Tutorial PnR: Placement and Routing for a Schematic

Created by Zeyong Shan and the AMSaC lab group, Spring 2008

Document Contents

Introduction

This document will provide students with the methodology for performing ***place and route*** with the Cadence Encounter tool. Students will learn to use Cadence Encounter with a standard cell library called *OSU_stdcells_ami05* to perform place and route to create the hardware layout from a schematic.

Environment Setup

The following steps will configure your directory as required by the tutorial:

- Log into your account with PuTTY and navigate to your class directory.
  cd /egr/courses/personal/ece410/<username>/cadence
- Copy the map file from the resources directory to your directory. The map file provides the mapping between the place and route tool's metal layers and polygon layers used in Cadence.
  cp /egr/courses/personal/ece410/resources/pnr/gds2_icfb.map gds2_icfb.map
- Copy the .il file from the resources directory to your directory.
  cp /egr/courses/personal/ece410/resources/pnr/ChangeAbstractView.il ChangeAbstractView.il

  This file will be used later in the tutorial and is described in the appendix.
- Return to the root of your user directory. Copy the compressed standard cell library.
  cd /egr/courses/personal/ece410/<username>
  cp /egr/courses/personal/ece410/resources/pnr/ami05.tar.gz ami05.tar.gz
- Decompress the OSU_stdcell_ami05 library with the following command.
  tar –xzvf ami05.tar.gz
- Navigate to your cadence directory again and launch Cadence tools.
  cd /egr/courses/personal/ece410/<username>/cadence
  source $SOFT/cadence
  icfb &
- To map the OSU_stdcells_ami05 library into Cadence, go to the ***Library Manager*** and select **File >> New >> Library**. In the *Name* text box, type **OSU_stdcells_ami05.** Under *Path*, enter **/egr/courses/personal/ece410/<username>/ami05.** In the *Technology*

Library section, select the **Attach to existing tech library** option. Select **AMI 0.6u C5N (3M, 2P, high-res)** from the drop down menu.

- Click **OK**. You should now be able to select the OSU_stdcells_ami05 library in the *Library* pane and open some cell schematics and layouts. If you cannot, ask the TA for assistance.
- The OSU_stdcells_ami05 library is a dependency of the NCSU_TechLib_ami06 library. If you do not have this library mapped to the Front-to-Back Design environment already (i.e., it does not show up in the *Library* pane), then repeat the process above to add it. The library can be found at the path **/egr/courses/personal/ece410/resources**.

This completes the environment setup.

Creating a Verilog Netlist for a Schematic

The Verilog netlist is necessary for automated layout (place and route) tools. It contains information about the I/O pins and the connectivity of the entire schematic. Here we take the XOR gate as an example and create its netlist. The cell name of this gate is called XOR.

STEP 1: Creating the schematic for XOR gate

- First create a new library called **PnR**. Make sure to attach it to the **AMI 0.60u C5N** technology library.

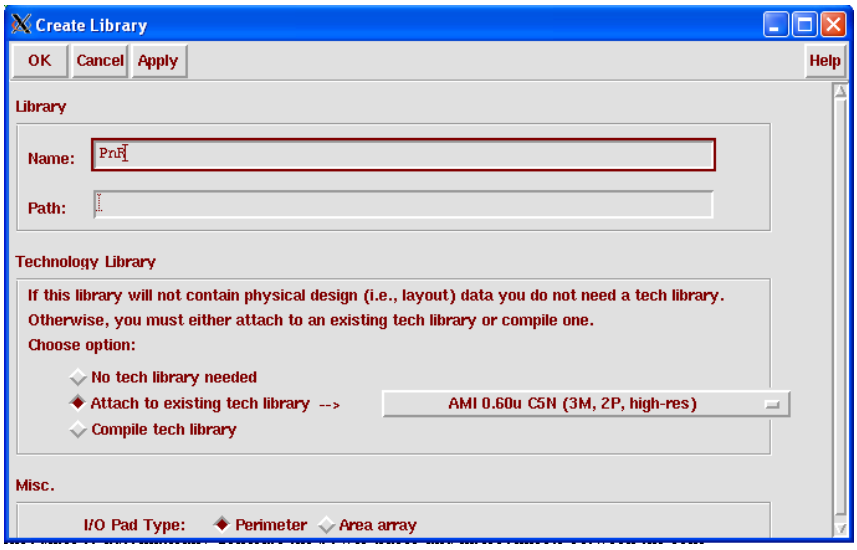


Fig. 1: Creating the PnR library.

- Create a new schematic named **XOR** under the **PnR** library.
- Instantiate the symbols for **INVX1**, **NAND2X1**, and **NOR2X1** from the **OSU_stdcells_ami05** library into the new/blank schematic.
- Build the schematic for an XOR gate shown in Fig. 2 below.

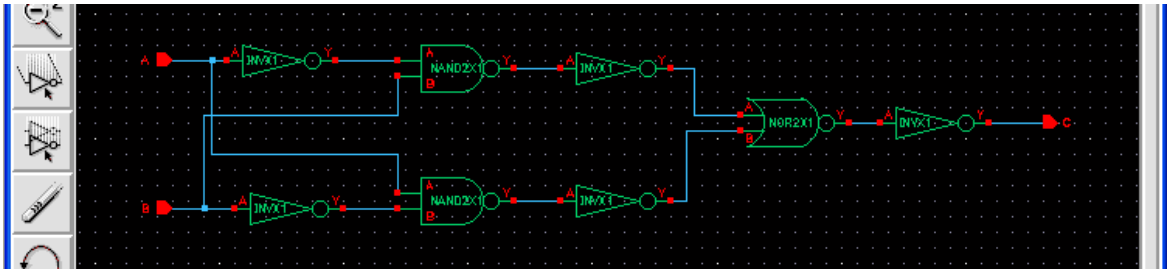


Fig. 2: Schematic for an XOR gate.

