DKTOOLS
USER MANUAL

UNICAD 24 / May 2006

CENTRAL R & D
Design Automation and Integrated Systems

DKTOOLS
v1.8
User Manual

UNICAD™ © 1989 All rights reserved

UNICAD™ is a trademark of STMicroelectronics

The following names are trademarks, registered trademarks, and service marks of other companies that appear in Unicad publications:

Adobe, the Adobe logo,FrameMaker, the FrameMaker logo, Acrobat, the Acrobat logo, Exchange, and PostScript are registered trademarks of Adobe Systems Incorporated.

ADS, AHB, AMBA, PrimeXsys, STD are trademarks of ARM, ltd.

Alanza, Allegro, Ambit, Analog Artist, Assura, BuildGates, Cadence, Cadence Logo, CoBALT, Composer, Concept, Connection, Diva, Dracula, Envisia, Formal Check, Gate Ensemble, GDSII, HDL_ICE, MegaSim, Mercury, NC Verilog, Orcad, Orcad Capture, Orcad Layout, Palladium, Pearl, PowerSuite, PSpice, Q/Bridge, QPlace, Quest, Quickturn, Radium, Transaction Explorer, Test Builder, Verification Cockpit Vampire, Verilog, Verilog XL, Verifault, Verifault XL, NC-Verilog, Cell3 Ensemble, Silicon Ensemble, Virtuoso, Spectre, and Leapfrog are trademarks and registered trademarks of Cadence Design Systems, Inc.

MemMaker, PureView, PureData are trademarks of Denali, Software Inc.

ApolloII, ApolloGAII, Apollo-DPII, Aurora, ASIC Synthesizer, AvanTestchip, AvanWaves, ChipPlanner, Columbia, Columbia-CE, Cyclelink, Davinci, DFM Workbench, Driveline, Dynamic Model Switcher, Electrically Aware, Enterprise, Evaccess, Hercules, Hercules-Explorer, HotPlace, HSPICE, HSPICE-LINK, Jupiter, LTL, Libra-Passport, Lynx, Lynx-LB, Lynx-VHDL, Mars, Mars-Rail, Mars-Xtalk, MASTER Toolbox, Medici, Michelangelo, Milkyway, Optimum Silicon, Passport, Pathfinder, Planet, Planet-PL, Planet-RTL, Polaris, Polaris-CBS, Polaris-MT, Progen, Prospector, Raphael, Raphael-NES, Saturn, Sirius, Silicon Blueprint, Smart Extraction, Solar, Solar II, Star, Star-Sim, Star-Hspice, Star-HspiceLink, Star-DC, Star-RC, Star-RCXT, Star-Power, Star-Time, Star-MTB, Star-XP, Taurus, Taurus-Device, Taurus-Layout, Taurus-Lithography, Taurus-OPC, Taurus-Process, Taurus-Topography, Taurus-Visual, Taurus-Workbench, TimeSlice, and TSUPREM-4 are trademarks of Avant! Corporation. Avant!, Avant! logo, AvanLabs, and avanticorp are trademarks and service-marks of Avant! Corporation.

ASAP, Aspire, C-FAS, CMPI, Eldo-FAS, EldoHDL, Eldo-Opt, Eldo-UDM, EldoVHDL, Eldo-XL, Elga, Elib, Elib-Plus, ESim, Fidel, Fideldo, GENIE, GENLIB, HDL-A, MDT, MGS-MEMT, MixVHDL, Model Generator Series (MGS), Opsim, SimLink, SimPilot, SpecEditor, Success, SystemEldo, VHDeLDO and Xelga are registered trademarks of ANACAD

Eldo, Calibre, ModelSim, FastScan, DFTAdvisor, Seamless, CVE, XRAY are registered trademarks of Mentor Graphics Corporation.

Debussy is a trademark of Novas Software, Inc.

DesignSync, DesignSync DFII, ProjectSync, IPGear are registered trademarks of Synchronicity Software, Inc.

FLEXlm is a trademark of Globetrotter Software, Inc.

HSPICE is a registered trademark of Meta-Software, Inc.

IKOS and Voyager are registered trademarks of IKOS Systems, Inc.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc.

