extraLayoutFiles<br>**Database** page, Library section | [false true]<br>(Boolean)<br>{false} |
| pex.layout.extraLayoutFiles.value | The file names of the additional layout files.<br>**Database** page, Library section | []<br>(list)<br>{} |
| pex.layout.format.value | The format of the layout database.<br>**Inputs** page, Layout Path section<br>**Note:** If the layout format is OPENACCESS, see cmn.layout.layoutLibraryPathFDI.value, cmn.layout.topCellLibraryFDI.value, and cmn.layout.topCellViewFDI.value. | [GDSII OASIS OPENACCESS]<br>()<br>{GDSII} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.layout.layoutFile.value | The name of the layout file for GDS or OASIS format layouts.<br>**Inputs** page, Layout Path section | []<br>(environment variable valid)<br>{} |
| pex.layout.topCell.value | The name of the layout top cell. Used for GDS, OASIS, and OPENACCESS input.<br>**Inputs** page, Layout Path section | []<br>()<br>{} |
| pex.layout.topCellLibrary.value | The library name of the layout top cell to run Calibre xACT on. Used only when the GUI is invoked from Cadence Virtuoso and “Export from layout viewer” is checked.<br>**Inputs** page, Layout Path section | []<br>()<br>{} |
| pex.layout.topCellView.value | The cell view name of the layout top cell to run Calibre xACT on. Used only when the GUI is invoked from Cadence Virtuoso and “Export from layout viewer” is checked.<br>**Inputs** page, Layout Path section | []<br>()<br>{} |
| pex.layoutCase.caseSensitive.specified | Boolean controlling pex.layoutCase.caseSensitive.<br>**Options** page, “Layout Case” | [false true]<br>(Boolean)<br>{false} |
| pex.layoutCase.caseSensitive.value | Specifies whether the layout should be processed in a case-sensitive manner. (Layout Case statement)<br>**Options** page, “Layout Case” | [NO YES]<br>()<br>{NO} |
| pex.lvsBlackBoxPort.specified | Boolean controlling pex.lvsBlackBoxPort<br>Available when pex.lvsBox.specified = t rue.<br>**LVS** page, LVS Black Box Port section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.lvsBlackBoxPort.parameters | A list of lists defining the contents of the table for the option "LVS Black Box Port". This table defines parameters for the for the LVS Black Box Port statement.<br>Each sublist defines parameters for one statement. The sublist syntax is:<br>`[ bool, "val1 val2 val3"]`<br>• bool — true or false. Indicates whether the statement is used.<br>• val1 — The original layer that forms the port.<br>• val2 — The text layer.<br>• val3 — The interconnect layer.<br>Both pex.lvsBlackBoxPort.specified and pex.lvsBox.specified must be true for this option to be used.<br>**LVS** page, LVS Black Box Port section | []<br>(list of lists)<br>{} |
| pex.lvsBlackBoxPortDepth.specified | Boolean controlling pex.lvsBlackBoxPortDepth<br>Available when pex.lvsBlackBoxPort.specified = true.<br>**LVS** page, LVS Black Box Port Depth section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.lvsBlackBoxPortDepth.hierarchyLevel.value | Setting for the "Hierarchy Level" option, which specifies the keyword used in the LVS Black Box Port Depth statement.<br>• DEPTHPRIMARY — PRIMARY keyword.<br>• DEPTHALL — ALL keyword.<br>• DEPTHLEVEL — Specify a level with pex.lvsBlackBoxPortDepth.level.value.<br>Used when pex.lvsBox.specified, pex.lvsBlackBoxPort.specified, and pex.lvsBlackBoxPortDepth.are all true.<br>**LVS** page, LVS Black Box Port Depth section | [DEPTHPRIMARY DEPTHALL DEPTHLEVEL]<br>()<br>{DEPTHPRIMARY} |
| pex.lvsBlackBoxPortDepth.level.value | Specifies the hierarchical level for the LVS Black Box Port Depth statement. 0 means top level. Used when pex.lvsBlackBoxPortDepth.hierarchyLevel.value = DEPTHLEVEL.<br>**LVS** page, LVS Black Box Port Depth section | []<br>(non-negative integer)<br>{0} |
| pex.lvsBox.specified | Boolean controlling pex.lvsBox<br>**LVS** page, LVS Box section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.lvsBox.parameters | A list of lists defining the contents of the table for the option “LVS Box”. This table defines parameters for the for the LVS Box statement.<br>Each sublist defines parameters for one statement. The sublist syntax is:<br>`[ bool, "option_string"]`<br>• bool — true or false. Indicates whether the statement is used.<br>• option_string — The parameters for the statement.<br>**LVS** page, LVS Box section | []<br>(list of lists)<br>{} |
| pex.lvsFilterUnusedOption.parameters | Controls the filtering of unused devices.<br>**LVS** page, LVS Filter Unused Option section | []<br>(list)<br>{} |
| pex.lvsFilterUnusedOption.specified | Boolean controlling pex.lvsFilterUnusedOption<br>**LVS** page, LVS Filter Unused Option section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsRecognizeGates.specified | Boolean controlling pex.lvsRecognizeGates.<br>When true, the LVS Recognize Gates statement is written to the control file.<br>**LVS** page, LVS Recognize Gates section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsRecognizeGates.gatesType.value | Specify the types of gates that the tool recognizes. The choices correspond to keywords for the LVS Recognize Gates statement.<br>**LVS** page, LVS Recognize Gates section | [ALL SIMPLE NONE]<br>()<br>{ALL} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.lvsRecognizeGates.mixSubTypes.value | Specifies whether to add the MIX SUBTYPES keyword to the LVS Recognize Gates statement.<br>**LVS** page, LVS Recognize Gates section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsRecognizeGates.xAlso.value | Specifies whether to add the XALSO keyword to the LVS Recognize Gates statement. This includes MOS transistors with X+ subtypes in logic gate recognition.<br>**LVS** page, LVS Recognize Gates section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsReduceGates.mixTypes.value | Specifies whether to add the keyword MIX TYPES to LVS Reduce Split Gates YES.<br>Enabled when pex.lvsReduceGates.splitGates.value = YES and pex.lvsReduceGates.splitGates.specified = true.<br>**LVS** page, Gate Reduction section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsReduceGates.reduceParallelMOS.specified | Boolean controlling pex.lvsReduceGates.reduceParallelMOS<br>**LVS** page, Gate Reduction section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsReduceGates.reduceParallelMOS.value | Specify whether to reduce MOS transistors connected in parallel into single transistors. (LVS Reduce Parallel MOS)<br>NO cannot be specified with pex.lvsReduceGates.splitGates.value = YES.<br>**LVS** page, Gate Reduction section | [YES NO]<br>()<br>{YES} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.lvsReduceGates.sameOrderSplitGates.value | Specifies whether to add the keyword SAME ORDER to LVS Reduce Split Gates YES.<br>Enabled when pex.lvsReduceGates.splitGates.value = YES and pex.lvsReduceGates.splitGates.specified = true.<br>**LVS** page, Gate Reduction section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsReduceGates.semiSplitGates.value | Specifies whether to add the keyword SEMI ALSO to LVS Reduce Split Gates YES.<br>Enabled when pex.lvsReduceGates.splitGates.value = YES and pex.lvsReduceGates.splitGates.specified = true.<br>**LVS** page, Gate Reduction section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsReduceGates.shortEquivalentNodes.specified | Boolean controlling pex.lvsReduceGates.shortEquivalentNodes<br>**LVS** page, Gate Reduction section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsReduceGates.shortEquivalentNodes.value | Specifies the primary keyword for the LVS Short Equivalent Nodes statement.<br>PARALLEL and SPLIT cannot be specified with pex.lvsReduceGates.splitGates.value = YES.<br>**LVS** page, Gate Reduction section | [NO PARALLEL SPLIT]<br>()<br>{NO} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.lvsReduceGates.spAlso.value | Specifies whether to add the keyword SP ALSO to LVS Reduce Split Gates YES.<br>Enabled when pex.lvsReduceGates.splitGates.value = YES and pex.lvsReduceGates.splitGates.specified = true.<br>**LVS** page, Gate Reduction section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsReduceGates.splitGates.specified | Boolean controlling pex.lvsReduceGates.splitGates<br>**LVS** page, Gate Reduction section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsReduceGates.splitGates.value | Specifies the value for LVS Reduce Split Gates. Several sub-options are enabled when this is set to YES.<br>In addition, this setting affects the value of pex.lvsReduceGates.reduceParallelMOS.value and pex.lvsReduceGates.shortEquivalentNodes.value.<br>**LVS** page, Gate Reduction section | [YES NO]<br>()<br>{YES} |
| pex.lvsReduceGates.withinTolerance.value | Specifies whether to add the keyword WITHIN TOLERANCE to LVS Reduce Split Gates YES<br>Enabled when pex.lvsReduceGates.splitGates.value = YES and pex.lvsReduceGates.splitGates.specified = true.<br>**LVS** page, Gate Reduction section | [false true]<br>(Boolean)<br>{false} |
| pex.lvsReportOption.specified | Boolean controlling pex.lvsReportOption<br>**LVS** page, LVS Report Option section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.lvsReportOption.optionKeywords.value | A list specifying the option keywords used for the LVS Report Option statement. Each option is quoted; for example:<br>`pex.lvsReportOption.optionKeywords.value = [ "A", "B" ]`<br>You can specify [“All”] to set all options, however, this greatly increases the report output.<br>**LVS** page, LVS Report Option section | [A B C D E EB EC EO ES F G H I N O P R S V W X BX AV BV CV E1 FX LPE MC NCA NE NOK NP RA RD SP SPE XR]<br>(list)<br>{} |
| pex.pexBA.specified | Boolean controlling pex.pexBA<br>**Options** page, PEX BA section | [false true]<br>(Boolean)<br>{false} |
| pex.pexBA.calibreView.value | Checkbox for “Suppress LPE values from CalibreView netlist”. When enabled, this add the CALIBREVIEW keyword to the PEX BA Mapfile statement.<br>**Options** page, PEX BA section | [false true]<br>(Boolean)<br>{false} |
| pex.pexBA.mapFile.value | The path to the back-annotation map file for the PEX BA Mapfile statement.<br>**Options** page, PEX BA section | []<br>()<br>{} |
| pex.pexCellMap.specified | Boolean controlling pex.pexCellMap<br>**Options** page, CALIBREVIW Cellmap Files section | [false true]<br>(Boolean)<br>{false} |
| pex.pexCellMap.mapFiles.value | The list of cellmap files for Calibre View generation.<br>**Options** page, CALIBREVIW Cellmap Files section | []<br>(list)<br>{} |
| pex.pexCornerCommand.corners.specified | Boolean controlling pex.pexCornerCommand.corners<br>**Outputs** page, Corners checkbox | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexCornerCommand.corners.value | The list of process corners. This adds the -corner option to the Calibre xACT command line. For example:<br>`["typical", "other"]`<br>**Outputs** page, Corners value | []<br>(list)<br>{} |
| pex.pexExcludeNets.layoutRecursiveNets.specified | Boolean controlling pex.pexExcludeNets.layoutRecursiveNets<br>**Options** page, PEX Extract Exclude Nets section, Exclude Recursive Layout Nets. | [false true]<br>(Boolean)<br>{false} |
| pex.pexExcludeNets.layoutRecursiveNets.value | The list of nets for the statement PEX EXTRACT EXCLUDE LAYOUTNAMES RECURSIVE.<br>**Options** page, PEX Extract Exclude Nets section, Exclude Recursive Layout Nets. | []<br>(list)<br>{} |
| pex.pexExcludeNets.layoutToplevelNets.specified | Boolean controlling pex.pexExcludeNets.layoutToplevelNets<br>**Options** page, PEX Extract Exclude Nets section, Exclude Toplevel Layout Nets | [false true]<br>(Boolean)<br>{false} |
| pex.pexExcludeNets.layoutToplevelNets.value | The list of nets for the statement PEX EXTRACT EXCLUDE LAYOUTNAMES TOPLEVEL.<br>**Options** page, PEX Extract Exclude Nets section, Exclude Toplevel Layout Nets | []<br>(list)<br>{} |
| pex.pexExcludeNets.sourceRecursiveNets.specified | Boolean controlling pex.pexExcludeNets.sourceRecursiveNets<br>**Options** page, PEX Extract Exclude Nets section, Exclude Recursive Source Nets. | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexExcludeNets.sourceRecursiveNets.value | The list of nets for the statement PEX EXTRACT EXCLUDE SOURCENAMES RECURSIVE.<br>**Options** page, PEX Extract Exclude Nets section, Exclude Recursive Source Nets. | []<br>(list)<br>{} |
| pex.pexExcludeNets.sourceToplevelNets.specified | Boolean controlling pex.pexExcludeNets.sourceToplevelNets<br>**Options** page, PEX Extract Exclude Nets section, Exclude Toplevel Source Nets. | [false true]<br>(Boolean)<br>{false} |
| pex.pexExcludeNets.sourceToplevelNets.value | The list of nets for the statement PEX EXTRACT EXCLUDE SOURCENAMES TOPLEVEL.<br>**Options** page, PEX Extract Exclude Nets section, Exclude Toplevel Source Nets. | []<br>(list)<br>{} |
| pex.pexExtractFloatingNets.specified | Boolean controlling pex.pexExtractFloatingNets<br>**Options** page, PEX Extract Floating Nets | [false true]<br>(Boolean)<br>{false} |
| pex.pexExtractFloatingNets.ignorePortText.value | When true, adds the IGNORE_PORT_TEXT keyword to the PEX Extract Floating Nets statement.<br>**Options** page, PEX Extract Floating Nets section | [false true]<br>(Boolean)<br>{false} |
| pex.pexExtractFloatingNets.netsType.value | GROUNDED, ALL, and REDUCED specify the primary keyword for the PEX Extract Floating Nets statement.<br>REDUCED DEPTH specifies to use the PEX Extract Floating Nets Reduced Depth statement.<br>**Options** page, PEX Extract Floating Nets section | [“GROUNDED” “ALL” “REDUCED” “REDUCED DEPTH”]<br>()<br>{GROUNDED} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexExtractFloatingNets.reducedDepthType.value | Specifies the primary keyword for the PEX Extract Floating Nets Reduced Depth statement. This is used when pex.pexExtractFloatingNets.netsType.value = “REDUCED DEPTH”.<br>**Options** page, PEX Extract Floating Nets section | [MAX MIN]<br>()<br>{MAX} |
| pex.pexExtractFloatingNets.reducedDepthMaxThreshold.specified | Checkbox for pex.pexExtractFloatingNets.reducedDepthMaxThreshold.<br>**Options** page, PEX Extract Floating Nets section | [false true]<br>(Boolean)<br>{false} |
| pex.pexExtractFloatingNets.reducedDepthMaxThreshold.value | Specifies the optional THRESHOLD value for PEX Extract Floating Nets Reduced Depth statement. Units are femtofarads. This is used with the MAX keyword. (pex.pexExtractFloatingNets.reducedDepthType.value = MAX)<br>Used when pex.pexExtractFloatingNets.reducedDepthMaxThreshold.specified = true.<br>**Options** page, PEX Extract Floating Nets section | [ ]<br>()<br>{1e-4} |
| pex.pexExtractFloatingNets.scaleFactor.specified | Boolean controlling pex.pexExtractFloatingNets.scaleFactor<br>This checkbox is only active when pex.pexExtractFloatingNets.netsType.value = GROUNDED.<br>**Options** page, PEX Extract Floating Nets section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexExtractFloatingNets.scaleFactor.value | The scale factor of the capacitance values between floating and signal nets. The value must be greater than 0.<br>Only used when pex.pexExtractFloatingNets.netsType.value = GROUNDED and pex.pexExtractFloatingNets.scaleFactor.specified = true.<br>**Options** page, PEX Extract Floating Nets section | []<br>()<br>{1.0} |
| pex.pexExtractRGate.specified | Boolean controlling pex.pexExtractRGate<br>**Options** page, PEX Extract Rgate section | [false true]<br>(Boolean)<br>{false} |
| pex.pexExtractRGate.parameters | A list of lists defining the contents of the table for the option “PEX Extract Rgate”. This table defines parameters for the for the PEX Extract Rgate statement.<br>Each sublist defines parameters for one statement. The sublist syntax is:<br>`[ bool, "parameters"]`<br>• bool — true or false. Indicates whether the statement is used.<br>• parameters — A string defining the parameters for the statement.<br>**Options** page, PEX Extract Rgate section | []<br>(list of lists)<br>{} |
| pex.pexExtractTemp.temperatures.specified | Boolean controlling pex.pexExtractTemp.temperatures<br>**Options** page, PEX Extract Temperature | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table D-2. General Runset Options for Calibre Interactive xACT (cont.)