STEP 2: Initializing Verilog Integration

- In the "Virtuoso Schematic Editing Window", select **Tools >> Simulation >> Verilog-XL**.
- In the "Setup Environment" window that opens (Fig. 3), enter **XOR.run1** in the **Run Directory** text box. All other default values are correct. Click ***OK***. to create the XOR.run1 directory and open the "Virtuoso Schematic Composer Analysis Environment for Verilog-XL Integration" window (Fig. 4).

STEP 3: Setting the Netlist Options

- In the window shown in Fig 4, set the netlisting options by selecting **Setup >> Netlist**. The *Verilog Netlisting Options* form opens as shown in Fig 5.
- In the …*Options* window, keep the default settings (shown in Fig 5). Click **More>>** to access additional options shown in Fig. 6.
- In the Fig. 6 options window,
  - set **Global Power Nets** to *vdd*
  - set **Global Ground Nets** to *gnd*
  - select **Generate Pin Map**
- The **Drop Port Range** and **Preserve Buses** options are selected by default and can be left selected for this tutorial.
- Click ***OK*** to finish setting the Verilog Netlisting Options.

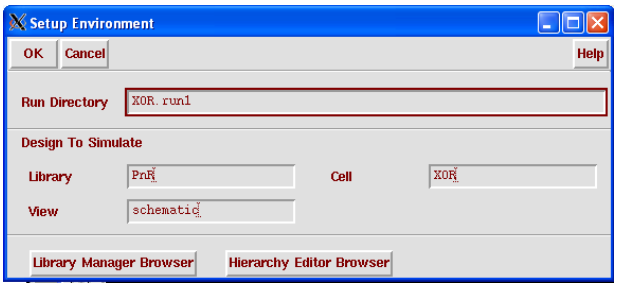


Fig. 3: Setup Environment

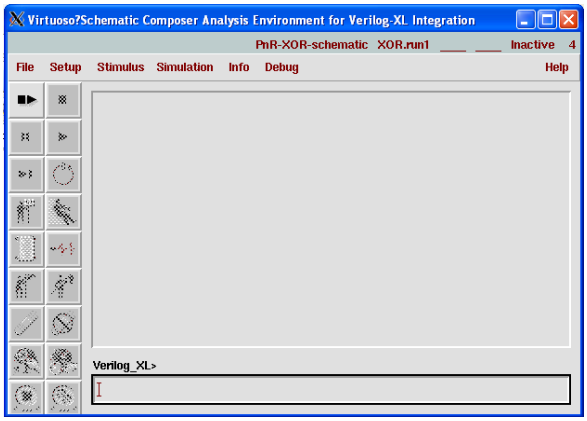


Fig 4: Environment for Verilog-XL Integration

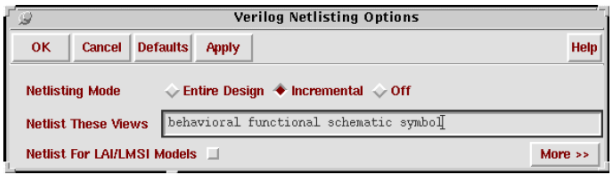


Fig 5: Verilog Netlisting Options

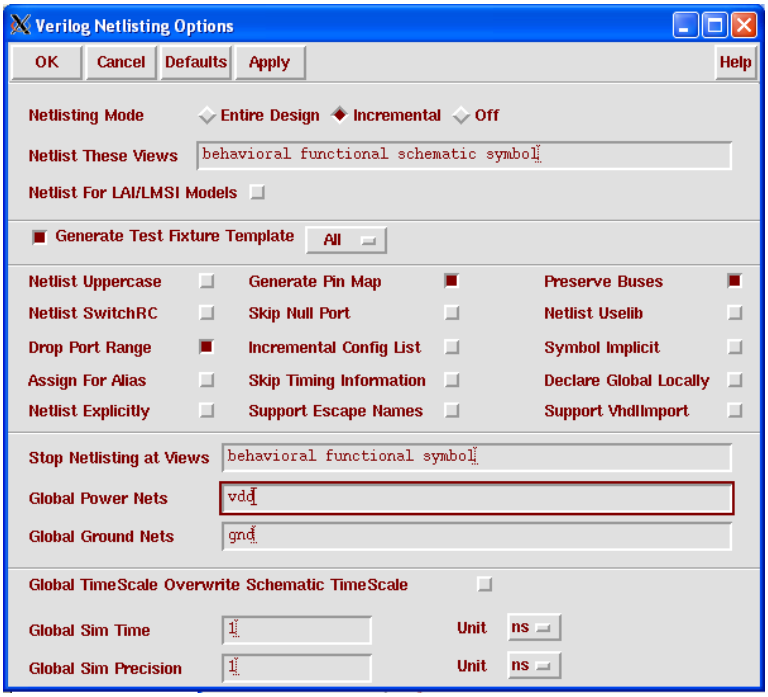


Fig 6: Additional Verilog Netlisting Options

STEP 4: Creating the Stimulus File

- In the "Virtuoso Schematic Composer Analysis Environment for Verilog-XL Integration" window, select **Stimulus >> Verilog.**
- A dialog box appears as shown in Fig. 7. Click ***Yes*** in the dialog box.
- In the "Stimulus Options" window that opens, select **textfixture.Verilog** as the **File Name** and then click ***OK***. The netlist will be created for the schematic.

STEP 5: Creating the Verilog Files

To make Verilog files from the netlists:

- In the "Virtuoso Schematic Composer Analysis Environment for Verilog-XL Integration" window, select **File >> View Netlist Result >> Netlist Run Files ….** The "View Netlist Run Files" window (Fig. 9) appears.
- Click on the **Library Name** appearing in the left box (e.g., PnR). The cells in that directory should appear in the **Cell Name** box.
- Click on the cell name (e.g., XOR) and then click **View**. to open the netlist (Fig. 11).