NeoCell, NeoCircuit are registered trademarks of Neolinear, Inc.

Netscape is a trademark of Netscape Communications Corporation.

SPARC is a registered trademark, and SPARCstation is a trademark, of SPARC International, Inc. Sun Microsystems, Sun Workstation, and NeWS are registered trademarks of Sun Microsystems, Inc.

Sun, Sun-2, Sun-3, Sun-4, OpenWindows, SunOS, SunView, NFS, and NSE are trademarks of Sun Microsystems, Inc.

Synopsys, Design Compiler, Physical Compiler, DFT Compiler, Test Compiler, Power Compiler, Module Compiler, Library Compiler, PrimeTime, Formality, Arcadia, Liberty, TetraMAX, LEDA, PathMill, PowerMill, TimeMill, Chip Architect and VSS are trademarks or registered trademarks of Synopsys, Inc.

UNIX, Solaris and OPEN LOOK are registered trademarks of UNIX System Laboratories, Inc.

Verisity is a trademark of Verisity Ltd. or its subsidiaries (Verisity),registered in the United States and other jurisdictions. The Verisity logo, Specman, Specview, Specman Elite, Verification Advisor, eVC, Pure IP, InvisibleSpecman, SureLint, Lintrpt, SureCov, and SureSight are trademarks of Verisity.

Versatec is a trademark of Xerox Engineering Systems, Inc.

Fire & Ice is a registered trademark, and ClockStorm, ElectronStorm, VoltageStorm, SubstrateStorm, IceCaps, Rain, SI Report, QIC Engine, 3D Adaptive Analytical Extraction, Accura, PGS Exploration, PerturbingPath, NetConnect, and SoC Design Foundry are trademarks of Simplex Solutions, Inc.

TransEDA, VeriSure, VHDLCover, HDLCover, CoverPlus, StateSure, Verification Navigator, State Navigator, TransEDA Verification Navigator and TransEDA State Navigator are trademarks of TransEDA Limited.

Other brand or product names that appear in Unicad publications are trademarks or registered trademarks of their respective holders.

TABLE OF CONTENTS

1 - Introduction **3**

1.1 Foreword 3

2 - Schematic Tools **4**

2.1 Select Current Mirror 4

3 - Layout Tools **5**

3.1 Power Supply Tool 5
3.2 Array Tools 6
3.2.1 General Description 6
3.2.2 Interface Overview. 7
3.2.3 Array Creation Algorithm 10
3.3 Group Tools 11
3.4 Lock Tools 11
3.4.1 Lock Selected Shapes 12
3.4.2 Unlock from selection 12
3.4.3 Select locked to selection. 12
3.5 Symmetry Tools 12
3.5.1 Symmetrical Copy 13
3.5.2 Flip functions: horizontal & Vertical. 13
3.6 Add/Update Labels 13
3.7 Add/Delete Obstruction 14
3.8 LSW Finder 14
3.9 LSW Sizer 14
3.10 Simple Abstract generator 14

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

1 - INTRODUCTION

1.1 Foreword

The DKTOOLS Module provides a set of tools, integrated in Cadence Opus framework, as well as environment settings, that enables the user to have access Layout/Schematic Aid for productivity.

This tool is based mainly on use of Virtuoso-XL, even if some tools are available too in layout standard environment. All tools are accessible through the *Design Kit menu.

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

2 - SCHEMATIC TOOLS

2.1 Select Current Mirror

This function is associated to schematic bindkey ‘s’. When a MOS or Bipolar (recognized by the name of their pins: D,S,G,B or E,C,B) is selected while applying this command, the tool will highlight all MOS or Bipolars which are recognized as belonging to the same current mirror - i.e. they all have G (B) pin in common, as well as D (E) or S (C)

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

3 - LAYOUT TOOLS

3.1 Power Supply Tool

This tool create power stripes on the top/bottom (or left/right) part of a rectangular boundary. Two command are proposed :

- create : to create the stripes. Use it the first time
- update : to update the position of the stripes when the boundary as moved

When launching stripes creations, the following form appears:

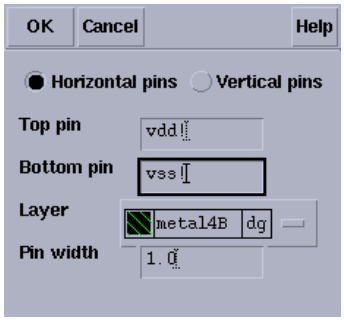

Figure 1:Array Tool interface

User must specify :

- the name of the two power pins
- whether stripes must be horizontal (on top/bottom part of the boundary) or vertical (on left/rigth part of the boundary)
- the layer and the width of the pins

Warning: pins must connectively exists before using this tool.

