MAVEN SILICON

Centre of Excellence in VLSI

Physical Verification Using Calibre Tool

Lab Manual

www.maven-silicon.com

Maven Silicon Confidential

All the presentations, books, documents [hard copies and soft copies] labs, and projects [source code] that you are using and developing as part of the training course are the proprietary work of Maven Silicon and it is fully protected under copyright and trade secret laws. You may not view, use, disclose, copy, or distribute the materials or any information except pursuant to a valid written license from Maven Silicon

LIST OF EXPERIMENTS

Lab 01 – Calibre Basic Concepts

Exercise 1: Invoke Calibre DESIGNrev

In this exercise you will invoke DESIGNrev from the command line, load a GDSII design, and load the layer palette.

1. From a terminal, change your directory to Lab-1.

   /home/User/PD_LABS/4.Calibre/Lab-1

2. Launch DESIGNrev.

   $ calibredrv

   This command opens the initial DESIGNrev window. Now you will load the GSDII file.

3. Choose Menu: File > Open Layout Files.
4. Select adcctrl.gds, by double-clicking.

   These steps open the layout design you will be using for the first parts of this lab.

   Next you load the layer properties file. This file names the layers (rather than just using numbers) and gives the layers their “expected” colours.

5. Load the layer properties by clicking Menu: Layer > Load Layer Properties.

   This step opens the Load Layer Properties dialog box.

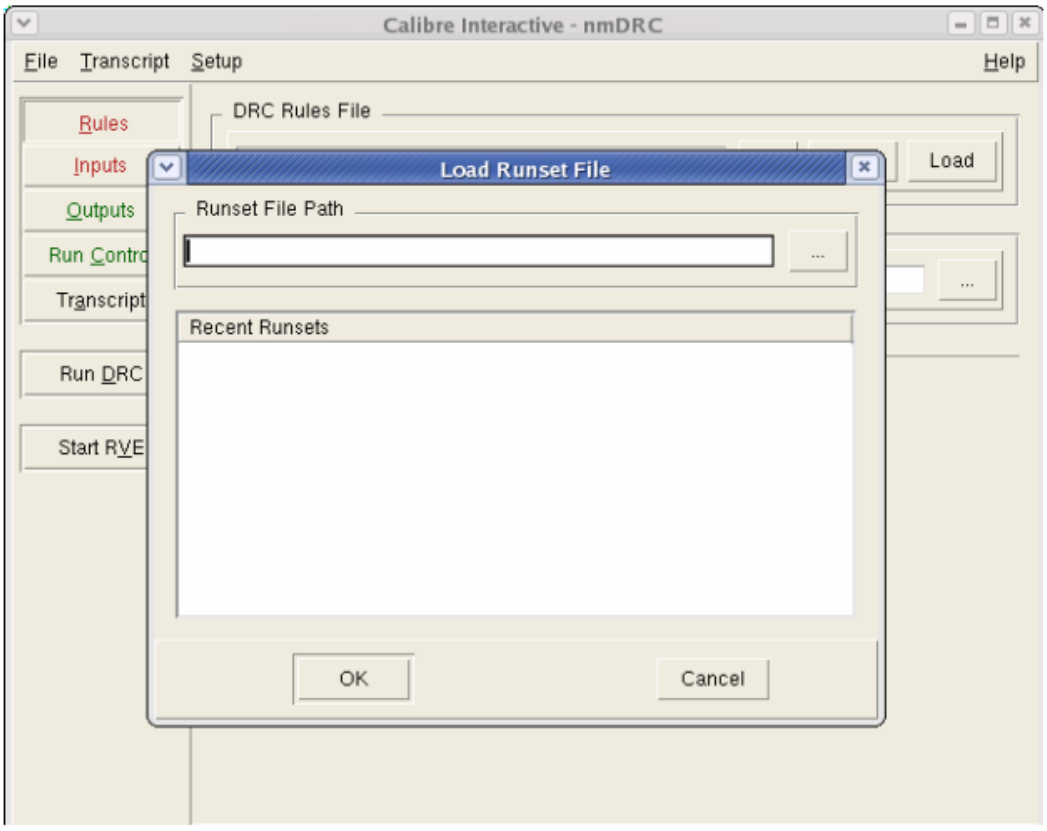
6. Select the process.layerprops file.
7. Click Open to load the layer properties.

   Observe the layout in the DESIGNrev window and try to change the background from black to white by clicking the white colour sample in the layers pane using MMB (middle mouse button)

Exercise 2: Launch Calibre Interactive - nmDRC

- Initially, Calibre In this exercise you will open Calibre Interactive - nmDRC and load a runset containing all the information required for a DRC run.
- You will then review all the various menus and options available from Calibre Interactive - nmDRC.
- Interactive - nmDRC asks you to choose a runset. A runset is a file containing saved settings of and entries in Calibre Interactive so you can have consistency between Calibre runs.

1. Invoke Calibre Interactive - nmDRC.
   Choose **Menu: Verification > Run nmDRC**
   The Calibre Interactive - nmDRC window and Load Runset File dialog box should now be displayed as shown below



2. Click the Browse button (...) in the Load Runset File dialog box and select lab1_runset.txt (if available). Otherwise cancel the runset window.
3. Before running DRC, check the all the information loaded to perform a DRC run.
4. Click **Outputs**

This window displays the information you want Calibre to output from this run and the proper format for RVE. Notice the “Show results in RVE” is selected causing RVE to start with the DRC results shown automatically.

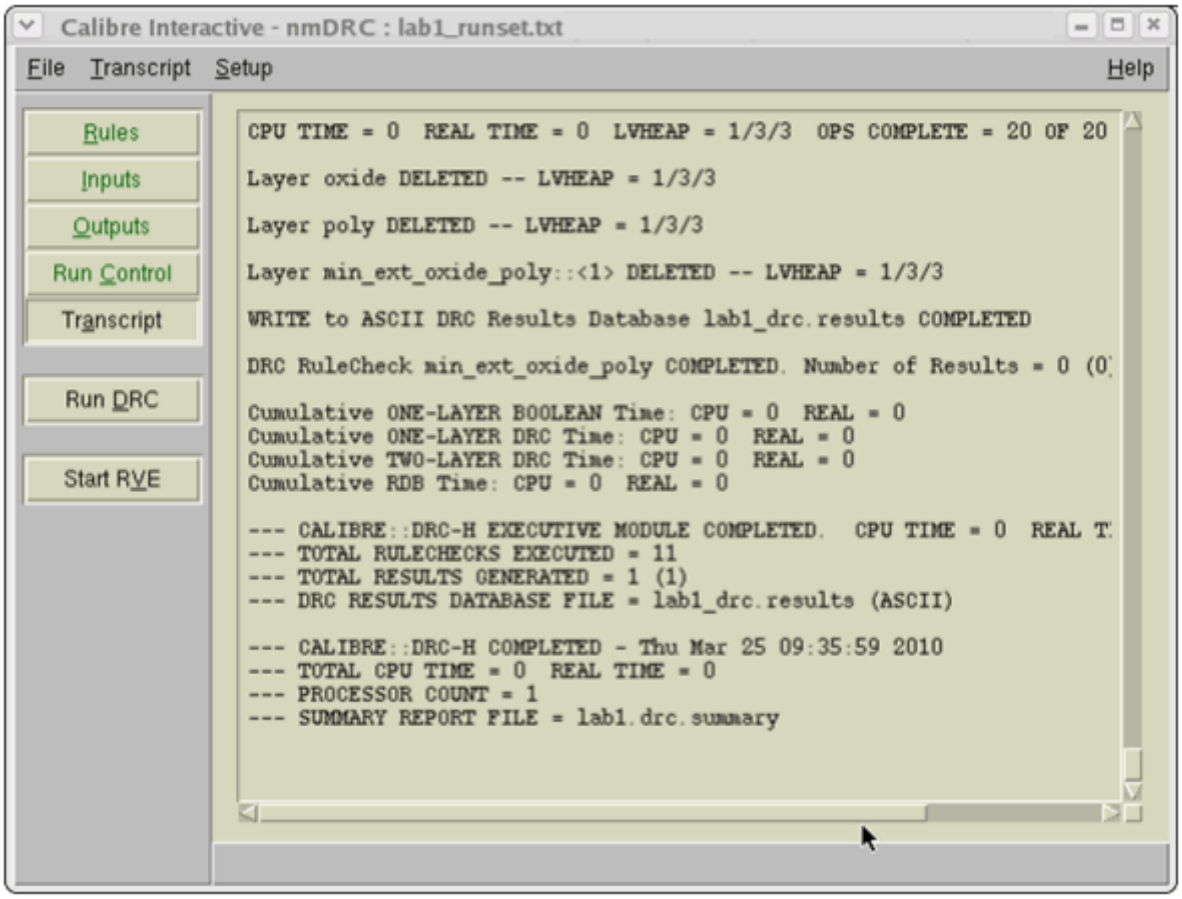
The DRC Summary Report will be written to the file lab1.drc.summary at the end the DRC run.

5. Click **Transcript**

   This window will display the transcript while Calibre nmDRC runs

6. Click **Run DRC**

   At the end of the DRC run, two new windows open, The DRC Summary Report window and Calibre RVE. In Calibre Interactive, the Transcript window will look similar to below:



7. Scroll through the transcript to take note of the type of information available.
8. Make the DRC Summary Report window active by selecting it.

