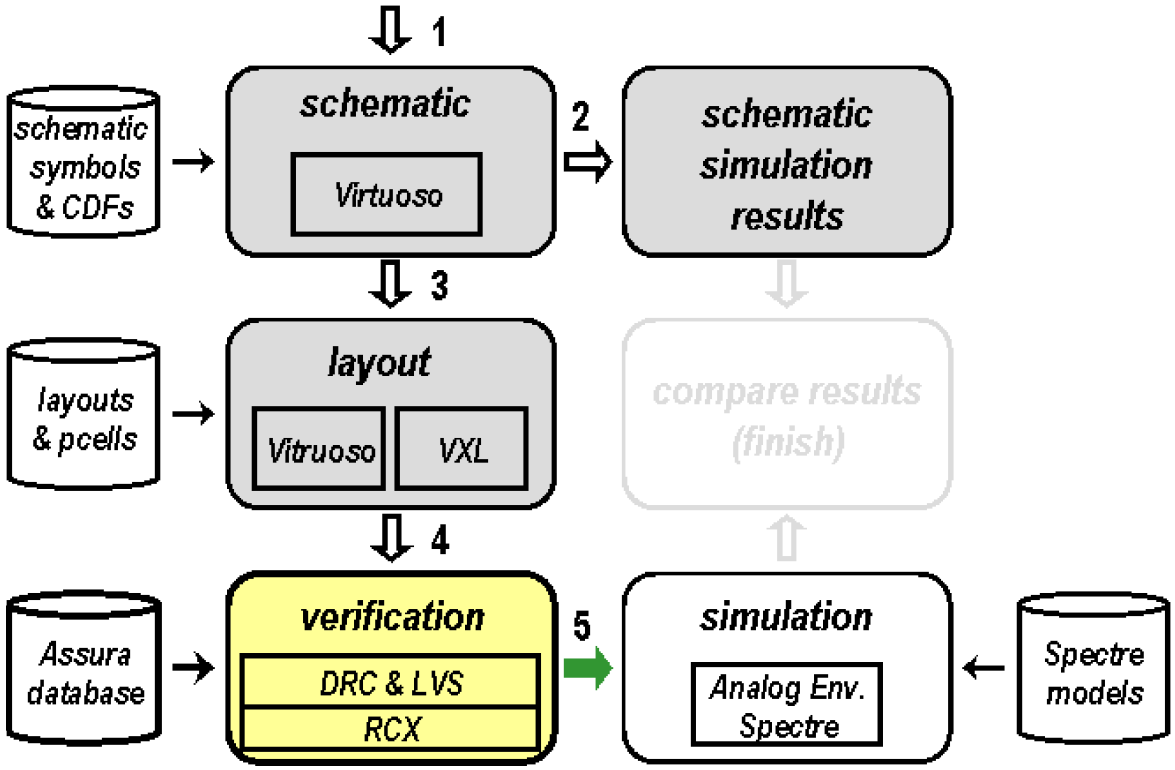
**EE115C: Digital Electronic Circuits** Design Flow UCLA Electrical Engineering

Prof. Dejan Markovic #4: Design Verification
Winter 2007

## EE115C Design Flow

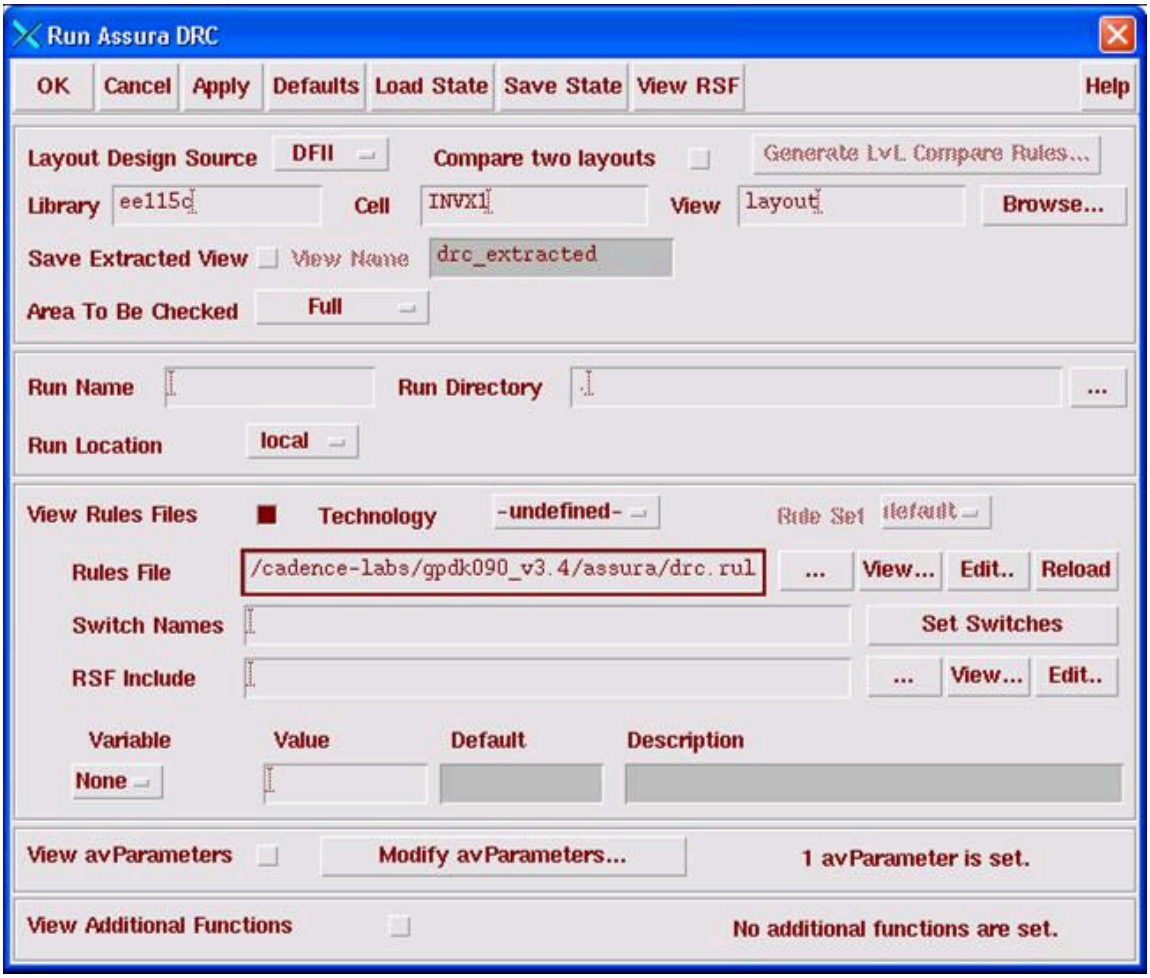
flow | main page | feedback

Your current position is highlighed in yellow. Click on the green arrow to advance through the flow. You can also go back to any of the previously completed steps (shaded) by clicking inside the shaded areas.




## Design Rule Checking (DRC) using Assura: INVX1 Example

As part of usual layout design experience, you will often need to perform design rule checking (DRC) to make sure that your design satisfies manufacturing rules. Let's do a DRC check on the above layout. Choose Assura > Run DRC, the following window will pop-up:




Specify `Rules File` as:

`/usr/public.2/ee115c/cadence-labs/gpdk090_v3.4/assura/drc.rul`

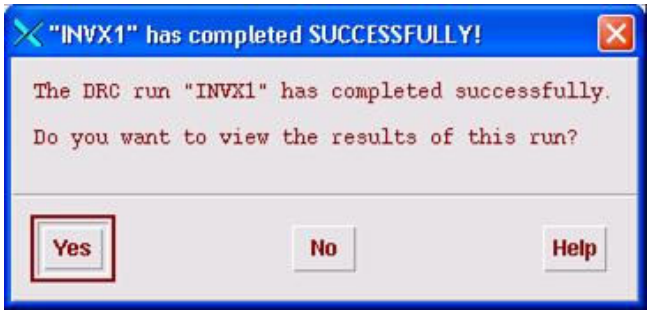
(remember: all technology-related files reside in the public folder under `ee115c/cadence-labs`) While it would be easier if you create local copies of the rules files and such, it is a good practice to learn how to work with a centralized database.