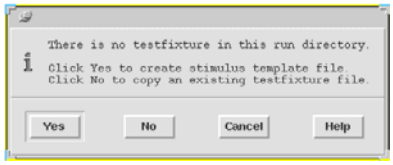


Fig 7: Dialog Box

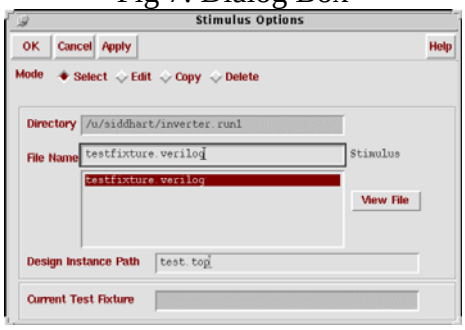


Fig 8: Simulation Options

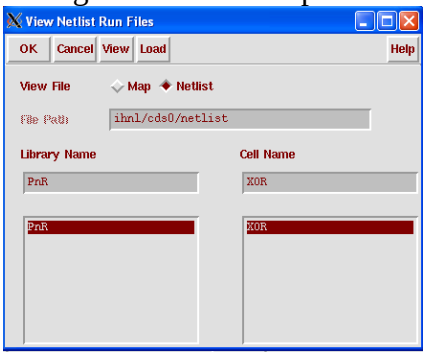


Fig 9: View Netlist Run Files

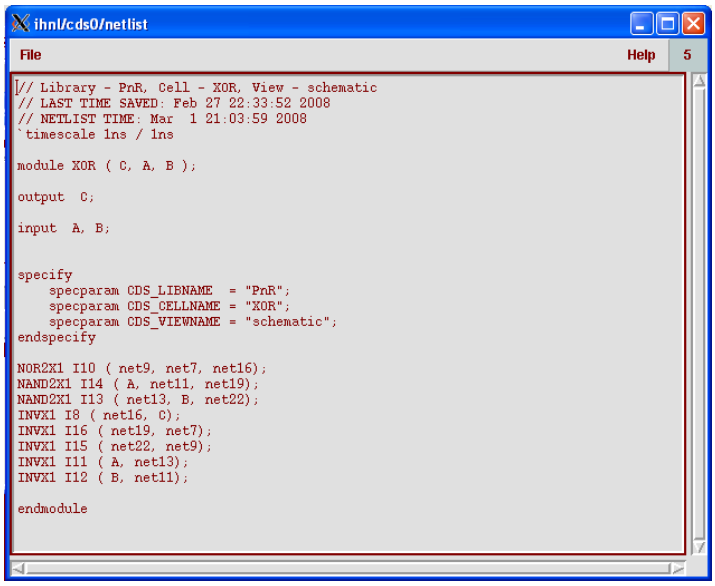


Fig 11: Netlist for XOR gate

- From the netlist window, save the netlist as a Verilog file by selecting **File >> Save As…**. Save the file with a ***.v*** extension (e.g., XOR.v) in any location you wish.
- If you have multiple cells, save the netlists for all the cells in a similar way.
- Exit "icfb".

Now the Verilog netlist for your design has been created. Next, we will perform placement and routing using Cadence Encounter.

Place and Route with Cadence Encounter

Cadence Encounter can be used to convert a Verilog netlist file into a layout. The appropriate sub-cell layouts in the *OSU_stdcells_ami05* standard cell library will be placed into a new cell layout, and then you can provide the commands for Encounter to automatically generate the power traces and interconnects between the cells necessary to implement the gate.

Place-and-route is a powerful tool to enable rapid creation of complex circuit layouts. Like all circuit automation technologies, a place-and-route utility has its limitations and cannot generate layouts that are as efficient, compact, or low-power as a human designer. As the complexity of modern integrated circuits continues to rise, however, automated tools are necessary to keep development time reasonable. Such tools enable designers to operate primarily at the schematic and architectural levels. Manual layout is primarily performed only on low-level cells that are instantiated thousands of times in a large design, thus intense manual optimization of these cells generates a significant payoff in performance and area savings.

STEP 1: Launch Cadence Encounter

- Load the Cadence support files and start Encounter.
  source $SOFT/cadence
  encounter &

- Encounter has the tendency to suspend itself when first launched. If this has occurred, you will get a message in the SSH window similar to the following:

  [1] + Suspended (tty output) encounter

  If Encounter is suspended, you will be unable to enter any commands in Encounter. To unsuspend Encounter, use the UNIX **fg** command, which moves the job with the number following *fg* to the foreground. The job number of encounter can be found in SSH message mentioned above or by entering the **jobs** command. For example, if Encounter is job number 1, use the following command to unsuspend Encounter: **fg 1**

STEP 2: Import the netlist file into Encounter

- Select **Design >> Import Design ….**
- Click the **Advanced** tab of the "Design Import" window (Fig. 12). Select **Power** and set **Power Nets** to *vdd* and **Ground Nets** to *gnd*.
- Click the **Basic** tab. In the section entitled **Verilog Netlist**, click the ellipsis button next to the **File** text box to specify the location of the source files of the XOR design.
- In the dialog box that pops up, click the folder icon next to the **Add** button to browse to the Verilog file **XOR.v**. Select this file, then press the **Add** button, and select the filepath in the left hand pane as shown in Fig 13. Then click the **Close** button. The filepath of the Verilog file of XOR gate should now appear in the **Files** text box in the Design Import window.