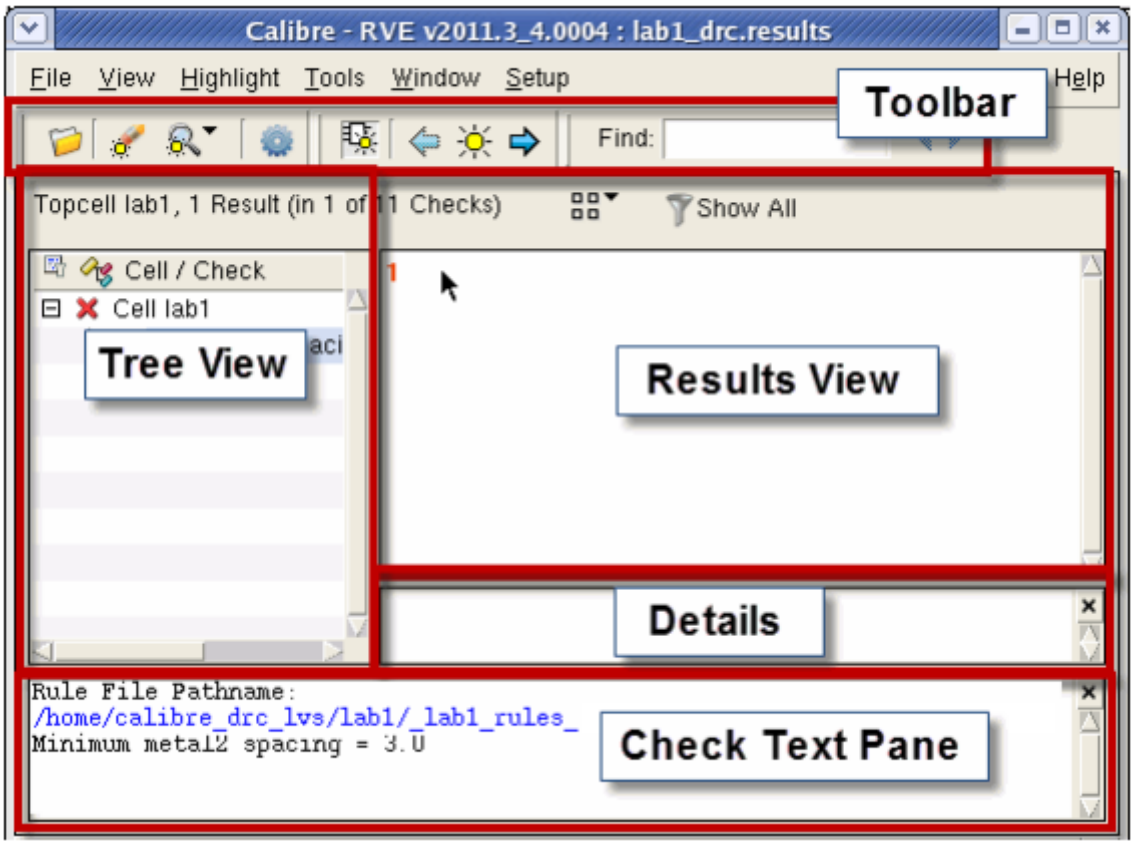
   This window displays the results of the DRC check in text format. Observe the report to see the type of information available.

9. After viewing the report, close the report window. (Choose Menu: File > Close).

Exercise 3: View a Discrepancy with Calibre RVE

In the exercise, you will learn how to use RVE to view discrepancies and highlight them in the layout.

1. Make the RVE window active.
2. In RVE, choose **Menu: View > Tree Options** and unselect the **Show Empty Checks** option if it is selected.
3. In RVE, choose **Menu: View > Tree Options > Group By > Cell/Check**.
   The RVE window will now looks as shown below



It also has a toolbar for the commands used most frequently. The icons from left to right are:

	Open database		Highlight in cell context
	Clear all highlights		Highlight previous result
	Choose layout window viewing options		Highlight current result
	Open RVE setup options tab		Highlight next result

- The Tree View contains a “tree” structure of the DRC results. This view provides a summary of the rule checks and cells involved in the DRC run.
- The Results Window, located to the right of the Tree View, displays the list of results associated with the currently-selected check.
- Below the Results View is the Details area. This area provides the layout coordinates for the discrepancies (useful when you manually track the location in the layout).
- The Check Text Pane is located at the bottom of the RVE window. This area displays the information provided from the rule file about the current discrepancy.

4. Observe and find out the name of the rule with the discrepancy under the Tree view.
5. Observe and analyse the information obtain while clicking on the result number.
6. Close the RVE window (**Menu: File > Exit**)

Exercise 4: Experiment With DESIGNrev

- In this exercise, you will learn how to perform some very simple operations in Calibre DESIGNrev.
- There are multiple ways to perform any task in DESIGNrev. In this lab, most tasks will be performed using the Toolbar or mouse button (RMB or LMB) commands whenever possible.
- You will cover the following topics in this lab:
  - Displaying the Contents of a Cell
  - Displaying Lower/Higher levels of hierarchy detail
  - Zooming Out of an Area
  - Centering the Display
  - Selecting Polygons
  - Unselecting Polygon(s)
  - Selecting more than one Polygon
  - Unselecting just one Polygon from a group
  - Moving Polygons
  - Making a Box
  - Making a Polygon
  - Adding a New Vertex
  - Change a Shape By Moving a Segment (Edge) of a Polygon
  - Notching an Existing Shape
  - Changing the Grid
  - Changing the Ruler
  - Finding Instances using RVE

1. Launch DESIGNrev.
2. To fit the entire layout into the window, click the **Z All**  toolbar icon. This step also puts you at a good starting point.

Displaying the content of a cell

1. Notice the Cells pane on the left side of the tool. This pane controls which cell in the design is currently displayed. The two tabs let you choose between seeing the cells in the design listed alphabetically, or displayed as hierarchical tree structure.
2. Notice the text entry field just below the tabs. This field lets you navigate to a specific cell in a large design by entering a partial cell name. Press Tab to complete a partial cell name or Enter (or Return) to select the cell. If the string you type in is ambiguous, the tool responds with the first cell it encounters that matches the string
3. If cells are not yet displayed as a hierarchical tree structure, click the Tree tab.
4. Expand the tree if it is not already expanded, by click the "+" by lab1 in the Cells tree to expand the hierarchy list of all the cells in the lab1 cell.
5. Select a cell in the hierarchy list to view and edit the internal layout of the cell.

Displaying Lower/Higher Levels of Hierarchy Detail

1. Choose **Menu: View > Change Hierarchy Depth > Increment To Depth**
   This displays the layout structures one level lower in the hierarchy. You cannot edit the contents of cells at this level, but you can see the underlying structure and avoid creating shorts, and so on. (Similarly try with **Decrement to Depth**)

Zooming into an Area

1. Press and hold down the **right mouse button** (RMB).
2. Draw a rectangle from **upper-left to lower-right** around the area you want to display.
3. Release the RMB. (When you release the mouse button, the surrounding area zooms in to fill the display window.)

Zooming Out of an Area

1. Press and hold down the RMB.
2. Draw a rectangle from the **lower-right to the upper-left**, centring on the area that you would like centred in the new display.
3. Release the RMB.

Centering the Display

1. Place the cursor over the area you want to be the new center of the display.
2. Click the middle mouse button (MMB), (The layout display recenters itself around the new center.)

Selecting Polygons

1. Choose the Select icon from the Toolbar Menu. (Make sure the Select icon is selected.) 

.
2. Unselect all selection types except Polygon.

3. Select any single item in the layout by clicking on it with the LMB. The selected polygon will highlight.
4. To select more than one polygon, choose the **select** icon from the Toolbar Menu. (Make sure the Select icon is selected.)
5. Hold down the Shift key and click the LMB on the desired polygon. The selected polygon will be highlighted.
6. Repeat the above step, until you have selected all the desired polygons.

Unselecting Polygons

1. Choose the Select icon from the Toolbar Menu. (Make sure the Select icon is selected.)
2. Click the LMB in an empty area of the layout.
3. The selected polygon will lose its highlight.
4. To unselect just one polygon from a group, chose **select** icon from the toolbar menu.

5. Hold the Ctrl Key and click the LMB on the undesired polygon, the unselected polygon will lose its highlight.

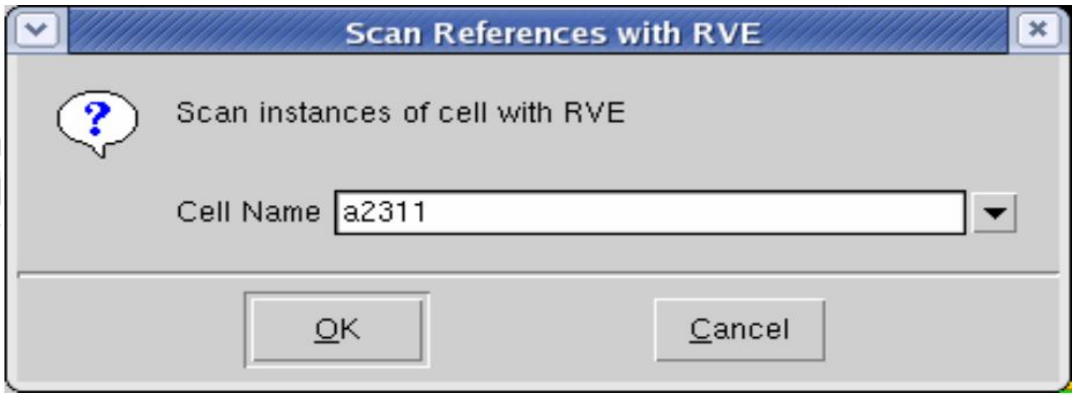
Moving Polygons

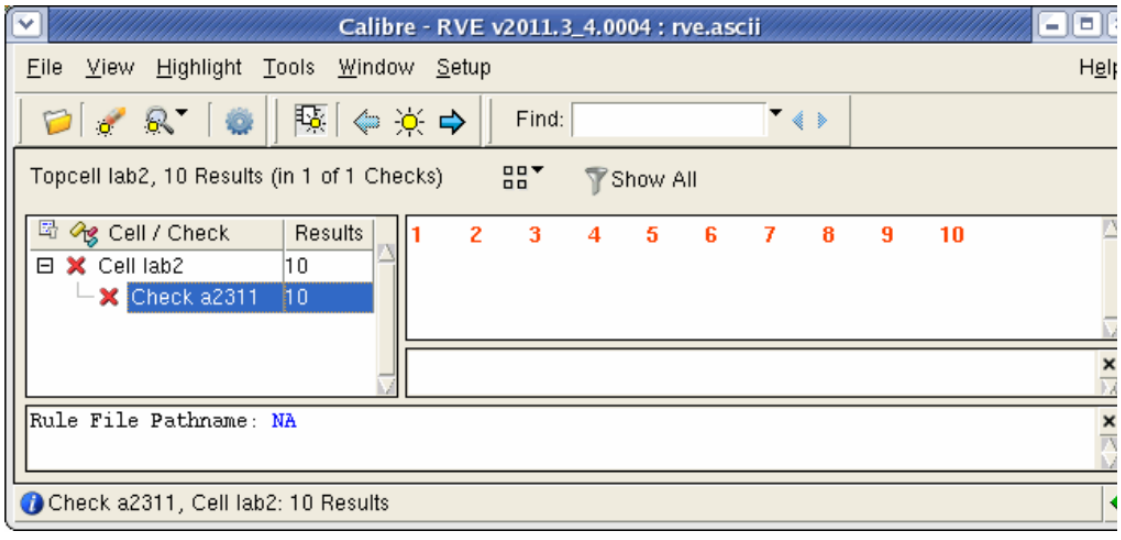
1. Select the polygon(s).
2. Choose the Move icon from the Toolbar Menu. 