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

3.2 Array Tools

3.2.1 General Description

This tool enables the user to quickly create regular arrays of devices. These arrays can be made of any number of components, of any size - however, it is designed for creating “matched” components arrays, i.e. arrays of identical components - may they be MOS, resistors, capacitors, bipolars, etc....

Array Tool is able to add ‘dummy’ components on side of arrays, as well as filling a ‘non-rectangular’ array with ‘dummy’ components. It also provides a way to connect these dummy components to a specified net.

For any array, X & Y spacing can be specified independently:

- by setting a ‘layer-to-layer’ spacing
- by setting a ‘bBox’ spacing

The Array Tool works on selected instances only: a message will be generated and nothing will happen if less than 2 instances are selected.

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

3.2.2 Interface Overview

Array tool provides a unique interface:

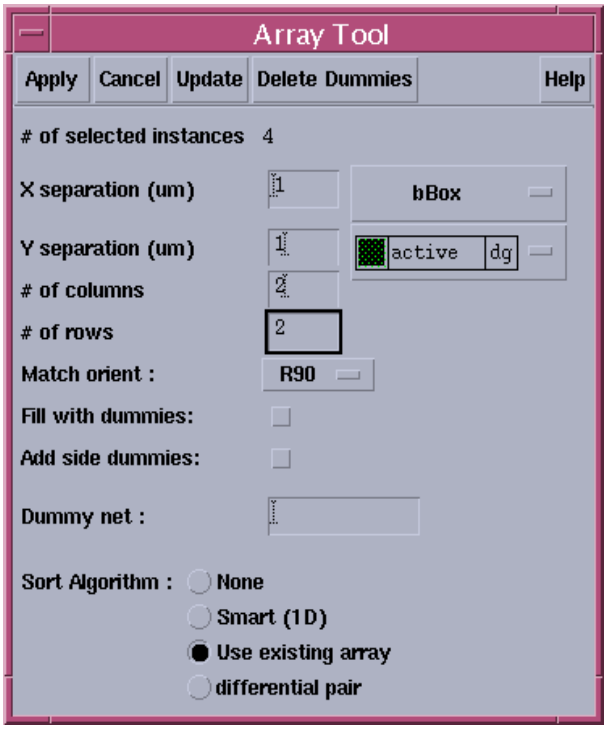

Figure 1: Array Tool interface

Here is a detailed description of each field, as well as its influence in the array tool behaviour:

# of selected instances: displays the number of instances that were selected when the array tool was launched. Only 'instances' (with Cadence DFII meaning are taken into account). Clicking on the '**Update**' button will update all fields (including this one) according to the current selected instances.

NOTE: 'dummy' instances are **not** taken into account in instances number.

X separation (um): distance between 2 instances of the array, on the X axis. The distance can be specified either as a distance between the closest shapes on a specific layer, or as the distance between two instances bBox.

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

Y separation (um): distance between 2 instances of the array, on the Y axis.

NOTE: displayed layers are retrieved from the layers that exist in selected instances. However, for performance reasons, only layers that are present on level 0 of each instance are taken into account. This means that you cannot use a 'sub-layer' (e.g. contact in most MOS pCells) for alignment reference.

# of columns: number of columns of resulting array. This field will not be taken into account if sort algorithm is '**Use existing array**'.

# of rows: number of rows of resulting array. This field will not be taken into account if sort algorithm is '**Use existing array**'.

Match Orient: orientation of all instances in resulting array.This field will not be taken into account if sort algorithm is '**Use existing array**'.

