

Guardian DRC vs. Other DRC Systems, I

1. Introduction

This article begins a series of articles that describe the compatibility of *Guardian* DRC system[1] in terms of functionality: command set, syntax and implementation with other leading industrial DRC systems. The purpose of this series is to facilitate the usage of *Guardian* DRC system by the designers who are already familiar with some other DRC systems.

Although basic DRC operations performed by various DRC systems correspond to “natural” operations possible with geometric shapes: Boolean operations, selection, resizing, distance checking, the implementations vary both in syntax and in the way of actual execution. This sometimes may lead to wrong expectations about certain functionalities, unless the designers are aware of these differences..

Guardian DRC system includes automatic DRC rule translators for several popular DRC formats. Nevertheless the knowledge of differences in commands may be useful not only when writing new DRC rules from scratch. For example, it may prove useful for optimization of rule sets, since automatic translation sometimes does not deliver the best possible implementation of certain checks, especially when converting from systems with limited command set. For example, Dracula(TM) DRC system has very limited capabilities to operate with separate edges from shape boundaries. Also, often it is necessary to put additional checks into the DRC deck, to comply with local technology variations and/or limitations of design methodologies..

The goal of *Guardian* DRC is to provide all the functionality required for performing modern DRC operations. When compared to other DRC systems, *Guardian* DRC either provides direct counterparts of commands or converts them into groups of several commands. In some cases this way opens possibilities for an additional optimization of execution time.

2. Compatibility with Calibre™ DRC System

This section is a general overview of the correspondence of Calibre and *Guardian* DRC commands. Detailed comparisons of particular commands will be provided in subsequent articles of the series.

2.1. Overall Execution and Syntax

The most important differences in DRC rule set execution are the following.

- Calibre DRC syntax allows layer generation statements to be written later in the DRC rule file, after their use in other commands. DRC rule compiler automatically defines proper order of execution of DRC commands. In *Guardian* DRC this possibility can be turned on by the setup directive “free_layer_definition_order”.

- Calibre merges original layers on input to the DRC system. In *Guardian* DRC this possibility can be turned on by the setup directive “merge_input”.
- The syntax of *Guardian* DRC is based on pairs “parameter-value”, and commands have an explicit terminator, a semicolon. This provides an additional flexibility in writing down the rule set, at the price of brevity.
- Newer versions of *Guardian* DRC have a simplified syntax for writing the constraints for numerical values in checks and selection operations.
- Usually *Guardian* DRC runs integrated into *Expert* Layout Editor [2]. By default, *Guardian* returns all layers, generated during DRC execution, unless their names start with the ampersand (&). This behavior may be flexibly controlled by the setup directive “Update_Layout”.
- DRC error reporting in *Guardian* DRC differs from that of Calibre. This will be covered in a separate article.
- Both Calibre and *Guardian* use layer names in DRC rule sets, that are independent of the names in a layout system. Calibre can define its layers as combinations of original layers. In *Guardian* DRC, to achieve the same functionality, the rule set must explicitly contain “OR” or “COMBINE” commands.

2.2. Particular Types of Commands

The commands are classified here in the following main groups: setup directives, connectivity/antenna checks, metric checks and all other operations. Although the Metric Checks group constitute only four commands, the total number of different actual operations performed by the set commands is comparable to the number of all commands, since metric checks are controlled by a large number of parameters.

2.2.1. Metric DRC Checks (See Table 1)

The “**Rectangle enclosure**” command of Calibre is in fact a more efficient implementation of enclosure checks for rectangular shapes. Currently *Guardian* DRC performs these operations by general means.

The way of actually performing measurements between line segments in *Guardian* DRC is described in detail in [3]. The described metrics correspond to those used in Calibre DRC system.

An important difference arises from the fact that *Guardian* DRC does not perform automatic merging of input layers. In certain cases this can lead to time optimization, if it is known that certain layers are created from non-overlapping shapes, e.g., contact or via layers. Therefore

to ensure consistency of metric check, one must be sure that the input layers are merged. Notice that many DRC operations with layers provide merged output.

