SIEMENS EDA

Calibre® RealTime Custom User's Manual

Software Version 2021.2 Document Revision 14



Unpublished work.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

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/.

Calibre® RealTime Custom Us	ser's Manual,	v2021.2
-----------------------------	---------------	---------

Table of Contents

Revision History

Chapter 1	
Calibre RealTime Custom Introduction	15
Calibre RealTime Custom Requirements	15 17
Differences Between Calibre nmDRC and Calibre RealTime Custom	18
Terms and Definitions in Calibre RealTime Custom	19
Documentation Syntax Conventions	21
Chapter 2	
Calibre RealTime Custom Basics	23
Halos in Calibre RealTime Custom	24
Halo Usage in Calibre RealTime	24
Inconsistent Results from Run DRC on Edit Mode with Small Halo Size	26
Density Checks in Calibre RealTime Custom	29
Spurious Boundary Results From Density Checks in Calibre RealTime	29
Calibre RealTime Options Dialog Box	31
Database Read Setup in Calibre RealTime Custom	39
Layer Maps in Calibre RealTime	39
Object Map File Format in Calibre RealTime	40
Session Configuration File in Calibre RealTime Custom	42
Location of the Calibre RealTime Session Configuration File	42
Multilevel Session Configuration File Support in Calibre RealTime Custom	44
Environment Variable for Multilevel Session Configuration Files	44
Creating Multilevel Configuration Files in Calibre RealTime Custom	48
Migrating Multilevel Session Configuration Files from an Earlier Release	50
Option and Environment Variable Control in Calibre RealTime Custom	52
Environment Variable Control in Calibre RealTime	53
GetEnvironmentVariable	54
IsEnvironmentVariableDefined	55
SetEnvironmentVariable	56
UnsetEnvironmentVariable	57
Option Control in the Design Environment for Calibre RealTime	58
AddConfiguration	60
DeleteConfiguration	61
GetConfigurationIndependentOption	62
GetConfigurationName	63
GetConfigurationOption	64
GetCurrentConfiguration	68
SaveConfiguration	69 70
SCIC OHITEUTALION COLOR	/U

SetConfigurationIndependentOption	76
GetAPIVersion	77
Environment Variable Settings for Calibre RealTime Input Files	77
License Timeout for Calibre RealTime Custom	78
Debugging Double Patterning Results with Calibre RealTime Custom	79
Viewing Results in the Calibre RealTime Results Window	83
Using Calibre Interactive Settings in a Calibre RealTime Custom Run	87
Specifying a Calibre Interactive Runset in Calibre RealTime	87
Using the Customization GUI with Calibre RealTime	88
DRC Rule Check Comments for Calibre RealTime Custom	90
Specifying the Check Text Override (CTO) File in Calibre RealTime	91
RVE Highlight Index	92
RVE Highlight Color	94
RVE Show Layers	96
RVE Link	99
Check Text Override (CTO) File for Calibre RealTime	101
Chapter 3	
Check Selection Recipes in Calibre RealTime Custom	105
Check Recipes in Calibre RealTime Custom	105
Specifying a Check Recipe	106
Built-In Check Recipes	107
Hiding Built-In Check Recipes	109
Creating a Custom Check Selection Recipe	110
Basic Editing of Check Selection Recipes	111
Advanced Editing of Check Selection Recipes	113
Calibre RealTime Recipe Editor Dialog Box	116
Chapter 4	
Calibre RealTime Custom with Pyxis Layout	125
Running Calibre RealTime with Pyxis Layout	127
Setting the Layer Map File for Calibre RealTime with Pyxis Layout	128
Specifying the Technology Layer Map as the Layer Map for Calibre RealTime with Pyr	_
Layout	
Manually Creating a Layer Map for Calibre RealTime with Pyxis Layout	128
AMPLE Function \$set_tech_layermap()	130
Invoking Pyxis Layout with Calibre RealTime	131
Adding the Calibre RealTime Integrated Toolbar	132
Specifying the Rule File, Check Recipe, and Calibre RealTime Options in Pyxis Layout	133
Editing the Calibre RealTime Check Recipe	134
Toolbars and Dialog Boxes in Calibre RealTime	136
Calibre RealTime Integrated Toolbar (Pyxis Layout)	137
Calibre RealTime Toolbar Run Controls	137
Calibre RealTime Toolbar Highlight Controls	138
Calibre RealTime Toolbar Option Controls	139
Calibre RealTime Options Dialog Box in Pyxis Layout	141
Calibre RealTime Recipe Editor Dialog Box in Pyxis Layout	144
Calibre RealTime DRC Hotkeys in Pyxis Layout	

Chapter 5	
Calibre RealTime Custom with Synopsys Laker3	151
Supported Versions of Synopsys Laker3 for Calibre RealTime	151
Running Calibre RealTime with Synopsys Laker	153
Invoking Synopsys Laker with Calibre RealTime	153
Specifying and Using Multiple Run Configurations in Calibre RealTime (Laker)	154
Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Laker)	155
Running Calibre RealTime in Window Mode with Synopsys Laker	157
Running Calibre RealTime in Run on Edit Mode (Laker)	159
Edit-in-Place in Calibre RealTime with Synopsys Laker	160
Toolbars and Dialog Boxes in Calibre RealTime (Laker)	161
Calibre RealTime Integrated Toolbar (Laker)	161
Calibre RealTime Options Dialog Box (Laker)	165 172
Database Read Setup for Calibre RealTime with Synopsys Laker	172
Specifying the Object Map for Calibre RealTime with Synopsys Laker	174
Approximation of Circles and Ellipses in Calibre RealTime with Synopsys Laker	175
Calibre RealTime Setup with Tcl Commands in Synopsys Laker	175
Enabling the Calibre RealTime Integration in Synopsys Laker	176
Calibre RealTime Configuration File (Laker)	177
Keyboard Shortcuts (Hotkeys) in Calibre RealTime for Synopsys Laker	179
Defining and Loading Keyboard Shortcuts (Hotkeys) in Calibre RealTime for Synopsys L	
Chapter 6 Calibre RealTime Custom in Synopsys Custom Compiler	183
Requirements and Setup for Calibre RealTime in Synopsys Custom Compiler	184
General Requirements for Calibre RealTime in Synopsys Custom Compiler	184
Tcl Script for Calibre Interface to Synopsys Custom Compiler	185
Synopsys Custom Compiler and Calibre RealTime Custom Version Compatibility	186
GDS Layer Map for Calibre RealTime in Synopsys Custom Compiler	186
Invoking and Running Calibre RealTime in Synopsys Custom Compiler	187
Invoking Synopsys Custom Compiler with Calibre RealTime	187
Specifying and Using Multiple Run Configurations in Calibre RealTime with Synopsys	1.00
Custom Compiler	188
Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Custom Compile	er)
190 Punning Calibra PaulTima in Synangya Custom Compiler	192
Running Calibre RealTime in Synopsys Custom Compiler	192
Keyboard Shortcuts in Calibre RealTime for Synopsys Custom Compiler	197
Defining and Loading Keyboard Shortcuts in Calibre RealTime Custom for Synopsys Cus	
Compiler	198
Comp	1,0
Chapter 7	
Calibre RealTime Custom with Cadence Virtuoso	201
Requirements and Setup for Calibre RealTime with Cadence Virtuoso	202
General Requirements for Calibre RealTime with Cadence Virtuoso	202
Environment Setup for Cadence Virtuoso with Calibre RealTime	203

Cadence Virtuoso and Calibre RealTime Version Compatibility	. 204
Setting the Calibre RealTime Version Newer than CALIBRE_HOME	
Invoking and Running Calibre RealTime with Cadence Virtuoso	
Invoking Cadence Virtuoso with Calibre RealTime	
Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)	
Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Virtuoso)	
Running Calibre RealTime with Cadence Virtuoso	
Starting a Batch Calibre Run from Calibre RealTime	
Specifying Highlight Color and Layer Visibility for Calibre RealTime (Virtuoso)	
Edit-in-Place, Descend, and Ascend in Calibre RealTime with Cadence Virtuoso	
Deleting Mentor Libraries When Calibre RealTime is Disabled	
Saving Mentor Libraries to a Specified cds.lib	
Toolbars, Menus, and Dialog Boxes in Calibre RealTime (Virtuoso)	
Calibre RealTime Integrated Toolbar (Virtuoso)	
Calibre RealTime DRC Menu (Virtuoso)	
DRC Run and Highlight Options Menu (Virtuoso)	
Calibre RealTime Custom Options Dialog Box (Virtuoso)	. 230
Batch Tab in the Calibre RealTime Options Dialog Box	. 238
Database Read Setup for Calibre RealTime with Cadence Virtuoso	
GDS Layer Map for Calibre RealTime with Cadence Virtuoso	
Specifying the Layer Map for Calibre RealTime with Cadence Virtuoso	
Specifying the Object Map for Calibre RealTime with Cadence Virtuoso	
Approximation of Circles and Ellipses in Calibre RealTime with Cadence Virtuoso	
Multi-Patterning Support in Calibre RealTime Custom in Cadence Virtuoso	
Calibre RealTime Setup with Cadence SKILL Commands	
SKILL Commands to Get and Set Calibre RealTime Options	
mgc_calibre_realtime_add_configuration	
mgc_calibre_realtime_capture_layer_visibility	
mgc_calibre_realtime_cur_win_trigger	. 254
mgc_calibre_realtime_delete_configuration	
mgc_calibre_realtime_drc_init	
mgc_calibre_realtime_get_api_version	
mgc_calibre_realtime_get_configuration_ids	
mgc_calibre_realtime_get_configuration_id_by_name	
mgc_calibre_realtime_get_configuration_names	
mgc_calibre_realtime_get_configuration_name_by_id	
mgc_calibre_realtime_get_configuration_option	
mgc_calibre_realtime_get_option	
mgc_calibre_realtime_get_options	
mgc_calibre_realtime_restore_layer_visibility	
mgc_calibre_realtime_save_configuration	
mgc_calibre_realtime_set_configuration_option	. 277
mgc_calibre_realtime_set_current_configuration	
mgc_calibre_realtime_set_option	
mgc_calibre_realtime_set_options	
mgc_calibre_realtime_set_show_single_error	. 294
Environment Variable Control in Calibre RealTime for Cadence Virtuoso	
mgc_calibre_realtime_setShellEnvVar	. 296
mgc_calibre_realtime_unsetShellEnvVar	

Table of Contents

mgc_calibre_realtime_getShellEnvVar	298
mgc_calibre_realtime_isShellEnvVarDefined	299
Calibre RealTime Configuration File (Virtuoso)	299
Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso	300
Defining and Loading Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso	301
Troubleshooting Calibre RealTime with Cadence Virtuoso	302
Chapter 8	
Calibre RealTime Custom with Calibre DESIGNrev	305
Requirements for Calibre RealTime with Calibre DESIGNrev	305
Running Calibre RealTime with Calibre DESIGNrev	307
Invoking Calibre DESIGNrev with Calibre RealTime	307
Specifying the Rule File, Check Recipe, and Calibre RealTime Options (DESIGNrev).	308
Running Calibre RealTime in Window, Area, or Cell Mode with Calibre DESIGNrev	310
Running Calibre RealTime in Run on Edit Mode with Calibre DESIGNrev	312
Specifying Highlight Color and Layer Visibility for Calibre RealTime with Calibre	
DESIGNrev	313
Viewing Result Tooltips in Calibre RealTime with Calibre DESIGNrev	315
Toolbars, Menus, and Dialog Boxes in Calibre RealTime (DESIGNrev)	317
Calibre RealTime Integrated Toolbar (DESIGNrev)	317
Verification Menu in Calibre RealTime for Calibre DESIGNrev	320
DRC Run and Highlight Options Menu (DESIGNrev)	320
Calibre RealTime Options Dialog Box (DESIGNrev)	322
Database Read Options in Calibre RealTime with Calibre DESIGNrev	328
Specifying the Layer Map for Calibre RealTime with Calibre DESIGNrev	328
Calibre RealTime Setup with Tcl Commands in Calibre DESIGNrev	329
Calibre RealTime Configuration File (DESIGNrev)	330
Keyboard Shortcuts in Calibre RealTime for Calibre DESIGNrev	330
Appendix A	
Environment Variables in Calibre RealTime Custom	333
Calibre RealTime Environment Variables	333
Appendix B	220
Support for SVRF Statements in Calibre RealTime Custom	339
SVRF Statements with Limited Support in Calibre RealTime	341
Density in Calibre RealTime	341
DFM Copy in Calibre RealTime	342
DFM Text in Calibre RealTime	342
Expand Text in Calibre RealTime	343
Extent Cell in Calibre RealTime	343
Extent Drawn in Calibre RealTime	344
Inside Cell in Calibre RealTime	344
Layout Text in Calibre RealTime	345
Layout Rename Text in Calibre RealTime	345 345
Not Inside Cell in Calibre RealTime	345 346
With Text in Calibre RealTime	340 346
WILL TOAL III CALIULE REALTHIE	J40

Index

Third-Party Information

List of Figures

Figure 2-1. Calibre RealTime Custom Operation With Halo	25
Figure 2-2. Results from Run DRC on Edit Mode with Small Halo Size	27
Figure 2-3. Spurious Density Results in Calibre RealTime Window Run	29
Figure 2-4. Calibre RealTime Options Dialog Box	31
Figure 2-5. Multilevel Configuration Files in Calibre RealTime Custom	47
Figure 2-6. Double Patterning Results in Calibre RealTime Custom	82
Figure 2-7. Double Patterning Fix in Calibre RealTime Custom	83
Figure 3-1. Advanced Recipe Editor Controls	114
Figure 3-2. Recipe Definition Controls	115
Figure 3-3. Edit Recipe Dialog Box Basic Controls	116
Figure 3-4. Edit Recipe Dialog Box Advanced Controls	117
Figure 4-1. Calibre RealTime Toolbar Placement	132
Figure 4-2. Calibre RealTime Integrated Toolbar	137
Figure 4-3. Calibre RealTime Toolbar Run Controls	137
Figure 4-4. Calibre RealTime Toolbar Highlight Controls	138
Figure 4-5. Calibre RealTime Toolbar Option Controls	139
Figure 4-6. Edit Recipe Dialog Box Basic Mode	144
Figure 4-7. Calibre RealTime Recipe Editor Advance Mode	145
	154
Figure 5-2. Selecting a Check Recipe in Calibre RealTime (Laker)	157
Figure 5-3. Calibre RealTime Results in Synopsys Laker	159
Figure 5-4. Calibre RealTime Options Dialog Box (Laker)	165
Figure 6-1. Calibre RealTime in Synopsys Custom Compiler	188
Figure 6-2. Selecting a Check Recipe in Calibre RealTime (Custom Compiler)	191
Figure 6-3. Calibre RealTime Results in Synopsys Custom Compiler	193
ϵ	210
Figure 7-2. Selecting a Check Recipe in Calibre RealTime (Virtuoso)	215
Figure 7-3. Calibre RealTime Results in Cadence Virtuoso	217
Figure 7-4. Calibre RealTime Options Dialog Box (Virtuoso)	230
	308
Figure 8-2. Selecting a Check Recipe in Calibre RealTime (DESIGNrev)	310
Figure 8-3. Calibre RealTime Results in Calibre DESIGNrev	312
Figure 8-4. Result Tooltip in Calibre RealTime with Calibre DESIGNrev	316
Figure 8-5. Calibre RealTime Options Dialog Box (DESIGNrev)	322

List of Tables

Table 1-1. Syntax Conventions	21
Table 2-1. Tabs in the Calibre RealTime Options Dialog Box	32
Table 2-2. Calibre RealTime Options Dialog Box Contents	32
Table 2-3. ObjectType and SubType Values in Calibre RealTime	40
Table 2-4. Multilevel Configuration File Support in Calibre RealTime Custom	44
Table 2-5. Tcl Commands to Control Environment Variables in Calibre RealTime	53
Table 2-6. Tcl Commands to Set Calibre RealTime Options	58
Table 2-7. optionName Values for GetConfigurationOption	64
Table 2-8. optionName and optionValue for SetConfigurationOption	70
Table 2-9. Keyboard Shortcuts in Calibre RealTime Results Window	86
Table 2-10. DRC Rule Check Comments in Calibre RealTime Custom	90
Table 2-11. Specifying DRC Rule Check Comments in Calibre RealTime Custom	90
Table 3-1. Built-in Check Recipes in Calibre RealTime	107
	117
Table 3-3. Edit Recipe Dialog Box Contents — Basic Controls	118
Table 3-4. Edit Recipe Dialog Box Contents — Advanced Controls	119
	120
	125
Table 4-2. Calibre RealTime Toolbar Run Control Options	137
Table 4-3. Calibre RealTime Toolbar Highlight Control Options	138
1	140
Table 4-5. Calibre Options Dialog Box Settings	141
Table 4-6. Calibre RealTime Recipe Editor Action Buttons	145
Table 4-7. Calibre RealTime Recipe Editor Basic Options	147
Table 4-8. Calibre RealTime Recipe Editor Advanced Options	147
Table 4-9. Calibre RealTime DRC Hotkeys in Pyxis Layout	149
Table 5-1. Supported Versions of Synopsys Laker3 with Calibre RealTime	151
	162
	162
1 ' '	163
	166
Table 5-6. Calibre RealTime Options Dialog Box Contents (Laker)	166
Table 5-7. Keyboard Shortcuts in Calibre RealTime for Synopsys Laker	
Table 6-1. Synopsys Custom Compiler and Calibre RealTime Custom Version Compatibilit	y
186	
` ' '	194
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \	194
1	195
Table 6-5. Keyboard Shortcuts in Calibre RealTime Custom for Synopsys Custom Compiler	r
197	

Table 7-1. Cadence Virtuoso and Calibre RealTime Version Compatibility	204
Table 7-2. Run Control Toolbar Selections (Virtuoso)	226
Table 7-3. Highlight Control Toolbar Selections (Virtuoso)	227
Table 7-4. Option Control Toolbar Selections (Virtuoso)	227
Table 7-5. RealTime DRC Menu Contents (Virtuoso)	228
Table 7-6. DRC Run and Highlight Options Menu Contents (Virtuoso)	229
Table 7-7. Tabs in the Calibre RealTime Options Dialog Box (Virtuoso)	230
Table 7-8. Calibre RealTime Options Dialog Box Contents (Virtuoso)	231
Table 7-9. optionName Values for SKILL get commands	263
Table 7-10. optionName Values for Batch Calibre Options for GetConfigurationOption	266
Table 7-11. optionName Values for SKILL get commands	268
Table 7-12. optionName Values for Batch Calibre Options for GetConfigurationOption	271
Table 7-13. optionName and optionValue for SKILL set commands	277
Table 7-14. OptionName and OptionValue for SKILL set commands for Batch operation	281
Table 7-15. optionName and optionValue for SKILL set commands	285
Table 7-16. OptionName and OptionValue for SKILL set commands for Batch operation	289
Table 7-17. Commands to Control Environment Variables in Calibre RealTime for Cadence	e
Virtuoso	295
Table 7-18. Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso	300
Table 8-1. Run Control Toolbar Selections (DESIGNrev)	318
Table 8-2. Highlight Control Toolbar Selections (DESIGNrev)	319
Table 8-3. Option Control Toolbar Selections (DESIGNrev)	319
Table 8-4. Verification Menu Contents (DESIGNrev)	320
Table 8-5. DRC Run and Highlight Options Menu Contents (DESIGNrev)	321
Table 8-6. Tabs in the Calibre RealTime Options Dialog Box (DESIGNrev)	322
Table 8-7. Calibre RealTime Options Dialog Box Contents (DESIGNrev)	323
Table 8-8. Keyboard Shortcuts in Calibre RealTime for Calibre DESIGNrev	330
	333
Table B-1. SVRF Commands with Limited or No Support	339

Chapter 1 Calibre RealTime Custom Introduction

Calibre® RealTime Custom integrates flat Calibre® nmDRC™ with supported layout design tools so that Calibre nmDRC can be run directly from the design tool. Calibre RealTime loads the complete Calibre rule file, but "check recipes" give you control over what checks are executed for each Calibre run. Calibre nmDRC is run on a selected region of the layout. If supported by the design tool, Calibre nmDRC can be run automatically on the region around a design edit.

The output of a Calibre RealTime Custom run is the set of results from the rule checks that were run. These results are highlighted in the design tool, providing immediate feedback if a design change violates a rule check. Only results from the main results database (RDB) are included; auxiliary or side RDBs are not created.

See the following chapters for information on running Calibre RealTime Custom:

- "Calibre RealTime Custom Basics" on page 23
- "Check Selection Recipes in Calibre RealTime Custom" on page 105
- "Calibre RealTime Custom with Pyxis Layout" on page 125
- "Calibre RealTime Custom with Synopsys Laker3" on page 151
- "Calibre RealTime Custom in Synopsys Custom Compiler" on page 183
- "Calibre RealTime Custom with Cadence Virtuoso" on page 201
- "Calibre RealTime Custom with Calibre DESIGNrev" on page 305

The following topics describe the requirements and general concepts of Calibre RealTime Custom:

Calibre RealTime Custom Requirements	15
Calibre RealTime Custom License Options	17
Differences Between Calibre nmDRC and Calibre RealTime Custom	18
Terms and Definitions in Calibre RealTime Custom	19
Documentation Syntax Conventions	21

Calibre RealTime Custom Requirements

Certain requirements for the design tool, environment variables, and Calibre rule file must be met before running Calibre RealTime Custom.

• Design Tool Requirements

Calibre RealTime Custom is supported for the following layout tools:

- Synopsys[®] Laker[®] OpenAccess (see "Supported Versions of Synopsys Laker3 for Calibre RealTime" on page 151).
- o Synopsys® Custom Compiler® (see "Calibre RealTime Custom in Synopsys Custom Compiler" on page 183).
- o Cadence® Virtuoso® with Calibre version 2013.1 or later. See "Requirements and Setup for Calibre RealTime with Cadence Virtuoso" on page 202.
- o Calibre® DESIGNrev™ with Calibre version 2012.3 or later. The integration between Calibre RealTime Custom and Calibre DESIGNrev is automatic; no special user setup is necessary. See "Requirements for Calibre RealTime with Calibre DESIGNrev" on page 305 for specific requirements.
- o Pyxis[™] Layout version 10.0 or later from Siemens Digital Industries Software. The integration between Calibre RealTime Custom and Pyxis Layout is automatic; no special user setup is necessary. See "Calibre RealTime Custom with Pyxis Layout" on page 125 for more information.
- o Tanner L-Edit[™] version 2020.1 Update 5 or later from Siemens Digital Industries Software. See the Calibre RealTime chapter in the *L-Edit User's Manual* for more information.

• Calibre Requirements

- o Calibre version 2011.1 or later; a later version of Calibre may be required depending on the specific design tool.
- The environment variable MGC_HOME or CALIBRE_HOME set to the location of the Calibre software tree. See "Setting the CALIBRE_HOME Environment Variable" in the Calibre Administrator's Guide for details.
- A writable directory for temporary files created by Calibre RealTime Custom. By default the temporary directory is \$MGC_HOME/tmp; to use a different location, define the environment variable MGC_TMPDIR to point to the desired temporary directory.
- o The Calibre RealTime Custom or Calibre RealTime Custom Lite product license. See "Calibre RealTime Custom License Options" on page 17 for more information.
 - Additional Calibre licenses are required for some Calibre features. See the section "Licensing Physical Verification Products" in the *Calibre Administrator's Guide* for details.

• Rule File Requirements

o An SVRF rule file suitable for Calibre nmDRC is required.

 A Layer statement specifying a name for an original layer is required to use the "Visible layer checks," "Visible layer only checks," and "Recently edited layer checks" selections in the Edit Recipe dialog box. See "Check Recipes in Calibre RealTime Custom" on page 105 for more information.

See "Differences Between Calibre nmDRC and Calibre RealTime Custom" on page 18 for information on unsupported SVRF statements.

Calibre RealTime Custom License Options

Calibre RealTime Custom requires certain product licenses for DRC runs and highlight operations in the toolbar.

Either the Calibre RealTime Custom or Calibre RealTime Custom Lite product license is acquired when you start a Calibre RealTime Custom run. The Calibre RealTime Custom Lite license can be used for the Calibre DESIGNrev, Synopsys Laker, Cadence Virtuoso, Siemens Tanner L-Edit, and Synopsys Custom Compiler integrations.

Both licenses pertain to the default way of running DRC (server mode) and highlighting in Calibre RealTime Custom. They both support the same set of SVRF commands, including the DRC SVRF commands supported by flat Calibre and basic DFM SVRF commands such as DFM COPY and DFM PROPERTY. However, there are some differences between the two licenses:

- The Calibre RealTime Custom Lite license requires an RVE license to operate, while the Calibre RealTime Custom license does not. The RVE license is not checked out, but it is required to be present.
- For server mode runs, the Calibre RealTime Custom Lite license has a hard limit of one million shapes, as opposed to three million for the standard license.
- Calibre Multi-Patterning and Calibre Pattern Matching checks are not authorized with the Calibre RealTime Custom Lite license, but are with the Calibre RealTime Custom license.

To configure which license to use, use the environment variable MGC_REALTIME_USE_LICENSE. When MGC_REALTIME_USE_LICENSE = lite, the Calibre RealTime Custom Lite license is used. When it is not set or is set to default, the standard Calibre RealTime Custom license is used.

For more information on these licenses, see Calibre RealTime Custom and Calibre RealTime Custom Lite in the Calibre Administrator's Guide.

Differences Between Calibre nmDRC and Calibre RealTime Custom

The Calibre RealTime Custom interface integrates flat Calibre nmDRC with supported layout design tools so that Calibre nmDRC can be run directly from the design tool. Calibre and Calibre RealTime Custom differ in run mode support, layout processing, SVRF command support, and result output.

Calibre RealTime Custom is designed to work best on a limited region of the layout and provide immediate results. Calibre RealTime Custom is not intended for sign-off or tapeout DRC verification.

Calibre RealTime Custom differs from a command-line invocation of Calibre nmDRC as follows:

Run Mode and Execution

- Calibre RealTime Custom runs in flat mode.
- Check selection is controlled by check recipes in Calibre RealTime Custom.

DRC [Un]Select Check statements are not obeyed except when using the "Checks selected in the rules file" check recipe; see "Check Selection Recipes in Calibre RealTime Custom" on page 105.

Layout Processing

• Property information included with a layout is not exported to Calibre RealTime Custom.

As a result, operations that make use of user-defined property attachments in the layout do not perform as expected. For example, DFM Property operating on properties in the rule file works, but DFM Property operating on user-defined property attachments in the layout does not work.

SVRF Support

Details for SVRF command support are discussed in "Support for SVRF Statements in Calibre RealTime Custom" on page 339.

DRC Results

- Auxiliary or side RDBs are not generated. For example, the Density operation can be used to generate results output but not to generate a side RDB (the RDB keyword).
- Density and connectivity checks run on a small region of the design may give different results than a similar run in batch Calibre or a run on the complete design.

Density and connectivity checks are intended to run on the complete design, while Calibre RealTime Custom is intended to run on a limited region of the design. See "Density Checks in Calibre RealTime Custom" on page 29 for more information.

If Density checks are run, then the window size returned by Calibre RealTime Custom is bigger than the window size returned by a Calibre batch run. This difference does not affect the accuracy of the results.

Caution A tapeout or sign-off Calibre nmDRC run should always be done using batch Calibre to ensure that all results are reported.

Terms and Definitions in Calibre RealTime Custom

Certain terms and concepts, such as check selection recipe, halo, run configuration, and run context, are used when discussing Calibre RealTime Custom operation.

Run Configuration

A run configuration is the settings in the Calibre RealTime Options dialog box that control a run, such as the rule file, layer map file, and check recipe. Multiple run configurations can be defined. The run configuration can be selected in the toolbar, making it easy to switch between configurations.

Run configurations are supported with Cadence Virtuoso, Synopsys Custom Compiler, and Synopsys Laker³; they are not supported with Calibre DESIGNrev or Pyxis Layout.

Multiple run configurations can be used in the following modes (the configuration run mode):

- **Single** Run only the currently selected configuration when a run is started.
- **Serial** Run each configuration selected in the "Configuration Run Control" dialog box in order.

See one of the following topics:

"Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)" on page 212.

"Specifying and Using Multiple Run Configurations in Calibre RealTime (Laker)" on page 154

"Specifying and Using Multiple Run Configurations in Calibre RealTime with Synopsys Custom Compiler" on page 188

Check Selection Recipe

A check recipe is a set of rules for selecting the checks to execute for a Calibre nmDRC run. See "Check Selection Recipes in Calibre RealTime Custom" on page 105 for more information.

Session Configuration File

Calibre RealTime Custom run configurations, option settings, and check recipe definitions are saved in the session configuration file. The configuration file is saved automatically and named .realtime.cfg. The configuration file is saved in your home directory by default; however, you may specify a different location for the configuration file.

See "Session Configuration File in Calibre RealTime Custom" on page 42.

Halo

Halos are used to ensure that new errors caused by changes in the geometry get reported, and false errors caused by edge effects are not reported. A halo expands the area around a region selected for a DRC run so that surrounding geometries are included. Halos are used for all Calibre RealTime Custom runs.

The optimum halo size depends on the technology node and the typical constraints being checked in the run. A halo size of 1 user unit (the default) is usually appropriate for 45 nm technology; a smaller halo size is probably needed for smaller technology nodes. See the section "Halos in Calibre RealTime Custom" on page 24 for more information.

DRC Hierarchical Levels Mode

You can specify the hierarchy levels in the design that are checked during the run. This option is only available in the integration to Cadence Virtuoso.

- All Check all hierarchy levels in the design.
- **Displayed** Check only the hierarchy levels that are displayed in the design tool.

Checking only the displayed hierarchy levels can shorten the run time and is useful for targeted debug at specific levels in the hierarchy. Depending on the selected rule checks, some spurious rule check results may occur in the "Displayed" mode due to the fact that not all levels of the design are checked.

Run Context

The run context refers to the window for the Calibre nmDRC run, including information such as the library, cell, and view, depending on the design tool. You can click the button in the toolbar to zoom to the window for the most recent Calibre RealTime Custom run.

Note

For 2015.1 and earlier releases, the toolbar included a dropdown list of the ten most recent run contexts. Beginning with the 2015.2 release the run context is no longer displayed by default. Define the environment variable

MGC_CALIBRE_REALTIME_SHOW_CONTEXT_IN_TOOLBAR to restore the run context display.

DRC Run Modes

Calibre RealTime Custom runs flat Calibre nmDRC and executes the rule checks selected in the check recipe for the current run configuration. These are the run modes:

- **Run DRC in Window** Calibre RealTime Custom runs Calibre nmDRC on the area visible in the design window extended by the halo region. All shapes which intersect the resulting region are included in the run. Available in all integrations.
- Run DRC in Cell— Calibre RealTime Custom runs Calibre nmDRC on the current cell. Available in the Calibre DESIGNrev and Cadence Virtuoso integrations.
- Run DRC in Area—Calibre RealTime Custom runs Calibre nmDRC on a selected area. After starting the run, you are prompted to select an area by drawing a rectangle in the window. Available in the Calibre DESIGNrev and Cadence Virtuoso integrations
- Run DRC in Past Area or Region— Calibre RealTime Custom runs Calibre nmDRC on a past selected area. Available in the Calibre DESIGNrev and Cadence Virtuoso integrations; the most recent area is saved as Area 1 in Cadence Virtuoso or Past Region 1 in Calibre DESIGNrev.

The maximum number of polygons that Calibre RealTime Custom processes can be set in the Calibre RealTime Options dialog box; the default is set at 50,000, and the maximum is 3 million.

The Calibre RealTime Custom integration to Cadence Virtuoso can launch a batch Calibre run if the number of polygons being processed is greater than a specified limit. See "Starting a Batch Calibre Run from Calibre RealTime" on page 218.

Documentation Syntax Conventions

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

Table 1-1. Syntax Conventions

Convention	Description
Bold	Bold fonts indicate a required item.

Table 1-1. Syntax Conventions (cont.)

Convention	Description	
Italic	Italic fonts indicate a user-supplied argument.	
Monospace	Monospace fonts indicate a shell command, line of code, or URL. A bold monospace font identifies text you enter.	
<u>Underline</u>	Underlining indicates either the default argument or the default value of an argument.	
UPPercase	For certain case-insensitive commands, uppercase indicates the minimum keyword characters. In most cases, you may omit the lowercase letters and abbreviate the keyword.	
[]	Brackets enclose optional arguments. Do not include the brackets when entering the command unless they are quoted.	
{ }	Braces enclose arguments to show grouping. Do not include the braces when entering the command unless they are quoted.	
٠,	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:	Example:	
DEVice {element_name ['('model_name')']}		
device_la	device_layer {pin_layer ['('pin_name')']}	
['<'auxil	['<'auxiliary_layer'>']	
['('swap_	['('swap_list')']	
[BY NET BY SHAPE]		

Chapter 2 Calibre RealTime Custom Basics

All integrations of Calibre RealTime Custom share some basic methodology, such as halo usage and configuration file usage. Some integrations support API control for setting options and controlling environment variables.

See "Check Selection Recipes in Calibre RealTime Custom" on page 105 for information on check selections recipes.

Halos in Calibre RealTime Custom	24
Density Checks in Calibre RealTime Custom	29
Calibre RealTime Options Dialog Box	31
Database Read Setup in Calibre RealTime Custom	39
Session Configuration File in Calibre RealTime Custom	42
Option and Environment Variable Control in Calibre RealTime Custom	52
License Timeout for Calibre RealTime Custom	7 8
Debugging Double Patterning Results with Calibre RealTime Custom	7 9
Viewing Results in the Calibre RealTime Results Window	83
Using Calibre Interactive Settings in a Calibre RealTime Custom Run	87
DRC Rule Check Comments for Calibre RealTime Custom	9(

Halos in Calibre RealTime Custom

Halos ensure that new errors caused by changes in the geometry are reported, and false errors caused by edge effects are not reported. A halo expands the area around a region selected for a DRC run so that surrounding geometries are included. Halos are used for all Calibre RealTime Custom runs.

Halo Usage in Calibre RealTime	24
Inconsistent Results from Run DRC on Edit Mode with Small Halo Size	20

Halo Usage in Calibre RealTime

Halos ensure that new errors caused by changes in the geometry are reported, and false errors caused by edge effects are not reported. Halos are used for all Calibre RealTime Custom runs. Each time Calibre RealTime Custom runs Calibre nmDRC, a halo expands the area around the region selected for the run so that surrounding geometries are included.

In order to remove false errors due to edge effects, a "trim region" is defined and results that are fully outside the trim region are discarded.

The different halo usages and trim regions are summarized here:

- **Standard Halo Usage** The trim region is defined as the region selected for the DRC run *plus half the specified halo size*. Results that are fully outside the trim region are discarded.
- Reduced Trim Region The trim region is equal to the region for the DRC run when
 the environment variable MGC_REALTIME_USE_WINDOW_AS_TRIM_REGION is
 set.
- **Keep Enclosing Results** Beginning with the 2015.2 release, results that fully enclose the trim region are discarded by default. You can include such results by defining the environment variable
 - MGC_REALTIME_INCLUDE_RESULTS_ENCOMPASSING_WINDOW; this returns the behavior to that found in 2015.1 and earlier releases.

Standard Halo Usage

Standard halo usage is shown in Figure 2-1. Standard halo usage is in effect when no environment variables related to halo and trim region are set. The trim region is set to the region selected for the DRC run plus half the specified halo size, and results that are fully outside the trim region are discarded.

DRC in process on Original design and Edited design and final edited design result result Halo 2 Halo size size Metal layer Area selected "Trim region" — Only for DRC results inside or Result intersecting the selected DRC run on selected Fixed Result DRC area plus 1/2 the area plus halo halo size are kept The original design The final results keep the The edited design fixes the has one metal new spacing error. original error, but introduces a spacing error. new spacing error. Results completely outside the "trim regions" are discarded.

Figure 2-1. Calibre RealTime Custom Operation With Halo

Without a halo, new errors introduced by a design change may not be detected; however, a halo that is too small may cause false errors to be reported.

A halo size of 1 user unit (the default) is usually appropriate for 45 nm technology; a smaller halo size is typically needed for smaller technology nodes. The halo size is set in the Calibre RealTime Options dialog box. See the section "Inconsistent Results from Run DRC on Edit Mode with Small Halo Size" on page 26 for information on the effect of a small halo size in relation to rule check spacing constraints.

Reduced Trim Region

Set the environment variable MGC_REALTIME_USE_WINDOW_AS_TRIM_REGION to define the trim region as equal to the region for the DRC run.

When MGC_REALTIME_USE_WINDOW_AS_TRIM_REGION is defined, only DRC results that are inside or overlap the window for the DRC run are kept. Referring to the last diagram in Figure 2-1, the new result is discarded if the trim region is set equal to the region of the DRC run.

. Note



The mode with MGC_REALTIME_USE_WINDOW_AS_TRIM_REGION defined is similar to what is used by Calibre Interactive Area DRC runs.

Keep Enclosing Results

Beginning with the 2015.2 release, results that fully enclose the trim region are discarded by default. You can include such results by defining the environment variable MGC_REALTIME_INCLUDE_RESULTS_ENCOMPASSING_WINDOW; this returns the behavior to that found in 2015.1 and earlier releases.

For example, if the new result is a polygon that surrounds the trim region, it is discarded unless you define MGC_REALTIME_INCLUDE_RESULTS_ENCOMPASSING_WINDOW.

Related Topics

Halos in Calibre RealTime Custom

Inconsistent Results from Run DRC on Edit Mode with Small Halo Size

You may see inconsistent error reporting when Calibre RealTime is run in Run DRC on Edit mode with a small halo size. In particular, this can occur if the halo size is smaller than a spacing constraint being checked. The Run DRC on Edit mode is not available in all integrations.

The effect of a small halo size is illustrated in the following figure, where a polygon is moved closer to a second polygon, then away again.

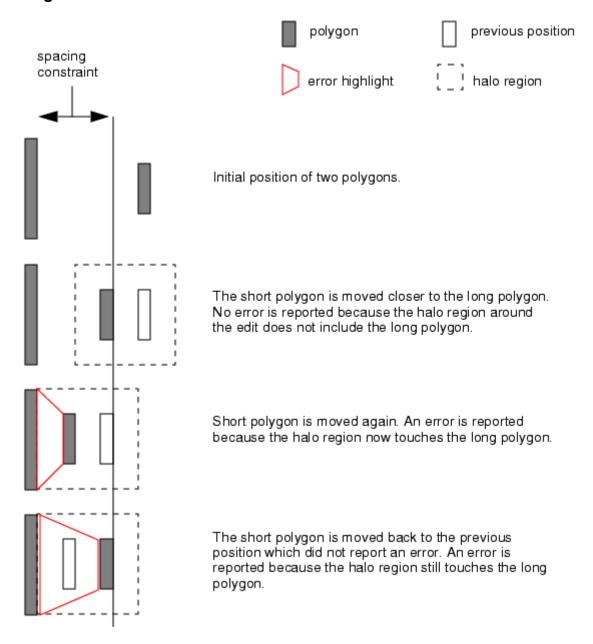
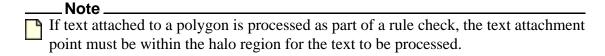


Figure 2-2. Results from Run DRC on Edit Mode with Small Halo Size

Do one or both of the following to make sure that you see all errors:

- Increase the halo size to be equal or greater than the spacing constraint being checked. The halo size is set in the Calibre RealTime Options dialog box (button).
- Run a window check after completing a series of design edits (v or button).



Related Topics

Halos in Calibre RealTime Custom

Density Checks in Calibre RealTime Custom

As with other rule checks, Density checks in Calibre RealTime Custom analyze the visible region in the layout window plus a halo region or the region of the design edit plus a halo region, depending on the selection Run DRC in Window or Run DRC on Edit (if available). Although the Density results are accurate for the specific run context, the results may differ from those of a Calibre batch run or a Calibre RealTime Custom run with a different run context.

Spurious Boundary Results From Density Checks in Calibre RealTime

Density checks may give spurious results in Calibre RealTime if they are run on a small region of the design.

The following figure illustrates how spurious results can occur from a Density check.

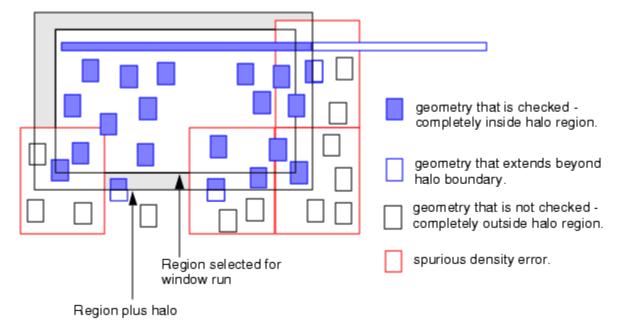


Figure 2-3. Spurious Density Results in Calibre RealTime Window Run

Calibre RealTime handles geometry that extends outside the halo region in the following manner, depending on the design tool:

Synopsys Laker, Cadence Virtuoso, and Calibre DESIGNrev — If a polygon extends
outside the halo region, only the portion of the geometry within the halo region is
analyzed.

• Pyxis Layout — If a polygon extends outside the halo region, the entire polygon is analyzed.

In a typical scenario, a Calibre run on the complete design may return zero density results. However, a Calibre RealTime run on a small region of the design may return spurious density results at the boundaries of the region, and these results are kept if they intersect the "trim region" (see the figure in "Halo Usage in Calibre RealTime" on page 24). Results completely outside the trim region are discarded.

The toolbar button zooms to the window region for the most recent Calibre RealTime run. If the window returns to the complete design or any context that includes the region of these spurious results, the spurious results are counted in the total results reported for the context.



Thus, the number of results for the design may appear to increase when density or connectivity checks are included in a run done on a small region of the design.

Calibre RealTime Options Dialog Box

To access: Choose Calibre > RealTime DRC > Options or click the was button in the toolbar.

The Calibre RealTime Options dialog box specifies the Calibre rule file, the check recipe, and other options. The settings in the Calibre RealTime Options dialog box are saved in the Calibre RealTime Configuration File.

Description

. Note

The options for the Calibre RealTime integrations to Calibre DESIGNrev, Cadence Virtuoso, Synopsys Custom Compiler, and Synopsys Laker are documented here. The following image is for the Cadence Virtuoso integration.

If an option is not supported in all integrations, it is noted in the table of options. See "Calibre RealTime Custom with Pyxis Layout" on page 125 for information about the integration to Pyxis Layout.

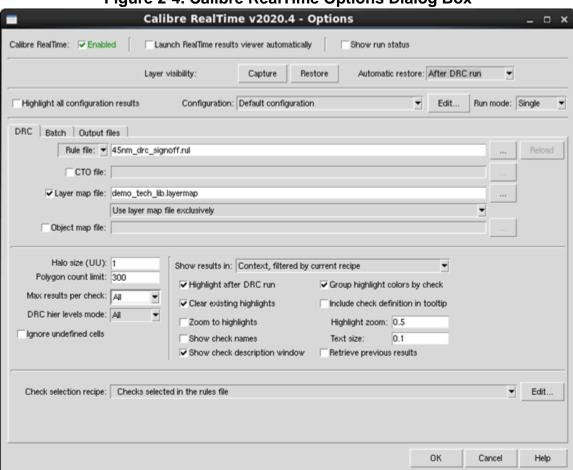


Figure 2-4. Calibre RealTime Options Dialog Box

Table 2-1. Tabs in the Calibre RealTime Options Dialog Box

Tab	Description
DRC	Specify options for the Calibre DRC run. See the table in the next section.
Batch (Only available in Cadence Virtuoso)	Specify options to launch a batch Calibre nmDRC run if the design window contains more than a specified number of polygons. See "Starting a Batch Calibre Run from Calibre RealTime" on page 218.
Output files	Specify to save a summary file and/or to save the DRC results to an ASCII results database (RDB). If a specified directory does not exist, the tool prompts you to confirm that the directory should be created.

Fields

Table 2-2. Calibre RealTime Options Dialog Box Contents

Control	Description
Calibre RealTime	Toggles display of the Calibre RealTime toolbar.
(Enabled/Enable)	The Calibre RealTime toolbar is automatically displayed if a DRC run is launched using Calibre > RealTime DRC > Run DRC in Window.
	Also see the environment variable MGC_REALTIME_DISABLE in "Calibre RealTime Environment Variables" on page 333.
Run DRC on edit ^{1, 5}	When enabled, start a flat Calibre nmDRC automatically after any design edit.
Launch RealTime results viewer automatically ²	Specifies whether to open the Calibre RealTime Results Window automatically after a run. See "Viewing Results in the Calibre RealTime Results Window" on page 83.
Highlight all configuration results	Specifies whether to automatically highlight DRC results from all configurations after a DRC run. When this option is enabled:
(Only available in Cadence Virtuoso, Synopsys Laker ³ , and Synopsys Custom Compiler)	 Highlights are preserved when switching configurations. When highlighting, the "Clear existing highlights" option is ignored and existing highlights are not cleared. Results from all configurations are highlighted in both single and serial runs, and the "Highlight after DRC" option is ignored. Note: When this option is enabled, you can only crosshighlight results from the layout editor to the Results window if they are from the current configuration. You can view the check description of any result regardless of its corresponding configuration.

Table 2-2. Calibre RealTime Options Dialog Box Contents (cont.)

Control	Description
Configuration ²	Specifies the configuration name. To add or delete a configuration, click Edit to open the Configuration Run Control dialog box.
	A configuration is the set of options and filenames below the configuration name.
Run Mode ²	 Single — Run with the currently selected configuration. Serial — Run each selected configuration in series, in the order specified in the Configuration Run Control dialog box.
Rule file/Runset dropdown list	Selects whether to load a rule file or a Calibre Interactive runset.
Rule file	Specifies the Calibre rule file used for the run.
	See "Environment variable usage" in the "Usage Notes" section for information on using environment variables.
Runset	Specifies a Calibre Interactive runset to use for the run. See "Using Calibre Interactive Settings in a Calibre RealTime Custom Run" on page 87 for details.
	See the "Usage Notes" for information on using environment variables.
CTO file ^{3, 5}	Specifies a CTO file for controlling the highlight color and layer visibility per rule check. See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90.
	See the "Usage Notes" for information on using environment variables.
Layer map file	Specifies a custom layer map file.
	Dropdown list options (not available in Calibre DESIGNrev or Synopsys Custom Compiler):
	• Merge layer map with existing mapping — (default) Merge layer map data from the specified file with the layer map derived from the technology file. Layer map entries in the layer map file override the corresponding layer map entry based on the technology file; all other layer map entries based on the technology file remain valid.
	• Use layer map file exclusively— Use only layer map data from the specified file. Layer map data derived from the technology file is <i>not</i> used.
	See "Layer Maps in Calibre RealTime" on page 39.
	See "Environment variable usage" in the "Usage Notes" section for information on using environment variables.

Table 2-2. Calibre RealTime Options Dialog Box Contents (cont.)

Control	Description
Object map file ²	Specifies an object map file. See "Object Map File Format in Calibre RealTime" on page 40.
	See the "Usage Notes" for information on using environment variables.
(Browse)	Open a file browser to select the rule file, CTO file, or layer map file.
Reload	Reload the indicated file.
	You must click Reload for changes in the rule file, runset, or CTO file to take effect; changes in the layer map file or object map file cause an automatic reload prior to the DRC run.
Halo size (UU)	The halo size in user units. See "Halos in Calibre RealTime Custom" on page 24 and "Halo" in "Terms and Definitions in Calibre RealTime Custom" on page 19 for more information on halos.
Polygon count limit	Sets the maximum number of polygons that are passed to Calibre. The limit should be set low enough that the Calibre server returns control to the design tool in a short amount of time.
	The maximum number of geometries Calibre RealTime will process is 3,000,000.
Max results per check	Specifies the maximum number of results returned per rule check; this setting overrides the setting in the rule file.
DRC hier levels mode ^{4, 5}	Specifies the hierarchy levels in the design that are checked during the run.
	• All — Check all hierarchy levels in the design.
	• Displayed — Check only the hierarchy levels that are displayed in the layout tool.
	Depending on the selected rule checks, some rule check results may occur in the "Displayed" mode due to the fact that not all levels of the design are checked.
Ignore undefined cells ^{2, 4, 5}	When checked, specifies for the Calibre nmDRC run to issue a warning and continue if undefined cells are found in the design.

Table 2-2. Calibre RealTime Options Dialog Box Contents (cont.)

Control	Description
Show results in	This setting controls which results are reported in the integrated toolbar after the run.
	• Whole cell, all checks — Report all results for the cell in the integrated toolbar after each DRC run. Reported results include those from the current run, previous results in the cell that are outside of the current run context, and possibly previous results within the current run context from rules checks not selected by the current recipe.
	• Whole cell, filtered by current recipe — Report results for the whole cell and from the current recipe only in the integrated toolbar after each DRC run. Reported results include those from the current run plus previous results in the cell that are outside of the current run context.
	• Context, filtered by current recipe — (default) Report only results within the current run context and from the current check recipe in the integrated toolbar after each DRC run.
	For each case, all results in the current window and belonging to the current check selection recipe are deleted before the run.
	The settings "Whole cell, filtered by current recipe" and "Context, filtered by current recipe" should be used with caution, since they filter out results from previous runs. If these settings are used, it is recommended that a later run be performed with "Whole cell, all checks" selected to make sure that all results are reported.
Highlight after DRC run	Highlight results in the viewer after the DRC run.
Group highlight colors by	Assign the highlight color for results according to the rule check.
check	The assigned highlight color for a rule check is not fixed—rather, the highlight colors are assigned to checks in a repeated cycle, starting with the color red. See "Behavior of "Group highlight colors by check"" in the "Usage Notes" for details.
	You can also set highlight colors with rule check comments, see "DRC Rule Check Comments for Calibre RealTime Custom" on page 90.
Clear existing highlights	Clear existing highlights before each new highlight action.
	Highlights are cleared before each run, and when using the
	⇔ buttons to step though highlights.
	Also see "Behavior of "Group highlight colors by check"" in the Usage Notes section.

Table 2-2. Calibre RealTime Options Dialog Box Contents (cont.)

Control	Description
Include check definition in tool-tip	When enabled, the complete rule check definition is included in the result tool tip; otherwise, only the check text comments are shown.
	You must run Calibre RealTime after changing this setting for the change to take effect.
Zoom to highlights	Specifies whether to zoom to the extent of the results when highlighting.
	When automatically highlighting after a DRC run, RealTime does not zoom to highlights to avoid changing the currently selected viewpoint. ^{2, 5}
Highlight zoom	Specifies the magnification used when zooming to highlight a result.
Show check names ^{2, 5}	Specifies to display the check name at the center of each result.
Text size ^{2, 5}	Specify the text size when "Show check names" is enabled.
Show check description window ^{2, 4, 5}	Specifies to display the check description window when highlighting.
Retrieve previous results ^{2, 4, 5}	Specifies whether to retrieve previous results for the cell of interest and display them in the toolbar after changing cells with an open, descend, or ascend action. Retrieval of previous results takes additional time.
	You can disable the option by defining the environment variable MGC_CALIBRE_REALTIME_NO_DRC_RESULT_RETRIEV AL.
Check selection recipe	The check recipe used for the Calibre run. See "Check Selection Recipes in Calibre RealTime Custom" on page 105.
Edit	Open the Calibre RealTime Recipe Editor Dialog Box.

¹ Not available in Cadence Virtuoso

² Not available in Calibre DESIGNrev

³ The CTO file is not supported in Synopsys Custom Compiler or Synopsys Laker³. You can set highlight colors using DRC rule check comments in the rule file when using Synopsys Laker³.

⁴ Not available in Synopsys Laker³

⁵ Not available in Synopsys Custom Compiler

Usage Notes

• Environment Variable Usage

- You can use environment variables in pathnames for files and directories. The environment variable is not expanded when viewed in the dialog box or saved to the configuration file. For example, you can enter \$CAL_RULES/drc.rul for the rule file, and \$CAL_RULES/drc.rul is saved in the configuration file.
- You can enter an environment variable for a directory path in a file entry field and click the ... button to open the file browser to that directory. This makes it easier to browse for a file if it is not in the working directory and the directory path is long. When you select a file, the absolute path is entered in the text entry file and saved in the configuration file. This capability is available for the rule file, CTO file, layer map file, and object map file, but not for the runset file.
- There are system defined environment for the input files; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77. The Calibre RealTime Custom environment variables specific for input files take precedence over user-defined environment variables.

• Behavior of "Group highlight colors by check"

When "Group highlight colors by check" is enabled, colors are assigned to results according to the rule check. However, the association between the highlight color and the rule check is not fixed; it depends on the history of your Calibre RealTime session. Highlight colors are assigned to checks in a repeated cycle, starting with the color red. The color assignment starts at the beginning (with red) each time highlights are cleared. Highlights are cleared in the following cases:

- O Automatically when "Clear Existing Highlights" in the options menu is enabled; highlights are cleared at the start of a run, or at each new highlight when using the → → buttons.
- o By clicking the **Clear Highlights** button ().
- o Automatically when the button in the integrated toolbar is clicked. The button deletes a rule check from the current check recipe and clears all highlights.

The following behavior is observed when "Group highlight colors by check" is enabled:

o Highlight color assignments from a DRC run:

Clear Existing Highlights ON — Highlights are cleared and highlight color assignment starts at red at the beginning of each Calibre RealTime run.

Clear Existing Highlights OFF — Highlights from previous runs are kept. New results from a check with an existing highlighted result are highlighted in the same color. New results from a check that does not have an existing highlighted result are highlighted in the next color in the highlight color cycle.

Clicking or — Highlights are cleared and highlight color assignment starts at red for the next run, regardless of the setting of "Clear Existing Highlights."

"Clear Existing Highlights" ON — All highlights are cleared each time one of the

→ ⇒ buttons is clicked; each result is highlighted in red.

"Clear Existing Highlights" OFF — Each result is highlighted in the color used originally.

Clear Highlights button clicked when "Clear Existing Highlights" OFF — All highlights are cleared. The next result highlighted with one of the buttons is highlighted in red; the highlight color for further results is assigned according to the rule check. All highlights are kept until is clicked again.

Database Read Setup in Calibre RealTime Custom

Calibre RealTime Custom can read object maps and layer maps. These files specify the mapping of database layers and objects to the GDS layer number and datatype pairs expected by the Calibre rule file.

