What’s New in LEF/DEF 5.8

Product Version 5.8
May 2017

© 2012 Cadence Design Systems, Inc. All rights reserved.
Printed in the United States of America.

Cadence Design Systems, Inc., 555 River Oaks Parkway, San Jose, CA 95134, USA

Trademarks: Trademarks and service marks of Cadence Design Systems, Inc. (Cadence) contained in this document are attributed to Cadence with the appropriate symbol. For queries regarding Cadence's trademarks, contact the corporate legal department at the address shown above or call 800.862.4522.

Open SystemC, Open SystemC Initiative, OSCI, SystemC, and SystemC Initiative are trademarks or registered trademarks of Open SystemC Initiative, Inc. in the United States and other countries and are used with permission.

All other trademarks are the property of their respective holders.

Restricted Print Permission: This publication is protected by copyright and any unauthorized use of this publication may violate copyright, trademark, and other laws. Except as specified in this permission statement, this publication may not be copied, reproduced, modified, published, uploaded, posted, transmitted, or distributed in any way, without prior written permission from Cadence. This statement grants you permission to print one (1) hard copy of this publication subject to the following conditions:

1. The publication may be used solely for personal, informational, and noncommercial purposes;
2. The publication may not be modified in any way;
3. Any copy of the publication or portion thereof must include all original copyright, trademark, and other proprietary notices and this permission statement; and
4. Cadence reserves the right to revoke this authorization at any time, and any such use shall be discontinued immediately upon written notice from Cadence.

Disclaimer: Information in this publication is subject to change without notice and does not represent a commitment on the part of Cadence. The information contained herein is the proprietary and confidential information of Cadence or its licensors, and is supplied subject to, and may be used only by Cadence's customer in accordance with, a written agreement between Cadence and its customer. Except as may be explicitly set forth in such agreement, Cadence does not make, and expressly disclaims, any representations or warranties as to the completeness, accuracy or usefulness of the information contained in this document. Cadence does not warrant that use of such information will not infringe any third party rights, nor does Cadence assume any liability for damages or costs of any kind that may result from use of such information.

Restricted Rights: Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor.

Contents

About This Manual 5
How This Document Is Organized 5
Related Documents 5

1
New Features 7
Overview 8
Multi-Mask Patterning Technology 8
New LEF Features 9
LEF Layers Support Masks 9
Fixed Masks for Layers and Macros 9
LEF Via Adds Masks for Fixed Vias 9
Macro Layer Geometries Support Masks 10
New DEF Features 12
Routing Blockages Add Mask Information 12
Component Mask Shift Multi-Patterning Layers 12
Components Adds Mask Shift Information 12
Fills Adds Masks 13
Nets Routing Points Enhancements 13
Special Nets Special Wiring Enhancements 14
DEF Via Adds Mask 15
Tracks Adds Mask 16
Pins Adds Mask 16
Blockages Change Component, Pushdown, Exceptpgnet to be Independent And Not Exclusive 17

© 2012 All Rights Reserved.

© 2012 All Rights Reserved.

About This Manual

This document provides information on new and changed features for version 5.8 of the Cadence® Library Exchange Format (LEF) and Design Exchange Format (DEF) integrated circuit description languages.

How This Document Is Organized

This *What's New* document is organized into the following chapters:

- New Features

  This chapter describes features that were added or changed since version 5.7 of LEF and DEF.

  New features are those that introduce new functionality into LEF and DEF. Any enhancements made to existing statements to support a new feature are also described in this chapter.

Related Documents

The following documents provide detailed information about LEF and DEF, and the LEF and DEF application programming interfaces.

- *DEF C/C++ Programming Interface (Open Licensing Program)*
- *LEF C/C++ Programming Interface (Open Licensing Program)*
- *LEF/DEF Language Reference*
- *What's New in DEF C/C++ Programming Interface*
- *What's New in LEF C/C++ Programming Interface*

1/20/17

© 2012 All Rights Reserved.

© 2012 All Rights Reserved.