CALIBRE	Guardian DRC	Note
External	OutDistance, Distance	+
Internal (two-layer)	OvDistance	+
Internal (single-layer)	Width	+
Enclosure	InDistance	+
Enclosure...outside also	CompDistance	
Rectangle Enclosure		-

Table 1. Metric DRC Checks

2.2.2. Parameters of Metric Checks (See Table 2)

Metric checks in Calibre and *Guardian* DRC differ in the way of the output of the results. This difference will be covered in detail in one of the subsequent articles.

Another difference is in the reporting error segments. The way Calibre does it corresponds to *Guardian's* option "Report error subsegments". This option may be set either for each DRC command individually, or globally, by a setup directive.

CALIBRE	Guardian DRC	Note
'[<layer>]', '(<layer>)'	ElayerR, CelayeR	[+](-)
region	Option=(RP)	=
region extents		-
region centerline		-
abut [range]	abut	+
singular	sing	+
overlap	cross	+
square	metric=square	=
opposite	metric=opposite	+
opposite symmetric	metric=symmetric	+/-
opposite extended <val>	metric=extended	+
measure coincident	T	=
measure all	(~)VI	-
notch space	N' no notch	+
connected not connected	C' C	=
paral also only not paral	- P P'	=
perp also only not perp	perp	+
acute also only not acute	acute	+
obtuse also only not obtuse	obtuse	+
projecting [range] not projecting	O[range], O'	+[-]
angled[range]	slanted	+
corner to corner[range]	C2C	+
corner to edge	C2E	+
[not] corner	C2CE, C2CE-	+
intersecting only	NoMeasure	+
inside also		-
outside also	(~)CompDistance	~+
exclude false notch	irrelevant	

Table 2. Parameters of Metric DRC Check Commands.

Legend:

- = already existed in Guardian DRC
- not implemented yet
- + added
- ~ difference in implementation exist

2.2.2. Connectivity/Antenna Checks (See Table 3)

Antenna checks of Calibre are converted into pairs of operations: obtaining of layer parameters and the subsequent checking of them. This way of implementation allows for optimization of execution time when the same parameters for certain layers are used in several antenna checks.

CALIBRE	Guardian DRC	Note
text layer	Attach_Labels	+
connect	Connect	=+
sconnect	Connect ... (S)	=+
disconnect	Connect_order::;	+
attach	Attach_Labels	=
label order	Label_Order	+
net, not net	Select...Label (C)	=
net area	Get_Node_Params, Check_Node_Params	+
net area ratio	Get_Node_Params, Check_Node_Params	+
net area ratio print	Check_Node_Params (LogR)	+
polynet	Connect (one-layer)	+
stamp	Select...stamp	=
ornet	Combine	+

Table 3. Connectivity/antenna DRC Checks

Legend:

- = already existed in Guardian DRC
- + added

2.2.3. Other DRC Operations (See Table 4)

The notion of "check" of Calibre rule files is converted into preprocessor "block" notion of *Guardian* DRC script file.

CALIBRE	Guardian DRC	Note
@	comment	+
include	#include	+
#define/#undefine	#define/#undefine	+
#ifdef/#ifndef	#ifdef/#ifndef	+
#else	#Else	+
#endif	#Endif	+
layer		~+
group		~+
flag acute, offgrid, skew, nonsimple (path)	input data checks	~+
snap offgrid	Snap_Input ~	+
and	And	+
or	Or	=
xor	Xor	=
not	Dif	=
inside	Select...Inside	=
outside	Select...Outside	=
cut	Select...Cut	=
enclose	Select...Enclose	=
touch	Select...Touch	=
interact	Select...Overlap	=
area	Parea	=
perimeter	Perimeter	+
vertex	Select...Vertex	=