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexExtractTemp.temperatures.value | The temperature(s) in degrees Celsius for the PEX Extract Temperature statement.<br>**Options** page, PEX Extract Temperature | []<br>(list)<br>{25} |
| pex.pexExtractionType.inductanceMode.value | Specifies the type of parasitic inductance to extract:<br>• “” — No inductance extracted<br>• “l” — Self inductance<br>• “lm” — Self and mutual inductance<br>**Outputs** page, Extraction Type, Inductance | [“” “l” “lm”]<br>()<br>{“”} |
| pex.pexExtractionType.rcMode.value | Specifies the type of parasitics to extract:<br>• r — Resistance (R)<br>• c — Capacitance and coupling capacitance (C+CC)<br>• rc — Resistance and capacitance (RC)<br>• rcc — Resistance, capacitance, and coupling capacitance (RCC)<br>• simple — No resistance or capacitance (No R/C)<br>**Outputs** page, Extraction Type, Resistance/Capacitance | [r c rc rcc simple]<br>()<br>{rcc} |
| pex.pexFSBoundaryConditionX.specified | Boolean controlling pex.pexFSBoundaryConditionX<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Type | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSBoundaryConditionX.btype.value | Specifies whether to use the PEX Fieldsolver Boundary X or PEX Fieldsolver Cell_array X statement.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Type | [BOUNDARY CELL_ARRAY]<br>()<br>{BOUNDARY} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSBoundaryConditionX.condition.value | Specifies the primary keywords for the PEX Fieldsolver Boundary X or PEX Fieldsolver Cell_array X statement.<br>For either statement<br>• REFLECTIVE<br>• PERIODIC<br>For Cell_array statements only<br>• POSITIVE REFLECTIVE<br>• NEGATIVE REFLECTIVE<br>• POSITIVE PERIODIC<br>• NEGATIVE PERIODIC<br>• POSITIVE REFLECTIVE NEGATIVE PERIODIC<br>• POSITIVE PERIODIC NEGATIVE REFLECTIVE<br>• POSITIVE GROUND<br>• NEGATIVE GROUND<br>• POSITIVE GROUND NEGATIVE GROUND<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Condition | [see description]<br>()<br>{REFLECTIVE} |
| pex.pexFSBoundaryConditionX.location.specified | Boolean controlling pex.pexFSBoundaryConditionX.location<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Location | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSBoundaryConditionX.location.value | Specifies the keyword defining the location of the cell boundary.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Location | [BY_EXTENT BY_ENCLOSURE]<br>()<br>{BY_EXTENT} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSBoundaryConditionX.max.value | Specifies the maximum coordinate along the orientation in microns for the BY_EXTENT or BY_ENCLOSURE keyword.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Maximum | []<br>()<br>{0} |
| pex.pexFSBoundaryConditionX.min.value | Specifies the minimum coordinate along the orientation in microns for the BY_EXTENT or BY_ENCLOSURE keyword.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Minimum | []<br>()<br>{0} |
| pex.pexFSBoundaryConditionX.negGround.value | Specifies the location of the ground plane relative to the cell in the negative direction in microns.<br>Used when the primary keyword includes NEGATIVE GROUND.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Negative Ground | []<br>()<br>{0} |
| pex.pexFSBoundaryConditionX.open.value | Specifies whether to include the OPEN_BOUNDARY keyword with the PEX Fieldsolver Cell_array X statement. Not used if the primary keyword includes GROUND.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Open Boundary | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSBoundaryConditionX.posGround.value | Specifies the location of the ground plane relative to the cell in the positive direction in microns.<br>Used when the primary keyword includes POSITIVE GROUND.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition X, Positive Ground | []<br>()<br>{0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSBoundaryConditionY.specified | Boolean controlling pex.pexFSBoundaryConditionY<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Type | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSBoundaryConditionY.btype.value | Specifies whether to use the PEX Fieldsolver Boundary Y or PEX Fieldsolver Cell_array Ystatement.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Type | [BOUNDARY CELL_ARRAY]<br>()<br>{BOUNDARY} |
| pex.pexFSBoundaryConditionY.condition.value | Specifies the primary keywords for the PEX Fieldsolver Boundary Y or PEX Fieldsolver Cell_array Y statement.<br>For either statement<br>• REFLECTIVE<br>• PERIODIC<br>For Cell_array statements only<br>• POSITIVE REFLECTIVE<br>• NEGATIVE REFLECTIVE<br>• POSITIVE PERIODIC<br>• NEGATIVE PERIODIC<br>• POSITIVE REFLECTIVE NEGATIVE PERIODIC<br>• POSITIVE PERIODIC NEGATIVE REFLECTIVE<br>• POSITIVE GROUND<br>• NEGATIVE GROUND<br>• POSITIVE GROUND NEGATIVE GROUND<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Condition | [see description]<br>()<br>{REFLECTIVE} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSBoundaryConditionY.location.specified | Boolean controlling pex.pexFSBoundaryConditionY.location<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Location | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSBoundaryConditionY.location.value | Specifies the keyword defining the location of the cell boundary.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Location | [BY_EXTENT BY_ENCLOSURE]<br>()<br>{BY_EXTENT} |
| pex.pexFSBoundaryConditionY.max.value | Specifies the maximum coordinate along the orientation in microns for the BY_EXTENT or BY_ENCLOSURE keyword.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Maximum | []<br>()<br>{0} |
| pex.pexFSBoundaryConditionY.min.value | Specifies the minimum coordinate along the orientation in microns for the BY_EXTENT or BY_ENCLOSURE keyword.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Minimum | []<br>()<br>{0} |
| pex.pexFSBoundaryConditionY.negGround.value | Specifies the location of the ground plane relative to the cell in the negative direction in microns.<br>Used when the primary keyword includes NEGATIVE GROUND.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Negative Ground | []<br>()<br>{0} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSBoundaryConditionY.open.value | Specifies whether to include the OPEN_BOUNDARY keyword with the PEX Fieldsolver Cell_array Y statement. Not used if the primary keyword includes GROUND.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Open Boundary | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSBoundaryConditionY.posGround.value | Specifies the location of the ground plane relative to the cell in the positive direction in microns.<br>Used when the primary keyword includes POSITIVE GROUND.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Y, Positive Ground | []<br>()<br>{0} |
| pex.pexFSBoundaryConditionZ.specified | Boolean controlling pex.pexFSBoundaryConditionZ<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Z, Condition | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSBoundaryConditionZ.condition.value | Specifies the primary keywords for the PEX Fieldsolver Cell_array Z statement.<br>• POSITIVE GROUND<br>• NEGATIVE GROUND<br>• POSITIVE GROUND NEGATIVE GROUND<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Z, Condition | [see description]<br>()<br>{POSITIVE GROUND} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSBoundaryConditionZ.negGround.value | Specifies the location of the ground plane relative to the cell in the negative direction in microns.<br>Used when the primary keyword includes NEGATIVE GROUND.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Z, Negative Ground | []<br>()<br>{0} |
| pex.pexFSBoundaryConditionZ.posGround.value | Specifies the location of the ground plane relative to the cell in the positive direction in microns.<br>Used when the primary keyword includes POSITIVE GROUND.<br>**FieldSolver** page, PEX Fieldsolver Boundary Condition Z, Positive Ground | []<br>()<br>{0} |
| pex.pexFSMExcludeNets.rec200LayoutNames.specified | Boolean controlling pex.pexFSMExcludeNets.rec200LayoutNames<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Recursive Layout net(s) (200) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMExcludeNets.rec200LayoutNames.value | List of nets specified for PEX FIELDSOLVER MODE 200 EXCLUDE_NETS LAYOUTNAMES RECURSIVE<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Recursive Layout net(s) (200) | []<br>(list)<br>{} |
| pex.pexFSMExcludeNets.rec200SourceNames.specified | Boolean controlling pex.pexFSMExcludeNets.rec200SourceNames<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Recursive Source net(s) (200) | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSMExcludeNets.rec200SourceNames.value | List of nets specified for PEX FIELDSOLVER MODE 200 EXCLUDE_NETS SOURCENAMES RECURSIVE.<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Recursive Source net(s) (200) | []<br>(list)<br>{} |
| pex.pexFSMExcludeNets.rec600LayoutNames.specified | Boolean controlling pex.pexFSMExcludeNets.rec600LayoutNames<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Recursive Layout net(s) (600) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMExcludeNets.rec600LayoutNames.value | List of nets specified for PEX FIELDSOLVER MODE 600 EXCLUDE_NETS LAYOUTNAMES RECURSIVE.<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Recursive Layout net(s) (600) | []<br>(list)<br>{} |
| pex.pexFSMExcludeNets.rec600SourceNames.specified | Boolean controlling pex.pexFSMExcludeNets.rec600SourceNames<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Recursive Source net(s) (600) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMExcludeNets.rec600SourceNames.value | List of nets specified for PEX FIELDSOLVER MODE 600 EXCLUDE_NETS SOURCENAMES RECURSIVE.<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Recursive Source net(s) (600) | []<br>(list)<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSMExcludeNets.top200LayoutNames.specified | Boolean controlling pex.pexFSMExcludeNets.top200LayoutNames<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Toplevel Layout net(s) (200) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMExcludeNets.top200LayoutNames.value | List of nets specified for PEX FIELDSOLVER MODE 200 EXCLUDE_NETS LAYOUTNAMES TOPLEVEL.<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Toplevel Layout net(s) (200) | []<br>(list)<br>{} |
| pex.pexFSMExcludeNets.top200SourceNames.specified | Boolean controlling pex.pexFSMExcludeNets.top200SourceNames<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Toplevel Source net(s) (200) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMExcludeNets.top200SourceNames.value | Extract these source nets with 200 accuracy mode<br>List of nets specified for PEX FIELDSOLVER MODE 200 EXCLUDE_NETS SOURCENAMES TOPLEVEL.<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Toplevel Source net(s) (200) | []<br>(list)<br>{} |
| pex.pexFSMExcludeNets.top600LayoutNames.specified | Boolean controlling pex.pexFSMExcludeNets.top600LayoutNames<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Toplevel Layout net(s) (600) | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSMExcludeNets.top600LayoutNames.value | List of nets specified for PEX FIELDSOLVER MODE 600 EXCLUDE_NETS LAYOUTNAMES TOPLEVEL.<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Toplevel Layout net(s) (600) | []<br>(list)<br>{} |
| pex.pexFSMExcludeNets.top600SourceNames.specified | Boolean controlling pex.pexFSMExcludeNets.top600SourceNames<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Toplevel Source net(s) (600) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMExcludeNets.top600SourceNames.value | List of nets specified for PEX FIELDSOLVER MODE 600 EXCLUDE_NETS SOURCENAMES TOPLEVEL.<br>**FieldSolver** page, PEX Field Solver Mode Exclude Nets, Toplevel Source net(s) (600) | []<br>(list)<br>{} |
| pex.pexFSMLayers.layers200.specified | Boolean controlling pex.pexFSMLayers.layers200<br>**FieldSolver** page, PEX Field Solver Mode Layers, Layers (200) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMLayers.layers200.value | List of layers specified for PEX FIELDSOLVER MODE 200 LAYERS.<br>**FieldSolver** page, PEX Field Solver Mode Layers, Layers (200) | []<br>(list)<br>{} |
| pex.pexFSMLayers.layers600.specified | Boolean controlling pex.pexFSMLayers.layers600<br>**FieldSolver** page, PEX Field Solver Mode Layers, Layers (600) | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSMLayers.layers600.value | List of layers specified for PEX FIELDSOLVER MODE 600 LAYERS.<br>**FieldSolver** page, PEX Field Solver Mode Layers, Layers (600) | []<br>(list)<br>{} |
| pex.pexFSMNets.rec200LayoutNames.specified | Boolean controlling pex.pexFSMNets.rec200LayoutNames<br>**FieldSolver** page, PEX Field Solver Mode Nets, Recursive Layout net(s) (200) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMNets.rec200LayoutNames.value | List of nets specified for PEX FIELDSOLVER MODE 200 NETS LAYOUTNAMES RECURSIVE.<br>**FieldSolver** page, PEX Field Solver Mode Nets, Recursive Layout net(s) (200) | []<br>(list)<br>{} |
| pex.pexFSMNets.rec200SourceNames.specified | Boolean controlling pex.pexFSMNets.rec200SourceNames<br>**FieldSolver** page, PEX Field Solver Mode Nets, Recursive Source net(s) (200) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMNets.rec200SourceNames.value | List of nets specified for PEX FIELDSOLVER MODE 200 NETS SOURCENAMES RECURSIVE.<br>**FieldSolver** page, PEX Field Solver Mode Nets, Recursive Source net(s) (200) | []<br>(list)<br>{} |
| pex.pexFSMNets.rec600LayoutNames.specified | Boolean controlling pex.pexFSMNets.rec600LayoutNames<br>**FieldSolver** page, PEX Field Solver Mode Nets, Recursive Layout net(s) (600) | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSMNets.rec600LayoutNames.value | List of nets specified for PEX FIELDSOLVER MODE 600 NETS LAYOUTNAMES RECURSIVE.<br>**FieldSolver** page, PEX Field Solver Mode Nets, Recursive Layout net(s) (600) | []<br>(list)<br>{} |
| pex.pexFSMNets.rec600SourceNames.specified | Boolean controlling pex.pexFSMNets.rec600SourceNames<br>**FieldSolver** page, PEX Field Solver Mode Nets, Recursive Source net(s) (600) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMNets.rec600SourceNames.value | List of nets specified for PEX FIELDSOLVER MODE 600 NETS SOURCENAMES RECURSIVE.<br>**FieldSolver** page, PEX Field Solver Mode Nets, Recursive Source net(s) (600) | []<br>(list)<br>{} |
| pex.pexFSMNets.top200LayoutNames.specified | Boolean controlling pex.pexFSMNets.top200LayoutNames<br>**FieldSolver** page, PEX Field Solver Mode Nets, Toplevel Layout net(s) (200) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMNets.top200LayoutNames.value | List of nets specified for PEX FIELDSOLVER MODE 200 NETS LAYOUTNAMES TOPLEVEL.<br>**FieldSolver** page, PEX Field Solver Mode Nets, Toplevel Layout net(s) (200) | []<br>(list)<br>{} |
| pex.pexFSMNets.top200SourceNames.specified | Boolean controlling pex.pexFSMNets.top200SourceNames<br>**FieldSolver** page, PEX Field Solver Mode Nets, Toplevel Source net(s) (200) | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexFSMNets.top200SourceNames.value | List of nets specified for PEX FIELDSOLVER MODE 200 NETS SOURCENAMES TOPLEVEL.<br>**FieldSolver** page, PEX Field Solver Mode Nets, Toplevel Source net(s) (200) | []<br>(list)<br>{} |
| pex.pexFSMNets.top600LayoutNames.specified | Boolean controlling pex.pexFSMNets.top600LayoutNames<br>**FieldSolver** page, PEX Field Solver Mode Nets, Toplevel Layout net(s) (600) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMNets.top600LayoutNames.value | List of nets specified for PEX FIELDSOLVER MODE 600 NETS LAYOUTNAMES TOPLEVEL.<br>**FieldSolver** page, PEX Field Solver Mode Nets, Toplevel Layout net(s) (600) | []<br>(list)<br>{} |
| pex.pexFSMNets.top600SourceNames.specified | Boolean controlling pex.pexFSMNets.top600SourceNames<br>**FieldSolver** page, PEX Field Solver Mode Nets, Toplevel Source net(s) (600) | [false true]<br>(Boolean)<br>{false} |
| pex.pexFSMNets.top600SourceNames.value | List of nets specified for PEX FIELDSOLVER MODE 600 NETS SOURCENAMES TOPLEVEL.<br>**FieldSolver** page, PEX Field Solver Mode Nets, Toplevel Source net(s) (600) | []<br>(list)<br>{} |
| pex.pexFieldSolverMode.accuracy.value | The accuracy mode for Calibre xACT 3D that is specified in the PEX Fieldsolver Mode statement.<br>**Outputs** page, Accuracy Mode | [200 600]<br>()<br>{200} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexGroundLayer.specified | Boolean controlling pex.pexGroundLayer<br>**Options** page, PEX Ground Layer | [false true]<br>(Boolean)<br>{false} |
| pex.pexGroundLayer.gndLayers.value | A list of ground layer names for the PEX Ground Layer statement.<br>**Options** page, PEX Ground Layer | []<br>(list)<br>{} |
| pex.pexGroundLayer.stray.value | Specifies whether to include the STRAY keyword for the PEX Ground Layer statement.<br>**Options** page, PEX Ground Layer | [false true]<br>(Boolean)<br>{false} |
| pex.pexIncludeNets.layoutRecursiveNets.specified | Boolean controlling pex.pexIncludeNets.layoutRecursiveNets<br>**Options** page, PEX Extract Include Nets, Include Recursive Layout Nets | [false true]<br>(Boolean)<br>{false} |
| pex.pexIncludeNets.layoutRecursiveNets.value | The list of nets for the PEX Extract Include LAYOUTNAMES RECURSIVE statement.<br>**Options** page, PEX Extract Include Nets, Include Recursive Layout Nets | []<br>(list)<br>{} |
| pex.pexIncludeNets.layoutToplevelNets.specified | Boolean controlling pex.pexIncludeNets.layoutToplevelNets<br>**Options** page, PEX Extract Include Nets, Include Toplevel Layout Nets | [false true]<br>(Boolean)<br>{false} |
| pex.pexIncludeNets.layoutToplevelNets.value | The list of nets for the PEX Extract Include LAYOUTNAMES TOPLEVEL statement.<br>**Options** page, PEX Extract Include Nets, Include Toplevel Layout Nets | []<br>(list)<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table D-2. General Runset Options for Calibre Interactive xACT (cont.)