Layer Maps in Calibre RealTime	39
Object Map File Format in Calibre RealTime	40

Layer Maps in Calibre RealTime

Calibre RealTime requires that layers in the design database are mapped to GDS layer number and datatype pairs for layout export; the mapping should correspond to what is expected by the Calibre rule file and any Layer Map statements within the rule file.

Related Topics

Setting the Layer Map File for Calibre RealTime with Pyxis Layout

Specifying the Layer Map for Calibre RealTime with Synopsys Laker

Database Read Setup for Calibre RealTime with Cadence Virtuoso

Database Read Options in Calibre RealTime with Calibre DESIGNrev

Object Map File Format in Calibre RealTime

Object Map File Format in Calibre RealTime

Used in: Calibre RealTime for Synopsys Laker, Synopsys Custom Compiler, and Cadence Virtuoso

The object map file specifies how OpenAccess database objects are mapped to GDS layer number and datatype pairs expected by the Calibre rule file. The object map file is specified in the Calibre RealTime Options dialog box.

If you use both an object map file and a layer map file, the same GDS layer number and datatype pairs should not be used in both files.

Format

The object map file has the following format:

• One entry per line with the following format:

ObjectType SubType [oaLayerName] GDS_layer_number GDS_datatype

• Comment lines start with a # character.

Parameters

ObjectType SubType [oaLayerName]

The object type and subtype to be mapped, and an optional OpenAccess layer name. The object type and subtype can have the values given in the following table:

Table 2-3. ObjectType and SubType Values in Calibre RealTime

ObjectType	SubType		oaLayerName used?
Boundary	PR	Area	No
	Cluster	Snap	
LayerBlockage	Routing	ViaRouting	Yes, optional
	Slot	Wiring	
	Pin	Fill	
	Feedthru	Screen	
AreaBlockage	placement		No

Note

If duplicate mappings exists for the same ObjectType and SubType, then both mappings are performed. This should be avoided in most cases. This behavior may change in a future release.

Usage of oaLayerName:

The parameter *oaLayerName* is optional and only used for the LayerBlockage object type. The following process is used when mapping the LayerBlockage object type:

- 1. Map objects for which the *oaLayerName* parameter is specified.
- 2. Map remaining objects; objects that have already been mapped in Step 1 are not mapped again. See the "Examples" section.
- *GDS_layer_number GDS_datatype*

The GDS layer number and datatype pair to map the object to. The layer number and datatype pairs should correspond to what is expected in the Calibre rule file. See the Layer and Layer Map statements in the Standard Verification Rule Format (SVRF) Manual.

Examples

# ObjectType	SubType	[oaLayerName]	GDS_layer_number	GDS_datatype
LayerBlockage	Routing	MO1	195	0
LayerBlockage	Routing		109	0
Boundary	PR		2518	0
Boundary	Area		2518	1
Boundary	Cluster		2518	2
Boundary	Snap		2518	3

All PR Boundary objects are mapped to GDS layer 2518 datatype 0; other Boundary subtypes are mapped as indicated.

Routing layer blockage objects for layer M01 are mapped to GDS layer 195 datatype 0; routing layer blockages for all other layers are mapped to GDS layer 109 datatype 0.

Related Topics

Specifying the Object Map for Calibre RealTime with Cadence Virtuoso Specifying the Object Map for Calibre RealTime with Synopsys Laker Layer Maps in Calibre RealTime

Session Configuration File in Calibre RealTime Custom

Calibre RealTime Custom run configurations, option settings, and check recipe definitions are saved in the session configuration file. The configuration file is saved automatically and named .realtime.cfg. The configuration file is saved in your home directory by default; however, you may specify a different location for the configuration file.

You can delete .realtime.cfg to return to the default session configuration; however, this will also delete the custom check recipe definitions contained in the configuration file and any Run Configurations.

Calibre RealTime Custom supports multilevel session configuration files, which enable you to read configuration files from multiple directories and define the order of precedence. Multilevel configuration files can be used by an administrator group to manage options settings for project teams.

Caution

Editing the session configuration file is not recommended unless setting up multilevel configuration files.

Location of the Calibre RealTime Session Configuration File	42
Multilevel Session Configuration File Support in Calibre RealTime Custom	4 4

Location of the Calibre RealTime Session Configuration File

The Calibre RealTime session configuration file is saved in your home directory by default. It can also be saved in a user-defined directory. The configuration file is named .realtime.cfg.

The location of the configuration file is determined as follows:

- **Home directory** (default) Your home directory is the default location of the configuration file. No action is needed if you are not using multilevel configuration file support.
 - If you are using multilevel configuration file support, \$HOME must be listed as one of the directories in MGC_REALTIME_CONFIG_DIRS; see "Environment Variable for Multilevel Session Configuration Files" on page 44.
- **User-defined directory** Include WRITE=*directory* in the definition of MGC_REALTIME_CONFIG_DIRS to instruct Calibre RealTime to write the updated configuration file to *directory*.

For example, to save the configuration file to the /proj/design1/RTcfg directory:

setenv MGC REALTIME CONFIG DIRS WRITE=/proj/design1/RTcfg

In the preceding example the configuration file is both read from and saved to the directory specified by WRITE=. (Multilevel configuration files are not specified in the preceding example, however, WRITE= is supported in multilevel configuration file support.)

Related Topics

Multilevel Session Configuration File Support in Calibre RealTime Custom

Session Configuration File in Calibre RealTime Custom

Environment Variable for Multilevel Session Configuration Files

Creating Multilevel Configuration Files in Calibre RealTime Custom

Multilevel Session Configuration File Support in Calibre RealTime Custom

Calibre RealTime Custom supports reading the session configuration file from a specified directory or multiple directories. If configuration files are located in multiple directories, the order of precedence is set with the environment variable MGC_REALTIME_CONFIG_DIRS. This feature allows system and project administrators to define settings for a group or project while still allowing settings to be adjusted by individual users.

You can also save the session configuration file in a directory other than your home directory.

The following table indicates which design tools support multilevel configuration files with Calibre RealTime Custom.

Table 2-4. Multilevel Configuration File Support in Calibre RealTime Custom

Design Tool	Support
Calibre DESIGNrev	Yes
Cadence Virtuoso	Yes
Synopsys Laker ³	Yes
Synopsys Custom Compiler	Yes
Siemens Pyxis Layout	Yes, for Pyxis Layout version 10.1 and later. The WRITE keyword is not available.

Environment Variable for Multilevel Session Configuration Files	4 4
Creating Multilevel Configuration Files in Calibre RealTime Custom	48
Migrating Multilevel Session Configuration Files from an Earlier Release	50

Environment Variable for Multilevel Session Configuration Files

The MGC_REALTIME_CONFIG_DIRS environment variable specifies a list of one or more directories separated by a colon (:), where each directory contains a .realtime.cfg file. The order of precedence is from left to right, as with the Unix \$PATH variable.

MGC_REALTIME_CONFIG_DIRS also specifies the location of the writable session configuration file for saving updated configuration settings.

The following two examples explain the use of MGC_REALTIME_CONFIG_DIRS:

Example 1 — Writable configuration file in \$HOME:

setenv MGC REALTIME CONFIG DIRS dir1:dir2:\$HOME

Example 2 — Writable configuration file in user-defined directory (not available in Pyxis Layout):

```
setenv MGC REALTIME CONFIG DIRS dir1:dir2:WRITE=cfg dir
```

Each of the directories dir1, dir2, \$HOME, and cfg_dir contains a .realtime.cfg file.

Precedence

The order of precedence is from left to right. In both examples the settings from *dir1* take precedence over those from *dir2* if there is a conflict.

Location of Writable Configuration File

The writable configuration file is specified with either \$HOME (Example 1) or WRITE=cfg_dir (Example 2). Changes to the Calibre RealTime configuration settings are saved to this directory, subject to the rules described in "Saving Options". If both \$HOME and the WRITE keyword are specified, the WRITE keyword takes precedence. If MGC_REALTIME_CONFIG_DIRS is defined and neither \$HOME or the WRITE keyword is specified, then changes to the configuration are not saved.

The WRITE keyword is not available with Pyxis Layout.

Saving Options

In general, and referring to Example 1, the user may change any option during the Calibre RealTime session, but only options that are not defined in *dir1* or *dir2* are saved to the configuration file in *\$HOME*.

The following statements describe in detail the rules for saving changes to Calibre RealTime settings:

- Individual options are saved only if they are not defined in a configuration file that has higher precedence than the writable configuration file.
 - If the writable configuration file includes an option that is overridden by a configuration file with higher precedence, then the option is removed from the writable configuration file the first time it is saved.
- Individual options that are not defined in any configuration file are saved to the writable configuration file.

Options not defined in any configuration file are added to the writable configuration file the first time it is saved. If \$HOME or WRITE=cfg_dir has higher precedence than other directories listed in MGC_REALTIME_CONFIG_DIRS, this may cause unexpected results.

See Figure 2-5 on page 47 for an illustration of these rules.

Check Recipe Handling

User-defined check recipes are handled differently depending on whether they are defined in the writable configuration file or in a read-only configuration file.

- Writable configuration file Check recipes defined in the writable configuration file may be edited, and these recipes appear in the USER RECIPES category in the "Check selection recipe" dropdown list.
- **Read-only configuration file** Check recipes defined in read-only configuration files appear in the BUILT-IN RECIPES section and may not be edited. To edit these recipes, create a copy with a unique name and then edit and save the recipe as a new user-defined recipe.

Referring to Example 1, check recipes defined in *dir1/.realtime.cfg* appear in the BUILT-IN RECIPES section and cannot be edited. Check recipes defined in *\$HOME/.realtime.cfg* appear in the USER RECIPES section and can be edited. New check recipes are saved to the configuration file in *\$HOME*.

Editing Configuration Files

It is necessary to edit the configuration files to make full use of multilevel configuration files; see "Creating Multilevel Configuration Files in Calibre RealTime Custom" on page 48 for instructions.

Multi-level Configuration Setup Behavior Unmanaged MGC_REALTIME_CONFIG_DIRS not defined Configuration Tool Setup: managed by saved configuration optA optA L individual users. optB optB optC optC defaults all settings \$HOME \$HOME Managed, MGC_REALTIME_CONFIG_DIRS with user \$PROJ:\$HOME preferences Configuration optB managed by optC project admins. Tool Setup: saved configuration optA optA optB \$PROJ Users can adjust optB(\$PROJ) settings during a optA optC \$HOME session, but only optD optD default unmanaged optD — not defined other defaults other defaults settings are \$HOME saved. Managed, MGC REALTIME_CONFIG_DIRS no user \$PROJ:\$SYS preferences Configuration optB controlled by optC project admins. Tool Setup: saved configuration optA optA \$PROJ Users can adjust optB optB(\$PROJ) settings during a optC not saved \$SYS \$HOME session, but defaults optD - not defined configuration settings are not \$HOME saved.

Figure 2-5. Multilevel Configuration Files in Calibre RealTime Custom

Creating Multilevel Configuration Files in Calibre RealTime Custom

You can create and define multilevel configuration files for Calibre RealTime Custom. This allows Calibre RealTime Custom to read settings from more than one configuration file.

Note_

These instructions assume that the writable configuration file is in your \$HOME directory. See "Environment Variable for Multilevel Session Configuration Files" on page 44 for information on using the WRITE keyword to specify a different location for the writable configuration file.

Prerequisites

• You are using Calibre RealTime Custom with a design tool that supports multilevel configuration files; see "Multilevel Session Configuration File Support in Calibre RealTime Custom" on page 44.

Procedure

1. Specify that the configuration file is written to \$HOME (the default). For example, using csh:

```
unsetenv MGC_REALTIME_CONFIG_DIRS
```

- 2. Create the configuration file using the following steps:
 - a. Rename the existing configuration file:

```
mv ~/.realtime.cfg ~/.realtime.cfg saveme
```

This removes the configuration file, which causes Calibre RealTime to open with default settings. If you skip this step, the tool opens with the settings in \$HOME!.realtime.cfg, and later steps overwrite the configuration file.

- b. Save a configuration file with your desired settings as follows:
 - i. Invoke your design tool with Calibre RealTime Custom enabled.
 - ii. Open the Calibre RealTime Options dialog box and set options.
 - iii. Create any system-configured check recipes. (Check recipes should always be created and edited using the Calibre RealTime Recipe Editor dialog box.)
 - iv. Delete user-defined check recipes that are not needed.
 - v. Exit Calibre RealTime.
- c. Open the configuration file at \$HOME/.realtime.cfg in a text editor. The option settings consist of lines similar to the following:

```
<halo size>1<halo size>
```

- d. Delete the complete line for option settings that meet the following criteria:
 - o The option is set by a higher precedence configuration file.
 - You are editing a project-level configuration file (one that will not be saved in \$HOME), and the option is allowed to be saved by the user.

If you want users to be able to save the active recipe, delete the line starting with <active_recipe>; this option is near the end of the configuration file.

- e. Save the configuration file to the desired directory with the name .realtime.cfg.
- 3. Repeat Step 2 for as many configuration files as needed.
- 4. When you are done creating configuration files, place the desired user-level configuration file in \$HOME if needed.
- 5. Define MGC_REALTIME_CONFIG_DIRS with the desired precedence for configuration files. For example:

```
setenv MGC REALTIME CONFIG DIRS $PROJ: $HOME
```

where \$PROJ defines the project-level location. See Figure 2-5 in "Environment Variable for Multilevel Session Configuration Files" on page 44.

Examples

The following session Configuration file is used with Synopsys Laker. It sets the rule file and layer map file and defines one user recipe.

```
<realtime>
  <options>
   <rule file>/user/45 design/45nm drc.rul</rule file>
   <layer map_file>/user/45_design/gds_layer.map</layer_map_file>
   <enable layer map file>1</enable layer map file>
   <user recipes>
      <recipe>
        <name>Selected no dens conn</name>
        <operations>
          <group unselect>all</group unselect>
          <group select>rule file/group select>
          <group unselect>connectivity/group unselect>
          <group unselect>density/group unselect>
        </operations>
      </recipe>
    </user recipes>
  </options>
</realtime>
```

To use this configuration file to set project-level defaults, it is placed in the directory |user|45 design, then multilevel configuration is defined as follows:

```
setenv MGC REALTIME CONFIG DIRS /user/45 design:$HOME
```

When a user starts Calibre RealTime, the rule file and layer map file are set. The recipe Selected_no_dens_conn is defined and listed with the built-in recipes; it cannot be edited. The user can change all settings during the Calibre RealTime session, but only settings not defined in the above configuration file are saved to \$HOME/.realtime.cfg on exit.

Related Topics

Multilevel Session Configuration File Support in Calibre RealTime Custom

Session Configuration File in Calibre RealTime Custom

Environment Variable for Multilevel Session Configuration Files

Location of the Calibre RealTime Session Configuration File

Migrating Multilevel Session Configuration Files from an Earlier Release

Some releases of Calibre RealTime include a syntax update for the session configuration file .realtime.cfg. If you use the environment variable MGC_REALTIME_CONFIG_DIRS to set up multiple Calibre RealTime session configuration files in different directories, the session configuration files in each directory must be updated to the new syntax.

The 2016.2 release includes a syntax update for the session configuration file. If you have multilevel session configuration files created in a release prior to 2016.2, follow these instructions to update the configuration files.

Procedure

1. Instruct Calibre RealTime to read and write the .*realtime.cfg* file in your \$HOME directory.

```
unsetenv MGC REALTIME CONFIG DIRS
```

2. Save your existing .realtime.cfg file:

```
cp $HOME/.realtime.cfg $HOME/.realtime.cfg.save
```

- 3. Do the following for each *dir* with a session configuration file:
 - a. Copy the .realtime.cfg in dir to \$HOME:

```
cp dir/.realtime.cfg $HOME
```

- b. Open your design tool with Calibre RealTime enabled.
- c. Exit your design tool.
- d. Move the updated .realtime.cfg file back to dir:

```
mv $HOME/.realtime.cfg dir
```

Repeat steps a-d for the next directory with a session configuration file.

4. Copy your saved \$HOME/.realtime.cfg from Step 2 back to standard filename:

cp \$HOME/.realtime.cfg.save \$HOME/.realtime.cfg

Results

These steps update each session configuration file used for multilevel configuration management. When you open Calibre RealTime in the new release the multilevel configuration setup should work as intended.

Option and Environment Variable Control in Calibre RealTime Custom

Calibre RealTime Custom provides commands to control options and environment variables from the command interface of supported design tools. You can also set environment variables to specify the path to certain input files.

Environment Variable Control in the Design Tool

"Environment Variable Control in Calibre RealTime" on page 53

(Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker)

"Environment Variable Control in Calibre RealTime for Cadence Virtuoso" on page 295 (Cadence Virtuoso)

Option Control in the Design Tool

"Option Control in the Design Environment for Calibre RealTime" on page 58 (Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker)

"SKILL Commands to Get and Set Calibre RealTime Options" on page 249 (Cadence Virtuoso)

Environment Variable Settings for Input Files

"Environment Variable Settings for Calibre RealTime Input Files" on page 77 (All tools)

Environment Variable Control in Calibre RealTime

Calibre RealTime provides commands to get and set environment variables from the command interface of supported design tools. Calibre RealTime recompiles the rule file if an environment variable changes due to one of these commands.

Note_

If your rule file uses DRC Select Check and DRC Unselect Check statements with conditional directives and environment variables to select the checks that are run, keep in mind that the check selection recipe "Checks selected in the rules file" is the only recipe that obeys DRC [Un]Select Check statements.

Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker
 The following Tcl commands are available.

Table 2-5. Tcl Commands to Control Environment Variables in Calibre RealTime

Command	Description
SetEnvironmentVariable	Set an environment variable in Calibre RealTime and the design tool environment.
UnsetEnvironmentVariable	Unset an environment variable in Calibre RealTime and the design tool environment.
GetEnvironmentVariable	Get the value of an environment variable.
IsEnvironmentVariableDefined	Determine if an environment variable exists in the Calibre RealTime process.

• Cadence Virtuoso

See "Environment Variable Control in Calibre RealTime for Cadence Virtuoso" on page 295 for Skill commands available in Calibre RealTime for Cadence Virtuoso.

Tip



You can also use environment variables to set the path to certain input files; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

GetEnvironmentVariable

Available in: Calibre RealTime for Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker

Get the value of an environment variable.

Usage

::calibre::realtime::GetEnvironmentVariable *varName* [-check_existence]

Arguments

• varName

The name of the environment variable.

-check_existence

An optional argument that causes the tool to check if *varName* is defined before retrieving the value.

Return Values

-check_existence	Returned Value	
not present	String containing the value of <i>varName</i> .	
	A Tcl error is issued if <i>varName</i> does not exist.	
present	{varPresent varValue}, a two-element Tcl list with the following elements:	
	• varPresent — Boolean indicating whether varName exists.	
	• <i>varValue</i> — String containing the value of <i>varName</i> . <i>varValue</i> is an empty string if <i>varName</i> does not exist.	

Description

Retrieve the value of the environment variable *varName* in the context of Calibre RealTime. Enter the command in the command window for the design tool. The format of the returned value depends on the presence of the switch -check_existence.

Examples

This example gets the value of environment variable M1_CHECKS; the value is returned.

```
% ::calibre::realtime::GetEnvironmentVariable M1_CHECKS
1
```

This example gets the value of environment variable M1_CHECKS and specifies the switch -check_existence; a two element Tcl list is returned.

```
% ::calibre::realtime::GetEnvironmentVariable M1_CHECKS -check_existence
1 1
```

IsEnvironmentVariableDefined

Available in: Calibre RealTime for Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker

Determine if an environment variable exists in the Calibre RealTime process.

Usage

::calibre::realtime::IsEnvironmentVariableDefined varName

Arguments

• varName

The name of the environment variable.

Return Values

True (1) if the environment variable exists in the Calibre RealTime process; false (0) otherwise.

Examples

This example checks for the existence of the environment variable M1_CHECKS.

```
% ::calibre::realtime::IsEnvironmentVariableDefined M1_CHECKS
1
```

SetEnvironmentVariable

Available in: Calibre RealTime for Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker.

Set an environment variable in Calibre RealTime and the design tool environment.

Usage

::calibre::realtime::SetEnvironmentVariable varName varValue

Arguments

• varName

The name of the environment variable.

• varValue

The value of the environment variable to be set.

Return Values

None.

Description

Sets an environment variable in both Calibre RealTime and the host process for the design tool. Enter the command in the command window for the design tool.

If this command is issued, Calibre RealTime automatically recompiles the rule file the next time a Calibre DRC run is launched or the check recipe editor is invoked.

Examples

This example sets the environment variable M1_CHECKS to 1.

```
% ::calibre::realtime::SetEnvironmentVariable M1 CHECKS 1
```

UnsetEnvironmentVariable

Available in: Calibre RealTime Custom for Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker.

Unset an environment variable in Calibre RealTime Custom and the design tool environment.

Usage

::calibre::realtime::UnsetEnvironmentVariable varName

Arguments

• varName

The name of the environment variable.

Return Values

None.

Description

Unsets an environment variable in both Calibre RealTime and the host process for the design tool. Enter the command in the command window for the design tool.

If this command is issued, Calibre RealTime automatically recompiles the rule file the next time a Calibre DRC run is launched or the check recipe editor is invoked.

Examples

This example unsets the environment variable M1_CHECKS.

% ::calibre::realtime::UnsetEnvironmentVariable M1 CHECKS

Option Control in the Design Environment for Calibre RealTime

You can set the options in the Calibre RealTime Options dialog box using commands issued in supported design tools. This allows you to configure Calibre RealTime within the design environment.

The following commands are available:

Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker³
 Commands and options related to multiple Run Configurations are only available in Synopsys Custom Compiler and Synopsys Laker³, not in Calibre DESIGNrev.

Table 2-6. Tcl Commands to Set Calibre RealTime Options

Command	Description
AddConfiguration	Add a run configuration with a specified name.
DeleteConfiguration	Delete the run configuration with the specified name.
GetAPIVersion	Get the version of the Tcl API.
GetConfigurationIndependentOption ¹	Gets the value of a Calibre RealTime option which is independent of the run configuration.
GetConfigurationName ¹	Get the name of the run configuration corresponding to the specified ID.
GetConfigurationOption	Get the value of a Calibre RealTime option.
GetCurrentConfiguration ¹	Gets the configuration ID of the current run configuration.
SetConfigurationIndependentOption ¹	Set the value of a Calibre RealTime option which is independent of the run configuration.
SetConfigurationOption	Set the value of a Calibre RealTime option.
SaveConfiguration	Save the current Calibre RealTime configuration.

¹ Valid for Synopsys Custom Compiler and Synopsys Laker³ only

• Cadence Virtuoso

See "SKILL Commands to Get and Set Calibre RealTime Options" on page 249 for commands to use in the Cadence environment.

__ Tip _

You can also use environment variables to set the path to certain input files; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

AddConfiguration

Available in: Calibre RealTime for Synopsys Custom Compiler and Synopsys Laker³ Add a run configuration with a specified name.

Usage

::calibre::realtime::AddConfiguration configName

Arguments

• configName

A string specifying the name for the new Run Configuration.

Return Values

The configID of the new configuration, or -1 on failure.

Description

Creates a new configuration named *configName*. The configuration has default settings and is set as the current configuration.

Examples

This example adds a new configuration "MP_config". The ID of the new configuration is returned.

```
tcl> ::calibre::realtime::AddConfiguration MP_config
2
```

DeleteConfiguration

Available in: Calibre RealTime in Synopsys Custom Compiler and Synopsys Laker³ Delete the run configuration with the specified name.

Usage

::calibre::realtime::DeleteConfiguration configName

Arguments

• configName

A string specifying the name of the Run Configuration to delete.

Return Values

None.

Description

Deletes the configuration named *configName*. If there is only one configuration or the named configuration does not exist, the command is ignored.

Examples

This example deletes the configuration "MP_config".

```
tcl> ::calibre::realtime::DeleteConfiguration MP_config
```

GetConfigurationIndependentOption

Available in: Calibre RealTime in Synopsys Custom Compiler and Synopsys Laker³ Gets the value of a Calibre RealTime option which is independent of the run configuration.

Usage

::calibre::realtime::GetConfigurationIndependentOption optionName

Arguments

• optionName

The name of the option, where *optionName* can take the values given in the following table. The option names are case-sensitive.

optionName	Description	
RealTimeEnabled	Returns a Boolean value specifying whether Calibre RealTime is enabled.	
RunDRCOnEdit	Returns a Boolean value specifying whether the Run DRC On Edit mode is enabled.	
RunRveOnDrc	Returns a Boolean value specifying whether the Calibre RealTime Results Window is opened automatically after a run.	
CurrentConfiguration	Returns the ID of the current Run Configuration.	
MultiConfigRunMode	Returns the configuration run mode:	
	• Single — Run only the current configuration.	
	• Serial — Run the selected configurations in order.	

Return Values

As indicated in the preceding table.

GetConfigurationName

Available in: Calibre RealTime in Synopsys Custom Compiler and Synopsys Laker³ Get the name of the run configuration corresponding to the specified ID.

Usage

::calibre::realtime::GetConfigurationName configID

Arguments

configID

An integer value specifying the Run Configuration ID.

Return Values

A string containing the configuration name.

Examples

This example first uses GetCurrentConfiguration to get the ID of the current configuration, then finds the name.

```
tcl> ::calibre::realtime::GetCurrentConfiguration
1
tcl> ::calibre::realtime::GetConfigurationName 1
Default configuration
```

GetConfigurationOption

Available in: Calibre RealTime in Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker

Get the value of the specified Calibre RealTime option. In Calibre RealTime with Synopsys Laker³ and Synopsys Custom Compiler, the run configuration ID is an optional value—the value in the current run configuration is returned by default.

Usage

::calibre::realtime::GetConfigurationOption optionName [configID]

___Note



The *configID* option and *optionName*s related to multiple Run Configurations are only available in Synopsys Custom Compiler and Synopsys Laker³.

Arguments

• optionName

The name of the option, where *optionName* can take the values given in the following table. The option names are case-sensitive.

Table 2-7. optionName Values for GetConfigurationOption

optionName	Definition
RealTimeEnabled	Returns a Boolean value specifying whether Calibre RealTime is enabled.
RunDRCOnEdit	Returns a Boolean value specifying whether the Run DRC On Edit mode is enabled.
RunRveOnDrc (Valid for Synopsys Laker and Synopsys Custom Compiler only)	Returns a Boolean value specifying whether the Calibre RealTime Results Window is opened automatically after a run.
CurrentConfiguration ¹	Returns the ID of the current Run Configuration.
MultiConfigRunMode ¹	Returns the configuration run mode: • Single — Run only the current configuration. • Serial — Run the selected configurations in order.
SelectedToRun ¹	Returns a Boolean value specifying whether the given configuration is selected or not selected to run in Serial mode.
RuleFile	Returns the path to the rule file, if set.
RunsetFile	Returns the path to the runset file, if set.

Table 2-7. optionName Values for GetConfigurationOption (cont.)

optionName	Definition
RuleSource	Returns the value "rulefile" or "runset" indicating whether a rule file or a Calibre Interactive runset file is the source of DRC rules.
CTOFile	Returns the path to the CTO file, if set.
CTOFileEnabled	Returns a Boolean value specifying whether Calibre RealTime is using a CTO file.
LayerMapFile	Returns the path to the layer map file, if set.
LayerMapFileEnabled	Returns a Boolean value specifying whether Calibre RealTime is using a layer map file.
LayerMapFileMode (Not available in Calibre	Returns the mode used to process the layer map file. Returns one of the following values:
DESIGNrev or Synopsys Custom Compiler)	 merge — data from the layer map file is merged with information obtained from the technology library. replace — data from the layer map file is used exclusively.
ObjectMapFile	Returns the path to the object map file, if set.
ObjectMapFileEnabled (Not available in Calibre DESIGNrev)	Returns a Boolean value specifying whether Calibre RealTime is using an object map file.
HaloSize	Returns the size of halo in user units.
PolygonCountLimit	Returns the maximum number of polygons that are passed to Calibre, as set in the Calibre Options dialog box.
ResultLimitPerCheck	Returns the maximum number of results per check.
DrcHierLevelsMode (Valid for Calibre DESIGNrev only)	Returns the setting for "DRC hier levels mode," which specifies the hierarchy levels in the design that are checked during the run.
Zzstoruev omy)	 All — Check all hierarchy levels in the design. Displayed — Check only the hierarchy levels that are displayed in Calibre DESIGNrev.

Table 2-7. optionName Values for GetConfigurationOption (cont.)

optionName	Definition	
ResultDisplayMode	Returns the result display mode as a two element list of the form {region checks}, where the elements can have the following values:	
	region:	
	 area — only display results in the area of the previous DRC run. cell — display all known results in the cell. 	
	checks:	
	• all_checks — display results generated from all checks.	
	 recipe_checks — only display results generated from checks selected by the current recipe. 	
HighlightAfterDRCRun	Returns a Boolean value specifying whether DRC violations are highlighted after a DRC run.	
ClearExistingHighlights	Return a Boolean value specifying whether previously drawn highlights are cleared before displaying new ones.	
ZoomToHighlights	Returns a Boolean value specifying whether Calibre RealTime zooms to drawn highlights.	
HighlightZoomRatio	Returns the value of the highlight zoom ratio.	
	The zoom ratio is the ratio of the highlight area to the window area. A ratio less than one indicates a zoom window larger than the highlighted area.	
GroupHighlightColorsByChe ck	Returns a Boolean value specifying whether Calibre RealTime groups highlight colors by check.	
IncludeCheckDefinitionInTo oltip	Returns a Boolean value specifying whether check definitions are included in tooltips associated with DRC highlights.	
HighlightShowCheckNames	Returns a Boolean value specifying the state of the option "Show check names" to display the check name at the center of each result.	
	Only available in Synopsys Laker.	
HighlightTextSize	Returns the value of the text size for displaying check names when HighlightShowCheckNames is enabled.	
	Only available in Synopsys Laker.	
CheckSelectionRecipe	Returns the currently selected check selection recipe.	

¹ Only available in Synopsys Custom Compiler and Synopsys Laker³.

• configID

An optional integer value specifying the Run Configuration ID; only valid with Synopsys Laker³ and Synopsys Custom Compiler. The option value for the current run configuration is returned if *configID* is not specified. Also see GetConfigurationName.

Return Values

As indicated in Table 2-7.

Examples

The following command gets the check selection recipe. When running with Synopsys Laker³ the value for the current run configuration is returned.

```
% ::calibre::realtime::GetConfigurationOption CheckSelectionRecipe
Checks selected in the rules file
%
```

GetCurrentConfiguration

Available in: Calibre RealTime Custom in Synopsys Custom Compiler and Synopsys Laker³ Gets the configuration ID of the current run configuration.

Usage

::calibre::realtime::GetCurrentConfiguration

Arguments

None.

Description

Returns the ID of the current Run Configuration; the ID is an integer value.

Examples

This example first gets the ID of the current configuration, then finds the name using GetConfigurationName.

```
tcl> ::calibre::realtime::GetCurrentConfiguration
1
tcl> ::calibre::realtime::GetConfigurationName 1
Default configuration
```

SaveConfiguration

Available in: Calibre RealTime in Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker

Save the current Calibre RealTime configuration.

Usage

::calibre::realtime::SaveConfiguration

Arguments

None.

Return Values

None.

Description

The current configuration is saved to the file .realtime.cfg as described in "Location of the Calibre RealTime Session Configuration File" on page 42. If you are using multilevel configuration files, the options saved are determined by the order of precedence of the configuration files, as described in "Multilevel Session Configuration File Support in Calibre RealTime Custom" on page 44.

New option values set by SetConfigurationOption are not saved until you call SaveConfiguration, however, the new value can be seen in the Calibre RealTime Options dialog box immediately.

SetConfigurationOption

Available in: Calibre RealTime Custom in Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker

Set the value of a Calibre RealTime Custom option. In Calibre RealTime with Synopsys Laker³ and Synopsys Custom Compiler, the run configuration ID is an optional value—the value in the current run configuration is set by default.

Usage

::calibre::realtime::SetConfigurationOption optionName optionValue [configID]

___Note



The *configID* option and *optionName*s related to multiple Run Configurations are only available in Synopsys Custom Compiler and Synopsys Laker³.

Arguments

• optionName optionValue

The name of the option and its value, as given in the following table. The option names and values are case-sensitive.

Table 2-8. optionName and optionValue for SetConfigurationOption

optionName	Description	Allowed option Value
RealTimeEnabled	Enable or disable Calibre RealTime.	Boolean
	No return value.	
RunDRCOnEdit	Enable or disable "Run DRC on	Boolean
(Not available in	Edit" mode.	
Synopsys Custom Compiler)	No return value.	
- '		
RunRveOnDrc	Enable or disable automatically	Boolean
(Valid for Synopsys Laker and Synopsys	opening the Calibre RealTime Results Window after a run.	
Custom Compiler)	Returns the value set.	
CurrentConfiguration ¹	Set the current Run Configuration to the specified ID.	Integer.
		Configuration IDs start at 1.

Table 2-8. optionName and optionValue for SetConfigurationOption (cont.)

optionName	Description	Allowed option Value
MultiConfigRunMode ¹	Set the configuration run mode.	 Single — Run only the current configuration. Serial — Run the selected configurations in order.
SelectedToRun ¹	Specifies the given configuration as selected or not selected to run in Serial mode.	Boolean
RuleFile	Define the rule file. No return value.	pathname
RunsetFile	Define the runset file. Returns the path to the control file on success, an empty string on failure.	pathname
RuleSource	Set the source of rules for the DRC run. Returns true or false.	 rulefile — use a rule file. runset — use a Calibre Interactive runset.
CTOFile	Set the path to the CTO file. Returns name of CTO file.	pathname
CTOFileEnabled	Enable or disable using the CTO file. Returns an empty string.	Boolean
LayerMapFile	Set the path to the layer map file. Returns the pathname.	pathname
LayerMapFileEnabled	Enable or disable using the layer map file. Returns 0 or 1.	Boolean

Table 2-8. optionName and optionValue for SetConfigurationOption (cont.)

optionName	Description	Allowed optionValue
LayerMapFileMode (Not available in Calibre DESIGNrev)	Set the mode used to process data from the layer map file. Returns MERGE or REPLACE.	 merge — data from the layer map file is merged with information from the technology library. replace — data from the layer map file is used exclusively.
ObjectMapFile	Set the path to the object map file. Returns the pathname.	pathname
ObjectMapFileEnabled (Not available in Calibre DESIGNrev)	Enable or disable using the object map file. Returns 0 or 1.	Boolean
HaloSize	Set the size of halo expressed in user units.	positive real
PolygonCountLimit	Set the maximum number of polygons that are passed to Calibre, as set in the Calibre Options dialog box.	integer
ResultLimitPerCheck	Set the maximum number of results per check. Returns true or false.	positive integer or "all"
DrcHierLevelsMode (Valid for Calibre DESIGNrev only)	Specifies the hierarchy levels in the design that are checked during the run. This setting is not saved when you exit Calibre RealTime.	 All — Check all hierarchy levels in the design. Displayed — Check only the hierarchy levels that are displayed in Calibre DESIGNrev.

Table 2-8. optionName and optionValue for SetConfigurationOption (cont.)

optionName	Description	Allowed optionValue
ResultDisplayMode	Define which results are displayed by RealTime. The optionValue is a two element list defining the region in which results are displayed and the checks whose results are displayed. Returns true or false.	A two element list of the form {region checks}, with the following allowed values: region: • area — only display results in the area of the previous DRC run. • cell — display all known results in the cell. checks: • all_checks — results generated by all checks are displayed. • recipe_checks — only results generated by checks selected by the current recipe are displayed. Example: "cell recipe_checks"
HighlightAfterDRCRun	Enable or disable automatic highlighting of DRC violations after a DRC run.	Boolean
ClearExistingHighlights	Set the value of the option which determines whether previously drawn highlights are cleared before displaying new ones.	Boolean
ZoomToHighlights	Set the value of the option which determines whether RealTime zooms to drawn highlights.	Boolean
HighlightZoomRatio	Set the value of the highlight zoom ratio. The zoom ratio is the ratio of the highlight area to the window area. A ratio less than one results in a zoom window larger than the highlighted area.	positive real

Table 2-8. optionName and optionValue for SetConfigurationOption (cont.)

optionName	Description	Allowed optionValue
GroupHighlightColorsBy Check	Set the value of the option which determines whether RealTime groups highlight colors by check.	Boolean
IncludeCheckDefinitionI nTooltip	Set the value of the option which determines whether RealTime includes the check definitions in tooltips associated with DRC highlights.	Boolean
HighlightShowCheckNa mes	Set the state of the option "Show check names" to display the check name at the center of each result.	Boolean
HighlightTextSize	Only available in Synopsys Laker. Set the text size for displaying check names when HighlightShowCheckNames is enabled. Only available in Synopsys Laker.	positive real
CheckSelectionRecipe	Set the current check selection recipe. Returns true or false.	Check selection recipe name, enclosed in quotes.

¹ Only available in Synopsys Custom Compiler and Synopsys Laker³

configID

An optional integer value specifying the run configuration ID; only valid with Synopsys Laker³. The option value for the current run configuration is set if *configID* is not specified.

Return Values

Returns the new value unless otherwise stated in Table 2-8.

Description

The new value is not saved until you call SaveConfiguration, however, the new value can be seen in the Calibre RealTime Options dialog box immediately.

Calibre RealTime with Synopsys Laker³ uses Run Configurations. The following options are independent of the run configuration: RealTimeEnabled, RunDRCOnEdit, RunRveOnDrc, CurrentConfiguration, and MultiConfigRunMode.

Examples

```
% ::calibre::realtime::SetConfigurationOption RuleFile drc.rul
drc.rul
% ::calibre::realtime::SaveConfiguration
%
```

SetConfigurationIndependentOption

Available in: Calibre RealTime in Synopsys Custom Compiler and Synopsys Laker³ Set the value of a Calibre RealTime option which is independent of the run configuration.

Usage

::calibre::realtime::SetConfigurationIndependentOption optionName optionValue

Arguments

• optionName optionValue

The name of the option and its value. The available options and values are listed in the following table. The option setting applies to all Run Configurations.

optionName	Description	Allowed optionValue
RealTimeEnabled	Enable or disable Calibre RealTime.	Boolean
	No return value.	
RunDRCOnEdit	Enable or disable "Run DRC on Edit" mode.	Boolean
	No return value.	
RunRveOnDrc	Enable or disable automatically opening the Calibre RealTime Results Window after a run. Returns the value set.	Boolean
CurrentConfiguration	CurrentConfiguration Set the current Run Configuration to	
	the specified ID.	Configuration IDs start at 1.
MultiConfigRunMode	Set the configuration run mode.	 Single — Run only the current configuration. Serial — Run the selected configurations in order.

Return Values

Returns the new value unless otherwise stated in the preceding table.

GetAPIVersion

Available in: Calibre RealTime in Calibre DESIGNrev, Synopsys Custom Compiler, and Synopsys Laker

Get the version of the Tcl API.

Usage

::calibre::realtime::GetAPIVersion

Arguments

None.

Return Values

Returns the API version as a two element list: {major_version_num minor_version_num}

Examples

```
% ::calibre::realtime::GetAPIVersion
1 0
```

Environment Variable Settings for Calibre RealTime Input Files

You can set environment variables to specify the path to certain input files for Calibre RealTime. The environment variables apply to all supported design tools.

The following environment variables set the path to the corresponding input file:

```
MGC_REALTIME_RULE_FILE
MGC_REALTIME_RUNSET_FILE
MGC_REALTIME_CTO_FILE
MGC_REALTIME_LAYER_MAP_FILE
MGC_REALTIME_OBJECT_MAP_FILE
```

The Calibre Interactive environment variable MGC_CALIBRE_DRC_RUNSET_FILE takes precedence over MGC_REALTIME_RUNSET_FILE. Multiple runset files can be specified using either variable; for example:

```
setenv MGC REALTIME RUNSET FILE "drc runset A drc runset B"
```

If multiple runsets are used, they are loaded in order and the settings loaded last take precedence.

The environment variables are not expanded when the settings are saved in the configuration file or viewed in the Calibre RealTime Options dialog box. In other words, if MGC_REALTIME_RULE_FILE is set, \$MGC_REALTIME_RULE_FILE appears in the Calibre RealTime Options dialog box for the rule file path.

Environment variable settings for input files take precedence over settings in the Calibre RealTime configuration file.

Note_

These environment variables are not suitable when using multiple configurations; only the first configuration in the list uses the files specified by these environment variables. When using multiple configurations, it is recommended that you disable these environment variables by setting MGC REALTIME DISABLE FILE SETTINGS ENVS.

Also see "Calibre RealTime Environment Variables" on page 333.

License Timeout for Calibre RealTime Custom

The Calibre RealTime Custom license is checked out when you start a Calibre RealTime Custom run. You can set the timeout period for the Calibre RealTime Custom license. The license timeout period also applies to the Calibre RealTime Results window.

License timeout is set with the following environment variable:

MGC_REALTIME_RELEASE_LICENSE_TIME

Sets the timeout period for the Calibre RealTime Custom license. The license is released after the specified timeout period if there is no activity. The license timeout also applies to the Calibre RVE license used for the Calibre RealTime Results Window.

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_REALTIME_RELEASE_LICENSE_TIME 0.3
setenv MGC_REALTIME_RELEASE_LICENSE_TIME 18m
setenv MGC_REALTIME_RELEASE_LICENSE_TIME "18 min"

The default timeout is 10 minutes if MGC_REALTIME_RELEASE_LICENSE_TIME is not defined. The minimum timeout is 2 minutes; if a timeout of less than 2 minutes is specified the timeout is set to 2 minutes.

Related Topics

Viewing Results in the Calibre RealTime Results Window

Debugging Double Patterning Results with Calibre RealTime Custom

A typical Double Patterning (DP) rule file derives layers for each mask and for the conflict and warning rings associated with a DP violation. Debugging a DP violation involves testing various possible fixes to the layout and choosing the optimum fix. This process is made easier with Calibre RealTime Custom because you can immediately see the results of a layout change.

To do DP debug with Calibre RealTime Custom, you output each mask and the warning and conflict layers as rule checks, so that the layers are output as results in Calibre RealTime Custom. Then, you assign highlight colors to each rule check using DRC Rule Check Comments for Calibre RealTime Custom — this step makes it easy to interpret the results.

When you run Calibre RealTime Custom, the mask layers and warning and conflict rings are easily identified by their highlight color (typically red for conflict and yellow for warning). When you make a layout change in order to fix a DP violation, you can easily see whether the violation was fixed or if it caused new violations.

The video "How To Debug Double Patterning Results with Calibre RealTime" covers these items:

- Debugging double patterning results
- Setting highlight color with the CTO file and RVE Highlight Color statement
- Fixing errors in Calibre RealTime



Prerequisites

- Calibre RealTime Custom Requirements are met.
- A DP rule file that outputs each mask layer and the conflict and warning layers as rule checks. The following rule file excerpt shows an example:

Procedure

Create a Check Text Override (CTO) File for Calibre RealTime to assign highlight
colors to the different masks, conflict and warning rings, and DRC results. The
following example corresponds to the rule file excerpt shown in the "Prerequisites"
section.

```
#CTO file for DP results
m1_warning_rings
RVE Highlight Color: orange
m1_conflict_rings
RVE Highlight Color: red
m1_mask0
RVE Highlight Color: green
m1_mask1
RVE Highlight Color: blue
m1_width
RVE Highlight Color: magenta
```

- 2. Save the CTO file with a file extension of ".cto"; for example, *dp_rule.cto*. The .cto file extension is not required, but the Calibre RealTime Options dialog box filters on this file extension, making it easier to locate the CTO file.
- 3. Open the design with Calibre RealTime Custom enabled in your design tool. See the chapter for your design tool for more information.
- 4. Click the **Options** button () to open the Calibre Options dialog box.
 - a. (Optional) Enable "Run DRC on Edit" if it is available in your integration.
 - b. Specify the "Rule file."

- c. Enable the checkbox for "CTO file" and specify the CTO (Check Text Override) file.
- d. If necessary, specify a "Layer map file." See "Layer Maps in Calibre RealTime" on page 39.
- e. (Optional) Set other options; see the description of the Calibre Options dialog box in the chapter for your design tool.
- f. (Optional) Select the "Check selection recipe" in the dropdown list. The default recipe (Checks selected in the rule file) may be appropriate for your rule file, or you can select a different check recipe. See "Check Recipes in Calibre RealTime Custom" on page 105 for more information.
- g. Click **OK** to close the dialog box.
- 5. Zoom to the region in your layout that you want to analyze.
- 6. Click (or in Pyxis Layout) in the Calibre RealTime Custom toolbar to start a DRC run on the window.

The following figure shows results for a design with the rule file excerpt shown in the Prerequisites. One conflict and one warning ring are reported. The warning ring indicates the area to avoid when making layout changes to fix the DP violation.

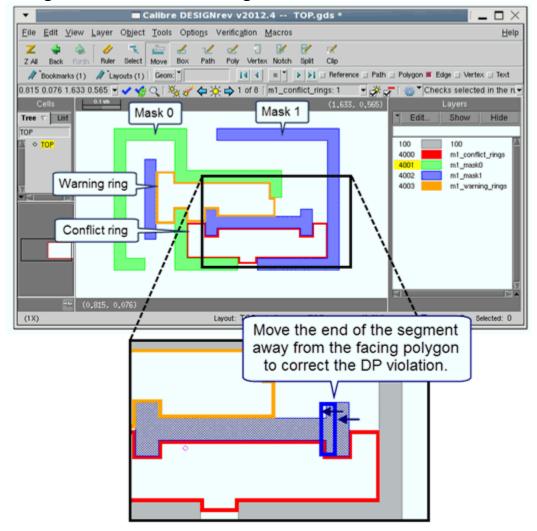


Figure 2-6. Double Patterning Results in Calibre RealTime Custom

7. Correct the violation, as shown in the inset for the preceding figure.

If "Run DRC on Edit" is not enabled, Click or be to start another run. (With "Run DRC on Edit" enabled, new results from a design edit are shown immediately)

Results are shown in the next figure. For this example, the change fixed the double patterning error, but introduced a DRC width error (m1_width).

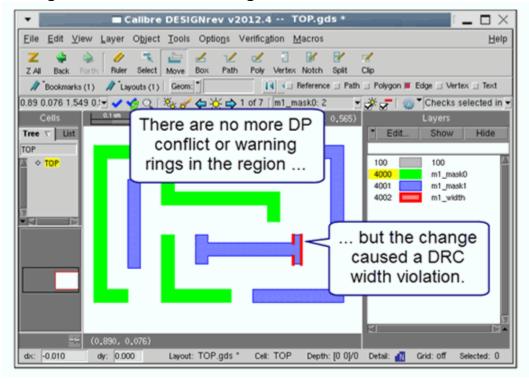


Figure 2-7. Double Patterning Fix in Calibre RealTime Custom

If the layout change causes new or different DP violations, these are shown immediately as new conflict and warning rings. The immediate feedback in Calibre RealTime Custom allows you to try several possible fixes for a DP violation, and then choose the best fix.

Related Topics

Calibre Multi-Patterning Users and Reference Manual

Multi-Patterning Support in Calibre RealTime Custom in Cadence Virtuoso

Viewing Results in the Calibre RealTime Results Window

You can view results from a Calibre RealTime run in the Calibre RealTime Results window, which is similar to Calibre RVE for DRC. The window includes a tree view, a details view, a check text pane, and controls for highlighting. The Calibre RealTime Results Window is available in the integrations to Cadence Virtuoso, Synopsys Custom Compiler, and Synopsys Laker³.

Prerequisites

- You are using Calibre RealTime with Cadence Virtuoso, Synopsys Custom Compiler, or Synopsys Laker and have completed a Calibre RealTime run. See one of the following:
 - "Invoking and Running Calibre RealTime with Cadence Virtuoso" on page 209
 - o "Running Calibre RealTime in Synopsys Custom Compiler" on page 192
 - o "Running Calibre RealTime with Synopsys Laker" on page 153
- (Optional) A Check Text Override File to specify highlight colors and layers; see "Specifying the Check Text Override (CTO) File in Calibre RealTime" on page 91.
- A Calibre RVE license. See "License Timeout for Calibre RealTime Custom" on page 78 for setting the license timeout period.

Tip

Enable "Launch RealTime results viewer automatically" at the top of the Calibre RealTime Options dialog box to have the Calibre RealTime Results Window open automatically after a run.

Video

The video "How to Review Calibre RealTime DRC Results" demonstrates basic procedures in reviewing results using the Calibre RealTime results viewer.

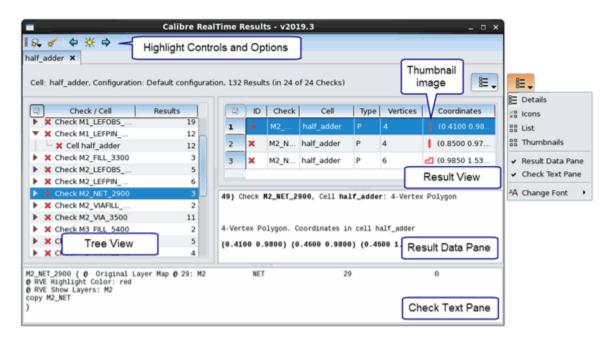


Procedure

1. Click the toolbar icon to open the Calibre RealTime Results window if it did not open automatically.

The results are shown in a window that is similar to Calibre RVE for DRC. Results from different cells are shown in different result tabs. If you are using multiple Run

Configurations, results from different configurations are displayed in different result tabs.



- 2. Set highlighting options with the R dropdown menu in the upper left corner.
- 3. Set font size with the Change Font option under the dropdown menu in the upper right corner. The default font size is Normal. Changing this configuration impacts the size of the text in all tabs and panes.
- 4. Click the icon above the result view to set the result view format; the icon image indicates the current result view format.
- 5. To highlight a result, double-click a result in the result view or click the highlight icon (--).

Usage Notes:

- In Cadence Virtuoso, if you do a descend edit the results tab associated with the cellview closes automatically.
- If a cell is closed in the design tool, the corresponding result tab closes automatically.

You can use the following keyboard shortcuts:

Table 2-9. Keyboard Shortcuts in Calibre RealTime Results Window

Keyboard Shortcut	Description
H, h	Highlight the current result, or the selection in the tree view if no result is selected.
F4	Clear highlights.
Left arrow, p, P	Move selection in Result View to previous result (up) and highlight.
Right arrow, n, N	Move selection in Result View to next result (down) and highlight.
Ctrl-up arrow, Ctrl-down arrow	Move selection in Tree View to next or previous leaf. No selection is made in the Result view.
Ctrl-left arrow,	Move selection in Tree View to previous leaf
Ctrl-P, Ctrl-p	and highlight the first result in the cluster.
Ctrl-right arrow,	Move selection in Tree View to next leaf and
Ctrl-N, Ctrl-n	highlight the first result in the cluster.

Tip

When using Cadence Virtuoso you can redefine the keyboard shortcuts for highlighting the previous, current, and next result by using the following command syntax:

```
hiSetBindKeys( "Layout" list(
   list("<Key>H" "mgc_calibre_realtime_send_highlight_cmd(\"CurrentError\")")
   list("<Key>N" "mgc_calibre_realtime_send_highlight_cmd(\"NextError\")")
   list("<Key>P" "mgc_calibre_realtime_send_highlight_cmd(\"PrevError\")")
)) ;; hiSetBindKeys
```

Using Calibre Interactive Settings in a Calibre RealTime Custom Run

If you use Calibre[®] Interactive[™] nmDRC and want to use the same options for your Calibre RealTime Custom run, you can specify the Calibre Interactive runset in the Calibre RealTime Options dialog box. Calibre RealTime Custom calls Calibre Interactive to generate the Calibre Interactive control file, and the control file is used for the Calibre RealTime Custom run. If the runset includes a customization file, the Customization GUI is opened and the settings are applied to the control file.

Specifying a Calibre Interactive Runset in Calibre RealTime	87
Using the Customization GUI with Calibre RealTime	88

Specifying a Calibre Interactive Runset in Calibre RealTime

You can specify a Calibre Interactive runset instead of a rule file for a Calibre RealTime run. Calibre RealTime calls Calibre Interactive to generate the Calibre Interactive control file. The generated control file is used for the Calibre RealTime run, so that all the options set in the runset are used for the Calibre RealTime run. You can use the Calibre RealTime Options dialog box or an environment variable to specify Calibre Interactive runsets.

You can specify a runset in the Calibre RealTime Options dialog box or using the environment variable MGC_REALTIME_RUNSET_FILE or MGC_CALIBRE_DRC_RUNSET_FILE. See "Environment Variable Settings for Calibre RealTime Input Files" on page 77 for more information on using the environment variables. If the runset file is set with an environment variable, the environment variable is displayed in the Calibre RealTime Options dialog box. The environment variable takes precedence over settings saved in the configuration file.

The following procedure describes how to specify the runset using the Calibre RealTime Options dialog box.

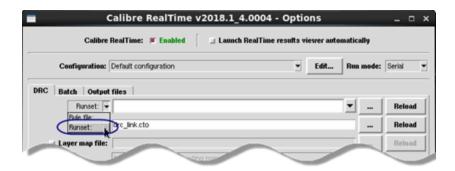
Prerequisites

- A Calibre nmDRC rule file.
- A Calibre Interactive runset. The runset must compile in Calibre Interactive and the DRC Run Directory must exist.
- A Calibre Interactive license.
- Your design tool is open with Calibre RealTime enabled.

Procedure

1. Choose **Verification > RealTime > Options** to open the Calibre RealTime Options Dialog Box, or click the **Options** button () on the integrated toolbar.

2. Select Runset in the Rule file/Runset dropdown list.



- 3. Enter the runset filename. If you are using multiple runsets, click the down arrow button and enter multiple filenames.
- 4. If the runset has changed on disk or if you are changing runset paths, click **Reload** to update the runset.
- 5. Close the Calibre RealTime Options dialog box.

Results

When you start a Calibre RealTime run or click **Reload**, Calibre RealTime calls Calibre Interactive to generate the Calibre Interactive control file from the runset. The control file is named_rule_file_, where rule_file is the name of the rule file specified in the runset. The file _rule_file_ is used as the rule file for the Calibre RealTime run.