rectangle	Contours	+
donut	Contours...holes	+
enclose rectangle	Contours	-
holes	Contours...opt=holes	+
size	Size...cut	+
size.. truncate	Size...cut	+
size..overlap only		-
size..overunder	underSize+oversize	+
size..underover	overSize+undersize	+
size..inside of	Size...within	+
size..outside of	Size...without	+
size..step	Size...step	+
grow	Size...T,B,L,R	=
shrink	Size...T,B,L,R	=
expand edge	Size_edge	+
exp edge.. corner fill	Opt=(Fill)	-
density	Density	+
extent [<layer>]	Substrate	=[+]
extents	boxes	+
extent cell		-
rectangles		-
expand text	size...list=()	+
[not] inside cell		-
inside edge	select_edges...Inside	+
outside edge	select_edges...Outside	+
coincident edge	select_edges...Coincide	+
coincident inside edge	select_edges...Coincide	+
coincident outside edge	select_edges...Coincide	+
touch edge	select_edges...Touch	+
touch inside edge	select_edges...Touch	+
touch outside edge	select_edges...Touch	+
with edge	Select..Touch	+
with text	Select...Label	+
convex edge	Select_edges...corners	+~
angle	Slope	+
length	Length	=
path length	Path_length	+
drawn acute, offgrid, skew	Check_*_Input: on;	+
offgrid	Offgrid	+
COPY		
copy	Copy	=
flatten	Copy	+
merge	Copy	+
push	Copy	+
shift, magnify, rotate	Copy	-
snap	copy	+

Table 4: Other DRC Operations

Legend:

- = already existed in Guardian DRC
- not implemented yet
- + added
- ~ difference in implementation exist

2.2.4. Setup Commands (See Table 5)

Guardian DRC provides the setup for execution of the DRC script in three ways: through user interface, through command line parameters of the batch mode of execution and by means of setup directives of the DRC script itself.

CALIBRE	Guardian DRC	Note
Precision	Database_Precision	+
Resolution	Grid_Resolution	+
layer resolution	Grid_Resolution	+
title	Title	+
unit length	Units	=
variable	Variable	+
polygon		
text		
layer map	Layers	+
layout primary	Top_cell	=
layout window	Window	+~
layout windel	Exclude_window	+~
layout window clip		-
exclude cell	Exclude_cells	=
virtual connect name		
virtual connect colon		
virtual connect box name		
virtual connect box colon		
virtual connect report		
layout text		
layout depth [primary all]	Setup: primitives whole	+
layout allow duplicate cells		-
expand cell		
expand cell text		
flatten cell		hier
flatten inside cell		hier
push cell		hier
hcell		hier
layout base cell		hier
layout rename cell		hier
layout process box record		UI
layout magnify		
layer directory		UI
layout error on input		
layout merge on input		
layout property text		
layout property placement name		
text print maximum		
layout rename text		
drc tolerance factor	tolerance	+
drc exclude false notch		hier
drc maximum results		UI
drc check text		
drc summary report		
drc results database	output_to_edb/cell	
drc maximum vertex		
drc keep empty		
drc select check	Exec	+
drc unselect check	skip	+
drc check map		
drc print area	larea	--
drc print perimeter		
drc map text		hier
drc incremental connect		+
drc cell text		hier
drc cell name		hier
drc magnify results		
drc results database precision	Output_Precision	--
drc maximum cell name length		

Table 5. DRC Setup Commands

Legend:

- = already existed in **Guardian** DRC
- not implemented yet
- + added
- ~ difference in implementation exist
- hier marks commands related to hierarchical execution, not supported in current implementation of **Guardian** DRC
- UI marks commands implemented only by means of user interface of **Guardian** DRC

3. References

- [1] *Guardian User's Manual*, Silvaco International.
- [2] *Expert and ExpertViews User's manual*, Silvaco International.
- [3] Measurements in Spacing Checks of Guardian DRC, *Simulation Standard*, vol. 13, no 9.