3. Press and hold down the LMB.
4. Drag the polygon(s) to their new location.
5. Release the LMB. Notice that your polygon(s) are still selected after the move operation.
6. Undo the move by selecting **Menu: Edit > Undo: Move**.

Finding an Instance with RVE

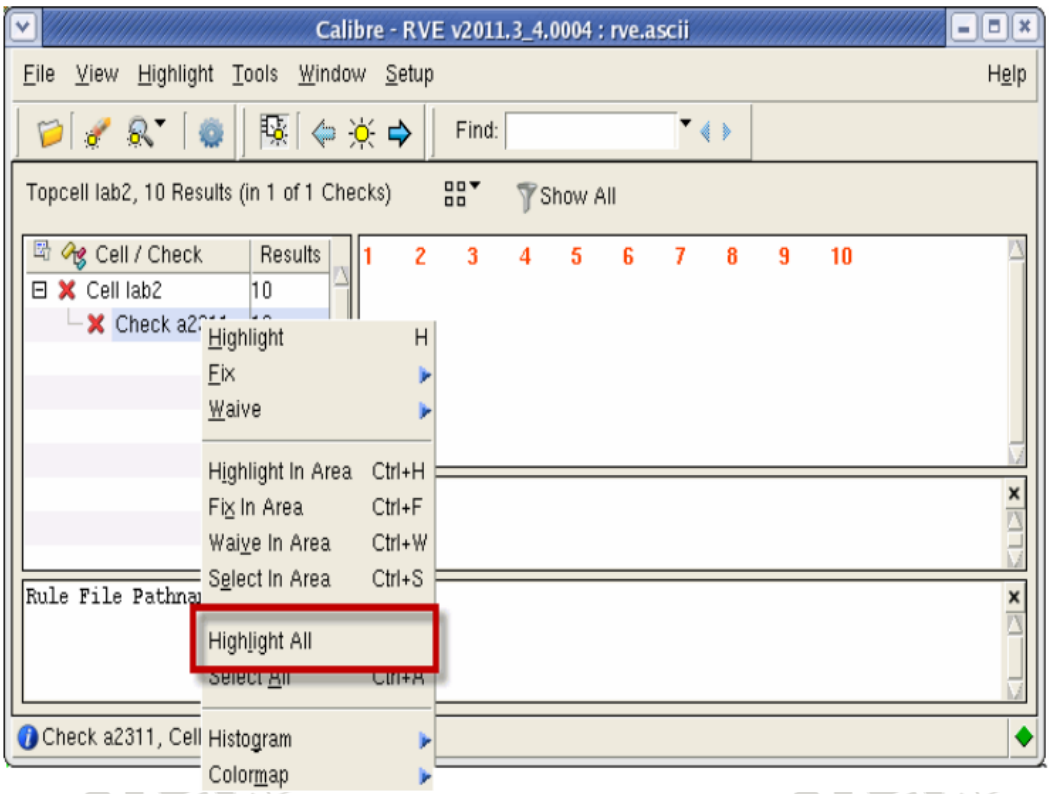
1. Display the entire layout (Z all).
2. Choose **Menu: Tools > Scan with RVE > Scan References with RVE.**
3. This act opens the Scan References with RVE dialog box.
4. Type a2311 in the text box, or click the down arrow to the right of the Cell Name field, and select a2311 from the dropdown list





You will see that there are 10 instances of the a2311 cell

5. Click the RMB over cell a2311 and choose Popup: Highlight > Highlight All to highlight all the instances of a2311



All the instances are now highlighted in DESIGNrev.

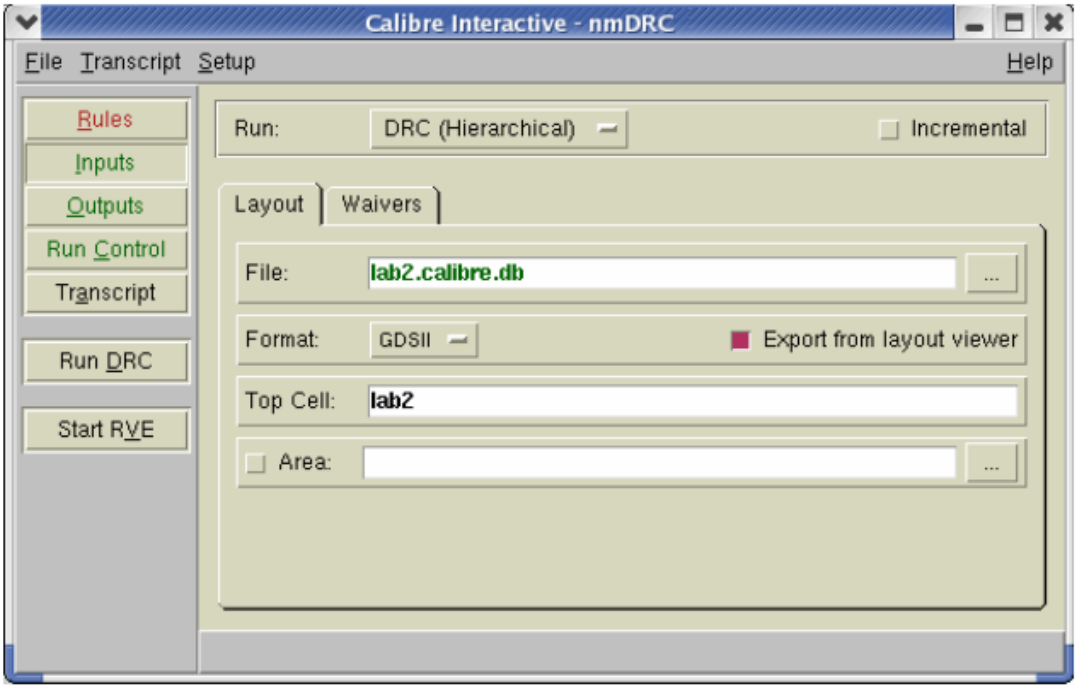
Lab 02 - Calibre nmDRC

Exercise 1: Set Up and Run Calibre nmDRC

1. Open the terminal and change your directory to the location of the lab 2 as follows:
   cd $HOME/calibre_drc_lvs/lab
2. Launch the layout tool and load lab2.gds.
   a. Launch DESIGNrev. calibredrv -m lab2.gds
   b. Launch Calibre Interactive - nmDRC for cell lab2.
   (Menu: Verification > Run nmDRC)

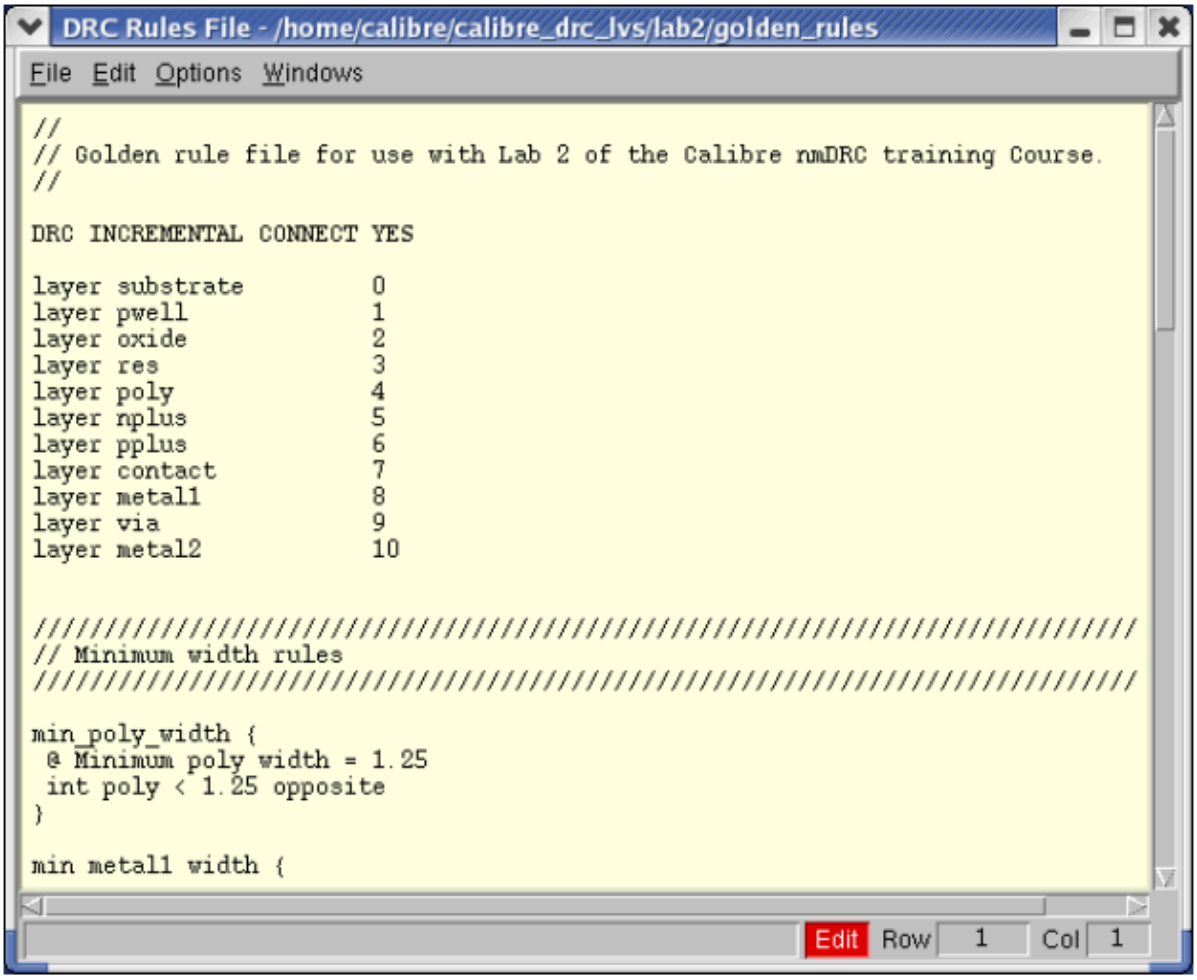
   These actions launch Calibre Interactive - nmDRC, displaying the Load Runset dialog box.

