

DRCエラー・データベースを用いた LVL実行結果の分析手法

はじめに

Expertレイアウト・エディターはLVL(Layout Versus Layout)、DRC(Design Rule Check、10コマンドまで)、そして独自の専用言語LISA(Language for Interfacing Silvaco Applications)を利用することができます。全てExpert単独ライセンスで利用可能です。

ここでは、DRCエラー・データベースを用いたLVL実行結果のフラグ表示による分析手法です。

概要

LVLでは、1つのプロジェクト内で2つのセルを比較し、差分のある箇所やレイヤを検出します。検出された差分はDRCエラー・データベースに保存され、指定したセル上にフラグを表示して分析できます。

LVLの実行については、現時点のバージョンでコマンドがない為、GUIを使ったマニュアル作業で実行する必要があります。実行後に表示される差分セルから、LISAを使った自動処理が実行できます。

この時、エラーを1つのレイヤに統合するか、各レイヤのままにするかを選択できます。保持する場合は、10種類以上のレイヤが検出されていると、Expert単体のライセンスではエラー・データベースに出力できませんので、1つに統合して出力するようにします。

LVLの実行

LVLで指定できるセルは、カレント・プロジェクトかアクティブ化されている参照ライブラリに存在しているセルのみです。他のGDS IIファイルにあるセルを指定する場合は、カレント・プロジェクトにインポートするか、ELDファイルに変換してから参照ライブラリに設定する必要があります。

実行は、Expertのメニュー・バーから呼び出します。

- ・ 『検証ツール』→『レイアウトの比較』

形状マージ後は、各図形をマージ演算して計算します。階層の比較をonで、全階層フラットとなります。

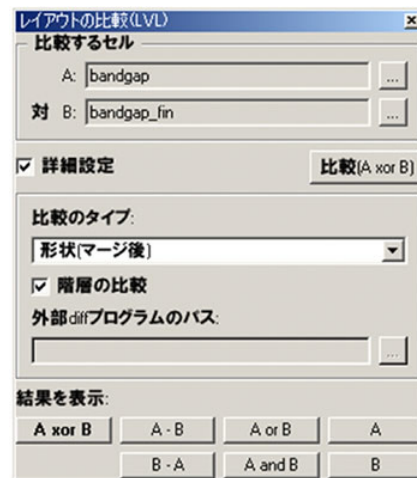


図1. LVL実行ダイアログ

差分の演算式は複数ありますが、2つのセル間の差分を見たい時には、XOR演算を利用します。

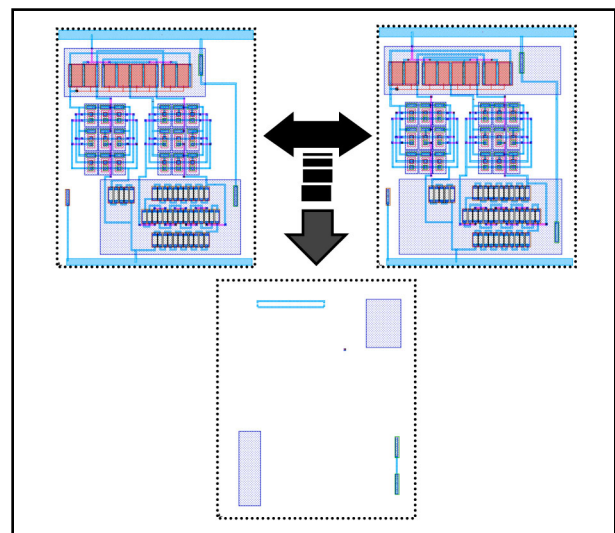


図2. 検出された差分セルの例

差分セルのエラー・データベース出力

LVLの実行後に表示される差分のセルは、テンポラリであるため、セルとして保存したい場合は、別名で保存するコマンドを実行する必要があります。

付録のLISAスクリプトはエラー・データベースに保存して、エラー・フラグを表示させるコマンドを新規登録します。

エラー・データベースに出力すると、差分箇所をDRCエラーと同様に巡回チェックする事ができ、フラグ内にはその座標で差分のあるレイヤが表示するためには、フラグ・レイヤを”_As_Is_”にします。