Click `OK` to star the Assura DRC check. During the run, following window will appear:

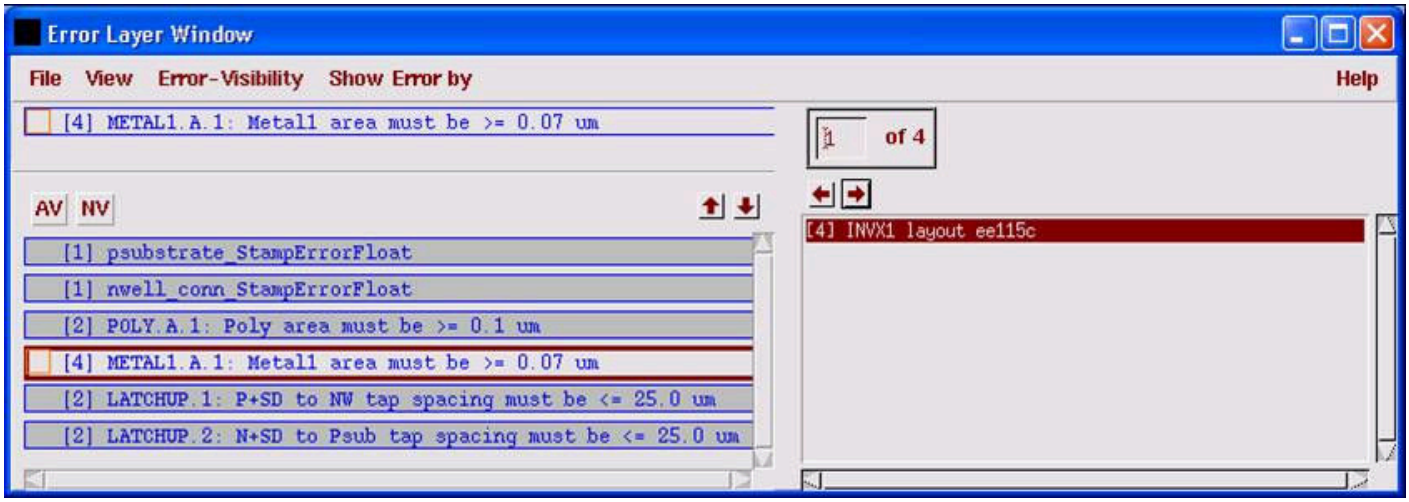

You can click `Watch Log File` to monitor the DRC progress.

After the run is complete, you will have another pop-up window.

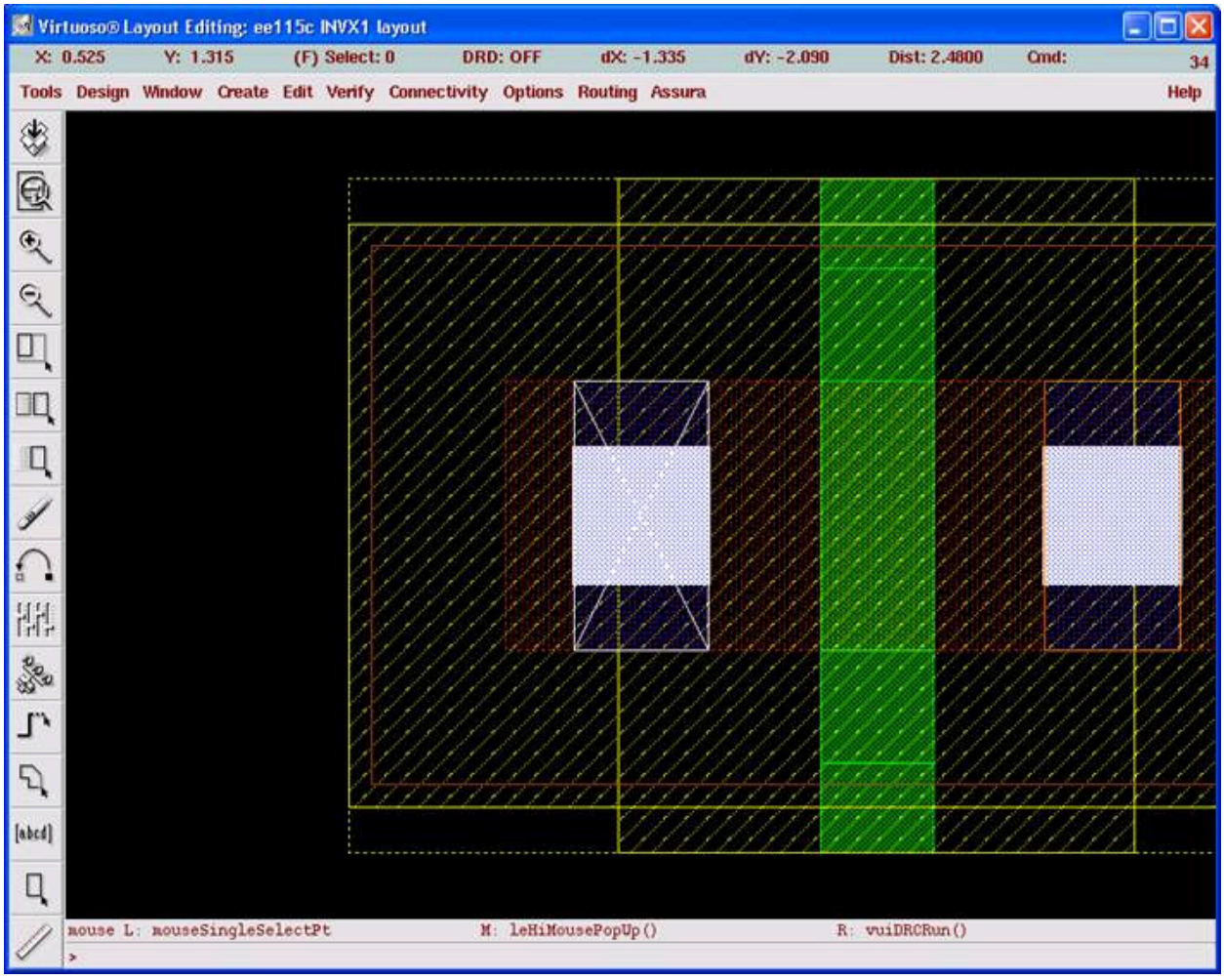
Click `Yes` to see the final report.



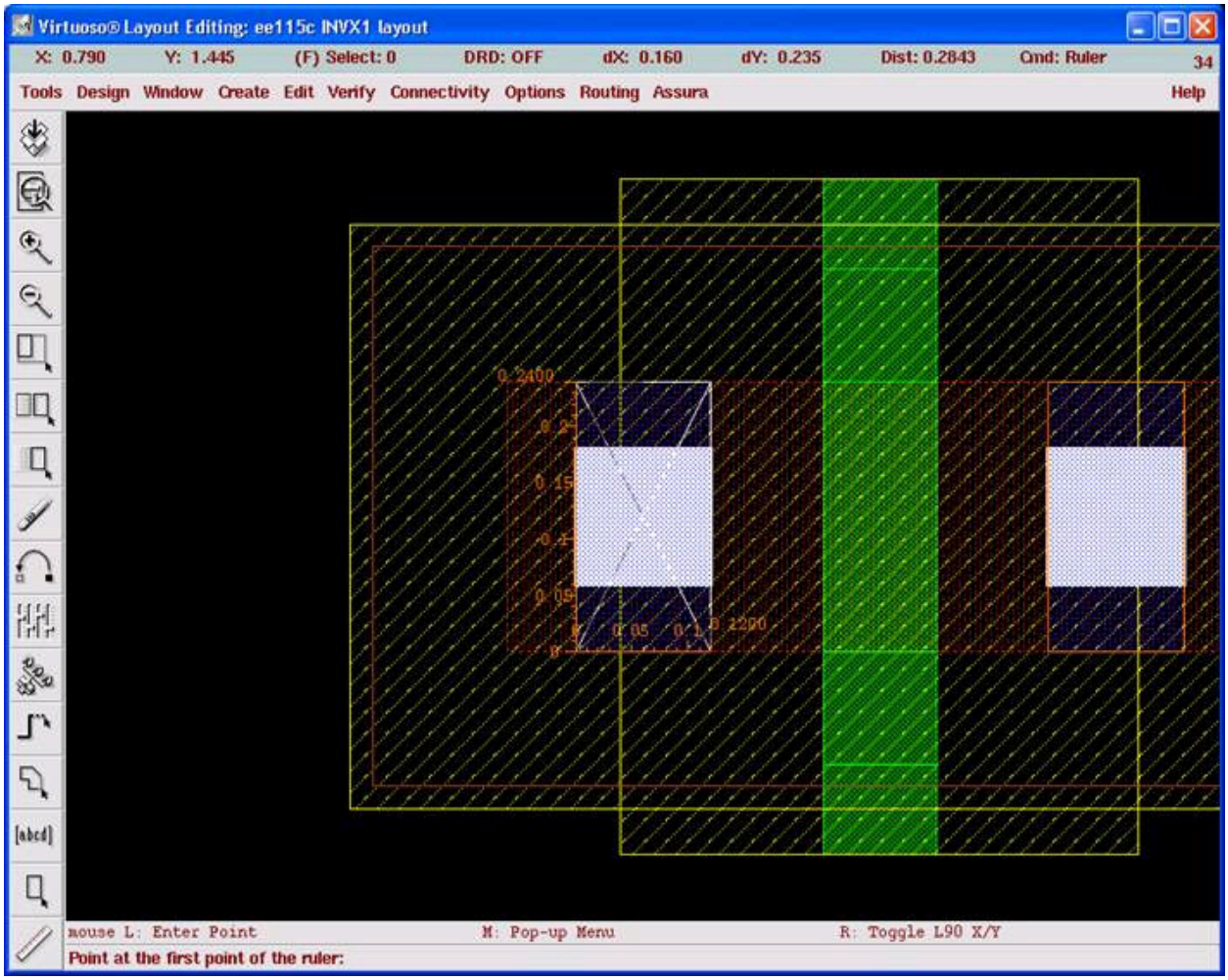

The Error Layer Window will appear:



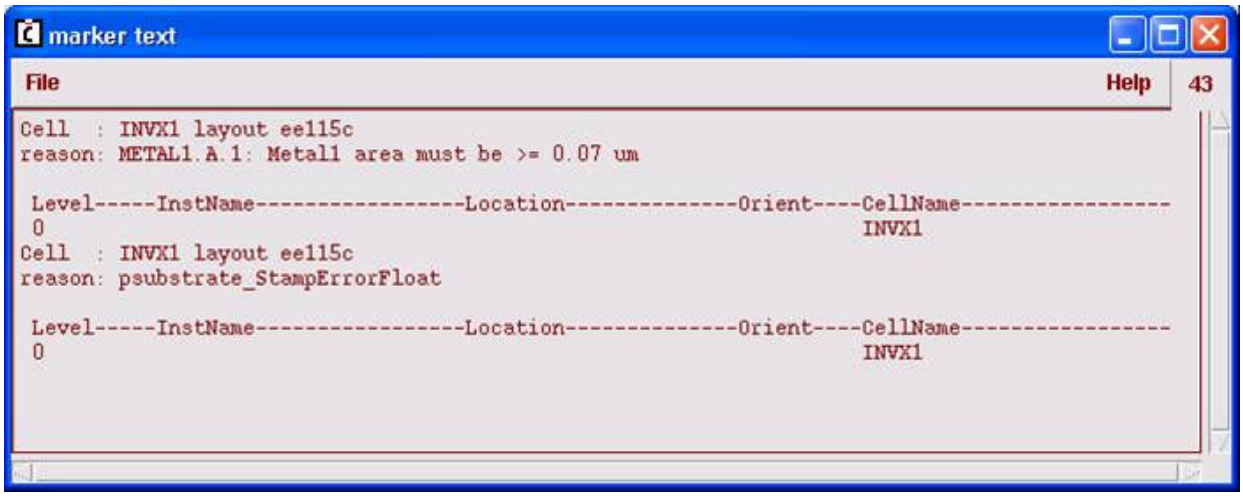

As you can see, there are several errors being reported. For each error listed, the number in square brackets indicates how many errors of this type occurred, followed by the description of error. You can scroll the errors by using arrows on the right. At the same time, if you look at your Virtuoso layout window, errors will be highlighted as you scroll. For example, the first out of four errors indicating violation of `METAL1.A.1` rule is highlighted in the layout below.



This error in particular says that the minimum metal1 area cannot be less than 0.07$\mu m^2$. To verify the size of the highlighted metal1 object, we can measure the object size using ruler. The ruler is invoked with shortcut key `k`. (all ruler marking can be deleted with `K`)




The size of the metal1 object is 0.24 x 0.12 = 0.0288$\mu m^2$, which violates the rule. To get more information about the error, in the `Error Layer Window` choose `View > Explain`, and then click on the highlighted object in layout. Marker text window will show up providing details about the highlighted error.




Explanation of the design rules can also be found in the gpdk090 technology documentation:

`/usr/public.2/ee115c/cadence-labs/gpdk090_v3.4/docs/gpdk090_DRM.pdf`

## A Note about Assura

Assura is a physical verification tool from Cadence. It replaces old tool Diva, which used to work well for technologies up to 0.18$\mu m$ node. For the deep sub-micron technologies below 0.18$\mu m$ such as our 90nm technology, Assura provides more accurate results than Diva, particularly in parasitic extraction. The Assura extraction is based on advanced 3D transistor-level parasitic R and C extraction. In terms of CAD database, Assura can replace Diva for the Cadence design framework II (DFII) database, versions 4.4 and later. Unlike Diva which is a flat verification tool, Assura offers hierarchical verification capability.

For information and help pages, invoke cdsdoc from your Unix command prompt and open:

`Assura DRC/LVS > Physical Verification User Guide`

The manual is the reference to Assura and it contains lots of information that you will not find in this simple tutorial.

Before continuing to the next step, you may want to go through section on wiring and contacts from the Layout Capture step.

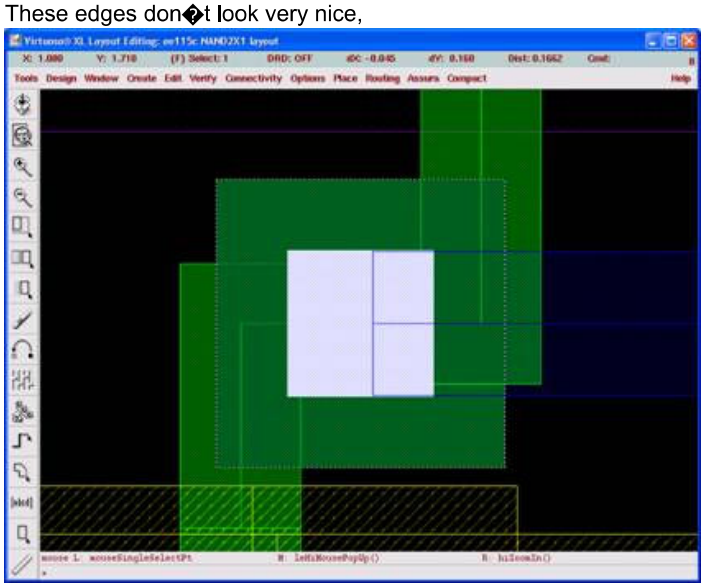
Invoke Assura DRC check one more time. If you have done everything correctly, your design should be DRC-clean and you should get the following message:


## Example of a Contact (from the NAND2X1 Gate)

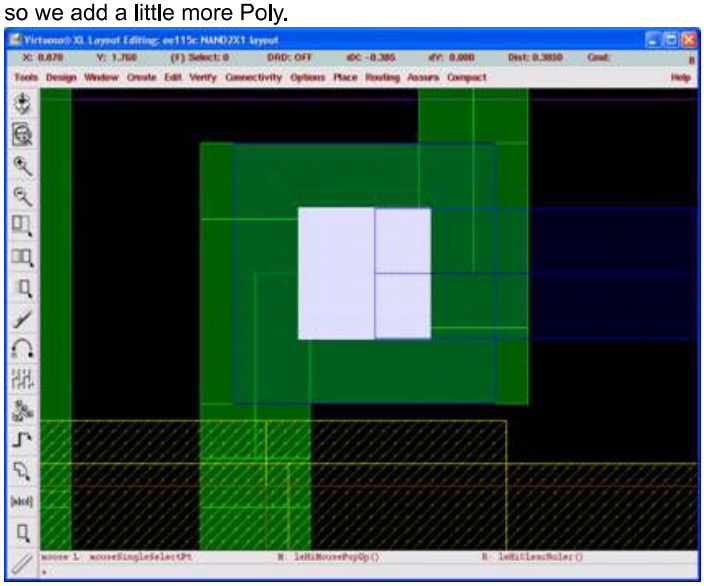
Two more things are we are ready to verify DRC and LVS:
- take a close look into the poly contact and fix the edges
- label pins

The poly contact which was automatically created using `Create Path` command looks like this:

These edges don't look very nice,



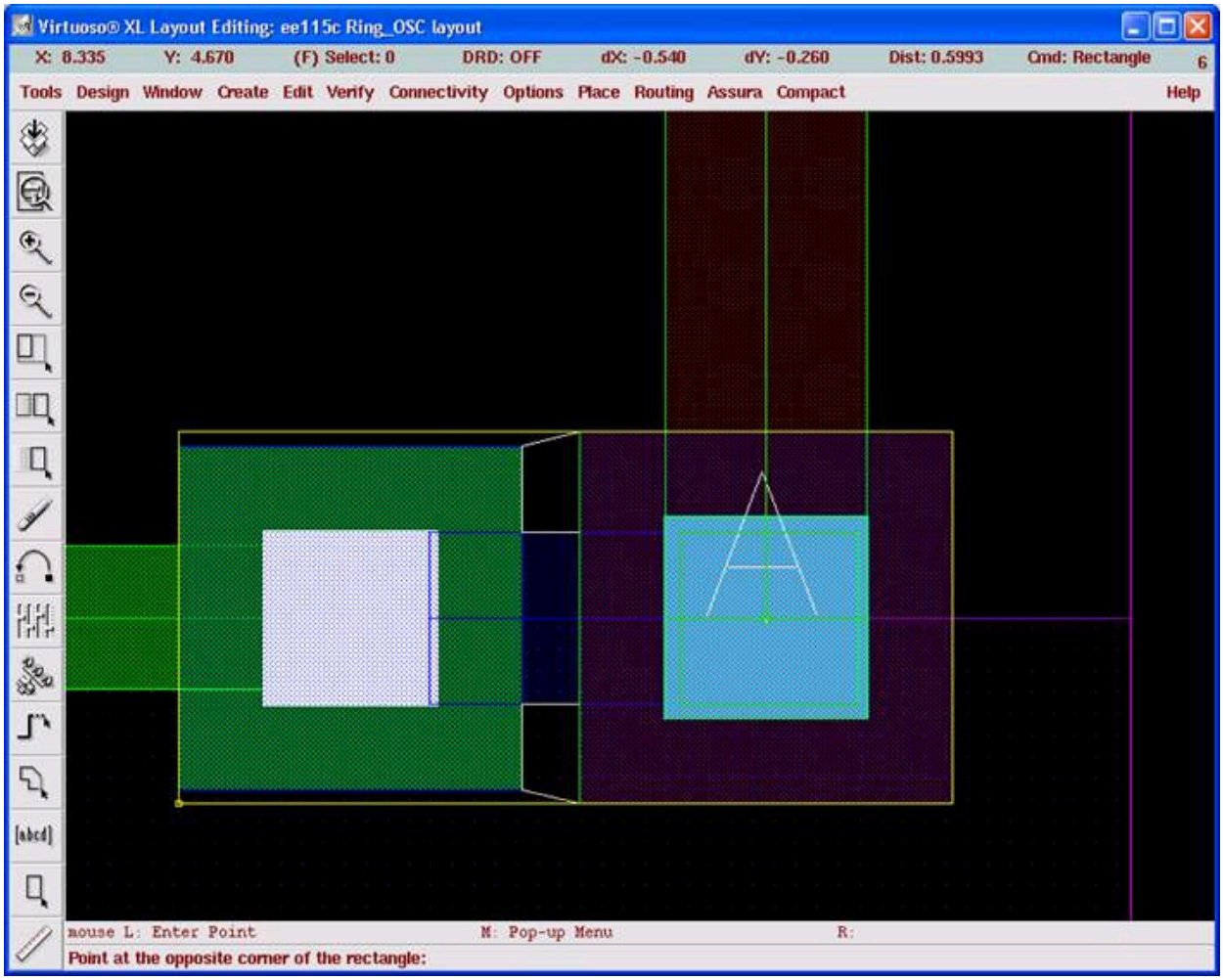
so we add a little more Poly.