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexIncludeNets.sourceRecursiveNets.specified | Boolean controlling pex.pexIncludeNets.sourceRecursiveNets<br>**Options** page, PEX Extract Include Nets, Include Recursive Source Nets | [false true]<br>(Boolean)<br>{false} |
| pex.pexIncludeNets.sourceRecursiveNets.value | The list of nets for the PEX Extract Include SOURCENAMES RECURSIVE statement.<br>**Options** page, PEX Extract Include Nets, Include Recursive Source Nets | []<br>(list)<br>{} |
| pex.pexIncludeNets.sourceToplevelNets.specified | Boolean controlling pex.pexIncludeNets.sourceToplevelNets<br>**Options** page, PEX Extract Include Nets, Include Toplevel Source Nets | [false true]<br>(Boolean)<br>{false} |
| pex.pexIncludeNets.sourceToplevelNets.value | The list of nets for the PEX Extract Include SOURCENAMES TOPLEVEL statement.<br>**Options** page, PEX Extract Include Nets, Include Toplevel Source Nets | []<br>(list)<br>{} |
| pex.pexInductanceFilter.specified | Boolean controlling pex.pexInductanceFilter<br>Controls whether the PEX Inductance Filter statement is used.<br>**Inductance** page, PEX Inductance Filter | [false true]<br>(Boolean)<br>{false} |
| pex.pexInductanceFilter.drtFile.value | The name of the driver/receiver file used for inductance extraction. The filename is specified with PEX Inductance Driver File.<br>Used when pex.pexInductanceFilter.filterType.value = FILE.<br>**Inductance** page, PEX Inductance Filter | []<br>()<br>{driver.xl} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexInductanceFilter.filterTags.specified | Boolean controlling pex.pexInductanceFilter.filterTags<br>**Inductance** page, PEX Inductance Filter | [false true]<br>(Boolean)<br>{false} |
| pex.pexInductanceFilter.filterTags.value | A list of tags for the PEX Inductance Filter statement.<br>Used when pex.pexInductanceFilter.filterType.value = FILE.<br>**Inductance** page, PEX Inductance Filter | []<br>(list)<br>{} |
| pex.pexInductanceFilter.filterType.value | Specifies the primary keyword for the PEX Inductance Filter statement.<br>**Inductance** page, PEX Inductance Filter | [APPROX OFF FILE EXCLUSIVE]<br>()<br>{APPROX} |
| pex.pexInductanceFrequency.specified | Boolean controlling pex.pexInductanceFrequency<br>**Inductance** page, PEX Inductance Frequency | [false true]<br>(Boolean)<br>{false} |
| pex.pexInductanceFrequency.broadBand.value | Checkbox controlling whether the broadband model is used in inductance extraction.<br>Adds the MAXIMUM keyword to the PEX Inductance Frequency statement and enables the use of the PEX Inductance Switch Time statement.<br>**Inductance** page, PEX Inductance Frequency | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexInductanceFrequency.frequency.value | Specifies the frequency in GHz for the PEX Inductance Frequency statement.<br>The MAXIMUM keyword is added if the broadband model is used (pex.pexInductanceFrequency.broadBand.value = true).<br>Used when pex.pexInductanceFrequency.gHzORSeconds.value = GHZ.<br>**Inductance** page, PEX Inductance Frequency | []<br>()<br>{10} |
| pex.pexInductanceFrequency.gHzORSeconds.value | Specifies whether to specify the operating frequency for inductance extraction as a frequency in GHz or as a switch time in seconds.<br>To specify SECONDS, you must also use the broadband model (pex.pexInductanceFrequency.broadBand.value = true).<br>**Inductance** page, PEX Inductance Frequency | [GHZ SECONDS]<br>()<br>{GHZ} |
| pex.pexInductanceFrequency.switchTime.value | The switch time in seconds for the PEX Inductance Switch Time statement.<br>Used with the broadband model when pex.pexInductanceFrequency.gHzORSeconds.value = SECONDS.<br>**Inductance** page, PEX Inductance Frequency | []<br>()<br>{1e-10} |
| pex.pexInductanceMinLength.minLength.specified | Boolean controlling pex.pexInductanceMinLength.minLength<br>**Inductance** page | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexInductanceMinLength.minLength.value | The minimum path length in microns for the PEX Inductance Minlength statement.<br>**Inductance** page | []<br>()<br>{100} |
| pex.pexInductanceMode.extractMode.specified | Boolean controlling pex.pexInductanceMode.extractMode<br>**Inductance** page | [false true]<br>(Boolean)<br>{false} |
| pex.pexInductanceMode.extractMode.value | The inductance extraction mode for the PEX Inductance Mode statement.<br>**Inductance** page | [LOOP PEEC]<br>()<br>{LOOP} |
| pex.pexInductanceRange.distance.specified | Boolean controlling pex.pexInductanceRange.distance<br>**Inductance** page | [false true]<br>(Boolean)<br>{false} |
| pex.pexInductanceRange.distance.value | The search range in microns for the PEX Inductance Range statement.<br>**Inductance** page | []<br>()<br>{0} |
| pex.pexInductanceReturnPath.netType.specified | Boolean controlling pex.pexInductanceReturnPath.netType<br>**Inductance** page | [false true]<br>(Boolean)<br>{false} |
| pex.pexInductanceReturnPath.netType.value | The primary keyword for the PEX Inductance Returnpath statement.<br>**Inductance** page | [GROUND POWER GANDP GANDPAVG NONE]<br>()<br>{GROUND} |
| pex.pexInductanceSameNetMutual.offOn.specified | Boolean controlling pex.pexInductanceSameNetMutual.offOn<br>**Inductance** page | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table D-2. General Runset Options for Calibre Interactive xACT (cont.)