実行コマンド構文: lvl_after

```
XI > lvl_after [ /base = A|B /flag=value ]
```

引数は名前付きで、差分をのせるベースセルを、A/Bで指定します。他を指定するとダイアログが表示されます。フラグは差分検出されたレイヤを1つに統合する時に使用するGDS II番号を入力します。そのレイヤが存在しない場合は、キャンセルされます。

最後に

今回紹介した機能は、Expert単体のライセンスで利用する事ができます。(ただし、差分レイヤの種類が10種類以上ある場合は、レイヤ統合の必要があります。)

LVLの実行に関しては、GUIモードである為、対象が大規模になるとパフォーマンスが低下します。

エラー・データベース出力は検出された差分の個数にのみ依存します。

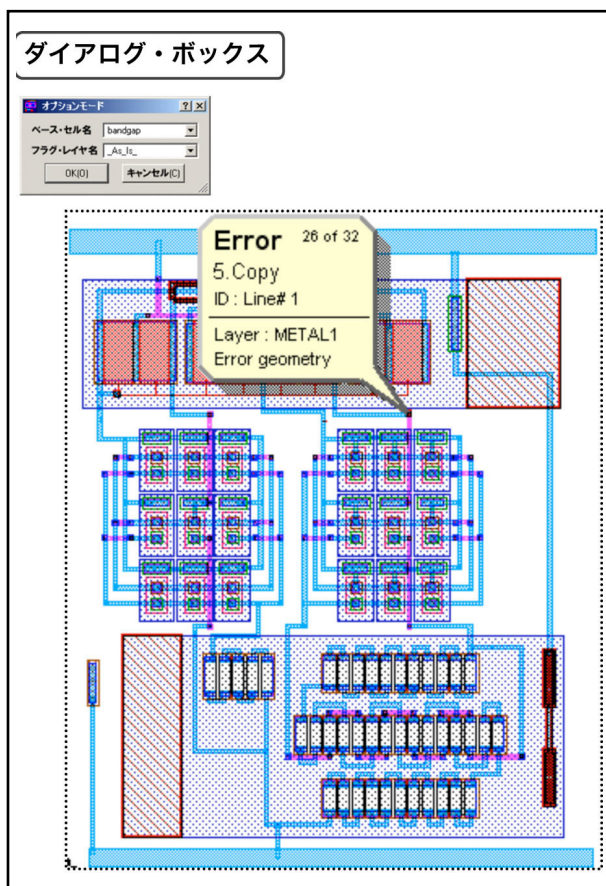


図3. エラー・フラグ表示の例

```

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!   Output Error DataBase after LVL                               !
!   *Presented by SILVACO Japan Co., Ltd.                       !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
define command/replace "lvl_after";
define action
parameter base
parameter flag
do begin

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!! make "drop-down-list" of two cells. !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
    cur_cell = get_edited_cell_name();
    cell_buf = buffer_create();
    buffer_insert_at_end(cell_buf, &cur_cell[10..-1]);
    loop begin
        if(cell_buf.at_end) then (leave loop);
        buf_move = buffer_match(cell_buf, "_xor_");
        if(buf_move EQL False) then (buffer_advance(cell_buf, 1));
        if(buf_move EQL True) then begin
            buffer_advance(cell_buf, -5);
            buffer_replace(cell_buf, cell_buf.index, (cell_buf.index+4), "¥t");
            leave loop;
        end;
    end;
    seq_cell = {(&cell_buf)[1..cell_buf.index-1], (&cell_buf)[(cell_buf.index+1)..-1]};

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!! make "drop-down-list" of all layers. !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!   (Except for the Derive Layer, Datatype=/0 & GDSnumber<0. !!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
    layers = get_layer_list();
    c=1;
    loop begin
        if(c GTR layers.size) then (leave loop);
        if((get_gds_datatype(&layers[c]) NEQ 0) OR (get_gds_number(&layers[c]) LSS 1))
            then(seq_remove_ith(layers, c)) else(c=c+1);
    end;
    layers_buf = buffer_create();
    buffer_insert_at_end(layers_buf, &layers);
    loop begin
        if(layers_buf.at_end) then (leave loop);
        buffer_scan(layers_buf, "¥n");
        buffer_replace(layers_buf, layers_buf.index, layers_buf.index, "¥t");
        buffer_advance(layers_buf, 1);
    end;

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!! Action. !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
    if(&base EQL "A") then (cell_name = seq_cell[1])
    elseif(&base EQL "B") then (cell_name = seq_cell[2])
    else (cell_name = "");
    layer_name = "_As_Is_";
    buffer_insert(layers_buf, 1, (&layer_name &"¥t"));