   In the previous lab, you used a runset to load all the required information. In this lab you will create your own runset by entering all the information by hand first.
3. Choose Cancel in the Load Runset File dialog box.
   This step makes the Calibre Interactive - nmDRC dialog box active with the **Inputs Menu** button selected.

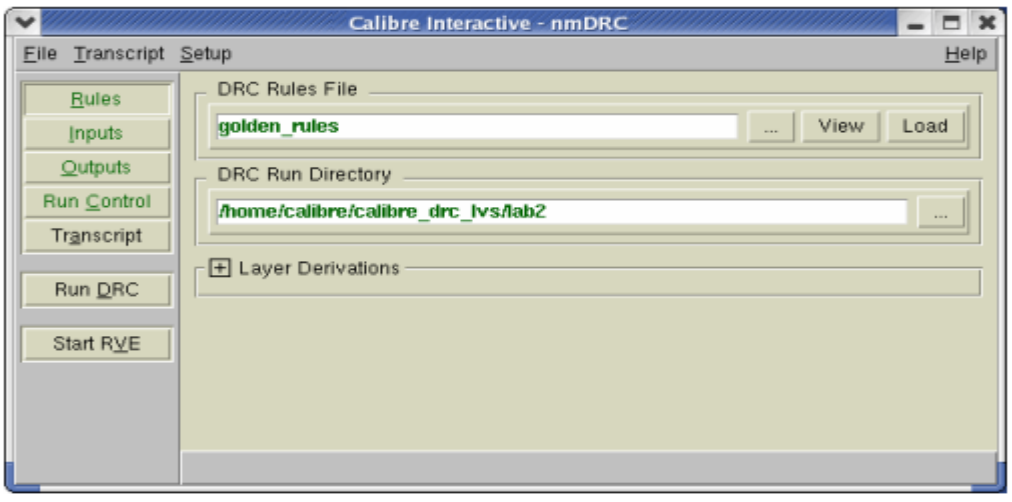


4. Unselect Export from layout viewer.
5. In the Run field, select DRC (flat/Hierarchical, based on the design).
6. Notice that the layout file name and the inputs button both are green, where as "lab2.calibre.db" is not the name of your layout file.
7. Make sure the file format is GDSII
8. Click the rules menu button, this displays the rules information needed for a DRC run
9. Browse and select the golden rule file from lab2 directory .

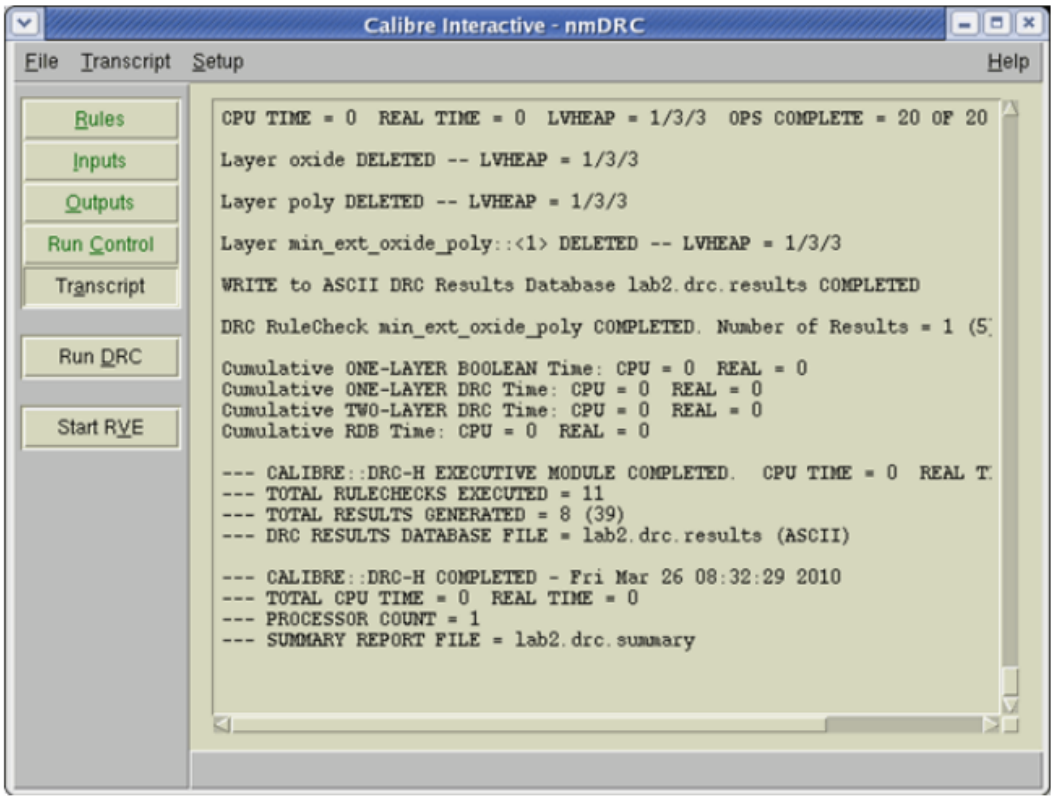
10. Click **View** to have brief look at the rule file



11. Notice that this is a very simple rule file and only contains layer definitions and a handful of DRC rules. Also notice the red Edit button in the lower right-hand corner of the window. It indicates that the file is not available for edit. (It is a safety feature so you do not accidentally make edits to a “golden” rule file.
12. Note the value in the Calibre nmDRC Run Directory. This will place all the resulting files either in the current directory



13. Click the Outputs menu button. This displays the dialog box where you will set the names of the output files.
14. Accept the defaults for all the fields in the pane. Here, you are creating files lab2.drc.results (the DRC Results Database) and lab2.drc.summary (the DRC Summary Report).
15. Click the Transcript menu button - This displays the Transcript during the DRC run. From here you can quickly note any problems that may occur during the run.
16. Click Run DRC to start the run - When the run completes: RVE launches, the DRC Summary Report displays, and the Transcript Window should look similar to below:



Exercise 2: Check the Results

1. In this exercise, you will review the error messages found in the transcript, summary report, and RVE. You will also highlight the errors in the layout.
2. Look at the transcript window and answer the following questions:

   How many rules were executed? _____

   How many discrepancies (results) were found? _____

   What does the number in parentheses () mean? _____

3. Scan the Summary Report and answer the following questions:

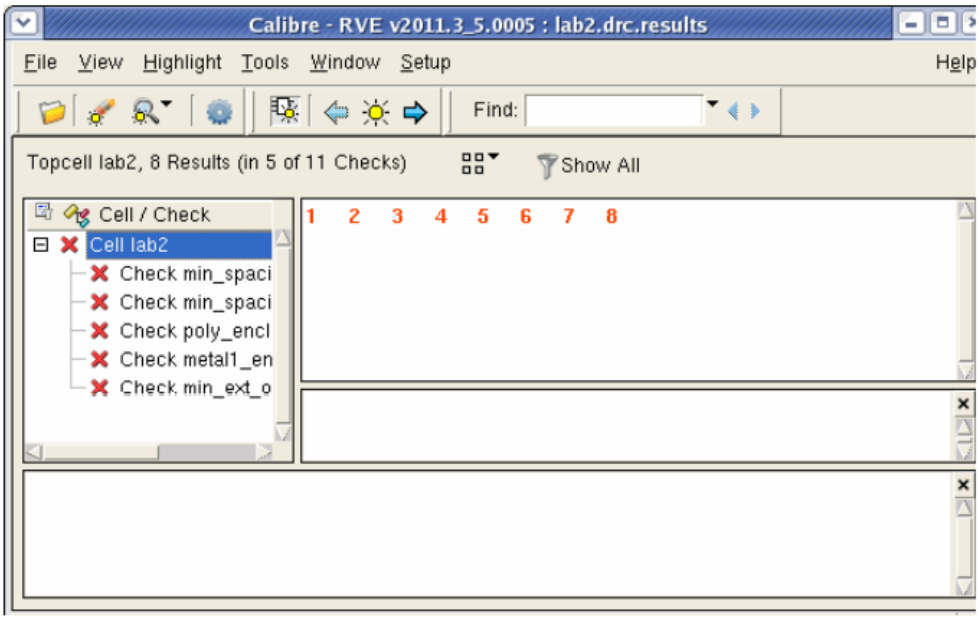
   Which rules have discrepancies?