The following items govern use of the settings in the runset:

- If there is a conflict, Calibre RealTime settings take precedence over settings in the runset.
- Settings in the runset that do not apply to Calibre RealTime are ignored. For example, most run control settings are not relevant for Calibre RealTime.
- If you are using multiple runsets, the runsets are loaded in the order they are listed in the Calibre RealTime Options dialog box or the MGC_CALIBRE_DRC_RUNSET_FILE environment variable. The settings in the most recently loaded runset file take precedence if there is a conflict.

Related Topics

Using the Customization GUI with Calibre RealTime

Using the Customization GUI with Calibre RealTime

If you use a customization file with Calibre Interactive nmDRC, you can use the same customization file with Calibre RealTime. This allows you to make customization settings at any time and apply them to your Calibre RealTime session.

Prerequisites

- A Calibre nmDRC rule file.
- A Calibre Interactive customization file. See "Customization Files" in the *Calibre Interactive User's Manual*.
- A Calibre Interactive runset. The runset must compile in Calibre Interactive and the DRC Run Directory must exist.

The runset should specify the customization file.

You can alternatively specify the customization file with the environment variable MGC_CALIBRE_CUSTOMIZATION_FILE. If a customization file is specified with both the environment variable and the runset, the runset setting takes precedence.

- A Calibre Interactive license.
- Your design tool is open with Calibre RealTime enabled.

Procedure

1. Specify the runset in the Calibre RealTime Options dialog box, as described in "Specifying a Calibre Interactive Runset in Calibre RealTime" on page 87.

The Customization GUI is automatically opened the first time the runset is compiled.

- 2. Make settings as needed in the Customization Settings dialog box, then click **OK**.
- 3. Run Calibre RealTime.
- 4. (Optional) If you want to change customization settings, click the button in the Calibre RealTime toolbar to open the Customization GUI.

If you make customization changes, the runset and rule file are recompiled and used for the next Calibre RealTime run. The changes you make to the Customization GUI are retained during the Calibre RealTime session until you click **Reload** in the Calibre RealTime Options dialog box for the runset file—the reload action returns the Customization GUI to the settings saved in the runset.

Related Topics

Specifying a Calibre Interactive Runset in Calibre RealTime

DRC Rule Check Comments for Calibre RealTime Custom

DRC Rule Check Comments are one way to specify the highlight color or highlight layer for Calibre RealTime Custom results. Rule check comments are supported for all tools with Calibre RealTime Custom. The rule check comments specify the highlight color or highlight layer index for a given rule check so that results from a given rule check are always highlighted in the same color.

The following table lists the available rule check comments in Calibre RealTime Custom:

Table 2-10. DRC Rule Check Comments in Calibre RealTime Custom

Rule Check Comment	Description
RVE Highlight Index	Specifies the highlight layer index used for the rule check.
RVE Highlight Color	Specifies the highlight layer color used for the rule check in supported layout viewers.
RVE Show Layers	Specifies the layers that are shown in the layout editor when highlighting a result from the rule check. Supported for Cadence Virtuoso and Calibre DESIGNrev.
RVE Link (Cadence Virtuoso only)	Creates a hyperlink that is displayed in the check text pane of the Calibre RealTime Results window for the rule check.

Place the rule check comments in your rule file or a configuration file called the Check Text Override (CTO) file. Rule check comments in the CTO file take precedence over those in the rule file if there is a conflict. CTO files are not supported with Synopsys Laker. Specify the rule check comments as described in the following table:

Table 2-11. Specifying DRC Rule Check Comments in Calibre RealTime Custom

Location of Rule Check Comments	Method to Specify Rule Check Comments
In the rule file with a preceding ampersand (@) character.	No further action needed. See the RVE Highlight Index and RVE Highlight Index commands for examples.
In the Check Text Override (CTO) File for Calibre RealTime.	In the Calibre RealTime Options dialog box; see "Specifying the Check Text Override (CTO) File in Calibre RealTime" on page 91.
	The CTO file is not supported with Synopsys Laker.



Calibre RVE for DRC supports additional rule check comments; Calibre RealTime Custom ignores rule check comments that it does not support.

You may want to see these topics in other manuals:

- "Rule Comments" in the Calibre Verification User's Manual
- "DRC Rule Check Comments for Calibre RVE" in the Calibre RVE User's Manual

Specifying the Check Text Override (CTO) File in Calibre RealTime

Supported in: Calibre RealTime with Cadence Virtuoso, Pyxis Layout, and Calibre DESIGNrev You can specify DRC Rule Check comments for Calibre RealTime in a Check Text Override file (CTO file). The CTO file is specified in the Calibre RealTime Options dialog box.

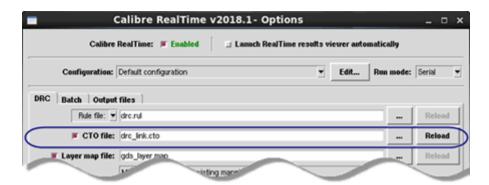
Rule check comments in the check text override file take precedence over those in the rule file if there is a conflict.

Prerequisites

• A CTO file; see "Check Text Override (CTO) File for Calibre RealTime" on page 101. The file browser for the CTO file is set up to search for files with a .cto file extension.

Procedure

- 1. In Calibre RealTime, click the button to open the Calibre RealTime Options dialog box.
- 2. Enable "CTO file" and specify a check text override file.



Tip

You can also use the environment variable MGC_REALTIME_CTO_FILE to set the path to the CTO file; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

Related Topics

DRC Rule Check Comments for Calibre RealTime Custom

RVE Highlight Index

Tool Support: Calibre RealTime with all supported design tools

A DRC RVE rule check comment that specifies the highlight layer index used for the rule check.

Usage

RVE Highlight Index: index

Arguments

• index

A required argument, where index values up to 32767 can be used. However, most layout viewers limit the number of layers available for highlighting, so the highlight layer index may start over after the maximum index is reached.

A space is required between the colon (:) and *index*.

Description

This rule check comment specifies the highlight layer index. Highlights for the rule check are always made to that layer index.

See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90 for instructions on specifying the rule check comment. The rule check comment can be added directly to the rule file or specified in a Check Text Override (CTO) file.

RVE Highlight Index and RVE Highlight Color rule check comments should not be used together in a rule file or a Check Text Override (CTO) File for Calibre RealTime. If both are found in the DRC results database, the RVE Highlight Index statements take precedence and the RVE Highlight Color rule check comments are ignored.

For a description of this rule check comment in the Calibre RVE environment, see "RVE Highlight Index" in the Calibre RVE User's Manual."

Examples

Example 1: In a Rule Check

```
// rule file
// ...
metal1_space_rule {
@ comment
@ RVE Highlight Index: 5
< rules ...>
}
```

Example 2: In a CTO File

DRC RVE .cto file
metal1_space_rule
RVE Highlight Index: 5

Related Topics

DRC Rule Check Comments for Calibre RealTime Custom Check Text Override (CTO) File for Calibre RealTime RVE Highlight Color

RVE Highlight Color

Tool Support: Calibre RealTime with all supported design tools

A DRC RVE rule check comment that specifies the highlight layer color used for the rule check in supported layout viewers.

Usage

RVE Highlight Color: color

Arguments

• color

A required argument which specifies the highlight color used for the rule check. The color may be any valid color in Tk. For example, you can use the RGB color specification starting with a '#' or a color name understood by your X server, so the color blue can be indicated in these ways: blue, #0000FF, or #00f.

A space is required between the colon (:) and *color*.

Description

This rule check comment specifies the highlight layer color for the rule check; all results from the rule check are highlighted with this color in the attached layout viewer.

See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90 for instructions on specifying the rule check comment. The rule check comment can be added directly to the rule file or specified in a Check Text Override (CTO) file.

RVE Highlight Index and RVE Highlight Color rule check comments should not be used together in a rule file or a Check Text Override (CTO) File for Calibre RealTime. If both are found in the DRC results database, the RVE Highlight Index statements take precedence and the RVE Highlight Color rule check comments are ignored.

For a description of this rule check comment in the Calibre RVE environment, see "RVE Highlight Color" in the Calibre RVE User's Manual."

Examples

Example 1: In a Rule Check

```
// rule file
// ...
my_rule {
@ comment
@ RVE Highlight Color: blue
< rules ...>
}
```

Example 2: In a CTO File

DRC RVE .cto file
my_rule
RVE Highlight Color: blue

Related Topics

DRC Rule Check Comments for Calibre RealTime Custom Check Text Override (CTO) File for Calibre RealTime RVE Highlight Index

RVE Show Layers

Tool Support: Calibre RealTime with Calibre DESIGNrev/WORKbench and Cadence Virtuoso A DRC RVE rule check comment that specifies the layers that are shown in the layout editor when highlighting a result from the rule check.

Usage

RVE Show Layers: *layer1* [*layer2* ... *layerN*]

Arguments

• *layer1* [*layer2* ... *layerN*]

A list of layers; at least one layer must be specified. Layer names or numbers can be specified. A space is required between the colon (:) and *layer1*. The terminating delimiter for the layers list is a newline. Wildcards (*) may be used in the layer designation.

Some specific usage notes apply to the following design tools:

o Cadence Virtuoso

The layer argument can be specified in these ways for Cadence Virtuoso:

layer:purpose — You can use names or numbers for the layer and layer purpose. For example, MT1:drawing or 6:1. Wildcards (*) may be used only for the layer name. Using a wildcard for the layer purpose results in no match being found.

layer_name — You can specify the layer name by itself. Wildcards may be used for the layer name. If no wildcard is used in the layer name the purpose layer defaults to the drawing purpose layer. If a wildcard is used in the layer name, all purpose values are matched.

Calibre DESIGNrev

If layer names are used in the rule check comment, you must load a layer properties file in Calibre DESIGNrev to specify the correspondence between layer numbers and layer names.

You can specify a layer and datatype pair as *layer.datatype*. Wildcards (*) can be used for both the layer and datatype.

See "Example 3: Wildcard Matching in Cadence Virtuoso and Calibre DESIGNrev" on page 97.

Description

This rule check comment specifies the layers that are displayed in the layout editor when highlighting a result from the rule check. All layers not specified in this statement are automatically hidden when any results from the check are highlighted.

See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90 for instructions on specifying the rule check comment. The rule check comment can be added directly to the rule file or specified in a Check Text Override (CTO) file.

_Note

If you highlight results from a rule check with a RVE Show Layers rule check comment, and next highlight results from a rule check *without* a RVE Show Layers rule check comment, the layer visibility set with the first rule check is retained. You can use the controls in your design tool to set the layer visibility as desired.

For a description of this rule check comment in the Calibre RVE environment, see "RVE Show Layers" in the Calibre RVE User's Manual."

Examples

Example 1: Rule Check Comment in the Rule File

The following rule check shows an example using GDS layer names for use with the Calibre DESIGNrev layout editor.

```
// rule file ...
min_ext_oxide_poly {
    @ minimum enclosure of poly by oxide
    @ RVE Show Layers: poly oxide
    @ RVE Highlight Color: blue
    enc oxide poly < 1.25
}</pre>
```

Example 2: Rule Check Comment in the CTO File

The following example shows a CTO file with the RVE Show Layers and RVE Highlight Color rule check comments:

```
# DRC RVE check text override file for XYZ
#
min_ext_oxide_poly
RVE Show Layers: poly oxide
RVE Highlight Color: blue
# ...
```

Example 3: Wildcard Matching in Cadence Virtuoso and Calibre DESIGNrev

The following examples show how wildcard matching occurs for Calibre DESIGNrev and Cadence Virtuoso:

• Calibre DESIGNrev

Suppose these layers exist in the design: 237, 237.2, 238, 238.2 and 238.3.

The following table shows how wildcard matching is performed:

Command	Matched Layers
RVE Show Layers: 23*.*	237.2, 238.2 and 238.3
RVE Show Layers: 23*	All layers (237, 237.2, 238, 238.2 and 238.3)
RVE Show Layers: *.2	237.2 and 238.2

• Cadence Virtuoso

Suppose the following layer:purpose pairs exist in the design: M1:drawing, M12:label, M144:drawing, M2:drawing, M22:label, L11:drawing, L181:drawing, L25:drawing.

The following table shows how wildcard matching is performed:

Command	Matched Layers
RVE Show Layers: M1	M1:drawing
RVE Show Layers: M1*	M1:drawing, M12:label, M144:drawing
RVE Show Layers: L1*1:drawing	L11:drawing, L181:drawing
RVE Show Layers: M1*:draw*	No match.

Related Topics

DRC Rule Check Comments for Calibre RealTime Custom

Check Text Override (CTO) File for Calibre RealTime

RVE Highlight Index

RVE Highlight Color

RVE Link

Tool Support: Calibre RealTime Custom with Cadence Virtuoso and Synopsys Laker in the Calibre RealTime Results window only

A DRC RVE rule check comment which creates a hyperlink that is displayed in the check text pane of the Calibre RealTime Results window for the rule check.

Usage

RVE Link: *document*[#anchor_tag] [TEXT *display_text*]

Arguments

• document

A required argument giving the path to the document to be opened.

Some HTML browsers may require "file://" to precede the file path of the document.

A space is required between the colon (:) and *document*.

#anchor_tag

An optional character (#) and argument (anchor_tag), where anchor_tag specifies a named destination within **document**.

TEXT display_text

An optional keyword and argument specifying the display text for the hyperlink. If this keyword and argument are not included, the command itself is used for the hyperlink text.

Description

This rule check comment creates a hyperlink which is displayed in the check text pane for the rule check. When the hyperlink is clicked, the specified *document* is opened in an HTML browser at the named destination given by *anchor_tag*, if present.

Environment variables may not be used in the *document* path.

See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90 for instructions on specifying the rule check comment. The rule check comment can be added directly to the rule file or specified in a Check Text Override (CTO) file.

HTML Browser Specification and Compatibility When Using RVE LINK

Set the environment variable MGC_RVE_HTML_BROWSER to the path for a browser executable, and that browser is opened when viewing a document with the RVE Link rule check comment. A recent browser version is recommended, as unexpected results may occur in older browsers.

Troubleshooting

If PDF documents do not open correctly, or do not open to a page destination, check the following:

- The Adobe Reader plug-in is correctly specified for your HTML browser.
- Other add-on tools or extensions do not duplicate or conflict with the plug-in specification.
- HTML browser preferences for PDF documents are set correctly.
- Shell scripts do not cause conflicts with the Adobe Reader plug-in specification.

Examples

Example 1

The following example shows the RVE Link rule check comment placed in a rule check. The path to the document is preceded by "file://", as is required by some HTML browser versions.

```
// rule file
// ...
min_spacing_metal1 {
@ Minimum metal1 spacing = 1.0
@ RVE Link: file:///docs/wide_space.html#Procedure TEXT metal rules
ext metal1 < 1.0
}</pre>
```

The named destination can be provided in the HTML document with code similar to this:

```
<a name="Procedure"></a>
```

Example 2

The following example shows the RVE Link rule check comment placed in a Check Text Override (CTO) File for Calibre RealTime. An environment variable DOC_DIR is defined to specify the directory name of the document being referenced. (In this example the document path is not preceded by "file://", but this may be necessary for some HTML browsers).

```
# DRC RVE check text override file for XYZ
min_spacing_metal1
RVE HIGHLIGHT COLOR: green
RVE Link: /docs/wide_space.html TEXT doc:wide metal rules
```

Check Text Override (CTO) File for Calibre RealTime

Input for: Calibre RealTime with Cadence Virtuoso, Pyxis Layout, and Calibre DESIGNrev The check text override file (CTO file) allows you to specify DRC RVE rule check comments for a Calibre RealTime run without editing the rule file. DRC RVE rule check comments can specify the highlight color or the highlight layer index for errors found during a Calibre RealTime run.

See "Specifying the Check Text Override (CTO) File in Calibre RealTime" on page 91 for instruction on loading the CTO file. DRC RVE rule check comments found in a CTO file take precedence over settings from the rule file.

Note_

For support with Calibre RVE for DRC and DRC HTML Reporting, see "DRC Rule Check Comments for Calibre RVE" in the Calibre RVE User's Manual.

Format

Formatting Rules

- ASCII text file. Blank lines are ignored, and white space at the beginning of a line is ignored.
- Comments Lines beginning with the # character are ignored.

Contents

The file contents are shown below, where [] indicates an optional element and "..." indicates optional repetition of the element.

```
checkname [checkname ...]
[@ check text comment]
RVE rule check comment
```

Parameters

checkname

A required parameter giving the name of one or more rule checks that the DRC RVE rule check comments apply to.

The * wildcard character may be used in *checkname*. Rule check comments accumulate if more than one rule check comment section matches a specific rule check due to the use of wildcard characters in *checkname*. If there is a conflict between accumulated rule check comments, then the latest rule check comment section in the Check Text Override file takes precedence; see the "Examples" section. Rule check sections without a wildcard in the check name always have the highest precedence.

@ check_text_comment

Text appearing after the @ character and on the same line is processed as a check text comment for the *checkname* rule check. Check text comments are displayed in the tooltip for the result. Check text present in the results database as a result of check text comments in the rule file is merged with check text comments in the CTO file.

• RVE_rule_check_comment

A DRC RVE rule check comment. More than one rule check comment may be specified. See Table 2-10 for a list of supported rule check comments.

Examples

Example 1: Basic Check Text Override File for Calibre RealTime

This example uses a wildcard to set the highlight color to blue for all rule checks with "metal1" in the rule check name and includes a check text comment. The highlight color is set to green for the rule check min_ext_diff_poly.

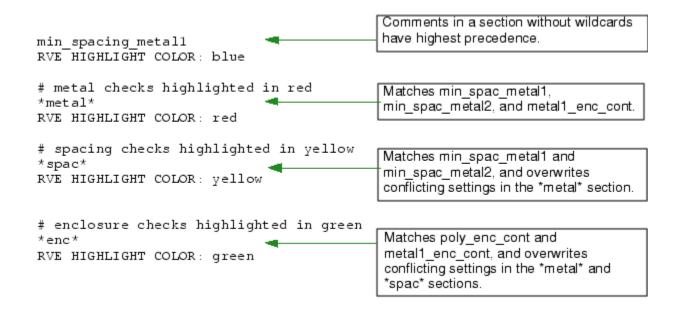
```
# check text override file for XYZ

*metal1*
@ Minimum metal1 spacing = 1.0
RVE Highlight Color: blue

min_ext_diff_poly
@ Minimum enclosure of poly by oxide = 1.25
RVE Highlight Color: green
```

Example 2: Precedence of Rule Check Comments With Wildcards in the Check Names

The following example demonstrates the precedence of rule check comments when the use of wildcards in the rule check name causes a rule check comment to apply to more than one rule check. For this example, assume rule checks with the following names: min_spac_metal1, min_spac_metal2, poly_enc_cont, and metal1_enc_cont.



Final settings:

```
min_spac_metal1:
   RVE HIGHLIGHT COLOR: blue
min_spac_metal2:
   RVE HIGHLIGHT COLOR: yellow
poly_enc_cont:
   RVE HIGHLIGHT COLOR: green
metal1_enc_cont:
   RVE HIGHLIGHT COLOR: green
```

Chapter 3 Check Selection Recipes in Calibre RealTime Custom

Calibre RealTime Custom uses a Check Selection Recipe to define the rule checks that are executed during the run. A number of built-in check recipes are provided, and you can define your own user-defined check recipes.

Check Recipes in Calibre RealTime Custom	105
Specifying a Check Recipe	106
Built-In Check Recipes	107
Hiding Built-In Check Recipes	109
Creating a Custom Check Selection Recipe	110
Basic Editing of Check Selection Recipes	111
Advanced Editing of Check Selection Recipes	113
Calibre RealTime Recipe Editor Dialog Box	116

Check Recipes in Calibre RealTime Custom

A check recipe is a set of rules for selecting the checks to execute for a Calibre RealTime run. Calibre RealTime Custom defines a set of built-in check recipes, and you can create custom check recipes.

Built-In Check Recipes

The built-in check recipes provide an easy way to select the checks to run. For example, the built-in recipe "Visible layer checks" runs checks that use at least one of the layers visible in the design tool window.

The check recipe "Checks selected in the rules file" is the only built-in recipe that executes the same set of checks that would be executed in a batch Calibre run, and the only built-in recipe that obeys DRC [Un]Select Check statements in the rule file. Other built-in recipes may execute checks in the rule file that would not be executed during a batch Calibre run.

Custom Check Recipes

You can create custom check recipes, also called *user recipes*, for specific tasks or designs. User recipes enable you to tailor the rule checks that are executed to the type of design work you

are doing. User recipes are automatically saved in the session configuration file when you exit the tool. You can also export user recipes to a file with a .rcp file extension.

You create and edit check recipes with the Calibre RealTime Recipe Editor dialog box.

Special Considerations and Tips

Care should be taken when evaluating the results of Density or Connectivity checks from a Calibre RealTime Custom run. These checks are often intended to run on the complete design, while Calibre RealTime Custom is often run on a targeted region.

You can hide certain built-in check recipes using an environment variable. This is useful if certain built-in check recipes should not be used in your environment. See "Hiding Built-In Check Recipes" on page 109.

Rule files often use environment variables to control check execution. You can change environment variables with the commands given in the section "Environment Variable Control in Calibre RealTime" on page 53.

Related Topics

Built-In Check Recipes

DRC Select Check [Standard Verification Rule Format (SVRF) Manual]

Session Configuration File in Calibre RealTime Custom

Calibre RealTime Recipe Editor Dialog Box

Density Checks in Calibre RealTime Custom

Specifying a Check Recipe

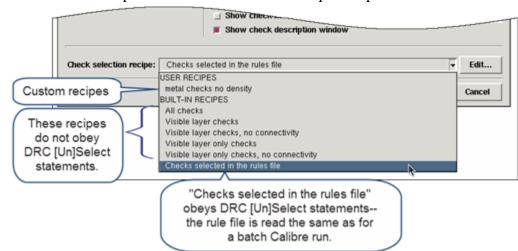
The check recipe selects which rule checks are executed during the Calibre RealTime run.

Prerequisites

• A supported design tool is open with Calibre RealTime enabled.

Procedure

- 1. Click the **Options** toolbar button () to open the Calibre RealTime Options Dialog Box.
- 2. Specify the rule file or a Calibre Interactive runset. This is required before selecting a check recipe.



3. Select the check recipe in the "Check selection recipe" dropdown list.

₋Tip

If the rule file is already loaded you can select the check recipe from the toolbar without opening the Calibre RealTime Options dialog box.

Built-In Check Recipes

Calibre RealTime includes several built-in check selection recipes. These built-in check recipes include commonly used check selections, such as "Checks selected in the rules file", "All checks", and "Visible layer checks."

The following table describes the available built-in check recipes. Except for the recipe "Checks selected in the rules file", the built-in recipes do *not* obey DRC [Un]Select Check statements in the rule file; therefore, these recipes may execute checks in the rule file that would not be executed during a batch Calibre run. The recipe "Checks selected in the rules file" is the only built-in recipe that executes the same set of checks that would be executed in a batch Calibre run, and the only built-in recipe that obeys DRC [Un]Select Check statements in the rule file. A Layer statement specifying a name for an original layer is required in the rule file in order to use the layer-based recipes.

Table 3-1. Built-in Check Recipes in Calibre RealTime

Recipe name	Description
All checks	Run all checks in the rule file, except those excluded by preprocessor directives. DRC [Un]Select Check statements are not considered.
Visible layer checks ¹	Run all checks which use at least one of the layers visible in the window.

Table 3-1. Built-in Check Recipes in Calibre RealTime (cont.)

Recipe name	Description
Visible layer checks, no connectivity ¹	Run all checks which use at least one of the layers visible in the window, but exclude checks which use connectivity.
Visible layer only checks ¹	Run checks which use visible layers only and no invisible layers.
Visible layer only checks, no connectivity ¹	Run checks which use visible layers only and no invisible layers, but exclude checks which use connectivity.
Recently edited layer checks (Only available with Synopsys	Run checks which use any layer containing shapes modified by the last shape edit operation.
Laker ³ and Calibre DESIGNrev)	Layers are only accumulated while this recipe is selected. If "Run DRC on Edit" is disabled, then layers are accumulated from all shape modifications since the last DRC run.
Recently edited layer checks, no connectivity (Only available with Synopsys Laker ³ and Calibre DESIGNrev)	Functions the same as the "Recently edited layer checks" recipe, but checks which use connectivity are excluded.
Checks selected in the rules file (default)	Run only checks selected in the rule file. This is the only built-in recipe that obeys the DRC [Un]Select Check statements in the rule file.
	Note that if the rule file does not include any DRC [Un]Select Check statements, then all checks are executed; this is the same behavior as for batch Calibre nmDRC.

Note

With the Cadence Virtuoso integration, in order to use the check recipes involving visible layers, the layer map file must exist in the technology library directory or be set with an environment variable. For information on specifying the layer map file see "GDS Layer Map for Calibre RealTime with Cadence Virtuoso" on page 242.

If your project team determines that certain built-in check recipes should not be used, they can be hidden from the list of built-in check recipes; see "Hiding Built-In Check Recipes" on page 109.

Hiding Built-In Check Recipes

You can use the environment variable MGC_CALIBRE_HIDDEN_RECIPES to hide specified built-in check recipes. Hidden check recipes do not appear in the dropdown list of check recipes in the toolbar, in the Calibre RealTime Options dialog box, or in the Calibre RealTime Recipe editor.

Hiding check recipes is useful if certain built-in check recipes, such as the ones for visible layers, should never be used. User-defined check recipes and the built-in check recipe "Checks selected in the rules file" cannot be hidden.

The environment variable value can be any of the following:

- A single string (the recipe name)
- One or more regular expressions or strings, where each expression or string is enclosed in braces or double quotes, and separated by a space.

Examples are given in the following table; in some cases two options for formatting the environment variable value are given.

Recipes to hide	MGC_CALIBRE_HIDDEN_RECIPES Value
The "All checks" recipe	"All checks"
All recipes that start with	{Visible.*} {Recent.*}
"Visible" or "Recent" 1	or
	"Visible.*" "Recent.*"
The recipes "All checks" and	{All checks} {Visible layer checks}
"Visible layer checks"	or
	"All checks" "Visible layer checks"
All recipes that end with "connectivity"	{.*connectivity}
	or
	".*connectivity"

¹ Some integrations do not include the "Recently edited ..." built-in recipes.

Procedure

- 1. Define the environment variable MGC_CALIBRE_HIDDEN_RECIPES.
- 2. Start your design tool with Calibre RealTime Custom enabled.

Examples

Some C shell (csh) examples:

```
setenv MGC_CALIBRE_HIDDEN_RECIPES "{Visible.*} {Recent.*}"
setenv MGC_CALIBRE_HIDDEN_RECIPES '".*connectivity"'
setenv MGC_CALIBRE_HIDDEN_RECIPES "All checks"
```

Some Bourne (sh) shell examples:

```
MGC_CALIBRE_HIDDEN_RECIPES="{Visible.*} {Recent.*}"
export MGC_CALIBRE_HIDDEN_RECIPES

MGC_CALIBRE_HIDDEN_RECIPES='".*connectivity"'
export MGC_CALIBRE_HIDDEN_RECIPES

MGC_CALIBRE_HIDDEN_RECIPES="\".*connectivity\""
export MGC_CALIBRE_HIDDEN_RECIPES

MGC_CALIBRE_HIDDEN_RECIPES="All checks"
export MGC_CALIBRE_HIDDEN_RECIPES="All checks"
```

Related Topics

Check Recipes in Calibre RealTime Custom

Built-In Check Recipes

Calibre RealTime Options Dialog Box

Calibre RealTime Recipe Editor Dialog Box

Creating a Custom Check Selection Recipe

Custom check recipes (also called user recipes) enable you to control the Calibre rule checks that are executed during a Calibre nmDRC run. You can create a new custom check recipe or start by copying an existing recipe.

Check recipe definitions are automatically saved in the Session Configuration File in Calibre RealTime Custom; you can also export a check recipe to a file.

Prerequisites

• A Calibre rule file is loaded; see "Calibre RealTime Options Dialog Box" on page 31.

Procedure

- 1. Click the **Options** toolbar button () to open the Calibre RealTime Options dialog box.
- 2. Click the **Edit** button to the right of the Check Selection Recipe entry.

A view of the recipe editor is shown in "Calibre RealTime Recipe Editor Dialog Box" on page 116.

- 3. Do one of the following:
 - Create a new recipe Click New in the top button bar. A new recipe is initialized with the checks selected in the rule file.
 - Copy an existing recipe Select the recipe to copy in the Recipe dropdown list, then click Copy in the top button bar. You are prompted for the name of the new recipe.
- 4. Proceed to one of the following tasks:
 - "Basic Editing of Check Selection Recipes" on page 111
 - "Advanced Editing of Check Selection Recipes" on page 113

Basic Editing of Check Selection Recipes

The basic editing controls in the Calibre RealTime Recipe editor dialog box 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 run. Check recipe definitions are automatically saved in the Session Configuration File in Calibre RealTime Custom; you can also export a check recipe to a file.

Prerequisites

• A Calibre rule file is loaded; see "Calibre RealTime Options Dialog Box" on page 31.

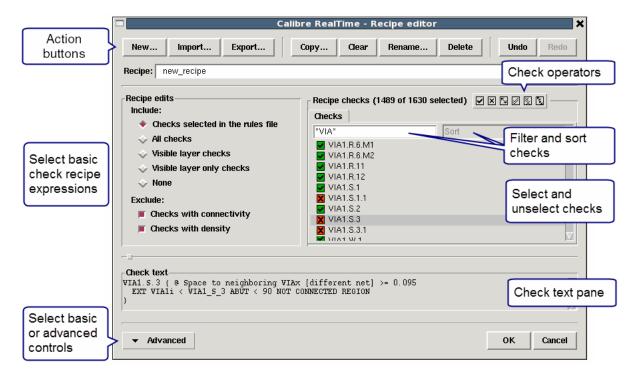
Procedure

- 1. Click the **Options** toolbar button () to open the Calibre RealTime Options dialog box.
- 2. Select 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.

3. Click **Edit** to open the Calibre RealTime Recipe Editor Dialog Box.

4. For basic recipe editing, make sure the dialog box appears as shown in the following figure. If it does not, click the **Advanced** button to collapse the advanced controls.



- 5. Filter and sort checks by their name. Checks can be filtered with different metacharacters.
 - * Matches any number of any characters. For example, "M1*" displays every check that beings with "M1". The default filter is "*", which displays every check.
 - ? Matches any single character. For example, "M?" displays every check that begins with "M" and has exactly one more character afterwards, such as "M1" and "M9".
 - **[chars]** Matches any character in the character set **chars**. For example, "M[1-3]" displays the checks "M1", "M2", and "M3".

If the check name has the meta-characters *?[] in it, escape the character with \. For example, "M*" displays the check "M*" only.

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

____Tip

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

9. Click **OK** to save the recipe and exit the dialog box.

Advanced Editing of Check Selection Recipes

The advanced editing controls in the Recipe Editor allow you to select checks by layer, select checks which did or did not produce results, and select and unselect individual rule checks. Unselecting individual checks is useful if a particular check gives false errors when run on a limited area.

Prerequisites

• A Calibre rule file is loaded; see "Calibre RealTime Options Dialog Box" on page 31.

Procedure

- 1. Open a recipe for editing and make selections using the basic editing controls, as described in "Basic Editing of Check Selection Recipes" on page 111.
- 2. If needed, click the **Advanced** button to expand the advanced editing controls. See Figure 3-4 in "Calibre RealTime Recipe Editor Dialog Box" on page 116 for a view of the dialog box.
- 3. View the check selection recipe in the Recipe definition area. If desired, remove expressions with the x button, or use the other selections in the button bar to operate on the recipe expressions.
- 4. Do the following to add an expression to the recipe definition:
 - a. Select an expression category in the Expression area. The expressions categories are defined in "Calibre RealTime Recipe Editor Dialog Box" on page 116.
 - b. Specify "Addition" or "Subtraction" below the Expression area to indicate whether you want to include or exclude the checks in the expression. Included checks are indicated with a and excluded checks are indicated with a ...
 - 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:
 - o Click the checkbox next to the check (or layer name if selecting by layer).
 - o Right-click for selection menu. Note: In the **Layers** tab, you must right-click a selection to open the menu. In addition, there is only one menu selection available for the **Layers** tab, unlike the other expression views.

o If the Pattern option is available, specify a wildcard pattern.

Some expression categories have a separate **Preview** tab which displays the checks that are selected by the expression.

- d. Click **Apply expression** or **Apply checks** to update the check recipe definition.
 - o **Apply expression** Add or subtract the expression to the recipe definition.
 - **Apply checks** Resolve the expression into a list of checks then add or subtract the checks to the recipe definition.

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

The following figure shows Steps a-d for the "Checks with selected layer only" expression, which selects checks by layer.

Expression categories Expression view Append expression to recipe Expression □ Pattern Checks by rule check comments Layers | Preview (0 checks) .AYERS Select AP_BSL (3 checks) Checks with selected layer only Initial view has no AP_BSP (3 checks) expression Checks including selected layer APi (3 checks) BJTDMY (3 checks) ONNECTIVITY elements selected category Checks with connectivity CB2Wi (3 checks) Checks without connectivity Addition Subtraction Apply expression Apply checks Select to add or (optional) subtract checks ☑ Pattern *VIA* Enter pattern to Layers | Preview (78 checks) filter view VIA7i (5 checks) VIA8_NEW (5 checks) Select Click in checkbox G.T.VIA8_NEW expression to enable/disable G 2 VIA8 NEW G.3.VIA8_NEW elements checks as a group VIA8.S.1 VIA8.S.2 or individually VIA8_OLD (3 checks) G.1.VIA8 Toggle all for addition Right-click menu G.3.VIA8 OLD Jpdate recipe Apply expression Apply checks definition Recipe definition 👔 🗟 🖂 🔁 - Enabled Checks by layer name pattern only: "VIA" G.1.VIA8 OLD Disabled This example used - Mixture of "Apply expression" enabled and disabled

Figure 3-1. Advanced Recipe Editor Controls

- 5. Repeat Step 4 as necessary to add more expressions to the check recipe definition.
- 6. View the check recipe in the Recipe definition area. You can use the two 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.
- 7. 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).

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 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.

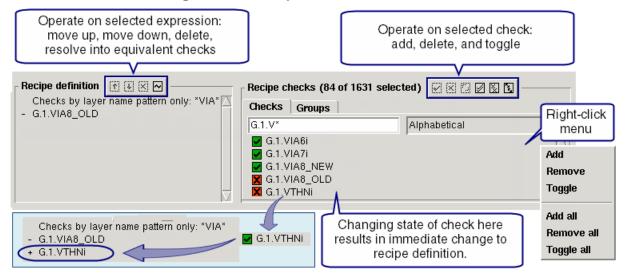


Figure 3-2. Recipe Definition Controls

- 8. (Optional) Click **Export** to save the check recipe to a file. The recipe file can be imported by other users with the **Import** button.
- 9. Click **OK** to save the recipe and exit the dialog box.

Calibre RealTime Recipe Editor Dialog Box

To access: Click **Edit** in the Calibre RealTime Options Dialog Box. The **Edit** button is next to the Check Selection Recipe dropdown list.

The Recipe editor dialog box includes controls to create, copy, save, and edit a check selection recipe for Calibre RealTime.

Description

The Edit Recipe Dialog box has two modes: Basic (Figure 3-3) and Advanced (Figure 3-4). See the tables following the figures for descriptions of the fields and selections in the dialog box.

Use the basic controls for the following tasks:

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

Use the advanced controls for the following tasks:

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

Calibre RealTime - Recipe editor Action New... Import... Export... Copy... Rename... Delete Undo buttons Recipe: new_recipe Check operators Recipe edits Recipe checks (1489 of 1630 selected) 📝 🗵 🗓 📝 🖫 Include: Checks Checks selected in the rules file *VIA Filter and sort All checks ✓ VIA1.R.6.M1 Visible layer checks checks ▼ VIA1.R.6.M2 Select basic VIA1.R.11 Visible layer only checks check recipe VIA1.R.12 None ✓ VIA1.S.1 expressions Select and ▼ VIA1.S.1.1 ✓ VIA1.S.2 unselect checks Checks with connectivity ▼ VIA1.S.3 Checks with density VIA1.S.3.1 Check text IA1.S.3 { @ Space to neighboring VIAx [different net] >= 0.095 EXT VIA1i < VIA1_S_3 ABUT < 90 NOT CONNECTED REGION Check text pane Select basic or advanced Advanced controls

Figure 3-3. Edit Recipe Dialog Box Basic Controls

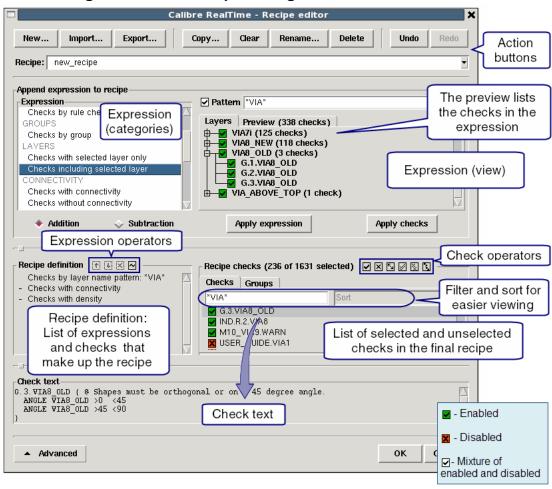


Figure 3-4. Edit Recipe Dialog Box Advanced Controls

Objects

Table 3-2. Edit Recipe Dialog Box Contents — Action Buttons

Control	Type	Description
New	Button	Open a new recipe.
Import	Button	Open a recipe which has been saved to file with the Export button.
		If the configuration file includes a recipe definition with the same name, the tool displays a warning message that the recipe already exists.
Export	Button	Save a recipe to file with the .rcp file extension.
		Note: User recipes are automatically saved to the Session Configuration File in Calibre RealTime Custom; saving a recipe to file provides a means to share recipes with other users and a backup for the check recipe definition.

Table 3-2. Edit Recipe Dialog Box Contents — Action Buttons (cont.)

Control	Type	Description
Сору	Button	Create a copy of the selected recipe.
Clear	Button	Clear all check selections.
Rename	Button	Rename the selected check recipe.
Delete	Button	Delete the selected a check recipe.
Undo	Button	As described.
Redo	Button	As described.

Table 3-3. Edit Recipe Dialog Box Contents — Basic Controls

Control	Туре	Description
Recipe	Dropdown list	Specifies the active recipe.
Include	Radio button	Specify checks to include:
		 Checks selected in the rules file — Include checks selected in the rule file with DRC [Un]Select Check statements. All checks — As described. Visible layer checks — Include all checks which use at
		least one of the layers visible in the window.
		 Visible layer only checks — Include all checks which use only layers visible in the window.
		• None — Unselect all checks.
		A Layer statement specifying a name for an original layer is required in the rule file in order to use the "Visible layer checks," "Visible layer only checks," and "Recently edited layer checks" selections.
		In addition, for the Cadence Virtuoso integration, the layer map file must exist in the technology library directory or be set with an environment variable; see "GDS Layer Map for Calibre RealTime with Cadence Virtuoso" on page 242.
Exclude	Checkbox	Specify checks to exclude:
		Checks with connectivity
		• Checks with density

Table 3-3. Edit Recipe Dialog Box Contents — Basic Controls (cont.)

Control	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.
		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 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. See the recipe definition controls in Figure 3-2 in "Advanced Editing of Check Selection Recipes" on page 113. 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 3-4. Edit Recipe Dialog Box Contents — Advanced Controls

Control	Туре	Description
Recipe	Dropdown list	Specifies the active recipe.
Expression (categories)	Pane	The list of available expression categories for the check selection recipe. See the next table for the definition of each expression.
Expression (view)	Pane	The expression view is to the right of the expression categories pane. It displays the elements you select to make up the expression. Click 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. Depending on the selected expression category, there may be a "Pattern" field. Enter a wildcard pattern in the "Pattern" field to filter the items displayed in the expression view.
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.
		The recipe definition is not modified until you click either Apply expression or Apply checks .

Table 3-4. Edit Recipe Dialog Box Contents — Advanced Controls (cont.)

Control	Туре	Description
Apply checks	Button	Resolve the selected expression into a list of rule checks and update the check recipe.
		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:
		→ Move expression up in the list
		■ — Move expression down in the list
		— Delete the expression
		Resolve recipe into equivalent list of rule checks
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.
		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 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. See the recipe definition controls in Figure 3-2 in "Advanced Editing of Check Selection Recipes" on page 113. 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 3-5. Expression Categories for Check Selection Recipes

Expression Category	Definition and Usage
The expression categories appear in the Expression categories pane in the dialog box, as show in Figure 3-4.	
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 <i>not</i> considered.
	All checks are listed and enabled in the expression view. Click 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.

Table 3-5. Expression Categories for Check Selection Recipes (cont.)

Expression Category	Definition and Usage
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 statements in the rule file.
	Note that if the rule file does not include any DRC [Un]Select Check statements, then all checks are executed; this is the same behavior as for batch Calibre nmDRC.
	Checks selected in the rule file are listed and enabled in the expression view. Click 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 not selected in the rules file	Checks <i>not</i> selected in the rule file. This expression and "Checks selected in the rules file" are the only expressions that obey DRC [Un]Select Check statements in the rule file.
	Checks not selected in the rule file are listed and enabled in the expression view. Click 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.
	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.
	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 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.
	See "Rule Comments" in the Calibre Verification User's Manual.
Checks by group	Choose checks according to rule check groups.

Table 3-5. 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 <i>only</i> the indicated layer.
	Original layers which are the only layer in a rule check are displayed in the expression view. Layers can be expanded to view the rule checks.
	Click 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.
	Original layers which are included in a rule check are displayed in the expression view. Layers which are included in a rule check can be expanded to view the rule checks.
	Click the check box 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.
Recently edited layer checks	Choose checks that include any layer containing shapes modified by the last shape edit operation.
(Synopsys Laker ³ and Calibre DESIGNrev only)	Layers are only accumulated while this expression is present in the active recipe. If "Run DRC on Edit" is disabled, then layers are accumulated from all shape modifications since the last DRC run.
Checks with connectivity	Choose checks that include connectivity.
	All checks with connectivity are listed and enabled in the expression view. Click 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.
	All checks without connectivity are listed and enabled in the expression view. Click 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.

Table 3-5. Expression Categories for Check Selection Recipes (cont.)

Expression Category	Definition and Usage
Checks with density	Choose Density checks.
	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 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.
	All checks without the Density operation are listed and enabled in the expression view. Click 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.
Visible layer only checks ¹	Choose checks which use visible layers only and no invisible layers.
Visible layer checks ¹	Choose all checks which use at least one of the layers visible in the window.
Invisible layer only checks ¹	Choose checks which use invisible layers only and no visible layers.
Invisible layer checks ¹	Choose all checks which use at least one of the invisible layers in the window.
Non-empty checks	Choose checks that produced results. This category is only present after a Calibre RealTime run.
Empty checks	Choose checks that have not produced results. This category is only present after a Calibre RealTime run.

Note

¹ In the Cadence Virtuoso integration, in order to use the check expressions involving visible layers, the layer map file must exist in the technology library directory or be set with an environment variable. For information on the layer map file, see "GDS Layer Map for Calibre RealTime with Cadence Virtuoso" on page 242.



Chapter 4 Calibre RealTime Custom with Pyxis Layout

Calibre RealTime Custom is integrated with Pyxis[™] Layout. Calibre RealTime Custom integrates flat Calibre nmDRC with supported layout design tools so that Calibre nmDRC can be run directly from the design tool.

For an introduction to Calibre RealTime Custom concepts and requirements, see the chapter "Calibre RealTime Custom Introduction" on page 15.

Calibre RealTime Custom is supported with Pyxis Layout for the versions given in the following table:

Table 4-1. Supported Versions for Calibre RealTime Custom and Pyxis Layout

Pyxis Layout Version	Calibre Version
Pyxis Layout v10.0	Calibre 2011.2 or 2011.3
Pyxis Layout v10.1 ¹	Calibre 2011.4
Pyxis Layout v10.2	Calibre 2012.2
Pyxis Layout v10.3	Calibre 2013.2 through Calibre 2014.1
Pyxis Layout v10.4_1	Calibre 2014.2
Pyxis Layout v10.5	Calibre 2014.2
Pyxis Layout v10.5_1	Calibre 2015.2
Pyxis Layout v10.5_2	Calibre 2015.3
Pyxis Layout v10.5_3	Calibre 2015.4
Pyxis Layout v10.5_4	Calibre 2016.1
Pyxis Layout v10.5_5	Calibre 2016.1
Pyxis Layout v10.5_6	Calibre 2016.3
Pyxis Layout v10.5_7	Calibre 2016.4
Pyxis Layout v17.1 and patches	Calibre 2017.1

^{1.} Beginning with the v10.1 release of Pyxis Layout, any mismatch between the Calibre version and the Pyxis Layout version causes an error.

See the *Pyxis Layout User's Manual* for information on the Pyxis Layout tool.

The following	sections	describe the	operation	and use	of Calibre	e RealTime	Custom	with	Pyxis
Layout:									

Running Cambre Real Lime with Pyxis Layout	14
Toolbars and Dialog Boxes in Calibre RealTime	136
Calibre RealTime DRC Hotkeys in Pyxis Layout	149

Running Calibre RealTime with Pyxis Layout

Setting the Layer Map File for Calibre RealTime with Pyxis Layout

The Calibre RealTime client in Pyxis Layout requires the use of a technology layer map to translate Pyxis Layout processes to GDS layers and datatypes.

The procedure you use to specify a layer map depends on whether a technology layer map is included with the design kit.

For this case	Use this procedure	
Technology layer map included in design kit	"Specifying the Technology Layer Map as the Layer Map for Calibre RealTime with Pyxis Layout" on page 128	
Layer map NOT available in design kit	"Manually Creating a Layer Map for Calibre RealTime with Pyxis Layout" on page 128 with the command "AMPLE Function \$set_tech_layermap()" on page 130	

Specifying the Technology Layer Map as the Layer Map for Calibre RealTime with Pyxis Layout

Design kits have a file containing the technology layer map within their technology configuration. The layer map is automatically loaded when the design kit is used and is available for use by the Calibre RealTime client.

Procedure

- 1. From Pyxis Project Manager, open the technology library.
- 2. Select the current project technology library configuration.
- 3. Set the OA GDS layer map file to the technology layer map.
- 4. Click **OK**.
- 5. Open the cell in Pyxis Layout that you want to run Calibre RealTime on.

Manually Creating a Layer Map for Calibre RealTime with Pyxis Layout

If you do not have a design kit or need to create a layer map manually, you can do so with the AMPLE Function \$set_tech_layermap().

You should manually create a layer map only under one of these conditions:

• A design kit containing the OpenAccess to GDS layer map file is not available.

• A custom design kit is used and that kit does not have a project technology library configuration.

Procedure

1. Before running the Calibre RealTime client, create an in-memory layer map with the AMPLE function \$set_tech_layermap().

See "AMPLE Function \$set_tech_layermap()" on page 130 for details.

2. Run Calibre RealTime.

The Calibre RealTime client uses the layer map for the duration of your Pyxis Layout session.

AMPLE Function \$set_tech_layermap()

The AMPLE function \$set_tech_layermap() is used to manually create a layer map when a layer map is not included in the design kit.

Usage

```
$set_tech_layermap('[' layermap, layermap, ... ']')
```

Arguments

• '['layermap, layermap, ...']'

The layermap argument maps a GDS layer number and datatype to a Pyxis Layout layer. The layermap argument has one of the following formats:

```
o '[' gds_layer, gds_datatype, ics_layer_number ']'
```

o '[' gds_layer, gds_datatype, ics_layer_number, ics_layer_purpose ']'

where the brackets ([]) surround the complete layermap list and each layermap entry.

Examples

```
// set the tech layer map with layer and purpose pairs
$set tech layermap(
   [[1,0,"PAD","drawing"],
   [2,0,"MT5","drawing"],
   [3,0,"V45","drawing"],
   [4,0,"MT4","drawing"],
   [5,0,"V34","drawing"],
   [6,0,"MT3","drawing"],
   [7,0,"V23","drawing"],
   [8,0,"MT2","drawing"],
   [9,0,"V12","drawing"],
   [10,0,"MT1","drawing"],
   [11,0,"CO1","drawing"],
   [12,0,"PO1","drawing"],
   [13,0,"NWL","drawing"],
   [14,0,"NPI","drawing"],
   [15,0,"PPI","drawing"],
   [16,0,"OXI","drawing"]]);
```

```
// set the tech layer map with lc layer numbers
$set tech layermap()
   [[6,1,340],
   [6,3,1161],
   [17,1,342],
   [31,1,331],
   [31,2,321],
   [32,1,332],
   [32,2,322],
   [33,1,333],
   [33,2,323],
   [34,1,334],
   [34,2,324],
   [35,1,335],
   [35,2,325],
   [36,1,336],
   [36,2,326],
   [37, 1, 337],
   [37,2,327],
   [38,1,338],
   [38,2,328],
   [39,1,1179],
   [39,2,1180],
   [63,63,3063]]);
```

Invoking Pyxis Layout with Calibre RealTime

Some files and setup actions are required in order to invoke Pyxis Layout with Calibre RealTime.

Prerequisites

- Verify you have Pyxis Layout 10.0 version or later.
- Verify you have Calibre version 2011.2 or 2011.3 if using Pyxis Layout 10.0; or Calibre version 2011.4 if you are using Pyxis Layout 10.1.
- Verify that you have set the layer map file. (See "Setting the Layer Map File for Calibre RealTime with Pyxis Layout" on page 128).
- Set the CALIBRE_HOME environment variable to point to a Calibre software tree.
- Use a Calibre nmDRC specific rule file in order to obtain correct results. Non-DRC rule files may cause incorrect results.

Procedure

1. Invoke Pyxis Layout with the following command:

```
$MGC HOME/bin/ic
```

2. Proceed to "Adding the Calibre RealTime Integrated Toolbar" on page 132 if this is your initial invocation.

Related Topics

License Timeout for Calibre RealTime Custom

Adding the Calibre RealTime Integrated Toolbar

After Pyxis Layout is open, you enable the Calibre RealTime integrated toolbar and place it in the Pyxis Layout display.

Prerequisites

• You have completed "Invoking Pyxis Layout with Calibre RealTime" on page 131.

Procedure

- 1. Add the Calibre RealTime toolbar to your application view with the **Setup > Toolbar > Calibre RealTime** menu. See "Calibre RealTime Integrated Toolbar (Pyxis Layout)" on page 137 for detailed information on each section of the toolbar.
- 2. Place the toolbar on its own row to ensure that all items are visible as shown in Figure 4-1. Once the toolbar is placed, it will appear in subsequent invocations of Pyxis Layout.
- 3. Proceed to "Specifying the Rule File, Check Recipe, and Calibre RealTime Options in Pyxis Layout" on page 133.

Figure 4-1. Calibre RealTime Toolbar Placement



Results

The Calibre RealTime toolbar appears as shown in Figure 4-1.

The Calibre RealTime toolbar components are deactivated under the following conditions:

- If there is no visible or active cell open.
- If the CALIBRE_HOME environment variable is not set.
- If a DRC rule file is not loaded.
- If the cell process is not correctly set.

Related Topics

Invoking Pyxis Layout with Calibre RealTime

Specifying the Rule File, Check Recipe, and Calibre RealTime Options in Pyxis Layout

Specifying the Rule File, Check Recipe, and Calibre RealTime Options in Pyxis Layout

Calibre RealTime in Pyxis Layout uses a standard Calibre nmDRC rule file. Check recipes control which checks in the rule file are executed during the Calibre RealTime run.

Prerequisites

- You have completed "Invoking Pyxis Layout with Calibre RealTime" on page 131 and "Adding the Calibre RealTime Integrated Toolbar" on page 132.
- You have a Calibre nmDRC rule file available.

Procedure

- 1. Click the **Calibre Options** button on the toolbar or enter **Shift-F12**. See Calibre RealTime Options Dialog Box in Pyxis Layout.
- 2. Browse to the Calibre rule file in the "Rule file" text field.
- 3. Select desired options in the dialog box. The defaults are suitable for most runs. See Calibre Options Dialog Box Settings for a definition of each option.
- 4. Select the recipe in the "Check selection recipe" field or select the **Edit Recipe** button to modify an existing recipe or to create a new one. See Editing the Calibre RealTime Check Recipe for instructions on recipe editing.
- 5. Press F12, click the **Run DRC** button , or choose **Tools > Calibre > Run RealTime DRC** to start the run.

Related Topics

Editing the Calibre RealTime Check Recipe

Calibre RealTime Options Dialog Box in Pyxis Layout

Adding the Calibre RealTime Integrated Toolbar

Calibre RealTime Recipe Editor Dialog Box in Pyxis Layout

Editing the Calibre RealTime Check Recipe

You can edit existing check selection recipes or create new user-defined check recipes. You can create new recipes from scratch or you can copy existing user-defined or built-in recipes to create new ones. Custom recipes can be deleted when no longer needed. Built-in recipes may not be deleted.

Prerequisites

- You have completed "Invoking Pyxis Layout with Calibre RealTime" on page 131.
- You have a Calibre nmDRC rule file available.
- You have invoked the Calibre RealTime Options Dialog Box in Pyxis Layout.

Procedure

- 1. Click the ___ button on the Calibre Options dialog box to open the Calibre RealTime Recipe Editor Dialog Box in Pyxis Layout.
- 2. Do one of the following to open a check recipe for editing:
 - Create a new recipe:
 - i. Click the **New Recipe** button on the Edit Recipe toolbar.
 - Copy an existing check recipe:
 - i. Use the Recipe combo box to scroll through all the existing recipes and select one.
 - ii. Click the **Copy Recipe** button on the Edit Recipe toolbar.

See Calibre RealTime Recipe Editor Action Buttons for button descriptions.

- 3. Select checks to include or exclude in your recipe. The basic editing controls are shown in Figure 4-6. See Calibre RealTime Recipe Editor Basic Options for descriptions of the basic controls.
- 4. Check the **Basic/Advanced** button to display advanced controls. The advanced editing controls are shown in Figure 4-7. See Calibre RealTime Recipe Editor Advanced Options for descriptions of the advanced controls.
- 5. Click **OK** in the Edit Recipe dialog.
- 6. Click **OK** in the Calibre Options dialog.

