# IPRAIL MANUAL

*Sambuddha Bhattacharya, Roy Hartono, Nuttorn Jangkrajarng and C-J. Richard Shi*
*MSCAD Research Group, University of Washington, USA*

## 1. Introduction

Analog circuit performance is strongly dependent on the layout intricacies. Several layout features like symmetry and matching, relative placement of devices, alleviation of parasitic effects are incorporated into analog layouts by expert layout designers. Complete automation of analog layout generation similar to digital design methodology is rather difficult.

Our analog layout automation methodology is based on the concept of design re-use. IPRAIL re-uses existing analog layouts crafted by layout designers and retargets them to different technologies and specifications.

## 2. Methodology

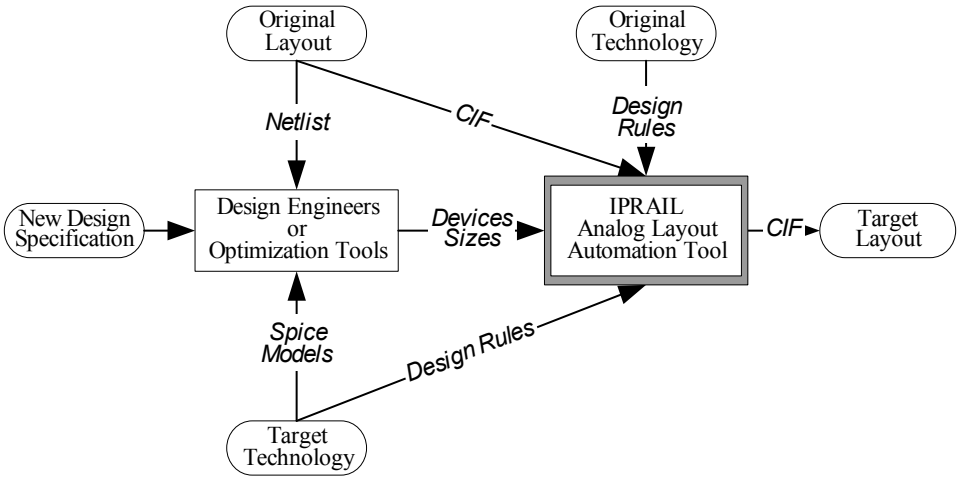


**Figure 1: IPRAIL Layout Automation Framework**

As illustrated in Figure 1, the *original layout* and its technology information are first fed into IPRAIL. The device sizes under the new specifications are obtained either by manual simulations or from an analog circuit synthesis tool. First, IPRAIL converts the original layout into a *resizable symbolic template*. It then generates the new layout, henceforth called *target layout*, by imposing the target process design rules and new device sizes as constraints on the symbolic template. The entire process of automatic creation of symbolic template and generation of target layout takes a few minutes of CPU time.

## 3. Installation

The platform required for *IPRAIL* is SunOS 5.8 (Solaris 8). *IPRAIL* also requires the X11 graphics package normally available in the Solaris environment. Please follow the instructions below to setup the environment for *IPRAIL*.

1. Copy the *iprail_demo.tar.gz* file into your home directory and deflate it.
    a. *gunzip iprail_demo.tar.gz*
    b. *tar -xvf iprail_demo.tar*
2. Check the directory structure created in your home.
    a. *~yourhome/demo*
    b. *demo* has the following subdirectories: *bin, env, icn, lib, sim* and *cadence*.
3. Execute the script *install.csh* from the *demo* directory. It includes some Cadence path setup in the directory *~yourhome/demo/cadence*.
4. *source .iprail* in *~yourhome/demo*
5. Prior to invoking IPRAIL, please exit *netscape* and other programs that use Graphics. Otherwise some technology layers may remain transparent in IPRAIL. That may be confusing for an inexperienced user.

This completes the environment setup for *IPRAIL*.

## 4. Tutorial

We will work with two examples of analog layouts.

a. Cascode Operational Amplifier
b. Two-stage Operational Amplifier

The schematics of the two examples are shown in the following figures.