   • _____

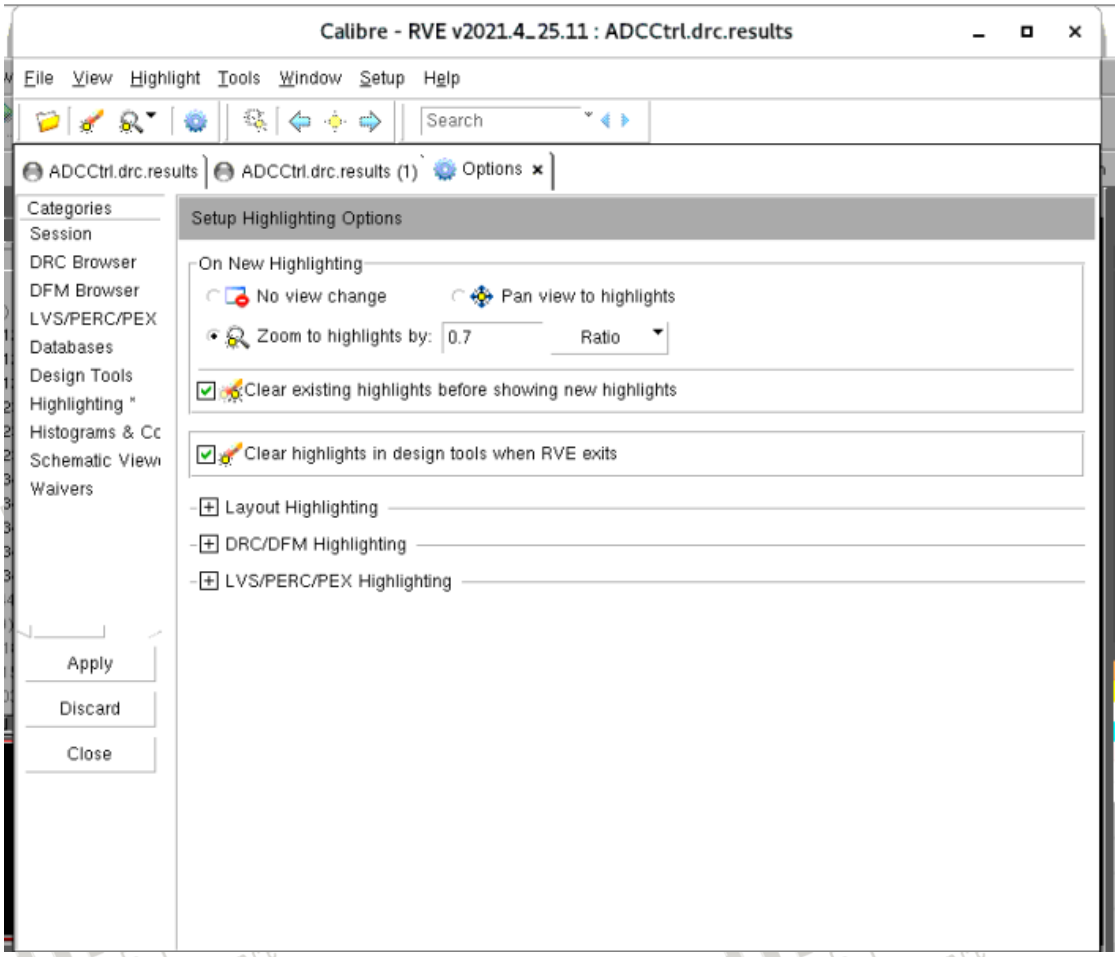
   • _____

   Which cell(s) have discrepancies?_____

4. Close the DRC Summary Report Window and make the RVE window active



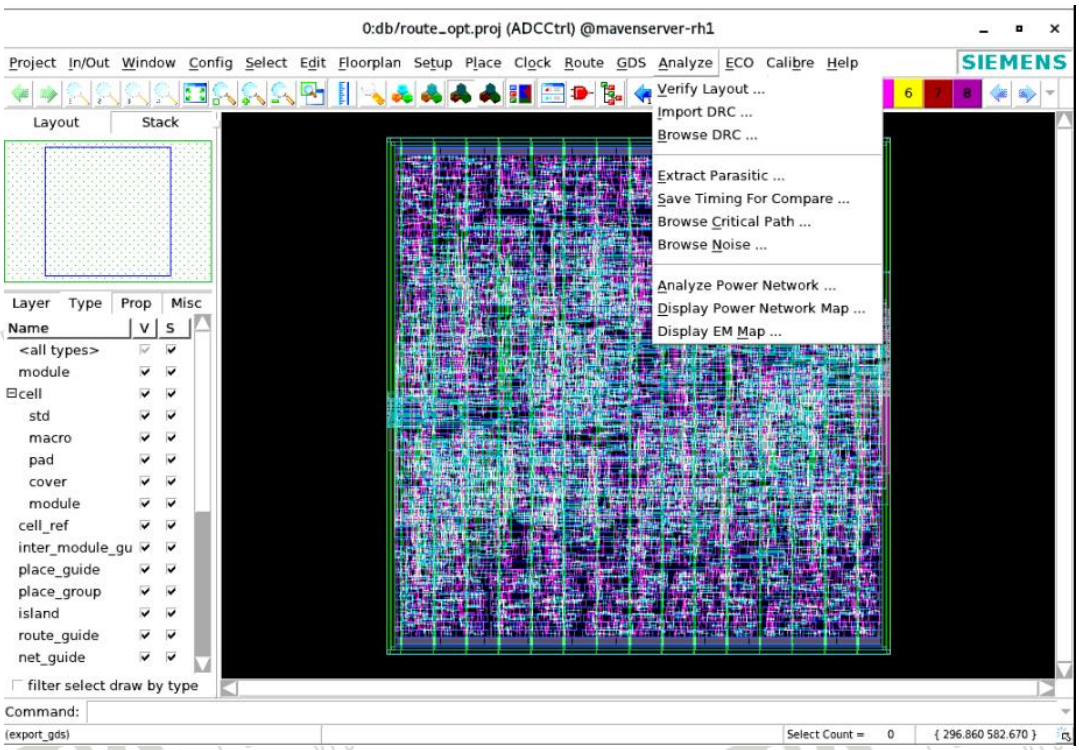
5. Display the details for each of the results by clicking on the "+" in front of the rule checks until fully expanded.
6. Set up highlighting:
   a. Choose **Menu: Setup > Options** - this opens the Options tab in Calibre RVE.
   b. Make sure the **Highlighting** category is displayed.
   c. Enable Zoom to highlights by.0.4 in the text box. (This zoom works well for the errors in the current layout.)
   d. Enable the **Clear existing highlights before showing new highlights**.
   e. Enable the **Clear highlights in design tools when RVE exits**.
   f. Click **Apply**.



7. Click the lab2.drc.results tab to return to the result display
8. Click on any of the RuleCheck.and understand the rule specifications.

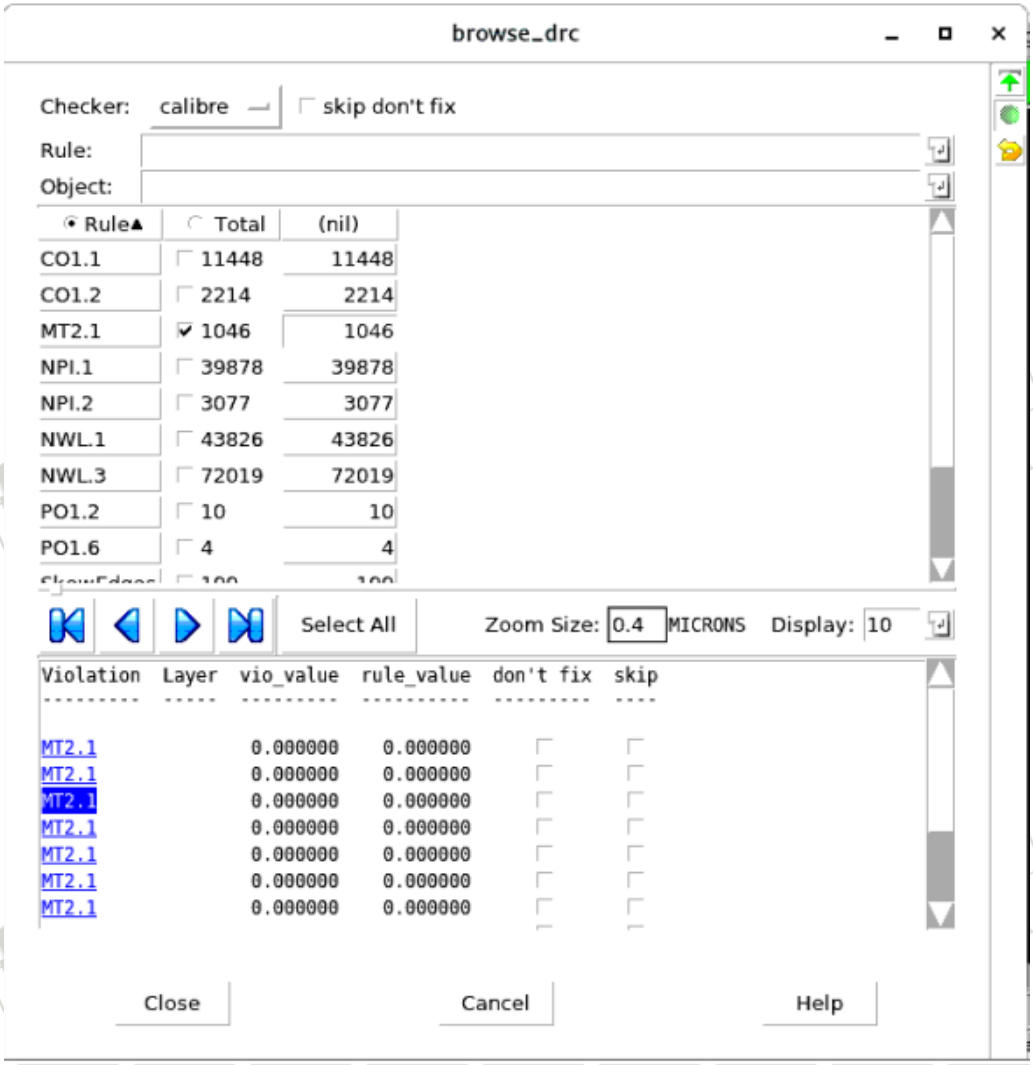
Exercise 3: Repair the Violations using Aprisa Tool