Fill with dummies: if (*# of columns * # of instances)* > *# of instances* and this field is checked, the array will be completed with dummy instances, otherwise, remaining space will be left empty. This field will not be taken into account if sort algorithm is '**Use existing array**'. If not all selected instances share the same parameters (w, l, r, c, orient) this field will be greyed & disabled.

NOTE: when you select some identical instances to align, but which orient is not matched, here is the procedure to follow to "fill" with dummies this array:

fill all other needed fields (except *fill with dummies*, which is disabled)

Click "Apply" and check the array is OK

Click "Update" ==> the '*fill with dummies*' field is now enabled

Tick the '*fill with dummies*' field

Click "Apply"

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

Add side dummies: if selected, will add dummy instances on two sides of the array. The sides are either top/bottom or left/right of the array. It will choose which side to use, depending on X size and Y size of the 'unitary' instance of the array. if X size > Y size, it will add top/bottom dummies, if not, left/right dummies will be added. It is not possible to modify this behaviour.

NOTE: '*Add side dummies*' needs *Fill with dummies* to be selected to work. If '*Fill with dummies*' is not selected, it will automatically tick it. If you un-select '*Fill with dummies*', '*Add side dummies*' will be automatically un-selected.

Dummy net: when this field is not empty, it will search for a net with this name in cellview, and when it founds it, will connect all dummy instances terminals to this net (**VXL only**)

'*Delete dummies*' button: it will delete all dummies that were generated for the array.

IMPORTANT: if you create an array with lots of components and dummies, and decide then to split this big array into several smaller arrays, all dummies will be destroyed even if you only want to destroy dummies for one of the smaller arrays. This bug can be corrected by typing following skill code in CIW:

mapcar('dbDeleteObject setof(x getEditRep()~>groups x~>name == "U2dktarray"))

However, this will dissociate all existing dummies to their respective arrays.

Sort Algorithm: this list enables to choose between 3 algorithms that modify the instances arrangement in resulting array.

- *None*: order will be random - it only depends on the order the selection was made.
- *Smart (1D)*: this will create a 1D distribution of all instances. First, all instances will be split into 'groups', that gather instances coming from the same schematic master (and which were divided because of m or s parameter). Each 'group' is then sorted, by order (R1.s1 --> R1.s10 for example). At last, beginning with the biggest 'group', each group will be inserted & distributed in the final 1D array. For resistors, a permutation algorithm is the run, so that there is not need to 'flip' resistors to connect series resistors.
- *Use Existing Array :* this will create distribution based on an approximate one and a modified one.

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

- *differentiel pair :* this will create a distribution special optimised for mos differential pair

NOTE: for resistor groups with an even number of members, that last resistor permutation will not be correct. Indeed, this is a feature that works-around a Cadence bug (PCR #480123), involving permute pins & ECO - and that should be corrected staring from Opus 4.4.6.100.100 (needs to be verified)

- *Use existing array*: this is not really a 'sorting' algorithm, it is an 'array builder'. When this option is selected, the array tool will try to 'build' an array, using a existing array on layout: It takes the lower left instance of all selected instances, and tries to construct a first line & a first column with selected instances. Then, it tries to 'put' one of all other selected instances in every empty place of the array. It relies on the instances centres, but has a tolerance margin on alignment, which is equal to 1/4 of cell width or length. If algorithm does not manage to 'construct' the array, it tells the user.

A recommended methodology is to use the 'Smart' Algorithm at first, then manually 'swap' instances that are not well-placed (Ctrl-M in VXL), and then, if array "spacings" need to be modify, use the 'use existing array' sort algorithm.

3.2.3 Array Creation Algorithm

Array filling is done: from bottom to top, and then from left to right. If you have selected 10 instances and ask for a 3-columns / 4-lines array, here will be the result:

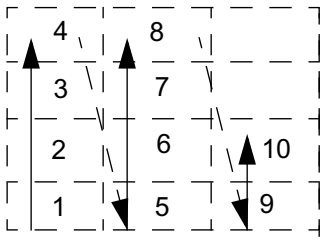

Figure 2: array creation

If dummies are requested, they will be inserted in the empty spaces.