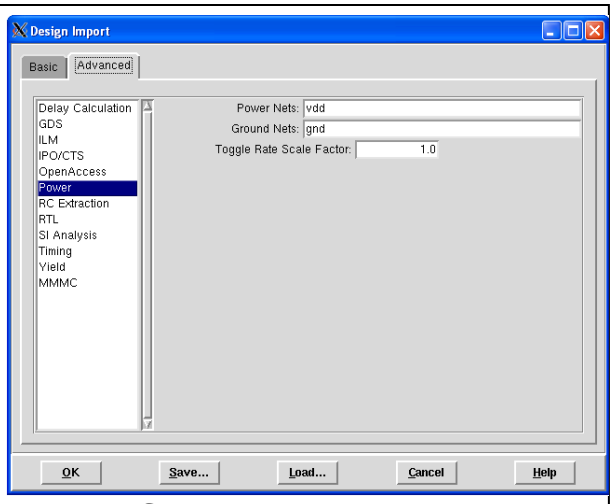


Fig 12: Set power nets in Design Import

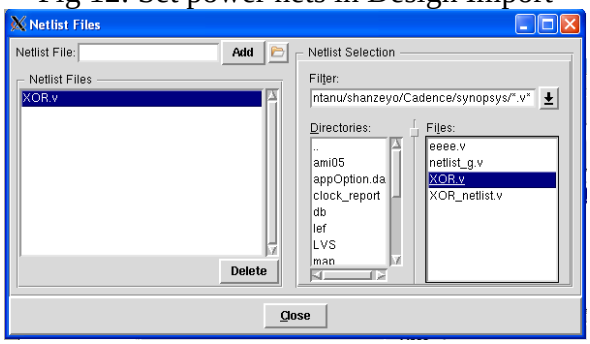


Fig 13: Select the Verilog netlist file

- Select the **Auto Assign** option for the **Top Cell** specification under **Verilog Netlist**. You have a single module for this design and thus do not to need to manually indicate which module is the top-level module, which specifies the connections between the instantiated component modules.
- In the **Timing Libraries** section, click the ellipsis button next to the **Common Timing Libraries** text box. Click the folder button and browse and select the file **osu05_stdcells.tlf**. This file can be found at **/egr/courses/personal/ece410/<username>/ami05/lib**. Select this file, press the **Add** button, select the filepath in the left-hand pane, and then press the **Close** button.

The .tlf file contains information on the timing and power parameters of the cell library. It is used to determine delays of I/O ports and interconnects of the final design.

- Click the ellipsis button next to the **LEF Files** text box. Following the same process as above, browse to and select the file **osu05_stdcells.lef**. This file can be found at **/egr/courses/personal/ece410/<username>/ami05/lib**.

  The .lef file contains information about the layer specifications for the technology implemented by the library. Such specifications include default sizes, minimum widths, and minimum spacing dimensions.

- Check to see that your ***Design Import*** dialog box is similar to the image of Fig 14 (filepaths may be different), and then press **OK**.

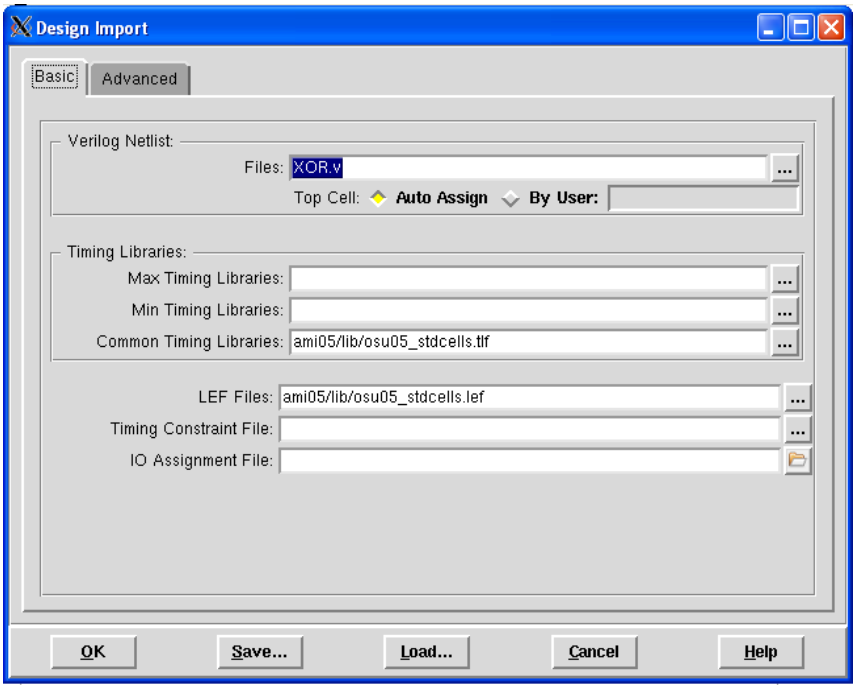


Fig 14: Design Import dialog box with necessary files entered

- If you receive an error as shown in Fig 15, just click **OK** to close the error window. Click OK again in ***Design Import*** dialog box.

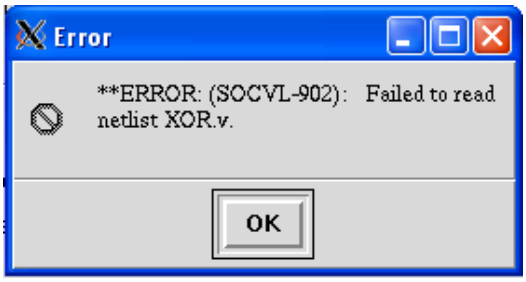


Fig 15: Error message

- The Encounter window for your cell will open as shown in Fig 16.