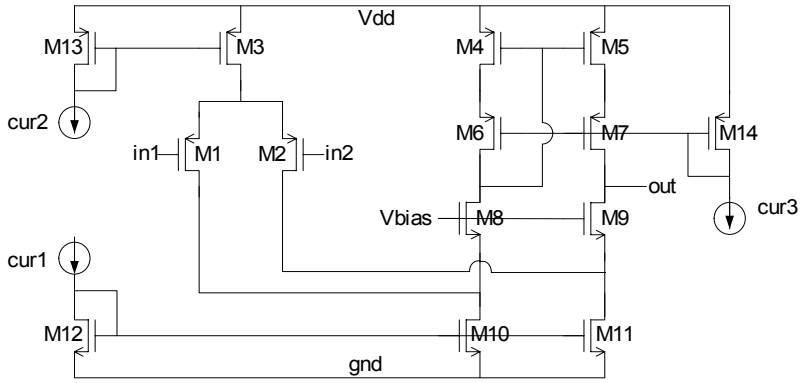


**Figure 2. Schematic of a Folded-cascode Single Ended Operational Amplifier**

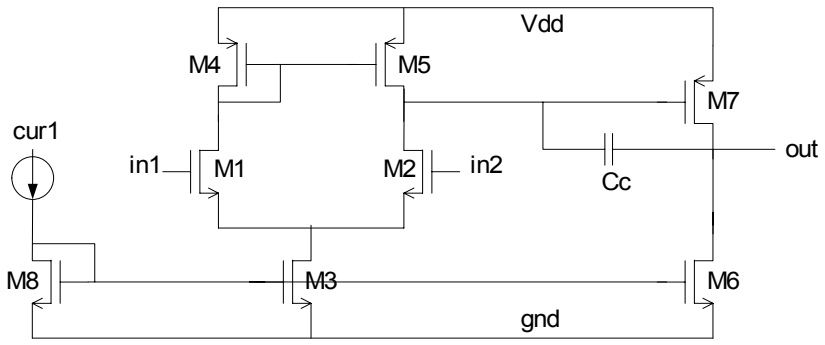


**Figure 3. Schematic of a Two-Stage Operational Amplifier with Compensation**

The handcrafted layouts for the two examples in TSMC 0.25um technology are available as CIF files. Our goal is to automatically generate corresponding layouts in the TSMC 0.18um technology. It is assumed that the design and simulation for the specifications in the new technology (0.18um) is already complete and available.

The layout (CIF), device sizes and symmetry information for the two examples are available in the directories *demo/bin/Cascode* and *demo/bin/TwoStage* . The technology and design rule files are available in the directories *demo/bin/Technology* and *demo/bin/DesignRule* respectively. The *demo/bin/BridgeFile* directory contains a *bridgefile* which relates between the 0.25um and 0.18um technologies. All these files need to be loaded into IPRAIL at some stage. At a later stage, some of these files may be made transparent to the user.

The complete tutorial is divided into five subsections. The first three subsections step through the entire automatic layout generation flow. The last two subsections step through the Design Rule Checking (DRC) and Post-Layout Simulation for verifying the correctness of the automatically generated layout. It is assumed that the Cadence environment is available to the user.

### 4.1 Loading the Original (0.25um) Layout into IPRAIL

This sub-section describes the process for invoking IPRAIL and loading the original hand-crafted layouts. Once the layout is loaded, the circuit structure is automatically extracted from the layout and stored in the database.