Related Topics

Specifying the Rule File, Check Recipe, and Calibre RealTime Options in Pyxis Layout Calibre RealTime Options Dialog Box in Pyxis Layout

Adding the Calibre RealTime Integrated Toolbar Calibre RealTime Recipe Editor Dialog Box in Pyxis Layout

Toolbars and Dialog Boxes in Calibre RealTime

Calibre RealTime is controlled through the toolbar and several dialog boxes.

Calibre RealTime Integrated Toolbar (Pyxis Layout)	13'
Calibre RealTime Options Dialog Box in Pyxis Layout	141
Calibre RealTime Recipe Editor Dialog Box in Pyxis Layout	144

Calibre RealTime Integrated Toolbar (Pyxis Layout)

The Calibre RealTime integrated toolbar in Pyxis Layout has selections for run control, highlight control, and option settings.

The toolbar is shown in the following figure. See "Invoking Pyxis Layout with Calibre RealTime" on page 131 for information on how to invoke Pyxis Layout with Calibre RealTime enabled.

Figure 4-2. Calibre RealTime Integrated Toolbar



Calibre RealTime Toolbar Run Controls

The run controls are located on the far left-hand side of the Calibre RealTime integrated toolbar.

The following figure shows the run control portion of the toolbar. Each control is described in Table 4-2.

Figure 4-3. Calibre RealTime Toolbar Run Controls

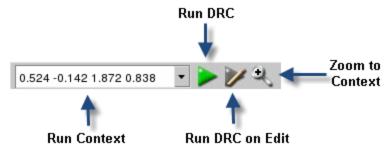


Table 4-2. Calibre RealTime Toolbar Run Control Options

Item	Button	Description
Run Context -0.726 -0.268 2.406 1.584		Lists the context for previous runs. The latest run will be the visible entry. Select a run context in the list to zoom to it.
Run DRC on Edit		Toggles whether or not DRC should be run after every completed edit.

Table 4-2. Calibre RealTime Toolbar Run Control Options (cont.)

Item	Button	Description
Run DRC		Runs a flat DRC in the current window. This button is equivalent to using the F12 key or the Tools > Calibre > Run RealTime DRC menu item.
Zoom to Window	્	Zooms to window for run context shown.

Related Topics

Invoking Pyxis Layout with Calibre RealTime

Calibre RealTime Toolbar Highlight Controls

The highlight controls are located on the middle section of the Calibre RealTime integrated toolbar.

The following figure shows the highlight controls in the toolbar. Each control is described in Table 4-3.

Figure 4-4. Calibre RealTime Toolbar Highlight Controls

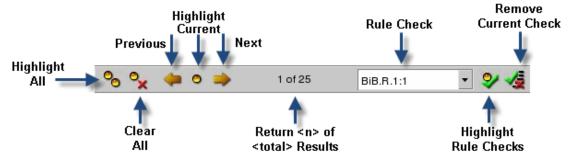


Table 4-3. Calibre RealTime Toolbar Highlight Control Options

Item	Button	Description
Highlight All	00	Highlights all DRC errors for the current context.
Clear All Highlights	°×	Clears all DRC error highlights.
Highlight Previous	4	Highlights the previous result for the current "Rule Check."
Highlight Current	0	Highlights the current result for the current "Rule Check."
Highlight Next	>	Highlights the next result for the current "Rule Check."

Table 4-3. Calibre RealTime Toolbar Highlight Control Options (cont.)

Item	Button	Description
Result <n> of <total> results</total></n>	1 of 44	Identifies which result is being highlighted out of the total results for the current "Rule Check."
Rule Check	•	Indicates the rule check which produced the displayed error. The number of generated errors is appended to the rule check name.
Highlight All Results	>	Highlights all of the results for the current "Rule Check."
Remove Rule Check	√ 2	Remove selected rule from current check recipe. If the current recipe is a built-in recipe, then a new user-defined recipe is created from the built-in name. You can change this default name using the Rename Recipe button. This new recipe becomes the current and active recipe. If the current active recipe is user-defined, then this recipe is modified. The current check name is subtracted from the current recipe, and all results generated by the deleted check are removed from the current cell context.

Related Topics

Invoking Pyxis Layout with Calibre RealTime

Calibre RealTime Toolbar Option Controls

The option controls are located on the right-hand side of the Calibre RealTime integrated toolbar.

The following figure shows the option controls in the toolbar. Each control is described in Table 4-4.

Figure 4-5. Calibre RealTime Toolbar Option Controls

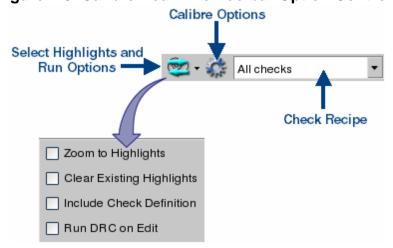


Table 4-4. Calibre RealTime Toolbar Option Controls

Item	Button	Description
Select Highlights	<u>⊚</u> +	Displays a pulldown menu with 4 toggle options. Option states are preserved from one session to the next.
and Run Options		Options include: "Zoom to Highlights," "Clear Existing Highlights," "Include Check Definition in Tooltip", and "Run DRC on Edit."
Calibre Options	*	Opens the Calibre RealTime Options Dialog Box in Pyxis Layout. Use Shift-F12 as a keyboard shortcut.
Check Recipe		Displays a list of recipes. Built-in recipes include:
All checks	•	• All checks
7111 01100110		Visible layer checks
		Visible layer checks, no connectivity
		Visible layer only checks
		 Visible layer only checks, no connectivity
		Recently edited layer checks
		Recently edited layer checks, no connectivity
		Checks selected in the rule file

The toolbar may be relocated only to the TOP or BOTTOM of the standard toolbar locations, but never to the LEFT or RIGHT. The toolbar may not be customized.

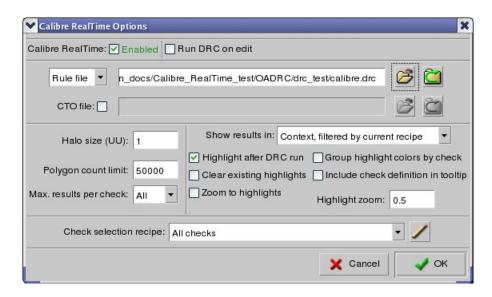
Related Topics

Invoking Pyxis Layout with Calibre RealTime

Calibre RealTime Options Dialog Box in Pyxis Layout

To access: Click the **Calibre Options** button or Shift-F12.

Use the Calibre Options dialog box to specify the Calibre rule file, the check recipe, and other options.



Fields

Table 4-5. Calibre Options Dialog Box Settings

Field	Description	
Calibre RealTime	Toggles the display of the RealTime toolbar. The toolbar is	
	automatically displayed if a DRC run is launched either through	
	Tools > Run RealTime DRC or by pressing the F12 shortcut	
Run DRC on edit	When enabled, start a flat Calibre nmDRC automatically after any design edit.	
Rule file/Runset	Specifies whether to load a rule file or Calibre Interactive runset.	
dropdown list		
Rule file	The Calibre rule file used for the run. Defaults to the rule file selected for previous runs. If this is not available, the rule file associated with the design process is loaded, if available.	
	NOTE:	
	Rule files loaded with \$load_rules() are not recommended. Use a properly constructed rule file designed for Calibre nmDRC rule checking for correct results.	

Table 4-5. Calibre Options Dialog Box Settings (cont.)

Field	Description
Browse	Opens a file browser to select a rule file.
Reload	Reloads the file specified in the Rule file entry. NOTE: If you click this button, then using the Cancel or Reset buttons will
	not retrieve toolbar conditions prior to clicking it.
CTO file	Use a CTO file to specify the highlight color for a rule check. See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90.
Halo size (UU)	Requires a positive floating point number in user-units greater than zero. The default value is 1.
Polygon count limit	Requires a positive, greater than zero integer. The default value is 50000. Value cannot be greater than 3000000.
Max. results per check	Allows you to choose between showing ALL of the results per DRC run or a user-specified quantity. The default value is ALL.
Show results in	This setting controls which results are reported in the integrated toolbar after the run.
	• Whole cell, all checks — Report all results for the cell in the integrated toolbar after each DRC run. Reported results include those from the current run, previous results in the cell that are outside of the current run context, and possibly previous results within the current run context from rules checks not selected by the current recipe.
	• Whole cell, filtered by current recipe — (default) Report results for the whole cell and from the current recipe only in the integrated toolbar after each DRC run. Reported results include those from the current run plus previous results in the cell that are outside of the current run context.
	• Context, filtered by current recipe — Report only results within the current run context and from the current check recipe in the integrated toolbar after each DRC run.
	For each case, all results in the current window and belonging to the current check selection recipe are deleted before the run.
	The settings "Whole cell, filtered by current recipe" and "Context, filtered by current recipe" should be used with caution, since they filter out results from previous runs. If these settings are used, it is recommended that a later run be performed with "Whole cell, all checks" selected to make sure that all results are reported.
Highlight after DRC run	Highlights results in the viewer after every DRC run. The default setting is ON.

Table 4-5. Calibre Options Dialog Box Settings (cont.)

Field	Description
Group highlight colors by check	Assign the highlight color for results according to the rule check. The Calibre RealTime toolbar must be visible and enabled for this feature to work.
Clear existing highlights	Clear existing highlights before each new highlight action.
	Highlights are cleared before each run, and when using the buttons to step though highlights.
Include check definition in tool-tip	When enabled, the complete rule check definition is included in the result tool tip; otherwise, only the check text comments are shown.
	Note: You must run Calibre RealTime after changing this setting for the change to take effect.
Zoom to highlights	Specifies whether to zoom to the result when highlighting.
Highlight zoom	Specifies the zoom ratio. Expects a positive, greater than zero, floating point number. The default value is 0.5.
Check selection recipe	Specifies the check recipe used for the Calibre run. See "Check Recipes in Calibre RealTime Custom" on page 105.
Edit recipe	Displays the Calibre RealTime Recipe Editor Dialog Box in Pyxis Layout.
Button row	Cancel: Closes the dialog without any actions.
	OK: Applies all settings according to the values specified, and then closes the dialog.

Related Topics

Specifying the Rule File, Check Recipe, and Calibre RealTime Options in Pyxis Layout

Calibre RealTime Recipe Editor Dialog Box in Pyxis Layout

To access: Click the / button in the Calibre Options dialog.

The Calibre RealTime Recipe Editor edits a check selection recipe. There are two different views to toggle between in the dialog box: basic editing and advanced editing. The basic editing view enables you to perform very basic edits to RealTime DRC run recipes. The advanced editing view gives you precise control over the check types and which specific checks to include or exclude for any particular DRC run.

Description

Use the Calibre RealTime Recipe Editor to edit RealTime DRC run recipes.

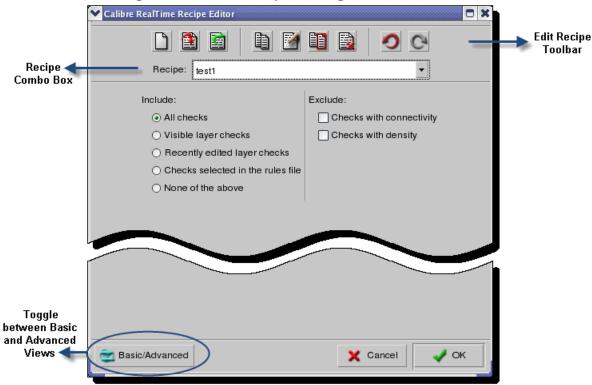


Figure 4-6. Edit Recipe Dialog Box Basic Mode

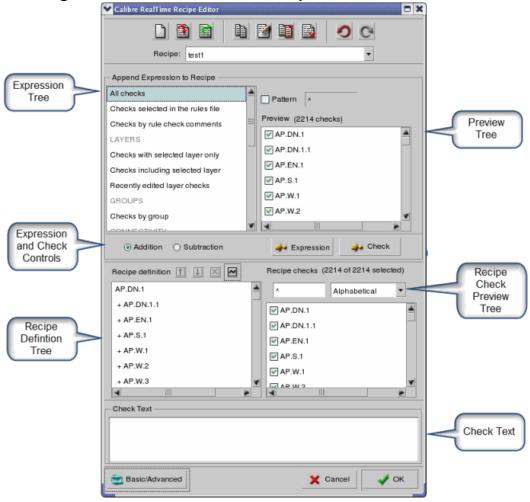


Figure 4-7. Calibre RealTime Recipe Editor Advance Mode

Objects

Table 4-6. Calibre RealTime Recipe Editor Action Buttons

Item	Button	Description
New Recipe		Displays the New Recipe Name dialog box. The new recipe is added to the list of recipes if it is not a duplicate, and becomes the active user-defined recipe.
Import Recipe		Opens an existing recipe from disk. Once imported, the recipe will be added to the user-defined list of recipes and will become the current recipe.

Table 4-6. Calibre RealTime Recipe Editor Action Buttons (cont.)

Item	Button	Description
Export Recipe		Displays a file browser to select the export location. Only user-defined recipes may be exported.
		• If you click OK without providing a file name, then the current recipe name is used and appended with ".rcp". If any blanks or commas exist in the name, they will be replaced by underscores.
		• If you click OK with a file name that is different than the recipe name, then that name will be used and appended with ".rcp".
Copy Recipe		Creates a copy of the current recipe. The copy is named using the current recipe plus an underscore and a numeric extension. For example, "my_recipe" becomes "my_recipe_1". If any blanks or commas exist in the name, they will be replaced by underscores.
		The new name is added to the list of recipes as a user-defined recipe, provided it is not a duplicate name. The copy becomes the current active recipe.
Clear Recipe		Clears the current user-defined recipe shown in the Recipe Combo box. Only user-defined recipes may be cleared.
Rename Recipe		Displays the Rename Recipe dialog box.
Delete Recipe		Deletes the recipe displayed in the Recipe Combo. You can only delete user-defined recipes.
Undo	•	Restores the previous recipe editing operation during the current session.
Redo	C	Restores the recipe editing operation performed during the current session by clicking the Undo button.

Table 4-7. Calibre RealTime Recipe Editor Basic Options

Item E	Button	Description
Recipe Combo		Presents a list of user-defined and built-in recipes. User-defined recipes, when they exist, display at the top of the combo list. The built-in recipes are fixed as follows:
		All checks
		Checks selected in the rule file
		Recently edited layer checks
		Recently edited layer checks, no connectivity
		Visible layer checks
		Visible layer checks, no connectivity
		Visible layer only checks
		Visible layer only checks, no connectivity
		A Layer statement specifying a name for an original layer is required in the rule file in order to use the layer-based selections.
Include		Specify checks to include:
		• All checks — As described.
		• Visible layer checks — Include all checks which use at least one of the layers visible in the window.
		• Recently edited layer checks — As described.
		• Checks selected in the rules file — (default) Include checks selected in the rule file with DRC [Un]Select Check statements.
		• None of the above — As described.
		A Layer statement specifying a name for an original layer is required in the rule file in order to use the "Visible layer checks" and "Recently edited layer checks" selections.
Exclude		Specify checks to exclude:
		Checks with connectivity
		Checks with density

Table 4-8. Calibre RealTime Recipe Editor Advanced Options

Item	Button	Description
Append Expression to Recipe		Displays the pattern option and preview entries for the recipe selected in the Expression Tree field.
Pattern option checkbox		Activates the pattern entry field as well as a fresh preview scan using the current entry in the Pattern entry field as a scanning filter.

Table 4-8. Calibre RealTime Recipe Editor Advanced Options (cont.)

Pattern entry field		Displays patterns to filter in the Preview tree. The default value is "^". The "^" filter searches for matches where the patterns match from the beginning. The "\$" filter searches for matches where the patterns match in the end. A blank pattern filter searches for matches within the pattern at any point.
Preview tree		Displays a flat view where each item is the name of a check from the rule file. Each item may be included or excluded by using the check box next to the item. All items are checked, by default.
Addition choice		Adds selected expressions or checks to the Recipe definition tree. The "+" (plus sign) indicates that the current recipe definition will include those expressions or checks. Swaps the check state of any checks in the Preview trees except the tabbed Pattern tree.
Subtraction choice		Adds selected expressions to the Recipe definition tree. The "-" (minus sign) indicates that the current recipe definition will exclude those definitions. Swaps the check state of any checks in the Preview trees except the tabbed Pattern tree.
Expression bu	itton	Adds the selected item from the Expression tree to the Recipe definition tree with either a + sign or a - sign.
Check button		Adds the selected items from the visible Layer or Preview tree with either a + sign or a - sign.
Recipe definition		Displays a flat tree containing a list of all short-hand composition check definitions as well as individual checks. Checks with the + sign are included in the recipe definition whereas checks with the - sign are excluded.
Move Up	1	Moves the selected items in the Recipe definition tree up one position.
Move Down	Ŧ	Moves the selected items in the Recipe definition tree down one position.
Delete	×	Deletes all of the selected items in the Recipe definition tree.
Resolve	~	Resolves the contents of the Recipe definition tree into the individual rule file checks. Removes short-hand notations.
Pattern field		Re-displays the Recipe checks tree contents using the regular expression pattern typed into this field. The default pattern is "^".
Sorting choices combo box		Offers four choices to control the check rules display order in the Recipe checks tree.
Recipe checks tree		Displays a flat tree containing a list of all the selected (checked) and unselected (unchecked) rules from the current rules file.

Calibre RealTime DRC Hotkeys in Pyxis Layout

Keyboard shortcuts, or hotkeys, are available for many functions in Calibre RealTime for Pyxis Layout.

The following table shows the hotkeys available in Calibre RealTime. F12 and Shift+F12 are always active within Calibre RealTime. The remaining keys can be activated by turning on an option in the Pyxis Layout preferences. The default option is off. To activate these keys, use **Setup > Preferences > Behavior > Enable Calibre RealTime hotkeys**.

Table 4-9. Calibre RealTime DRC Hotkeys in Pyxis Layout

Key	Behavior		
Default Keyboard Sl	Default Keyboard Shortcuts		
F12	Run DRC		
Shift+F12	Show Calibre RealTime Options Dialog Box in Pyxis Layout		
Optional Keyboard	Shortcuts, Off by Default		
Ctrl+Shift+J	Toggle DRC on Edit		
Ctrl+Shift+Q	Switch to next recipe		
Ctrl+Shift+W	Switch to previous recipe		
Ctrl+Shift+E	Switch to last used recipe		
Ctrl+Shift+<	Highlight previous result		
Ctrl+Shift+C	Highlight current result		
Ctrl+Shift+>	Highlight next result		
Ctrl+Shift+B	Switch to previous check and highlight first result		
Ctrl+Shift+N	Switch to next check and highlight first result		
Ctrl+Shift+A	Highlight all in check		
Ctrl+Shift+D	Remove check		

Chapter 5 Calibre RealTime Custom with Synopsys Laker³

Calibre RealTime Custom is integrated to Synopsys Laker³. Calibre RealTime Custom integrates flat Calibre nmDRC with supported layout design tools so that Calibre nmDRC can be run directly from the design tool.

The following topics may be useful in addition to the topics in this chapter:

- "Calibre RealTime Custom Introduction" on page 15 for an overview and system requirements
- Calibre RealTime Integration to Synopsys Laker Quick Reference
- "Calibre RealTime Environment Variables" on page 333

Supported Versions of Synopsys Laker3 for Calibre RealTime	151
Running Calibre RealTime with Synopsys Laker	153
Toolbars and Dialog Boxes in Calibre RealTime (Laker)	161
Database Read Setup for Calibre RealTime with Synopsys Laker	172
Calibre RealTime Setup with Tcl Commands in Synopsys Laker	175
Enabling the Calibre RealTime Integration in Synopsys Laker	176
Calibre RealTime Configuration File (Laker)	177
Keyboard Shortcuts (Hotkeys) in Calibre RealTime for Synopsys Laker	179

Supported Versions of Synopsys Laker³ for Calibre RealTime

The current release of Calibre RealTime is supported for certain version and platform combinations for Synopsys Laker.

Table 5-1. Supported Versions of Synopsys Laker³ with Calibre RealTime

Synopsys Laker version	CALIBRE_HOME Tree
Synopsys Laker ³ version 2014.9.SP-1	AOI (Linux ^{®1} RHEL 6, RHEL 7, SLES 11sp2, and SLES 11sp3)
	IXL (Linux RHEL 5 and SLES 11sp1)

Table 5-1. Supported Versions of Synopsys Laker³ with Calibre RealTime

Synopsys Laker version	CALIBRE_HOME Tree
Synopsys Laker ³ version L-2016.09	AOI (Linux RHEL 6, RHEL 7, SLES 11sp2, and SLES 11sp3)
	IXL (Linux RHEL 5 and SLES 11sp1)

^{1.} Linux[®] is a registered trademark of Linus Torvalds in the U.S. and other countries.

Previous versions of Synopsys Laker may not have all fixes required for this version of Calibre RealTime. Operation with later versions of Synopsys Laker³ has not been verified. See "Supported Operating Systems and Hardware" in the *Calibre Administrator's Guide* for more information on the supported platforms and corresponding CALIBRE_HOME tree.

Running Calibre RealTime with Synopsys Laker

Invoking Synopsys Laker with Calibre RealTime

Invoking Synopsys Laker with Calibre RealTime adds the Calibre RealTime integrated toolbar to the design window.

Prerequisites

- "Enabling the Calibre RealTime Integration in Synopsys Laker" on page 176 completed.
- System and license requirements described in "Calibre RealTime Custom Requirements" on page 15 are satisfied.
- (Recommended) "Database Read Setup for Calibre RealTime with Synopsys Laker" on page 172 completed.

Procedure

1. Start Synopsys Laker with the following command:

```
$LAKER_HOME/bin/laker <options>
```

A view of the design window in Synopsys Laker³ is shown in the following figure. The Calibre RealTime Integrated Toolbar (Laker) is loaded automatically in the Synopsys Laker window.

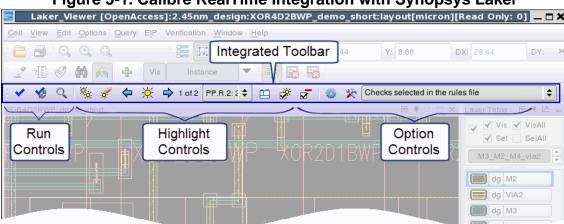


Figure 5-1. Calibre RealTime Integration with Synopsys Laker

2. Proceed to "Specifying and Using Multiple Run Configurations in Calibre RealTime (Laker)" on page 154.

Related Topics

Calibre RealTime Configuration File (Laker)

Calibre RealTime Environment Variables

License Timeout for Calibre RealTime Custom

Specifying and Using Multiple Run Configurations in Calibre RealTime (Laker)

A run configuration in Calibre RealTime is the set of options and input files for a run. You can define multiple run configurations with different options (such as the rule file, check recipe and other options), then quickly switch between different run configurations. You can run with one configuration (single mode) or run multiple configurations in sequence (serial mode).

Nota

Run configurations were introduced in the 2016.2 release. The first time you open Calibre RealTime in a release after 2016.1 the "Run mode" is set to Single and a run configuration named "Default configuration" is created based on the settings in your Calibre RealTime session configuration file (.realtime.cfg). The configuration file is updated to the 2016.2 syntax.

If you plan on using only one run configuration, no further action is needed.

Multiple run configurations can be used in the following modes:

- **Single** Run only the currently selected configuration when a run is started.
- **Serial** Run each configuration selected in the Configuration Run Control dialog box in order.

Prerequisites

• "Invoking Synopsys Laker with Calibre RealTime" on page 153

Procedure

- 1. Choose **Calibre > RealTime DRC > Options** to open the Calibre RealTime Options Dialog Box (Laker), or click the **Options** button (on the integrated toolbar.
- 2. If using multiple run configurations, define as many configuration names as needed:
 - a. Click the **Edit** button next to the configuration name to open the Configuration Run Control dialog box.
 - b. Click the "+" button to add a configuration and specify the name. Repeat this step as needed.
 - c. If using the Serial run mode, set the run order using the up and down arrows. Use the checkbox to include or remove a configuration from the execution list.
 - d. Click **OK** to close the Configuration Run Control dialog box.
- 3. Define the settings for each run configuration:
 - a. Select the configuration in the Configuration dropdown list.
 - b. Specify the settings as described in "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Laker)" on page 155.
- 4. Select the "Run mode":
 - **Single** Run only the currently selected configuration when a run is started.
 - **Serial** Run each configuration selected in the Configuration Run Control dialog box in order.
- 5. Proceed to "Running Calibre RealTime in Window Mode with Synopsys Laker" on page 157 or "Running Calibre RealTime in Run on Edit Mode (Laker)" on page 159.

Results are displayed separately for each run configuration—select the configuration to view with the toolbar selection:



Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Laker)

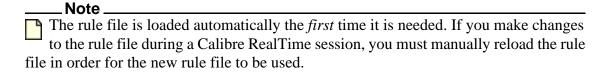
Calibre RealTime uses a standard Calibre nmDRC rule file. Check recipes control which rule checks are executed during the Calibre RealTime run.

Prerequisites

- You have completed "Invoking Synopsys Laker with Calibre RealTime" on page 153.
- You have completed "Specifying and Using Multiple Run Configurations in Calibre RealTime (Laker)" on page 154
- You have a Calibre nmDRC rule file available.

Procedure

- 1. Click the **Options** button () on the integrated toolbar to open the Calibre RealTime Options dialog box.
- 2. Select the configuration in the "Configuration" dropdown list.
- 3. Enter the Calibre rule file in the "Rule file" text field.
- 4. Click **Reload** to load the rule file.



5. (Optional) Select options in the dialog box; the defaults are suitable for most runs. See "Calibre RealTime Options Dialog Box" on page 31 for a definition of each option.

If you want the Calibre RealTime Results window to open automatically after a run, enable the option "Launch RealTime results viewer automatically."

If you want to save a summary file of the run and the DRC results as an ASCII results database (RDB), select the **Output files** tab and check the corresponding options.

- 6. Select the check recipe as follows:
 - Checks selected in the rules file (default) This built-in recipe uses the Calibre rule file just as a batch run of Calibre does.
 - Other built-in check recipes You can select a recipe that only runs checks on certain layers, excludes certain types of checks, or runs all checks in the rule file.

Because these built-in recipes do *not* obey DRC [Un]Select Check statements in the rule file, the recipes may execute checks in the rule file that would not be executed during a batch Calibre run. The recipe "Checks selected in the rules file" is the only built-in recipe that obeys DRC [Un]Select Check statements. See "Built-In Check Recipes" on page 107.

• User recipes — You can select a previously defined custom check recipe. See "Basic Editing of Check Selection Recipes" on page 111.

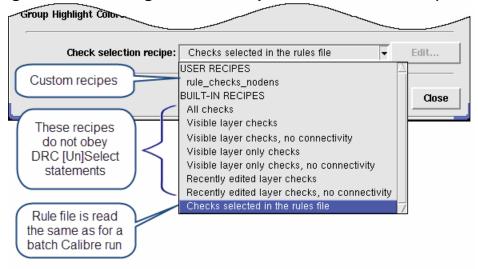


Figure 5-2. Selecting a Check Recipe in Calibre RealTime (Laker)

7. Proceed to "Running Calibre RealTime in Window Mode with Synopsys Laker" on page 157 or "Running Calibre RealTime in Run on Edit Mode (Laker)" on page 159.

You can change environment variables with the commands given in the section "Environment Variable Control in Calibre RealTime" on page 53. This is useful if you want to change environment variables used in the rule file.

You can set Calibre RealTime options using Tcl commands in the Synopsys Laker environment; see "Option Control in the Design Environment for Calibre RealTime" on page 58.

Related Topics

Using Calibre Interactive Settings in a Calibre RealTime Custom Run

Check Selection Recipes in Calibre RealTime Custom

Database Read Setup for Calibre RealTime with Synopsys Laker

Running Calibre RealTime in Window Mode with Synopsys Laker

In window mode, Calibre RealTime runs checks on the geometries visible in the Laker window, plus a halo around the region. The specified check recipe selects the checks that are executed. All Calibre RealTime nmDRC runs are performed in flat mode.

Prerequisites

 You have completed "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Laker)" on page 155

Procedure

- 1. Select a run configuration in the "Configuration" dropdown list, or, if running in Serial mode, make sure the "Run Mode" is set to Serial in the Calibre RealTime Options dialog box. See "Specifying and Using Multiple Run Configurations in Calibre RealTime (Laker)" on page 154.
- 2. In Synopsys Laker³, disable "Run DRC on Edit" by toggling the toolbar button to the OFF state (). You can also control this setting from the Calibre RealTime Options dialog box. (This step is optional—you can still run in window mode while "Run DRC on Edit" is enabled.)
- 3. Zoom to the region you want to run DRC on and make sure all the geometries you want to run checks on are visible. A region of 25 um x 25 um or smaller is recommended for best performance.
- 4. Start the run by clicking \checkmark (Run DRC) in the toolbar, or use the **F12** keyboard shortcut.

This runs flat Calibre nmDRC on the geometries visible in the Laker window plus the specified halo region around the area.

If running in Serial mode, a DRC run is executed for each of the configurations selected in the "Configuration Run Control" dialog box, in the order specified.

Results

The results of a Calibre RealTime run are shown in the following figure. You can use the highlight controls in the integrated toolbar to step through results. Result hover text is enabled with "Include Check Definition in Tool Tip" in the Calibre RealTime Options Dialog Box. If you are using multiple run configurations, the results are displayed separately for each configuration—select the configuration to view in the toolbar dropdown box.

You can also click the toolbar icon to open the Calibre RealTime Results window to view results in a separate window; see "Viewing Results in the Calibre RealTime Results Window" on page 83.

Rule check comments are shown in the Calibre RealTime window; the check definition is also included if "Include Check Definition in Tool Tip" was enabled at the time of the run. You can turn off display of the Calibre RealTime window by disabling the setting "Show check description window" in the Calibre RealTime Options dialog box.

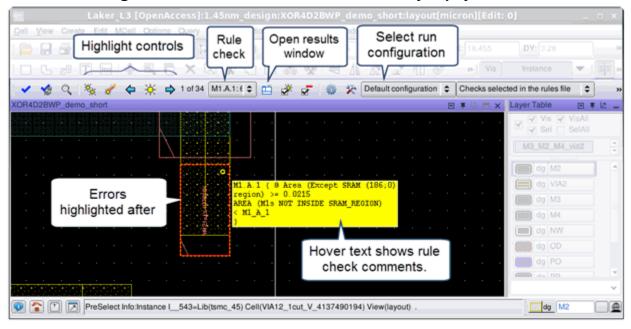


Figure 5-3. Calibre RealTime Results in Synopsys Laker

Related Topics

Calibre RealTime Integrated Toolbar (Laker)

Running Calibre RealTime in Run on Edit Mode (Laker)

In Run on Edit mode, Calibre nmDRC runs automatically after any design edit. Calibre is run on the region of the design edit plus a halo region around the area; the specified check recipe selects the checks that are executed. All Calibre RealTime nmDRC runs are performed in flat mode.

Prerequisites

• Completion of "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Laker)" on page 155.

Depending on your design, you may want to the select one of the built-in recipes that limit the checks that are run, such as "Visible layer checks, no connectivity"; see "Built-In Check Recipes" on page 107 for details.

• Calibre RealTime for Synopsys Laker³ version.

Procedure

1. Select a run configuration in the "Configuration" dropdown list, or, if running in Serial mode, make sure the "Run Mode" is set to Serial in the Calibre RealTime Options dialog

box. See "Specifying and Using Multiple Run Configurations in Calibre RealTime (Laker)" on page 154.

- 2. Enable "Run DRC on Edit" by toggling the toolbar button to the ON state (). You can also control this setting from the Calibre RealTime Options dialog box.
- 3. Make any design edit to trigger a flat Calibre nmDRC run.

Calibre nmDRC runs on the region of the design edit plus the specified halo around the region.

If running in Serial mode, a DRC run is executed for each of the configurations selected in the "Configuration Run Control" dialog box, in the order specified.

Results

Results are similar to those shown in "Running Calibre RealTime in Window Mode with Synopsys Laker" on page 157. You can use the highlight controls in the integrated toolbar to step through results. You can also click the toolbar icon to open the Calibre RealTime Results window to view results in a separate window; see "Viewing Results in the Calibre RealTime Results Window" on page 83. If you are using multiple run configurations, the results are displayed separately for each configuration—select the configuration to view in the toolbar dropdown box.

Related Topics

Inconsistent Results from Run DRC on Edit Mode with Small Halo Size Calibre RealTime Integrated Toolbar (Laker)

Edit-in-Place in Calibre RealTime with Synopsys Laker

There are several points to keep in mind when you use Edit-In-Place (EIP) to fix errors found with Calibre RealTime.

- When you descend to a lower level cell in EIP mode, Calibre RealTime retains the current context—you still see the error highlights from the upper level and the same listing of results in the Calibre RealTime toolbar.
- After you fix the error in the lower level cell, Calibre RealTime runs automatically for the region of the edit if you have "Run DRC on Edit" enabled, or you can click the Run DRC button (✓) to run in the window. The run takes place in the current context, which includes the level being edited up to the starting viewing level. In some cases a change in a lower level cell can cause errors at the upper levels.

Note: Cell instantiations outside the context of the current window are not checked.

Toolbars and Dialog Boxes in Calibre RealTime (Laker)

The Calibre RealTime integrated toolbar and the Calibre RealTime Options dialog box provide the controls and settings you need to run Calibre RealTime.

Calibre RealTime Integrated Toolbar (Laker)	16 1
Calibre RealTime Options Dialog Box (Laker)	16

Calibre RealTime Integrated Toolbar (Laker)

The Calibre RealTime integrated toolbar includes controls to start a DRC run, highlight results, and open the Calibre RealTime Options dialog box. The toolbar also includes option settings for run control and highlighting.

The Calibre RealTime toolbar is shown in the following figure.



The run controls, highlight controls, and option controls are described in the following subsections.

Note -

For 2015.1 and earlier releases, the toolbar included a dropdown list of the ten most recent run contexts. Beginning with the 2015.2 release the run context is no longer displayed by default. Define the environment variable

MGC_CALIBRE_REALTIME_SHOW_CONTEXT_IN_TOOLBAR to restore the run context display.

Run Controls (Laker)

The run controls in the Calibre RealTime integrated toolbar start a window run, zoom to the window for the most recent DRC run, and set the Run DRC on Edit option.

The run control portion of the Calibre RealTime toolbar is shown in the following figure. The options are described in Table 5-2.

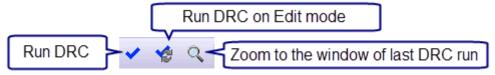


Table 5-2. Run Control Toolbar (Laker)

Toolbar item	Description
Run DRC on Edit	— "Run DRC on Edit" is enabled; Calibre nmDRC runs automatically after any design edit. (Only available with Synopsys Laker³ version.) — "Run DRC on Edit" is disabled. (Also called Run DRC in Window mode.) Calibre runs with the currently selected Run Configuration and configuration run mode (Serial or Single).
✓ (Run DRC)	Runs flat DRC on the geometries visible in the current window, plus a halo region around the area.
Q	Zoom to the window of the most recent DRC run.

Highlight Controls (Laker)

The highlight control section the Calibre RealTime integrated toolbar includes highlight controls and displays the rule check for the current result.

The highlight controls are shown in the following figure and described in Table 5-3.



Table 5-3. Highlight Control Toolbar (Laker)

Toolbar item	Description
***	Highlight all.
	Clear all highlights.
⇔ ★ →	Highlight previous, current, or next result for the rule check.
	Also see "Behavior of "Group highlight colors by check" in "Calibre RealTime Options Dialog Box" on page 31.

Table 5-3. Highlight Control Toolbar (Laker) (cont.)

Toolbar item	Description
1 of 34	Result <i>n</i> of <i>total</i> results. Also see the setting "Show results in" in the Calibre RealTime Options Dialog Box; this setting determines whether all results for the cell or only results from the most recent run are reported.
Rule check	Indicates the rule check which produced the displayed error when using the highlight previous, current, and next buttons. Use the dropdown arrow to select a different rule check.
ૐ	Highlight all results from the selected rule check.

Option Controls (Laker)

The option controls in the Calibre RealTime integrated toolbar select the check recipe and open the Calibre RealTime Options dialog box.

The option controls are shown in the following figure and described in Table 5-4.



Toolbar item	Description
	Open the Calibre RealTime Results window. See "Viewing Results in the Calibre RealTime Results Window" on page 83.
✓	Remove the displayed rule check from the current check recipe. All highlights are cleared, regardless of the setting of "Clear Existing Highlights" in the options list. If the recipe is a built-in recipe, you are prompted to
	save the new check recipe to a new filename; if the recipe is a user recipe, the updated recipe is saved automatically.
	Also see "Behavior of "Group highlight colors by check" in "Calibre RealTime Options Dialog Box" on page 31.
	Opens the Calibre RealTime Options Dialog Box.

Table 5-4. Option Control Toolbar (Laker) (cont.)

Toolbar item	Description
*	Opens the Customization GUI. This button is only available if a Calibre Interactive runset is loaded and includes a customization file. See "Using the Customization GUI with Calibre RealTime" on page 88.
Run Configuration	Selects the run configuration, both for a run in "Single" mode and for viewing results; see "Specifying and Using Multiple Run Configurations in Calibre RealTime (Laker)" on page 154.
Check recipe	Selects the check recipe from a dropdown list; the check recipe specifies the rule checks executed for all Calibre RealTime runs. See "Check Selection Recipes in Calibre RealTime Custom" on page 105 for more information.

Calibre RealTime Options Dialog Box (Laker)

To access: Click the 👸 button in the toolbar. You can also choose Verify > Calibre > **RealTime > Options** or **DRC > Options** from the main menu, depending on the version of Synopsys Laker.

The Calibre RealTime Options dialog box specifies the Calibre rule file, the check recipe, and other options. The settings in the Calibre RealTime Options dialog box are saved in the Calibre RealTime configuration file.

Description

You can also set Calibre RealTime options using Tcl commands in the Synopsys Laker environment; see "Option Control in the Design Environment for Calibre RealTime" on page 58.

Calibre RealTime v2020.3- Options Calibre RealTime: Frabled ☐ Run DRC on edit ■ Launch RealTime results viewer automatically ☐ Highlight all configuration results Configuration: Default configuration Run mode: Single DRC Output files Rule file: ▼ Reload CTO file: Layer map file: Use layer map file exclusively Object map file: Halo size (UU): 1 Show results in: Context, filtered by current recipe Polygon count limit: 50000 ☐ Highlight after DRC run Group highlight colors by check Max results per check: All Clear existing highlights Include check definition in tooltip Zoom to highlights Highlight zoom: 0.5 Text size: Show check names Check selection recipe: Checks selected in the rules file Edit. oĸ Cancel Help

Table 5-5. Tabs in the Calibre RealTime Options Dialog Box (Laker)

Tab	Description
DRC	Specify options for the Calibre DRC run. See the table in the next section.
Output files	Specify to save a summary file and/or to save the DRC results to an ASCII results database (RDB). If a specified directory does not exist, the tool prompts you to confirm that the directory should be created.

Objects

Table 5-6. Calibre RealTime Options Dialog Box Contents (Laker)

Control	Description
Calibre RealTime	Toggles display of the Calibre RealTime toolbar.
(Enabled/Enable)	Also see the environment variable MGC_REALTIME_DISABLE in "Calibre RealTime Environment Variables" on page 333.
Run DRC on edit	When enabled, start a flat Calibre nmDRC automatically after any design edit.
	(Only available with Synopsys Laker ³ version)
Launch RealTime results viewer automatically	Specifies whether to open the Calibre RealTime Results Window automatically after a run. See "Viewing Results in the Calibre RealTime Results Window" on page 83.
Highlight all configuration results	Specifies whether to automatically highlight DRC results from all configurations after a DRC run. When this option is enabled:
	• Highlights are preserved when switching configurations.
	• When highlighting, the "Clear existing highlights" option is ignored and existing highlights are not cleared.
	 Results from all configurations are highlighted in both single and serial runs, and the "Highlight after DRC" option is ignored.
	Note: When this option is enabled, you can only crosshighlight results from the layout editor to the Results window if they are from the current configuration. You can view the check description of any result regardless of its corresponding configuration.

Table 5-6. Calibre RealTime Options Dialog Box Contents (Laker) (cont.)

Control	Description
Configuration	Specifies the configuration name. To add or delete a configuration, click Edit to open the Configuration Run Control dialog box.
	A configuration is the set of options and filenames below the configuration name.
Run Mode	 Single — Run with the currently selected configuration. Serial — Run each selected configuration in series, in the order specified in the Configuration Run Control dialog box.
Rule file/Runset dropdown list	Selects whether to load a rule file or a Calibre Interactive runset.
Rule file	Specifies the Calibre rule file used for the run.
Runset	Specifies a Calibre Interactive runset to use for the run. See "Using Calibre Interactive Settings in a Calibre RealTime Custom Run" on page 87 for details.
CTO file	The CTO file is not supported with Synopsys Laker, however, you can set highlight colors using DRC rule check comments in the rule file; see "DRC Rule Check Comments for Calibre RealTime Custom" on page 90.
Layer map file	Specifies a custom layer map file.
	Dropdown list options:
	• Merge layer map with existing mapping — (default) Merge layer map data from the specified file with the layer map derived from the technology file. Layer map entries in the layer map file override the corresponding layer map entry based on the technology file; all other layer map entries based on the technology file remain valid.
	• Use layer map file exclusively— Use only layer map data from the specified file. Layer map data derived from the technology file is <i>not</i> used.
	See "Specifying the Layer Map for Calibre RealTime with Synopsys Laker" on page 172.
Object map file	Specifies an object map file. See "Object Map File Format in Calibre RealTime" on page 40.
(Browse)	Open a file browser to select the rule file, CTO file, or layer map file.

Table 5-6. Calibre RealTime Options Dialog Box Contents (Laker) (cont.)

Control	Description
Reload	Reload the rule file or runset.
	You must click Reload for changes in the rule file or runset to take effect. Changes in the layer map, object map, and CTO files cause an automatic reload prior to the DRC run.
Halo size (UU)	The halo size in user units. See "Halos in Calibre RealTime Custom" on page 24 and "Halo" in "Terms and Definitions in Calibre RealTime Custom" on page 19 for more information on halos.
Polygon count limit	Sets the maximum number of polygons that are passed to Calibre. The limit should be set low enough that the Calibre server returns control to the design tool in a short amount of time.
	The maximum number of geometries Calibre RealTime will process is 3,000,000.
Max results per check	Specifies the maximum number of results returned per rule check; this setting overrides the setting in the rule file.
	• All — report all results per check.
	• number — report only <i>number</i> results per check.

Table 5-6. Calibre RealTime Options Dialog Box Contents (Laker) (cont.)

Control	Description
Show results in	This setting controls which results are reported in the integrated toolbar after the run.
	• Whole cell, all checks — Report all results for the cell in the integrated toolbar after each DRC run. Reported results include those from the current run, previous results in the cell that are outside of the current run context, and possibly previous results within the current run context from rules checks not selected by the current recipe.
	• Whole cell, filtered by current recipe — Report results for the whole cell and from the current recipe only in the integrated toolbar after each DRC run. Reported results include those from the current run plus previous results in the cell that are outside of the current run context.
	• Context, filtered by current recipe — (default) Report only results within the current run context and from the current check recipe in the integrated toolbar after each DRC run.
	For each case, all results in the current window and belonging to the current check selection recipe are deleted before the run.
	The settings "Whole cell, filtered by current recipe" and "Context, filtered by current recipe" should be used with caution, since they filter out results from previous runs. If these settings are used, it is recommended that a later run be performed with "Whole cell, all checks" selected to make sure that all results are reported.
Highlight after DRC run	Highlight results in the viewer after the DRC run.
Group highlight colors by check	Assign the highlight color for results according to the rule check.
	The assigned highlight color for a rule check is not fixed—rather, the highlight colors are assigned to checks in a repeated cycle, starting with the color red. See "Usage Notes" for details.
Clear existing highlights	Clear existing highlights before each new highlight action.
	Highlights are cleared before each run, and when using the
	Also see Behavior of "Group highlight colors by check" in the Usage Notes section

Table 5-6. Calibre RealTime Options Dialog Box Contents (Laker) (cont.)

Control	Description
Include check definition in tool-tip	When enabled, the complete rule check definition is included in the result tool tip; otherwise, only the check text comments are shown.
	Note: You must run Calibre RealTime after changing this setting for the change to take effect.
Zoom to highlights	Specifies whether to zoom to the extent of the results when highlighting.
	When automatically highlighting after a DRC run, RealTime does not zoom to highlights to avoid changing the currently selected viewpoint.
Highlight zoom	Specifies the magnification used when zooming to highlight a result.
Show check names	Specifies to display the check name at the center of each result.
Text size	Specify the text size when "Show check names" is enabled.
Check selection recipe	The check recipe used for the Calibre run. See "Check Selection Recipes in Calibre RealTime Custom" on page 105.
Edit	Open the Calibre RealTime Recipe Editor Dialog Box.

Usage Notes

• Environment Variable Usage

- You can use environment variables in pathnames for files and directories. The environment variable is not expanded when viewed in the dialog box or saved to the configuration file. For example, you can enter \$CAL_RULES/drc.rul for the rule file, and \$CAL_RULES/drc.rul is saved in the configuration file.
- You can enter an environment variable for a directory path in a file entry field and click the ... button to open the file browser to that directory. This makes it easier to browse for a file if it is not in the working directory and the directory path is long. When you select a file, the absolute path is entered in the text entry file and saved in the configuration file. This capability is available for the rule file, CTO file, layer map file, and object map file, but not for the runset file.
- o There are system defined environment for the input files; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

Behavior of "Group highlight colors by check"

When "Group highlight colors by check" is enabled, colors are assigned to results according to the rule check. However, the association between the highlight color and

the rule check is not fixed; it depends on the history of your Calibre RealTime session. Highlight colors are assigned to checks in a repeated cycle, starting with the color red. The color assignment starts at the beginning (with red) each time highlights are cleared. Highlights are cleared in the following cases:

- By clicking the Clear Highlights button ().
- O Automatically when the button in the integrated toolbar is clicked. The button deletes a rule check from the current check recipe and clears all highlights.

The following behavior is observed when "Group highlight colors by check" is enabled:

- o Highlight color assignments from a DRC run:
 - "Clear Existing Highlights" ON Highlights are cleared and highlight color assignment starts at red at the beginning of each Calibre RealTime run.
 - "Clear Existing Highlights" OFF Highlights from previous runs are kept. New results from a check with an existing highlighted result are highlighted in the same color. New results from a check that does not have an existing highlighted result are highlighted in the next color in the highlight color cycle.
 - Clicking or Highlights are cleared and highlight color assignment starts at red for the next run, regardless of the setting of "Clear Existing Highlights."
- o Highlight color assignments using the ← 🌣 🖒 buttons

 - "Clear Existing Highlights" OFF Each result is highlighted in the color used originally.

Clear Highlights button ✓ clicked when "Clear Existing Highlights" OFF — All highlights are cleared. The next result highlighted with one of the ⇔ ★ ⇔ buttons is highlighted in red; the highlight color for further results is assigned according to the rule check. All highlights are kept until ✓ is clicked again.

Related Topics

Calibre RealTime Configuration File (Laker)

Database Read Setup for Calibre RealTime with Synopsys Laker

You can customize the database read process by specifying layer map and object map files. You can also specify the number of vertices in the polygon used to approximate a circle or ellipse.

Specifying the Layer Map for Calibre RealTime with Synopsys Laker	17 2
Specifying the Object Map for Calibre RealTime with Synopsys Laker	17 4
Approximation of Circles and Ellipses in Calibre RealTime with Synopsys Laker	175

Specifying the Layer Map for Calibre RealTime with Synopsys Laker

You can use automatic layer mapping based on the technology file or specify a custom layer map.

Calibre RealTime for Synopsys Laker OpenAccess automatically maps the OpenAccess layer and purpose names found in the technology file used to create the database to the corresponding GDS layer number and datatype pairs for layout export; the mapping should correspond to what is expected by the Calibre rule file and any Layer Map statements within the rule file.

You can provide a layer map file as described in this procedure; information in the layer map file overrides the automatic layer map based on the technology file. The layer map can also be specified with the environment variable MGC_REALTIME_LAYER_MAP_FILE; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

Prerequisites

• Synopsys Laker is open with Calibre RealTime enabled; see "Invoking Synopsys Laker with Calibre RealTime" on page 153.

Procedure

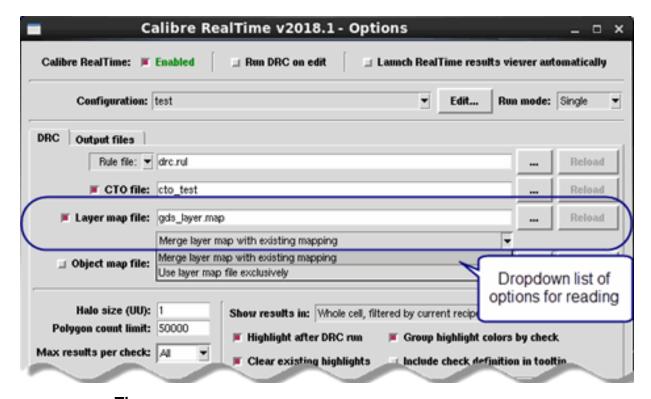
- 1. Create the layer map file as follows:
 - One entry per line with the following format:

```
OA_layer_name OA_purpose_name GDS_layer_number GDS_datatype
```

- ASCII text file.
- Comment lines start with #.
- Any filename is allowed, but the tool searches for files with a .map extension by default.

If you specify "Merge layer map with existing mapping" later in the procedure, it is not necessary to map all design layers, since layers not mapped in the layer map file are mapped according to the technology file. See "Results" for more information.

- 2. Click the 🐞 button to open the Calibre RealTime Options Dialog Box.
- 3. Enable the checkbox for "Layer map file."
- 4. Enter the layer map file you created in the beginning of the procedure.



You can enter an environment variable that is a directory path in the text entry field and click the ... button to have the file browser open to that directory.

Changes in the layer map file cause an automatic reload of the file prior to the DRC run.

- 5. Select the method for loading the layer map information:
 - Merge layer map with existing mapping (default) Merge layer map data from the specified file with the layer map derived from the technology file. Layer map entries in the layer map file override the corresponding layer map entry based on the technology file. All other layer map entries based on the technology file remain valid; only entries found in the layer map file are overridden
 - Use layer map file exclusively Use only layer map data from the specified file. Layer map data derived from the technology file is *not* used.

Caution

If "Use layer map file exclusively" is selected and the layer map file does not map all layers, then objects on unmapped layers are *not* analyzed by Calibre RealTime.

6. Click **Close** to close the Calibre RealTime Option dialog box.

Results

The following process is used when Calibre RealTime performs layer mapping:

- A layer map based on the technology file is developed when the design database is loaded. The layer map data based on the technology file is only used when "Merge layer map with existing mapping" (the default) is selected.
- If a layer map file is specified in the Calibre Options dialog box, the file is read the first time a DRC run is performed, and cached for later runs.

If the layer mapping file contains multiple mapping statements for a Laker design layer, then a warning is issued and the last layer mapping statement is used.

Errors in the layer map file are handled as follows:

- o If a format error is detected, an error is displayed and the layer map file is not used. The default technology-based layer map is used.
- If one or more layers in the layer map file are not present in the design database, an informational message is displayed with the list of layers. The remaining valid layer mappings take effect.
- To load a new or updated layer map file, specify the file in the Calibre RealTime
 Options dialog box, then click the **Reload** button. The new file is read before the next
 DRC run.

Specifying the Object Map for Calibre RealTime with Synopsys Laker

You can define a custom object map file in Calibre RealTime for Synopsys Laker.

Prerequisites

• Synopsys Laker is open with Calibre RealTime enabled; see "Invoking Synopsys Laker with Calibre RealTime" on page 153.

Procedure

1. Create the object map file; see "Object Map File Format in Calibre RealTime" on page 40 for the file format.

Note

If you use both an object map file and a layer map file, the same GDS layer number and datatype pairs should not be used in both files.

- 2. Click the button to open the Calibre RealTime Options Dialog Box.
- 3. Enable the checkbox for "Object map file."
- 4. Enter the object map file you created in Step 1.

Changes in the object map file cause an automatic reload of the file prior to the DRC run.

5. Click **Close** to close the Calibre RealTime Option dialog box.

Related Topics

Environment Variable Settings for Calibre RealTime Input Files

Object Map File Format in Calibre RealTime

Specifying the Layer Map for Calibre RealTime with Synopsys Laker

Approximation of Circles and Ellipses in Calibre RealTime with Synopsys Laker

Calibre RealTime approximates circles and ellipses found in Synopsys Laker databases as polygons. By default, Calibre RealTime uses a 20 point polygon to approximate an ellipse or circle.

You can specify the number of points used in the polygon approximation with the environment variable MGC_REALTIME_ELLIPSE_POINT_COUNT; the minimum allowed value is 3.

This environment variable only applies to Calibre RealTime with Synopsys Laker³.

Example

The following example instructs Calibre RealTime to use a 40 point polygon as the approximation for an ellipse or circle.

setenv MGC REALTIME ELLIPSE POINT COUNT 40

Calibre RealTime Setup with Tcl Commands in Synopsys Laker

Several Tcl commands are available to help you configure Calibre RealTime from within the Synopsys Laker environment.

The Tcl commands are described in the following sections in the chapter "Calibre RealTime Custom Basics":

- "Environment Variable Control in Calibre RealTime" on page 53
- "Option Control in the Design Environment for Calibre RealTime" on page 58

Enabling the Calibre RealTime Integration in Synopsys Laker

You must add some code to your \$HOME/.lakerenv file in order to enable Calibre RealTime. You must also set an environment variable to specify the location of the Calibre software tree.

Prerequisites

- See "Calibre RealTime Custom Requirements" on page 15 for system and licensing requirements.
- You are using a supported version of Synopsys Laker³; see "Supported Versions of Synopsys Laker³ for Calibre RealTime" on page 151.