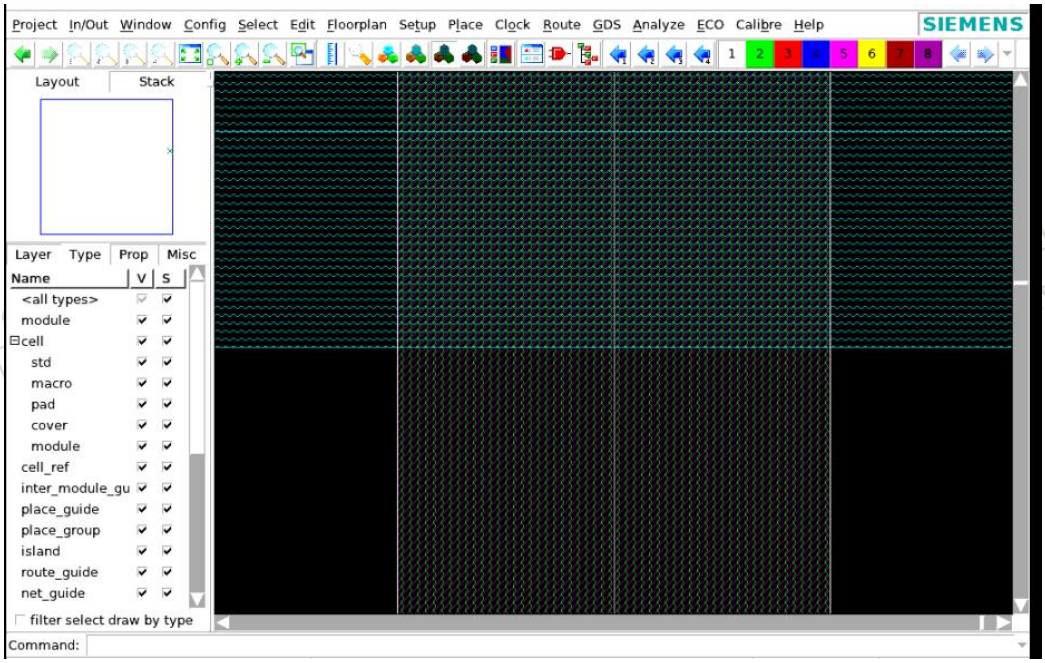
1. Invoke the Aprisa tool and Load the project, which is under verification.
2. Open the layout and click on **Analyze** tab.



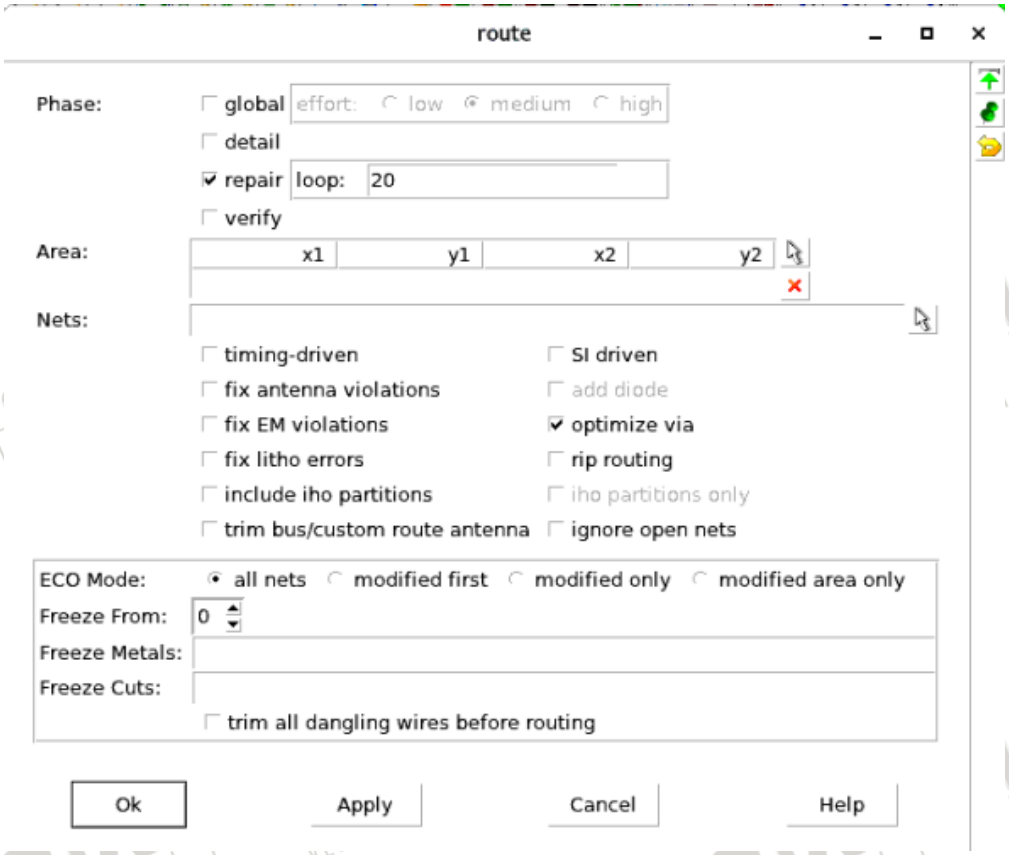
3. Select the **Import DRC** option.
4. Select the file Lab2.drc.results and **apply**

5. Click on Analyze and select **Browse DRC** option to view the uploaded errors
6. Select the calibre under the checker
7. Click on a particular violation to view in layout window





8. To repair these violation Click on Route menu and select route option



9. Select and apply the suitable options to repairs the violations
10. Export the modified GDS file and run the DRC check using calibre tool to see the minimised errors.

Lab 03 – Calibre PERC

Exercise 1: PERC control file based on job specifications

In this lab, you will create and test a basic PERC control file based on job specifications.

Create a Basic Control File

In this exercise, you will create a control file for a basic PERC AERC job. You will be given complete job specifications and you will write all of the needed PERC SVRF control statements "from scratch."

In this case, you will use an existing PERC rule file that checks for any MOS gates tied directly to supply ports at the top level. Your control file will include the PERC rule file and will specify all needed SVRF control statements.

1. While in your home directory, type:

```
[user@maven-silicon] $ cd /HOME/user/Calibre_PERC/PERC_LABS/LAB03/Ex_1
```

Display the contents of the PERC_introduction directory: ls

Q. List the directory contents and, based on the names, try to predict the function of each item found:

……

2. Open the PERC rule file, perc_a.rules, in a text editor and review the contents of the file.

- What is the name of the TVF FUNCTION?
  ……..
- What is the name of the PERC initialization procedure?
  ……..
- How many rules are there?
  ……..
- What is the name of the PERC rule?
  ……
- Does the rule file depend upon the control file to define power and ground?
  ……
- Based on your answers, write the required PERC LOAD statement:
  ……

Close the perc_a.rules file.

3. This job will analyze an existing source netlist name *lab3_1.spi*. Review the netlist file
   - What is the name of the top-level subcircuit?
     ……
   - What is the type of netlist file?
     ……

- What SVRF statements are needed to specify the netlist name, type, and top subcircuit?

  ……..

- What SVRF statement specifies the use of a source netlist for analysis?

  ……

Close Spice Netlist file.

4. Your job should create a PERC report file named *perc.rep*.

   - Write the SVRF statement that specifies the PERC report file name:

     ……

5. You will need to specify the name of the SVDB directory which will be used to store the PERC results.

   - Write the SVRF statement that specifies the name of the SVDB directory:

     ……

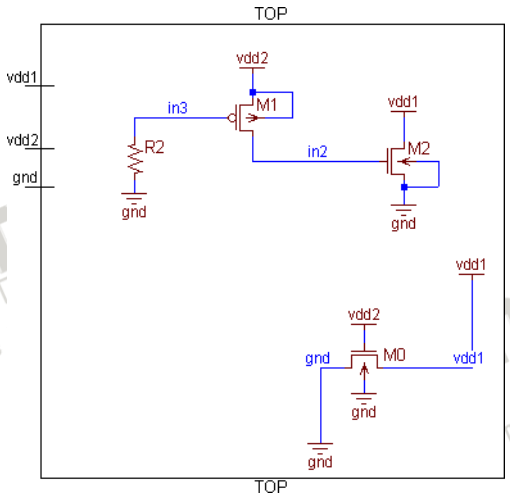
6. Open the PERC job file job_a.rules for editing and insert all of the required job control SVRF statements, using your answers to the previous questions as a guide (don't forget an INCLUDE statement for perc_a.rules).

7. Re-examine the *perc_a.rules* PERC rule file.

   - What circuit problem is the analysis looking for?

     ……

8. The schematic represents the *perc_a.spi* netlist is shown here:



   - Based on your answer above, which devices in the schematic will be reported in violation of the rule?