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexInductanceSameNetMutual.offOn.value | The primary keyword for the PEX Inductance Same Net Mutual statement.<br>**Inductance** page | [OFF ON]<br>()<br>{OFF} |
| pex.pexInductanceVictimPath.victimFile.specified | Boolean controlling pex.pexInductanceVictimPath.victimFile<br>**Inductance** page, Use PEX Inductance Victim File | [false true]<br>(Boolean)<br>{false} |
| pex.pexInductanceVictimPath.victimFile.value | Specifies the path to a file containing victim path or victim information for inductance extraction.<br>This file is specified in the control file with an Include statement.<br>**Inductance** page, Use PEX Inductance Victim File | []<br>()<br>{} |
| pex.pexNetlist.busdelim.specified | Boolean controlling xrc.pexNetlist.busdelim. Available when the netlist format is SPEF or DSPF.<br>**Outputs** page, “PEX Netlist” section, BUSDELIM checkbox | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.busdelim.value | Specifies whether to set the BUS_DELIMITER variable in the DSPF or SPEF header.<br>**Outputs** page, “PEX Netlist” section, BUSDELIM | [] () {[]} |
| xrc.pexNetlist.calibrated.value | Specifies whether to report only calibrated layers in the LAYER_MAP selection of the netlist. Available when xrc.pexNetlist.layermap.value = true.<br>**Outputs** page, “PEX Netlist” section, CALIBRATED checkbox | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetListConnectionSection.specified | Boolean controlling pex.pexNetListConnectionSection<br>Available when the netlist format is SPEF or DSPEF.<br>**Options** page, PEX Netlist Connection Section | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetListConnectionSection.instLoc.value | Specifies whether to add the INST_LOC keyword to the PEX Netlist Connection Section statement.<br>Only used when pex.pexNetListConnectionSection.section.value = YES.<br>**Options** page, PEX Netlist Connection Section | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetListConnectionSection.section.value | Specifies the YES or NO keyword for the PEX Netlist Connection Section statement. YES includes nonstandard SPICE statements in the connection section of the netlist.<br>**Options** page, PEX Netlist Connection Section | [YES NO]<br>()<br>{YES} |
| pex.pexNetlistEmirRes.analysis.value | Specifies whether and on which layers to include parasitic resistance width and length information for polygons.<br>• LAYERS — Include on specified original and derived layers<br>• ALL — Include on all layers<br>• NO — Do not include<br>**Options** page, PEX Netlist EMIR_RES section | [LAYERS ALL NO] ()<br>{LAYERS} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlistEmirRes.layers.value | Specifies the layers with polygons to include parasitic resistance width and length information for. Available when pex.pexNetlistEmirRes.analysis.value = LAYERS.<br>**Options** page, PEX Netlist EMIR_RES section | [] (list) {} |
| pex.pexNetlistEmirRes.specified | Boolean controlling pex.pexNetlistEmirRes.<br>**Options** page, PEX Netlist EMIR_RES checkbox | [false true] (Boolean) {false} |
| pex.pexNetListExportPorts.yesNo.specified | Boolean controlling pex.pexNetListExportPorts.yesNo<br>**Options** page | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetListExportPorts.yesNo.value | Specifies YES or NO for the PEX Netlist Export Ports statement.<br>**Options** page | [NO YES]<br>()<br>{NO} |
| pex.pexNetListNoXrefNames.yesNo.specified | Boolean controlling pex.pexNetListNoXrefNames.yesNo<br>**Options** page | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetListNoXrefNames.yesNo.value | Specifies YES or NO for the PEX Netlist Noxref Net Names statement.<br>**Options** page | [YES NO]<br>()<br>{YES} |
| pex.pexNetListShortToplevelPorts.yesNo.specified | Boolean controlling pex.pexNetListShortToplevelPorts.yesNo<br>Available when the netlist format is SPEF, DSPEF, HSPICE, or CALIBREVIEW.<br>**Options** page | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetListShortToplevelPorts.yesNo.value | Specifies YES or NO for the PEX Netlist Short Toplevel Ports statement. Applies to SPEF, DSPEF, HSPICE, and Calibre View netlists.<br>**Options** page | [YES NO]<br>()<br>{YES} |
| pex.pexNetlist.busdelim.specified | Boolean controlling pex.pexNetlist.busdelim<br>Checkbox for the BUSDELIM keyword in the PEX Netlist statement. Available when the netlist format is SPEF or DSPEF.<br>**Outputs** page, PEX Netlist, BUSDELIM | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.busdelim.value | Specifies the string value for the BUSDELIM keyword in the PEX Netlist statement. The delimiter information is added to the SPEF or DSPEF netlist header.<br>**Outputs** page, PEX Netlist, BUSDELIM | []<br>()<br>{[]} |
| pex.pexNetlist.clayer.value | Checkbox for CLAYER, which adds the CLAYER option to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.clocation.value | Checkbox for CLOCATION, which adds the CLOCATION option to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.coordscale.specified | Boolean controlling pex.pexNetlist.coordscale<br>Checkbox for COORDSCALE, which adds the COORDSCALE option to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.coordscale.value | The scaling factor for coordinates, in meters. Specify 1.0e-06 to report coordinates in microns. This is the value for the COORDSCALE keyword in PEX Netlist.<br>**Outputs** page, PEX Netlist | []<br>()<br>{1} |
| pex.pexNetlist.eldo_kr.value | Checkbox for ELDO_KR, which adds the ELDO_KR option to PEX Netlist.<br>Only available when the netlist format is CALIBREVIEW.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.ground.specified | Boolean controlling pex.pexNetlist.ground<br>Checkbox for GROUND, which adds the GROUND option to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.ground.value | The name of the ground node for the GROUND keyword in PEX Netlist..<br>**Outputs** page, PEX Netlist | []<br>()<br>{0} |
| pex.pexNetlist.icellmap.value | Checkbox for ICELLMAP, which adds the ICELLMAP option to PEX Netlist.<br>Only available when the netlist format is CALIBREVIEW.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.layermap.value | Checkbox for LAYERMAP, which adds the LAYERMAP option to PEX Netlist. Used with resistance parameter keywords RWIDTH, RLENGTH, RVIACOUNT, RAREA, RLAYER, and RLOCATION.<br>Only available when the netlist format is DSPF.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.listprobes.value | Checkbox for LISTPROBES, which adds the LISTPROBES option to PEX Netlist.<br>Only available when the netlist format is HSPICE, ELDO, or SPECTRE.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.location.value | Checkbox for LOCATION, which adds the LOCATION option to PEX Netlist.<br>The LOCATION keyword does not work with the SCHEMATICONLY and SOURCEBASED parameters.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.mapnames.value | Checkbox for MAPNAMES, which adds the MAPNAMES option to PEX Netlist.<br>Only available when the netlist format is SPEF.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.mlayer.value | Checkbox for MLAYER, which adds the MLAYER option to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.multitxtport.value | Checkbox for MULTITXTPORT, which adds the MULTITXTPORT option to PEX Netlist.<br>Only available when the netlist format is DSPF.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.multivalues.value | Checkbox for MULTIVALUES, which adds the MULTIVALUES option to PEX Netlist.<br>Multiple corners must be specified. MULTIVALUES creates a single netlist file with multiple corner values.<br>Only available when the netlist format is SPEF.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.netgeometry.value | Checkbox for “Net Geometry”. Only available when the netlist format is CALIBREVIEW.<br>When enabled, the LOCATION, RLAYER, RLENGTH, RLOCATION, and RWIDTH keywords are added to the PEX Netlist statement. In addition, SVDB output is enabled with the QUERY, XACT, and CCI keywords. The PEX XACT PDB statement is also enabled.<br>Specify this option to backannotate net geometries to the Calibre View.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.netlistFile.value | The filename of the generated parasitic netlist that is specified in the PEX Netlist statement.<br>**Outputs** page, PEX Netlist | []<br>()<br>{} |
| pex.pexNetlist.netlistFileViewAfterRun.value | Checkbox for “View Netlist after extraction run finishes”.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.netlistFormat.value | The format of the generated netlist. This is the keyword specified in the PEX Netlist statement.<br>**Outputs** page, PEX Netlist | [HSPICE DSPF SPEF SPECTRE ELDO CALIBREVIEW SPICE]<br>()<br>{HSPICE} |
| pex.pexNetlist.netlistMode.specified | Boolean controlling pex.pexNetlist.netlistMode<br>Checkbox for “Netlist Compatibilty”.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.netlistMode.value | Adds the specified keyword to the PEX Netlist statement in order to output a netlist compatible with the specified tool.<br>See the PEX Netlist statement to determine which keywords are supported for a particular netlist format and extraction mode.<br>**Outputs** page, PEX Netlist | [ADITRAIL PRIMETIME HSIM ULTRASIM TOTEM QUICKCAP]<br>()<br>{ADITRAIL} |
| pex.pexNetlist.nodelocation.value | Checkbox for NODELOCATION, which adds the NODELOCATION keyword to PEX Netlist.<br>Only available when the netlist format is SPEF.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.noinstancesection.value | Checkbox for NOINSTANCESECTION, which adds the NOINSTANCESECTION keyword to PEX Netlist.<br>Only available when the netlist format is DSPF.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.noinstancex.value | Checkbox for NOINSTANCEX , which adds the NOINSTANCEX keyword to PEX Netlist.<br>Only available when the netlist format is HSPICE, DSPF, SPEF, and SPECTRE.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.notopsubckt.value | Checkbox for NOTOPSUBCKT, which adds the NOTOPSUBCKT keyword to PEX Netlist.<br>Only available when the netlist format is DSPF.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.pindelim.value | The value for the PINDELIM option in the PEX Netlist statement, which specifies the delimiter between instance names and pin names. The allowed values are a colon (:) or a period (.). A colon is the default.<br>Only available when the netlist format is SPEF.<br>**Outputs** page, PEX Netlist | [: .]<br>()<br>{:} |
| pex.pexNetlist.pinnamesep.specified | Checkbox for PINNAMESEP, which adds the PINNAMESEP keyword to PEX Netlist. The required value is specified with pex.pexNetlist.pinnamesep.value.<br>Only available when the netlist format is HSPICE.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.pinnamesep.value | The value for the PINNAMESEP option in PEX Netlist.<br>Used only for the HSPICE netlist format and when pex.pexNetlist.pinnamesep.specified = true.<br>**Outputs** page, PEX Netlist | []<br>()<br>{_} |
| pex.pexNetlist.prune.value | Checkbox for PRUNE, which adds the PRUNE keyword to PEX Netlist.<br>Only available when the netlist format is HSPICE, DSPF, or ELDO.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.ractual.value | Checkbox for RACTUAL, which adds the RACTUAL keyword to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.rarea.value | Checkbox for RAREA, which adds the RAREA keyword to PEX Netlist.<br>Only available when the netlist format is DSPF, HSPICE, and SPEF.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.rcnamed.value | Checkbox for RCNAMED, which adds the RCNAMED keyword to PEX Netlist.<br>Only available when the netlist format is HSPICE, DSPF, or ELDO.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.rlayer.value | Checkbox for RLAYER, which adds the RLAYER keyword to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.rlength.value | Checkbox for RLENGTH, which adds the RLENGTH keyword to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.rlocation.value | Checkbox for RLOCATION, which adds the RLOCATION keyword to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.rscale.specified | Checkbox for RSCALE, which adds the RSCALE keyword to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.rscale.value | The value for the RSCALE option in PEX Netlist.<br>Used when pex.pexNetlist.rscale.specified = true.<br>**Outputs** page, PEX Netlist | []<br>()<br>{1} |
| pex.pexNetlist.rthickness.value | Checkbox for RTHICKNESS, which adds the RTHICKNESS keyword to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.rviacount.value | Checkbox for RVIACOUNT, which adds the RVIACOUNT keyword to PEX Netlist.<br>Not available for the CALIBREVIEW netlist format.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.rwidth.value | Checkbox for RWIDTH, which adds the RWIDTH keyword to PEX Netlist.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.rwidthMin.value | Checkbox for MINIMUM, which adds the MINIMUM option for RWIDTH to PEX Netlist.<br>Only available when RWIDTH is checked (pex.pexNetlist.rwidth.value = true).<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.scale.specified | Checkbox for “Scale”, which adds a scale to the PEX Netlist statement. The scale value is specified with pex.pexNetlist.scale.value.<br>Not available for the SPEF netlist format.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.scale.value | The scale for the PEX Netlist statement. The scale specifies a size multiplier in meters for certain parameters of certain devices.<br>Used only when pex.pexNetlist.scale.specified = true. Not available for the SPEF netlist format.<br>**Outputs** page, PEX Netlist | []<br>()<br>{1} |
| pex.pexNetlist.schematicOnlyFile.specified | Checkbox for “Device Info File” in the PEX Netlist section of the GUI. Only available when pex.pexNetlist.system.value is SOURCENAMES or SCHEMATICONLY.<br>When enabled, the filename is given by pex.pexNetlist.schematicOnlyFile.value.<br>**Outputs** page, PEX Netlist, Device Info File | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.schematicOnlyFile.value | An optional input file that defines device parameters for the PEX Netlist statement.<br>Used only with SOURCENAMES or SCHEMATICONLY, and when pex.pexNetlist.schematicOnlyFile.specified = true.<br>**Outputs** page, PEX Netlist, Device Info File | []<br>()<br>{} |
| pex.pexNetlist.separator.specified | Boolean controlling pex.pexNetlist.separator<br>Checkbox for SEPARATOR, which adds the SEPARATOR keyword to the PEX Netlist statement. The separator string is specified with pex.pexNetlist.separator.value.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.separator.value | The separator string specified for the SEPARATOR option in PEX Netlist. The default is a slash (/), except for the ELDO netlist, which has a default of a period (.).<br>Used only when pex.pexNetlist.separator.specified = true.<br>**Outputs** page, PEX Netlist | [/ . \| ^ # @ _]<br>()<br>{} |
| pex.pexNetlist.shortpinnames.value | Checkbox for SHORTPINNAMES, which adds the SHORTPINNAMES keyword to PEX Netlist.<br>Only available for the HSPICE netlist format.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.singlefile.value | Checkbox for SINGLEFILE, which adds the SINGLEFILE keyword to PEX Netlist.<br>Only available for the HSPICE, ELDO, and SPECTRE netlist formats.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.stringquotes.specified | Boolean controlling pex.pexNetlist.stringquotes<br>Checkbox for STRINGQUOTES, which adds the STRINGQUOTES keyword to PEX Netlist. The value is given by pex.pexNetlist.stringquotes.value.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexNetlist.stringquotes.value | The value for the STRINGQUOTES option for the PEX Netlist statement.<br>Used when pex.pexNetlist.stringquotes.specified = true.<br>**Outputs** page, PEX Netlist | [YES NO]<br>()<br>{YES} |
| pex.pexNetlist.system.value | Specifies how to derive names in the generated netlist. The value is a required keyword in the PEX Netlist statement.<br>**Outputs** page, PEX Netlist | [LAYOUTNAMES SOURCENAMES SOURCEBASED SCHEMATICONLY]<br>()<br>{SOURCENAMES} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexNetlist.tcoeff.value | Checkbox for TCOEFF, which adds the TCOEFF keyword to PEX Netlist.<br>Not available for the SPEF netlist format.<br>**Outputs** page, PEX Netlist | [false true]<br>(Boolean)<br>{false} |
| pex.pexPinOrder.specified | Boolean controlling pex.pexPinOrder<br>Checkbox for PEX Pin Order, which adds the PEX Pin Order statement.<br>**Options** page | [false true]<br>(Boolean)<br>{false} |
| pex.pexPinOrder.allPins.value | Adds the ALLPINS keyword to the PEX Pin Order statement.<br>Used when pex.pexPinOrder.specified = true and pex.pexPinOrder.orderBy.value is set to SOURCE or FILE.<br>**Options** page | [false true]<br>(Boolean)<br>{false} |
| pex.pexPinOrder.netlistFile.value | The filename parameter for the PEX Pin Order statement when FILE is specified.<br>Used when pex.pexPinOrder.specified = true and pex.pexPinOrder.orderBy.value = FILE.<br>**Options** page | []<br>()<br>{} |
| pex.pexPinOrder.noXref.value | Adds the NOXREF keyword to the PEX Pin Order statement.<br>Used when pex.pexPinOrder.specified = true and pex.pexPinOrder.orderBy.value = FILE.<br>**Options** page | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexPinOrder.orderBy.value | The primary keyword for the PEX Pin Order statement.<br>Used when pex.pexPinOrder.specified = true.<br>**Options** page | [LAYOUT SOURCE FILE]<br>()<br>{LAYOUT} |
| pex.pexReduceAnalog.specified | Boolean controlling pex.pexReduceAnalog<br>Checkbox for PEX Reduce Analog.<br>**Options** page | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceAnalog.delayError.specified | Checkbox for “Timing Delay Threshold”, which adds the DELAY_ERROR keyword to the PEX Reduce Analog statement.<br>The value is specified by pex.pexReduceAnalog.delayError.value.<br>**Options** page | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceAnalog.delayError.value | The timing delay threshold in seconds for the DELAY_ERROR keyword in PEX Reduce Analog statement.<br>Used when pex.pexReduceAnalog.delayError.specified = true.<br>**Options** page | []<br>()<br>{4.9999999999999999e-13} |
| pex.pexReduceAnalog.noiseError.specified | Checkbox for “Noise Threshold”, which adds the NOISE_ERROR keyword to the PEX Reduce Analog statement.<br>The value is specified by pex.pexReduceAnalog.noiseError.value.<br>**Options** page | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexReduceAnalog.noiseError.value | The noise ratio threshold for the NOISE_ERROR keyword in PEX Reduce Analog statement.<br>Used when pex.pexReduceAnalog.noiseError.specified = true.<br>**Options** page | []<br>()<br>{0.01} |
| pex.pexReduceAnalog.yesNo.value | The primary keyword for the PEX Reduce Analog statement.<br>Used when pex.pexReduceAnalog.specified = true.<br>**Options** page | [NO YES]<br>()<br>{YES} |
| pex.pexReduceCC.absoluteORratio.value | When both ABSOLUTE and RATIO are specified for PEX Reduce CC, specifies how combine the conditions.<br>Used when pex.pexReduceCC.absoluteThreshold.specified and pex.pexReduceCC.ratioPercent.specified are both true.<br>**Options** page | [AND OR]<br>()<br>{AND} |
| pex.pexReduceCC.absoluteThreshold.specified | Checkbox for “Absolute Threshold”, which adds the ABSOLUTE keyword to PEX Reduce CC.<br>The threshold is given by pex.pexReduceCC.absoluteThreshold.value.<br>**Options** page, PEX Reduce CC | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceCC.absoluteThreshold.value | The “Absolute Threshold” value for the PEX Reduce CC statement.<br>Used when pex.pexReduceCC.absoluteThreshold.specified = true.<br>**Options** page, PEX Reduce CC | []<br>()<br>{3} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexReduceCC.ratioPercent.specified | Checkbox for “Ratio Percent”, which adds the RATIO keyword to PEX Reduce CC.<br>The percent value is given by pex.pexReduceCC.ratioPercent.value.<br>**Options** page, PEX Reduce CC | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceCC.ratioPercent.value | The “Ratio Percent” value for the PEX Reduce CC statement. The value is a number between 0 and 1.<br>Used when pex.pexReduceCC.ratioPercent.specified = true.<br>**Options** page, PEX Reduce CC | []<br>()<br>{1} |
| pex.pexReduceCC.scale.specified | Checkbox for “Scale”, which adds the SCALE keyword to PEX Reduce CC.<br>The value is given by pex.pexReduceCC.scale.value.<br>**Options** page, PEX Reduce CC | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceCC.scale.value | The “Scale” value for the PEX Reduce CC statement.<br>Used when pex.pexReduceCC.scale.specified = true.<br>**Options** page, PEX Reduce CC | []<br>()<br>{1} |
| pex.pexReduceDigital.specified | Boolean controlling pex.pexReduceDigital<br>Checkbox for PEX Reduce Digital, which adds the PEX Reduce Digital statement.<br>**Options** page, PEX Reduce Digital | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table D-2. General Runset Options for Calibre Interactive xACT (cont.)