(Note: the contact shown on the left would still pass DRC)

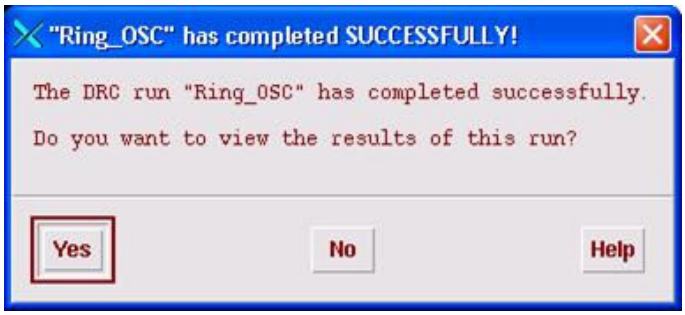
## Example of a DRC Error (Ring_OSC Example from Tutorial 5)

If you placed `M2/M1` and `M1/Poly` close, then you might run into `M1` spacing error like the one illustrated below:



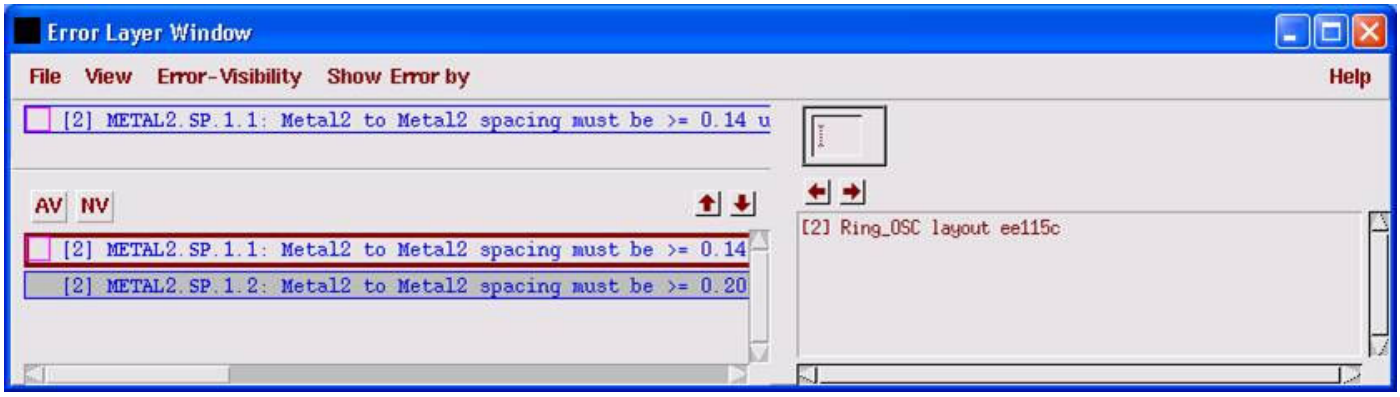

You can cover both contacts with a Metal1 polygon to fix this error.

## Example of a Metal Spacing DRC Error (Ring_OSC Example from Tutorial 5)

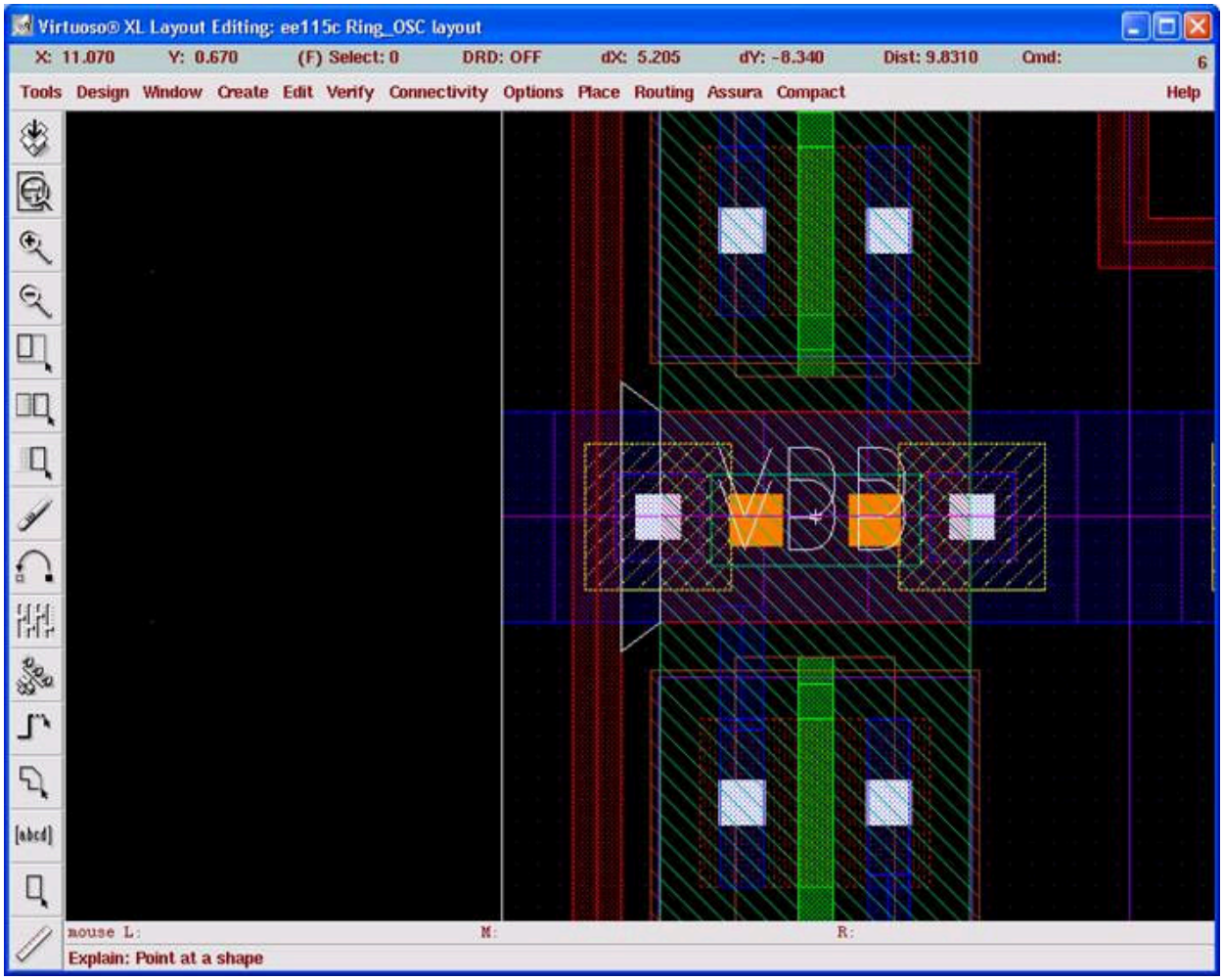



Click Yes to see the final DRC report.

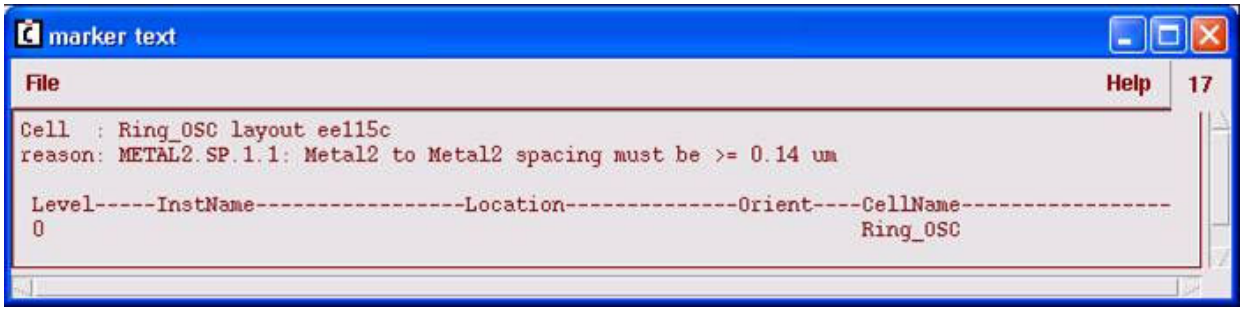
Error Layer Window pops up showing few DRC erros:




This indicated Metal2 spacing problem. Click on arrows above to see the errors in layout. It looks like Metal2 wires are too close to Metal2 from the contact stack as highlighted below:




For more details, in the `Error Layer Window` you can click `View > Explain`, and then click on the highlighted shape above. Following window will appear:



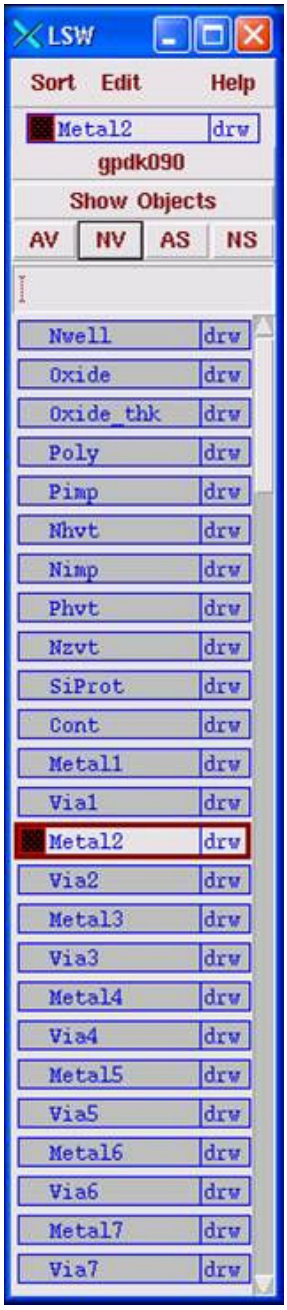

This is the `marker text` window that provides details about the DRC errors. You can also find this information in the manual (may also review Tutorial 3):

`/usr/public.2/ee115c/cadence-labs/gpdk090_v3.4/docs/gpdk090_DRM.pdf`

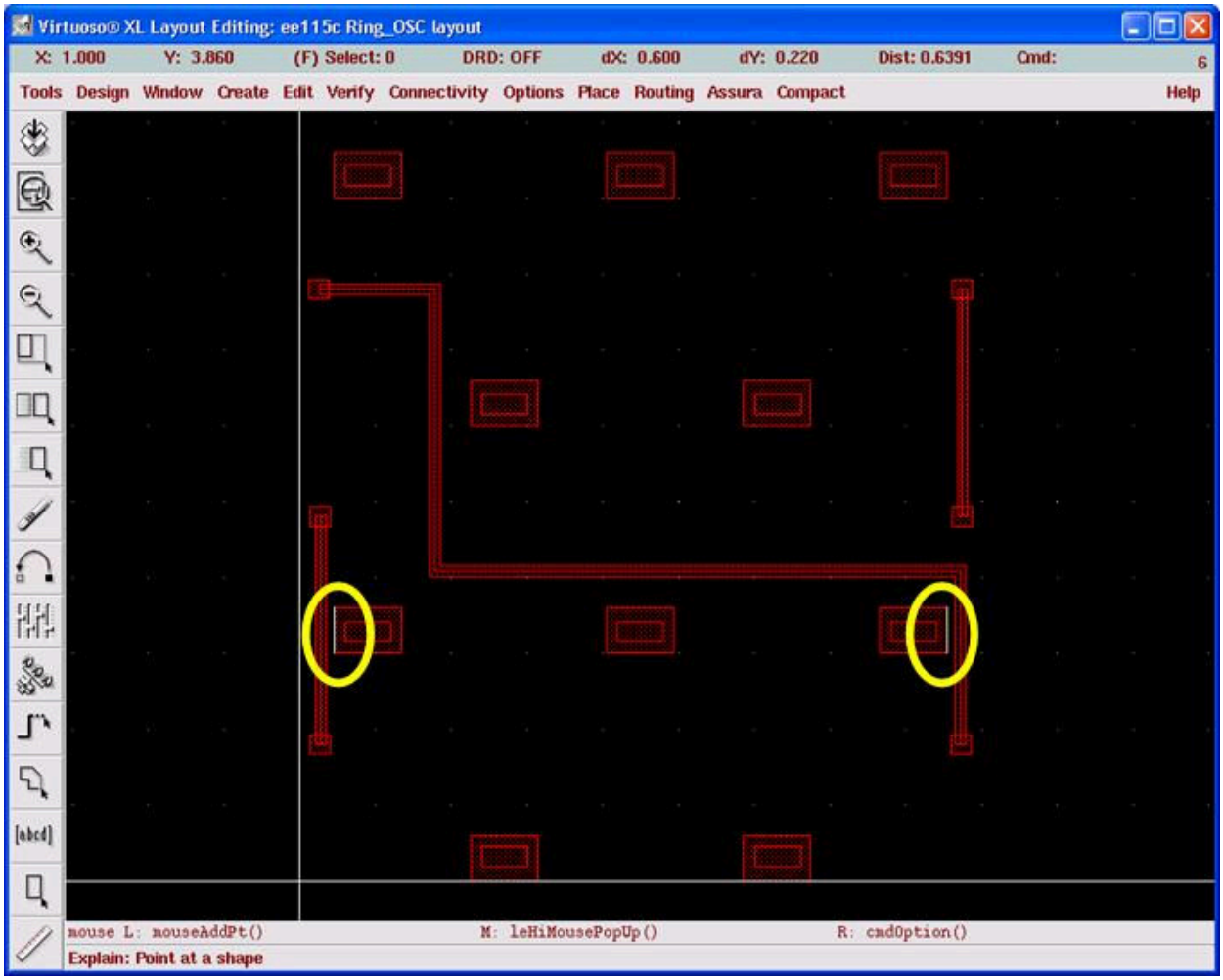
Since we want to fix only `Metal2` spacing errors, we can select to display only this layer. Go over to the `LSW` window.

Select `Metal2 drw` and click `NV`.

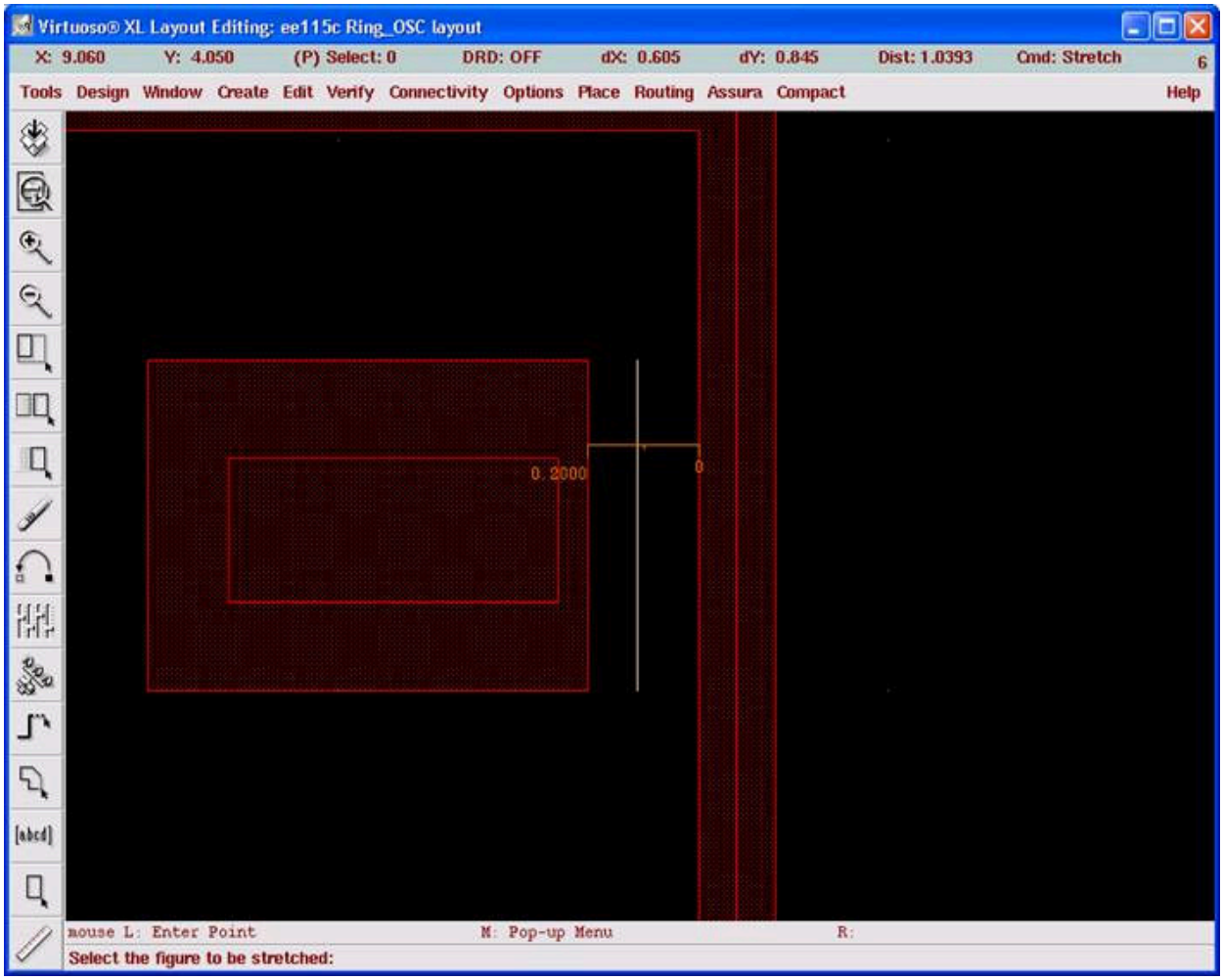
You will see that only `Metal2` remains active.