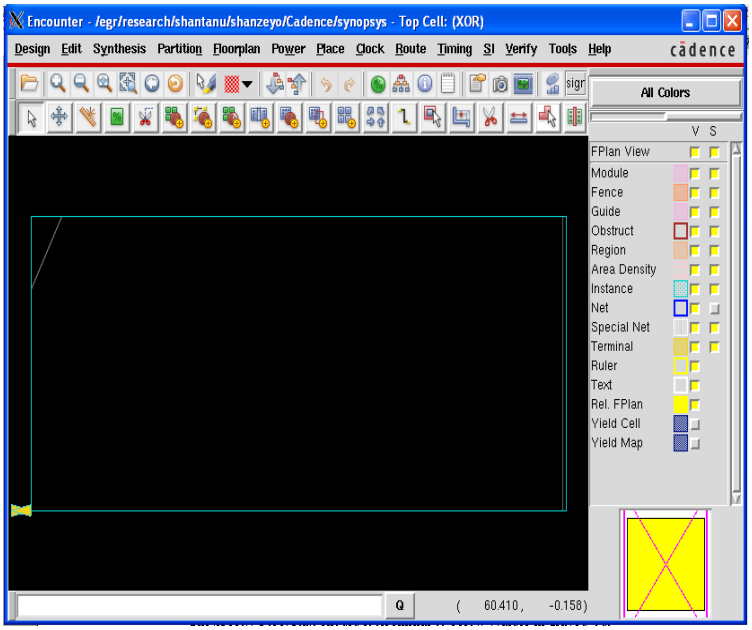


Fig 16: Encounter window for XOR gate

STEP 3: Lay down the chip infrastructure

In this step, we will specify the silicon dimensions on which our circuit will be laid out and have Encounter lay down the power rails used by the sub-cells of the circuit being routed.

- Select **Floorplan** >> **Specify Floorplan**. The **Specify Floorplan** dialog box will open.
- In the **Core Margins by** section, change all **Core to** dimensions to 12. This will create a 12 micron buffer on all sides of the XOR module for I/O ports. Set the remaining options to match Fig. 17 (values in grayed-out textboxes are ignored and may be different). Then press **OK**. A floorplan core boundary will be added to the main Encounter window.

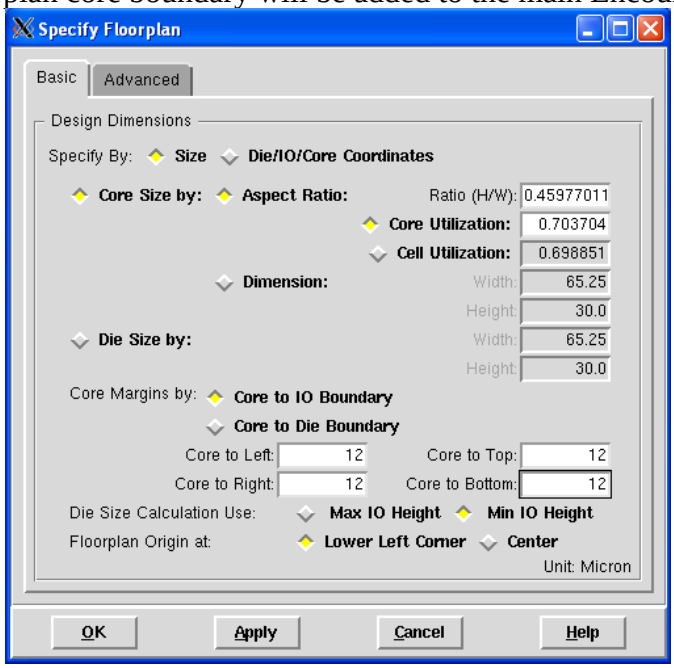


Fig 17: Specify Floorplan dialog box with proper settings

- Add the main power rails with the menu command **Power >> Power Planning >> Add Rings.**
- In the **Add Rings** dialog that pops up (Fig. 19), make sure the net names **gnd** and **vdd** are in the **Nets(s)** text box.
- In the **Ring Configuration** section, select **Center in channel**. Review your settings (they should match Fig. 19). Then click **OK**. In the Encounter window, you will see power rings created around the periphery of your design.