     ……

9. Run the PERC job by entering the following command:

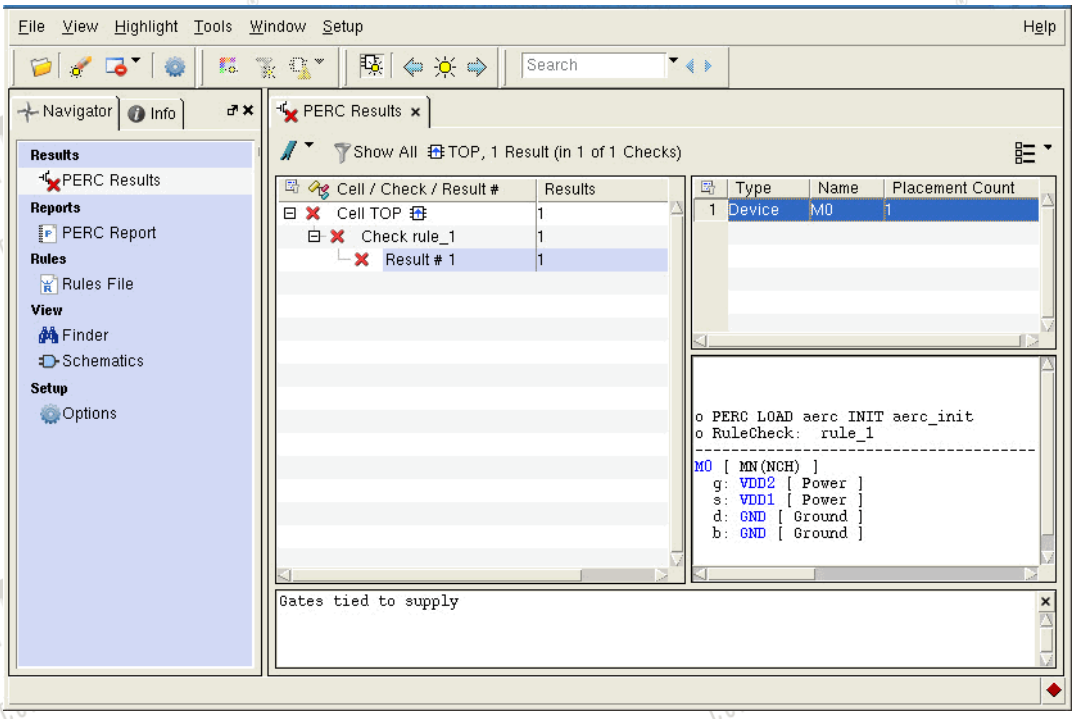
```
[user@maven-silicon] $ calibre –perc –hier job_a.rules >log
```

10. Review the log file for any warning or error messages. If there are any errors, edit your job control file and run the job again. Ask the instructor for help if you cannot determine the cause of the error.
11. Open the results in RVE by typing:

```
[user@maven-silicon] $ calibre -rve -perc svdb
```

Select **View → Tree Options→ Group By → Cell / Check / Result** #.

The RVE window will look like this:



12. Click the LMB on the device name M0. This will highlight M0 in the schematic and the netlist.
    - Did you get the result you expected? Explain
    - If you did not get the expected results, review your control file
13. Close RVE. - This concludes the lab.

Exercise 2: PERC control file based on SVRF varibales

In this lab, you will create and test a basic PERC control file based on job that uses SVRF variables.

Create a Basic Control File

In this exercise, you will create a PERC control file that defines SVRF variables which are used by the PERC rule file. In this case, you have been told that the PERC rule file contains references to specific SVRF variables that you must define.

1. While in your home directory, type:
   [user@maven-silicon] $ cd /HOME/user/Calibre_PERC/PERC_LABS/LAB03/Ex_2
   Display the contents of the PERC_introduction directory: ls

   Q. List the directory contents and, based on the names, try to predict the function of each item found:

   ……

2. Open the PERC rule file, perc_a.rules, in a text editor and review the contents of the file.

   - What is the name of the TVF FUNCTION?
     ……..
   - What is the name of the PERC initialization procedure?
     ……..
   - How many rules are there?
     ……..
   - What is the name of the PERC rule?
     …….
   - Based on your answers, write the required PERC LOAD statement:
     …….....

   Close the perc_a.rules file.

3. This job will analyze an existing source netlist name *lab3_2.spi*. Review the netlist file
   - What is the name of the top-level subcircuit?
     ……
   - What is the type of netlist file?
     ……
   - What SVRF statements are needed to specify the netlist name, type, and top subcircuit?
     ……..
   - What SVRF statement specifies the use of a source netlist for analysis?
     …….

     Close Spice Netlist file.

4. Your job should create a PERC report file named *perc.rep*.
   - Write the SVRF statement that specifies the PERC report file name:

     ……

5. Re-examine the perc_b.rules PERC rule file
   - What circuit problem is the analysis looking for?

     ……

   - What SVRF variables are used by this file?

     ……

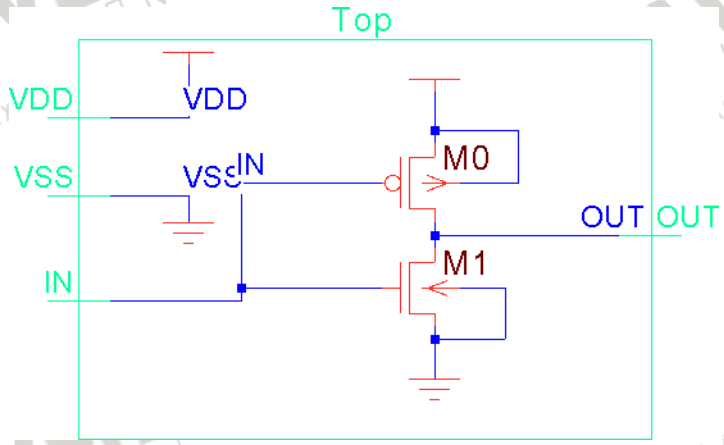
   - What SVRF statements are required to define these variables?

     ……..…..

6. You will need to specify the name of the SVDB directory which will be used to store the PERC results:
   - Write the SVRF statement that specifies the name of the SVDB directory:

     ……

7. Open the PERC job file job_b.rules for editing and insert all of the required job control SVRF statements, using your answers to the previous questions as a guide (don't forget an INCLUDE statement for perc_b.rules).
8. The schematic represents the perc_b.spi netlist is shown here:



   - Which devices in the schematic will be reported in violation of the rule?

     ……

9. Run the PERC job by entering the following command:

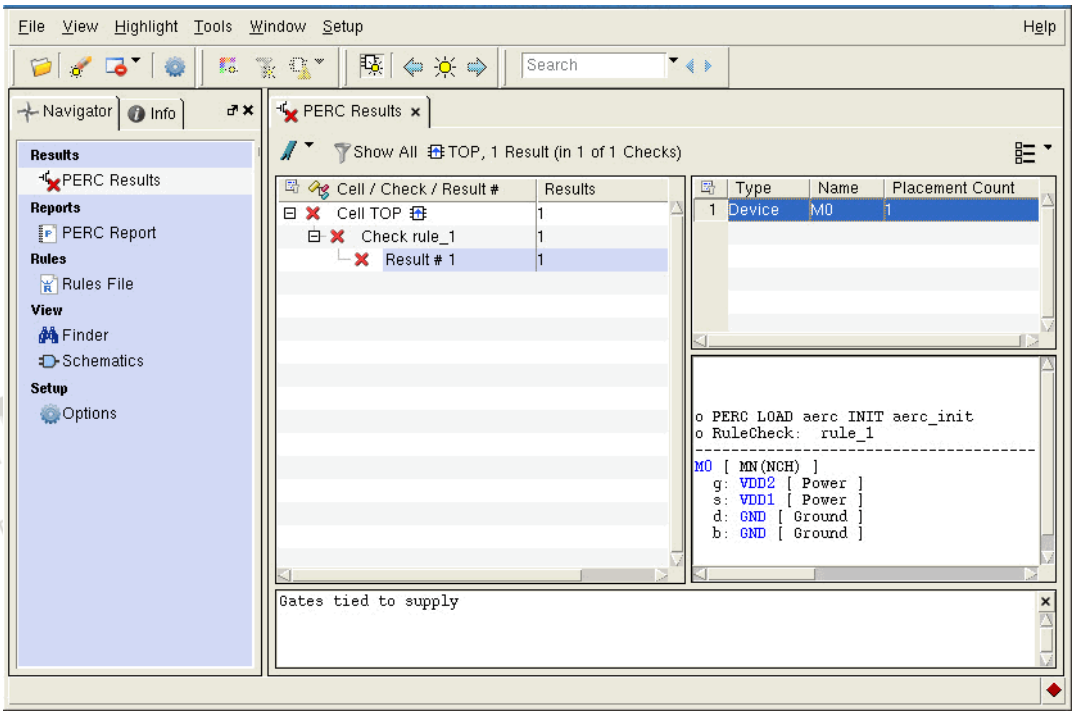
```
[user@maven-silicon] $ calibre –perc –hier job_b.rules >log
```

10. Review the log file for any warning or error messages. If there are any errors, edit your job control file and run the job again. Ask the instructor for help if you cannot determine the cause of the error.
11. Open the results in RVE by typing:

```
[user@maven-silicon] $ calibre -rve -perc svdb
```

Select **View → Tree Options→ Group By → Cell / Check / Result** #.

The RVE window will look like this:



12. Open the PERC report in any text editor.
    - How many results were generated?

      …….
13. If you did not get the expected results, review your control file
14. Close RVE. - This concludes the lab.

Exercise 3: PERC control file based on device properties

In this lab, you will create and test a basic PERC control file based on device properties

Create a Basic Control File

In this exercise, you will create a PERC control file that enables the use of a device property in the PERC rule file. In this case, you have been told that the PERC rule requires that MOS gate pins which are connected to pads have at least a 200-ohm resistor connected between the pad and the gate pin.

1. While in your home directory, type:

   ```
   [user@maven-silicon] $ cd /HOME/user/Calibre_PERC/PERC_LABS/LAB03/Ex_3
   ```

   Display the contents of the PERC_introduction directory: ls

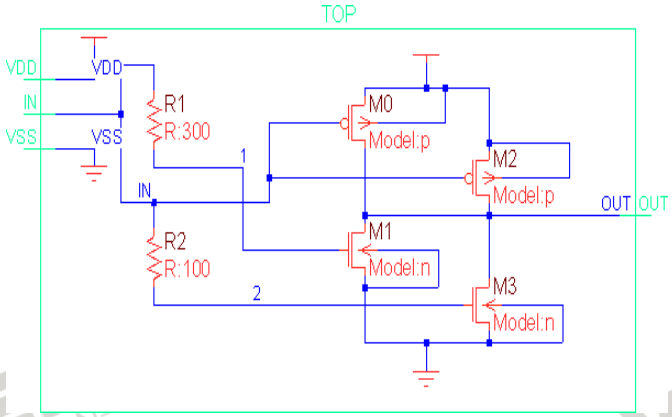
   Q. List the directory contents and, based on the names, try to predict the function of each item found:

   …….

2. Open the PERC rule file, perc_a.rules, in a text editor and review the contents of the file.
   - Based on your review, what is the needed PERC LOAD statement?
     ……..
   - What is the PERC statement that specifies the required resistor value?
     ……..
   - What is the name of the device specified by the above statement?
     ……..
   - What is the name of the property specified by the above statement?
     …….
   - What SVRF statement is required to enable use of the specified device property?
     ……

   Close the perc_a.rules file.