Now, go to the layout editor and refresh the window using `Window > Redraw` (hotkey `Ctrl-r`). Only `Metal2` will be visible in layout as shown below:

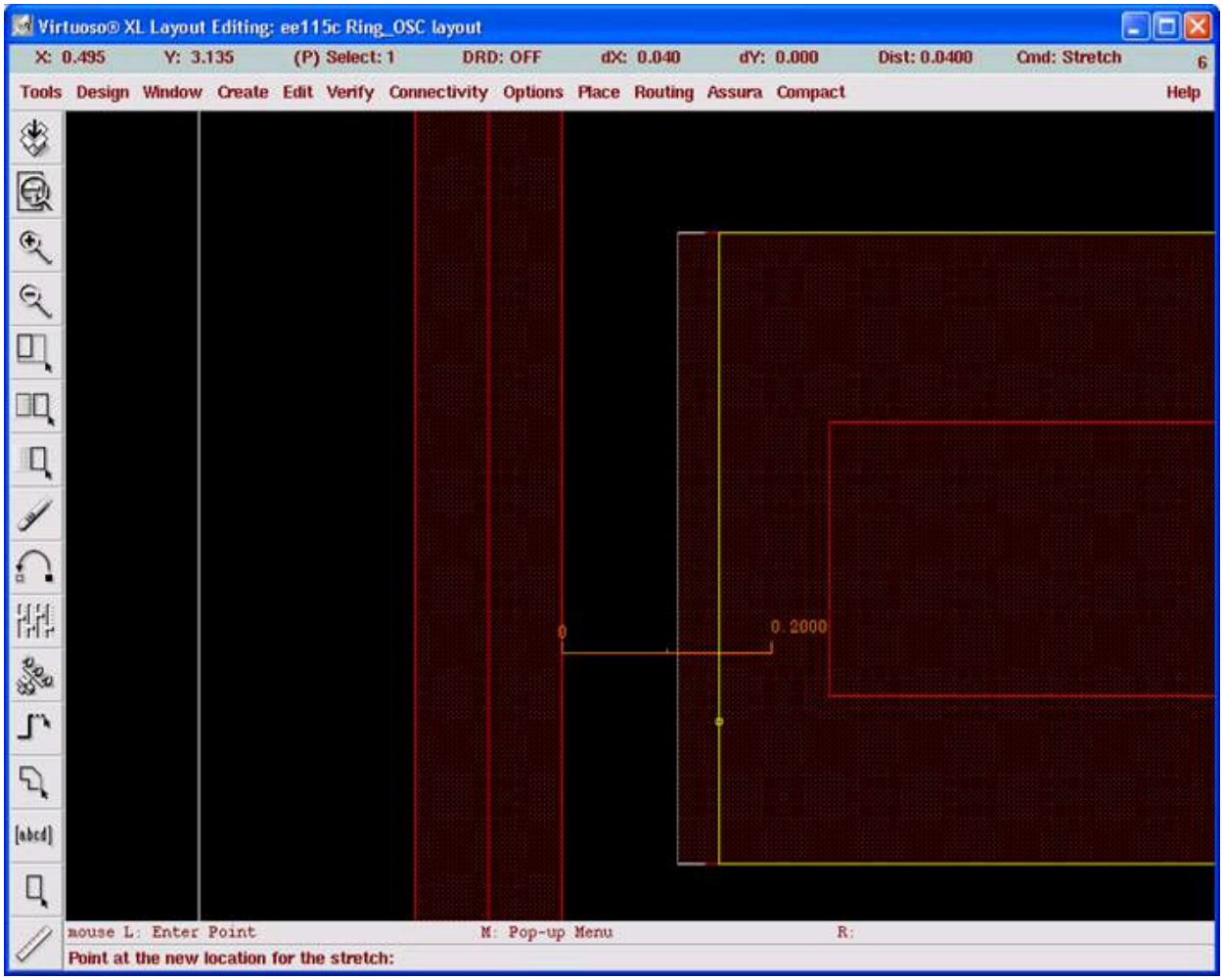



DRC violations are indicated with white lines (also circled for better emphasis). The easiest way to fix these errors is to shrink the Metal2 area in the contact stack. Use ruler to measure 0.2μm away from the signal lines and shrink the Metal2 from the contact stack accordingly. This process is illustrated below:




## Layout Editing: Stretch Command

You can resize the object by stretching (`Edit > Stretch`). After you enter stretch mode (hotkey: `s`), left-click and sweep over the edge you would like to stretch, that edge will become highlighted, left-click to take the edge and stretch the object. The stretching is illustrated below:




Click the `AV` button in the `LSW` window and press `Ctrl-r` to refresh the layout. Now, let's run DRC one more time (if you haven't closed previous run, first select `Assura > Close Run`).

Your design should be DRC clean!


Close the run: `Assura > Close Run`.

## Documentation: Using the Design Rule Manual (DRM)

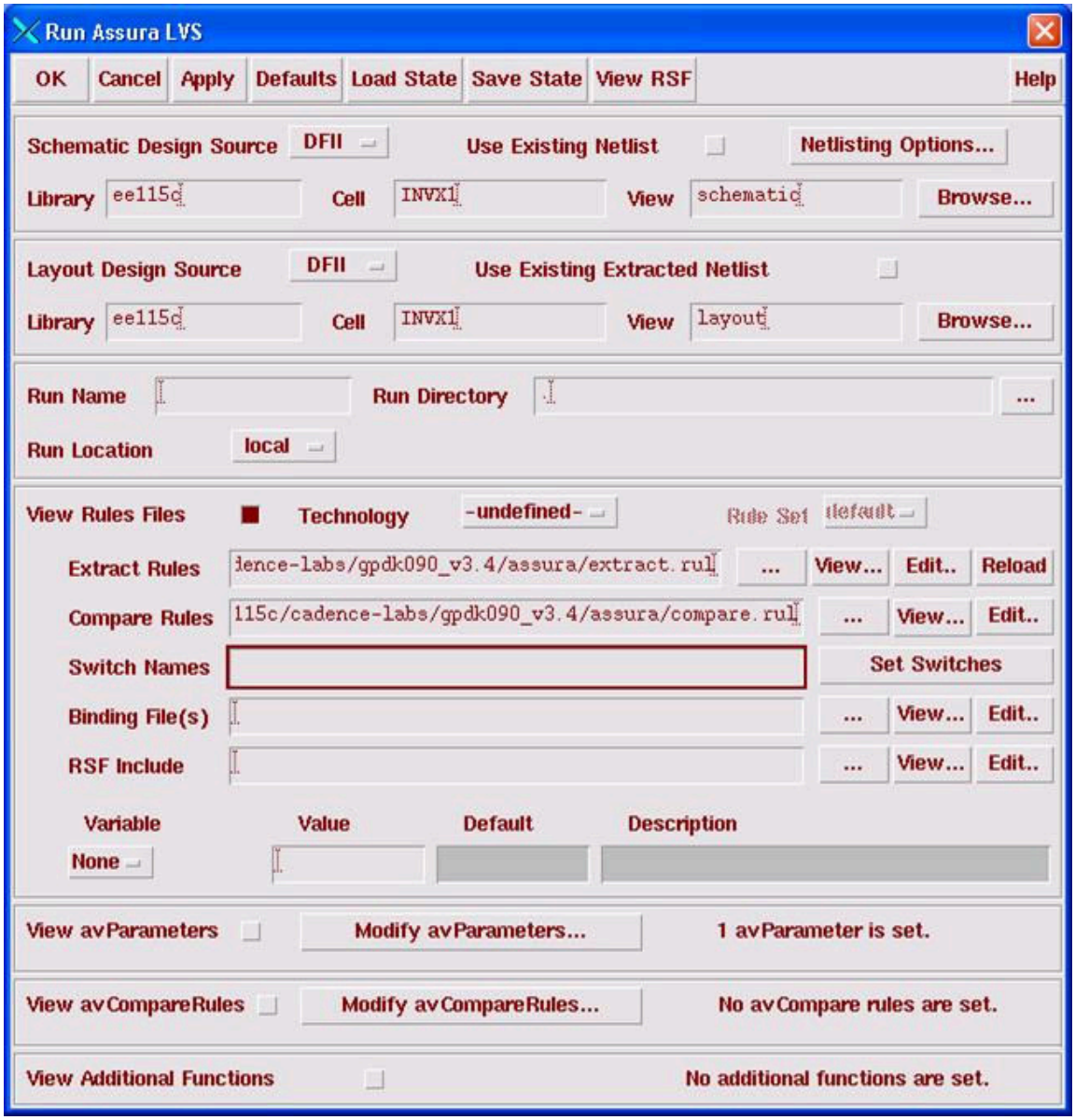
Use ruler to measure Poly enclosure around the contact (0.06μm). The `Minimum Poly to Contact enclosure` rule is 0.04μm, unless for end of line. If you are not sure about the rules, check the DRM manual. In this case, check rule: `CONT.E.2`, page 40 of the manual:

`/usr/public.2/ee115c/cadence-labs/gpdk090_v3.4/docs/gpdk090_DRM.pdf`

## Layout Versus Schematic (LVS) Check using Assura: INVX1 Example

As you can see, we still can't declare complete victory! We also have to verify that the layout we just designed matches the schematic created in Tutorial 2. This verification is accomplished by checking Layout Versus Schematic (LVS) rules in Assura.