1

New Features

- Overview on page 8
  - Multi-Mask Patterning Technology on page 8
- New LEF Features on page 9
  - LEF Layers Support Masks on page 9
  - Fixed Masks for Layers and Macros on page 9
  - LEF Via Adds Masks for Fixed Vias on page 9
  - Macro Layer Geometries Support Masks on page 10
- New DEF Features on page 12
  - Routing Blockages Add Mask Information on page 12
  - Component Mask Shift Multi-Patterning Layers on page 12
  - Components Adds Mask Shift Information on page 12
  - Fills Adds Masks on page 13
  - Nets Routing Points Enhancements on page 13
  - Special Nets Special Wiring Enhancements on page 14
  - DEF Via Adds Mask on page 15
  - Tracks Adds Mask on page 16
  - Pins Adds Mask on page 16
  - Blockages Change Component, Pushdown, Exceptpgnet to be Independent And Not Exclusive on page 17

© 2012 All Rights Reserved.

Overview

Multi-Mask Patterning Technology

The multi-mask patterning technology allows you to define multi-mask patterns for layers. You can define multi-mask statements that split a single layer into two or three masks. The syntax enhancements made to LEF/DEF v5.8 support more than three masks, though most applications support two or three masks only.

The subsequent sections describe the new syntax changes that have been made to incorporate this new feature.

© 2012 All Rights Reserved.

New LEF Features

The following sections describe the enhancements that have been made to support multi-mask patterning for LEF.

LEF Layers Support Masks

The MASK statement was added to specify how many masks for double- or triple-patterning will be used for a specified layer. You can use this statement for layers - cut, routing, implant, overlap, and/or masterslice.

Includes the following new syntax (in bold):

```
LAYER layerName
TYPE ROUTING ;                #or TYPE CUT, IMPLANT, MASTERSLICE, OVERLAP
[MASK numMasks ;]
```

Fixed Masks for Layers and Macros

Some technologies do not allow mask shifting for cells using multi-mask patterning, e.g., pin and routing shapes are all pre-colored and must not be shifted to other masks. The FIXEDMASK statement was added to indicate that mask shifting is not allowed. All the LEF macro pin mask assignments are kept fixed and cannot be shifted to a different mask.

Includes the following new syntax (in bold):

For layers:

```
FIXEDMASK ;
LAYER <statement>
```

For macros:

```
MACRO macroName
    [CLASS className [subclassName] ;]
    [FIXEDMASK ;]
    ...
```

LEF Via Adds Masks for Fixed Vias

For fixed vias - rectangle or polygon - the masks in the via-master for cut layer are required for multi-cut vias. The VIARULE statement was enhanced to include masks. The MASK

© 2012 All Rights Reserved.

statement indicates which mask for double- or triple-patterning lithography is to be applied to the shapes defined using the RECT or POLYGON statements of the via master.

Includes the following new syntax (in bold):

```
VIA viaName [DEFAULT]
    { VIARULE viaRuleName ;
        CUTSIZE xSize ySize ;
        LAYERS botMetalLayer cutLayer topMetalLayer ;
        CUTSPACING xCutSpacing yCutSpacing ;
        ENCLOSURE xBotEnc yBotEnc xTopEnc yTopEnc ;
        [ROWCOL numCutRows numCutCols ;]
        [ORIGIN xOffset yOffset ;]
        [OFFSET xBotOffset yBotOffset xTopOffset yTopOffset ;]
        [PATTERN cutPattern ;]
    }
    | [RESISTANCE resistValue ;]
      {LAYER layerName ;
        { RECT [MASK maskNum] pt pt ;
        | POLYGON [MASK maskNum] pt pt pt ...; } ...
      }...
  }
    [PROPERTY propName propVal ;] ...
END viaName
```

Macro Layer Geometries Support Masks

Macro layer geometries support multi-mask patterning. Three different masks can be added to the same layer geometries. The MASK statement was added to indicate which mask from double- or triple-patterning is to be used for path, rectangle, polygon, or via geometries.

Includes the following new syntax (in bold):

```
{ LAYER layerName
        [EXCEPTPGNET]
        [SPACING minSpacing | DESIGNRULEWIDTH value] ;
    [WIDTH width ;]
    { PATH [MASK maskNum] pt ... ;
    | PATH [MASK maskNum] ITERATE pt ... stepPattern ;
    | RECT [MASK maskNum] pt pt ;
    | RECT [MASK maskNum] ITERATE pt pt stepPattern ;
    | POLYGON [MASK maskNum] pt pt pt pt ... ;
    | POLYGON [MASK maskNum]  ITERATE pt pt pt pt ... stepPattern ;
```


© 2012 All Rights Reserved.

```
    } ...
  | VIA [MASK viaMaskNum] pt viaName ;
  | VIA ITERATE [MASK viaMaskNum] pt viaName stepPattern ;
  } ...
```

For syntax information, see "LEF Syntax" chapter in the *LEF/DEF Language Reference*.