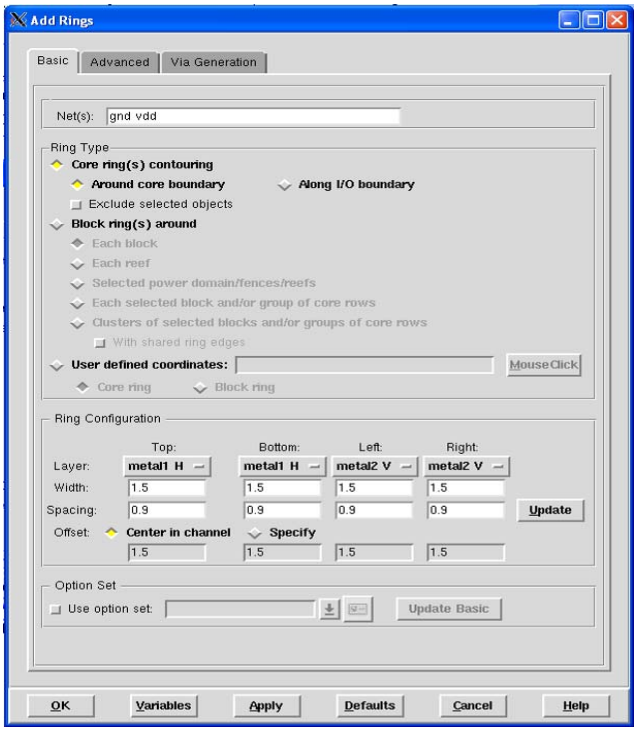


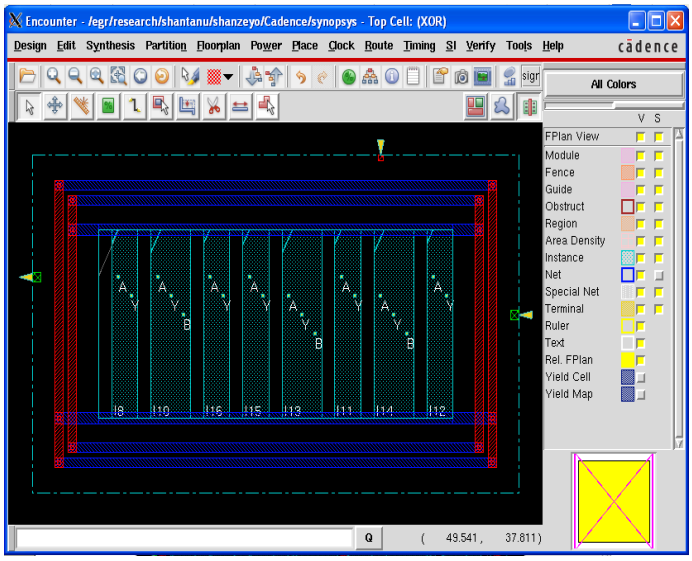
Fig 19: Add power rings dialog settings

- Connect the cell row power rails to the created power rings with the ***Encounter*** menu command **Route >> Special Route.** Press **OK** to choose the default setting.

STEP 4: Create gate cell layout

In this step, the gate-level implementation specified by the Verilog file will be laid out using cells in the OSU standard cell library. Autorouting will then be performed in which Cadence will route the traces to connect the cells to the power rails and form the cell interconnections.

- Place the layouts for the gate cells with the ***Encounter*** menu command **Place >> Standard Cells and Blocks**. Uncheck the **Include Pre-Place Optimization** option, and press **OK**. Encounter will go through several iterations of placement, depending on the strictness of the area, power, and geometry demands placed on the design. Encounter may require several minutes to perform this step, especially for more elaborate designs.
- To view the cell layout just created, switch to **Physical View** by pressing the  button in the upper right corner of the ***Encounter*** window. Your encounter window will look similar to the image as shown in Fig 20.



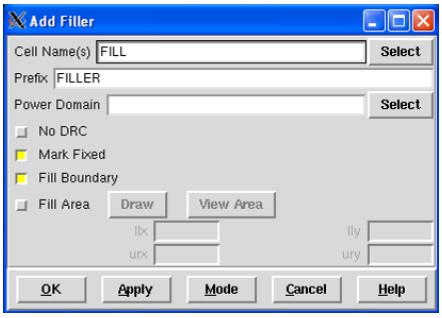


Fig 20: Physical view of XOR layout with standard cells instantiated. Fig 21: Add Filler dialog

- Finish the module routing with the ***Encounter*** menu command **Route >> NanoRoute >> Route**. Accept the default settings by clicking **OK**.
- It is necessary to fill up any empty space in the chip core not occupied with cells or routing. Select **Place >> Filler >> Add Filler**. In the **Add Filler** dialog box, press the **SELECT** button. Select **FILL** in the right-hand pane, press the **ADD** button. Then **Close** and click **OK** to add the filler.
- The final layout for XOR gate is shown in Fig 22.

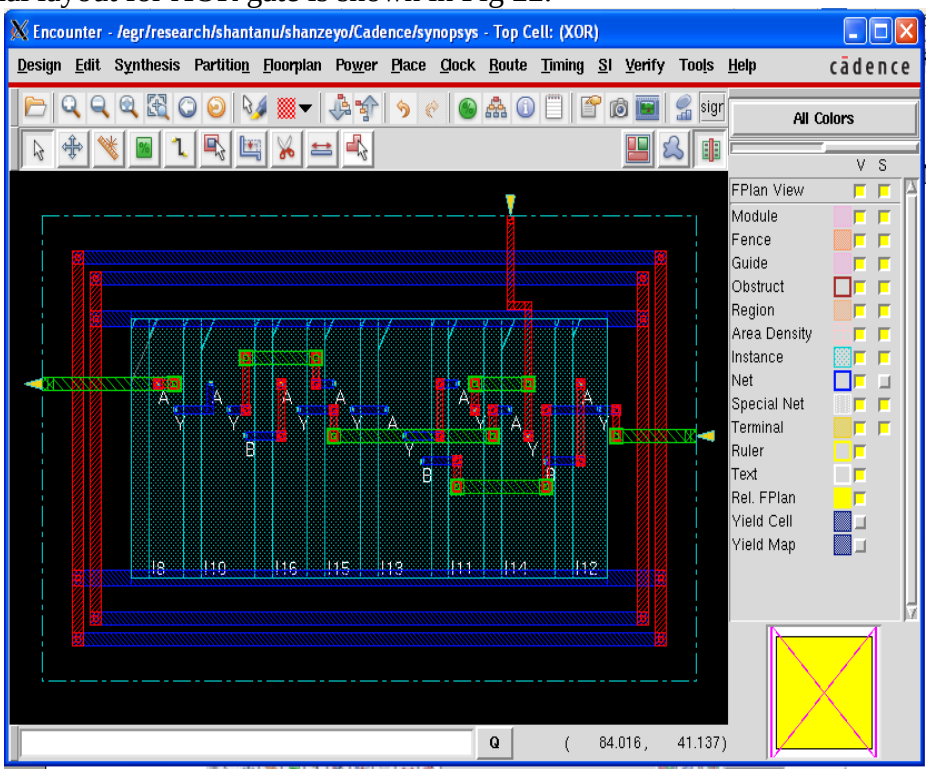


Fig 22: XOR cell with cell interconnects routed and filler added

STEP 5: Check and save the design

- Verify that the layout does not violate any technology design rules. Select **Verify** >> **Verify Geometry.** Select the default settings by clicking **OK.** Encounter will output an error report to the SSH window and highlight any potential geometry errors in white in the main Encounter window.
- Analyze the design for potential flaws with interconnects and pins with the menu commands **Verify** >> **Verify Connectivity**. Select the default settings by clicking **OK**. Encounter outputs error information to the SSH window.
- To save the design, select **Design** >> **Save Design As** >> **SoCE**. Name the file **XOR.enc** and press **Save**.
- Select **Design** >> **Save** >> **GDS.** In the *GDS Export* dialog box that comes up, name the file **XOR.gds2** in the **Output Stream File** text box. Under **Map File**, enter the filename

  /egr/courses/personal/ece410/<username>/cadence/gds2_icfb.map.