Because 'Smart Sort' is a 1D algorithm, if you use it on a 2D array, it will be 'folded' into the 2D array: in this example, distribution is done on 10 elements, without taking into account the final array size, but just like if it was a 1x10 or a 10x1 array.

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

Please note that if you use a ‘Use selected array’ with an array that contains ‘holes’, the algorithm will not keep these ‘holes’, and result will not correspond to original array: to do so, manually add a dummy in the ‘hole’, use the array tool, and then manually remove the dummy instance.

3.3 Group Tools

This tool allows to manage any cadence groups and particulary groups that are created when using the ’Array Tools’. The whole group can be selected from a single instance using ’shift+G’ binkey or ‘*Design Kit -> Group Tools -> Select Group’.

The ’group Manager’ allows to split and merge groups.

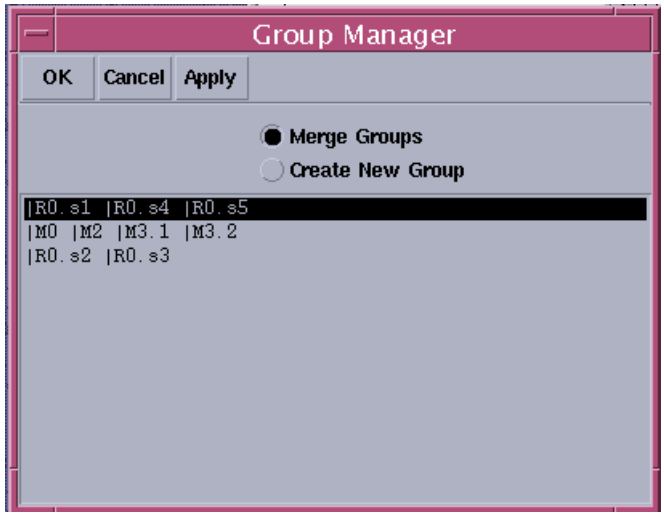

Figure 3: Group Manager

To merge groups, select at least two groups and ’Merge group’ command. Then click on ’apply’. To split a group, select it and choose ’Create New Group’ command. Then click on ’apply’. A new form will be displayed to select instances that will be in the new group.

3.4 Lock Tools

This set of tools, accessible through ‘*Design Kit -> (Un)Lock Tools’, will basically create a so -called ‘ROD alignment’ between all selected objects (whatever their ‘type’), which means that when objects are aligned together, their relative position is fixed: if you move one object, all “aligned” objects will move with the same translation vector.

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

IMPORTANT: object are aligned on a ‘xy-to-xy’ (reference point) basis: if you change the orient of an instance to rotate it, its reference point does not move.

3.4.1 Lock Selected Shapes

This menu enables to create a lock: select whatever you want to lock, then click on ‘lock selected shapes’: everything is locked together.

If you move any of the shapes, all ‘locked’ shapes will be automatically moved to respect the lock (after the move is completed).

3.4.2 Unlock from selection

To unlock shapes, select any of the locked shapes, and click on ‘Unlock from selection’: it will remove all the alignments directly or indirectly associated to this instance.

IMPORTANT: if you have attached A to B, and then C to D, and then E to F. You have made three independent, locked, groups. If you attach B to C, then A,B,C,D are locked together. If you select A and do ‘Unlock from selection’, ALL instances attached to A, directly or indirectly will be unlocked: in this case, A,B,C,D will be unlocked - and the link from C to D will have disappeared, which can be a surprising behaviour.

3.4.3 Select locked to selection

When you select a shape and click on ‘Select locked to selection’ all shapes that are directly or indirectly linked (See “Unlock from selection” on page 12.) to this shape will be selected. This is a good way to see dependencies between locked objects, and to avoid errors when doing an ‘Unlock from selection’.

3.5 Symmetry Tools

These tools are accessible through ‘*Design Kit -> Symmetry tools’ menu. Basically 2 types of symmetry functions are available:

- copy shapes relatively to a symmetry axe
- flip instances.

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

3.5.1 Symmetrical Copy

To do a symmetrical copy, a symmetry axe should be created first: use ‘Create Symmetry Axe’ menu, that will propose you to draw a symmetry line.