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexReduceDigital.delayError.specified | Checkbox for “Timing Delay Threshold”, which adds the DELAY_ERROR keyword to the PEX Reduce Digital statement.<br>The value is specified by pex.pexReduceDigital.delayError.value.<br>**Options** page, PEX Reduce Digital | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceDigital.delayError.value | The timing delay threshold in seconds for the DELAY_ERROR keyword in PEX Reduce Digital statement.<br>Used when pex.pexReduceDigital.delayError.specified = true.<br>**Options** page, PEX Reduce Digital | []<br>()<br>{9.999999999999998e-13} |
| pex.pexReduceDigital.noiseError.specified | Checkbox for “Noise Threshold”, which adds the NOISE_ERROR keyword to the PEX Reduce Digital statement.<br>The value is specified by pex.pexReduceDigital.noiseError.value.<br>**Options** page, PEX Reduce Digital | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceDigital.noiseError.value | The noise error threshold for the NOISE_ERROR keyword in PEX Reduce Digital statement.<br>Used when pex.pexReduceDigital.noiseError.specified = true.<br>**Options** page, PEX Reduce Digital | []<br>()<br>{0.029999999999999999} |
| pex.pexReduceDigital.yesNo.value | The primary keyword for the PEX Reduce Digital statement.<br>**Options** page, PEX Reduce Digital | [NO YES]<br>()<br>{YES} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexReduceMinCap.remove.specified | Boolean controlling pex.pexReduceMinCap.remove<br>Checkbox for the REMOVE keyword in PEX Reduce Mincap.<br>**Options** page, PEX Reduce MinCap | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceMinCap.remove.value | The threshold value for the REMOVE keyword in PEX Reduce Mincap.<br>Used when pex.pexReduceMinCap.remove.specified = true.<br>**Options** page, PEX Reduce MinCap | []<br>()<br>{} |
| pex.pexReduceMinMutual.threshold.specified | Boolean controlling pex.pexReduceMinMutual.threshold<br>Checkbox for the PEX Reduce MinMutual statement. The required value is given by pex.pexReduceMinMutual.threshold.value.<br>**Inductance** page, PEX Reduce MinMutual | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceMinMutual.threshold.value | The threshold value for the PEX Reduce Minmutual statement. The value must be greater than zero and less than one (0 < K < 1)<br>Used when pex.pexReduceMinMutual.threshold.specified = true.<br>**Inductance** page, PEX Reduce MinMutual | []<br>()<br>{} |
| pex.pexReduceMinRes.combine.specified | Checkbox for “Combine”, which adds the COMBINE keyword for PEX Reduce Minres. The required value is given by pex.pexReduceMinRes.combine.value.<br>**Options** page, PEX Reduce MinRes | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexReduceMinRes.combine.value | Threshold value for the COMBINE keyword of PEX Reduce Minres.<br>Used when pex.pexReduceMinRes.combine.specified = true.<br>**Options** page, PEX Reduce MinRes | []<br>()<br>{} |
| pex.pexReduceMinRes.layerwise.value | Checkbox for “Layerwise”, which adds the LAYERWISE keyword to PEX Reduce Minres.<br>Available when pex.pexReduceMinRes.combine.specified = true.<br>**Options** page, PEX Reduce MinRes | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceMinRes.shortres.specified | Checkbox for “Short”, which adds the SHORT keyword for PEX Reduce Minres. The required value is given by pex.pexReduceMinRes.shortres.value.<br>**Options** page, PEX Reduce MinRes | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceMinRes.shortres.value | Threshold value for the SHORT keyword of PEX Reduce Minres.<br>Used when pex.pexReduceMinRes.shortres.specified = true.<br>**Options** page, PEX Reduce MinRes | []<br>()<br>{} |
| pex.pexReduceMinRes.yesNo.value | Setting for “Apply Reduction”, which specifies the primary keyword for the PEX Reduce Minres statement.<br>**Options** page, PEX Reduce MinRes | [YES NO]<br>()<br>{YES} |
| pex.pexReduceROnly.specified | Checkbox for the PEX Reduce ROnly statement.<br>Available for R-only extraction.<br>**Options** page, PEX Reduce ROnly | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexReduceROnly.reductionBy.specified | Checkbox for "Further Reduction by", which enables an optional keyword for PEX Reduce ROnly. The keyword is specified by pex.pexReduceROnly.reductionBy.value.<br>Available when pex.pexReduceROnly.specified = true.<br>**Options** page, PEX Reduce ROnly | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceROnly.reductionBy.value | The value for the "Further Reduction by" setting, which specifies an optional secondary keyword for the PEX Reduce ROnly statement.<br>Available when pex.pexReduceROnly.reductionBy.specified = true.<br>**Options** page, PEX Reduce ROnly | [SPARSIFY AGGRESSIVE]<br>()<br>{SPARSIFY} |
| pex.pexReduceTicer.specified | Checkbox for "PEX Reduce Ticer Reduction", which add the PEX Reduce Ticer statement.<br>**Options** page, PEX Reduce Ticer Reduction | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceTicer.frequency.value | The frequency in Hertz for PEX Reduce Ticer, expressed using scientific notation.<br>**Options** page, PEX Reduce Ticer Reduction | []<br>()<br>{1e+13} |
| pex.pexReduceTicer.maxDegree.specified | Checkbox for "Max Degree", which adds the MAXDEG keyword to the PEX Reduce Ticer statement. The value is given by pex.pexReduceTicer.maxDegree.value.<br>**Options** page, PEX Reduce Ticer Reduction | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexReduceTicer.maxDegree.value | The value for the MAXDEG keyword of PEX Reduce Ticer.<br>Used when pex.pexReduceTicer.maxDegree.specified = true.<br>**Options** page, PEX Reduce Ticer Reduction | []<br>(integer)<br>{2} |
| pex.pexReduceTicer.mergeRange.specified | Checkbox for “Port Merge Range”, which adds the PORTMERGE keyword to the PEX Reduce Ticer statement. The required range value is given by pex.pexReduceTicer.mergeRange.value.<br>**Options** page, PEX Reduce Ticer Reduction | [false true]<br>(Boolean)<br>{false} |
| pex.pexReduceTicer.mergeRange.value | The range value for the PORTMERGE keyword of PEX Reduce Ticer.<br>Used when pex.pexReduceTicer.mergeRange.specified = true.<br>**Options** page, PEX Reduce Ticer Reduction | []<br>()<br>{0} |
| pex.pexReduceTicer.global.value | Checkbox for “Global”, which adds the GLOBAL keyword to the PEX Reduce Ticer statement.<br>**Options** page, PEX Reduce Ticer Reduction | [false true]<br>(Boolean)<br>{false} |
| pex.pexReportPoint2Point.specified | Checkbox for PEX Report Point2Point.<br>**Outputs** page, PEX Report Point2Point | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexReportPoint2Point.calibreView.value | Checkbox for “CALIBREVIEW” for PEX Report Point2Point. Specifies to add the point to point report to the CalibreView netlist.<br>**Outputs** page, PEX Report Point2Point | [false true]<br>(Boolean)<br>{false} |
| pex.pexReportPoint2Point.p2pInputFile.value | The required input file for the PEX Report Point2Point statement.<br>**Outputs** page, PEX Report Point2Point | []<br>()<br>{} |
| pex.pexReportPoint2Point.p2pOutputFile.specified | Checkbox for “Output File” for PEX Report Point2Point. The report is written to the filename specified with pex.pexReportPoint2Point.p2pOutputFile.value.<br>**Outputs** page, PEX Report Point2Point | [false true]<br>(Boolean)<br>{false} |
| pex.pexReportPoint2Point.p2pOutputFile.value | The output file for PEX Report Point2Point.<br>Used when pex.pexReportPoint2Point.p2pOutputFile.specified = true.<br>**Outputs** page, PEX Report Point2Point | []<br>()<br>{} |
| pex.pexReportPoint2Point.p2pSamePort.specified | Checkbox for SAMEPORT, which enables the SAMEPORT option for PEX Report Point2Point. The complete keyword value is specified with pex.pexReportPoint2Point.p2pSamePort.value.<br>**Outputs** page, PEX Report Point2Point | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexReportPoint2Point.p2pSamePort.value | The keyword and option for SAMEPORT in PEX Report Point2Point.<br>**Outputs** page, PEX Report Point2Point | [“SAMEPORT SHORT” “SAMEPORT OPEN”]<br>()<br>{“SAMEPORT SHORT”} |
| pex.pexReportPoint2Point.useDBU.value | Checkbox for “Use DBU”, which adds the DBU keyword to PEX Report Point2Point.<br>**Outputs** page, PEX Report Point2Point | [false true]<br>(Boolean)<br>{false} |
| pex.pexSupply.pexGroundLayoutNames.specified | Checkbox for “PEX Ground Layout Nets”, which adds the PEX Ground statement with the LAYOUT keyword. Used in inductance extraction.<br>The ground nets are specified with pex.pexSupply.pexGroundLayoutNames.value.<br>**Inductance** page, PEX Power Supply Nets section | [false true]<br>(Boolean)<br>{false} |
| pex.pexSupply.pexGroundLayoutNames.value | A list of ground net names for the PEX Ground statement with the LAYOUT keyword.<br>Used when pex.pexSupply.pexGroundLayoutNames.specified = true.<br>**Inductance** page, PEX Power Supply Nets section | []<br>(list)<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexSupply.pexGroundSourceNames.specified | Checkbox for "PEX Ground Source Nets", which adds the PEX Ground statement with the SOURCE keyword. Used in inductance extraction.<br>The ground nets are specified with pex.pexSupply.pexGroundSourceNames.value.<br>**Inductance** page, PEX Power Supply Nets section | [false true]<br>(Boolean)<br>{false} |
| pex.pexSupply.pexGroundSourceNames.value | A list of ground net names for the PEX Ground statement with the SOURCE keyword.<br>Used when pex.pexSupply.pexGroundSourceNames.specified = true.<br>**Inductance** page, PEX Power Supply Nets section | []<br>(list)<br>{} |
| pex.pexSupply.pexPowerLayoutNames.specified | Checkbox for "PEX Power Layout Nets", which adds the PEX Power statement with the LAYOUT keyword. Used in inductance extraction.<br>The power nets are specified with pex.pexSupply.pexPowerLayoutNames.value.<br>**Inductance** page, PEX Power Supply Nets section | [false true]<br>(Boolean)<br>{false} |
| pex.pexSupply.pexPowerLayoutNames.value | A list of power net names for the PEX Power statement with the LAYOUT keyword.<br>Used when pex.pexSupply.pexPowerLayoutNames.specified = true.<br>**Inductance** page, PEX Power Supply Nets section | []<br>(list)<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexSupply.pexPowerSourceNames.specified | Checkbox for "PEX Power Source Nets", which adds the PEX Source statement with the SOURCE keyword. Used in inductance extraction.<br>The power nets are specified with pex.pexSupply.pexPowerSourceNames.value.<br>**Inductance** page, PEX Power Supply Nets section | [false true]<br>(Boolean)<br>{false} |
| pex.pexSupply.pexPowerSourceNames.value | A list of power net names for the PEX POWER statement with the SOURCE keyword.<br>Used when pex.pexSupply.pexPowerSourceNames.specified = true.<br>**Inductance** page, PEX Power Supply Nets section | []<br>(list)<br>{} |
| pex.pexXactControlFile.controlFile.value | The filename for the PEX XACT Control statement.<br>Used when pex.pexXactControlFile.specified = true.<br>**Options** page, PEX Xact Control File | []<br>()<br>{} |
| pex.pexXactControlFile.specified | Boolean controlling pex.pexXactControlFile<br>**Options** page, PEX Xact Control File | [false true]<br>(Boolean)<br>{false} |
| pex.pexXactCorner.specified | Boolean controlling pex.pexXactCorner<br>**Outputs** page, PEX Xact Corner | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexXactCorner.parameters | Specifies the parameters in the table for the "PEX Xact Corner" option. These parameters are used to add the PEX XACT Corner statement. Used when pex.pexXactCorner.specified = true.<br>The parameter value is a list of lists, where each sub-list corresponds to one row in the table and one PEX XACT Corner statement.<br>Each sub-list has this syntax:<br>`[ bool, "corner", "rule_file", "dp_corners", "temperatures"]`<br>• bool — true or false, indicating whether this set of parameters is used.<br>• corner — The process corner.<br>• rule_file — The name of the rule file.<br>• dp_corners — An optional space-separated list of DP corners. At most three can be listed. Enter "" if there are no DP corners.<br>• temperatures — An optional space-separated list of temperatures. Enter "" if temperatures are not listed.<br>**Outputs** page, PEX Xact Corner | []<br>(list)<br>{} |
| pex.pexXactNoXref.instanceSuffix.specified | Boolean controlling pex.pexXactNoXref.instanceSuffix<br>**Options** page, PEX Xact NoXref | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.pexXactNoXref.instanceSuffix.value | The suffix string for PEX XACT Noxref with the INSTANCE_SUFFIX keyword.<br>Used when pex.pexXactNoXref.instanceSuffix.specified = true.<br>**Options** page, PEX Xact NoXref | []<br>()<br>{_noxref} |
| pex.pexXactNoXref.netPrefix.specified | Boolean controlling pex.pexXactNoXref.netPrefix<br>**Options** page, PEX Xact NoXref | [false true]<br>(Boolean)<br>{false} |
| pex.pexXactNoXref.netPrefix.value | The prefix string for PEX XACT Noxref with the NET_PREFIX keyword.<br>Used when pex.pexXactNoXref.netPrefix.specified = true.<br>**Options** page, PEX Xact NoXref | []<br>()<br>{noxref_} |
| pex.pexXcellPrecedence.specified | Checkbox for PEX Xcell Precedence.<br>**Options** page, PEX Xcell Precedence | [false true]<br>(Boolean)<br>{false} |
| pex.pexXcellPrecedence.best.value | Specifies whether to add the BEST keyword to the PEX Xcell Precedence statement.<br>Used when pex.pexXcellPrecedence.specified = true.<br>**Options** page, PEX Xcell Precedence | [false true]<br>(Boolean)<br>{false} |
| pex.pexXcellPrecedence.precedence.value | Specifies the primary keyword for the PEX Xcell Precedence statement.<br>Used when pex.pexXcellPrecedence.specified = true.<br>**Options** page, PEX Xcell Precedence | [STATEMENT FILE]<br>()<br>{STATEMENT} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.recipePrefs.userRecipeFiles.specified | Boolean controlling pex.recipesPrefs.userRecipeFiles.<br>**Rules** page | [false true]<br>(Boolean)<br>{false} |
| pex.recipePrefs.userRecipeFiles.value | Specifies the list of user recipe files.<br>**Rules** page | []<br>()<br>{} |
| pex.reports.lvsReport.value | The filename for the LVS Report statement.<br>Available when Calibre nmLVS is run as part of the parasitic extraction run.<br>**Outputs** page, Report section | []<br>()<br>{lvs.report} |
| pex.reports.lvsReportViewAfterRun.value | Checkbox for “View LVS Report after run finishes”.<br>**Outputs** page, Report section | [false true]<br>(Boolean)<br>{false} |
| pex.reports.pexReport.specified | Checkbox for PEX Report. The report filename is specified by pex.reports.pexReport.value.<br>**Outputs** page, Report section | [false true]<br>(Boolean)<br>{false} |
| pex.reports.pexReport.value | The filename for the PEX Report statement.<br>Used when pex.reports.pexReport.specified = true.<br>**Outputs** page, Report section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.reports.pexReportSystem.value | Specifies the required keyword LAYOUTNAMES or SOURCENAMES for the PEX Report statement.<br>Used when pex.reports.pexReport.specified = true and pex.pexNetlist.system.value is set to SOURCEBASED or SCHEMATICONLY.<br>**Outputs** page, Report section | [LAYOUTNAMES SOURCENAMES]<br>()<br>{SOURCENAMES} |
| pex.reports.pexReportViewAfterRun.value | Checkbox for “View PEX Report after run finishes.”<br>Used when pex.reports.pexReport.specified = true.<br>**Outputs** page, Report section | [false true]<br>(Boolean)<br>{false} |
| pex.rulesFile.value | The path to the rule file.<br>**Rules** page | []<br>()<br>{rules} |
| pex.runDir.value | The run directory.<br>**Rules** page | []<br>()<br>{. (current directory)} |
| pex.runPrefs.stopAfterLVSMismatch.value | Checkbox for “Stop parasitic extraction if LVS indicates a mismatch”.<br>**Preferences** page, Run section | [false true]<br>(Boolean)<br>{false} |
| pex.runSteps.runLvsStep.value | Check box for “Run LVS --- Run Calibre LVS step if PEX Netlist specifies the use of source names.”<br>**Run Control** page, Run Steps section | [true false]<br>(Boolean)<br>{true} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.runsetPrefs.runsetInfo.value | A string providing runset information. This information is displayed in the Load Runset dialog box.<br>**Preferences** page, Runset section, Info | []<br>()<br>{} |