© 2012 All Rights Reserved.

New DEF Features

The following sections describe the enhancements that have been made to support multi-mask patterning for DEF.

Routing Blockages Add Mask Information

The BLOCKAGES section was enhanced to include masks. The MASK statement indicates which mask for double- or triple-patterning lithography is to be used for the specified shapes.

Includes the following new syntax (in bold):

```
[BLOCKAGES numBlockages ;
    [- LAYER layerName
        [+ COMPONENT compName | + SLOTS | + FILLS | + PUSHDOWN
           | + EXCEPTPGNET]
        [+ SPACING minSpacing | + DESIGNRULEWIDTH effectiveWidth]
        [+ MASK maskNum]
           {RECT pt pt | POLYGON pt pt pt ...} ...
    ;] ...
END BLOCKAGES]
```

Component Mask Shift Multi-Patterning Layers

The COMPONENTMASKSHIFT statement defines which layers of a component are allowed to be shifted from the original mask colors in the LEF. This can be useful to shift all the layers of a specific component in order to align the masks with other component or router mask settings to increase routing density. This definition allows a specific component to compactly describe the mask shifting for that component.

Includes the following new syntax:

```
[COMPONENTMASKSHIFT layer1 [layer2 ...] ;]
```

Components Adds Mask Shift Information

Some applications require the ability to “shift” the mask of pins and obstructions on a certain layer for a specific component to better align the pin masks with the routing masks.

The MASKSHIFT statement was added to allow shifting of the cell-master masks used in double- or triple-patterning for specific layers of an instance of the cell-master.

© 2012 All Rights Reserved.

Includes the following new syntax (in bold):

```
COMPONENTS numComps ;
    [- compName modelName
        [+ EEQMASTER macroName]
        [+ SOURCE {NETLIST | DIST | USER | TIMING}]
        [+ {FIXED pt orient | COVER pt orient | PLACED pt orient
             | UNPLACED} ]
        [+ MASKSHIFT shiftLayerMasks]
        [+ HALO [SOFT] left bottom right top]
        [+ ROUTEHALO haloDist minLayer maxLayer]
        [+ WEIGHT weight]
        [+ REGION regionName]
        [+ PROPERTY {propName propVal} ...]...
    ;] ...
END COMPONENTS
```

Fills Adds Masks

In order to support multi-patterning technologies up to three different masks for the same layer can be added.

The MASK statement was added to specify which mask for double- or triple-patterning lithography is to be applied to the specified rectangles, polygons, or via shapes on each layer.

Includes the following new syntax (in bold):

```
[FILLS numFills ;
    [- LAYER layerName [+ MASK maskNum] [+ OPC]
        {RECT pt pt
        | POLYGON pt pt pt ...} ... ;] ...
    [- VIA viaName [+ MASK viaMaskNum] [+ OPC] pt ... ;] ...
END FILLS]
```

Nets Routing Points Enhancements

Nets Adds Rectangle and/or Virtual Connection

The NETS section was enhanced to provide the ability to create a rectangle or add a virtual connection to maintain the symbolic routing graph.

© 2012 All Rights Reserved.

The RECT statement allows creating a rectangle from the previous routing points. The VIRTUAL statement indicates that there is a virtual connection between the previous point and the new routing point.

Includes the following new syntax (in bold):

```
{ ( x y [extValue] )
    { ( x y [extValue]  )
    | viaName [orient]
    | RECT  ( deltax1 deltay1 deltax2 deltay2 )
    | VIRTUAL  ( x y ) } } ...
```

Nets Adds Masks

The NETS section routing points was enhanced to add masks - up to three different masks for the same layer can be added.

Includes the following new syntax (in bold):

```
{ ( x y [extValue] )
    [MASK maskNum] ( x y [extValue]  )
    | [MASK viaMaskNum] viaName [orient]]
    | [MASK maskNum] RECT  ( deltax1 deltay1 deltax2 deltay2 )
    | VIRTUAL  ( x y ) } } ...
```

Special Nets Special Wiring Enhancements

To support multi-mask patterning up to three different masks for the same layers can be added to special nets.

A new VIA statement was added to make it more efficient to specify several power-vias.

Additionally, the route status - cover, fixed, routed, shield, and shape - can be applied to polygons, rectangles, and vias.