1. Go to *~yourhome/demo/bin* directory
2. Invoke IPRAIL by typing *iprail* in the Unix command line. Maximize IPRAIL GUI by clicking the button on top-right corner.
3. Load the IPRAIL's technology file. (*.tch)
    a. Click on 'Load' in the menu bar
    b. Click on 'Technology'
    c. Select the Technology file *Technology/MCNC.tch*
    d. Click 'OK'

4. Load the bridge file for the loaded technology
    a. Click on 'Load' in the menu bar
    b. Click on 'BridgeFile'
    c. Select the Bridge file *BridgeFile/bridgeFileTSMC*
    d. Click 'OK'
5. Load the initial design rule
    a. Click on 'Load' in the menu bar
    b. Click on 'InitialDesignRule'
    c. Select the design rule file *DesignRule/UWtsmc25.tech*
    d. Click 'OK'
    e. Clicking the correct technology file is very important for successful operation of IPRAIL. Make sure you've chosen the right technology file.
6. Load the target design rule
    a. Click on 'Load' in the menu bar
    b. Click on 'TargetDesignRule'
    c. Select the target design rule file *DesignRule/UWtsmc18_updated.tech*
    d. Click 'OK'
    e. Clicking the correct technology file is very important for successful operation of IPRAIL. Make sure you've chosen the right technology file.
7. Load the layout CIF file
    a. Click on 'Load' in the menu bar
    b. Click on 'Layout(CIF)'
    c. Select the layout file *Cascode/*.cif* or *TwoStage/*.cif*
    d. Click 'OK'

With this, you should be able to view the layout in 0.25um in the IPRAIL GUI.

### 4.2 Resizing the Layout

This sub-section steps through the most important functions in IPRAIL. The symmetry information in the layout is extracted in the symmetry detection phase. A symbolic template is created internally from which the new layout will be generated. The new device sizes obtained from simulation in 0.18um technology are loaded into IPRAIL. The resizing process generates the new layout from the symbolic template and the new device sizes.

1. Load the symmetry information
    a. Click on 'Symmetry' on the menu bar
    b. Click on Textual, for textual file input
    c. Select the symmetry file *Cascode/*.sym* or *TwoStage/*.sym*
    d. Click OK
2. Load the device size information
    a. Click on 'Resize' on the menu bar
    b. Click on 'Trans_Size_File'

    c. Select the device size file *Cascode/*.size* or *TwoStage/*.size*
    d. Click OK
3. In order to resize the layout according the new sizes defined in the file
    a. Click on 'Resize' on the menu bar
    b. Click on 'Resize'
4. Exit IPRAIL by clicking the EXIT menu.

At the end of this process, IPRAIL generates the new layout in TSMC 0.18um technology. The layout is exported as a CIF file *sq_out.cif*.

### 4.3 Checking the Retargeted Layout in IPRAIL

The new layout can be loaded into IPRAIL for viewing.

1. Invoke IPRAIL by typing *iprail* in the Unix command line.
2. Load the IPRAIL's technology file. (*.tch)
    a. Click on 'Load' in the menu bar
    b. Click on 'Technology'
    c. Select the technology file *Technology/MCNC.tch*
    d. Click 'OK'
3. Load the bridge file for the loaded technology
    a. Click on 'Load' in the menu bar
    b. Click on 'BridgeFile'
    c. Select the Bridge File *BridgeFile/bridgeFileTSMC*
    d. Click 'OK'
4. Load the initial design rule
    a. Click on 'Load' in the menu bar
    b. Click on 'InitialDesignRule'
    c. Select the design rule file *DesignRule/UWtsmc18_updated.tech*
    d. Click 'OK'
5. Load the layout CIF file
    a. Click on 'Load' in the menu bar
    b. Click on 'Layout(CIF)'
    c. Select the layout file *sq_out.cif*
    d. Click 'OK'

With this, you should be able to view the generated layout in IPRAIL. The layout is then passed through Design Rule Checking (DRC) in the Cadence Environment.

### 4.4 Design Rule Checking in Cadence

The CIF file is imported into the Cadence environment and checked for DRC by the following steps. If the cadence paths are different from the ones in *install.csh*, you will need to update the *install.csh*, *cds.lib* and *display.drf*.