In `Virtuoso Layout Editing` window, invoke `Assura > Run LVS`. The following pop-up window will appear:




Make sure the settings in the `Schematic Design Source` and the `Layout Design Source` are set as shown above. Also, double check that the `Extract Rules` and the `Compare Rules` are the following:

```
/usr/public.2/ee115c/cadence-labs/gpdk090_v3.4/libs.cdb/gpdk090/extract.rul
/usr/public.2/ee115c/cadence-labs/gpdk090_v3.4/libs.cdb/gpdk090/compare.rul
```

Delete the `Switch Names` field. Click `OK` to start the LVS run. The LVS environment is similar to the DRC environment. You can also monitor the progress of your LVS run:


Choose to `Watch Log File`

Click `OK`.

After the LVS run is finished, you should see the following pop-up window:


Click `Yes`.

The LVS Debug window will appear (in case you need it for debugging).


**Congratulations! Your design is LVS-clean!**

Close the LVS run from `Assura > Close Run`.

Our `INVX1` design is now complete.

## Final Verification: Extraction (Ring_OSC Example from Tutorial 5)

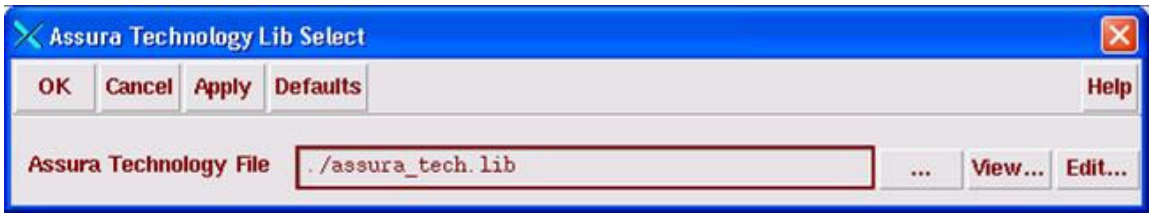
We are now going to introduce another step in the verification chain: layout extraction and post layout simulation. Copy Assura technology file to your cadence-labs directory:

```
> pwd
> /w/fac.01/ee/dejan/ee115c/cadence-labs
> cp /usr/public.2/ee115c/cadence-labs/gpdk090_v3.4/assura/assura_tech.lib .
```

Modify the `assura_tech.lib` file to read as follows:

```
DEFINE gpdk090 /usr/public.2/ee115c/cadence-labs/gpdk090_v3.4/assura
```

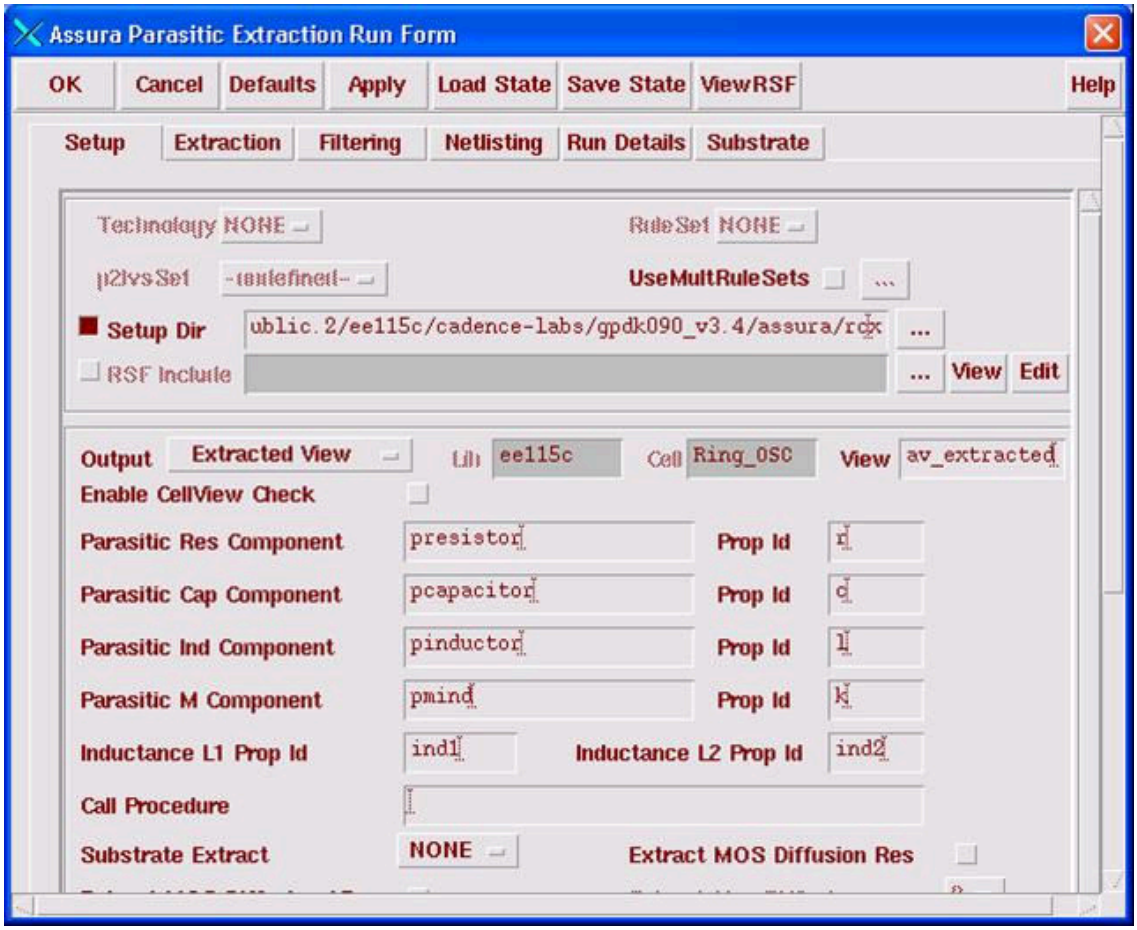
Set up `Assura > Technology`:




Click `OK`.

To start the extraction, go to `Assura > Run RCX` (note: if you previously did `Assura > Close Run`, this menu won't show up, so you need to do `Assura > Open Run`).

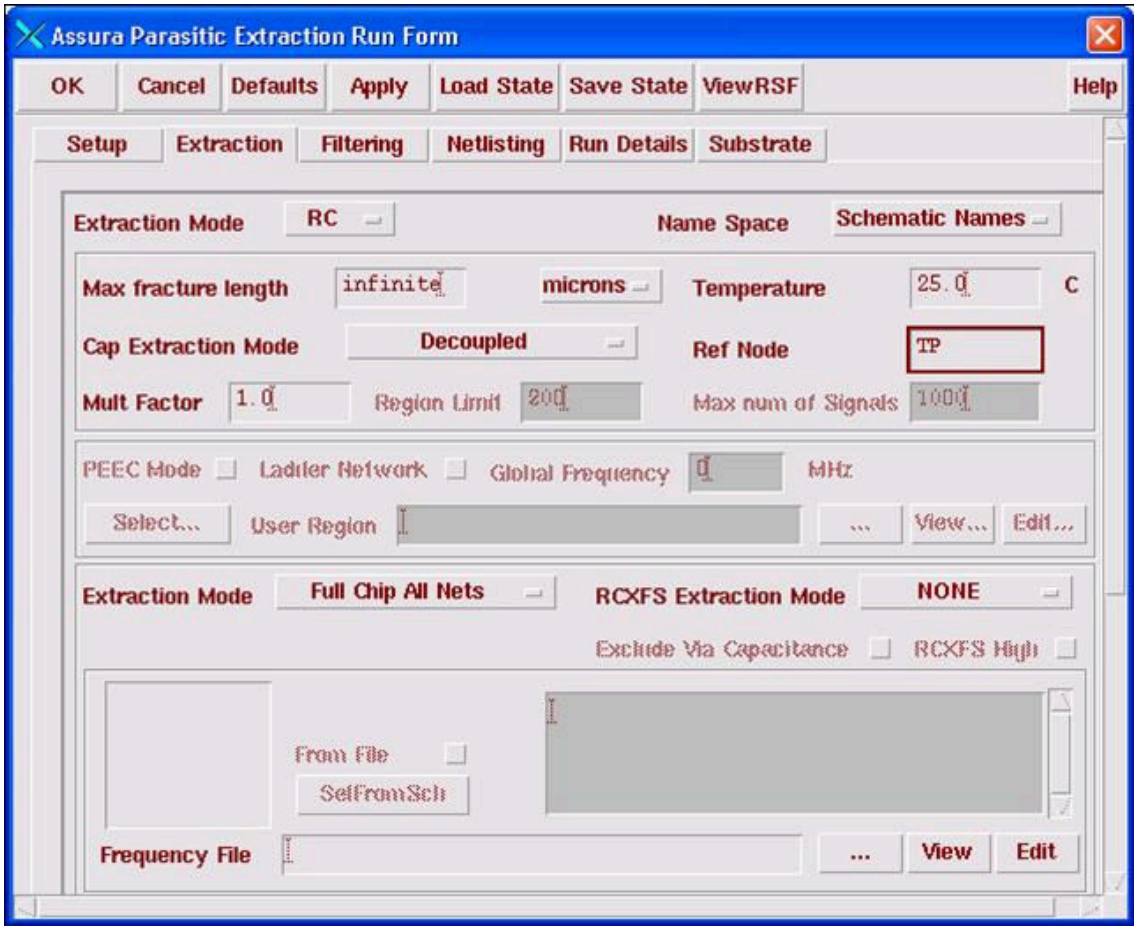
Following window will pop-up:




In the `Setup` tab, choose `Extracted View` for `Output`. Make sure `Setup Dir` is the following:

`/usr/public.2/ee115c/cadence-labs/gpdk090_v3.4/assura/rcx`

Click on the `Extraction` tab.