Includes the following new syntax (in bold):

```
[ [+ COVER | + FIXED | + ROUTED | + SHIELD shieldNetName]
    [+ SHAPE shapeType] [+ MASK maskNum]
    + POLYGON layerName pt pt pt...
    | + RECT layerName pt pt
    | + VIA viaName [orient] pt ...
| {+ COVER | + FIXED | + ROUTED | + SHIELD shieldNetName}
    layerName routeWidth
```


© 2012 All Rights Reserved.

```
        [+ SHAPE shapeType]
        [+ STYLE styleNum]
        routingPoints
[NEW layerName routeWidth
    [+ SHAPE shapeType]
    [+ STYLE styleNum]
    routingPoints
    ] ...
] ...
```

New syntax for routing points:

```
(x y [extValue] )
    { [MASK maskNum]  (x y [extValue] )
    | [MASK viaMaskNum] viaName [orient]
        [DO numX BY numY STEP stepX stepY]
 }...
```

DEF Via Adds Mask

For fixed vias, rectangle or polygon, the masks in the via-master for cut-layer shapes are required for multi-cut vias. The MASK statement indicates which mask for double- or triple-patterning lithography is to be applied to the defined shapes.

Includes the following new syntax (in bold):

```
[VIAS numVias ;
    [- viaName
       { + VIARULE viaRuleName
          + CUTSIZE xSize ySize
         + LAYERS botmetalLayer cutLayer topMetalLayer
         + CUTSPACING xCutSpacing yCutSpacing
         + ENCLOSURE xBotEnc yBotEnc xTopEnc yTopEnc
         [+ ROWCOL numCutRows NumCutCols]
         [+ ORIGIN xOffset yOffset]
         [+ OFFSET xBotOffset yBotOffset xTopOffset yTopOffset]
         [+ PATTERN cutPattern]
       |[ + RECT layerName [+ MASK maskNum] pt pt
         | + POLYGON layerName [+ MASK maskNum] pt pt pt] ...
         }
    ;] ...
END VIAS]
```


© 2012 All Rights Reserved.

Tracks Adds Mask

The routing tracks can now have mask assignments to aid alignment of routing masks inside a block with the level above. The MASK statement indicates which mask for double- or triple-patterning lithography to use for the first routing track.

Includes the following new syntax (in bold):

```
[TRACKS
    [{X | Y} start DO numtracks STEP space
         [MASK maskNum [SAMEMASK]]
         [LAYER layerName ...]
   ;] ...]
```

Pins Adds Mask

The PINS syntax has been modified in include mask attributes. The MASK statement indicates which mask from double- or triple-patterning to use for pins or via shapes on each layer.

Includes the following new syntax (in bold):

```
[PINS numPins ;
[ [- pinName + NET netName]
...
    [[+ PORT]
        [+ LAYER layerName
              [MASK maskNum]
              [SPACING minSpacing | DESIGNRULEWIDTH effectiveWidth]
                   pt pt
         |+ POLYGON layerName
              [MASK maskNum]
              SPACING minSpacing | DESIGNRULEWIDTH effectiveWidth]
                   pt pt pt ...
         |+ VIA viaName
              [MASK viaMaskNum]
                   pt
         ] ...
         [+ COVER pt orient | FIXED pt orient | PLACED pt orient]
    ]...
; ] ...
END PINS]
```


© 2012 All Rights Reserved.

Blockages Change Component, Pushdown, Exceptpgnet to be Independent And Not Exclusive

The BLOCKAGES section was enhanced to allow a blockage to simultaneously have COMPONENT, PUSHDOWN, and EXCEPTPGNET statements with the other attributes. These statements are now independent and not exclusive. The meaning of the keywords is unchanged.

Includes the following syntax changes (in bold):

```
[BLOCKAGES numBlockages ;
[- LAYER layerName
    [+ SLOTS | + FILLS]
    [+ PUSHDOWN]
    [+ EXCEPTPGNET]
    [+ COMPONENT compName]
    [+ SPACING minSpacing | + DESIGNRULEWIDTH effectiveWidth]
    [+ MASK maskNum]
        {RECT pt pt | POLYGON pt pt pt ...} ...
;] ...
[- PLACEMENT
    [ + SOFT | + PARTIAL maxDensity]
    [+ PUSHDOWN]
    [+ COMPONENT compName]
        {RECT pt pt} ...
    ;] ...
END BLOCKAGES
```

For syntax information, see "DEF Syntax" chapter in the *LEF/DEF Language Reference*.

© 2012 All Rights Reserved.

© 2012 All Rights Reserved.