Procedure

- 1. Set the environment variable MGC_HOME or CALIBRE_HOME to the location of the Calibre software tree. See "Setting the CALIBRE_HOME Environment Variable" in the Calibre Administrator's Guide for details.
- 2. Add the following code to your \$HOME/.lakerenv file

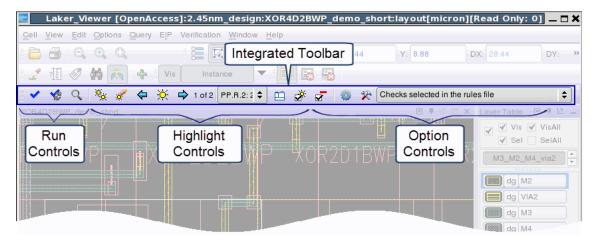
₋Tip

You can also view a README file at

\$MGC_HOME|shared|pkgs|icv|tools|realtime|laker|lakerREADME

Results

When Calibre RealTime is enabled, the Calibre RealTime Integrated Toolbar (Laker) is added to the Synopsys Laker window, as shown below:



See "Invoking Synopsys Laker with Calibre RealTime" on page 153 and the rest of the tasks in "Running Calibre RealTime with Synopsys Laker" on page 153 for more information on using Calibre RealTime with Synopsys Laker³.

Related Topics

License Timeout for Calibre RealTime Custom

Calibre RealTime Configuration File (Laker)

Calibre RealTime settings are saved in the Calibre RealTime configuration file. The configuration file is saved automatically and named .*realtime.cfg*; the file is saved in your \$HOME directory by default.

Settings in the Calibre RealTime Options Dialog Box and check recipe definitions are saved in the Calibre RealTime configuration file.

You can delete .realtime.cfg to return to the default configuration; however, this will also delete the custom check recipe definitions contained in the configuration file. You can save check recipes to file; see Export and Import in the description of the "Calibre RealTime Recipe Editor Dialog Box" on page 116.

Calibre RealTime for Synopsys Laker also supports management of the configuration settings by an administrator group; see "Multilevel Session Configuration File Support in Calibre RealTime Custom" on page 44 for information.

Caution_

Editing the configuration file is not recommended unless deleting complete lines in order to create a multilevel configuration file setup.

Keyboard Shortcuts (Hotkeys) in Calibre RealTime for Synopsys Laker

Several keyboard shortcuts are defined in Calibre RealTime for Synopsys Laker, and others can be enabled if desired.

The keyboard shortcuts are defined in the following files for each of the supported version of Laker

• Laker³:

\$CALIBRE HOME/shared/pkgs/icv/tools/realtime/laker/laker 3 realtime bindkeys.tcl

• Laker OA:

\$CALIBRE HOME/shared/pkgs/icv/tools/realtime/laker/calibre.key

The keyboard shortcuts file is not loaded by default to avoid possible conflicts with existing keyboard shortcuts. The definition of the bindkey functions is given in the following table. See "Defining and Loading Keyboard Shortcuts (Hotkeys) in Calibre RealTime for Synopsys Laker" on page 180 for instructions on using keyboard shortcuts.

The available keyboard shortcuts are listed in the following table:

Table 5-7. Keyboard Shortcuts in Calibre RealTime for Synopsys Laker

Keyboard key	Definition	
Default Keyboard Shortcuts		
F12	Run DRC in window.	
Shift-F12	Open the Calibre RealTime Options Dialog Box.	
Ctrl-i	Toggle the option "Run DRC on Edit" in the Calibre RealTime Options dialog box.	
User-Enabled Keyboard Shortcuts		
q	Select the next recipe in the toolbar dropdown list.	
W	Select the previous recipe in the toolbar dropdown list.	
е	Select the most recently used recipe in the toolbar dropdown list.	
p	Highlight the previous result.	
c	Highlight the current result.	
n	Highlight the next result.	
Ctrl-p	Select the previous check in the toolbar dropdown list.	

Table 5-7. Keyboard Shortcuts in Calibre RealTime for Synopsys Laker

Keyboard key	Definition
Ctrl-c	Highlight results in the check currently selected in the toolbar dropdown list.
Ctrl-d	Remove the selected check from the current recipe.
Ctrl-n	Select the next check in the toolbar dropdown list.

Defining and Loading Keyboard Shortcuts (Hotkeys) in Calibre RealTime for Synopsys Laker

You can define custom keyboard shortcuts for Calibre RealTime. Several default keyboard shortcuts are defined, along with option user-defined keyboard shortcuts which can be enabled.

Prerequisites

• Locate the pre-defined keyboard shortcuts file corresponding to the Laker version you are using. The directory path is:

\$CALIBRE_HOME/shared/pkgs/icv/tools/realtime/laker

The filenames are:

- Laker³ laker_3_realtime_bindkeys.tcl
- o **Laker OA** calibre.key

Procedure

1. Copy the appropriate keyboard shortcuts file to a file in your home or project directory. For example:

```
cp <shortcuts file> ~/my calibre.key
```

- 2. Edit the shortcut definitions as desired. You can enable shortcuts listed as user-enabled in "Keyboard Shortcuts (Hotkeys) in Calibre RealTime for Synopsys Laker" on page 179 by un-commenting the definition.
- 3. Load the new shortcut definitions into Synopsys Laker.

For example, for Laker³, use the following command:

```
source ~/my_calibre.key
```

For Laker OA, use the following command:

```
gtSetBindKey -windowType LakerDsgWnd -file ~/my calibre.key
```

Related Topics

Keyboard Shortcuts (Hotkeys) in Calibre RealTime for Synopsys Laker



Chapter 6 Calibre RealTime Custom in Synopsys Custom Compiler

Calibre RealTime Custom is integrated with Synopsys® Custom Compiler $^{\text{\tiny{TM}}}$. Calibre RealTime Custom integrates flat Calibre nmDRC with supported layout design tools so that Calibre nmDRC can be run directly from the design tool.

Refer to "Calibre RealTime Custom Introduction" on page 15 for an overview and general system requirements.

Requirements and Setup for Calibre RealTime in Synopsys Custom Compiler	184
Invoking and Running Calibre RealTime in Synopsys Custom Compiler	18'
Keyboard Shortcuts in Calibre RealTime for Synopsys Custom Compiler	19'

Requirements and Setup for Calibre RealTime in Synopsys Custom Compiler

Requirements for tool versions, input files, and environment variables must be met in order to run Calibre RealTime in Synopsys Custom Compiler.

General Requirements for Calibre RealTime in Synopsys Custom Compiler	184
Tcl Script for Calibre Interface to Synopsys Custom Compiler	185
$ Synopsys\ Custom\ Compiler\ and\ Calibre\ Real Time\ Custom\ Version\ Compatibility\ .\ .$	186
GDS Layer Map for Calibre RealTime in Synopsys Custom Compiler	186

General Requirements for Calibre RealTime in Synopsys Custom Compiler

Certain requirements for design tool versions, layer map files, and environment variables must be met to run Calibre RealTime in Synopsys Custom Compiler.

- Synopsys Custom Compiler version L-2016.06. See "Synopsys Custom Compiler and Calibre RealTime Custom Version Compatibility" on page 186 for the complete list of supported releases.
- Calibre version 2017.3 or later, and general requirements met, as described in "Calibre RealTime Custom Requirements" on page 15.
- Integration Script

See "Tcl Script for Calibre Interface to Synopsys Custom Compiler" on page 185.

• Layer Map Requirement

A layer map file that maps the Synopsys Custom Compiler layer objects to GDS layers and datatypes is required. The mapping should correspond to what is expected by the Calibre rule file and any Layer Map statements within the rule file.

The layer map file must be specified in the Calibre RealTime Options dialog box. See "GDS Layer Map for Calibre RealTime in Synopsys Custom Compiler" on page 186 for more information.

• Environment Variable Requirements

No special environment variables are required.

Tip

You can change environment variables using Tcl API commands in the Synopsys Custom Compiler environment. This is useful if you want to change environment variables used in the rule file. See "Environment Variable Control in Calibre RealTime" on page 53.

You can also set Calibre RealTime options using Tcl API commands; see "Option Control in the Design Environment for Calibre RealTime" on page 58.

Tcl Script for Calibre Interface to Synopsys Custom Compiler

The Calibre integration to Synopsys Custom Compiler requires a Tcl script to be loaded.

Tcl Integration Script

Copy and paste the following code into \$HOME/.synopsys_custom.tcl:

```
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 is not set."
      puts "Calibre RealTime will not be loaded."
      return
   } else {
     puts "Environment variable CALIBRE HOME is not set, "
      puts "using MGC HOME instead."
      set env(CALIBRE HOME) $env(MGC HOME)
  set initFile [file join $env(CALIBRE HOME) shared pkgs icv tools \
    realtime custom compiler custom compiler realtime install.tcl]
  if { [file readable $initFile] } {
    if { [catch { source $initFile } msg] ne 0 } {
     puts "ERROR: Could not load Calibre RealTime: $msg"
  } else {
      puts "ERROR: Could not find Calibre RealTime initialization files."
     puts "Calibre RealTime will not be loaded."
      return
::load calibre
```

Synopsys Custom Compiler and Calibre RealTime Custom Version Compatibility

The support of Calibre RealTime Custom in Synopsys Custom Compiler depends on the Calibre release and the Synopsys version.

Table 6-1. Synopsys Custom Compiler and Calibre RealTime Custom Version Compatibility

Synopsys Custom Compiler Version	Supporting Calibre Releases
L-2016.06	2017.3 and later
M-2017.03	2019.1_18 and later
Q-2020.03	2020.3_16 and later

GDS Layer Map for Calibre RealTime in Synopsys Custom Compiler

A layer map file is required to map the Synopsys Custom Compiler layer objects to GDS layers and datatypes for layout export. The layer map file is specified in the Calibre RealTime Options dialog box.

File Format

Consult the Synopsys Custom Compiler documentation. This is the general format:

OA_layer OA_purpose GDS_layer_num GDS_datatype

Loading the Layer Map File

See "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Custom Compiler)" on page 190.

Invoking and Running Calibre RealTime in Synopsys Custom Compiler

Invoking Synopsys Custom Compiler with Calibre RealTime

After invoking Synopsys Custom Compiler with Calibre RealTime, you see the Calibre RealTime integrated toolbar.

Prerequisites

- "Requirements and Setup for Calibre RealTime in Synopsys Custom Compiler" on page 184 completed.
- System and license requirements described in "Calibre RealTime Custom Requirements" on page 15 are satisfied.

Procedure

1. Start Synopsys Custom Compiler and open your design.

It may take a minute or so to acquire all licenses and complete the initialization. The following lines are displayed in the transcript after successful initialization:

```
// Calibre RealTime: Server initialized successfully at host:port
// Calibre RealTime: Initialization complete
```

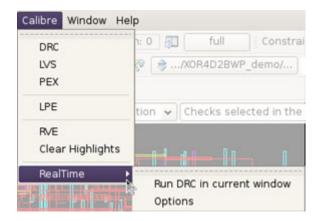
If you do not see the preceding notices, check the items listed in the Prerequisites.

A partial view of the design window in Synopsys Custom Compiler is shown in the following figure. The Calibre RealTime Integrated Toolbar (Custom Compiler) is loaded.

Figure 6-1. Calibre RealTime in Synopsys Custom Compiler

2. Select **Tools > Calibre** to enable the **Calibre** menu. Verify that the **RealTime** item was added to the **Calibre** menu, as shown in the following figure.

If you do not see the **RealTime** menu item, exit Synopsys Custom Compiler and check the items listed in the Prerequisites.



3. Proceed to "Specifying and Using Multiple Run Configurations in Calibre RealTime with Synopsys Custom Compiler" on page 188.

Note

Calibre RealTime option settings and check recipe definitions are automatically saved in the Session Configuration File in Calibre RealTime Custom and restored the next time you open Synopsys Custom Compiler.

Specifying and Using Multiple Run Configurations in Calibre RealTime with Synopsys Custom Compiler

A run configuration in Calibre RealTime is the set of options and input files for a run. You can define multiple run configurations with different options (such as the rule file, check recipe and

other options), then quickly switch between different run configurations. You can run with one configuration (single mode) or run multiple configurations in sequence (serial mode).

The first time you open Calibre RealTime, the configuration un mode is set to Single and a run configuration named "Default configuration" is created. If you plan on using only one run configuration, no further action is needed.

Multiple run configurations can be used in the following modes:

- **Single** Run only the currently selected configuration when a run is started.
- **Serial** Run each configuration selected in the Configuration Run Control dialog box in order.

Tip

0

You can also set options related to the run configurations with Tcl API commands; see "Option Control in the Design Environment for Calibre RealTime" on page 58.

Prerequisites

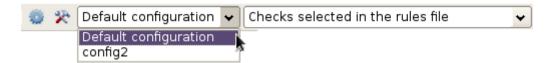
• You have completed "Invoking Synopsys Custom Compiler with Calibre RealTime" on page 187.

Procedure

- 1. Choose **Calibre > RealTime DRC > Options** to open the "Calibre RealTime Options Dialog Box" on page 31, or click the **Options** button () on the integrated toolbar.
- 2. If using multiple run configurations, define as many configuration names as needed:
 - a. Click the **Edit** button next to the configuration name to open the Configuration Run Control dialog box.
 - b. Click the "+" button to add a configuration and specify the name. Repeat this step as needed.
 - c. If using the Serial run mode, set the run order using the up and down arrows. Use the checkbox to include or remove a configuration from the execution list.
 - d. Click **OK** to close the Configuration Run Control dialog box.
- 3. Define the settings for each run configuration:
 - a. Select the configuration in the Configuration dropdown list.
 - b. Specify the settings as described in "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Custom Compiler)" on page 190.
- 4. Select the "Run mode":
 - **Single** Run only the currently selected configuration when a run is started.

- **Serial** Run each configuration selected in the Configuration Run Control dialog box in order.
- 5. Proceed to "Running Calibre RealTime in Synopsys Custom Compiler" on page 192.

Results are displayed separately for each run configuration. Select the configuration to view with the toolbar—click the toolbar setting to select the configuration or check recipe from a menu:



Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Custom Compiler)

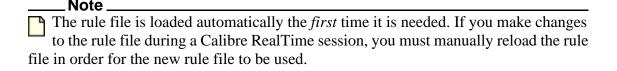
Calibre RealTime uses a standard Calibre nmDRC rule file. The check recipe controls which checks in the rule file are executed during the Calibre RealTime run.

Prerequisites

- You have completed "Invoking Synopsys Custom Compiler with Calibre RealTime" on page 187.
- You have completed "Specifying and Using Multiple Run Configurations in Calibre RealTime with Synopsys Custom Compiler" on page 188.
- You have a Calibre nmDRC rule file available; see "Calibre RealTime Custom Requirements" on page 15 and "Differences Between Calibre nmDRC and Calibre RealTime Custom" on page 18 for additional information about rule file requirements.

Procedure

- 1. Choose **Calibre > RealTime DRC > Options** to open the Calibre RealTime Options dialog box, or click the **Options** button () on the integrated toolbar.
- 2. Select the configuration in the "Configuration" dropdown list.
- 3. Specify the Calibre rule file in the "Rule file" text field.
- 4. Click **Reload** to load the rule file.



- 5. Specify a layer map in the "Layer map" field. This is required. See "GDS Layer Map for Calibre RealTime in Synopsys Custom Compiler" on page 186.
- 6. (Optional) Select options in the dialog box; the defaults are suitable for most runs. See "Calibre RealTime Options Dialog Box" on page 31 for a definition of each option.

If you want the Calibre RealTime Results window to open automatically after a run, enable the option "Launch RealTime results viewer automatically."

If you want to save a summary file of the run and the DRC results as an ASCII results database (RDB), select the **Output files** tab and check the corresponding options.

- 7. Select the check recipe as follows:
 - Checks selected in the rules file (default) This built-in recipe uses the Calibre rule file just as a batch run of Calibre does.
 - Other built-in check recipes You can select a recipe that only runs checks on certain layers, excludes certain types of checks, or runs all checks in the rule file.

Because these built-in recipes do *not* obey DRC [Un]Select Check statements in the rule file, the recipes may execute checks in the rule file that would not be executed during a batch Calibre run. The recipe "Checks selected in the rules file" is the only built-in recipe that obeys DRC [Un]Select Check statements. See "Built-In Check Recipes" on page 107.

• User recipes — You can select a previously defined custom check recipe. See "Creating a Custom Check Selection Recipe" on page 110.

Show check description window Check selection recipe: Checks selected in the rules file Edit... USER RECIPES metal checks no density Custom recipes BUILT-IN RECIPES All checks Visible layer checks These recipes Visible layer checks, no connectivity do not obey Visible layer only checks DRC [Un]Select Visible layer only checks, no connectivity statements. "Checks selected in the rules file" obeys DRC [Un]Select statements-the rule file is read the same as for a batch Calibre run.

Figure 6-2. Selecting a Check Recipe in Calibre RealTime (Custom Compiler)

8. Proceed to "Running Calibre RealTime in Synopsys Custom Compiler" on page 192.

Running Calibre RealTime in Synopsys Custom Compiler

Calibre RealTime runs checks on the geometries visible in the Synopsys Custom Compiler window plus a halo around the region. The specified check recipe selects the checks that are executed. All Calibre RealTime DRC runs are performed in flat mode.

Prerequisites

• You have completed "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Custom Compiler)" on page 190.

Procedure

- 1. Select a run configuration in the "Configuration" dropdown list, or, if running in Serial mode, make sure the "Run Mode" is set to Serial in the Calibre RealTime Options dialog box. See "Specifying and Using Multiple Run Configurations in Calibre RealTime with Synopsys Custom Compiler" on page 188.
- 2. Zoom to the region you want to run DRC on and make sure all the geometries you want to run checks on are visible. A region of 25 um x 25 um or smaller is recommended for the best performance.
- 3. Start the run by clicking the \checkmark (Run DRC in window) toolbar button.

This runs flat Calibre nmDRC on the geometries visible in the design window plus the specified halo region around the area. (Keyboard shortcut F12)

If running in Serial mode, a DRC run is executed for each of the configurations selected in the "Configuration Run Control" dialog box, in the order specified.

Results

The results of a Calibre RealTime run are shown in Figure 6-3. You can use the highlight controls in the integrated toolbar to step through results. If you are using multiple run configurations, the results are displayed separately for each configuration—select the configuration to view in the toolbar dropdown box.

You can also click the toolbar icon to open the Calibre RealTime Results window to view results in a separate window; see "Viewing Results in the Calibre RealTime Results Window" on page 83.

Rule check comments can be seen by hovering over the rule check name in the toolbar; the check definition is also included if "Include Check Definition in Tool Tip" was enabled at the time of the run.

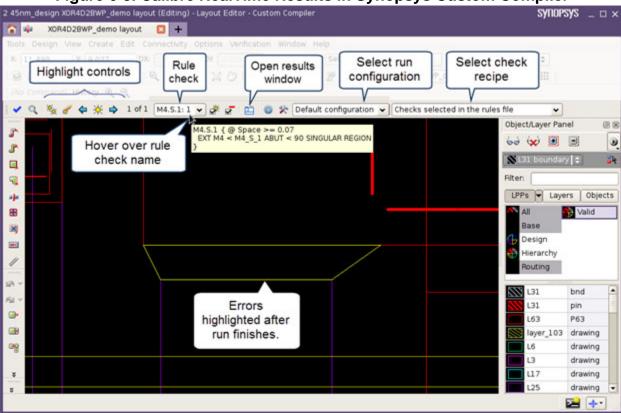


Figure 6-3. Calibre RealTime Results in Synopsys Custom Compiler

Note

Calibre RealTime option settings and check recipe definitions are automatically saved in the Session Configuration File in Calibre RealTime Custom and restored the next time you open Synopsys Custom Compiler.

Calibre RealTime Integrated Toolbar (Custom Compiler)

The Calibre RealTime integrated toolbar includes controls to start a DRC run, highlight results, and open the Calibre RealTime Options dialog box. The toolbar also includes option settings for run control and highlighting.

The Calibre RealTime integrated toolbar in Synopsys Custom Compiler is shown in the following figure.



The run controls, highlight controls, and options controls are described in the following sections.

Run Controls (Custom Compiler)

The run controls in the Calibre RealTime integrated toolbar start a window run and zoom to the window for the most recent DRC run. The run controls are shown below and described in the table following the figure.



Table 6-2. Run Control Toolbar Selections (Custom Compiler)

Toolbar item	Description
(Run DRC in window)	Runs flat DRC on the geometries visible in the current window, plus a halo region around the area. Calibre runs with the currently selected Run Configuration and configuration run mode (Serial or Single).
	Keyboard shortcut: F12
Q	Zoom to the window of the most recent DRC run.

Highlight Controls (Custom Compiler)

The highlight control section the Calibre RealTime integrated toolbar includes highlight controls and displays the rule check for the current result. The highlight controls are shown below and described in the table following the figure.



Table 6-3. Highlight Control Toolbar Selections (Custom Compiler)

Toolbar item	Description
%	Highlight all.
#	Clear all highlights.
⇔ 🔆 🗢	Highlight previous, current, or next result for the rule check.
	Also see "Group highlight colors by check" in "Calibre RealTime Options Dialog Box" on page 31.

Table 6-3. Highlight Control Toolbar Selections (Custom Compiler) (cont.)

Toolbar item	Description
1 of 5	Result <i>n</i> of <i>total</i> results.
	Also see the setting "Show results in" in the Calibre RealTime Options Dialog Box; this setting determines whether all results for the cell or only results from the most recent run are reported.
Rule check	Indicates the rule check which produced the displayed error when using the highlight previous, current, and next buttons. Use the dropdown arrow to select a different rule check.
ૐ	Highlight all results from the selected rule check.
∡_	Remove the displayed rule check from the current check recipe. All highlights are cleared, regardless of the setting of "Clear Existing Highlights".
	If the recipe is a built-in recipe, you are prompted to save the new check recipe to a new filename; if the recipe is a user recipe, the updated recipe is saved automatically.
Ë	Open the Calibre RealTime Results window. See "Viewing Results in the Calibre RealTime Results Window" on page 83.

Option Controls (Custom Compiler)

The option controls in the Calibre RealTime integrated toolbar select the check recipe, run configuration, and open the Calibre RealTime Options dialog box. The option controls are described in the table following the figure.



Table 6-4. Option Control Toolbar Selections (Custom Compiler)

Toolbar item	Description
	Open the Calibre RealTime Options Dialog Box.
*	Open the Customization GUI. This button is only available if a Calibre Interactive runset is loaded and includes a customization file.
	See "Using the Customization GUI with Calibre RealTime" on page 88.

Table 6-4. Option Control Toolbar Selections (Custom Compiler) (cont.)

Toolbar item	Description
Run Configuration	Selects the run configuration, both for a run in "Single" mode and for viewing results; see "Specifying and Using Multiple Run Configurations in Calibre RealTime with Synopsys Custom Compiler" on page 188.
Check recipe	Select the check recipe from the dropdown list; the check recipe specifies the rule checks executed for all Calibre RealTime runs. See "Check Selection Recipes in Calibre RealTime Custom" on page 105 for more information.

Keyboard Shortcuts in Calibre RealTime for Synopsys Custom Compiler

A set of standard keyboard shortcuts are provided for Calibre RealTime Custom with Synopsys Custom Compiler.

The keyboard shortcuts are defined in the following file:

```
$CALIBRE_HOME/shared/pkgs/icv/tools/realtime/custom_compiler/custom compiler realtime bindkeys.tcl
```

This file is not loaded by default to avoid possible conflicts with existing keyboard shortcuts. The definition of the bindkey functions is given in table Table 6-5. The section "Defining and Loading Keyboard Shortcuts in Calibre RealTime Custom for Synopsys Custom Compiler" on page 198 describes how to define and load your own keyboard shortcuts.

Table 6-5. Keyboard Shortcuts in Calibre RealTime Custom for Synopsys

Custom Compiler

Bindkey Function Definition	Standard Keyboard Shortcut
::calibre::realtime::RunDRC	F12
Run DRC in window.	
::calibre::realtime::ShowOptionsForm	Shift-F12
Open the Calibre RealTime Options Dialog Box.	
::calibre::realtime::HighlightAllResults	Ctrl-a
Highlight all results.	
::calibre::realtime::ClearAllHighlights	Ctrl-b
Clear all highlights.	
::calibre::realtime::HighlightPreviousResult	p
Highlight the previous result.	
::calibre::realtime::HighlightCurrentResult	c
Highlight the current result.	
::calibre::realtime::HighlightNextResult	n
Highlight the next result.	
::calibre::realtime::HighlightCurrentCheck	Shift-c
Highlight results in the check currently selected in the toolbar dropdown list.	
::calibre::realtime::RemoveCurrentCheckFromCurrentRecipe	Shift-d
Remove the selected check from the current recipe.	

Table 6-5. Keyboard Shortcuts in Calibre RealTime Custom for Synopsys Custom Compiler (cont.)

Bindkey Function Definition	Standard Keyboard Shortcut
::calibre::realtime::NextRecipe	q
Select the next recipe in the toolbar dropdown list.	
::calibre::realtime::PreviousRecipe	w
Select the previous recipe in the toolbar dropdown list.	
::calibre::realtime::LastUsedRecipe	e
Select the most recently used recipe in the toolbar dropdown list.	

Defining and Loading Keyboard Shortcuts in Calibre RealTime Custom for Synopsys Custom Compiler

You can define custom keyboard shortcuts for Calibre RealTime Custom. The Calibre installation provides a file of standard keyboard shortcuts.

Procedure

1. Copy the Tcl file \$CALIBRE_HOME/shared/pkgs/icv/tools/realtime/custom_compiler/custom_compiler_realtime_bindkeys.tcl to the desired location for your keyboard shortcut definitions. "Keyboard Shortcuts in Calibre RealTime for Synopsys Custom Compiler" on page 197 lists the shortcuts defined in this file.

For example:

- 2. Edit the shortcut definitions as desired.
- 3. Load the new shortcut definitions into Synopsys Custom Compiler with the following command:

```
source ~/my realtime bindkeys.tcl
```

Tip

If you want to use the standard shortcuts, and are not concerned about possible conflicts with other shortcuts, you can load the standard shortcuts. Issue the following command at the Synopsys Custom Compiler command prompt, where the command is all one line:

source \$env(CALIBRE_HOME)/shared/pkgs/icv/tools/realtime/
custom_compiler/custom_compiler_realtime_bindkeys.tcl



Chapter 7 Calibre RealTime Custom with Cadence Virtuoso

Calibre RealTime Custom is integrated with the OpenAccess version of Cadence Virtuoso. Calibre RealTime Custom integrates flat Calibre nmDRC with supported layout design tools so that Calibre nmDRC can be run directly from the design tool.

Refer to "Calibre RealTime Custom Introduction" on page 15 for an overview and general system requirements.

Also see Calibre RealTime Integration to Cadence Virtuoso Quick Reference.

Requirements and Setup for Calibre RealTime with Cadence Virtuoso	202
Invoking and Running Calibre RealTime with Cadence Virtuoso	209
Toolbars, Menus, and Dialog Boxes in Calibre RealTime (Virtuoso)	225
Database Read Setup for Calibre RealTime with Cadence Virtuoso	242
Calibre RealTime Setup with Cadence SKILL Commands	248
Calibre RealTime Configuration File (Virtuoso)	299
Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso	300
Troubleshooting Calibre RealTime with Cadence Virtuoso	302

Requirements and Setup for Calibre RealTime with Cadence Virtuoso

Requirements for tool versions, input files, and environment variables must be met in order to run Calibre RealTime with Cadence Virtuoso.

General Requirements for Calibre RealTime with Cadence Virtuoso	20 2
Environment Setup for Cadence Virtuoso with Calibre RealTime	203
Cadence Virtuoso and Calibre RealTime Version Compatibility	204
Setting the Calibre RealTime Version Newer than CALIBRE_HOME	205

General Requirements for Calibre RealTime with Cadence Virtuoso

Certain requirements for design tool versions, layer map files, and environment variables must be met to run Calibre RealTime for Cadence Virtuoso.

The following list gives the detailed requirements:

- Cadence Virtuoso Open Access version 6.1.5, 6.1.6, 12.1, or 12.2, running in 64-bit mode. See "Cadence Virtuoso and Calibre RealTime Version Compatibility" on page 204 for the complete list of supported releases.
- Calibre version 2013.1 or later, and general requirements met, as described in "Calibre RealTime Custom Requirements" on page 15.
- Layer Map Requirement

A layer map file that maps the Cadence Virtuoso layer and purpose pairs to GDS layers and datatypes is required. The mapping should correspond to what is expected by the Calibre rule file and any Layer Map statements within the rule file.

The layer map file is read automatically if it located in the technology library directory and has the name *tech_lib.layermap*, where *tech_lib is* the name of the technology library. You can also manually specify the layer map file, as described in "GDS Layer Map for Calibre RealTime with Cadence Virtuoso" on page 242.

• Environment Variable Requirements

Environment variables are defined as described in "Environment Setup for Cadence Virtuoso with Calibre RealTime" on page 203.

You may specify that Calibre RealTime run a different version than your Calibre version specified by CALIBRE_HOME. See "Setting the Calibre RealTime Version Newer than CALIBRE_HOME" on page 205.

• If using pre-coloring for multi-patterning, see "Multi-Patterning Support in Calibre RealTime Custom in Cadence Virtuoso" on page 247.

Environment Setup for Cadence Virtuoso with Calibre RealTime

Several environment variable settings are needed for Calibre RealTime operation with Cadence Virtuoso.

The following environment variable settings are required, where the examples are shown for the csh shell:

• Define the following environment variable to enable Calibre RealTime:

```
setenv MGC CALIBRE REALTIME VIRTUOSO ENABLED 1
```

Calibre RealTime with Cadence Virtuoso is enabled when this environment variable is set to any value. To disable Calibre RealTime with Cadence Virtuoso, unset MGC_CALIBRE_REALTIME_VIRTUOSO_ENABLED. (MGC_REALTIME_DISABLE is not used in the Cadence Virtuoso integration)

• Set the environment variable OA_PLUGIN_PATH as follows:

```
setenv OA PLUGIN PATH ${MGC HOME}/shared/pkgs/icv/tools/queryskl
```

However, if you are using MGC_REALTIME_HOME, use MGC_REALTIME_HOME rather than MGC_HOME in defining OA_PLUGIN_PATH.

• The LD_LIBRARY_PATH environment variable should include the following path:

```
$MGC HOME/shared/pkgs/icv/tools/calibre client/lib/64
```

For example (all one line):

```
setenv LD_LIBRARY_PATH $MGC_HOME/shared/pkgs/icv/tools/
calibre_client/lib/64:${LD_LIBRARY_PATH}
```

However, if you are using MGC_REALTIME_HOME, use MGC_REALTIME_HOME rather than MGC_HOME in defining LD_LIBRARY_PATH.

Make sure the environment variables are set correctly for your Cadence installation.
 Typically this means defining the location of the Cadence installation (with CDS_INST_DIR or some other environment variable), and including the Cadence executables in your path environment variable.

Setting the following environment variables is optional:

• MGC_CALIBRE_REALTIME_VIRTUOSO_SAVE_MESSENGER_CELL

Setting this environment variable causes Calibre RealTime to automatically save the MentorMessengerLib. The MentorMessengerLib is used internally by Calibre RealTime. If this environment variable is not set when you are using Calibre RealTime for Cadence Virtuoso, other Cadence tools in the design flow sometimes give a prompt to save the MentorMessengerLib.

MGC_CALIBRE_REALTIME_VIRTUOSO_REMOVE_LIBS_ON_EXIT

Setting this environment variable causes Calibre RealTime to delete the MentorMessengerLib and MentorObserverLib entries from the cds.lib file if Calibre RealTime is disabled when you exit Cadence Virtuoso.

See "Deleting Mentor Libraries When Calibre RealTime is Disabled" on page 223.

MGC_REALTIME_HOME

You can set MGC_REALTIME_HOME to specify a version for Calibre RealTime that is different than the Calibre version set with CALIBRE_HOME. This allows you to run a later version of Calibre RealTime while leaving CALIBRE_HOME pointing to a qualified version of Calibre. See "Setting the Calibre RealTime Version Newer than CALIBRE_HOME" on page 205.

MGC_CALIBRE_REALTIME_SHOW_DRC_RULE_DESCRIPTION_IN_TOOLBAR

By default the check text (if present) for a rule check is shown in the Calibre RealTime window; see the figure in "Running Calibre RealTime with Cadence Virtuoso" on page 215. Define this environment variable to also display the check text as a tooltip when you hover over the rule check name in the Calibre RealTime integrated toolbar.

Related Topics

Environment Variables in Calibre RealTime Custom

Cadence Virtuoso and Calibre RealTime Version Compatibility

The support of Calibre RealTime with Cadence Virtuoso depends on the Calibre release and the Cadence version.

Table 7-1. Cadence Virtuoso and Calibre RealTime Version Compatibility

Cadence Virtuoso Version	Minimum Calibre Release	Recommended Calibre Releases
IC 6.15 Series		
IC 6.15.500 (ISR11-ISR15)	2014.2_35	2016.2_18 ¹
IC 6.15.500 (ISR16 - ISR17)	2014.2_35	2016.2_18 ¹

Table 7-1. Cadence Virtuoso and Calibre RealTime Version Compatibility

Cadence Virtuoso Version	Minimum Calibre Release	Recommended Calibre Releases
IC 6.16 Series		
IC 6.16.005	2015.2_19	2016.2_18 ¹
IC 6.16.500 (ISR2 - ISR12)	2015.2_19	2016.2_18 ¹
IC 6.16.500 (ISR13)	2015.4_33	2016.2_18 ¹
IC 6.17 Series		
IC 6.17.720 (ISR1 - ISR20)	2018.2_15	2020.3_16
IC 6.18 Series		
IC 6.18.000	2019.3_8	2021.2_18
IC 12.1 Series		
IC12.1.500 (ISR1 - ISR2)	2014.2_35	2016.2_18 ¹
IC12.1.500 (ISR3 - ISR14)	2015.4_33	2016.2_18 ¹
IC12.1.500 (ISR15)	2015.4_33	2016.2_18 ¹
IC 12.2 Series		
IC12.2.100 (ISR1 - ISR6)	2016.4_27	2017.2_16
IC12.2.700 (ISR1 - ISR7)	2016.4_27	2017.2_16
IC 12.3 Series		
IC12.3.720 (ISR1 - ISR20)	2018.2_15	2021.2_18
IC 18.10 Series		
IC 18.10.000	2019.3_8	2021.2_18

¹ Later Calibre releases may work but have not been tested and officially qualified.

If Calibre RealTime detects an unsupported Cadence Virtuoso version an error message is issued and the integration is not loaded.

Setting the Calibre RealTime Version Newer than CALIBRE_HOME

When using Calibre RealTime for Cadence Virtuoso, you can set the environment variable MGC_REALTIME_HOME to specify a version for Calibre RealTime that is newer than the Calibre version set with CALIBRE_HOME. This is useful if you want to run the latest version

of Calibre RealTime while leaving CALIBRE_HOME pointing to an earlier, qualified version of Calibre.

The use of MGC_REALTIME_HOME is only intended for cases in which MGC_REALTIME_HOME points to a newer release than CALIBRE_HOME. This is enforced starting with the 2014.3 release; however the details about the enforcement process depend on the release as follows:

MGC_REALTIME_HOME	Enforcement of MGC_REALTIME_HOME newer than CALIBRE_HOME
2015.2 and later	Calibre RealTime is not loaded and an error message is issued; this behavior occurs if both software versions are 2015.2 or later.
2015.1	Enforced in all cases; a warning message is issued that MGC_REALTIME_HOME is ignored.
2014.3 through 2014.4	Enforced if major versions differ; a warning message is issued that MGC_REALTIME_HOME is ignored.
	However, for releases differing by the minor version, a warning message is still issued but the version specified by MGC_REALTIME_HOME is used and <i>not</i> changed to CALIBRE_HOME.
2014.2 and earlier	No enforcement; MGC_REALTIME _HOME is used as specified.

Note:

While there are no restrictions on the value of CALIBRE_HOME, compatibility testing is only performed for MGC_REALTIME_HOME set to the current version of Calibre and CALIBRE_HOME set to 1 and 2 releases earlier. MGC_REALTIME_HOME must point to version 2013.4_37 (UR2) or later.

Prerequisites

- "Requirements and Setup for Calibre RealTime with Cadence Virtuoso" on page 202 completed.
- System and license requirements described in "Calibre RealTime Custom Requirements" on page 15 are satisfied.

Procedure

1. Make sure that your .*cdsinit* file includes the correct logic for loading the *calibre.skl* file. The .*cdsinit* file should include code similar to the following:

The above code is just an excerpt from the complete code for setting up your Calibre integration in the .*cdsinit* file. See the following file for complete instructions and example code:

\$CALIBRE_HOME/shared/pkgs/icv/tools/queryskl/skillREADME

- 2. Set the CALIBRE_HOME environment variable to the version you want to run for batch Calibre, Calibre Interactive, and Calibre RVE. CALIBRE_HOME also determines the version for the Calibre DESIGNrev family of layout viewers.
- 3. Set the environment variable MGC_REALTIME_HOME to point to the version of Calibre you want Calibre RealTime to run. MGC_REALTIME_HOME must point to version 2013.4_37 (UR2) or later.
- 4. Make sure that OA_PLUGIN_PATH and LD_LIBRARY_PATH are defined using MGC_REALTIME_HOME rather than MGC_HOME, as described in "Environment Setup for Cadence Virtuoso with Calibre RealTime" on page 203.

For example:

setenv OA_PLUGIN_PATH \${MGC_REALTIME_HOME}/shared/pkgs/icv/tools/queryskl
setenv LD_LIBRARY_PATH \${MGC_REALTIME_HOME}/shared/pkgs/icv/tools/
 calibre client/lib/64:\${LD_LIBRARY_PATH}

Note

Calibre RealTime issues an error in the CIW if the version of LD_LIBRARY_PATH does not match the version of Calibre RealTime.

5. Start Cadence Virtuoso as described in "Invoking Cadence Virtuoso with Calibre RealTime" on page 209.

Results

If MGC_REALTIME_HOME and CALIBRE_HOME are different, then the CIW reports the value of each environment variable in a notice similar to the following:

```
// Calibre RealTime: MGC_HOME = "path", MGC_HOME is different with
MGC_REALTIME_HOME.
// Calibre RealTime: MGC_REALTIME_HOME = "rt_path", SKILL code in
MGC REALTIME HOME will be used!
```

where "path" and "rt_path" are replaced with the paths to the Calibre software.

Related Topics

Requirements and Setup for Calibre RealTime with Cadence Virtuoso

Invoking and Running Calibre RealTime with Cadence Virtuoso

Invoking Cadence Virtuoso with Calibre RealTime

After invoking the Cadence Virtuoso editor with Calibre RealTime, you see the Calibre RealTime integrated toolbar.

Prerequisites

- "Requirements and Setup for Calibre RealTime with Cadence Virtuoso" on page 202 completed.
- System and license requirements described in "Calibre RealTime Custom Requirements" on page 15 are satisfied.
- (Recommended) "Database Read Setup for Calibre RealTime with Cadence Virtuoso" on page 242 completed.

Procedure

1. Start Cadence Virtuoso OA in 64-bit mode and open a layout.

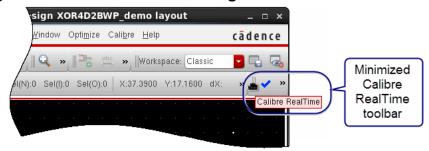
A partial view of the design window in Cadence Virtuoso is shown in Figure 7-1. The Calibre RealTime Integrated Toolbar (Virtuoso) is loaded, but the toolbar is initially minimized, as shown.

It may take a minute or so to acquire all licenses and complete the initialization. The following lines are displayed in the CIW after successful initialization:

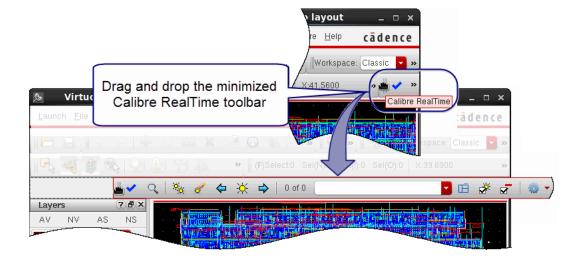
```
Calibre server launched successfully
// Calibre RealTime embedded client initialized successfully at
host:port
// Calibre RealTime: Initialization complete
```

If you do not see notices that both the server and embedded client started, check that all environment variables are set as described in "Environment Setup for Cadence Virtuoso with Calibre RealTime" on page 203.

Figure 7-1. Calibre RealTime Integration with Cadence Virtuoso

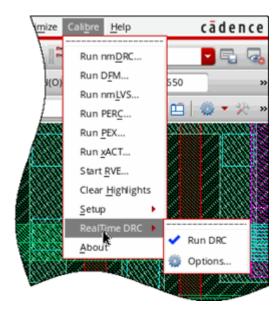


2. Drag and drop the Calibre RealTime toolbar to expand it, as shown in the following figure:



3. Verify that the **RealTime DRC** item was added to the **Calibre** menu, as shown in the following figure.

If you do not see the **RealTime DRC** menu item, exit Cadence Virtuoso, then make sure all environment variables are defined properly, as described in "Environment Setup for Cadence Virtuoso with Calibre RealTime" on page 203.



4. Proceed to "Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)" on page 212.

Results

In addition to seeing the Calibre RealTime toolbar and menu selection, you should see these lines in the CIW:

```
Calibre server launched successfully
// Calibre RealTime embedded client initialized successfully at host:port
// Calibre RealTime: Initialization complete
```

Calibre RealTime creates the library MentorObserverLib. In general, it should not be necessary to do anything with this library. However, if you want to delete it when Calibre RealTime is not enabled, see "Deleting Mentor Libraries When Calibre RealTime is Disabled" on page 223.

Related Topics

Session Configuration File in Calibre RealTime Custom

Calibre RealTime Environment Variables

License Timeout for Calibre RealTime Custom

Setting the Calibre RealTime Version Newer than CALIBRE_HOME

Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)

A run configuration in Calibre RealTime is the set of options and input files for a run. You can define multiple run configurations with different options (such as the rule file, check recipe and other options), then quickly switch between different run configurations. You can run with one configuration (single mode) or run multiple configurations in sequence (serial mode).

Note

Run configurations were introduced in the 2016.2 release. The first time you open Calibre RealTime in a release after 2016.1 the "Run Mode" is set to Single and a run configuration named "Default configuration" is created based on the settings in your Calibre RealTime session configuration file (.realtime.cfg). The configuration file is updated to the 2016.2 syntax.

If you plan on using only one run configuration, no further action is needed.

Multiple run configurations can be used in the following modes:

- **Single** Run only the currently selected configuration when a run is started.
- **Serial** Run each configuration selected in the Configuration Run Control dialog box in order.

Prerequisites

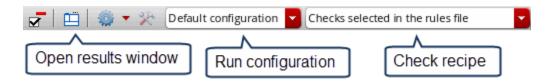
• You have completed "Invoking Cadence Virtuoso with Calibre RealTime" on page 209.

Procedure

- 1. Choose Calibre > RealTime DRC > Options to open the Calibre RealTime Options Dialog Box, or click the Options button () on the integrated toolbar.
- 2. If using multiple run configurations, define as many configuration names as needed:
 - a. Click the **Edit** button next to the configuration name to open the Configuration Run Control dialog box.
 - b. Click the "+" button to add a configuration and specify the name. Repeat this step as needed.
 - c. If using the Serial run mode, set the run order using the up and down arrows. Use the checkbox to include or remove a configuration from the execution list.
 - d. Click **OK** to close the Configuration Run Control dialog box.
- 3. Define the settings for each run configuration:
 - a. Select the configuration in the "Configuration" dropdown list.

- b. Specify the settings as described in "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Virtuoso)" on page 213.
- 4. Select the "Run mode":
 - **Single** Run only the currently selected configuration when a run is started.
 - **Serial** Run each configuration selected in the Configuration Run Control dialog box in order.
- 5. Proceed to "Running Calibre RealTime with Cadence Virtuoso" on page 215.

Results are displayed for separately for each run configuration—select the configuration to view with the toolbar selection:



Related Topics

SKILL Commands to Get and Set Calibre RealTime Options

Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Virtuoso)

Calibre RealTime uses a standard Calibre nmDRC rule file. The check recipe controls which checks in the rule file are executed during the Calibre RealTime run.

Prerequisites

- You have completed "Invoking Cadence Virtuoso with Calibre RealTime" on page 209.
- You have completed "Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)" on page 212.
- You have a Calibre nmDRC rule file available; see "Calibre RealTime Custom Requirements" on page 15 and "Differences Between Calibre nmDRC and Calibre RealTime Custom" on page 18 for additional information about rule file requirements.

Procedure

- 1. Choose **Calibre > RealTime DRC > Options** to open the Calibre RealTime Options dialog box, or click the **Options** button () on the integrated toolbar.
- 2. Select the configuration in the "Configuration" dropdown list.
- 3. Specify the Calibre rule file in the "Rule file" text field.

4. Click **Reload** to load the rule file.

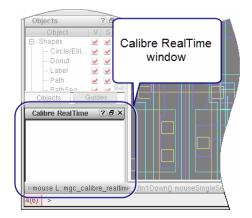
Note

The rule file is loaded automatically the *first* time it is needed. If you make changes to the rule file during a Calibre RealTime session, you must manually reload the rule file in order for the new rule file to be used.

The rule file for the active configuration is loaded automatically on RealTime startup.

After the rule file is loaded, the Calibre RealTime window appears in the Cadence Layout window, as shown at right. The Calibre RealTime window displays check text when "Include Check definition in tooltip" is enabled in the Calibre RealTime Options dialog box.

Note: You can turn off display of the Calibre RealTime window by disabling the option "Show check description window."



5. (Optional) Select options in the dialog box; the defaults are suitable for most runs. See "Calibre RealTime Options Dialog Box" on page 31 for a definition of each option.

If you want the Calibre RealTime Results window to open automatically after a run, enable the option "Launch RealTime results viewer automatically."

If you want to save a summary file of the run and the DRC results as an ASCII results database (RDB), select the **Output files** tab and check the corresponding options.

- 6. Select the check recipe as follows:
 - Checks selected in the rules file (default) This built-in recipe uses the Calibre rule file just as a batch run of Calibre does.
 - Other built-in check recipes You can select a recipe that only runs checks on certain layers, excludes certain types of checks, or runs all checks in the rule file.

Because these built-in recipes do *not* obey DRC [Un]Select Check statements in the rule file, the recipes may execute checks in the rule file that would not be executed during a batch Calibre run. The recipe "Checks selected in the rules file" is the only built-in recipe that obeys DRC [Un]Select Check statements. See "Built-In Check Recipes" on page 107.

• User recipes — You can select a previously defined custom check recipe. See "Basic Editing of Check Selection Recipes" on page 111.

Show check description window Check selection recipe: Checks selected in the rules file Edit... USER RECIPES metal checks no density Custom recipes Cancel BUILT-IN RECIPES All checks Visible layer checks These recipes Visible layer checks, no connectivity do not obey Visible layer only checks DRC [Un]Select Visible layer only checks, no connectivity Checks selected in the rules file statements. "Checks selected in the rules file" obeys DRC [Un]Select statements-the rule file is read the same as for a batch Calibre run.

Figure 7-2. Selecting a Check Recipe in Calibre RealTime (Virtuoso)

7. Proceed to "Running Calibre RealTime with Cadence Virtuoso" on page 215.

You can change environment variables with the commands given in the section "Environment Variable Control in Calibre RealTime for Cadence Virtuoso" on page 295. This is useful if you want to change environment variables used in the rule file.

Related Topics

Using Calibre Interactive Settings in a Calibre RealTime Custom Run

Check Selection Recipes in Calibre RealTime Custom

Calibre RealTime Custom Options Dialog Box (Virtuoso)

Database Read Setup for Calibre RealTime with Cadence Virtuoso

mgc_calibre_realtime_set_option

Running Calibre RealTime with Cadence Virtuoso

Calibre RealTime runs checks on the geometries specified by the DRC run type. The specified check recipe selects the checks that are executed. All Calibre RealTime nmDRC runs are performed in flat mode.

Prerequisites

 You have completed "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Virtuoso)" on page 213.

Procedure

- 1. Select a run configuration in the "Configuration" dropdown list, or, if running in Serial mode, make sure the "Run Mode" is set to Serial in the Calibre RealTime Options dialog box. See "Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)" on page 212.
- 2. Select the DRC run type and start a run.

The DRC run type is selected in the toolbar, in a dropdown menu next to the

button:





DRC Run Type	Instructions
Window	Runs DRC on the geometries visible in the design tool window:
	1. Choose the Window run type.
	2. Zoom to the region you want to run DRC on and make sure all the geometries you want to run checks on are visible. 25 um x 25 um or smaller is recommended for the best performance.
	3. Click the vun.
Area	Runs DRC on a selected region:
	1. Choose the Area run type.
	2. Click the 🗸 button.
	3. Use the mouse to drag a rectangle around the area you want to run DRC on.
Cell	Runs DRC on the whole cell that is open in the active window.
	1. Choose the Cell run type.
	2. Click the vun.
Area 1, Area 2,	Runs DRC on a previously selected area:
	1. Choose the area to run on. The most recent area is always named Area 1.
	2. Click the vun.

If running in Serial mode, a DRC run is executed for each of the configurations selected in the "Configuration Run Control" dialog box, in the order specified.

Results

The results of a Calibre RealTime run are shown in Figure 7-3. You can use the highlight controls in the integrated toolbar to step through results. If you are using multiple run configurations, the results are displayed separately for each configuration—select the configuration to view in the toolbar dropdown box.

If error markers overlap, all results are highlighted at once by default. You can change this behavior with the SKILL command mgc_calibre_realtime_set_show_single_error.

You can also click the toolbar icon to open the Calibre RealTime Results window to view results in a separate window; see "Viewing Results in the Calibre RealTime Results Window" on page 83.

Rule check comments are shown in the Calibre RealTime window; the check definition is also included if "Include Check Definition in Tool Tip" was enabled at the time of the run. You can turn off display of the Calibre RealTime window by disabling the setting "Show check description window" in the Calibre RealTime Options dialog box.

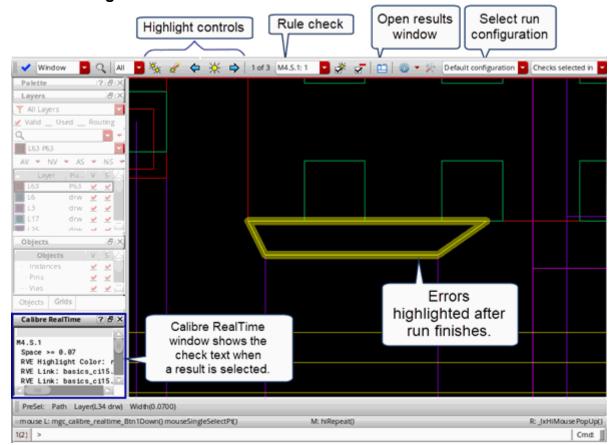


Figure 7-3. Calibre RealTime Results in Cadence Virtuoso

Starting a Batch Calibre Run from Calibre RealTime

You can start a batch Calibre run from Calibre RealTime if the design window contains more than three million polygons. The controls for starting a batch run are on the **Batch** tab in the Calibre RealTime Options dialog box.

__ Tip _

If the option to start a batch run is not enabled as described in the following procedure, a pop-up dialog with a prompt to start a batch run is displayed if the geometry count reaches three million polygons. Reply "Yes" to start a batch run, or reply "No" to end the run.

Prerequisites

- You have completed "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (Virtuoso)" on page 213.
- Run configurations are set up. See "Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)" on page 212.
- (Optional) Set the environment variable MGC_REALTIME_DO_NOT_HIGHLIGHT_BATCH_RESULTS if you do not want results from the batch run automatically highlighted when the run completes. If the run produces many results, the highlight process can be slow. The option "Highlight after DRC run" is ignored when the environment variable is set.

Procedure

- 1. Click the **Options** button () in the integrated toolbar to open the Calibre RealTime Custom Options Dialog Box (Virtuoso).
- 2. Select the desired run configuration in the "Configuration" dropdown list.
- 3. Click the **Batch** tab and set options as follows:
 - a. Check one of the following options:

 - o Run batch Calibre nmDRC on layouts containing more than 3 million polygons
 - b. Select settings for "Run Calibre DRC" and "Performance."
 - c. (Optional) If you are submitting the job to a remote cluster, check "Queue command" and enter the command. See the tooltip for replaceable parameters.
 - See "Batch Queue Command" on page 240 for requirements regarding the batch queue command and a sample command script suitable for testing.
 - d. (Optional) Specify the "Batch working directory." The directory must exist and have write permission. Specify a different working directory for each run configuration.

The wrapper rule file, transcript file, and results database (RDB) are written to the batch working directory. If you do not specify a batch working directory, the output is written to the directory from which you invoked the design tool.

e. Set options for "Licensing mode."

See "Batch Tab in the Calibre RealTime Options Dialog Box" on page 238 for details on the Calibre run and licensing options.

- 4. If you are using multiple run configurations, repeat Steps 2 and 3 for each configuration.
- 5. Click **OK** to close the options dialog box.
- 6. Select your run type and click \checkmark (**Run DRC** button) in the toolbar to start the run.

When a batch run starts, the **Run DRC** button changes to (Stop batch DRC jobs).

Results

The run starts as a Calibre RealTime DRC run. If the polygon count for the run is greater than the specified number, a batch run starts and a message similar to the following is printed:

```
Calibre RealTime is running batch DRC using configuration: "config_DP" DRC analysis window: {-382049 -1173620} {1524120 141969} Batch Calibre transcript file: "./batch_dir/
RealTime_batch_username_45nm_design_demo_fir_multi_layout_transcript.log"
DRC COMPLETE. TOTAL CHECK COUNT = 158 TOTAL RESULT COUNT = 86
Timing: Total run time : 4,498 ms
```

You can use the design tool at this point, but the Calibre RealTime toolbar is unavailable except for the button. The Calibre RealTime license is checked out for the duration of the batch run.

When the batch run is complete, the results are reported in the Calibre RealTime toolbar and it becomes active again. You can now use the highlight controls, or click the toolbar icon to open the Calibre RealTime Results window to view results in a separate window; see "Viewing Results in the Calibre RealTime Results Window" on page 83. (The option "Launch RealTime results viewer automatically" is ignored in batch runs.)