  Confirm overwriting the file if a warning box pops up.
- In the *GDS Export* dialog box, in the **Library Name** text box, type **OSU_stdcells_ami05** to indicate the standard cell library upon which the XOR layout is based.
- Verify that the *GDS Export* dialog box appears like Fig 23, and then click **OK** to export the gds file.

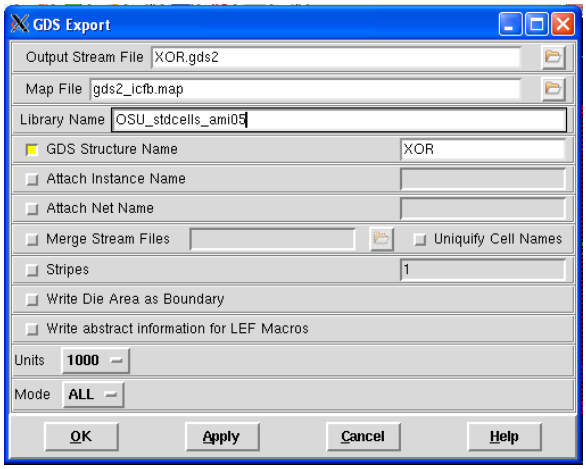


Fig 23: GDS Export dialog

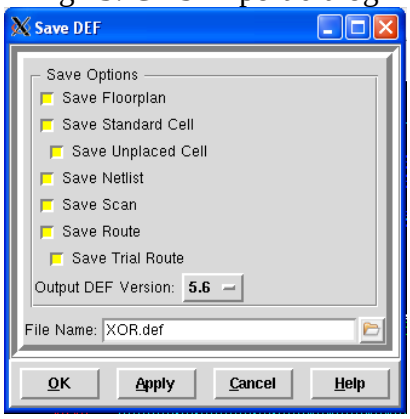


Fig 24: Example save DEF dialog

All of the layout information on cells, fillers, routing, etc. needed to import the layout into Virtuoso can be saved within a DEF file. We will save a DEF file and then import it into Virtuoso. This will enable us to work with both manually routed custom cells and automatically routed cells together.

- Save a DEF file for the design. Select **Design** >> **Save** >> **DEF.** Check all the **Save Options.** Choose **XOR.def** for the **File Name** and press **OK.**
- Now that the design has been exported, we are now finished with Encounter. Exit encounter with the command **Design** >> **Exit.** Press **OK** to confirm the exit.

*If you wish to reload a previously-saved placed-and-routed design in Encounter, read in the .enc file created for the design with the menu command **Restore Design** >> **SoCE.***

Encounter uses a format for specifying vias that is not recognized by Virtuoso. The DEF file must be edited to correct this problem if we are to import the design into Virtuoso. Go to the SSH window and load the DEF file in a text editor such as ***emacs***, ***vi***, ***pico***, etc. If you prefer a GUI based text editor, you can load the file in ***nedit*** by navigating to the directory where you saved the DEF file and entering the command: **nedit XOR.def**

- Replace all instances of "viagen21_1" with "M2_M1" except the first instance.
- Replace all instances of "viagen21_3" with "M2_M1" except the first instance. The instances that should remain unchanged are circled red in the figure below.
- Save the file and exit the text editor.

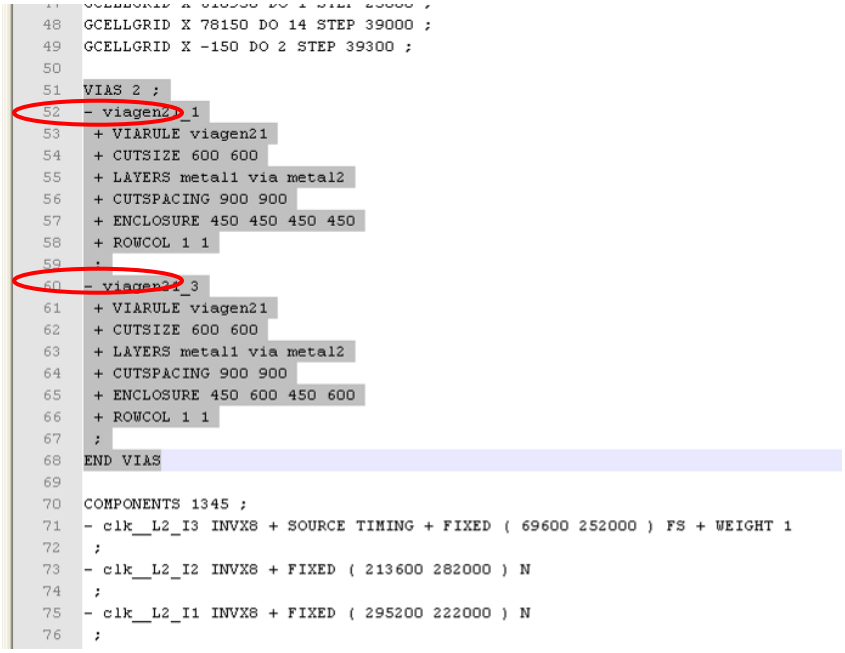


Fig 25: Viagen statements in DEF file to keep

Importing the GDSII file into a Virtuoso Library

In the final section in this tutorial, we will import the layout created by Encounter into Virtuoso. If you are not there already, proceed to the cadence directory in your user 410 account with the command **cd /egr/courses/personal/ece410/<username>/cadence**. The Cadence support files should already be loaded. If not, source them.