3. Edit control file job_c.rules to specify all required SVRF statements according to these requirements
   - The source netlist is lab2_c.spi
   - The PERC report is named perc.rep
   - The PERC LOAD statement specifies the needed information.
   - The SVDB directory name is svdb.
   - Include file perc_c.rules.
4. The schematic that represents the perc_c.spi netlist is shown here:



- Based on the problem statement at the start of this exercise, which devices do you expect to be reported by PERC?

  ……

5. Run the PERC job by entering the following command:

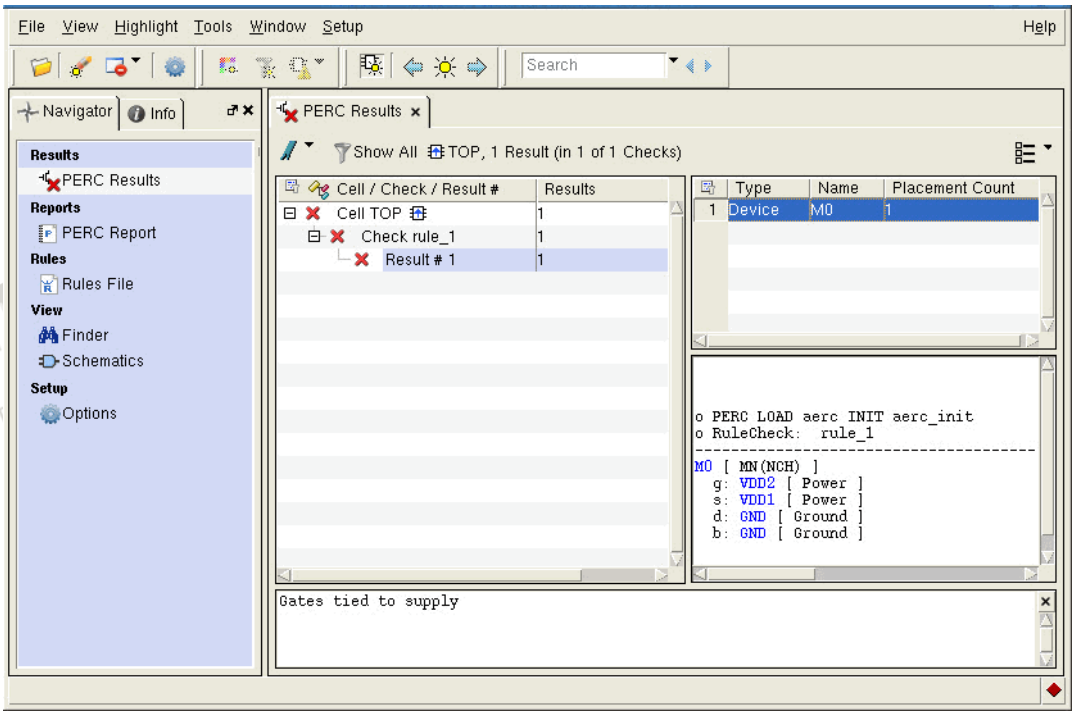
```
[user@maven-silicon] $ calibre –perc –hier job_c.rules >log
```

6. Review the log file for any warning or error messages. If there are any errors, edit your job control file and run the job again. Ask the instructor for help if you cannot determine the cause of the error.
7. Open the results in RVE by typing:

```
[user@maven-silicon] $ calibre -rve -perc svdb
```

Select **View → Tree Options→ Group By → Cell / Check / Result #.**

The RVE window will look like this:



8. Examine the results found in RVE
   - Did you get the results you expected? Explain.
9. Did you get the result you expected? Explain

   ……..

   If you did not get the expected results, review your control file
10. Close RVE. - This concludes the lab.

Lab 04 – Coding Calibre PERC

Exercise 1: Writing PERC Initialization Commands

In this lab, you will work with PERC initialization commands. You will write and test initialization statements based on job specifications.

This PERC application wants to find all nets in the circuit connected to floating input, power, and ground pins. **A floating pin is defined to be a pin that is not connected to either a pad or to another cell**. For this simple example, we can assume that each cell uses the following pin names:

- a (Input pin)
- b (Input pin)
- VDD (Power pin)
- VSS (Ground pin)

The top-level power and ground net names are specified via SVRF statements.

A PERC rule file has already been provided that contains a set of rules to find nets connected to floating cell pins. Your task is to write the required initialization procedure for this application.

1. While in your home directory, type:

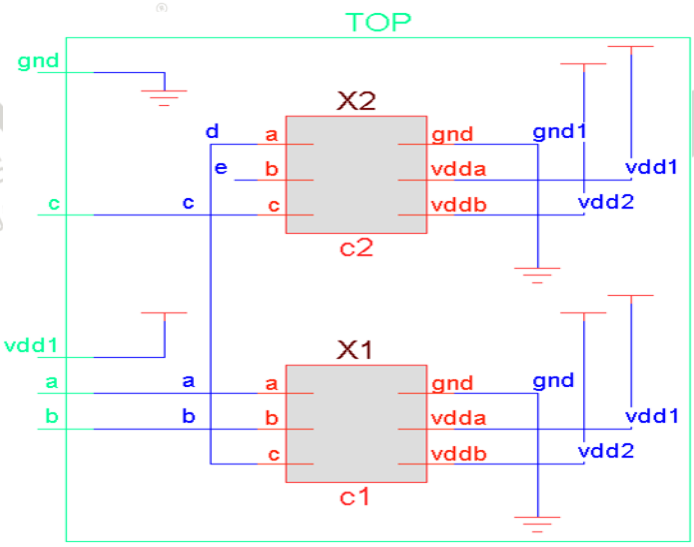
```
[user@maven-silicon] $ cd /HOME/user/Calibre_PERC/PERC_LABS/LAB04/Ex_1
```

2. Open the PERC rule file, perc_a.rules, in a text editor and review the contents of the file.
   - What do you think each PERC rule does?

     rule_1:

     rule_2:

     rule_3:
3. The netlist for this case is *lab2_b.spi*. The schematic is shown here:



- Based on the information you have, which nets should be reported?

……

4. Given the above circuit and PERC rules, answer the following questions:

   Q. What net types will you need to define?

   _____

   Q. Are any path definitions required?

   _____

5. Enter the net type definitions into the initialization procedure in *perc_b.rules*.

6. Test your answer by executing the PERC job:

```
calibre -perc -hier perc_b.rules > log
```

7. Review the results in RVE:

```
calibre -rve -perc svdb
```

8. If you did not get the expected results, review your initialization commands and try again. Close all applications when you are finished.

.

Exercise 2: Initialization Procedure Challenge

The purpose of the challenge exercises is to present you with typical coding problems which must be debugged. In each case, you will be given a completely-coded solution that contains one or more errors. Your task will be to locate and fix the problems and then verify that your fixes are good.

In this exercise, you will debug a problem in a PERC initialization procedure. In this case, the PERC rule file is intended to identify all MOS devices that have a path to the output pad OUTA from an s/d pin. Both direct connections to the pad and connection via a path through a resistor or through MOS s/d pins should be considered. When the application was run using the existing rule file, not all of the expected devices were found. Your CAD support engineer took a quick look at your code and told you that the problem is in your initialization procedure.

Now, you must find and fix the problem.

1. While in your home directory, type:

```
[user@maven-silicon] $ cd /HOME/user/Calibre_PERC/PERC_LABS/LAB04/Ex_2
```

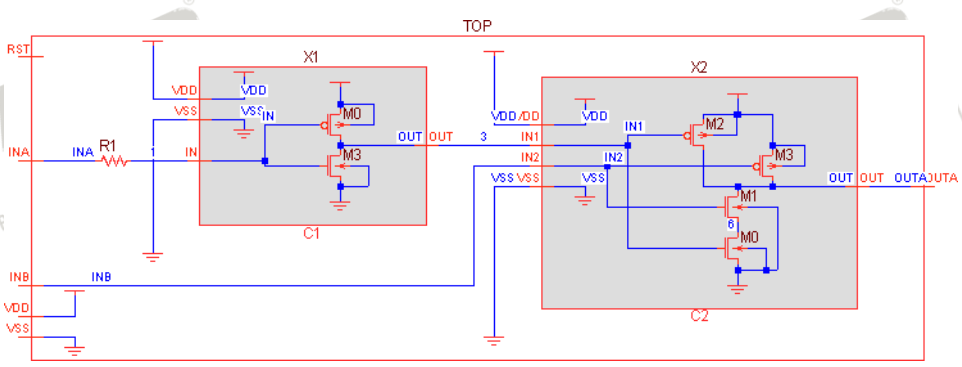
2. Open PERC rule file perc_c.rules in a text editor.

   Q. What do you think PERC rule rule_1 does?

   ……..

   We have not yet talked about the topology analysis commands but you might have figured out that *rule_1* will identify all MOS devices whose s/d pins have a path to the output pad OUTA. In this case, the intent of the rule is to find all such paths through resistors and through the s/d pins of MOS devices (in other words, through traditional SVRF PATHCHK paths).

3. The netlist for this case is *lab4_2.spi*. The schematic is shown here:



➢ Based on your answer to the question in Step 1, which devices should be reported?

……

4. Test your answer by executing the PERCjob:

```
calibre -perc -hier perc_c.rules > log
```

5. Review the results in RVE:

```
calibre -rve -perc svdb
```

   - Which devices were reported by this run?

     _____

   - Which device is missing from the output?

     _____

   All four MOS devices in the X2 instance should have been reported since a path can be traced from an s/d pin of each of these devices to the output pad OUTA. So, what is the problem?

6. Examine the initialization procedure in *perc_c.rules* carefully.
   - What do you think the problem is?

     …….
   - What statement do you need to add to this procedure?

     ……..……..
7. Edit file *perc_c.rules* to add the required initialization statement and re-run the job. If you do not get the expected results, edit the initialization procedure and try again.
8. This concludes the lab.