| pex.rve.autoStart.value | Checkbox for "Show Results in RVE." When enabled, the results are displayed automatically in Calibre RVE when the run finishes.<br>**Run Control** page, RVE Options | [true false]<br>(Boolean)<br>{false} |
| pex.rve.startRVEAfterBatch.value | Checkbox for "Start RVE after batch run."<br>**Run Control** page, RVE Options | [false true]<br>(Boolean)<br>{false} |
| pex.selectCheckSettings.checkSortOrder.value | Specifies whether the checks in the Recipe Editor are displayed alphabetically.<br>**Rules** page, Recipe Editor | [SELECTION ALPHA ORDER PRIORITY RES_COUNT]<br>()<br>{ORDER} |
| pex.selectCheckSettings.groupssShowHier.value | Specifies whether the groups in the Recipe Editor are displayed from the top down.<br>**Rules** page, Recipe Editor | [false true]<br>(Boolean)<br>{false} |
| pex.selectCheckSettings.groupsShowTop.value | Specifies whether the groups in the Recipe Editor are displayed alphabetically.<br>**Rules** page, Recipe Editor | [false true]<br>(Boolean)<br>{false} |
| pex.selectCheckSettings.groupsSortOrder | Specifies whether the checks in the Recipe Editor are displayed alphabetically.<br>**Rules** page, Recipe Editor | [SELECTION ALPHA ORDER]<br>()<br>{ORDER} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.source.exportFromSourceViewer.value | Checkbox for “Export from source viewer”.<br>**Inputs** page, Source Path section | [true false]<br>()<br>{false} |
| pex.source.extraSourceFiles.specified | Checkbox for “Additional SPICE Files”.<br>The additional SPICE netlist files are specified with pex.source.extraSourceFiles.value.<br>**Database** page, Library section | [false true]<br>(Boolean)<br>{false} |
| pex.source.extraSourceFiles.value | A list of additional SPICE netlist files.<br>Used when pex.source.extraSourceFiles.specified = true.<br>**Database** page, Library section | []<br>()<br>{} |
| pex.source.extraVerilogSourceFiles.specified | Checkbox for “Additional VERILOG Files”.<br>The additional VERILOG netlist files are specified with pex.source.extraVerilogSourceFiles.value.<br>**Database** page, Library section | [false true]<br>(Boolean)<br>{false} |
| pex.source.extraVerilogSourceFiles.value | A list of additional VERILOG netlist files.<br>Used when pex.source.extraVerilogSourceFiles.specified = true.<br>**Database** page, Library section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.source.format.value | The format of the source netlist input files.<br>• SPICE — SPICE format<br>• VERILOG — VERILOG format<br>• MIXED — A combination of SPICE and VERILOG files.<br>**Inputs** page, Source Path section | [SPICE VERILOG MIXED]<br>()<br>{SPICE} |
| pex.source.sourceFile.value | The path to the top level source netlist file.<br>When pex.source.format.value is SPICE or MIXED, this is the top level SPICE file. When pex.source.format.value is VERILOG, this is the top level VERILOG file. (For MIXED input, the VERILOG file is specified by pex.source.verilogSourceFile.value.)<br>**Inputs** page, Source Path section | []<br>(environment variable valid)<br>{} |
| pex.source.topCell.value | The name of the primary (top) cell or subcircuit of the source netlist. This is the parameter for the Source Primary statement.<br>**Inputs** page, Source Path section | []<br>()<br>{} |
| pex.source.topCellLibrary.value | The library name for the top cell of the source netlist. Used when exporting the source netlist from a schematic design tool such as Cadence Composer.<br>**Inputs** page, Source Path section | []<br>()<br>{} |
| pex.source.topCellView.value | The view name for the top cell of the source netlist. Used when exporting the source netlist from a schematic design tool such as Cadence Composer.<br>**Inputs** page, Source Path section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.source.verilogSourceFile.value | The path to the VERILOG source netlist file when a combination of SPICE and VERILOG netlist input files is used (pex.source.format.value = MIXED)<br>**Inputs** page, Source Path section | []<br>()<br>{} |
| pex.sourceCase.caseSensitive.specified | Checkbox for Source Case.<br>The value for the Source Case statement is given by pex.sourceCase.caseSensitive.value.<br>**Options** page, Source Case | [false true]<br>(Boolean)<br>{false} |
| pex.sourceCase.caseSensitive.value | Primary keyword for the Source Case statement, which specifies whether the source netlist is processed in a case-sensitive manner.<br>Used when pex.sourceCase.caseSensitive.specified = true.<br>**Options** page, Source Case | [NO YES]<br>()<br>{NO} |
| pex.supply.groundNames.specified | Checkbox for “Ground Nets”, which adds the LVS Ground Name statement.<br>The ground net names are specified by pex.supply.groundNames.value.<br>**LVS** page, Power Supply section | [false true]<br>(Boolean)<br>{false} |
| pex.supply.groundNames.value | A list of ground net names for the LVS Ground Name statement.<br>Used when pex.supply.groundNames.specified = true.<br>**LVS** page, Power Supply section | []<br>(list)<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.supply.powerNames.specified | Checkbox for "Power Nets", which adds the LVS Power Name statement.<br>The ground net names are specified by pex.supply.powerNames.value.<br>**LVS** page, Power Supply section | [false true]<br>(Boolean)<br>{false} |
| pex.supply.powerNames.value | A list of power net names for the LVS Power Name statement.<br>Used when pex.supply.powerNames.specified = true.<br>**LVS** page, Power Supply section | []<br>(list)<br>{} |
| pex.svdb.specified | Checkbox for "Mask SVDB," which adds the Mask SVDB Directory statement.<br>This option enables the other options that begin with pex.svdb.<br>**Outputs** page, Mask SVDB section | [true false]<br>(Boolean)<br>{true} |
| pex.svdb.asciixref.value | Checkbox for "Generate ASCII cross-reference files," which adds the IXF, NXF, and SLPH keywords to the Mask SVDB Directory statement.<br>**Outputs** page, Mask SVDB section | [false true]<br>(Boolean)<br>{false} |
| pex.svdb.cci.value | Checkbox for "Generate Calibre Connectivity Interface data," which adds the CCI keyword to the Mask SVDB Directory statement.<br>**Outputs** page, Mask SVDB section | [false true]<br>(Boolean)<br>{false} |
| pex.svdb.dirPath.value | The directory name for the Mask SVDB Directory statement.<br>**Outputs** page, Mask SVDB section | []<br>()<br>{svdb} |
| pex.svdb.genParasiticData.value | Checkbox for "Generate parasitic data for RVE," which adds the PEX XACT PDB YES statement.<br>**Outputs** page, Mask SVDB section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.svdb.pinloc.specified | Checkbox for "Pin Location," which includes the PINLOC or NOPINLOC keyword for the Mask SVDB Directory statement.<br>The parameter value is specified with pex.svdb.pinloc.value.<br>**Outputs** page, Mask SVDB section | [false true]<br>(Boolean)<br>{false} |
| pex.svdb.pinloc.value | Specifies the PINLOC or NOPINLOC keyword for the Mask SVDB Directory statement.<br>Used when pex.svdb.pinloc.specified = true.<br>**Outputs** page, Mask SVDB section | [PINLOC NOPINLOC]<br>()<br>{PINLOC} |
| pex.svdb.query.value | Checkbox for "Generate cross-reference data for RVE," which adds the QUERY keyword to the Mask SVDB Directory statement.<br>**Outputs** page, Mask SVDB section | [false true]<br>(Boolean)<br>{false} |
| pex.svrfIncludes.specified | Checkbox for "Include Rule Statements". The statements are specified with pex.svrfIncludes.svrfStatements.value or pex.svrfIncludes.tvfStatements.value.<br>**Options** page, Include Rule Statements | [false true]<br>(Boolean)<br>{false} |
| pex.svrfIncludes.statementType.value | Available only when the main rule file is TVF.<br>Dropdown selection for "Type", which specifies the language of the included rule file statements. SVRF statements are included within a VERBATIM block in the generated control file.<br>**Options** page, Include Rule Statements | [none SVRF TVF]<br>()<br>{none} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.svrfIncludes.svrfStatements.value | SVRF statements to be included in the control file generated by Calibre Interactive.<br>Used when the main rule file is SVRF, or when the main rule file is TVF and pex.svrfIncludes.statementType.value = SVRF.<br>**Options** page, Include Rule Statements | []<br>()<br>{} |
| pex.svrfIncludes.tvfStatements.value | TVF statements to be included in the control file generated by Calibre Interactive.<br>Used when the main rule file is TVF and pex.svrfIncludes.statementType.value = TVF.<br>**Options** page, Include Rule Statements | []<br>()<br>{} |
| pex.v2LVSLibraries.sPICEIncFiles.specified | Checkbox for “Include SPICE Files”. Files are specified with pex.v2LVSLibraries.sPICEIncFiles.value.<br>**V2LVS** page, Libraries | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSLibraries.sPICEIncFiles.value | List of SPICE files that are included in the v2lvs command line with the -s option.<br>Used when pex.v2LVSLibraries.sPICEIncFiles.specified = true.<br>**V2LVS** page, Libraries section | []<br>(list)<br>{} |
| pex.v2LVSLibraries.sPICELibFiles.specified | Checkbox for “SPICE Library Files”. Files are specified with pex.v2LVSLibraries.sPICELibFiles.value.<br>**V2LVS** page, Libraries section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.v2LVSLibraries.sPICELibFiles.value | List of SPICE files that are included in the v2lvs command line with the -lsr or -lsp option. (See pex.v2LVSOptions.useRangeMode.value)<br>Used when pex.v2LVSLibraries.sPICELibFiles.specified = true.<br>**V2LVS** page, Libraries section | []<br>()<br>{} |
| pex.v2LVSLibraries.vLOGLibFiles.specified | Checkbox for "Verilog Library Files". Files are specified with pex.v2LVSLibraries.vLOGLibFiles.value.<br>**V2LVS** page, Libraries section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSLibraries.vLOGLibFiles.value | List of Verilog files that are included in the v2lvs command line with the -l option.<br>Used when pex.v2LVSLibraries.vLOGLibFiles.specified = true.<br>**V2LVS** page, Libraries section | []<br>()<br>{} |
| pex.v2LVSOptions.addPinNames.specified | Boolean controlling pex.v2LVSOptions.addPinNames<br>Checkbox for "Add pins to each subckt". The pins are specified with pex.v2LVSOptions.addPinNames.value.<br>**V2LVS** page, Options section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSOptions.addPinNames.value | List of pin names that are added with the -addpin argument to the v2lvs command line.<br>Used when pex.v2LVSOptions.addPinNames.specified = true.<br>**V2LVS** page, Options section | []<br>(list)<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.v2LVSOptions.connectUPins.value | Checkbox for "Create numbered nets for unconnected pins," which adds the -n argument to the v2lvs command line.<br>**V2LVS** page, Options section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSOptions.dontConnectPower.value | Checkbox for "Don't connect supply0/supply1 nets to global supply," which adds the -sk argument to the v2lvs command line.<br>**V2LVS** page, Options section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSOptions.keepBackSlash.value | Checkbox for "Preserve backslashes in escaped identifiers," which adds the -b argument to the v2lvs command line.<br>**V2LVS** page, Options section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSOptions.positionalPins.value | Checkbox for "Use positional pin order (don't use $PINS)", which adds the -i argument to the v2lvs command line.<br>**V2LVS** page, Options section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSOptions.translateAlways.value | Checkbox for "Always translate before running Calibre." If this setting is false, the translator does not run if the Verilog files are older than the translated SPICE file.<br>**V2LVS** page, Options section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSOptions.useCBLicense.value | Checkbox for "Prefer Calibre-CB license during license search", which adds the -cb argument to the v2lvs command line.<br>**V2LVS** page, Options section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.v2LVSOptions.useRangeMode.value | Checkbox for "Use aggregate mode for bus bits".<br>Specifies how SPICE library files are added on the v2lvs command line:<br>• true — Use the -lsr argument, for range mode.<br>• false — Use the -lsp argument, for pin mode.<br>**V2LVS** page, Options section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSOptions.warningLevel.value | Specifies the amount of warning level output for the v2lvs translator. This is the value for the -w argument.<br>**V2LVS** page, Options section | [0 1 2 3 4]<br>()<br>{2} |
| pex.v2LVSOptions.warningsAreErrors.value | Checkbox for "Treat warnings as errors", which adds the -werror argument to the v2lvs command line.<br>**V2LVS** page, Options section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSStrings.achars.specified | Checkbox for "Change array delimiters to." The value is given by pex.v2LVSStrings.achars.value.<br>**V2LVS** page, Strings section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSStrings.achars.value | The value for the -a option in the v2lvs command line, which changes the array delimiter from the default []. The first character replaces [, the optional second character replaces ].<br>Used when pex.v2LVSStrings.achars.specified = true.<br>**V2LVS** page, Strings section | []<br>()<br>{} |
| pex.v2LVSStrings.cchars.specified | Checkbox for "Change illegal SPICE characters to." The value is given by pex.v2LVSStrings.cchars.value.<br>**V2LVS** page, Strings section | [false true]<br>(Boolean)<br>{false} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.v2LVSStrings.cchars.value | The value for the -c option in the v2lvs command line. Enter one or two characters enclosed in quotes. To include single quotes in the command line, as recommended, enclose the characters in single quotes, then outer double quotes.<br>Used when pex.v2LVSStrings.cchars.specified = true.<br>**V2LVS** page, Strings section | []<br>()<br>{} |
| pex.v2LVSStrings.pprefix.specified | Checkbox for "Prefix for gate level primitives" The value is given by pex.v2LVSStrings.pprefix.value.<br>**V2LVS** page, Strings section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSStrings.pprefix.value | The value for the -p option in the v2lvs command line, which specifies a prefix to add to Verilog gate level primitive cells.<br>Used when pex.v2LVSStrings.pprefix.specified = true.<br>**V2LVS** page, Strings section | []<br>()<br>{} |
| pex.v2LVSStrings.s0Net.specified | Checkbox for "Connect pins with 0 value to net." The value is given by pex.v2LVSStrings.s0Net.value.<br>**V2LVS** page, Strings section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSStrings.s0Net.value | The value for the -s0 option in the v2lvs command line.<br>Used when pex.v2LVSStrings.s0Net.specified = true.<br>**V2LVS** page, Strings section | []<br>()<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.v2LVSStrings.s1Net.specified | Checkbox for "Connect pins with 1 value to net." The value is given by pex.v2LVSStrings.s1Net.value.<br>**V2LVS** page, Strings section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSStrings.s1Net.value | The value for the -s1 option in the v2lvs command line.<br>Used when pex.v2LVSStrings.s1Net.specified = true.<br>**V2LVS** page, Strings section | []<br>()<br>{} |
| pex.v2LVSStrings.uprefix.specified | Checkbox for "Prefix for unnamed pins and gates." The value is given by pex.v2LVSStrings.uprefix.value.<br>**V2LVS** page, Strings section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSStrings.uprefix.value | The value for the "Prefix for unnamed pins and gates" option, which adds the -u argument to the v2lvs command line.<br>Used when pex.v2LVSStrings.uprefix.specified = true.<br>**V2LVS** page, Strings section | []<br>()<br>{} |
| pex.v2LVSStrings.userOptions.specified | Checkbox for "More v2lvs program options." The value is given by pex.v2LVSStrings.userOptions.value.<br>**V2LVS** page, Strings section | [false true]<br>(Boolean)<br>{false} |
| pex.v2LVSStrings.userOptions.value | A list of options that are included without modification in the v2lvs command line.<br>Used when pex.v2LVSStrings.userOptions.specified = true.<br>**V2LVS** page, Strings section | []<br>(list)<br>{} |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