A message similar to the following is printed to the CIW when the run completes:

```
DRC COMPLETE. TOTAL CHECK COUNT = 1631 TOTAL RESULT COUNT = 24990
```

The following files are written to the batch working directory:

- RealTime_batch_<*user*>_cfg<*configuration*>_<*library*>_<*cell*>_<*view*>_results.rdb
 The results database (RDB) file.
- RealTime_batch_<*user*>_cfg<*configuration*>_<*library*>_<*cell*>_<*view*>_transcript.log
 The Calibre transcript.
- RealTime_batch_<*user*>_cfg<*configuration*>_<*library*>_<*cell*>_<*view*>_wrapper.svrf

The "wrapper" rule file is used in the Calibre command line that starts the batch run. The wrapper rule file specifies the layout window and the checks to execute, along with other settings for the run. The rule file for a run configuration is specified with an SVRF Include statement.

Specifying Highlight Color and Layer Visibility for Calibre RealTime (Virtuoso)

You can use a Check Text Override (CTO) file to assign the highlight color and layer visibility for results according to the rule check. The Check Text Override file is specified in the Calibre RealTime Options dialog box.

Note.

For setting the highlight color, you can also choose **Calibre > Setup > RVE** and specify the highlight layers if you do not want to use a CTO file. This is similar to the case when using Calibre RVE to highlight results.

Procedure

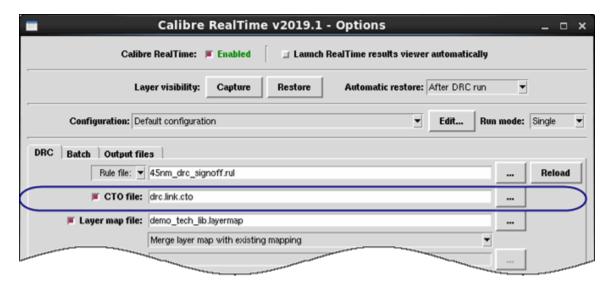
1. Create a Check Text Override (CTO) File for Calibre RealTime using a text editor. The following example CTO file specifies the highlight color and layer visibility for the min_ext_diff_poly rule:

```
# check text override file for XYZ
#
min_ext_diff_poly
RVE Highlight Color: blue
RVE Show Layers: diff poly
```

where the drawing purpose layer is used by default in the RVE Show Layers statement. See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90 for complete information about the CTO file syntax and supported rule check comments.

- 2. Save the check text override file with a file extension of .cto.
- 3. In Cadence Virtuoso, choose **Calibre > Setup > RVE**, then enable "Allow custom colors."
- 4. Click the button in the Calibre RealTime toolbar to open the Calibre RealTime Options dialog box.
- 5. Enable "CTO file" and specify a check text override file.

The .cto file extension is not required, but the file browser for the CTO file searches for files with a .cto file extension.



Tip

You can enter an environment variable that is a directory path in the text entry field and click the ... button to have the file browser open to that directory.

You can also specify the CTO file with the environment variable MGC_REALTIME_CTO_FILE; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

- 6. (Optional) If using the RVE Show Layers rule check comment, configure the "Automatic restore" setting to specify when to restore the layer visibility from before the first DRC run:
 - No Do not automatically restore layer visibility
 - After DRC run Restore layer visibility after a DRC run
 - After clear highlight Restore layer visibility after highlights are cleared

Results

With the example shown in Step 1, all results from the rule check min_ext_diff_poly are highlighted in blue. Layers diff and poly are shown and all other layers are hidden.

Note:

If you highlight results from a rule check with a RVE Show Layers rule check comment, and next highlight results from a rule check *without* a RVE Show Layers rule check comment, the layer visibility set with the first rule check is retained. You can use the controls in your design tool to set the layer visibility as desired.

Related Topics

mgc_calibre_realtime_set_option

Edit-in-Place, Descend, and Ascend in Calibre RealTime with Cadence Virtuoso

There are several points to keep in mind when you do a descend or ascend in Cadence Virtuoso or use Edit-In-Place (EIP) to fix errors found with Calibre RealTime.

Edit-In-Place

- When you descend to a lower level cell in EIP mode, Calibre RealTime retains the current context—you still see the error highlights from the upper level and the same listing of results in the Calibre RealTime toolbar.
- After you fix the error in the lower level cell, you can click the **Run DRC** () button to run in the window. The run takes place in the current context, which includes the level being edited up to the starting viewing level. In some cases a change in a lower level cell can cause errors at the upper levels.

Note: Cell instantiations outside the context of the current window are not checked.

Ascend, Descend, and Open

When you change cells with descend, ascend, or open in Cadence Virtuoso, Calibre RealTime can retrieve the most recent DRC run results for the cell of interest and display them in the toolbar. This result retrieval takes additional time, so you may want to disable it. You can control the result retrieval with the following setting in the Calibre RealTime Options dialog box:

• **Retrieve previous results** — Controls whether previous results for the cell of interest are retrieved and displayed in the toolbar. The setting is disabled by default.

When this setting is disabled, the results section of the toolbar displays "0 of 0" and no check names after a descend, ascend, or open.

You can also disable the results retrieval by defining the environment variable MGC_CALIBRE_REALTIME_NO_DRC_RESULT_RETRIEVAL.

Related Topics

Running Calibre RealTime with Cadence Virtuoso

Deleting Mentor Libraries When Calibre RealTime is Disabled

You can set an environment variable to instruct Calibre RealTime to delete the MentorMessengerLib and MentorObserverLib entries from the *cds.lib* file if Calibre RealTime is disabled when you exit Cadence Virtuoso.

Prerequisites

• "Requirements and Setup for Calibre RealTime with Cadence Virtuoso" on page 202 is completed.

Procedure

- Define the environment variable MGC_CALIBRE_REALTIME_VIRTUOSO_REMOVE_LIBS_ON_EXIT before starting Cadence Virtuoso.
- 2. Start Cadence Virtuoso.
- 3. Use Calibre RealTime as desired.
- 4. Disable Calibre RealTime as follows:
 - a. Click the button in the Calibre RealTime toolbar to open the Calibre RealTime Options dialog box.
 - b. Disable the "Calibre RealTime: Enabled" checkbox at the top of the dialog box.
 - c. Click **OK** to close the dialog box.

The Calibre RealTime toolbar is no longer present.

5. Exit Cadence Virtuoso.

Results

The MentorMessengerLib and MentorObserverLib entries are removed from the *cds.lib* file.

Related Topics

Saving Mentor Libraries to a Specified cds.lib

Invoking and Running Calibre RealTime with Cadence Virtuoso

Environment Variables in Calibre RealTime Custom

Saving Mentor Libraries to a Specified cds.lib

You set the environment variable MGC_REALTIME_FORCED_LIB_LOCATION to specify a non-default *cds.lib* in which to save the Mentor libraries. Calibre creates libraries that are used internally and it can be useful to save these in a different location than your default *cds.lib*.

For example, if your Cadence Virtuoso environment uses version control software, you may get pop-up messages each time Calibre RealTime saves the Mentor libraries to *cds.lib*. You can set MGC_REALTIME_FORCED_LIB_LOCATION to point to a different *cds.lib* file (one that is not under version control) to prevent this problem.

Prerequisites

• A valid extra *cds.lib* file, which must be included in the master *cds.lib* file and must have write permissions.

Procedure

1. Define the environment variable MGC_REALTIME_FORCED_LIB_LOCATION to point to the extra *cds.lib* file, as follows:

```
setenv MGC REALTIME FORCED LIB LOCATION mgc cds.lib
```

- 2. Add the extra *cds.lib* file defined with MGC_REALTIME_FORCED_LIB_LOCATION to your master *cds.lib*. Also make sure the master *cds.lib* does not include MentorObserverLib or MentorMessengerLib.
- 3. Invoke Cadence Virtuoso with Calibre RealTime.

Results

The Mentor libraries are saved to the *cds.lib* specified with MGC_REALTIME_FORCED_LIB_LOCATION.

Related Topics

Deleting Mentor Libraries When Calibre RealTime is Disabled Invoking and Running Calibre RealTime with Cadence Virtuoso Environment Variables in Calibre RealTime Custom

Toolbars, Menus, and Dialog Boxes in Calibre RealTime (Virtuoso)

Toolbars, Menus, and Dialog Boxes in Calibre RealTime (Virtuoso)

The Calibre RealTime integrated toolbar and the Calibre RealTime Options dialog box provide the controls and settings you need to run Calibre RealTime. Some frequently accessed options are provided in the DRC Run and Highlight Options dropdown menu.

Calibre RealTime Integrated Toolbar (Virtuoso)	225
Calibre RealTime DRC Menu (Virtuoso)	228
DRC Run and Highlight Options Menu (Virtuoso)	229
Calibre RealTime Custom Options Dialog Box (Virtuoso)	230
Batch Tab in the Calibre RealTime Options Dialog Box	238

Calibre RealTime Integrated Toolbar (Virtuoso)

The Calibre RealTime integrated toolbar includes controls to start a DRC run, highlight results, and open the Calibre RealTime Options dialog box. The toolbar also includes option settings for run control and highlighting.

The Calibre RealTime integrated toolbar is shown in the following figure.



The run controls, highlight controls, and options controls are described in the following sections.

Note_

For 2015.1 and earlier releases, the toolbar included a dropdown list of the ten most recent run contexts. Beginning with the 2015.2 release the run context is no longer displayed by default. Define the environment variable

MGC_CALIBRE_REALTIME_SHOW_CONTEXT_IN_TOOLBAR to restore the run context display.

Run Controls (Virtuoso)

The run controls in the Calibre RealTime integrated toolbar start a window run and zoom to the window for the most recent DRC run.

The run controls in the Calibre RealTime integrated toolbar are shown below and described in Table 7-2.

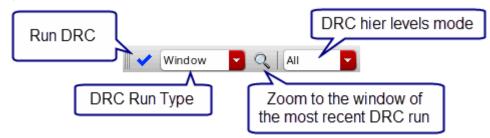


Table 7-2. Run Control Toolbar Selections (Virtuoso)

Toolbar item	Description
(Run DRC)	Runs flat DRC using the selected DRC Run Type. Calibre runs with the currently selected Run Configuration and configuration run mode (Serial or Single).
	Keyboard shortcut: F12
DRC Run Type	Selects the DRC run type:
	• Window — Run DRC on the geometries visible in the current window, plus a halo region around the area.
	• Area — Run DRC in a selected area. Click the Run DRC
	button () and use the mouse cursor to select an area in the design.
	• Cell — Run DRC in the whole block that is open in the active window.
	• Area 1, Area 2, — Run DRC on a previously selected area.
Q	Zoom to the window of the most recent DRC run.
DRC hier levels mode	Specifies the DRC Hierarchical Levels Mode:
	All — Check all hierarchy levels in the design.
	Displayed — Check only the hierarchy levels that are displayed in Cadence Virtuoso.

Highlight Controls (Virtuoso)

The highlight control section the Calibre RealTime integrated toolbar includes highlight controls and displays the rule check for the current result.

The highlight controls in the Calibre RealTime integrated toolbar are shown below and described in Table 7-3.

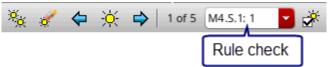


Table 7-3. Highlight Control Toolbar Selections (Virtuoso)

Toolbar item	Description
×.	Highlight all.
	Clear all highlights.
⇔ 🔆 💠	Highlight previous, current, or next result for the rule check.
	Also see Behavior of "Group highlight colors by check" in "Calibre RealTime Options Dialog Box" on page 31.
1 of 5	Result <i>n</i> of <i>total</i> results.
	Also see the setting "Show results in" in the Calibre RealTime Options Dialog Box; this setting determines whether all results for the cell or only results from the most recent run are reported.
Rule check	Indicates the rule check which produced the displayed error when using the highlight previous, current, and next buttons. Use the dropdown arrow to select a different rule check.
ॐ	Highlight all results from the selected rule check.

Option Controls (Virtuoso)

The option controls in the Calibre RealTime integrated toolbar select the check recipe and open the Calibre RealTime Options dialog box.

The option controls in the Calibre RealTime integrated toolbar are described in Table 7-4.



Table 7-4. Option Control Toolbar Selections (Virtuoso)

Toolbar item	Description
✓	Remove the displayed rule check from the current check recipe. All highlights are cleared, regardless of the setting of "Clear Existing Highlights" in the options menu. If the recipe is a built-in recipe, you are prompted to save the new check recipe to a new filename; if the recipe is a user recipe, the updated recipe is saved automatically.

Table 7-4. Option Control Toolbar Selections (Virtuoso) (cont.)

Toolbar item	Description
	Open the Calibre RealTime Results window. See "Viewing Results in the Calibre RealTime Results Window" on page 83.
Options dropdown menu 💮 💌	Open the DRC Run and Highlight Options Menu (Virtuoso).
©	Open the Calibre RealTime Options Dialog Box.
*	Open the Customization GUI. This button is only available if a Calibre Interactive runset is loaded and includes a customization file. See "Using the Customization GUI with Calibre RealTime" on page 88.
Run Configuration	Selects the run configuration, both for a run in "Single" mode and for viewing results; see "Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)" on page 212.
Check recipe	Select the check recipe from the dropdown list; the check recipe specifies the rule checks executed for all Calibre RealTime runs. See "Check Selection Recipes in Calibre RealTime Custom" on page 105 for more information.

Calibre RealTime DRC Menu (Virtuoso)

The Calibre RealTime integration adds the **RealTime DRC** menu item the **Calibre** menu in the main Cadence Virtuoso menu bar.

The **RealTime DRC** menu has the entries described in the following table:

Table 7-5. RealTime DRC Menu Contents (Virtuoso)

Menu Item	Description
Run DRC	Starts a flat Calibre nmDRC run using the DRC Run type selected in the RealTime integrated toolbar. Calibe runs with the currently selected Run Configuration and configuration run mode (Serial or Single).
Options	Opens the Calibre RealTime Options Dialog Box.

DRC Run and Highlight Options Menu (Virtuoso)

Certain frequently used Calibre RealTime settings are available in the DRC Run and Highlight Options dropdown menu.

Click the down arrow () in the integrated toolbar to open the DRC Run and Highlight Options menu. The settings in the DRC Run and Highlight Options menu are saved in the Calibre RealTime Configuration File (Virtuoso). The menu items are described in the following table:

Table 7-6. DRC Run and Highlight Options Menu Contents (Virtuoso)

Menu Item	Description
Zoom to Highlights	Zoom to result when highlighting.
Clear Existing Highlights	Clear existing highlights before each new highlight action. Highlights are cleared before each run, and when using the
Include Check Definition in Tool Tip	buttons to step though highlights. When enabled, the complete rule check definition is included in the result tool tip; otherwise, only the check text comments are shown.
_	You must run Calibre RealTime after changing this setting for the change to take effect.

Related Topics

Environment Setup for Cadence Virtuoso with Calibre RealTime

Calibre RealTime Custom Options Dialog Box (Virtuoso)

To access: Choose **Calibre > RealTime DRC > Options** or click the **button** in the toolbar.

The Calibre RealTime Options dialog box specifies the Calibre rule file, the check recipe, and other options. The settings in the Calibre RealTime Options dialog box are saved in the Calibre RealTime Configuration File.

Description

Figure 7-4. Calibre RealTime Options Dialog Box (Virtuoso)

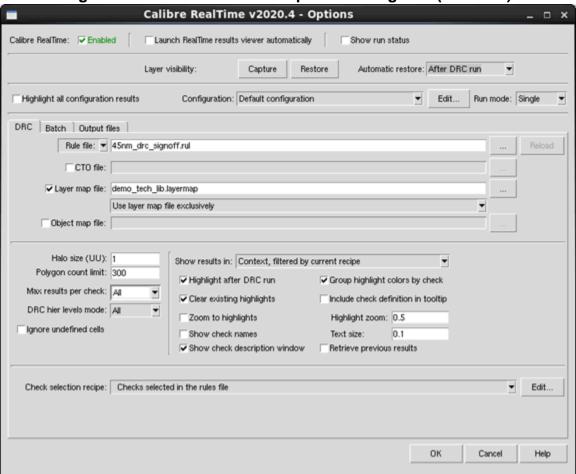


Table 7-7. Tabs in the Calibre RealTime Options Dialog Box (Virtuoso)

Tab	Description
DRC	Specify options for the Calibre nmDRC run. See the table in the next section.

Table 7-7. Tabs in the Calibre RealTime Options Dialog Box (Virtuoso) (cont.)

Tab	Description
Batch (Only available in Cadence Virtuoso)	Specify options to launch a batch Calibre nmDRC run if the design window contains more than a specified number of polygons. See "Starting a Batch Calibre Run from Calibre RealTime" on page 218.
Output files	Specify to save a summary file and/or to save the DRC results to an ASCII results database (RDB). If a specified directory does not exist, the tool prompts you to confirm that the directory should be created.

Fields

Table 7-8. Calibre RealTime Options Dialog Box Contents (Virtuoso)

Control	Description
Calibre RealTime	Toggles display of the Calibre RealTime toolbar.
(Enabled/Enable)	The Calibre RealTime toolbar is automatically displayed if a DRC run is launched using Calibre > RealTime DRC > Run DRC in Window.
	Also see the environment variable MGC_REALTIME_DISABLE in "Calibre RealTime Environment Variables" on page 333.
Launch RealTime results viewer automatically	Specifies whether to open the Calibre RealTime Results Window automatically after a run. See "Viewing Results in the Calibre RealTime Results Window" on page 83.
Show run status	Specifies whether to display processing status updates in the shell during a Calibre RealTime run.
Layer visibility	Manually captures or restores layer visibility.
	When using a CTO file, layer visibility changes after highlighting a result from a check with a RVE Show Layers rule check comment. The following buttons enable you to save and set layer visibility:
	• Capture — Save the current layer visibility
	• Restore — Revert the layer visibility to the saved version
Automatic restore	Specify when to set the layer visibility to the saved version. If the layer visibility has not been manually saved, then the version before the first DRC run is used.
	• No — Do not automatically restore layer visibility
	• After DRC run —Restore layer visibility after a DRC run
	After clear highlight — Restore layer visibility after highlights are cleared

Table 7-8. Calibre RealTime Options Dialog Box Contents (Virtuoso) (cont.)

Control	Description
Highlight all configuration results	Specifies whether to automatically highlight DRC results from all configurations after a DRC run. When this option is enabled:
	 Highlights are preserved when switching configurations. When highlighting, the "Clear existing highlights" option is ignored and existing highlights are not cleared. Results from all configurations are highlighted in both single and serial runs, and the "Highlight after DRC" option is ignored.
	Note: When this option is enabled, you can only crosshighlight results from the layout editor to the Results window if they are from the current configuration. You can view the check description of any result regardless of its corresponding configuration.
Configuration	Specifies the configuration name. To add or delete a configuration, click Edit to open the Configuration Run Control dialog box.
	A configuration is the set of options and filenames below the configuration name.
Run Mode	 Single — Run with the currently selected configuration. Serial — Run each selected configuration in series, in the order specified in the Configuration Run Control dialog box.
Rule file/Runset dropdown list	Selects whether to load a rule file or a Calibre Interactive runset.
Rule file	Specifies the Calibre rule file used for the run.
Runset	Specifies a Calibre Interactive runset to use for the run. See "Using Calibre Interactive Settings in a Calibre RealTime Custom Run" on page 87 for details.
CTO file	Specifies a CTO file for controlling the highlight color and layer visibility per rule check. See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90 and "Specifying Highlight Color and Layer Visibility for Calibre RealTime (Virtuoso)" on page 220.

Table 7-8. Calibre RealTime Options Dialog Box Contents (Virtuoso) (cont.)

Control	Description
Layer map file	Specifies a custom layer map file.
	Dropdown list options:
	• Merge layer map with existing mapping — (default) Merge layer map data from the specified file with the layer map derived from the technology file. Layer map entries in the layer map file override the corresponding layer map entry based on the technology file; all other layer map entries based on the technology file remain valid.
	• Use layer map file exclusively— Use only layer map data from the specified file. Layer map data derived from the technology file is <i>not</i> used.
	See "Specifying the Layer Map for Calibre RealTime with Cadence Virtuoso" on page 244.
Object map file	Specifies an object map file. See "Object Map File Format in Calibre RealTime" on page 40.
(Browse)	Open a file browser to select the rule file, CTO file, or layer map file.
Reload	Reload the rule file or runset.
	You must click Reload for changes in the rule file or runset to take effect. Changes in the layer map, object map, and CTO files cause an automatic reload prior to the DRC run.
Halo size (UU)	The halo size in user units. See "Halos in Calibre RealTime Custom" on page 24 and "Halo" in "Terms and Definitions in Calibre RealTime Custom" on page 19 for more information on halos.
Polygon count limit	Sets the maximum number of polygons that are passed to Calibre. The limit should be set low enough that the Calibre server returns control to the design tool in a short amount of time.
	The maximum number of geometries Calibre RealTime will process is 3,000,000.
Max results per check	Specifies the maximum number of results returned per rule check; this setting overrides the setting in the rule file.

Table 7-8. Calibre RealTime Options Dialog Box Contents (Virtuoso) (cont.)

Control	Description
DRC hier levels mode	Specifies the hierarchy levels in the design that are checked during the run.
	• All — Check all hierarchy levels in the design.
	• Displayed — Check only the hierarchy levels that are displayed in Cadence Virtuoso.
	Depending on the selected rule checks, some rule check results may occur in the "Displayed" mode due to the fact that not all levels of the design are checked.
Ignore undefined cells	When checked, specifies for the Calibre nmDRC run to issue a warning and continue if undefined cells are found in the design.
Show results in	This setting controls which results are reported in the integrated toolbar after the run.
	• Whole cell, all checks — Report all results for the cell in the integrated toolbar after each DRC run. Reported results include those from the current run, previous results in the cell that are outside of the current run context, and possibly previous results within the current run context from rules checks not selected by the current recipe.
	• Whole cell, filtered by current recipe — Report results for the whole cell and from the current recipe only in the integrated toolbar after each DRC run. Reported results include those from the current run plus previous results in the cell that are outside of the current run context.
	• Context, filtered by current recipe — (default) Report only results within the current run context and from the current check recipe in the integrated toolbar after each DRC run.
	For each case, all results in the current window and belonging to the current check selection recipe are deleted before the run.
	The settings "Whole cell, filtered by current recipe" and "Context, filtered by current recipe" should be used with caution, since they filter out results from previous runs. If these settings are used, it is recommended that a later run be performed with "Whole cell, all checks" selected to make sure that all results are reported.
Highlight after DRC run	Highlight results in the viewer after the DRC run.

Table 7-8. Calibre RealTime Options Dialog Box Contents (Virtuoso) (cont.)

Control	Description
Group highlight colors by	Assign the highlight color for results according to the rule check.
check	The assigned highlight color for a rule check is not fixed—rather, the highlight colors are assigned to checks in a repeated cycle, starting with the color red. See "Usage Notes" for details.
	Also see "Specifying Highlight Color and Layer Visibility for Calibre RealTime (Virtuoso)" on page 220.
Clear existing highlights	Clear existing highlights before each new highlight action.
	Highlights are cleared before each run, and when using the
	Also see Behavior of "Group highlight layers by check" in the Usage Notes section.
Include check definition in tool-tip	When enabled, the complete rule check definition is included in the result tool tip; otherwise, only the check text comments are shown.
	You must run Calibre RealTime after changing this setting for the change to take effect.
Zoom to highlights	Specifies whether to zoom to the extent of the results when highlighting.
	When automatically highlighting after a DRC run, RealTime does not zoom to highlights to avoid changing the currently selected viewpoint.
Highlight zoom	Specifies the magnification used when zooming to highlight a result.
Show check names	Specifies to display the check name at the center of each result.
Text size	Specify the text size when "Show check names" is enabled.
Show check description window	Specifies to display the check description window when highlighting.
Retrieve previous results	Specifies whether to retrieve previous results for the cell of interest and display them in the toolbar after changing cells with an open, descend, or ascend action. Retrieval of previous results takes additional time.
	You can disable the option by defining the environment variable MGC_CALIBRE_REALTIME_NO_DRC_RESULT_RETRIEV AL.
Check selection recipe	The check recipe used for the Calibre run. See "Check Selection Recipes in Calibre RealTime Custom" on page 105.
Edit	Open the Calibre RealTime Recipe Editor Dialog Box.

Table 7-8. Calibre RealTime Options Dialog Box Contents (Virtuoso) (cont.)

Control	Description
Help	Opens the <i>Calibre RealTime User's Manual</i> in your default web browser.

Usage Notes

• Environment Variable Usage

- You can use environment variables in pathnames for files and directories. The environment variable is not expanded when viewed in the dialog box or saved to the configuration file. For example, you can enter \$CAL_RULES/drc.rul for the rule file, and \$CAL_RULES/drc.rul is saved in the configuration file.
- You can enter an environment variable for a directory path in a file entry field and click the ... button to open the file browser to that directory. This makes it easier to browse for a file if it is not in the working directory and the directory path is long. When you select a file, the absolute path is entered in the text entry file and saved in the configuration file. This capability is available for the rule file, CTO file, layer map file, and object map file, but not for the runset file.
- There are system defined environment for the input files; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

• Behavior of "Group highlight colors by check"

When "Group highlight colors by check" is enabled, colors are assigned to results according to the rule check. However, the association between the highlight color and the rule check is not fixed; it depends on the history of your Calibre RealTime session. Highlight colors are assigned to checks in a repeated cycle, starting with the color red. The color assignment starts at the beginning (with red) each time highlights are cleared. Highlights are cleared in the following cases:

- o Automatically when "Clear Existing Highlights" in the options menu is enabled; highlights are cleared at the start of a run, or at each new highlight when using the

 → buttons.
- o By clicking the **Clear Highlights** button (ℯ⁄).
- o Automatically when the button in the integrated toolbar is clicked. The button deletes a rule check from the current check recipe and clears all highlights.

The following behavior is observed when "Group highlight colors by check" is enabled:

o Highlight color assignments from a DRC run:

"Clear Existing Highlights" ON — Highlights are cleared and highlight color assignment starts at red at the beginning of each Calibre RealTime run.

"Clear Existing Highlights" OFF — Highlights from previous runs are kept. New results from a check with an existing highlighted result are highlighted in the same color. New results from a check that does not have an existing highlighted result are highlighted in the next color in the highlight color cycle.

Clicking or — Highlights are cleared and highlight color assignment starts at red for the next run, regardless of the setting of "Clear Existing Highlights."

Highlight color assignments using the buttons

"Clear Existing Highlights" ON — All highlights are cleared each time one of the

→ → buttons is clicked; each result is highlighted in red.

"Clear Existing Highlights" OFF — Each result is highlighted in the color used originally.

Clear Highlights button clicked when "Clear Existing Highlights" is OFF—
All highlights are cleared. The next result highlighted with one of the buttons is highlighted in red; the highlight color for further results is assigned according to the rule check. All highlights are kept until is clicked again.

Related Topics

SKILL Commands to Get and Set Calibre RealTime Options

Calibre RealTime Configuration File (Virtuoso)

Environment Setup for Cadence Virtuoso with Calibre RealTime

Hiding Built-In Check Recipes

Batch Tab in the Calibre RealTime Options Dialog Box

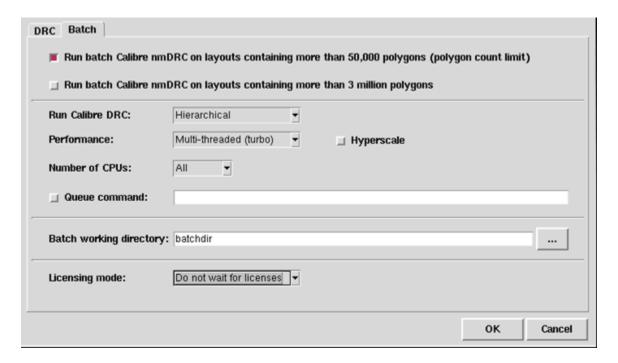
To access: Open the Calibre RealTime Options dialog box and click the **Batch** tab.

The **Batch** tab has settings for controlling a Calibre batch run started from Calibre RealTime. You can set the number of polygons in the layout window that triggers a batch run, and execution and licensing options.

Note_



Also see "Starting a Batch Calibre Run from Calibre RealTime" on page 218.



Objects

Object	Description
Run batch Calibre nmDRC on layouts containing more than <polygon_count_limit> polygons (Polygon count limit)</polygon_count_limit>	As described. The <i><polygon_count_limit></polygon_count_limit></i> is set on the DRC tab with the "Polygon count limit" option. The default polygon count limit is 50,000.
Run batch Calibre nmDRC on layouts containing more than 3 million polygons	As described.
Run Calibre DRC	Specify a Flat or Hierarchical run.
Performance (only available for hierarchical runs)	Specify a single-threaded, multithreaded (turbo), or distributed (MTflex) run.

Object	Description			
Hyperscale	Enables hyperscaling.			
(only available for multithreaded and MTflex runs)				
Number of CPUs	• All — Use the maximum number of available			
(only available for multithreaded runs)	CPUs.Positive integer — Use the specified number of CPUs.			
Remote file	Specify the remote file.			
(only available for MTflex runs)				
Queue command	Specify a command to submit the Calibre batch run to a remote cluster. If the queue command is specified with a relative pathname, the path is relative to the "Batch working directory" setting.			
	See "Batch Queue Command" on page 240 for replaceable parameters that can be used and for requirements regarding the queue command.			
Batch working directory	Specify the directory in which to save the results database, log file, and "wrapper" rule file. Temporary working files are also written to this directory. The directory must exist.			
	If not specified, the directory from which the design tool was launched is used.			
Licensing mode	Specify licensing options. The corresponding command line options is given in parentheses.			
	• Do not wait for licenses — Stop the batch run immediately if licenses are not available. (-nowait)			
	• Wait for licenses — If licenses do not become available in the specified time, attempt to acquire substitute licenses. (-wait < time>)			
	• Retry license acquisition — If licenses are not available, queue and periodically try to acquire the licenses. (-lmretry loop)			
	• Retry for: minutes (MAXRETRY)			
	Wait for: seconds between retries (INTERVAL)			
	 Retry even if a licensing error occurs (RETRY_ON_ERROR) 			

Usage Notes

Batch Queue Command

There are some special requirements on the queue command script executed from Calibre RealTime. The queue command script must do the following:

- Monitor the status of the remote job and not exit until the job is complete.
- Make any required environment variables available to the remote host.
- Launch Calibre and redirect the Calibre transcript output to a file. The following replaceable parameters can be passed to the queue command script:
 - o %c Path to the Calibre executable
 - o %o Calibre invocation options
 - o %t Calibre transcript file, which has the automatically generated name of RealTime_batch_<user>_<lib>_<cell>_<view>_transcript.log.
 - o %C Calibre command with options and redirect to transcript (%c %o > %t)
 - %d Batch working directory

The queue command script must use either the three parameters %c, %o, and %t, or the parameter %C. When %C is not used, the %t parameter must be used as the file to which the Calibre transcript output is redirected. The %d parameter is optional.

The (Stop batch DRC jobs) button is not available during execution of the queue command script, as there is no way for Calibre RealTime to terminate a job on a remote host.

Sample Batch Queue Command

The following queue command script demonstrates the use of the replaceable parameters and how to write the output of queue command script to a log file. The script prints some debug information and starts a Calibre run on the local host—it does not launch a job on a remote host.

The queue command script, *queue_command_sample*, is the following:

```
#!/usr/bin/tclsh
puts "argc=$argc"
puts "argv=$argv"

set calibre_binary    [lindex $argv 0]
set calibre_options    [lindex $argv 1]
set calibre_transcript [lindex $argv 2]

puts "calibre_binary=$calibre_binary"
puts "calibre_options=$calibre_options"
puts "calibre_transcript=$calibre_transcript"
puts ""

eval exec "$calibre_binary $calibre_options > $calibre_transcript"
```

To use this queue command place the following in the "Queue command" field on the Batch tab of the Calibre RealTime Options dialog box:

```
./queue_command_sample %c "%o" %t >& log.txt
```

Quotes are placed around the %0 parameter so that the Calibre options are passed to the script as a single string. After execution, the *log.txt* file contains the output of the puts statements and the transcript file contains the output from the Calibre command execution.

Database Read Setup for Calibre RealTime with Cadence Virtuoso

You can customize the database read process by specifying layer map and object map files. You can also specify the number of vertices in the polygon used to approximate a circle or ellipse.

GDS Layer Map for Calibre RealTime with Cadence Virtuoso	242
Specifying the Layer Map for Calibre RealTime with Cadence Virtuoso	244
Specifying the Object Map for Calibre RealTime with Cadence Virtuoso	246
Approximation of Circles and Ellipses in Calibre RealTime with Cadence Virtuoso .	247
Multi-Patterning Support in Calibre RealTime Custom in Cadence Virtuoso	247

GDS Layer Map for Calibre RealTime with Cadence Virtuoso

A layer map file is required to map the Cadence Virtuoso layer and purpose pairs to GDS layers and datatypes for layout export. The layer map file can be read automatically from the technology library, specified with an environment variable, or specified in the Calibre RealTime Options dialog box.

- **In the technology library** The layer map file is read automatically if it is located in the technology library directory and has the name *tech_lib.layermap*, where *tech_lib* is the name of the technology library.
- With an environment variable Set the environment variable MGC_RVE_GDS_LAYER_MAP_FILE to the location of the layer map file. Information in the layer map file overrides the automatic layer map located in the technology library.

The environment variable MGC_REALTIME_LAYER_MAP_FILE can also be used; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

• In the Calibre RealTime Options dialog box — See the procedure "Specifying the Layer Map for Calibre RealTime with Cadence Virtuoso" on page 244.

File Format

The layer map file has the following format:

• One entry per line with the following syntax:

```
OA layer OA purpose GDS layer num GDS datatype [CUT [CUTSIZE:x:y] | pin]
```

• ASCII text file.

• Comments start with # .; (semicolon), or // characters. All characters following the comment characters are ignored.

The mapping from Cadence Virtuoso design layers to GDS layers and datatypes should correspond to what is expected by the Calibre rule file and any Layer Map statements within the rule file. The layer map file must map all layers that you want Calibre RealTime to analyze. Layers that are not mapped are not analyzed.

For example, the following layer map entries map two design layers:

# Layer	Name ,	Purpose Name,	GDS	Layer,	GDS	Datatype
L3		drawing		3	0	
L6		drawing		6	0	

where the L3/drawing layer and purpose pair is mapped to GDS layer 3 and datatype 0, and the L6/drawing layer and purpose pair is mapped to GDS layer 6 and datatype 0.

See "CUT and CUTSIZE Options" and "pin Option" for information on those keywords.

File Naming Conventions

Name the layer map file *tech_lib.layermap*, where *tech_lib* is the name of the technology library, and place it in the technology library directory in order to have it read automatically. When specifying the layer map file in the Calibre RealTime Options dialog box, any filename is allowed, but the tool searches for files with a *.map* extension by default.

CUT and CUTSIZE Options

The CUT and CUTSIZE options in the layer map file allow shapes to map to different layers depending on the size of the shape. The behavior is best described with the following excerpt from a layer map file:

V1	drawing	16	0	CUT CUTSIZE:0.032:0.032
V1	drawing	16	15	CUT CUTSIZE:0.064:0.064
V1	drawing	16	1	CUT

Shapes from the V1/drawing layer and purpose pair in Cadence Virtuoso are mapped as follows to GDS:

- 32nm x 32nm shapes are mapped to GDS layer 16, datatype 0.
- 64nm x 64nm shapes are mapped to GDS layer 16, datatype 15.
- All other shapes on V1/drawing are mapped to GDS layer 16, datatype 1.

pin Option

The pin option maps pin shapes to a specified layer and datatype. For example:

```
M1 drawing 100 0
M1 drawing 100 9 pin
M2 drawing 101 0
```

These layer map entries map pin shapes on M1 to GDS layer 100, datatype 9; all other shapes on M1 are mapped to GDS layer 100, datatype 0. All shapes on M2, including pin shapes, are mapped to GDS layer 101, datatype 0.

These rules govern the mapping of pin shapes:

- Geometries that are not part of a pin are mapped by statements without the pin keyword.
- Pin geometries are mapped by statements with the pin keyword if it is present. If a statement with the pin keyword is not present, the pin geometry is mapped according to the mapping statement for the appropriate layer.

Error Handling in Layer Mapping

- If the layer mapping file contains multiple mapping statements for a design layer, then a warning is issued and the last layer mapping statement is used.
- Errors in the layer map file are handled as follows:
 - o If a format error is detected, an error is displayed and the layer map file is not used. The default technology-based layer map is used.
 - If one or more layers in the layer map file are not present in the design database, an
 informational message is displayed with the list of layers. The remaining valid layer
 mappings take effect.

Specifying the Layer Map for Calibre RealTime with Cadence Virtuoso

You specify the layer map file in the Calibre RealTime Options dialog box.

Note

You can also specify the layer map file with the environment variable or have it read automatically if it is in the technology library; see "GDS Layer Map for Calibre RealTime with Cadence Virtuoso" on page 242 for details.

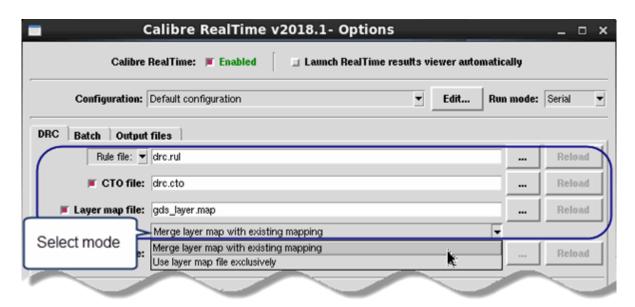
Prerequisites

• A layer map file, as described in "GDS Layer Map for Calibre RealTime with Cadence Virtuoso" on page 242.

• A design is open in Cadence Virtuoso and Calibre RealTime is enabled; see "Invoking Cadence Virtuoso with Calibre RealTime" on page 209.

Procedure

- 1. Click the button to open the Calibre RealTime Options Dialog Box.
- 2. Enable the checkbox for "Layer map file."
- 3. Enter the layer map file.



___Tip

You can enter an environment variable that is a directory path in the text entry field and click the ... button to have the file browser open to that directory.

Changes in the layer map file cause an automatic reload of the file prior to the DRC run.

- 4. Select the method for loading the layer map information:
 - Merge layer map with existing mapping (default) Merge layer map data from the specified file with the layer map file *tech_lib.layermap* found in the technology library. Layer map entries in the layer map file override the corresponding entry in *tech_lib.layermap*. All other layer map entries found in *tech_lib.layermap* remain valid; only entries found in the layer map file are overridden.
 - Use layer map file exclusively Use only layer map data from the specified file. Layer map data found in *tech_lib.layermap* in the technology library is *not* used.

a		

If "Use layer map file exclusively" is selected and the layer map file does not map all layers, then objects on unmapped layers are *not* analyzed by Calibre RealTime.

5. Click **Close** to close the Calibre RealTime Option dialog box.

Results

The following statements apply when Calibre RealTime performs layer mapping:

- If a layer map file is specified in the Calibre Options dialog box, the file is read the first time a DRC run is performed, and cached for later runs.
- To load a new or updated layer map file, specify the file in the Calibre RealTime Options dialog box, then click the **Reload** button. The new file is read before the next DRC run.

Also see "GDS Layer Map for Calibre RealTime with Cadence Virtuoso" on page 242 for a discussion on error handling.

Specifying the Object Map for Calibre RealTime with Cadence Virtuoso

You can define a custom object map file in Calibre RealTime for Cadence Virtuoso.

Prerequisites

• Cadence Virtuoso is open with Calibre RealTime enabled; see "Invoking Cadence Virtuoso with Calibre RealTime" on page 209.

Procedure

1. Create the object map file; see "Object Map File Format in Calibre RealTime" on page 40 for the file format.

If you use both an object map file and a layer map file, the same GDS layer number and datatype pairs should not be used in both files.

- 2. Click the button to open the Calibre RealTime Options Dialog Box.
- 3. Enable the checkbox for "Object map file."
- 4. Enter the object map file you created in Step 1.

Changes in the object map file cause an automatic reload of the file prior to the DRC run.

5. Click **Close** to close the Calibre RealTime Option dialog box.

Results

The object map file is used when exporting the layout to Calibre RealTime. A warning is issued if the design includes objects that are not mapped.

Related Topics

Object Map File Format in Calibre RealTime

GDS Layer Map for Calibre RealTime with Cadence Virtuoso

Approximation of Circles and Ellipses in Calibre RealTime with Cadence Virtuoso

Calibre RealTime approximates circles, ellipses, arcs, and donuts found in Cadence Virtuoso databases as polygons. By default, Calibre RealTime uses a 360 point polygon to approximate these shapes.

You can specify the number of points used in the polygon approximation with the environment variable MGC_REALTIME_ELLIPSE_POINT_COUNT; the minimum allowed value is 3.

Example

The following example instructs Calibre RealTime to use a 40 point polygon as the approximation for an ellipse, circle, arc, or donut.

setenv MGC_REALTIME_ELLIPSE_POINT_COUNT 90

Multi-Patterning Support in Calibre RealTime Custom in Cadence Virtuoso

Multi-patterning is supported in Calibre RealTime Custom in Cadence Virtuoso.

Calibre RealTime Custom can read locked and unlocked colors that are assigned to shape objects and vias. Mask colors 1 through 8 are supported.

Calibre RealTime Setup with Cadence SKILL Commands

Several SKILL commands are available to help you configure Calibre RealTime from within the Cadence environment.

SKILL Commands to Get and Set Calibre RealTime Options	249
Environment Variable Control in Calibre RealTime for Cadence Virtuoso	29

SKILL Commands to Get and Set Calibre RealTime Options

You can read and define the settings in the Calibre RealTime options dialog box using SKILL commands. The commands enable you to dynamically set up the Calibre RealTime configuration based on the current library, cell, and view.

____Tip _



You can also use environment variables to set the path to certain input files; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

The commands are divided into the following categories:

SKILL Commands for the Run Configuration

SKILL Commands to Set Calibre RealTime Options

SKILL Commands to Get Calibre RealTime Options

SKILL Commands for the Calibre RealTime Session

• SKILL Commands for the Run Configuration

mgc_calibre_realtime_add_configuration	Adds a configuration.
mgc_calibre_realtime_delete_configuration	Deletes a configuration.
mgc_calibre_realtime_get_configuration_ids	Returns a list of configuration IDs.
mgc_calibre_realtime_get_configuration_id_by_name	Returns the configuration ID associated with the specified name.
mgc_calibre_realtime_get_configuration_names	Returns a list of configuration names.
mgc_calibre_realtime_get_configuration_ name_by_id	Returns the configuration name associated with the specified ID.
mgc_calibre_realtime_set_current_config uration	Sets the current configuration to the specified ID or name.

Note

If you are using the Serial run configuration mode, use the "Configuration Run Control" dialog box to set the execution order for the configurations. See "Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)" on page 212.

SKILL Commands to Set Calibre RealTime Options

mgc_calibre_realtime_set_configura tion_option	Sets the value of a Calibre RealTime option. The run configuration is a required argument.
mgc_calibre_realtime_set_option	Sets the value of a Calibre RealTime option. The run configuration is an optional argument which is set to the current run configuration by default.
mgc_calibre_realtime_set_options	Sets all Calibre RealTime option values using the values set in the specified option table.
mgc_calibre_realtime_set_show_sin gle_error	Instructs Calibre RealTime to highlight a single error at a time when error markers overlap. Subsequent clicks on the same error marker cycle through the errors. By default (without calling this command) all errors are highlighted at once.

• SKILL Commands to Get Calibre RealTime Options

mgc_calibre_realtime_get_configura tion_option	Gets the value of a Calibre RealTime option. The run configuration is a required argument.
mgc_calibre_realtime_get_option	Gets the value of a Calibre RealTime option. The run configuration is an optional argument which is set to the current run configuration by default.
mgc_calibre_realtime_get_options	Gets all Calibre RealTime options and saves them in the global table mgc_realtime_options.

• SKILL Commands for the Calibre RealTime Session

mgc_calibre_realtime_cur_win_trig ger	SKILL procedure that is called with the window ID each time a new cell view is opened or becomes active. This procedure can be defined in the local Cadence environment and call the function mgc_calibre_realtime_drc_init to configure the options for Calibre RealTime.
mgc_calibre_realtime_drc_init	SKILL function that sets the options in the Calibre RealTime Options dialog box. This command should be called from the procedure mgc_calibre_realtime_cur_win_trigger defined in your local Cadence environment. The argument in the function call correspond to options in the Calibre RealTime Options dialog box.
mgc_calibre_realtime_get_api_versi on	Gets the version of the SKILL API for Calibre RealTime.
mgc_calibre_realtime_save_configuration	Saves the current Calibre RealTime configuration.

mgc_calibre_realtime_capture_layer _visibility	Saves the current layer visibility to be restored later. This procedure is useful when using a CTO file with the RVE Show Layers rule check comment, which can alter layer visibility.
mgc_calibre_realtime_restore_layer _visibility	Restores the saved layer visibility. This procedure is useful when using a CTO file with the RVE Show Layers rule check comment, which can alter layer visibility.

mgc_calibre_realtime_add_configuration

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that adds a new run configuration. The new configuration is automatically selected as the current run configuration.

Usage

mgc_calibre_realtime_add_configuration(tConfigName)

Arguments

tConfigName

The configuration name (a string). The name must be enclosed in quotes.

Description

mgc_calibre_realtime_add_configuration adds a new Run Configuration. The new configuration is automatically selected as the current run configuration.

See "SKILL Commands to Set Calibre RealTime Options" for commands to set the options in the configuration. If you are using the Serial run configuration mode, use the "Configuration Run Control" dialog box to set the execution order for the configurations. See "Specifying and Using Multiple Run Configurations in Calibre RealTime (Virtuoso)" on page 212.

Examples

mgc_calibre_realtime_add_configuration("DP_config")

mgc_calibre_realtime_capture_layer_visibility

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that saves the current layer visibility to be restored later. This procedure is useful when using a CTO file with the RVE Show Layers rule check comment, which can alter layer visibility.

Usage

mgc_calibre_realtime_capture_layer_visibility

Arguments

None.

Return Values

None.

Related Topics

mgc_calibre_realtime_restore_layer_visibility

mgc_calibre_realtime_cur_win_trigger

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that is called with the window ID each time a new cell view is opened or becomes active. This procedure can be defined in the local Cadence environment and call the function mgc_calibre_realtime_drc_init() to configure the options for Calibre RealTime.

Usage

mgc_calibre_realtime_cur_win_trigger(win)

Arguments

win

The window ID.

Related Topics

mgc_calibre_realtime_drc_init

mgc_calibre_realtime_delete_configuration

Available in: Calibre RealTime for Cadence Virtuoso SKILL procedure to delete a run configuration.

Usage

mgc_calibre_realtime_delete_configuration(gConfigNameOrID)

Arguments

• gConfigNameOrID

The configuration name (a string) or ID (an integer); a configuration name must be enclosed in quotes. See "SKILL Commands for the Run Configuration" for commands to get configuration names and IDs.

Examples

mgc calibre realtime delete configuration("Default configuration")

mgc_calibre_realtime_drc_init

Available in: Calibre RealTime for Cadence Virtuoso

SKILL function that sets the options in the Calibre RealTime Options dialog box. This command should be called from the procedure mgc_calibre_realtime_cur_win_trigger() defined in your local Cadence environment. The arguments in the function call correspond to options in the Calibre RealTime Options dialog box.

Note

Users are recommended to no longer use the mgc_calibre_realtime_drc_init command. Instead, use the mgc_calibre_realtime_set_configuration_option command, which replaces it.

Usage

Arguments

• ?enableRealTime enableRealTime

Keyword and value specifying whether to enable Calibre RealTime.

• ?ruleFile ruleFile

Keyword and value specifying the rule file.

• ?runset ""

Keyword and value specifying a Calibre Interactive runset.

• ?enableRunset nil

Keyword and value specifying whether to create a control file from a Calibre Interactive runset.

• ?layerMapFile layerMapFile

Keyword and value specifying the layer map file.

• ?enableLayerMapFile t

Keyword and value specifying whether to use a layer map file.

• ?layerMapFileMode layerMapFileMode

Keyword and value specifying the mode to use when processing the layer map file.

- o merge data from the layer map file is merged with information from the technology library.
- o replace data from the layer map file is used exclusively.

• ?ctoFile ctoFile

Keyword and value specifying the CTO file.

• ?enableCtoFile t

Keyword and value specifying whether to use a CTO file.

• ?haloSize *haloSize*

Keyword and value specifying the halo size.

• ?polygonCountLimit polygonCountLimit

Keyword and value specifying the polygon count limit.

Related Topics

mgc_calibre_realtime_cur_win_trigger

mgc_calibre_realtime_get_api_version

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that gets the version of the SKILL API for Calibre RealTime.

Usage

mgc_calibre_realtime_get_api_version

Arguments

None.

Related Topics

mgc_calibre_realtime_get_configuration_ids

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that returns a list of run configuration IDs.

Usage

mgc_calibre_realtime_get_configuration_ids

Arguments

None.

Description

The Run Configuration IDs are returned in the same order as the list returned from mgc_calibre_realtime_get_configuration_names.

See "SKILL Commands for the Run Configuration" for a list of commands related to the run configuration.

```
mgc_calibre_realtime_get_configuration_ids
(2 1 3)
```

mgc_calibre_realtime_get_configuration_id_by_name

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that returns the run configuration ID for the specified configuration name.

Usage

mgc_calibre_realtime_get_configuration_id_by_name(tConfigName)

Arguments

• tConfigName

A text string specifying a run configuration name. The name must be enclosed in quotes.

Description

Returns the integer Run Configuration ID associated with the specified configuration name.

See "SKILL Commands for the Run Configuration" for a list of commands related to the run configuration.

```
mgc_calibre_realtime_get_configuration_id_by_name("Default
configuration")
2
```

mgc_calibre_realtime_get_configuration_names

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that returns a list of run configuration names.

Usage

mgc_calibre_realtime_get_configuration_names

Arguments

None.

Description

The Run Configuration names are returned in the same order as the list returned from mgc_calibre_realtime_get_configuration_ids.

See "SKILL Commands for the Run Configuration" for a list of commands related to the run configuration.

```
mgc_calibre_realtime_get_configuration_names
("Default configuration" "45nm metal1" "45nm PP")
```

mgc_calibre_realtime_get_configuration_name_by_id

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that returns the run configuration name for the specified configuration ID.

Usage

mgc_calibre_realtime_get_configuration_name_by_id(nConfigID)

Arguments

• nConfigID

An integer configuration ID.

Description

Returns a text string giving the name for the specified configuration ID. Configuration IDs can be obtained with mgc_calibre_realtime_get_configuration_ids.

See "SKILL Commands for the Run Configuration" for a list of commands related to the run configuration.

```
mgc_calibre_realtime_get_configuration_ids
  (2 1 3)

mgc_calibre_realtime_get_configuration_name_by_id(1)
"45nm_metal1"
```

mgc_calibre_realtime_get_configuration_option

Available in: Calibre RealTime for Cadence Virtuoso

SKILL command to get the value for the specified option in the given run configuration.

Usage

mgc_calibre_realtime_get_configuration_option(gConfigNameOrID tOptionName)

Arguments

• gConfigNameOrID

The configuration name (a string) or ID number (an integer). A configuration name must be enclosed in quotes. See "SKILL Commands for the Run Configuration" for commands to get the configuration name or ID.

• tOptionName

The name of the option, where *optionName* can take the values given in the following table. The option names are case-sensitive. The *optionName* must be enclosed in quotes.

The first table provides the option names corresponding to options on the **DRC** tab; the second table provides the options names corresponding to options on the **Batch** tab.

Table 7-9. optionName Values for SKILL get commands

optionName value	Definition	
RealTimeEnabled	Returns a Boolean value specifying whether Calibre RealTime is enabled.	
CurrentConfiguration	Returns the name of the current Run Configuration.	
MultiConfigRunMode	Returns the configuration run mode.	
	 Single — Run only the current configuration. Serial — Run the selected configurations in order. 	
SelectedToRun	Returns a Boolean value specifying whether the given configuration is selected or not selected to run in Serial mode.	
RuleFile	Returns the path to the rule file, if set.	
RunsetFile	Returns the path to the runset file, if set.	
RuleSource	Returns the value "rulefile" or "runset" indicating whether a rule file or a Calibre Interactive runset file is the source of DRC rules.	
CTOFile	Returns the path to the CTO file, if set.	
CTOFileEnabled	Returns a Boolean value specifying whether Calibre RealTime is using a CTO file.	

Table 7-9. optionName Values for SKILL get commands (cont.)

optionName value	Definition	
RunRveOnDrc	Returns a Boolean value specifying whether the Calibre RealTime Results window is opened automatically after a run.	
LayerMapFile	Returns the path to the layer map file, if set.	
LayerMapFileEnabled	Returns a Boolean value specifying whether Calibre RealTime is using a layer map file.	
LayerMapFileMode	Returns the mode used to process the layer map file. Returns one of the following values:	
	 merge — data from the layer map file is merged with information obtained from the technology library. replace — data from the layer map file is used exclusively. 	
ObjectMapFile	Returns the path to the object map file, if set.	
ObjectMapFileEnabled	Returns a Boolean value specifying whether Calibre RealTime is using an object map file.	
HaloSize	Returns the size of halo in user units.	
PolygonCountLimit	Returns the maximum number of polygons that are passed to Calibre, as set in the Calibre Options dialog box.	
ResultLimitPerCheck	Returns the maximum number of results per check.	
DrcHierLevelsMode	Returns the setting for "DRC hier levels mode," which specifies the hierarchy levels in the design that are checked during the run.	
	• All — Check all hierarchy levels in the design.	
	Displayed — Check only the hierarchy levels that are displayed in Cadence Virtuoso.	
ResultDisplayMode	Returns the result display mode as a two element list of the form {region checks}, where the elements can have the following values:	
	region:	
	• area — only display results in the area of the previous DRC run.	
	• cell — display all known results in the cell. checks:	
	• all_checks — display results generated from all checks.	
	• recipe_checks — only display results generated from checks selected by the current recipe.	