```

```

if((&base EQL "" ) OR (flag EQL 0)) then begin
  t1={ &cell_buf , "ベース・セル名"};
  t2={ &layers_buf , "フラグ・レイヤ名"};
  tt={t1,t2};
  tt1={" オプションモード "};
  vars1 = (form create (tt) (tt1));

  if(&vars1.type NEQ "Null") then begin      !!! Form
    cell_name = vars1[1];
    layer_name = vars1[2];
    end; !!!!!!! Form Cancel !!!!!!!
end;

cell status /rw;
select all;
tools cutbyvertex /all /selected /regionvertices=512 /wirevertices=256;
copy selection;
!!!! Version 4.4.X !!!!
all_shapes = (find objects (SEARCH_ANY_SHAPE) /selected /seq_output);
x_pos = (all_shapes[1].bbox).xpos;
y_pos = (all_shapes[1].bbox).ypos;
if(all_shapes.size GEQ 2) then begin
  l = 1;
  loop begin
    l = l + 1;
    if(l GTR all_shapes.size) then (leave loop);
    if((all_shapes[l].bbox).xpos LSS x_pos) then (x_pos = (all_shapes[l].bbox).xpos);
    if((all_shapes[l].bbox).ypos LSS y_pos) then (y_pos = (all_shapes[l].bbox).ypos);
  end;
end;
!!!!!!!!!!!!!!!!!!!!!!!!!!!!
cell close;

if((cell_exists(cell_name) EQL True) AND ((layer_exists(layer_name) EQL True) OR (&layer_name EQL "_As_Is_"))) then begin      !!! Execution
  if(cell_exists("lvl_result") EQL True) then (cell delete "lvl_result");
  cell new "lvl_result";
  cell open "lvl_result";
  instance 0 0 /cell = (cell_name);
  if(&layer_name EQL "_As_Is_") then begin
    paste selection (x_pos) (y_pos);          !!! Version 4.4.X
! paste selection;                          !!! Version 4.5.X
    all_layers = get_layer_list();
    m = 0;
    n = 0;
    check = "";
    loop begin
      m = m + 1;
      if(m GTR all_layers.size) then (leave loop);
      layer_count = get_layer_object_count( (get_edited_cell_name()), (all_
layers[m]));

      if(layer_count GTR 0 ) then begin
        n = n + 1;
        if(n EQL 10) then begin
          ret = message_box("差分のあるレイヤが10種類以上あります。 \n"
&"GuardianDRCのライセンスが必要となります。 \n\n実行しますか?", {});
          if(ret EQL false) then (leave loop);
        end;
      end;
    end;
  end;
end;

```

```
        check = &check &"Copy: Layer=" &all_layers[m] &";";
    end;
    if(m EQL all_layers.size) then (drc command (check) /cell); ! /selection
end;
end
else begin
    paste layer (layer_name);
    merge selection;
    tools cutbyvertex /all /selected /regionvertices=512 /wirevertices=256;
    zoom /all;
    check = "Copy: Layer=" &layer_name &";";
    drc command (check) /cell; ! /selection
end;

end;
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

end; !!!!!!! define end !!!!!!!
define argument base /named /coerce_to=(String) /default="" /is_default;
define argument flag /named /coerce_to=(Integer) /default=0 /is_default;
complete command;
```