1. Go to the directory *demo/cadence* and copy the *sq_out.cif* from *demo/bin*.
2. Execute Cadence layout editor
    *a.* Type **icfb** on the command line
3. Create a new library if it has not been created.
    b. On the Library Manager menu bar, click on 'File'
    c. Select 'New'
    d. Select 'Library…' and a new window will pop up
    e. Enter the new library name (e.g. *mylibrary*)
    f. Select 'Attach to existing tech library' radio button, and menu box will appear
    g. Select the appropriate technology library (e.g. *Uwtsmc18*)
    h. Hit 'OK'
4. To import the CIF file
    i. On the icfb main window, click 'File'
    j. Select 'import'
    k. Choose 'CIF…' and a "CIF In" window will appear
    l. Click on the 'User – Defined Data' button.
    m. Set the correct Layer Map Table (e.g. */usr/nikola/groups/vlsi/pkgs/cadence/.current/cds/local.18/pipo/cifInLayermap*)
    n. Click 'OK'
    o. Fill in the input file with the complete path of the output cif file (i.e. ~/demo/bin/*sq_out.cif*)
    p. Fill in the Library Name field with the library name that you created
    q. Set the Scale UU/DBU to 0.0100000 micron
    r. Hit 'OK'
5. To open the imported layout
    s. On the Library Manager window, look at the *library* column and double click on the corresponding library name (*e.g. mylibrary*)
    t. In the "Cell" column, single click on '__out__'. This the default name of the resized layout
    u. In the "View" column, double click on *layout*, and the layout will be shown in a new window
6. To run Design Rule Check (DRC)
    v. On the Virtuoso window's menu bar, click 'Verify'
    w. Select 'DRC…'
    x. Make sure the Rules file is correctly setup (e.g. *divaDRC.rul*)
    y. Hit 'OK'
    z. Go to the icfb main window to check the DRC report and errors

After the DRC checking is complete, you should see the following message on the icfb main window

*********  *Summary of rule violation for cell "__out__ layout"*  *********
*Total errors found: 0*

### 4.5 Post-Extraction Simulation

The simulation setup is stored inside the directories *demo/sim/cascode_opamp* and *demo/sim/2stage_opamp*. Each has two subdirectories inside *original_25* and *target_18*. These contain the extracted netlists of the original design (0.25um) and the target generated layout (0.18um). We will run HSPICE simulations to verify that the specifications have been met for either design in both 0.25um and 0.18um technologies. We shall specify the steps for the Cascode opamp. The TwoStage opamp can be simulated similarly.

1. Go to the directory *demo/sim/cascode_opamp/original_25*
2. From the command line execute *hspice cascode_25.sp*
3. Open the GUI by executing *awaves* from the command line.
4. Use the anchor and other measure options to verify the gain, bandwidth, phase_margin and gain_margin of the design.

This verifies the original Cascode layout in TSMC 0.25um technology. Now we shall simulate the layout generated by IPRAIL for TSMC 0.18um technology.

1. Go to the directory *demo/sim/cascode_opamp/target_18*
2. From the command line execute *hspice cascode_18.sp*
3. Open the GUI by executing *awaves* from the command line.
4. Use the anchor and other measure options to verify the gain, bandwidth, phase_margin and gain_margin of the design.

Here is the comparison of the specifications for the Cascode Opamp.

**Table 1. Post-Layout Simulation Results for Cascode Opamp**

| | 0.25um TSMC | 0.18um TSMC |
|---|---|---|
| Gain | 60.9 dB | 60.6 dB |
| Bandwidth | 51.7 MHz | 63.4 MHz |
| Gain Margin | 12 dB | 10.5 dB |
| Phase Margin | 63 deg | 61 deg |

The specifications for the two stage opamp can be verified similarly.

**Table 2. Post-Layout Simulation Results for 2Stage Opamp**

| | 0.25um TSMC | 0.18um TSMC |
|---|---|---|
| Gain | 57.6 dB | 64.3 dB |
| Bandwidth | 135 MHz | 103 MHz |
| Gain Margin | 9.5 dB | 9.1dB |
| Phase Margin | 50 deg | 56 deg |

This concludes the IPRAIL tutorial!