Table 7-9. optionName Values for SKILL get commands (cont.)

optionName value	Definition	
HighlightAfterDRCRun	Returns a Boolean value specifying whether DRC violations are highlighted after a DRC run.	
ClearExistingHighlights	Return a Boolean value specifying whether previously drawn highlights are cleared before displaying new ones.	
ZoomToHighlights	Returns a Boolean value specifying whether Calibre RealTime zooms to drawn highlights.	
HighlightZoomRatio	Returns the value of the highlight zoom ratio.	
	The zoom ratio is the ratio of the highlight area to the window area. A ratio less than one indicates a zoom window larger than the highlighted area.	
GroupHighlightColorsByChe ck	Returns a Boolean value specifying whether Calibre RealTime groups highlight colors by check.	
IncludeCheckDefinitionInTo oltip	Returns a Boolean value specifying whether check definitions are included in tooltips associated with DRC highlights.	
HighlightShowCheckNames	Returns a Boolean value specifying the state of the option "Show check names" to display the check name at the center of each result.	
HighlightTextSize	Returns the value of the text size for displaying check names when HighlightShowCheckNames is enabled.	
ShowCheckDescriptionWind ow	Returns a Boolean value specifying the state of the option "Show check description window."	
RetrievePreviousResults	Returns a Boolean value specifying the state of the option "Retrieve previous results."	
CheckSelectionRecipe	Returns the currently selected check selection recipe.	
RestoreLayerVisibilityMode	Returns a string indicating when layer visibility is automatically restored.	
	 No — Layer visibility is not automatically restored AfterDRCRun — Layer visibility is restored after a DRC run 	
	AfterClearHighlight — Layer visibility is restored after clearing highlights	

Table 7-10. optionName Values for Batch Calibre Options for GetConfigurationOption

optionName value	Definition	
BatchAutomaticRunMode	Returns the run mode for automatically starting a batch Calibre run:	
	 Never — Do not automatically start a batch run. A dialog box is displayed if the maximum limit of three million polygons is exceeded. 	
	WhenHardLimitExceeded — Automatically start a batch Calibre run when the maximum limit of three million polygons is reached.	
	• WhenSoftLimitExceeded — Automatically start a batch Calibre run when the adjustable polygon limit is reached. The adjustable polygon limit is set with the PolygonCountLimit optionName, or on the DRC tab of the Calibre RealTime Options dialog box, and has a default of 50,000 polygons.	
BatchExecutiveMode	Returns the batch mode "Run Calibre DRC" setting of "Hierarchical" or "Flat".	
BatchThreadingMode	Returns the batch mode Performance setting:	
	• SingleThreaded — Run single-threaded. This is the only allowed mode for flat runs.	
	• MultiThreaded — Run multithreaded (-turbo command-line option).	
	• Distributed — Run in distributed MTflex mode. This mode requires a remote configuration file (-remotefile command-line option).	
BatchHyperscaleEnabled	Returns the string "t" or "nil", specifying whether hyperscaling is enabled for batch Calibre runs.	
BatchTurboNumberOfCPU s	Returns the number of processors specified for multithreaded batch Calibre runs:	
	0 — Use the maximum number of available CPUs. This is equivalent to the "All" setting in the Calibre RealTime Options dialog box.	
	<i>number</i> — Use the specified number of CPUs.	
BatchRemoteFile	Returns the path to the remote configuration file for batch Calibre runs in distributed MTflex mode.	
BatchQueueCommandEnab led	Returns "t" or "nil" indicating whether the remote cluster queue command is used to submit the Calibre batch run.	
BatchQueueCommand	Returns a string containing the queue command.	

Table 7-10. optionName Values for Batch Calibre Options for GetConfigurationOption (cont.)

optionName value	Definition		
BatchWorkingDirectory	Returns the path to the batch working directory.		
BatchLicensingMode	Returns the licensing mode for batch Calibre runs:		
	 NoWait — Exit if licenses are not available. (default) Wait — Wait for a license. If a license is not available within the specified wait time, attempt to acquire a substitute license. 		
	• Retry — Queue for a license if one is not available.		
BatchLicensingWaitTime	Returns the time in minutes to wait for a license in "Wait" mode.		
	1-4500; default=1 (minutes)		
BatchLicensingMaxRetryE nabled	Returns "t" or "nil" indicating whether a custom setting for the maximum time to queue for a license is enabled. ("Retry for" on the Batch tab of the options dialog box)		
	default: nil		
BatchLicensingMaxRetryTi me	Returns the maximum time in minutes to queue for a license in "Retry" mode.		
	1- "long max"; default: 180 (minutes)		
BatchLicensingRetryWaitE nabled	Returns "t" or "nil" indicating whether a custom setting for the wait time between license retries is enabled. ("Wait for" on the Batch tab of the options dialog box)		
	default: nil		
BatchLicensingRetryWaitT ime	Returns the time in seconds to wait between license retries in "Retry" mode.		
	1 - long max; default: 60 (seconds)		
BatchLicensingRetryOnErr orEnabled	Returns "t" or "nil" specifying whether to retry for a license when errors are detected. By default, license retry is only triggered when at least one required license is busy. default: nil		
	ucraum, mi		

mgc_calibre_realtime_get_option

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that gets the value of a Calibre RealTime option for the current run configuration or the specified configuration.

Note_

Beginning with the 2016.2 release, this command is replaced by mgc_calibre_realtime_get_configuration_option, which requires a Run Configuration to be specified. mgc_calibre_realtime_get_option is provided for backward compatibility.

Usage

mgc_calibre_realtime_get_option("optionName" @optional (gConfigNameOrID nil))

Arguments

"optionName"

The name of the option, where *optionName* can take the values given in the following table. The option names are case-sensitive. The *optionName* must be enclosed in quotes.

Table 7-11. optionName Values for SKILL get command	Table 7-11	. optionName	Values for	r SKILL	qet	command
---	-------------------	--------------	------------	---------	-----	---------

optionName value	Definition	
RealTimeEnabled	Returns a Boolean value specifying whether Calibre RealTime is enabled.	
CurrentConfiguration	Returns the name of the current Run Configuration.	
MultiConfigRunMode	Returns the configuration run mode.	
	• Single — Run only the current configuration.	
	• Serial — Run the selected configurations in order.	
SelectedToRun	Returns a Boolean value specifying whether the given configuration is selected or not selected to run in Serial mode.	
RuleFile	Returns the path to the rule file, if set.	
RunsetFile	Returns the path to the runset file, if set.	
RuleSource	Returns the value "rulefile" or "runset" indicating whether a rule file or a Calibre Interactive runset file is the source of DRC rules.	
CTOFile	Returns the path to the CTO file, if set.	
CTOFileEnabled	Returns a Boolean value specifying whether Calibre RealTime is using a CTO file.	

Table 7-11. optionName Values for SKILL get commands (cont.)

optionName value	Definition	
RunRveOnDrc	Returns a Boolean value specifying whether the Calibre RealTime Results window is opened automatically after a run.	
LayerMapFile	Returns the path to the layer map file, if set.	
LayerMapFileEnabled	Returns a Boolean value specifying whether Calibre RealTime is using a layer map file.	
LayerMapFileMode	Returns the mode used to process the layer map file. Returns one of the following values:	
	 merge — data from the layer map file is merged with information obtained from the technology library. replace — data from the layer map file is used exclusively. 	
ObjectMapFile	Returns the path to the object map file, if set.	
ObjectMapFileEnabled	Returns a Boolean value specifying whether Calibre RealTime is using an object map file.	
HaloSize	Returns the size of halo in user units.	
PolygonCountLimit	Returns the maximum number of polygons that are passed to Calibre, as set in the Calibre Options dialog box.	
ResultLimitPerCheck	Returns the maximum number of results per check.	
DrcHierLevelsMode	Returns the setting for "DRC hier levels mode," which specifies the hierarchy levels in the design that are checked during the run.	
	• All — Check all hierarchy levels in the design.	
	Displayed — Check only the hierarchy levels that are displayed in Cadence Virtuoso.	
ResultDisplayMode	Returns the result display mode as a two element list of the form {region checks}, where the elements can have the following values:	
	region:	
	• area — only display results in the area of the previous DRC run.	
	• cell — display all known results in the cell. checks:	
	• all_checks — display results generated from all checks.	
	• recipe_checks — only display results generated from checks selected by the current recipe.	

Table 7-11. optionName Values for SKILL get commands (cont.)

optionName value	Definition	
HighlightAfterDRCRun	Returns a Boolean value specifying whether DRC violations are highlighted after a DRC run.	
ClearExistingHighlights	Return a Boolean value specifying whether previously drawn highlights are cleared before displaying new ones.	
ZoomToHighlights	Returns a Boolean value specifying whether Calibre RealTime zooms to drawn highlights.	
HighlightZoomRatio	Returns the value of the highlight zoom ratio.	
	The zoom ratio is the ratio of the highlight area to the window area. A ratio less than one indicates a zoom window larger than the highlighted area.	
GroupHighlightColorsByChe ck	Returns a Boolean value specifying whether Calibre RealTime groups highlight colors by check.	
IncludeCheckDefinitionInTo oltip	Returns a Boolean value specifying whether check definitions are included in tooltips associated with DRC highlights.	
HighlightShowCheckNames	Returns a Boolean value specifying the state of the option "Show check names" to display the check name at the center of each result.	
HighlightTextSize	Returns the value of the text size for displaying check names when HighlightShowCheckNames is enabled.	
ShowCheckDescriptionWind ow	Returns a Boolean value specifying the state of the option "Show check description window."	
RetrievePreviousResults	Returns a Boolean value specifying the state of the option "Retrieve previous results."	
CheckSelectionRecipe	Returns the currently selected check selection recipe.	
RestoreLayerVisibilityMode	Returns a string indicating when layer visibility is automatically restored.	
	 No — Layer visibility is not automatically restored AfterDRCRun — Layer visibility is restored after a DRC run 	
	AfterClearHighlight — Layer visibility is restored after clearing highlights	

Table 7-12. optionName Values for Batch Calibre Options for GetConfigurationOption

optionName value	Definition	
BatchAutomaticRunMode	Returns the run mode for automatically starting a batch Calibre run:	
	 Never — Do not automatically start a batch run. A dialog box is displayed if the maximum limit of three million polygons is exceeded. 	
	WhenHardLimitExceeded — Automatically start a batch Calibre run when the maximum limit of three million polygons is reached.	
	• WhenSoftLimitExceeded — Automatically start a batch Calibre run when the adjustable polygon limit is reached. The adjustable polygon limit is set with the PolygonCountLimit optionName, or on the DRC tab of the Calibre RealTime Options dialog box, and has a default of 50,000 polygons.	
BatchExecutiveMode	Returns the batch mode "Run Calibre DRC" setting of "Hierarchical" or "Flat".	
BatchThreadingMode	Returns the batch mode Performance setting:	
	• SingleThreaded — Run single-threaded. This is the only allowed mode for flat runs.	
	• MultiThreaded — Run multithreaded (-turbo command-line option).	
	• Distributed — Run in distributed MTflex mode. This mode requires a remote configuration file (-remotefile command-line option).	
BatchHyperscaleEnabled	Returns the string "t" or "nil", specifying whether hyperscaling is enabled for batch Calibre runs.	
BatchTurboNumberOfCPU s	Returns the number of processors specified for multithreaded batch Calibre runs:	
	0 — Use the maximum number of available CPUs. This is equivalent to the "All" setting in the Calibre RealTime Options dialog box.	
	<i>number</i> — Use the specified number of CPUs.	
BatchRemoteFile	Returns the path to the remote configuration file for batch Calibre runs in distributed MTflex mode.	
BatchQueueCommandEnab led	Returns "t" or "nil" indicating whether the remote cluster queue command is used to submit the Calibre batch run.	
BatchQueueCommand	Returns a string containing the queue command.	

Table 7-12. optionName Values for Batch Calibre Options for GetConfigurationOption (cont.)

optionName value	Definition		
BatchWorkingDirectory	Returns the path to the batch working directory.		
BatchLicensingMode	Returns the licensing mode for batch Calibre runs:		
	 NoWait — Exit if licenses are not available. (default) Wait — Wait for a license. If a license is not available within the specified wait time, attempt to acquire a substitute license. 		
	• Retry — Queue for a license if one is not available.		
BatchLicensingWaitTime	Returns the time in minutes to wait for a license in "Wait" mode.		
	1-4500; default=1 (minutes)		
BatchLicensingMaxRetryE nabled	Returns "t" or "nil" indicating whether a custom setting for the maximum time to queue for a license is enabled. ("Retry for" on the Batch tab of the options dialog box)		
	default: nil		
BatchLicensingMaxRetryTi me	Returns the maximum time in minutes to queue for a license in "Retry" mode.		
	1- "long max"; default: 180 (minutes)		
BatchLicensingRetryWaitE nabled	Returns "t" or "nil" indicating whether a custom setting for the wait time between license retries is enabled. ("Wait for" on the Batch tab of the options dialog box)		
	default: nil		
BatchLicensingRetryWaitT ime	Returns the time in seconds to wait between license retries in "Retry" mode.		
	1 - long max; default: 60 (seconds)		
BatchLicensingRetryOnErr orEnabled	Returns "t" or "nil" specifying whether to retry for a license when errors are detected. By default, license retry is only triggered when at least one required license is busy. default: nil		

• gConfigNameOrID

An optional argument specify the configuration name (a string) or ID number (an integer). If *gConfigNameOrID* is not specified the currently selected configuration is used.

See "SKILL Commands for the Run Configuration" for commands to get the configuration name or ID.

Return Values

The return values are given in Table 7-11. If an option such as a file path has been set with an environment variable, the environment variable is included in the return value, not the expanded value.

Examples

Example 1

This example prints the value of the rule file setting for the current run configuration.

```
value = mgc_calibre_realtime_get_option("RuleFile")
printf("Rule file = %L\n" value)
```

Example 2

If a path is set with an environment variable, mgc_calibre_realtime_get_option returns the option value using the environment variable, not the expanded value. For example, if the layer map file is set to \$LMAP28, this command in the CIW:

```
mgc_calibre_realtime_get_option("LayerMapFile")
returns this:
    "$LMAP28"
```

Related Topics

```
mgc_calibre_realtime_get_options
mgc_calibre_realtime_set_option
SKILL Commands to Get and Set Calibre RealTime Options
```

mgc_calibre_realtime_get_options

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that gets all Calibre RealTime options and saves them to the global table mgc_realtime_options.

Usage

mgc_calibre_realtime_get_options

Arguments

None.

Description

The available table entries for mgc_realtime_options are the same as those listed in Table 7-11 on page 268 for the SKILL function mgc_calibre_realtime_get_option. Additional options may be seen when viewing the complete table, such as with the printstruct() command—options not listed in Table 7-11 are either obsolete, for internal use, or for future use.

If an option such as a file path has been set with an environment variable, the environment variable is included in the returned option value, not the expanded value.

Related Topics

 $mgc_calibre_real time_get_option$

mgc_calibre_realtime_set_options

mgc_calibre_realtime_restore_layer_visibility

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that restores the saved layer visibility. If there is none saved, then the version before the first DRC run is used. This procedure is useful when using a CTO file with the RVE Show Layers rule check comment, which can alter layer visibility.

Usage

mgc_calibre_realtime_restore_layer_visibility

Arguments

None.

Return Values

None.

Related Topics

mgc_calibre_realtime_capture_layer_visibility

mgc_calibre_realtime_save_configuration

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that saves the current Calibre RealTime session configuration to the configuration file (.*realtime.cfg*).

Usage

mgc_calibre_realtime_save_configuration

Arguments

None.

Return Values

None.

Description

The current session configuration is saved to the file .realtime.cfg as described in "Location of the Calibre RealTime Session Configuration File" on page 42. If you are using multilevel configuration files, the options saved are determined by the order of precedence of the configuration files, as described in "Multilevel Session Configuration File Support in Calibre RealTime Custom" on page 44.

New option values set by SKILL commands are automatically saved if the new values are different from the current values. See "SKILL Commands to Set Calibre RealTime Options."

Related Topics

mgc_calibre_realtime_set_option
mgc_calibre_realtime_set_options

mgc_calibre_realtime_set_configuration_option

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that sets the value of a Calibre RealTime option for the specified run configuration. File paths may include environment variables.

Usage

mgc_calibre_realtime_set_configuration_option(gConfigNameOrID "optionName" optionValue)

Arguments

• gConfigNameOrID

The configuration name (a string) or ID number (an integer). See "SKILL Commands for the Run Configuration" for commands to get the configuration name or ID.

• "optionName" optionValue

The name of the option and its value, as given in the following table. The option names and values are case-sensitive. The *optionName* must be enclosed in quotes.

The first table provides the corresponds to options on the **DRC** tab; the second table corresponds to options on the **Batch** tab.

Table 7-13. optionName and optionValue for SKILL set commands

optionName	Description	Allowed optionValue
RealTimeEnabled	Enable or disable Calibre RealTime. Configuration independent.	Boolean
CurrentConfiguration	Specifies the current run configuration. Configuration independent.	A configuration ID (an integer) or name (a text string enclosed in quotes).
MultiConfigRunMode	Set the configuration run mode. Configuration independent.	 Single — Run only the current configuration. Serial — Run the selected configurations in order.
SelectedToRun	Specifies the given configuration as selected or not selected to run in Serial mode.	Boolean
RuleFile	Define the rule file. No return value.	pathname

Table 7-13. optionName and optionValue for SKILL set commands (cont.)

optionName	Description	Allowed option Value
RunsetFile	Define the runset file. Returns the path to the control file on success, an empty string on failure.	pathname
RuleSource	Set the source of rules for the DRC run.	 rulefile — use a rule file. runset — use a Calibre Interactive runset.
CTOFile	Set the path to the CTO file.	pathname
CTOFileEnabled	Enable or disable using the CTO file.	Boolean
RunRveOnDrc	Enable or disable automatically opening the Calibre RealTime Results window after a run. Configuration independent.	Boolean
LayerMapFile	Set the path to the layer map file.	pathname
LayerMapFileEnabled	Enable or disable using the layer map file.	Boolean
LayerMapFileMode	Set the mode used to process data from the layer map file.	 merge — data from the layer map file is merged with information from the technology library. replace — data from the layer map file is used exclusively.
ObjectMapFileEnabled	Enable or disable using the object map file.	Boolean
ObjectMapFile	Set the path to the object map file.	pathname
HaloSize	Set the size of halo expressed in user units.	positive real
PolygonCountLimit	Set the maximum number of polygons that are passed to Calibre, as set in the Calibre Options dialog box.	integer
ResultLimitPerCheck	Set the maximum number of results per check.	positive integer or "all"

Table 7-13. optionName and optionValue for SKILL set commands (cont.)

optionName	Description	Allowed option Value
DrcHierLevelsMode	Specifies the hierarchy levels in the design that are checked during the run. This setting is not saved when you exit Calibre RealTime.	 All — Check all hierarchy levels in the design. Displayed — Check only the hierarchy levels that are displayed in Cadence Virtuoso.
ResultDisplayMode	Define which results are displayed by RealTime. The optionValue is a two element list defining the region in which results are displayed and the checks whose results are displayed.	A two element list of the form {region checks}, with the following allowed values: region: • area — only display results in the area of the previous DRC run. • cell — display all known results in the cell. checks: • all_checks — results generated by all checks are displayed. • recipe_checks — only results generated by checks selected by the current recipe are displayed. Example: "cell recipe_checks"
HighlightAfterDRCRun	Enable or disable automatic highlighting of DRC violations after a DRC run.	Boolean
ClearExistingHighlights	Set the value of the option which determines whether previously drawn highlights are cleared before displaying new ones.	Boolean
ZoomToHighlights	Set the value of the option which determines whether RealTime zooms to drawn highlights.	Boolean

Table 7-13. optionName and optionValue for SKILL set commands (cont.)

optionName	Description	Allowed option Value
HighlightZoomRatio	Set the value of the highlight zoom ratio.	positive real
	The zoom ratio is the ratio of the highlight area to the window area. A ratio less than one results in a zoom window larger than the highlighted area.	
GroupHighlightColorsBy Check	Set the value of the option which determines whether RealTime groups highlight colors by check.	Boolean
IncludeCheckDefinitionIn Tooltip	Set the value of the option which determines whether RealTime includes the check definitions in tooltips associated with DRC highlights.	Boolean
HighlightShowCheckNam es	Set the state of the option "Show check names" to display the check name at the center of each result.	Boolean
HighlightTextSize	Set the value of the text size for displaying check names when HighlightShowCheckNames is enabled.	positive real
ShowCheckDescriptionW indow	Set the state of the option "Show check description window."	Boolean
RetrievePreviousResults	Set the state of the option "Retrieve previous results."	Boolean
CheckSelectionRecipe	Set the current check selection recipe.	Check selection recipe name, enclosed in quotes.
RestoreLayerVisibilityMo de	Set when to automatically restore layer visibility.	 No — Do not automatically restore layer visibility AfterDRCRun — Restore layer visibility after a DRC run
		AfterClearHighlight Restore layer visibility after clearing highlights

Table 7-14. OptionName and OptionValue for SKILL set commands for Batch operation

optionName	Description	Allowed option Value	
BatchAutomaticRunMode	Set the run mode for automatically starting a batch Calibre run:	 Never — Do not automatically start a batch run. A dialog box is displayed if the maximum limit of three million polygons is exceeded. WhenHardLimitExceeded — Automatically start a batch Calibre run when the maximum limit of three million polygons is reached. WhenSoftLimitExceeded — Automatically start a batch Calibre run when the adjustable polygon limit is reached. The adjustable polygon limit is reached. The adjustable polygon CountLimit optionName, or on the DRC tab of the Calibre RealTime Options dialog box, and has a default of 50,000 polygons. 	
BatchExecutiveMode	Set the batch mode "Run Calibre DRC" setting.	HierarchicalFlat	
BatchThreadingMode	Set the Performance setting for a batch Calibre run.	 SingleThreaded — Run single-threaded. This is the only allowed mode for flat runs. MultiThreaded — Run multithreaded (-turbo command-line option). Distributed — Run in distributed MTflex mode. This mode requires a remote configuration file (-remotefile command-line option). 	
BatchHyperscaleEnabled	Enable or disable hyperscaling for batch Calibre runs.	• t • nil	

Table 7-14. OptionName and OptionValue for SKILL set commands for Batch operation (cont.)

optionName	Description	Allowed option Value
BatchTurboNumberOfCPU s	Set the number of processors for multithreaded batch Calibre runs.	0 — Use the maximum number of available CPUs. This is equivalent to the "All" setting in the Calibre RealTime Options dialog box. number — Use the specified number of CPUs.
BatchRemoteFile	Set the path to the remote configuration file for batch Calibre runs in distributed MTflex mode.	pathname
BatchQueueCommandEnab led	Set whether a remote cluster queue command is used to submit the Calibre batch job.	t — Use a queue command. nil — Do not use a queue command.
BatchQueueCommand	Specify the queue command.	The queue command, with arguments. The queue command must exist in the batch working directory. See "Batch Queue Command" on page 240 for requirements on the queue command script, replaceable parameters, and for a sample script.
BatchWorkingDirectory	Set the path to the batch working directory.	pathname
BatchLicensingMode	Set the licensing mode for batch Calibre runs.	 NoWait — Exit if licenses are not available. (default) Wait — Wait for a license. If a license is not available within the specified wait time, attempt to acquire a substitute license. Retry — Queue for a license if one is not available.
BatchLicensingWaitTime	Set the time in minutes to wait for a license in "Wait" mode.	Integer between 1 and 45,000. The default is 1.

Table 7-14. OptionName and OptionValue for SKILL set commands for Batch operation (cont.)

optionName	Description	Allowed option Value
BatchLicensingMaxRetryE nabled	Set whether a custom setting for the maximum time to queue for a license is enabled. ("Retry for" on the Batch tab of the options dialog box)	nil — Use the default max retry time (180 minutes). t — Use the custom max retry time, set with BatchLicensingMaxRetryTime.
BatchLicensingMaxRetryTi me	Set a custom value for the maximum time in minutes to queue for a license in "Retry" mode.	Integer between 1 and (2 ³¹ - 1).
BatchLicensingRetryWaitE nabled	Set whether a custom setting for the wait time between license retries is enabled. ("Wait for" on the Batch tab of the options dialog box)	nil — Use the default wait time between retries (60 seconds). t — Use the custom wait time between retries, set with BatchLicensingRetryWaitTime.
BatchLicensingRetryWaitT ime	Set a custom value for the time in seconds to wait between license retries in "Retry" mode.	Integer between 1 and (2 ³¹ - 1)
BatchLicensingRetryOnErr orEnabled	Set whether to retry for a license when errors are detected.	nil — License retry is only triggered when at least one required license is busy (default). t — Retry when license errors are detected.

Description

The new value is saved automatically if it is different than the current value, or you call mgc_calibre_realtime_save_configuration. The new value can be seen in the Calibre RealTime Options dialog box immediately. Pathnames for files may include environment variables.

mgc_calibre_realtime_set_current_configuration

Available in: Calibre RealTime for Cadence Virtuoso SKILL procedure that sets the current run configuration.

Usage

mgc_calibre_realtime_set_current_configuration(gConfigNameOrID)

Arguments

• gConfigNameOrID

The configuration name (a string) or ID (an integer). See "SKILL Commands for the Run Configuration" for commands to get configuration names and IDs.

Description

Sets the currently active Run Configuration. If the configuration run mode is "Serial," the current configuration cannot be set to a configuration that is disabled in the "Configuration Run Control" dialog box (the SelectedToRun option).

mgc_calibre_realtime_set_option

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that sets the value of a Calibre RealTime option. File paths may include environment variables. The value is set for the current run configuration, or an optional configuration argument can be specified.

Note

Beginning with the 2016.2 release, this command is replaced by mgc_calibre_realtime_set_configuration_option, which requires a Run Configuration to be specified. mgc_calibre_realtime_get_option is provided for backward compatibility.

Usage

Arguments

• "optionName" optionValue

The name of the option and its value, as given in the following table. The option names and values are case-sensitive. The *optionName* must be enclosed in quotes.

Table 7-15. optionName and optionValue for SKILL set commands

optionName	Description	Allowed option Value
RealTimeEnabled	Enable or disable Calibre RealTime. Configuration independent.	Boolean
CurrentConfiguration	Specifies the current run configuration. Configuration independent.	A configuration ID (an integer) or name (a text string enclosed in quotes).
MultiConfigRunMode	Set the configuration run mode. Configuration independent.	 Single — Run only the current configuration. Serial — Run the selected configurations in order.
SelectedToRun	Specifies the given configuration as selected or not selected to run in Serial mode.	Boolean
RuleFile	Define the rule file. No return value.	pathname

Table 7-15. optionName and optionValue for SKILL set commands (cont.)

optionName	Description	Allowed option Value
RunsetFile	Define the runset file. Returns the path to the control file on success, an empty string on failure.	pathname
RuleSource	Set the source of rules for the DRC run.	 rulefile — use a rule file. runset — use a Calibre Interactive runset.
CTOFile	Set the path to the CTO file.	pathname
CTOFileEnabled	Enable or disable using the CTO file.	Boolean
RunRveOnDrc	Enable or disable automatically opening the Calibre RealTime Results window after a run. Configuration independent.	Boolean
LayerMapFile	Set the path to the layer map file.	pathname
LayerMapFileEnabled	Enable or disable using the layer map file.	Boolean
LayerMapFileMode	Set the mode used to process data from the layer map file.	 merge — data from the layer map file is merged with information from the technology library. replace — data from the layer map file is used exclusively.
ObjectMapFileEnabled	Enable or disable using the object map file.	Boolean
ObjectMapFile	Set the path to the object map file.	pathname
HaloSize	Set the size of halo expressed in user units.	positive real
PolygonCountLimit	Set the maximum number of polygons that are passed to Calibre, as set in the Calibre Options dialog box.	integer
ResultLimitPerCheck	Set the maximum number of results per check.	positive integer or "all"

Table 7-15. optionName and optionValue for SKILL set commands (cont.)

optionName	Description	Allowed option Value
DrcHierLevelsMode	Specifies the hierarchy levels in the design that are checked during the run. This setting is not saved when you	 All — Check all hierarchy levels in the design. Displayed — Check
	exit Calibre RealTime.	only the hierarchy levels that are displayed in Cadence Virtuoso.
ResultDisplayMode	Define which results are displayed by RealTime. The <i>optionValue</i> is a two element	A two element list of the form { region checks}, with the following allowed values:
	list defining the <i>region</i> in which results are displayed and the	region:
	checks whose results are displayed.	 area — only display results in the area of the previous DRC run. cell — display all known results in the cell.
		checks:
		 all_checks — results generated by all checks are displayed. recipe_checks — only results generated by checks selected by the current recipe are displayed.
		Example:
W 12 1 A C DDCD		"cell recipe_checks"
HighlightAfterDRCRun	Enable or disable automatic highlighting of DRC violations after a DRC run.	Boolean
ClearExistingHighlights	Set the value of the option which determines whether previously drawn highlights are cleared before displaying new ones.	Boolean
ZoomToHighlights	Set the value of the option which determines whether RealTime zooms to drawn highlights.	Boolean

Table 7-15. optionName and optionValue for SKILL set commands (cont.)

optionName	Description	Allowed option Value
HighlightZoomRatio	Set the value of the highlight zoom ratio.	positive real
	The zoom ratio is the ratio of the highlight area to the window area. A ratio less than one results in a zoom window larger than the highlighted area.	
GroupHighlightColorsBy Check	Set the value of the option which determines whether RealTime groups highlight colors by check.	Boolean
IncludeCheckDefinitionIn Tooltip	Set the value of the option which determines whether RealTime includes the check definitions in tooltips associated with DRC highlights.	Boolean
HighlightShowCheckNam es	Set the state of the option "Show check names" to display the check name at the center of each result.	Boolean
HighlightTextSize	Set the value of the text size for displaying check names when HighlightShowCheckNames is enabled.	positive real
ShowCheckDescriptionW indow	Set the state of the option "Show check description window."	Boolean
RetrievePreviousResults	Set the state of the option "Retrieve previous results."	Boolean
CheckSelectionRecipe	Set the current check selection recipe.	Check selection recipe name, enclosed in quotes.
RestoreLayerVisibilityMo de	Set when to automatically restore layer visibility.	 No — Do not automatically restore layer visibility AfterDRCRun — Restore layer visibility after a DRC run
		AfterClearHighlight Restore layer visibility after clearing highlights

Table 7-16. OptionName and OptionValue for SKILL set commands for Batch operation

optionName	Description	Allowed option Value
BatchAutomaticRunMode	Set the run mode for automatically starting a batch Calibre run:	 Never — Do not automatically start a batch run. A dialog box is displayed if the maximum limit of three million polygons is exceeded. WhenHardLimitExceeded — Automatically start a batch Calibre run when the maximum limit of three million polygons is reached. WhenSoftLimitExceeded — Automatically start a batch Calibre run when the adjustable polygon limit is reached. The adjustable polygon limit is reached. The adjustable polygon CountLimit optionName, or on the DRC tab of the Calibre RealTime Options dialog box, and has a default of 50,000 polygons.
BatchExecutiveMode	Set the batch mode "Run Calibre DRC" setting.	HierarchicalFlat
BatchThreadingMode	Set the Performance setting for a batch Calibre run.	 SingleThreaded — Run single-threaded. This is the only allowed mode for flat runs. MultiThreaded — Run multithreaded (-turbo command-line option). Distributed — Run in distributed MTflex mode. This mode requires a remote configuration file (-remotefile command-line option).
BatchHyperscaleEnabled	Enable or disable hyperscaling for batch Calibre runs.	• t • nil

Table 7-16. OptionName and OptionValue for SKILL set commands for Batch operation (cont.)

optionName	Description	Allowed option Value
BatchTurboNumberOfCPU s	Set the number of processors for multithreaded batch Calibre runs.	0 — Use the maximum number of available CPUs. This is equivalent to the "All" setting in the Calibre RealTime Options dialog box. number — Use the specified
BatchRemoteFile	Set the path to the remote configuration file for batch Calibre runs in distributed MTflex mode.	number of CPUs. pathname
BatchQueueCommandEnab led	Set whether a remote cluster queue command is used to submit the Calibre batch job.	t — Use a queue command. nil — Do not use a queue command.
BatchQueueCommand	Specify the queue command.	The queue command, with arguments. The queue command must exist in the batch working directory. See "Batch Queue Command" on page 240 for requirements on the queue command script, replaceable parameters, and for a sample script.
BatchWorkingDirectory	Set the path to the batch working directory.	pathname
BatchLicensingMode	Set the licensing mode for batch Calibre runs.	 NoWait — Exit if licenses are not available. (default) Wait — Wait for a license. If a license is not available within the specified wait time, attempt to acquire a substitute license. Retry — Queue for a license if one is not available.
BatchLicensingWaitTime	Set the time in minutes to wait for a license in "Wait" mode.	Integer between 1 and 45,000. The default is 1.

Table 7-16. OptionName and OptionValue for SKILL set commands for Batch operation (cont.)

optionName	Description	Allowed option Value
BatchLicensingMaxRetryE nabled	Set whether a custom setting for the maximum time to queue for a license is enabled. ("Retry for" on the Batch tab of the options dialog box)	nil — Use the default max retry time (180 minutes). t — Use the custom max retry time, set with BatchLicensingMaxRetryTime.
BatchLicensingMaxRetryTi me	Set a custom value for the maximum time in minutes to queue for a license in "Retry" mode.	Integer between 1 and (2 ³¹ - 1).
BatchLicensingRetryWaitE nabled	Set whether a custom setting for the wait time between license retries is enabled. ("Wait for" on the Batch tab of the options dialog box)	nil — Use the default wait time between retries (60 seconds). t — Use the custom wait time between retries, set with BatchLicensingRetryWaitTime.
BatchLicensingRetryWaitT ime	Set a custom value for the time in seconds to wait between license retries in "Retry" mode.	Integer between 1 and (2 ³¹ - 1)
BatchLicensingRetryOnErr orEnabled	Set whether to retry for a license when errors are detected.	nil — License retry is only triggered when at least one required license is busy (default). t — Retry when license errors are detected.

gConfigNameOrID

An optional argument specifying the configuration name (a string) or ID number (an integer) for which the option is set. If *gConfigNameOrID* is not specified the currently selected configuration is used. See "SKILL Commands for the Run Configuration" for commands to get the configuration name or ID.

The following options are configuration independent and apply to all configurations, regardless of the value of *gConfigNameOrID*: MultiConfigRunMode, RealTimeEnabled, CurrentConfiguration, and RunRveOnDrc.

Description

The new value is saved automatically if it is different than the current value, or you call mgc_calibre_realtime_save_configuration. The new value can be seen in the Calibre RealTime Options dialog box immediately. Pathnames for files may include environment variables.

Examples

This example sets the value of two different options for the current run configuration.

```
mgc_calibre_realtime_set_option("ResultLimitPerCheck" 10)
mgc_calibre_realtime_set_option("LayerMapFile" "$LMAP28")
```

Related Topics

```
mgc_calibre_realtime_get_option
mgc_calibre_realtime_save_configuration
SKILL Commands to Get and Set Calibre RealTime Options
```

mgc_calibre_realtime_set_options

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that sets all Calibre RealTime option values from a table of option values.

Usage

mgc_calibre_realtime_set_options(@optional (optTable nil))

Arguments

optTable

Optional argument specifying the table containing the option values.

If this argument is not given, the global table mgc_realtime_options is used. The table entries for mgc_realtime_options are the same as those listed for the SKILL function mgc_calibre_realtime_get_option. Pathnames for files may include environment variables.

Description

You can create the option table read by mgc_calibre_realtime_set_options() by calling mgc_calibre_realtime_get_options(), which creates the global table mgc_realtime_options. You can then edit mgc_realtime_options or create a copy for editing.

The available table entries and possible values for mgc_realtime_options are the same as those listed in Table 7-15 on page 285 for the SKILL function mgc_calibre_realtime_set_option. Additional options may be seen when viewing the complete table, such as with the printstruct() command—options not listed in Table 7-15 are either obsolete or for future use. These additional options are not used.

Related Topics

mgc_calibre_realtime_get_options

mgc_calibre_realtime_save_configuration

SKILL Commands to Get and Set Calibre RealTime Options

mgc_calibre_realtime_set_show_single_error

Available in: Calibre RealTime for Cadence Virtuoso

SKILL procedure that instructs Calibre RealTime to highlight a single error at a time when error markers overlap. Subsequent clicks on the same error marker cycle through the errors. By default (without calling this command) all errors are highlighted at once.

Usage

mgc_calibre_realtime_set_show_single_error(@optional (bSingle t))

Arguments

• bSingle

An optional Boolean argument specifying how to highlight errors when the result markers overlap.

- t Highlight a single error at a time when error markers overlap. Subsequent clicks on the same error marker cycle through the errors. This is the default if this command is issued without na argument.
- nil Highlight all errors at once when result markers overlap. This is the standard behavior in Calibre RealTime if this command is not issued.

Environment Variable Control in Calibre RealTime for Cadence Virtuoso

Calibre RealTime provides commands to get and set environment variables from the Cadence command interface.

The mgc_calibre_realtime_setShellEnvVar or mgc_calibre_realtime_unsetShellEnvVar commands change environment variable settings in both Calibre RealTime and in the host process for the design tool. Calibre RealTime recompiles the rule file if an environment variable changes due to one of these commands.

Note

If your rule file uses DRC Select Check and DRC Unselect Check statements with conditional directives and environment variables to select the checks that are run, keep in mind that the check selection recipe "Checks selected in the rules file" is the only recipe that obeys DRC [Un]Select Check statements.

The following commands are available in Calibre RealTime for Cadence Virtuoso:

Table 7-17. Commands to Control Environment Variables in Calibre RealTime for Cadence Virtuoso

Command	Description
mgc_calibre_realtime_setShellEnv Var	Sets an environment variable in Calibre RealTime and the Cadence design tool environment.
mgc_calibre_realtime_unsetShellE nvVar	SKILL command to unset an environment variable in Calibre RealTime and the Cadence design tool environment.
mgc_calibre_realtime_getShellEnv Var	SKILL command to get the value of an environment variable.
mgc_calibre_realtime_isShellEnvV arDefined	A Skill command that determines if an environment variable exists in the Calibre RealTime process.

$_{f L}$ ${\sf Tip}$



You can also use environment variables to set the path to certain input files; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

mgc_calibre_realtime_setShellEnvVar

Available in: Calibre RealTime for Cadence Virtuoso.

Sets an environment variable in Calibre RealTime and the Cadence design tool environment.

Usage

```
mgc_calibre_realtime_setShellEnvVar("varName=varValue")
mgc_calibre_realtime_setShellEnvVar("varName" "varValue")
```

Arguments

• varName

The name of the environment variable.

• varValue

The value of the environment variable to be set.

Return Values

t or nil.

Description

A SKILL command that sets an environment variable in both Calibre RealTime and the host process for Cadence Virtuoso.

If this command is issued, Calibre RealTime automatically recompiles the rule file the next time a Calibre DRC run is launched or the check recipe editor is invoked.

Examples

This example sets the environment variable M1_CHECKS to 1.

```
% mgc_calibre_realtime_setShellEnvVar("M1_CHECKS=1")
t
```

mgc_calibre_realtime_unsetShellEnvVar

Available in: Calibre RealTime for Cadence Virtuoso.

SKILL command to unset an environment variable in Calibre RealTime and the Cadence design tool environment.

Usage

mgc_calibre_realtime_unsetShellEnvVar("varName")

Arguments

• varName

The name of the environment variable.

Return Values

None.

Description

A Skill command that unsets an environment variable in both Calibre RealTime and the host process for Cadence Virtuoso.

If this is command issued, Calibre RealTime automatically recompiles the rule file the next time a Calibre DRC run is launched or the check recipe editor is invoked.

Examples

This example unsets the environment variable M1_CHECKS.

```
% mgc calibre realtime unsetShellEnvVar("M1 CHECKS")
```

mgc_calibre_realtime_getShellEnvVar

Available in: Calibre RealTime for Cadence Virtuoso

SKILL command to get the value of an environment variable.

Usage

mgc_calibre_realtime_getShellEnvVar("varName")

Arguments

• varName

The name of the environment variable.

Return Values

The assigned value or nil.

Description

A Skill command that retrieves the value of the environment variable *varName* in the context of Calibre RealTime.

Examples

This example gets the value of environment variable M1_CHECKS; the value is returned.

```
% mgc_calibre_realtime_getShellEnvVar("M1_CHECKS")
1
```

mgc_calibre_realtime_isShellEnvVarDefined

Available in: Calibre RealTime for Cadence Virtuoso

A Skill command that determines if an environment variable exists in the Calibre RealTime process.

Usage

mgc_calibre_realtime_isShellEnvVarDefined("varName")

Arguments

• varName

The name of the environment variable.

Return Values

Returns t if *varName* exists, or nil otherwise.

Examples

This example checks for the existence of the environment variable M1_CHECKS.

```
% mgc_calibre_realtime_isShellEnvVarDefined("M1_CHECKS")
1
```

Calibre RealTime Configuration File (Virtuoso)

Calibre RealTime settings are saved in the Calibre RealTime configuration file. The configuration file is saved automatically and named .*realtime.cfg*; the file is saved in your \$HOME directory by default.

The settings in the Calibre RealTime Options Dialog Box, DRC Run and Highlight Options Menu (Virtuoso), and check recipe definitions are saved in the Calibre RealTime configuration file.

You can delete .realtime.cfg to return to the default configuration; however, this will also delete the custom check recipe definitions contained in the configuration file. You can save check recipes to file; see Export and Import in the description of the "Calibre RealTime Recipe Editor Dialog Box" on page 116.

Calibre RealTime for Cadence Virtuoso also supports management of the configuration settings by an administrator group; see "Multilevel Session Configuration File Support in Calibre RealTime Custom" on page 44 for information.

Caution_

Editing the configuration file is not recommended unless deleting complete lines in order to create a multilevel configuration file setup.

Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso

A set of standard keyboard shortcuts are provided for Calibre RealTime with Cadence Virtuoso.

The keyboard shortcuts are defined in the following file:

\$CALIBRE HOME/shared/pkgs/icv/tools/queryskl/mgc bindkeys.skl

This file is not loaded by default to avoid possible conflicts with existing keyboard shortcuts. The definition of the bindkey functions is given in table Table 7-18. The section "Defining and Loading Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso" on page 301 describes how to define and load your own keyboard shortcuts.

Table 7-18. Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso

Bindkey Function Definition	Standard Keyboard Shortcut
mgc_calibre_realtime_current_window_run()	F12
Run DRC in window.	
mgc_calibre_realtime_current_cell_run()	F11
Run DRC on current cell.	
mgc_calibre_realtime_select_window_run()	Ctrl-F12
Use the mouse to select a rectangular area for the Calibre RealTime run. Select two points at opposite corners to define the rectangle.	
mgc_calibre_realtime_show_options()	Shift-F12
Open the Calibre RealTime Custom Options Dialog Box (Virtuoso).	
mgc_calibre_realtime_next_recipe()	q
Select the next recipe in the toolbar dropdown list.	
mgc_calibre_realtime_previous_recipe()	w
Select the previous recipe in the toolbar dropdown list.	
mgc_calibre_realtime_lastused_recipe()	e
Select the most recently used recipe in the toolbar dropdown list.	
mgc_calibre_realtime_highlight_clear()	Ctrl-b
Clear highlights.	
mgc_calibre_realtime_highlight_all()	Ctrl-a
Highlight all results.	

Table 7-18. Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso

Bindkey Function Definition	Standard Keyboard Shortcut
mgc_calibre_realtime_highlight_previous()	p
Highlight the previous result.	
mgc_calibre_realtime_highlight_current()	С
Highlight the current result.	
mgc_calibre_realtime_highlight_next()	n
Highlight the next result.	
mgc_calibre_realtime_highlight_currentcheck()	Shift-c
Highlight results in the check currently selected in the toolbar dropdown list.	
mgc_calibre_realtime_previous_check()	Shift-p
Select the previous check in the toolbar dropdown list.	
mgc_calibre_realtime_remove_check()	Shift-d
Remove the selected check from the current recipe.	
mgc_calibre_realtime_next_check()	Shift-n
Select the next check in the toolbar dropdown list.	

Defining and Loading Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso

You can define custom keyboard shortcuts for Calibre RealTime. The Calibre installation provides a file of standard keyboard shortcuts.

Procedure

1. Copy the Skill file \$CALIBRE_HOME/shared/pkgs/icv/tools/queryskl/mgc_bindkeys.skl to the desired location for your keyboard shortcut definitions. "Keyboard Shortcuts in Calibre RealTime for Cadence Virtuoso" on page 300 lists the shortcuts defined in this file.

For example:

2. Edit the shortcut definitions as desired.

3. Load the new shortcut definitions by executing the following command in the Cadence Virtuoso environment:

load("~/my_realtime_keys.skl")

Troubleshooting Calibre RealTime with Cadence Virtuoso

Some troubleshooting tips are provided to address common problems in Calibre RealTime with Cadence Virtuoso.

Problem: Calibre RealTime does not initialize correctly

Solution: Check the following:

- Cadence Virtuoso is running in 64-bit mode.
- All environment variables are defined, as described in "Environment Setup for Cadence Virtuoso with Calibre RealTime" on page 203.
- You are running a supported version of Cadence Virtuoso; see "Requirements and Setup for Calibre RealTime with Cadence Virtuoso" on page 202.

Problem: Get prompt to save the MentorMessengerLib

Solution: Set the environment variable

MGC_CALIBRE_REALTIME_VIRTUOSO_SAVE_MESSENGER_CELL to a non-zero value.

Setting this environment variable causes Calibre RealTime to automatically save the MentorMessengerLib. The MentorMessengerLib is used internally by Calibre RealTime. If this environment variable is not set when you are using Calibre RealTime for Cadence Virtuoso, other Cadence tools in the design flow sometimes give a prompt to save the MentorMessengerLib.

Problem: the Calibre RealTime toolbar is slow to load when doing descend or ascend or opening a cell

Solution: Disable "Retrieve previous results" in the Calibre RealTime Options dialog box or define the environment variable

MGC_CALIBRE_REALTIME_NO_DRC_RESULT_RETRIEVAL.

Disabling the option or setting the environment variable prevents Calibre RealTime from retrieving previous results for a cell, which takes additional time.

Problem: Cadence Virtuoso crash with "symbol lookup error"

Solution: Make sure that the MentorObserverLib entry in cds.lib is defined using MGC_REALTIME_HOME if MGC_REALTIME_HOME is used.

This error can occur with newer versions of Cadence Virtuoso (ICAD12.2 or later), which require a newer version of the OpenAccess libraries than used with older versions of Cadence Virtuoso. The problem occurs if the MentorObserverLib is defined with a CALIBRE_HOME version that points to the older OpenAccess libraries.

Related Topics

Invoking Cadence Virtuoso with Calibre RealTime

Requirements and Setup for Calibre RealTime with Cadence Virtuoso

Edit-in-Place, Descend, and Ascend in Calibre RealTime with Cadence Virtuoso

Saving Mentor Libraries to a Specified cds.lib

Deleting Mentor Libraries When Calibre RealTime is Disabled



Chapter 8 Calibre RealTime Custom with Calibre DESIGNrev

Calibre RealTime Custom is integrated with Calibre DESIGNrev. Calibre RealTime Custom integrates flat Calibre nmDRC with supported layout design tools so that Calibre nmDRC can be run directly from the design tool.

The following topics may be useful in addition to those in this section:

- "Calibre RealTime Custom Introduction" on page 15 for an overview and system requirements
- "Calibre RealTime Custom Basics" on page 23
- "Calibre RealTime Environment Variables" on page 333

Requirements for Calibre RealTime with Calibre DESIGNrev	305
Running Calibre RealTime with Calibre DESIGNrev	307
Toolbars, Menus, and Dialog Boxes in Calibre RealTime (DESIGNrev)	317
Database Read Options in Calibre RealTime with Calibre DESIGNrev	328
Calibre RealTime Setup with Tcl Commands in Calibre DESIGNrev	329
Calibre RealTime Configuration File (DESIGNrev)	330
Keyboard Shortcuts in Calibre RealTime for Calibre DESIGNrey	330

Requirements for Calibre RealTime with Calibre DESIGNrev

Several requirements must be met in order to run Calibre RealTime with Calibre DESIGNrev. The following list gives the detailed requirements:

- Using Calibre DESIGNrev. The other Calibre layout viewers, such as Calibre WORKbench and Calibre MDPview, are not supported.
- Database layers are properly set up in Calibre DESIGNrev to work correctly with the Calibre rule file. This requirement is typically already met when using a GDSII or OASIS®1 format layout database. When using other database formats, use the layer

^{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.

mapping options available with Calibre DESIGNrev. See the section "Database Read Options in Calibre RealTime with Calibre DESIGNrev" on page 328 for more information.

- Not using an overlay with the layout.
- Using Calibre 2012.3 or later.
- Met the system and licensing requirements described in "Calibre RealTime Custom Requirements" on page 15.

Running Calibre RealTime with Calibre DESIGNrev

Invoking Calibre DESIGNrev with Calibre RealTime

After you start the Calibre DESIGNrev layout viewer with Calibre RealTime you see the Calibre RealTime integrated toolbar.

Prerequisites

- You have met requirements described in "Requirements for Calibre RealTime with Calibre DESIGNrev" on page 305.
- Make sure that the environment variable MGC_REALTIME_DISABLE is not set to 1 in your shell environment.

Procedure

1. Start Calibre DESIGNrev with the following command:

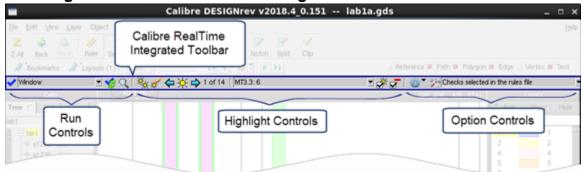
```
$CALIBRE_HOME/bin/calibredrv -m layout.database <options>
```

where *layout.database* is the layout file in a supported design format.

- 2. If the Calibre RealTime toolbar is not present, as shown in Figure 8-1, do the following:
 - a. Choose **Verification > RealTime > Options** to open the Calibre RealTime Options Dialog Box.
 - b. Select the "Enabled" checkbox for Calibre RealTime.
 - c. Click **OK** to exit the dialog box.

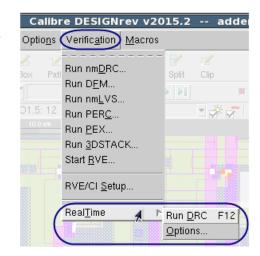
A view of Calibre DESIGNrev with the Calibre RealTime Integrated Toolbar is shown in the following figure.

Figure 8-1. Calibre RealTime Integration with Calibre DESIGNrev



Calibre RealTime also adds items to the **Verification** menu in Calibre DESIGNrev, as shown at right.

The Calibre RealTime server is started when the toolbar is loaded or the Calibre RealTime Options dialog box is opened.



3. Proceed to "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (DESIGNrev)" on page 308.

Related Topics

Calibre RealTime Configuration File (DESIGNrev)

Calibre RealTime Integrated Toolbar (DESIGNrev)

Calibre RealTime Environment Variables

License Timeout for Calibre RealTime Custom

Specifying the Rule File, Check Recipe, and Calibre RealTime Options (DESIGNrev)

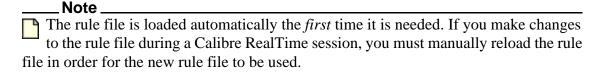
Calibre RealTime uses a standard Calibre nmDRC rule file. Check recipes control which checks in the rule file are executed during the Calibre RealTime run.

Prerequisites

- You have completed "Invoking Calibre DESIGNrev with Calibre RealTime" on page 307.
- You have a Calibre nmDRC rule file available.

Procedure

- 1. Choose **Verification > RealTime > Options** to open the Calibre RealTime Options dialog box, or click the **Options** button (on the integrated toolbar.
- 2. Specify the Calibre rule file in the "Rule file" text field.
- 3. Click **Reload** to load the rule file.



4. (Optional) Select options in the dialog box, if desired; the defaults are suitable for most runs. See "Calibre RealTime Options Dialog Box" on page 31 for a definition of each option.

If you want to save a summary file of the run and the DRC results as an ASCII results database (RDB), select the **Output files** tab and check the corresponding options.

- 5. Select the check recipe as follows:
 - Checks selected in the rules file (default) This built-in recipe uses the Calibre rule file just as a batch run of Calibre does.
 - Other built-in check recipes You can select a recipe that only runs checks on certain layers, excludes certain types of checks, or runs all checks in the rule file.

Because these built-in recipes do *not* obey DRC [Un]Select Check statements in the rule file, the recipes may execute checks in the rule file that would not be executed during a batch Calibre run. The recipe "Checks selected in the rules file" is the only built-in recipe that obeys DRC [Un]Select Check statements. See "Built-In Check Recipes" on page 107.

• User recipes — You can select a previously defined custom check recipe. See "Basic Editing of Check Selection Recipes" on page 111.

Group Highlight Color Check selection recipe: Edit... Checks selected in the rules file USER RECIPES Custom recipes rule_checks_nodens BUILT-IN RECIPES Close All checks Visible layer checks These recipes Visible layer checks, no connectivity do not obey Visible layer only checks DRC [Un]Select Visible layer only checks, no connectivity statements Recently edited layer checks Recently edited layer checks, no connectivity Checks selected in the rules file Rule file is read the same as for a batch Calibre run

Figure 8-2. Selecting a Check Recipe in Calibre RealTime (DESIGNrev)

6. Proceed to "Running Calibre RealTime in Window, Area, or Cell Mode with Calibre DESIGNrev" on page 310 or "Running Calibre RealTime in Run on Edit Mode with Calibre DESIGNrev" on page 312.

You can change environment variables with the commands given in the section "Environment Variable Control in Calibre RealTime" on page 53. This is useful if you want to change environment variables used in the rule file.

You can set Calibre RealTime options using Tcl commands in the Calibre DESIGNrev environment; see "Option Control in the Design Environment for Calibre RealTime" on page 58.

Related Topics

Using Calibre Interactive Settings in a Calibre RealTime Custom Run

Check Selection Recipes in Calibre RealTime Custom

Calibre RealTime Options Dialog Box (DESIGNrev)

Running Calibre RealTime in Window, Area, or Cell Mode with Calibre DESIGNrev

In window, area, or cell mode, Calibre RealTime runs checks on the geometries specified by the DRC run type. The specified check recipe selects the checks that are executed. All Calibre RealTime nmDRC runs are performed in flat mode.