The tool will let you create lines with any angle (depending on your snap mode), but you will only be able to use *horizontal & vertical* symmetry axes. The line will be drawn in ‘text drawing2’ layer and a text: “symmetry axe” will be attached to this line. Please note that a symmetry line can be copied, moved, rotated... without affecting its properties.

To copy shapes relatively to a symmetry axe, use ‘copy shapes with symmetry’ menu:

- method 1: before invocation, select all shapes to copy, and 1 symmetry axe.
- method 2: before invocation, select all shapes to copy. The tool will then ask you to click on a symmetry line.

All ‘new’ shapes will be added to the selection, which eases multiple copies.

Basically, the tool is useful within VXL for symmetrical routing, so it is forbidden to copy instances: only shapes (rectangles, polygons, path) & vias can be copied with this tool.

3.5.2 Flip functions: horizontal & Vertical

This function is basically used to emulate the VXL “permute pins” function that does not work correctly with ECO.

To use them, just select the instances to flip, and click on ‘flip selected horizontal’ or ‘flip selected vertical’

3.6 Add/Update Labels

This function will:

- If no label (text drawing) exists on the instance, create a label, referring to the master name of instance (e.g., for a resistor R1 split in 2 thanks to ‘s’ parameter, the 2 generated instances will be named ‘|R1.s1’ and ‘|R1.s2’. But their label will be ‘R1’). Its orientation will be ‘R0’ and it size will be calculated so as to cover 1/3 of smallest side size.
- If a label already exists, but instance name has changed, it will re-create the label name
- If label was rotated, it will put it back to R0
- if instance was moved, it will re-centre the label on new instance location

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

Update will not change the label size: you can manually modify size of label to fit your needs. Labels are attached to instances:if instance is moved, label will follow, but label can be moved without moving the instance.

3.7 Add/Delete Obstruction

The ‘Add/Delete’ obstructions enables add obstructions on selected MOS (e.g. differential pair MOS) so that ICC does not route over poly gates, but is enable, however to access metal pins of MOS.

Important : This tool is currently supported in CDBA only. Support of OpenAccess will be done only on demand.

3.8 LSW Finder

This menu enables to find quickly a purpose/layer in the LSW. If the layer is not visible, the tool adds it visible in the LSW windows.

3.9 LSW Sizer

Resize automatically the LSW when you run icfb, to have a better visibility on layer/purpose name. It uses the following function but with a trigger:

leSetLSWBBox(list((10:200) (260:900)))

No menu available, only started at the beginning of the icfb session.

3.10 Simple Abstract generator

The purpose of this small tool is to be able to generate simple abstract from a layout view.

To generate an abstract, launch the tool from a layout view. Immediatly, an abstract will be generated in the same cell but with viewName equals to a string property U2DKTabstractName that must be placed on the cellview or "abstract" if the property does not exists.

Requirement on the layout are :

- a unique shape in ’prBoundary’ ’boundary’ must exists.It can be rectilinear
- metal block must also exists in layout.

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics

- pins must be specified with cadence pins and not only labels

Important : The methology either in digital or mixed-signal flow for pins and blockage allows to overlap blockage with pins as long as either one edge of the pin is aligned on the edge of the boundary or no obstruction exists over the pin and the pin is big enouh to put a via inside.

The generated abstract will contains :

- the block boundary specified in the layout and the overlap shape if it exists.
- the copy of all pins that are in the design. The pins will be in the same layer name that the original one but in purpose "net".
- routing blockage that are in layout.
- all properties of the layout initial view and all properties on terminals and pins. This includes pin or cell abstract properties set trough cadence abstract menu.

When not existing in layout, the minimum required properties for LEF files will be automatically created :

propery name	default value
prCellClass	block
prCellType	macro
maskLayoutSubType	abstract (value is mandatory)

Table 4: abstract LEF default properties value

For a complete detail of mapping between LEF and dfII, refers to cadence documentation : "Design Data Translator's Reference", chapitre 7 "*Translating LEF and DEF Files*", "*LEF Data Map*".

© STMicroelectronics - Reproduction and Communication of this document is strictly prohibited unless specifically authorized in writing by STMicroelectronics