**Table D-2. General Runset Options for Calibre Interactive xACT (cont.)**

| Name | Description | [Available Values] (Type restrictions) {default} |
|---|---|---|
| pex.xCells.xCellsFile.specified | Checkbox for “x-Cells File”, which adds the -xcell command-line argument. The filename is specified with pex.xCells.xCellsFile.value.<br>**Inputs** page, xCells | [false true]<br>(Boolean)<br>{false} |
| pex.xCells.xCellsFile.value | The name of the xcell file for the -xcell command-line argument.<br>Used when pex.xCells.xCellsFile.specified = true.<br>**Inputs** page, xCells | []<br>()<br>{} |
| pex.xactMode.mode.value | The dropdown selection for “xACT Mode”. The choice determines the command line arguments that are used: -xact, -xact -3d, or -xact -3dselect.<br>**Outputs** page, xACT Mode | [“xACT” “xACT 3D” “xACT 3D Select”]<br>()<br>{xACT} |

# Template Definitions in the Calibre Interactive xACT Runset

File-naming templates for the Calibre Interactive xACT GUI are saved to the runset as function calls.

## Function Syntax

```
pex.setNameTemplate("option", "template_value")
```

## Parameters

- ***option***

  Specifies which template is being defined. The ***option*** can be one of the values listed in the following table. The corresponding runset option is also given—this is the runset option that the resolved template value is saved to.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| ***option*** | **Definition (Corresponding Runset Option)** | **Default** |
|---|---|---|
| nt_rules | The rule file template.<br>(pex.rulesFile.value) | No default |
| nt_runDir | The run directory template.<br>(pex.runDir.value) | No default |
| nt_layFile | The template for the exported layout filename.<br>(pex.layout.layoutFile.value) | %l.calibre.db |
| nt_layLib | The template for the layout library.<br>(pex.layout.topCellLibrary.value) | %L |
| nt_layView | The template for the layout view.<br>(pex.layout.topCellView.value) | %V |
| nt_srcFile | The template for the source filename.<br>(pex.source.sourceFile.value) | %s.src.net |
| nt_srcLib | The template for the source library.<br>(pex.source.topCellLibrary.value) | %S |
| nt_srcView | The template for the source view.<br>(pex.source.topCellView.value) | %W |
| nt_maskDir | The template for the SVDB directory name.<br>(pex.svdb.dirPath.value) | svdb |
| nt_pexReport | The template for the PEX report filename.<br>(pex.reports.pexReport.value) | %l.pex.report |
| nt_lvsReport | The template for the LVS report filename.<br>(pex.reports.lvsReport.value) | %l.lvs.report |
| nt_hCellFile | The template for the hcell file.<br>(pex.hCells.hCellsFile.value) | No default |
| nt_xCellFile | The template for the xcell file.<br>(pex.xCells.xCellsFile.value) | No default |
| nt_netFile | The template for the filename of the extracted parasitic netlist.<br>(pex.pexNetlist.netlistFile.value) | %l.pex.netlist |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

| ***option*** | **Definition (Corresponding Runset Option)** | **Default** |
|---|---|---|
| nt_efLayVwr | The template for the "Export from layout viewer setting". The ***template_value*** is an integer:<br>• 1 — Yes<br>• 0 — No<br>• -1 — Runset<br>(pex.layout.exportFromLayoutViewer.value)<br>**Note:** See "Export from Viewer Preferences" on page 152 for an explanation of the Yes, No, and Runset settings. | 1 |
| nt_efSrcVwr | The template for the "Export from source viewer setting". The ***template_value*** is an integer:<br>• 1 — Yes.<br>• 0 — No<br>• -1 — Runset<br>(pex.source.exportFromSourceViewer.value) | -1 |
| nt_transcriptFile | (cmn.transcriptPrefs.transcriptEchoFile.value) | No default |
| nt_controlFile | (cmn.rulesPrefs.controlFile.value) | No default |
| nt_calibreView | The template for Calibre View. | calibre |

- ***template_value***

  The value of the template, which is a string for most template options. The string value can include replaceable parameters given in the following table. The value can also include an environment variable if the relevant GUI option supports environment variables.

  Unlike the other template options, the ***template_value*** for export settings (nt_efLayVwr and nt_efSrcVwr) is an integer, not a string. The possible values are described in the preceding table of option values.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Table D-3. Replaceable Parameters in Template Values for Calibre Interactive xACT

| Parameter | Definition |
|---|---|
| %l | The primary cell in the layout database. |
| %s | The primary cell in the source netlist. |
| %L | The name of the layout library for OpenAccess databases. |
| %V | The name of the layout view for OpenAccess databases. |
| %S | The name of the source library for OpenAccess databases. |
| %W | The name of the source view for OpenAccess databases. |
| **Replaceable parameters for the output parasitic netlist filename** | |
| %X | The extraction type. %X is replaced by these values:<br>• rcc — R+C+CC extraction<br>• rc — R+C extraction<br>• r — R only extraction<br>• cc — C+CC extraction<br>• norc — No R/C extraction |
| %x | The parasitic netlist format. %x is replaced by these values:<br>• cv — Calibre View netlist<br>• dspf — DSPF format<br>• spi — Eldo format<br>• sp — HSPICE format<br>• scs — SPECTRE format<br>• spef — SPEF format<br>• spice — SPICE format |
| %o | The netlist format option. %o is replaced by these values:<br>• ar — ADiT Rail netlist compatibility<br>• pt — PrimeTime netlist compatibility<br>• hsim — HSIM netlist compatibility<br>• us — ULTRASIM netlist compatibility<br>• totem — TOTEM netlist compatibility<br>• qc — QUICKCAP netlist compatibility<br>• ng — Net Geometry |



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

### Example 1

```
pex.setNameTemplate("nt_layFile", "%l.exported.gds")
```

This sets the template for the layout file. The template is applied when Calibre Interactive xACT is invoked from a design tool. The replaceable parameter %l is replaced with the name of the open cell in the design tool.

If the open cell is opamp, the layout filename on the **Inputs** page is *opamp.exported.gds*. When the runset is saved, this is saved for the corresponding runset option:

```
pex.layout.layoutFile.value = “opamp.exported.gds”
```

### Example 2

The %X, %x, and %o replaceable parameters can be used to indicate the extraction type and netlist format:

```
pex.setNameTemplate("nt_netFile", "%l.pex.%X.%x")
```

For an open cell named opamp, RCC extraction, and a SPEF netlist format, the parasitic netlist file name is *opamp.pex.rcc.spef*. When the runset is saved, this is saved for the corresponding runset option:

```
pex.pexNetlist.netlistFile.value = "opamp.pex.rcc.spef"
```

# Trigger Functions for the Calibre Interactive xACT Runset

Trigger definitions for Calibre Interactive xACT are saved to the runset as an object definition.