Prerequisites

• You have completed "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (DESIGNrev)" on page 308

Procedure

- 1. In Calibre DESIGNrev, open the options dropdown menu () and disable "Run DRC on Edit"; this puts Calibre RealTime in the Run DRC in Window mode, which is indicated with the icon in the integrated toolbar. (This step is optional—you can still run in window, area, or cell mode while "Run DRC on Edit" is enabled.)

 See "DRC Run and Highlight Options Menu (DESIGNrev)" on page 320 for the
 - See "DRC Run and Highlight Options Menu (DESIGNrev)" on page 320 for the definitions of the remaining selections in the options menu.
- 2. Select the DRC run type and start a run.

The DRC run type is selected in the toolbar, in a dropdown menu next to the

button:



DRC Run Type	Instructions
Window	Runs DRC on the geometries visible in the design tool window:
	1. Choose the Window run type.
	2. Zoom to the region you want to run DRC on and make sure all the geometries you want to run checks on are visible.
	3. Click the button to start the run.
Area	Runs DRC on a selected region:
	1. Choose the Area run type.
	2. Click the 🗸 button.
	3. Use the mouse to drag a rectangle around the area you want to run DRC on.
Cell	Runs DRC on the whole cell that is open in the active window:
	1. Choose the Cell run type.
	2. Click the button to start the run.

DRC Run Type	Instructions
Past Region 1, Past Region 2,	Runs DRC on a previously selected area: 1. Choose the area to run on. The most recent area is always named Past Region 1.
	2. Click the vun.

Results

The results of a Calibre RealTime run are shown in Figure 8-3. You can use the highlight controls in the integrated toolbar to step through results.

Highlight controls 🗹 🍫 🔾 [🦠 🧹 🗘 🔆 🖒 1 of 26 [C01.1: 1 ✓ Window 🔻 🎉 🧽 🏻 👸 🐣 Checks selected in the rules file List Clips Show Hide 60 (CO1.5) POI VIA12\$\$33263660 C01 VIA12\$\$33264684 MTI V12 MID Object info popup shows rule check comments. TXT_MTI TXT MT2 (Make sure Options > Objects > Errors highlighted TXT_MTS Show Object info popup window after run finishes. is enabled, then select result) 4000 CO1.5 Layout: adder_4.gds * Cell: adder_4 Depth: [0 9]/2 Detail: 4 Grid: 100

Figure 8-3. Calibre RealTime Results in Calibre DESIGNrev

Related Topics

Viewing Result Tooltips in Calibre RealTime with Calibre DESIGNrev

Running Calibre RealTime in Run on Edit Mode with Calibre DESIGNrev

In Run on Edit mode, Calibre nmDRC runs automatically after any design edit. Calibre is run on the region of the design edit plus a halo region around the area; the specified check recipe selects the checks that are executed. All Calibre RealTime nmDRC runs are performed in flat mode.

Prerequisites

 You have completed "Specifying the Rule File, Check Recipe, and Calibre RealTime Options (DESIGNrev)" on page 308. Depending on your design, you may want to the choose one of the built-in recipes that limit the checks that are run, such as "Visible layer checks, no connectivity"; see "Built-In Check Recipes" on page 107 for details.

Procedure

- 1. Open the options dropdown menu (and enable "Run DRC on Edit"; this puts Calibre RealTime in the Run DRC on Edit mode, which is indicated with the icon in the integrated toolbar.
 - See "DRC Run and Highlight Options Menu (DESIGNrev)" on page 320 for the definitions of the remaining selections in the options menu.
- 2. Make any design edit to trigger a flat Calibre nmDRC run.
 - Calibre nmDRC runs on the region of the design edit plus the specified halo around the region.

Results

Results are similar to those shown in "Running Calibre RealTime in Window, Area, or Cell Mode with Calibre DESIGNrev" on page 310. You can use the highlight controls in the integrated toolbar to step through results.

Related Topics

Viewing Result Tooltips in Calibre RealTime with Calibre DESIGNrev Inconsistent Results from Run DRC on Edit Mode with Small Halo Size

Specifying Highlight Color and Layer Visibility for Calibre RealTime with Calibre DESIGNrev

You can use a Check Text Override (CTO) File to assign the highlight color and layer visibility for results according to the rule check.

Prerequisites

• You have completed "Invoking Calibre DESIGNrev with Calibre RealTime" on page 307.

Procedure

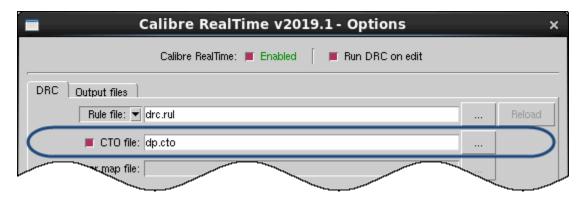
1. Create a Check Text Override (CTO) File for Calibre RealTime using a text editor. The following example CTO file specifies the highlight color and layer visibility for the min_ext_diff_poly rule:

```
# check text override file for XYZ
#
min_ext_diff_poly
RVE Highlight Color: blue
RVE Show Layers: diff poly
```

See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90 for complete information about the CTO file syntax and supported rule check comments.

- 2. Save the check text override file with a file extension of .cto.
- 3. In Calibre RealTime, click the button to open the Calibre RealTime Options dialog box.
- 4. Enable "CTO file" and specify a check text override file.

The .cto file extension is not required, but the file browser for the CTO file searches for files with a .cto file extension.



qiT.

You can enter an environment variable that is a directory path in the text entry field and click the ... button to have the file browser open to that directory.

You can also specify the CTO file with the environment variable MGC_REALTIME_CTO_FILE; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

Results

With the example shown in the procedure, all results from the rule check min_ext_diff_poly are highlighted in blue. Layers diff and poly are shown and all other layers are hidden.

Note

If you highlight results from a rule check with a RVE Show Layers rule check comment, and next highlight results from a rule check *without* a RVE Show Layers rule check comment, the layer visibility set with the first rule check is retained. You can use the controls in your design tool to set the layer visibility as desired.

Viewing Result Tooltips in Calibre RealTime with Calibre DESIGNrev

Result tooltips include the layer information, check name, check text, and optionally the check definition.

Prerequisites

• You have completed "Invoking Calibre DESIGNrev with Calibre RealTime" on page 307.

Procedure

- 1. Enable the object info popup window in Calibre DESIGNrev as follows:
 - a. Select **Options > Objects** to open the Preferences dialog box to the **Objects** tab.
 - b. Enable "Show object info popup window."
 - c. Click **OK** to close the Preferences dialog box.
- 2. (Optional) To include the full check definition in the tooltip, do the following:
 - a. Click the down arrow () in the Calibre RealTime integrated toolbar to open the DRC Run and Highlight Options menu.
 - b. Enable "Include Check Definition in Tool Tip."
- 3. Run Calibre RealTime as described in "Running Calibre RealTime in Window, Area, or Cell Mode with Calibre DESIGNrev" on page 310 or "Running Calibre RealTime in Run on Edit Mode with Calibre DESIGNrev" on page 312.

Note: If you changed the setting of "Include Check Definition in Tool Tip" in Step 2, then it is necessary to rerun Calibre RealTime for the new setting to take effect.

- 4. Use the highlight controls (← → →) to highlight a result.
- 5. Click a result highlight to select it—the object info popup window is displayed in the upper left of the layout window, as shown in the following figure.

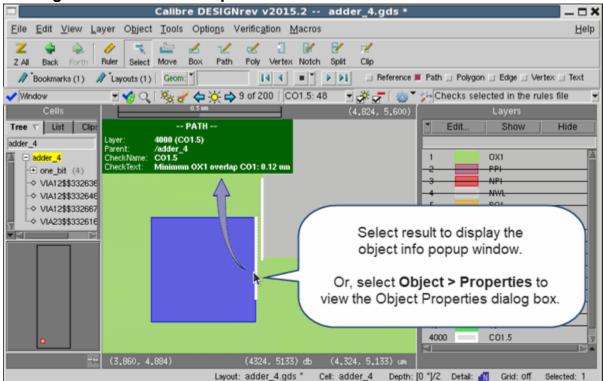


Figure 8-4. Result Tooltip in Calibre RealTime with Calibre DESIGNrev

You can also select **Object > Properties** while a result highlight is selected to view the result information in the Object Properties dialog box. This method for viewing object properties does not depend on the setting of "Show object info popup window" in the **Objects** tab of the Preferences dialog box.

Toolbars, Menus, and Dialog Boxes in Calibre RealTime (DESIGNrev)

The Calibre RealTime integrated toolbar and the Calibre RealTime Options dialog box provide the controls and settings you need to run Calibre RealTime. Some frequently accessed options are provided in the DRC Run and Highlight Options dropdown menu.

Calibre RealTime Integrated Toolbar (DESIGNrev)	317
Verification Menu in Calibre RealTime for Calibre DESIGNrev	320
DRC Run and Highlight Options Menu (DESIGNrev)	320
Calibre RealTime Options Dialog Box (DESIGNrev)	322

Calibre RealTime Integrated Toolbar (DESIGNrev)

The Calibre RealTime integrated toolbar includes controls to start a DRC run, highlight results, and open the Calibre RealTime Options dialog box. The toolbar also includes option settings for run control and highlighting.

The integrated toolbar is shown in the following figure:



The run controls, highlight controls, and option controls are described in the sections that follow.

Note.

For 2015.1 and earlier releases, the toolbar included a dropdown list of the ten most recent run contexts. Beginning with the 2015.2 release the run context is no longer displayed by default. Define the environment variable

MGC_CALIBRE_REALTIME_SHOW_CONTEXT_IN_TOOLBAR to restore the run context display.

Run Controls (DESIGNrev)

The run controls in the Calibre RealTime integrated toolbar are shown below and described in Table 8-1.

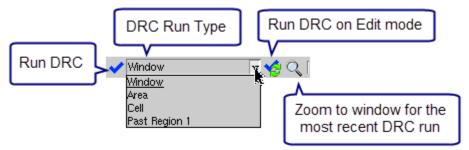


Table 8-1. Run Control Toolbar Selections (DESIGNrev)

Toolbar item	Description
✓	Runs flat DRC using the selected DRC Run Type.
(Run DRC)	Keyboard shortcut: F12
DRC Run Type dropdown	Selects the DRC run type: • Window — Run DRC on the geometries visible
	in the current window, plus a halo region around the area.
	• Area — Run DRC in a selected area. Click the Run DRC button () and use the mouse cursor to select an area in the design.
	• Cell — Run DRC in the whole cell that is open in the active window.
	• Past Region 1, Past Region 2, — Run DRC on a previously selected area.
Run DRC on Edit mode	— "Run DRC on Edit" is enabled; Calibre nmDRC runs automatically after any design edit.
	— "Run DRC on Edit" is disabled.
Q	Zoom to the window for the most recent DRC run.

Highlight Controls (DESIGNrev)

The highlight controls in the Calibre RealTime integrated toolbar are shown below and described in Table 8-2.



Table 8-2. Highlight Control Toolbar Selections (DESIGNrev)

Toolbar item	Description
%	Highlight all.
4	Clear all highlights.
⇔ 🔆 ⇒	Highlight previous, current, or next result for the rule check.
	Also see Behavior of "Group highlight colors by check" in "Calibre RealTime Options Dialog Box" on page 31.
1 of 5	Result <i>n</i> of <i>total</i> results.
	Also see the setting "Show results in" in the Calibre RealTime Options Dialog Box; this setting determines whether all results for the cell or only results from the most recent run are reported.
Rule check	Indicates the rule check which produced the displayed error when using the highlight previous, current, and next buttons. Use the dropdown arrow to select a different rule check.
*	Highlight all results from the selected rule check.

Option Controls (DESIGNrev)

The option controls in the Calibre RealTime integrated toolbar are shown below and described in Table 8-3.

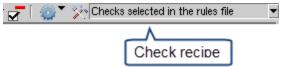


Table 8-3. Option Control Toolbar Selections (DESIGNrev)

Toolbar item	Description
✓	Remove the displayed rule check from the current check recipe. All highlights are cleared, regardless of the setting of "Clear Existing Highlights" in the options menu. If the recipe is a built-in recipe, you are prompted to save the new check recipe to a new filename; if the recipe is a user recipe, the updated recipe is saved automatically.

Table 8-3. Option Control Toolbar Selections (DESIGNrev) (cont.)

Toolbar item	Description
(Options dropdown menu)	Open the DRC Run and Highlight Options Menu (DESIGNrev).
	Open the Calibre RealTime Options Dialog Box.
**	Open the Customization GUI. This button is only available if a Calibre Interactive runset is loaded and includes a customization file. See "Using the Customization GUI with Calibre RealTime" on page 88.
Check recipe	Select the check recipe from the dropdown box; the check recipe specifies the rule checks executed for all Calibre RealTime runs. See "Check Selection Recipes in Calibre RealTime Custom" on page 105 for more information.

Verification Menu in Calibre RealTime for Calibre DESIGNrev

The Calibre RealTime integration adds the **RealTime** menu to the **Verification** menu in the main Calibre DESIGNrev menu bar.

The **Verification > RealTime** menu has the Calibre RealTime entries described in the following table:

Table 8-4. Verification Menu Contents (DESIGNrev)

Menu Item	Description
Run DRC	Starts a flat Calibre nmDRC run using the DRC Run type selected in the RealTime integrated toolbar. Calibre executes the rules checks selected in the currently specified check recipe.
Options	Opens the Calibre RealTime Options Dialog Box.

DRC Run and Highlight Options Menu (DESIGNrev)

Certain frequently used Calibre RealTime settings are available in the DRC Run and Highlight Options dropdown menu.

Click the down arrow (in the integrated toolbar to open the DRC Run and Highlight Options menu. The settings in the DRC Run and Highlight Options menu are saved in the

Session Configuration File in Calibre RealTime Custom. The menu items are described in the following table:

Table 8-5. DRC Run and Highlight Options Menu Contents (DESIGNrev)

Menu Item	Description
Zoom to Highlights	Zoom to result when highlighting.
Clear Existing Highlights	Clear existing highlights before each new highlight action.
	Highlights are cleared before each run, and when
	using the \Leftrightarrow \Rightarrow buttons to step though highlights.
Include Check Definition in Tool Tip	When enabled, the complete rule check definition is included in the result tool tip; otherwise, only the check text comments are shown.
	Note: You must run Calibre RealTime after changing this setting for the change to take effect.
	See "Viewing Result Tooltips in Calibre RealTime with Calibre DESIGNrev" on page 315.
Run DRC on Edit	Run flat Calibre nmDRC automatically after any design edit.

Calibre RealTime Options Dialog Box (DESIGNrev)

To access: Select **Verification > RealTime > Options** or click the **Options** toolbar button (



The Calibre RealTime Options dialog box specifies the Calibre rule file, the check recipe, and other options. The settings in the Calibre RealTime Options dialog box are saved in the Calibre RealTime Configuration file.

Description

 $\mathsf{_Tip}$. You can also set Calibre RealTime options using Tcl commands in the Calibre DESIGNrev environment; see "Option Control in the Design Environment for Calibre RealTime" on page 58.

Figure 8-5. Calibre RealTime Options Dialog Box (DESIGNrev)

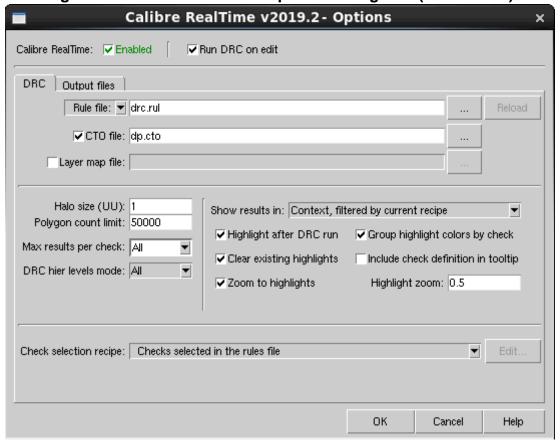


Table 8-6. Tabs in the Calibre RealTime Options Dialog Box (DESIGNrev)

Tab	Description
DRC	Specify options for the Calibre DRC run. See the table in the next section.

Table 8-6. Tabs in the Calibre RealTime Options Dialog Box (DESIGNrev)

Tab	Description
Output files	Specify to save a summary file and/or to save the DRC results to an ASCII results database (RDB). If a specified directory does not exist, the tool prompts you to confirm that the directory should be created.

Fields

Table 8-7. Calibre RealTime Options Dialog Box Contents (DESIGNrev)

Control	Description
Calibre RealTime	Enabled — Toggles display of the Calibre RealTime toolbar.
	The Calibre RealTime toolbar is automatically displayed if a DRC run is launched using Verification > RealTime > Run DRC or the F12 keyboard shortcut.
	Also see the environment variable MGC_REALTIME_DISABLE in "Calibre RealTime Environment Variables" on page 333.
Run DRC on edit	When enabled, start a flat Calibre nmDRC automatically after any design edit.
Rule file/Runset dropdown list	Selects whether to load a rule file or Calibre Interactive runset.
Rule file	Specifies the Calibre rule file used for the run.
Runset	Specifies a Calibre Interactive runset to use for the run. See "Using Calibre Interactive Settings in a Calibre RealTime Custom Run" on page 87 for details.
CTO file	Use a CTO file to specify the highlight color for a rule check. See "DRC Rule Check Comments for Calibre RealTime Custom" on page 90 and "Specifying Highlight Color and Layer Visibility for Calibre RealTime with Calibre DESIGNrev" on page 313.
Layer map file	Specifies a custom layer map file.
	See "Specifying the Layer Map for Calibre RealTime with Calibre DESIGNrev" on page 328.
(Browse)	Open a file browser.
Reload	Reload the run file or runset.
	You must click Reload for changes in the rule file and runset to take effect. Changes in the layer map and CTO files cause an automatic reload prior to the DRC run.

Table 8-7. Calibre RealTime Options Dialog Box Contents (DESIGNrev) (cont.)

Control	Description
Halo size (UU)	The halo size in user units. See "Halos in Calibre RealTime Custom" on page 24 and "Halo" in "Terms and Definitions in Calibre RealTime Custom" on page 19 for more information on halos.
Polygon count limit	Sets the maximum number of polygons that are passed to Calibre. The limit should be set low enough that the Calibre server returns control to the design tool in a short amount of time.
	The maximum number of geometries Calibre RealTime will process is 3,000,000.
Max results per check	Specifies the maximum number of results returned per rule check; this setting overrides the setting in the rule file.
	• All — report all results per check.
	• <i>number</i> — report only <i>number</i> results per check.
DRC hier levels mode	Specifies the hierarchy levels in the design that are checked during the run.
	• All — Check all hierarchy levels in the design.
	• Displayed — Check only the hierarchy levels that are displayed in Cadence Virtuoso.
	Depending on the selected rule checks, some rule check results may occur in the "Displayed" mode due to the fact that not all levels of the design are checked.

Table 8-7. Calibre RealTime Options Dialog Box Contents (DESIGNrev) (cont.)

Control	Description
Show results in	This setting controls which results are reported in the integrated toolbar after the run.
	• Whole cell, all checks — Report all results for the cell in the integrated toolbar after each DRC run. Reported results include those from the current run, previous results in the cell that are outside of the current run context, and possibly previous results within the current run context from rules checks not selected by the current recipe.
	• Whole cell, filtered by current recipe — Report results for the whole cell and from the current recipe in the integrated toolbar after each DRC run. Reported results include those from the current run plus previous results in the cell that are outside of the current run context.
	• Context, filtered by current recipe — (default) Report only results within the current run context and from the current check recipe in the integrated toolbar after each DRC run.
	For each case, all results in the current window and belonging to the current check selection recipe are deleted before the run.
	The settings "Whole cell, filtered by current recipe" and "Context, filtered by current recipe" should be used with caution, since they filter out results from previous runs. If these settings are used, it is recommended that a later run be performed with "Whole cell, all checks" selected to make sure that all results are reported.
Highlight after DRC run	Highlight results in the viewer after the DRC run.
Group highlight colors by check	Assign the highlight color for results according to the rule check.
	The assigned highlight color for a rule check is not fixed—rather, the highlight colors are assigned to checks in a repeated cycle, starting with the color red. See the "Usage Notes" section for details.
	Also see "Specifying Highlight Color and Layer Visibility for Calibre RealTime with Calibre DESIGNrev" on page 313.

Table 8-7. Calibre RealTime Options Dialog Box Contents (DESIGNrev) (cont.)

Control	Description
Clear existing highlights	Clear existing highlights before each new highlight action.
	Highlights are cleared before each run, and when using the buttons to step though highlights.
	Also see Behavior of "Group highlight colors by check" in the Usage Notes.
Include check definition in tooltip	When enabled, the complete rule check definition is included in the result tool tip; otherwise, only the check text comments are shown.
	Note: You must run Calibre RealTime after changing this setting for the change to take effect.
Zoom to highlights	Specifies whether to zoom to the extent of the results when highlighting.
	When automatically highlighting after a DRC run, RealTime does not zoom to highlights to avoid changing the currently selected viewpoint.
Highlight zoom	Specifies the magnification used when zooming to highlight a result.
Check selection recipe	The check recipe used for the Calibre run. See "Check Selection Recipes in Calibre RealTime Custom" on page 105.
Edit	Open the Calibre RealTime Recipe Editor Dialog Box.
Help	Opens the <i>Calibre RealTime User's Manual</i> in your default web browser.

Usage Notes

• Environment Variable Usage

- You can use environment variables in pathnames for files and directories. The environment variable is not expanded when viewed in the dialog box or saved to the configuration file. For example, you can enter \$CAL_RULES/drc.rul for the rule file, and \$CAL_RULES/drc.rul is saved in the configuration file.
- You can enter an environment variable for a directory path in a file entry field and click the ... button to open the file browser to that directory. This makes it easier to browse for a file if it is not in the working directory and the directory path is long. When you select a file, the absolute path is entered in the text entry file and saved in the configuration file. This capability is available for the rule file, CTO file, layer map file, and object map file, but not for the runset file.

There are system defined environment for the input files; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.

• Behavior of "Group highlight colors by check"

When "Group highlight colors by check" is enabled, colors are assigned to results according to the rule check. However, the association between the highlight color and the rule check is not fixed; it depends on the history of your Calibre RealTime session. Highlight colors are assigned to checks in a repeated cycle, starting with the color red. The color assignment starts at the beginning (with red) each time highlights are cleared. Highlights are cleared in the following cases:

- By clicking the Clear Highlights button (
- o Automatically when the button in the integrated toolbar is clicked. The button deletes a rule check from the current check recipe and clears all highlights.

The following behavior is observed when "Group highlight colors by check" is enabled:

Highlight color assignments from a DRC run:

"Clear Existing Highlights" ON — Highlights are cleared and highlight color assignment starts at red at the beginning of each Calibre RealTime run.

"Clear Existing Highlights" OFF — Highlights from previous runs are kept. New results from a check with an existing highlighted result are highlighted in the same color. New results from a check that does not have an existing highlighted result are highlighted in the next color in the highlight color cycle.

Clicking or — Highlights are cleared and highlight color assignment starts at red for the next run, regardless of the setting of "Clear Existing Highlights."

Highlight color assignments using the

"Clear Existing Highlights" OFF — Each result is highlighted in the color used originally.

Clear Highlights button clicked when "Clear Existing Highlights" is OFF—All highlights are cleared. The next result highlighted with one of the buttons is highlighted in red; the highlight color for further results is assigned according to the rule check. All highlights are kept until is clicked again.

Database Read Options in Calibre RealTime with Calibre DESIGNrev

You may need to set database read options, such as layer mapping and how to read text objects, when you use Calibre RealTime with Calibre DESIGNrev. All database read options may be set within the Calibre DESIGNrev environment. You can also specify a layer map file in the Calibre RealTime Options dialog box.

To set database read options within the Calibre DESIGNrev environment, see the following topics:

- "Using Layer Maps" in the Calibre DESIGNrev Layout Viewer User's Manual.
- Layout System description in the Standard Verification Rule Format (SVRF) Manual, which describes how the environment variable
 MGC_CALIBRE_DB_READ_OPTIONS is used to specify database read options for third-party layout formats (OpenAccess and LEF/DEF).

To specify a layer map file in the Calibre RealTime Options dialog box, see the following topic:

Specifying the Layer Map for Calibre RealTime with Calibre DESIGNrev...... 328

Specifying the Layer Map for Calibre RealTime with Calibre DESIGNrev

You can provide a layer map file in Calibre RealTime for Calibre DESIGNrev.

Prerequisites

• Calibre DESIGNrev is open with Calibre RealTime enabled; see "Invoking Calibre DESIGNrev with Calibre RealTime" on page 307.

Procedure

- 1. Create the layer map file as follows:
 - One entry per line with one of the following formats:

OpenAccess to GDS format used by Cadence Virtuoso and Synopsys Laker:

```
OA_layer_name OA_purpose_name GDS_layer_number GDS_datatype
```

Calibre DESIGNrev format:

```
DRV layer DRV datatype out GDS layer number.datatype
```

- ASCII text file.
- Comment lines start with #.

• Any filename is allowed, but the tool searches for files with a .map extension by default.

Caution.

- If the layer map file does not map all layers, then objects on unmapped layers are *not* analyzed by Calibre RealTime, and some DRC results may not be reported.
- 2. Click the button to open the Calibre RealTime Options Dialog Box.
- 3. Enable the checkbox for "Layer map file" and enter the layer map file you created in Step 1.

Changes in the layer map file cause an automatic reload of the file prior to the DRC run.

4. Click **Close** to close the Calibre RealTime Option dialog box.

Results

The following process is used when Calibre RealTime performs layer mapping:

- "Layer map file" Not Checked Calibre DESIGNrev develops a layer map according to its normal processes when it loads a layout or database.
- "Layer map file" Checked The specified layer map file is used. It is read the first time a DRC run is performed, and cached for later runs.

If the layer mapping file contains multiple mapping statements for a design layer, then a warning is issued and the last layer mapping statement is used.

Errors in the layer map file are handled as follows:

- o If a format error is detected, an error is displayed and the layer map file is not used. Calibre DESIGNrev performs layer mapping according to its normal processes.
- o If one or more layers in the layer map file are not present in the design, an informational message is displayed with the list of layers. The remaining valid layer mappings take effect.

To load a new or updated layer map file, specify the file in the Calibre RealTime Options dialog box, then click the **Reload** button. The new file is read before the next DRC run.

Calibre RealTime Setup with Tcl Commands in Calibre DESIGNrev

Several Tcl commands are available to help you configure Calibre RealTime from within the Calibre DESIGNrev environment.

See the following topics in the chapter "Calibre RealTime Custom Basics":

• "Environment Variable Control in Calibre RealTime" on page 53

• "Option Control in the Design Environment for Calibre RealTime" on page 58

Calibre RealTime Configuration File (DESIGNrev)

Calibre RealTime settings are saved in the Calibre RealTime configuration file. The configuration file is saved automatically and named .*realtime.cfg*; the file is saved in your \$HOME directory by default.

The settings in the Calibre RealTime Options Dialog Box, DRC Run and Highlight Options Menu (DESIGNrev), and check recipe definitions are saved in the Calibre RealTime configuration file.

You can delete .realtime.cfg to return to the default configuration; however, this will also delete the custom check recipe definitions contained in the configuration file. You can save check recipes to file; see Export and Import in the description of the "Calibre RealTime Recipe Editor Dialog Box" on page 116.

Calibre RealTime for Calibre DESIGNrev also supports management of the configuration settings by an administrator group; see "Multilevel Session Configuration File Support in Calibre RealTime Custom" on page 44 for information.

Caution

Editing the configuration file is not recommended unless deleting complete lines in order to create a multilevel configuration file setup.

Keyboard Shortcuts in Calibre RealTime for Calibre DESIGNrev

The F12 keyboard shortcut is defined by default to start a Calibre RealTime run on the current design window. Other keyboard shortcuts can be defined if desired, using the *keyprefs* file for Calibre DESIGNrev.

The available function calls for defining Calibre RealTime keyboard shortcuts in Calibre DESIGNrev are listed in the following table:

Table 8-8. Keyboard Shortcuts in Calibre RealTime for Calibre DESIGNrev

Function name	Definition
cwb Real Time Run DRC In Current Windo	Run DRC in window.
W	This function is bound to the F12 key by default.
cwbRealTimeRunDRCInCurrentCell	Run DRC in current cell.
	This function is bound to the F11 key by default.

Table 8-8. Keyboard Shortcuts in Calibre RealTime for Calibre DESIGNrev

Function name	Definition
cwbRealTimeShowOptions	Open the Calibre RealTime Options Dialog Box.
cwbRealTimeToggleRunOnEdit	Toggle the setting "Run DRC on Edit."
cwbRealTimeNextRecipe	Select the next recipe in the toolbar dropdown list.
cwbRealTimePrevRecipe	Select the previous recipe in the toolbar dropdown list.
cwbRealTimeLastUsedRecipe	Select the most recently used recipe in the toolbar dropdown list.
cwbRealTimeHighlightPrevResult	Highlight the previous result.
cwbRealTimeHighlightResult	Highlight the current result.
cwbRealTimeHighlightNextResult	Highlight the next result.
cwbRealTimePrevCheck	Select the previous check in the toolbar dropdown list.
cwbRealTimeNextCheck	Select the next check in the toolbar dropdown list.
cwbRealTimeHighlightCheck	Highlight results in the check currently selected in the toolbar dropdown list.
cwbRealTimeDeleteCheck	Remove the selected check from the current recipe.

Related Topics

Configuration Files [Calibre DESIGNrev Layout Viewer User's Manual] keyprefs File Format [Calibre DESIGNrev Layout Viewer User's Manual] \$cwb bindKey [Calibre DESIGNrev Reference Manual]



Appendix A Environment Variables in Calibre RealTime Custom

Calibre RealTime Custom reads a number of environment variables that specify run-time behavior.

You can set Calibre RealTime Custom environment variables within the design tool environment for Cadence Virtuoso, Calibre DESIGNrev, and Synopsys Laker. See the following topics:

- "Environment Variable Control in Calibre RealTime" on page 53 (Calibre DESIGNrev and Synopsys Laker)
- "Environment Variable Control in Calibre RealTime for Cadence Virtuoso" on page 295

The *Calibre Administrator's Guide* lists general and application specific Calibre environment variables in the appendix "Calibre Environment Variables."

For a list of environment variables used in Calibre RealTime Custom, see the following topic:

Calibre RealTime Environment Variables

Calibre RealTime uses a number of environment variables to specify run-time behavior. Some environment variables are used by all tools and other environment variables are tool-specific.

Table A-1. Calibre RealTime Environment Variables

Variable (applies to)	Description
CALIBRE_HOME	Sets the location of the software tree for Calibre tools; see "Setting the CALIBRE_HOME
(all tools)	Environment Variable" in the Calibre Administrator's Guide.

Table A-1. Calibre RealTime Environment Variables (cont.)

Variable (applies to)	Description
MGC_CALIBRE_DRC_RUNSET_FILE	Specifies a Calibre Interactive runset to be used to control the Calibre nmDRC run. You can specify multiple runsets as a space separated list enclosed in double quotes. See "Specifying a Calibre Interactive Runset in Calibre RealTime" on page 87 for complete information.
	This environment variable is also used by Calibre Interactive nmDRC. Also see MGC_REALTIME_RUNSET_FILE.
MGC_CALIBRE_HIDDEN_RECIPES	Specifies built-in check recipes to hide from the dropdown lists of check recipes. See "Hiding Built-In Check Recipes" on page 109.
MGC_CALIBRE_REALTIME_NO_DRC_R ESULT_RETRIEVAL (Cadence Virtuoso)	Define this environment variable to disable the retrieval of previous results for a cell after an open, descend, or ascend. Retrieval of previous results takes additional time. See "Edit-in-Place, Descend, and Ascend in Calibre RealTime with Cadence Virtuoso" on page 222.
MGC_CALIBRE_REALTIME_SHOW_CO NTEXT_IN_TOOLBAR (Cadence Virtuoso, Calibre DESIGNrev, and Synopsys Laker ³)	Define this environment variable to display the run context dropdown list in the toolbar. The run context dropdown list is not displayed by default beginning with the 2015.2 release.
MGC_CALIBRE_REALTIME_SHOW_DR C_RULE_DESCRIPTION_IN_TOOLBAR (Cadence Virtuoso)	By default the check text (if present) for a rule check is shown in the Calibre RealTime window; see the figure in "Running Calibre RealTime with Cadence Virtuoso" on page 215. Define this environment variable to also display the check text as a tooltip when you hover over the rule check name in the Calibre RealTime integrated toolbar.
MGC_CALIBRE_REALTIME_VIRTUOSO _ENABLED (Cadence Virtuoso)	Required environment variable to enable Calibre RealTime with Cadence Virtuoso. Set this environment variable to any value to enable Calibre RealTime. To disable Calibre RealTime with Cadence Virtuoso, unset MGC_CALIBRE_REALTIME_VIRTUOSO_E NABLED.

Table A-1. Calibre RealTime Environment Variables (cont.)

Variable (applies to)	Description
MGC_CALIBRE_REALTIME_VIRTUOSO _REMOVE_LIBS_ON_EXIT (Cadence Virtuoso)	Setting this environment variable causes Calibre RealTime to delete the MentorMessengerLib and MentorObserverLib entries from the <i>cds.lib</i> file if Calibre RealTime is disabled when you exit Cadence Virtuoso.
	See "Deleting Mentor Libraries When Calibre RealTime is Disabled" on page 223.
MGC_CALIBRE_REALTIME_VIRTUOSO _SAVE_MESSENGER_CELL (Cadence Virtuoso)	Setting this environment variable causes Calibre RealTime to automatically save the MentorMessengerLib. The MentorMessengerLib is used internally by Calibre RealTime. If this environment variable is not set when you are using Calibre RealTime for Cadence Virtuoso, other Cadence tools in the design flow sometimes give a prompt to save the MentorMessengerLib.
MGC_REALTIME_AUTO_LOAD_RULE_ FILES	Setting this environment variable to "none" disables automatic loading of rule files for all configurations.
MGC_REALTIME_CONFIG_DIRS (all tools)	Specifies an ordered list of multi-level configuration directories to read the Session Configuration File in Calibre RealTime Custom from. See "Multilevel Session Configuration File Support in Calibre RealTime Custom" on page 44. Not supported for all tools.
MGC_REALTIME_CTO_FILE	Specifies the path to the CTO file; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.
MGC_REALTIME_DISABLE (Synopsys Laker ³ and DB, Synopsys Custom Compiler, Calibre DESIGNrev, and Pyxis Layout)	Loading of the Calibre RealTime integration is disabled when this environment variable is set to 1. This environment variable is not used in the Cadence Virtuoso integration; see MGC_CALIBRE_REALTIME_VIRTUOSO_E NABLED.

Table A-1. Calibre RealTime Environment Variables (cont.)

Variable (applies to)	Description
MGC_REALTIME_ELLIPSE_POINT_COUNT (Synopsys Laker ³ and DB, Synopsys Custom Compiler, and Cadence Virtuoso)	Sets the number of vertices used to approximate an ellipse or circle when reading a Synopsys Laker database. The default is 20 for Synopsys Laker and 360 for Cadence Virtuoso; the minimum is 3.
MGC_REALTIME_FORCED_LIB_LOCAT ION	Specifies a non-default <i>cds.lib</i> in which to save the Mentor libraries. See "Saving Mentor Libraries to a Specified cds.lib" on page 223.
(Cadence Virtuoso)	Libraries to a specified cus.no on page 223.
MGC_REALTIME_HOME (Cadence Virtuoso)	Sets the Calibre RealTime version for Cadence Virtuoso to a different value than the Calibre version set by CALIBRE_HOME. See "Setting the Calibre RealTime Version Newer than CALIBRE_HOME" on page 205.
MGC_REALTIME_INCLUDE_RESULTS_ ENCOMPASSING_WINDOW	Beginning with the 2015.2 release, results that fully enclose the trim region are discarded by default. You can include such results by defining the environment variable MGC_REALTIME_INCLUDE_RESULTS_E NCOMPASSING_WINDOW; this returns the behavior to that found in 2015.1 and earlier releases. See "Keep Enclosing Results" on page 26 in the topic "Halo Usage in Calibre RealTime."
MGC_REALTIME_LAYER_MAP_FILE	Specifies the path to the layer map file; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.
MGC_REALTIME_OBJECT_MAP_FILE	Specifies the path to the object map file; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.
MGC_REALTIME_RELEASE_LICENSE_ TIME (all tools)	Sets the time in hours after which the Calibre RealTime license is released if there is no activity. A time-out of 10 minutes is used if this environment variable is not set. The minimum time-out period is 2 minutes; if a timeout of less than 2 minutes is specified the timeout is set to 2 minutes. See "License Timeout for Calibre RealTime Custom" on page 78.

Table A-1. Calibre RealTime Environment Variables (cont.)

Variable (applies to)	Description
MGC_REALTIME_RULE_FILE	Specifies the path to the rule file; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.
MGC_REALTIME_RUNSET_FILE	Specifies the path to the Calibre Interactive runset file; see "Environment Variable Settings for Calibre RealTime Input Files" on page 77.
MGC_REALTIME_USE_WINDOW_AS_T RIM_REGION	Set this environment variable to define the trim region as equal to the region for the DRC run. When this environment variable is defined, only results that are inside or intersect the region for the DRC run are kept. See "Reduced Trim Region" on page 25 in the topic "Halo Usage in Calibre RealTime."
MGC_RVE_GDS_LAYER_MAP_FILE (Cadence Virtuoso)	Sets the location of the layer map file in Calibre RealTime with Cadence Virtuoso. Information in the layer map file overrides the automatic layer map located in the technology library. See "GDS Layer Map for Calibre RealTime with Cadence Virtuoso" on page 242.
MGC_TMPDIR	Sets the location of the writable directory for temporary files created by Calibre RealTime. By default the temporary directory is \$MGC_HOME/tmp.

Related Topics

Calibre RealTime Custom Basics

Environment Variable Control in Calibre RealTime

Environment Variable Control in Calibre RealTime for Cadence Virtuoso

Appendix B Support for SVRF Statements in Calibre RealTime Custom

Calibre RealTime Custom provides limited support for some SVRF statements, particularly statements involving text objects and cell hierarchy. Some statements are supported with a limited syntax or for limited applications, some statements are not supported by all Calibre RealTime Custom integrations, and some statements are not supported at all.

The section "SVRF Statements with Limited Support in Calibre RealTime" on page 341 gives details for the commands with limited support.

Table B-1 summarizes the support status for SVRF statements that have limited or no support. This table is not necessarily complete and will be updated in future releases. The table uses the following terms:

Integration support: All — All integrations are supported.

Partial — Some integrations are not supported.

Syntax support: **Full** — The full syntax of the command is supported.

Limited — Some keywords or applications are not supported.

The Details column contains a link with more information in the case of partial and limited support.

Table B-1. SVRF Commands with Limited or No Support

Command	Integration Support	Syntax Support	Details
Density	All	Limited	Density in Calibre RealTime
DFM Copy	All	Limited	DFM Copy in Calibre RealTime
DFM Defaults	Not supported		
DFM NARAC	Not supported		
DRC Maximum Vertex	Not supported		
DFM Read	Not supported		
DFM RDB	Not supported		
DFM Text	Partial	Full	DFM Text in Calibre RealTime
DRC Results Database Precision	Not supported	l	Using this command may lead to unexpected results.

Table B-1. SVRF Commands with Limited or No Support (cont.)

Command	Integration Support	Syntax Support	Details
Expand Text	Partial	Full	Expand Text in Calibre RealTime
Extent Cell	All	Limited	Extent Cell in Calibre RealTime
Extent Drawn	Partial	Full	Extent Drawn in Calibre RealTime
Inside Cell	All	Limited	Inside Cell in Calibre RealTime
Layout Rename Text	All	Full	Layout Rename Text in Calibre RealTime
Layout Text	All	Full	Layout Text in Calibre RealTime
Layout Use Database	Not supported		
Layout Windel *	Not supported		
Layout Window *	Not supported		
Layout Polygon	Not supported		
Not Inside Cell	All	Limited	Not Inside Cell in Calibre RealTime
Not With Text	Partial	Full	Not With Text in Calibre RealTime
Precision	Not supported		
Rectangle	All	Full	
Rectangles	All	Full	
Text Depth	All	Full	
Virtual Connect *	Not supported		
With Text	Partial	Full	With Text in Calibre RealTime

SVRF Statements with Limited Support in Calibre RealTime

Some SVRF statements have limited syntax support in Calibre RealTime or are not supported by all Calibre RealTime integrations.

The following sections provide details about the statements with limited support. Table B-1 in "Support for SVRF Statements in Calibre RealTime Custom" on page 339 summarizes the support status.

Density in Calibre RealTime	34]
DFM Copy in Calibre RealTime	342
DFM Text in Calibre RealTime	342
Expand Text in Calibre RealTime	343
Extent Cell in Calibre RealTime	343
Extent Drawn in Calibre RealTime	344
Inside Cell in Calibre RealTime	344
Layout Text in Calibre RealTime	345
Layout Rename Text in Calibre RealTime	345
Not Inside Cell in Calibre RealTime	345
Not With Text in Calibre RealTime	340
With Text in Calibre RealTime	34

Density in Calibre RealTime

Calibre RealTime provides the following support for the Density statement.

Integration Support

Supported for: All integrations (Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, Pyxis Layout)

Supported Syntax in Calibre RealTime

```
DENSITY layer1 [layerN ...] [`['density_expression `]'] constraint
[WINDOW {wxy | wx wy} [STEP {sxy | sx sy}]]
[TRUNCATE | BACKUP | IGNORE | WRAP]
[INSIDE OF EXTENT | INSIDE OF x1 y1 x2 y2 |
      {INSIDE OF LAYER layerB [BY EXTENT [size_value]]} | BY POLYGON |
      {BY RECTANGLE [EXTENDED]} | CENTERED value]
[[GRADIENT constraint [RELATIVE | ABSOLUTE] [CORNER [value]]] |
      [MAGNITUDE constraint [RELATIVE | ABSOLUTE]]]
[CENTERS value]
[RDB [ONLY] filename [MAG value]
      {[COMBINE constraint [RELATIVE | ABSOLUTE]] | [MAXIMUM value]}]
```

See Density for a full description and argument definitions. Arguments not listed in the above syntax line, such as PRINT [ONLY], are not supported in Calibre RealTime; when included, unsupported arguments are ignored, and warnings are issued at the start of a run. You must use the argument order as shown to avoid ambiguity.

DFM Copy in Calibre RealTime

Calibre RealTime provides the following support for the DFM Copy statement.

Integration Support

Supported for: All integrations (Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, and Pyxis RealTime)

Supported Syntax in Calibre RealTime

```
DFM COPY layer [layer ...] [{REGION [UNMERGED]} | EDGE |
{CENTERLINE [CLUSTER]} | {TIE [CENTER] [CLUSTER]} |
{CLUSTER [LAYERID]} | UNMERGED] [PREMERGE]
```

See DFM Copy for a full description and argument definitions. Arguments not listed in the above syntax line, such as CELL LIST, are not supported in Calibre RealTime; when included, unsupported arguments are ignored, and warnings are issued at the start of a run. You must use the argument order as shown to avoid ambiguity.

DFM Text in Calibre RealTime

Calibre RealTime provides the following support for the DFM Text statement.

Integration Support

Supported for: Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, and Pyxis RealTime

Supported Syntax in Calibre RealTime

```
DFM TEXT layer [PRIMARY ONLY] PROPERTY
{STRING | [INVALID] NUMBER } property name
```

See DFM Text for a full description and argument definitions.

Expand Text in Calibre RealTime

Calibre RealTime provides the following support for the Expand Text statement.

Integration Support

Supported for: Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, and Pyxis RealTime

Supported Syntax in Calibre RealTime

```
EXPAND TEXT text_name [text_layer] BY number [PRIMARY ONLY] [CASE SENSITIVE]
```

See Expand Text for a full description and argument definitions.

Extent Cell in Calibre RealTime

Calibre RealTime provides the following support for the Extent Cell statement.

Integration Support

Supported for: All integrations (Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, Pyxis Layout)

Supported Syntax in Calibre RealTime

```
EXTENT CELL cell name [cell name ...] [{ORIGINAL | MAPPED} [OCCUPIED]]
```

See Extent Cell for a full description and argument definitions. Arguments not listed in the above syntax line, such as WITH MATCH, are not supported in Calibre RealTime; when included, unsupported arguments are ignored, and warnings are issued at the start of a run. You must use the argument order as shown to avoid ambiguity.

If a cell extends outside the halo region, Calibre RealTime only analyzes the portion of the geometry which is within the halo region. See "Halos in Calibre RealTime Custom" on page 24.

In the Cadence Virtuoso integration, exact cell name matching is not supported in the following cases:

• In SKILL and batch mode, colored cell variations named with additional suffixes such as _cv1 and _cv2

 In SKILL mode with Display level DRC, cell names with additional hierarchy level numbers

Extent Drawn in Calibre RealTime

Calibre RealTime provides the following support for the Extent Drawn statement.

Integration Support

Supported for: Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, and Pyxis RealTime

Supported Syntax in Calibre RealTime

```
EXTENT DRAWN [ORIGINAL | layer [layer ...]] [IGNORE layer [layer ...]]
```

See Extent Drawn for a full description and argument definitions.

Inside Cell in Calibre RealTime

Calibre RealTime provides the following support for the Inside Cell statement.

Integration Support

Supported for: All integrations (Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, Pyxis Layout)

Supported Syntax in Calibre RealTime

```
INSIDE CELL layer cell_name [cell_name ...]
[PRIMARY ONLY] [WITH LAYER layer2]
```

See Inside Cell for a full description and argument definitions. Arguments not listed in the above syntax line, such as WITH MATCH, are not supported in Calibre RealTime; when included, unsupported arguments are ignored, and warnings are issued at the start of a run. You must use the argument order as shown to avoid ambiguity.

If a cell extends outside the halo region, Calibre RealTime only analyzes the portion of the geometry which is within the halo region. See "Halos in Calibre RealTime Custom" on page 24.

In the Cadence Virtuoso integration, exact cell name matching is not supported in the following cases:

- In SKILL and batch mode, colored cell variations named with additional suffixes such as _cv1 and _cv2
- In SKILL mode with Display level DRC, cell names with additional hierarchy level numbers

Layout Text in Calibre RealTime

Calibre RealTime provides the following support for the Layout Text statement.

Integration Support

Supported for: All integrations (Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, Pyxis Layout)

Supported Syntax in Calibre RealTime

```
LAYOUT TEXT name x y layer [texttype] [cell name]
```

See Layout Text for a full description and argument definitions.

Layout Rename Text in Calibre RealTime

Calibre RealTime provides the following support for the Layout Rename Text statement.

Integration Support

Supported for: All integrations (Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, Pyxis Layout)

Supported Syntax in Calibre RealTime

```
LAYOUT RENAME TEXT
"delimiter regular_expression
delimiter replacement_pattern delimiter
[n | g] [e | b] [i | m] [Mc]
[delimiter regular_expression
delimiter replacement_pattern delimiter
[n | g] [e | b] [i | m] [Mc] ...]"
[CELL LIST name]
[DATABASE | RULES | DATABASE RULES]
[BY LAYER layer [TO destination layer]]
```

See Layout Rename Text for a full description and argument definitions.

Not Inside Cell in Calibre RealTime

Calibre RealTime provides the following support for the Not Inside Cell statement.

Integration Support

Supported for: All integrations (Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, Pyxis Layout)

Supported Syntax in Calibre RealTime

```
NOT INSIDE CELL layer cell_name [cell_name ...] [PRIMARY ONLY] [WITH LAYER layer2]
```

See Not Inside Cell for a full description and argument definitions; arguments not listed in the above syntax line are not supported in Calibre RealTime. You must use the argument order as shown to avoid ambiguity.

If a cell extends outside the halo region, Calibre RealTime only analyzes the portion of the geometry which is within the halo region. See "Halos in Calibre RealTime Custom" on page 24.

Not With Text in Calibre RealTime

Calibre RealTime provides the following support for the Not With Text statement.

Integration Support

Supported for: Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, and Pyxis RealTime

Supported Syntax in Calibre RealTime

```
NOT WITH TEXT layer name [text layer] [PRIMARY ONLY] [CASE SENSITIVE]
```

See Not With Text for a full description and argument definitions.

With Text in Calibre RealTime

Calibre RealTime provides the following support for the With Text statement.

Integration Support

Supported for: Synopsys Laker³, Calibre DESIGNrev, Cadence Virtuoso, and Pyxis RealTime

Supported Syntax in Calibre RealTime

```
WITH TEXT layer name [text layer] [PRIMARY ONLY] [CASE SENSITIVE]
```

See With Text for a full description and argument definitions.

Index

O	Pyxis Layout, 144
— Symbols —	Check selection recipe, defined, 20
[], 22	Check text override file, 101
{},22	Circle, in Synopsys Laker, 175
, 22	Circles, in Virtuoso, 247
\$set_tech_layermap, in Pyxis Layout, 130	Configuration file
— B —	defined, 42
Bold words, 21	environment variable, 44
Bold Wolds, 21	multi-level support, 44
— C —	Courier font, 22
Cadence Virtuoso	Custom Compiler, see Synopsys Custom
circles and ellipses, 247	Compiler
compatibility table, 204	Customization GUI, using Calibre Interactive,
configuration file, 299	89
enable Calibre RealTime, 203	_
environment variables, 203	— D —
keyboard shortcuts, 197, 300	Density results, 29
layer mapping, 242	Density, SVRF statement, 19
MentorMessengerLib, 204	DESIGNrev,seeCalibre DESIGNrev
multi-patterning support, 249	DFM Text, 342, 343
requirements, 202	Differences with Calibre nmDRC, 18
Skill commands, 249	Double Patterning debug, 79
troubleshooting, 302	Double pipes, 22
viewing RealTime toolbar, 211	DRC rule check comments
Calibre DESIGNrev	RVE Link, 99
highlight color, 314	RVE Show Layers, 96
invoking with, 307	DRC RVE
keyboard shortcuts, 330	specify layers shown in DESIGNrev when
layer mapping, 328	highlighting, 96
option control with Tcl commands, 58	-E-
requirements, 305	
Tcl commands for environment variables,	Edit-in-place, Synopsys Laker, 160
53	Ellipsis, in Synopsys Laker, 175
tooltips, 315	Ellipsis, in Virtuoso, 247 Environment variables
Calibre Interactive	
customization GUI, 89	Cadence Virtuoso, 203
using runset, 87	table, 333 Environment variables, controlling with Tol
Calibre nmDRC, differences with, 18	Environment variables, controlling with Tcl
Check recipes	commands, 53
overview 105	Expand Text, 343

Extent Cell, 342, 343 Extent Drawn, 344	MGC_CALIBRE_REALTIME_VIRTUOSO SAVE_MESSENGER_CELL, 204
— F —	Minimum keyword, 22
Function keys,seeKeyboard shortcuts — H —	— N — Not Inside Cell, 346 Not With Text, 346
Halo size, and inconsistent results, 27 Halo usage, 24 Halo, defined, 20 Heavy font, 21 Highlight color Calibre DESIGNrev, 314 DP debug, 81 general, 90	Object map file format, 40 Options dialog box in Synopsys Laker, 165 dialog box, Cadence Virtuoso, 32, 231 Tcl commands, 58
Pyxis Layout, 90 Hotkeys,seeKeyboard shortcuts	Parentheses, 22
Inside Cell, 344 Invoke Calibre RealTime Calibre DESIGNrev, 307 Italic font, 22	Pipes, 22 Property information, support, 18 Pyxis Layout \$set_tech_layermap, 130 Calibre Options dialog box, 141
 Keyboard shortcuts Cadence Virtuoso, 197, 300 Calibre DESIGNrev, 330 Pyxis Layout, 149 Synopsys Laker, 179 	check recipes, 144 hotkeys, 149 keyboard shortcuts, 149 layer mapping, 128 requirements, 16, 125 versions, 125
— L —	— Q — Quotation marks, 22
Laker, see Synopsys Laker Layer maps Cadence Virtuoso, 242 Calibre DESIGNrev, 328 general, 40 Pyxis Layout, 128 Synopsys Laker, 172 Layout Text, 345	Requirements Cadence Virtuoso, 202 Calibre, 16 Calibre DESIGNrev, 305 design tools, 16 general, 16
License environment variable, 336 timeout, 78 — M — MentorMessengerLib, 204	rule file, 16 Synopsys Custom Compiler, 184 Synopsys Laker, 151 Rule check comments check text override file, 101
ivicino iviesse igerbio, 204	general, 90 RVE Highlight Color, 94

RVE Highlight Index, 92	layer mapping, 172
specifying, 91	option control with Tcl commands, 58
Run configuration, 19	Tcl commands for environment variables,
Run context, defined, 21	53
Run DRC in cell, 21	version support, 151
Run DRC in window, 21	
Run DRC on Edit, 21	—T—
Run mode	Tcl commands, environment variable control,
configuration run mode, 19	53
Run DRC in Window, 21	Tel commands, option control, 58
Run DRC on Edit, 21	Tooltips
Runset, using Calibre Interactive, 87	Cadence Virtuoso, 214
RVE Highlight Color, 94	Calibre DESIGNrev, 315
RVE Highlight Index, 92	Troubleshooting
RVE Link, 99	Cadence Virtuoso, 302
RVE Show Layers, 96	— U —
•	Underlined words, 22
— S —	Underfined words, 22
Shortcuts, see Keyboard shortcuts	_ V _
Skill commands for environment variable	Virtuoso, see Cadence Virtuoso
control, 295	,
Skill commands for option control, 249	— W —
Slanted words, 22	With Text, 346
Square parentheses, 22	
Supported design tools, 16	
SVRF statements	
Density, support, 19	
DFM Text, 342, 343	
Expand Text, 343	
Extent Cell, 342, 343	
Extent Drawn, 344	
Inside Cell, 344	
Layout Text, 345	
Not Inside Cell, 346	
Not With Text, 346	
restrictions, 339	
unsupported, 18	
With Text, 346	
Synopsys Custom Compiler	
compatibility table, 186	
requirements, 184	
viewing RealTime toolbar, 187	
Synopsys Laker	
circles and ellipses, 175	
edit-in-place, 160	
keyboard shortcuts, 179	
,	

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.</your_software_installation_location>