Choose RC as the Extraction Mode (both resistance and capacitance) and Schematic Names as the Name Space (since we didn't label any nodes in the layout). Also type TP as the Ref Node.

Walk through other tabs just for exercise, but don't change anything. Click OK.

When the Assura run is complete, the following window pops-up:

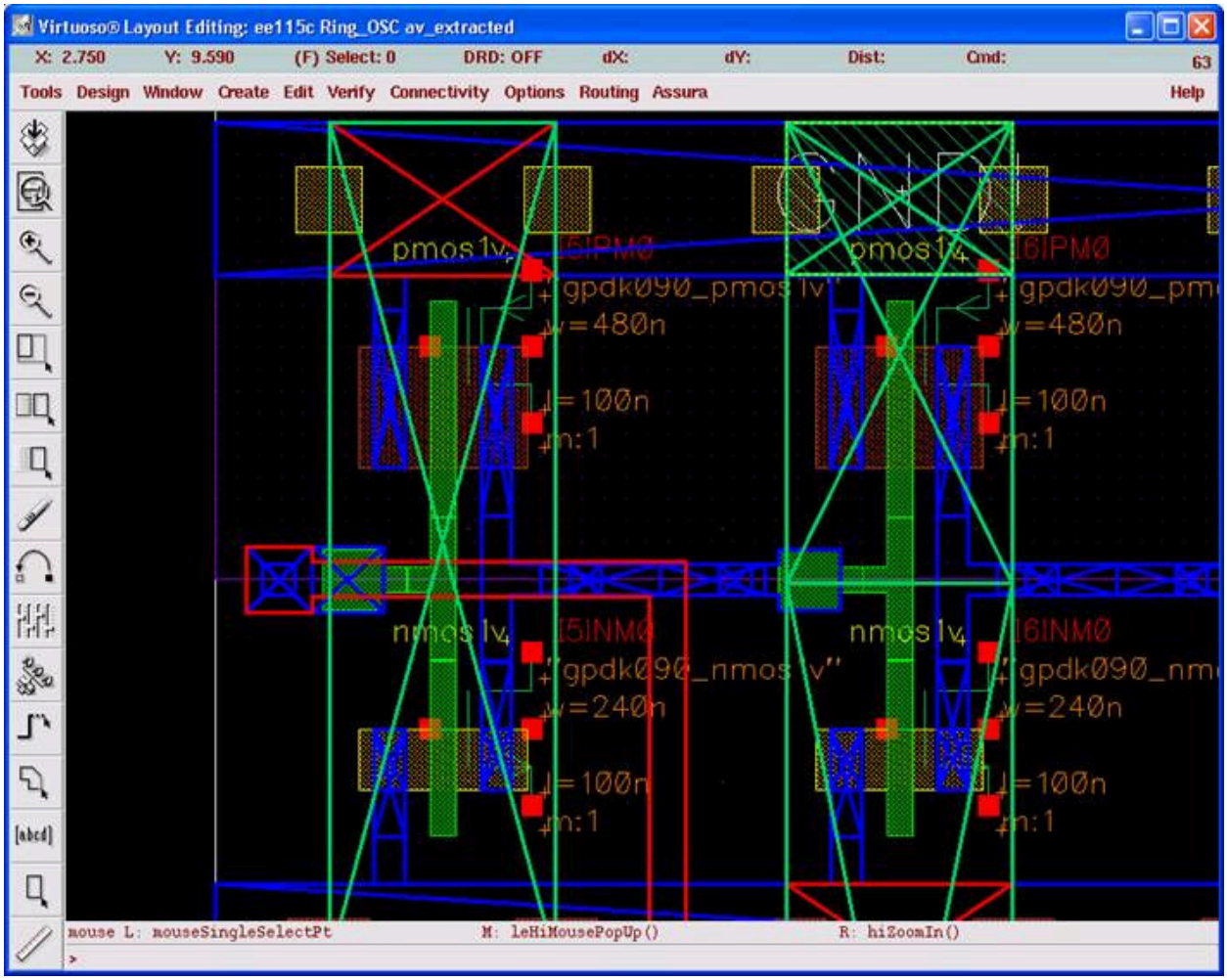

Click Close.

This will be part of the run report (sitting in your cadence-labs directory):

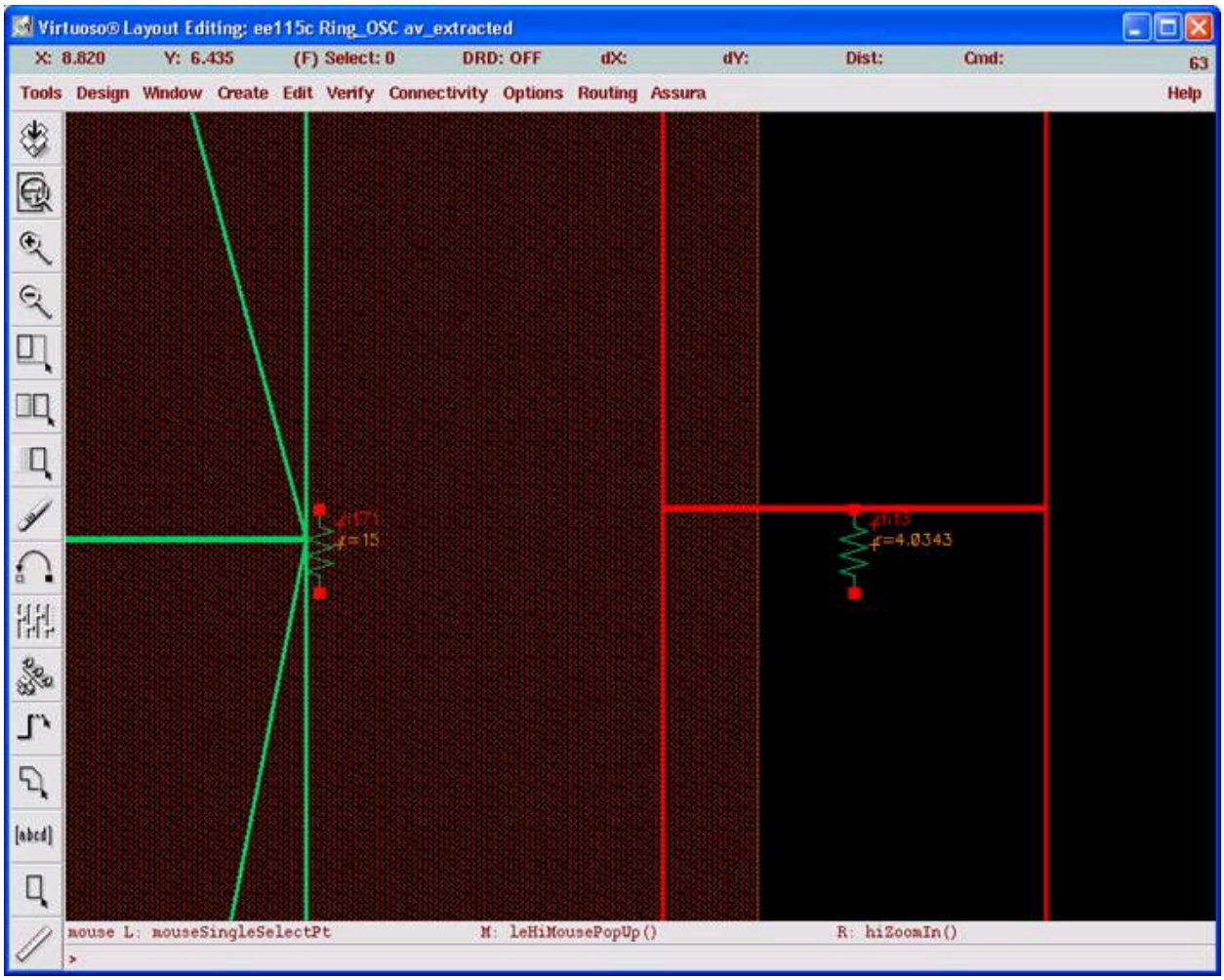
rcx.Ring_OSC.log

```
Summary for ee115c/Ring_OSC/av_extracted

instance count totals:

    lib             cell            view                    total
    analogLib       pcapacitor      symbol                    167
    analogLib       presistor       symbol                    322
    gpdk090         nmos1v          ivpcell                    15
    gpdk090         pmos1v          ivpcell                    15

extracted view creation completed
cpu: 0.27  elap: 2  pf: 57  in: 55  out: 88  virt: 55M  phys: 0M

Finished /usr/apps/cadence/ASSURA315/tools/assura/bin/rcxToDfII

Run ended: Thu Dec 28 01:16:41 2006

*****  Assura terminated normally  *****
```

Now let's go to the Library Manager window and open av_extracted cell view. Zoom in to the top left corner and you will see layout annotated with transistor symbols indicating extracted components:




Similarly, you can zoom in to see parasitic resistances:



*Last Modified on December 31, 2006 by Dejan Markovic*