### Trigger Definitions: add...Trigger Functions

The commands that define a trigger function in the runset are the same as those that define a trigger function in the configuration file. However, in the runset, the string “tool” in the trigger function name is replace with “pex”.

This is an example of a trigger definition in the runset. The trigger runs before Calibre xACT execution.

```
var preTrigger_2_runXact_trig = pex.addProcessPreTrigger("runXact",
                                                         "preTrigger_2")
preTrigger_2_runXact_trig.command = "echo \"Running on cell %L\""
```

The trigger object is created with the addProcessPreTrigger function, and the trigger command is defined with the command property. See the add ... Trigger command for details.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

## Options for Trigger Execution

You can set several options for post-execution triggers. These options are set as follows in the runset:

| Runset Entry | Definition |
|---|---|
| pex.runPostTriggerInBackground("runXact", true) | Run the xACT post-execution trigger in the background. |
| pex.runPostTriggerOnTerm("runXact", true) | Run the xACT post-execution trigger even if the Calibre xACT run was terminated. |
| pex.remotePostTrigger("runXact", true) | Run the xACT post-execution trigger on the remote host when Calibre is run on a remote host. |
| pex.runPostTriggerInBackground("runLvs", true) | Run the LVS post-execution trigger in the background. |
| pex.runPostTriggerOnTerm("runLVS", true) | Run the LVS post-execution trigger even is the Calibre xACT run was terminated. |
| pex.remotePostTrigger("runLvs", true) | Run the LVS post-execution trigger on the remote host when Calibre is run on a remote host. |

## Example

The trigger runs after Calibre LVS execution and in the shell environment. The function process_LVSreport is defined in the shell environment and take an argument of the LVS report name, given by the %n replaceable parameter. The trigger is run in the background.

```
var postTrigger_1_runLvs_trig = pex.addProcessPostTrigger("runLvs",
                                                          "postTrigger_1")
postTrigger_1_runLvs_trig.command = "process_LVSreport %n"
pex.runPostTriggerInBackground("runLvs", true)
```



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Glossary

### Control File

A system generated text file that controls the Calibre applications that you invoke from Calibre Interactive. Refer to "About Control Files" on page 46 for more information on this file.

### Intermediate Files

Files created by Calibre Interactive at runtime to make it possible to run Calibre using the settings in the Calibre Interactive with an existing rule file. If a layout viewer is open, Calibre Interactive also creates an intermediate file to make it possible to process the data in the layout viewer.

### Name Generation Template

Template for system generated names for the intermediate files that Calibre Interactive creates.

### Preference Files

System generated text files created by Calibre Interactive to save your preferences as defined in Setup Preferences dialog box. Calibre Interactive creates a different preference file for each type of run.

### Rule File

The user supplied primary SVRF rule file for the Calibre run invoked from Calibre Interactive. Certain settings in this file may be overridden or ignored by Calibre Interactive.

### Run Directory

The directory location from which you run the Calibre application and from which all relative pathnames are resolved. By default, Calibre Interactive sets the run directory to your current directory. Changing the run directory may require that you update the paths to input and output files.

### Runset

A *user created* text file created from within Calibre Interactive to store the settings you specify in Calibre Interactive. For more information on runsets, refer to "About Calibre Interactive Runsets" on page 33 for more information on this file.

### Template

see "Name Generation Template"

### Working Directory

The directory from which you invoke Calibre Interactive.



Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

# Index


## — Symbols —

.cgidfmdb, 146
.cgidrcdb, 146
.cgilvsdb, 146
.cgipercdb, 146
.cgipexdb, 147
[], 30
{}, 30
|, 30

## — A —

Area DFM
    output, 236
    *see also*DFM, Calibre Interactive, Area DFM
    specifying, 234
Area DRC
    output, 191
    *see also*DRC, Calibre Interactive, Area DRC
    specifying, 189
Arguments, in Calibre Interactive
    -3dstack, 68
    -batch, 70
    -custom, 69
    -customgui, 69
    -dfm, 68
    -drc, 68
    -gui, 68
    -lvs, 68
    -perc, 68
    -pex, 68
    -runset, 69

## — B —

Backannotation in Virtuoso, 813
-batch switch in Calibre Interactive, 70
Bold words, 29

## — C —

Cadence Composer, integration, 724, 816
Cadence Encounter integration, 817
Cadence Innovus integration, 820
Calibre Fast XOR, 192
Calibre Interactive
    batch operation, 68, 70
    command line, 68
    licenses, 26
    platform support, 26
    runsets, 33
Calibre job, entering and running, 98
Calibre menu
    in Cadence Encounter, 818
    in Cadence Innovus, 821
    in Synopsys IC Compiler, 827
    in Virtuoso, 728
Calibre Post-LVS Trigger, 664
Calibre View
    backannotation, 813
    case sensitivity, 774
    creating, 749
    creating SPECTRE netlist, 790
    delete backannotation, 815
    licenses, 750
    overview, 749
    post-LVS, 751
    post-parasitic extraction, 751
    requirements, 750
    schematic properties, 774
    *see also*Cellmap
    setup dialog box, 777
    setup file, 791, 794
    SKILL functions, 798
    SKILL interface, 750
    types, 751
Calibre xACT parasitic extraction, 295
Calibre xRC parasitic extraction, 295
CALIBRE_HOME variable, 26
Case sensitivity

with schematic properties for CalibreView, 774
Cellmap
intentional device example, 758, 759
intentional device syntax, 756, 758
layer mapping
identifying layers, 771
optimizing, 770
syntax, 762
overview, 751
parasitic device syntax, 774
Check selection
DFM, 109
DRC, 109
ERC, 284
CMP Tab, 220
Color coding
Calibre Interactive, 60
Command line
Calibre Interactive, 68
Control files
about, 46
comparing to rule file, 46, 54
creating, 48
Courier font, 30
Create directory, 60
-custom switch in Calibre Interactive, 69
-customgui switch in Calibre Interactive, 69
Customization commands
CHECK, 613
DEFINE, 617
getCalibreVersionString, 632
getRunsetVarValue, 633
LABEL, 622
runsetVarExists, 635
SEPARATOR, 623
setOKCallback, 638
setOKCallbackError, 640
setRunsetVarValue, 636
SETWINGEOM, 624
VARIABLE, 625
Customization files
coding a master-slave command, 641
controls, 610
defining startup preference, 145
examples, 648
language, 606
load multiple, 655
overview, 588
rule check selection, 615
specifying, 591

## — D —

DBdiff, in Fast XOR, 192
Define
disable preferences, 154
miscellaneous preferences, 156
templates, 148
Delete backannotation, 815
Device signatures in LVS, 286
DFM, Calibre Interactive
area DFM
reporting DFM results in halo region, 236
setting halo size, 235
specifying, 223, 225
using coordinate scaling, 236
CMP tab, 220
DFM Database annotations, 229
enabling DFM Options, 233
interface, 217
limiting DFM results count, 233
limiting rule check results, 233
overview, 217
reporting results in cell space, 234
specifying input data, 221
specifying output data, 227
specifying rule file information, 217
waiver flow, 226
YieldServer script, 231
Double pipes, 30
-drc switch in Calibre Interactive, 68
DRC, Calibre Interactive
area DRC
reporting DRC results in halo region, 190
setting halo size, 190
specifying, 171
using coordinate scaling, 191
FAST XOR, 192
interface, 161

limiting rule check results, 181
options, enabling, 180
overview, 161
preserve cells using RVE waivers, 180
running, 173
specifying a rule file, 163
specifying input data, 168
specifying output data, 173
specifying virtual net connections, 96
waiver flow, 202
DRC,Calibre Interactive
area DRC
specifying, 170

## — E —

Environment variables
CALIBRE_HOME, 26
complete list, 837
custom scripts, 815
defining, 130
in Pyxis, 723
in runsets, 848
MGC_HOME, 26
setting with a trigger, 815
ERC checks, running in LVS, 282
ERC checks, selecting, 284
Execution mode
backup data, 704
computer cluster, 686
distributed (MTflex), 696
Hyper Remote, 704
local host, 682
other cluster option, 694
Platform LSF, 686
remote host, 682
Univa Grid Engine, 691
Execution triggers, 815
External triggers, 673

## — F —

Fast XOR, 192
File browser
create folder, 60
dialog box, 60
multiple file selection, 61
options, 61

## — G —

Graphic al user interface, color coding, 60
Ground node, specifying name, 326
-gui switch in Calibre Interactive, 68

## — H —

Halo
reporting results in halo region, 190, 236
*see also*DFM, Calibre Interactive, Area DFM
*see also*DRC, Calibre Interactive, Area DRC
setting size, 190, 235
Hcell analysis in LVS, 260
Hcells, specifying in LVS, 258
Heavy font, 29
Hierarchy separator character, 327

## — I —

IC Station,*see*Pyxis
Ignoring pins, in Calibre Interactive nmLVS, 269
Inductance extraction
controlling in Calibre Interactive PEX, 343
Input files, for PEX, 296
Intentional device locations, outputting, 327
Intentional device values, outputting, 327
Internal triggers, 659
Invoking
Calibre Interactive
from command line, 66
from layout viewer, 64
Italic font, 29

## — L —

Layer Derivation tree, 127
Layout vs layout, 192
Licenses
Calibre Connectivity Interface, 317
Calibre Interactive, 26
CB license for flat PEX, 323
CB license for RVE, 701
for Calibre View, 750
queueing, 701
timeout, Calibre Interactive, 67
LSF,*see*Platform LSF

-lvs switch in Calibre Interactive, 68
LVS, Calibre Interactive
device signatures, 286
enabling options, 267
Hcell analysis, 260
Hcell data, specifying, 258
inputs, 243
introduction, 237
outputs, 263
rule file, 239
specifying layout and source, 243
Verilog to LVS Spice translator, 254

## — M —

Map Schematic Net in Virtuoso, 728
mgc_calibre_export_layout_oasis_template_file, 737
mgc_calibre_export_layout_template_file, 737
mgc_calibre_export_netlist_template_file, 745
MGC_HOME variable, 26

## — P —

Parasitic capacitor information, 329
Parasitic resistor information, 328
Parasitics
database (PDB), 313
output to RC netlist, 313
specifying in netlist, 338
Parentheses, 30
PDB, 313
PERC, Calibre Interactive
enabling options, 582
Hcell data, specifying, 572
inputs, 563
introduction, 557
outputs, 579
PERC LDL Probe Path, 573
rule file, 560
specifying layout and source, 564
PEX , Ca libre Interactive
outputting supplemental parasitic resistor information, 328
-pex switch in Calibre Interactive, 68
PEX, Calibre Interactive
controlling step execution, 321
extract floating nets, 329
ground layer name, 327
grounding coupling capacitors, 332
including/excluding specific nets, 325
input files, 296
interface, 295
mapfile, 329
outputting intentional device locations, 327
outputting intentional device values, 327
outputting supplemental parasitic capacitor information, 329
overview, 295
Post-LVS trigger, 664
probe points, specifying, 310, 368
resistance parameters, 338
specifying ground node name, 326
specifying hierarchy separator character, 327
specifying parasitics in netlist, 338
top verilog module, 309
Pipes, 30
Platform LSF
LSF_ENVDIR, 687
resource options, 688
specifying, 686
Platform support, 26
Plot schematic net in Virtuoso, 810
Post-execution triggers, 657
Post-LVS Calibre View
overview, 751
procedure, 780
Post-parasitic extraction Calibre View
overview, 751
Pre-execution triggers, 657
Preferences
about, 142
disable, 154
file naming templates, 148
miscellaneous, 156
saving, 143
*see also*Startup preferences
Probe points, specifying, 310, 368

## — Q —

Quotation marks, 30

## — R —

Remote execution
    remote environment, 699
    remote shell, 701
Rule file
    about, 52
    checking syntax, 53
    comparing to control file, 46, 54
    loading values into the GUI, 53
    selecting checks, 53
    viewing and editing, 52
Run control, 681
    *see also*Execution mode
-runset switch in Calibre Interactive, 69
Runsets
    about, 33
    contents, 35
    creating, 848
    loading multiple runsets, 43
    loading on invocation, 39
    option descriptions, 849
    saving all options, 848
    setting default options, 34
    setting in startup preferences, 145
    tagging, 45
    using triggers with, 815

## — S —

Select Checks
    ERC, 282
Setup files
    .cgidfmdb, 146
    .cgidrcdb, 146
    .cgilvsdb, 146
    .cgipercdb, 146
    .cgipexdb, 147
Short isolation
    specifying in Calibre Interactive, 279
Signatures,*see*Device signatures in LVS
Slanted words, 29
Socket connections with Virtuoso, 747
Square parentheses, 30
Startup preferences
    customization file, 145
    runset loading, 145
    saving, 146
Support, platform, 26
Synopsys IC Compiler
    Calibre menu, 827
    integration, 822
    loading interface, 824
Synopsys IC Compiler II
    loading interface, 825

## — T —

Template file, Cadence Virtuoso, 734
Templates, defining file naming, 148
Timeout, Calibre Interactive, 67
Transcript in Calibre Interactive, 100
Triggers
    execution order, 659
    external, 673
    external trigger examples, 678
    external trigger functions, 675
    internal, 659
    internal trigger examples, 668
    overview, 657
    post-execution, 657
    pre-execution, 657
    replaceable parameters, 659
    setting a runset with, 815

## — U —

Underlined words, 30

## — V —

v2lvs (Verilog translator), 254
Verilog module, in PEX, 309
Verilog translator, 254
View
    layer derivation tree, 127
    rule file, 52
    text files,*see*Text files
Virtuoso
    backannotation, 813
    creating interface to Calibre Interactive, 725
    layout export settings, 730
    Map Schematic Net, 728
    plot schematic net, 810
    *see also*Calibre View

socket connections with, 747
strmout, 730
template file, 734

## — W —

Waiver Flow
creating waiver shapes with Calibre Interactive, 205
DFM waiver flow, 226
DRC waiver flow, 173
ERC waiver flow, 262
overview, 202
starting waiver run, 209

## — X —

xRC, batch parasitic extraction tool, 295

## — Y —

YieldServer script, 231

# 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.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.

Note - Viewing PDF files within a web browser causes some links not to function. Use HTML for full navigation.