- Launch the Cadence Front-to-Back design environment with the command **icfb &**
- In the ***Command Interpreter Window (CIW)*** select **File >> Import >> Stream**
- In the **Virtuoso Stream In** Dialog that comes up, press the **Options** button. Check the **Snap XY to Grid Resolution** option. Click **OK**.
- Click the **User-Defined Data** button. In the **Layer Map Table** textbox, type in the pathname of the map file you copied to your directory: **/egr/courses/personal/ece410/<username>/cadence/gds2_icfb.map**.

  Then click **OK**.

- Back in the **Virtuoso Stream In** dialog, type in the filepath of the gds2 file you output in Encounter into the **Input File** text box. If you followed the tutorial exactly, the filepath should be **/egr/courses/personal/ece410/<username>/cadence/XOR.gds2**
- For **Library Name**, enter **OSU_stdcells_ami05**. Click **OK**. You will likely receive a warning. This warning can be safely dismissed. Click **OK**.

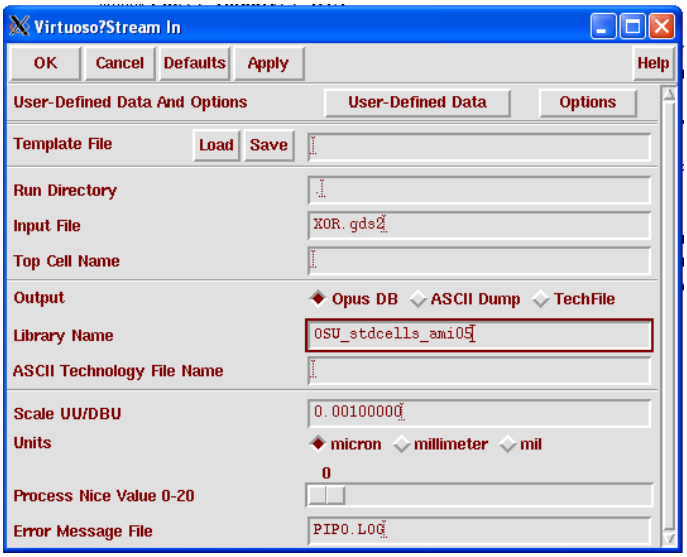


Fig 26: Virtuoso Stream In Dialog

- We now need to import the DEF file. In the ***Command Interpreter Window***, select **File >> Import >> DEF**.
- In the **Read DEF File into CellView** dialog box that comes up, click **Browse**. Select the XOR's layout under **OSU_stdcells_ami05 >> XOR >> layout**.
- In the **DEF File Name** text box, type in the pathname for the DEF file you created with Encounter. If you followed the tutorial exactly, the filepath should be **/egr/courses/personal/<username>/cadence/XOR.def.** Click **OK** to continue the operation.

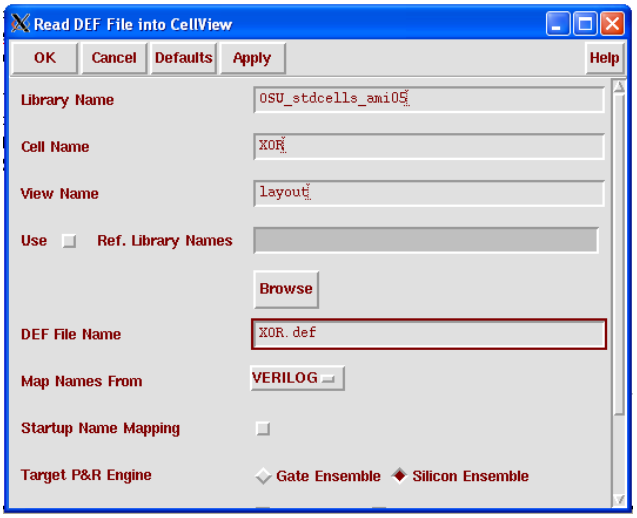


Fig 27: Import DEF file

- You should now be able to view in Virtuoso the layout created in Encounter. Open the layout view by selecting the **XOR** cell in the **OSU_stdcells_ami05** library in the ***Library Manager***. If you cannot see the **XOR** cell, select **View >> Refresh** to refresh the window.
- If you have missing cell layout views, go to the appendix for information on how to change abstract views in to layout views.

You have now completed this tutorial. The layout you have generated can now be used to generate an extracted file on which you can perform LVS and run simulations in the manner described in Tutorials A and B for this course. In this tutorial, we went through the process of converting a digital circuit design in from schematic to physical layout and metal polygons. Design tools such as Encounter can enable you to rapidly generate layouts from schematics. Place and Route tools generate cell masks for chip fabrication in a top-down fashion, opposite to the bottom-up approach used in the ECE 410 labs, in which you develop layouts for core logic cells such as gates, multiplexers, and flip-flops, and instantiate and route those together to form more complicated structures that implement desired functionality.

Appendix: ChangeAbstractView

After importing stream and def files into Cadence to obtain the layout of the design, there may be several abstracts of standard cells in the layout. The *ChangeAbstractView.il* file (copied early in this tutorial) can used to change all the abstract cells to layout cells. To use this program, follow these steps:

- In the *icfb* window command line, type: **load("ChangeAbstractView.il")**.
- Open the layout you want to change.
- In the *icfb* window command line, type:

ChangeAbstractView("OSU_stdcells_ami05" "CELLNAME")

where, "CELLNAME" is the cell name (e.g., XOR) that you want to change the layout for.

The function should list all the instances it is changing.