

# EE456 Lab Tutorial 1

## Cadence Virtuoso Schematic Composer Introduction

### 1.0 Introduction

The purpose of the first lab tutorial is to help you become familiar with the schematic editor, *Virtuoso Schematic Composer*. You will create a schematic and a symbol for a static CMOS inverter.

After completion of this tutorial, you should be able to:

- Insert instances into your design
- Connect instances together using wires
- Change instance properties
- Name nets
- Add pins to your design
- Create and edit a symbol cellview
- Check and save your design

### 2.0 Getting Started

- Log into your workstation. You should already have access to the workstations using your ECN login and password. If you have problems logging in, please consult your TA.
- Copy the files `cds.lib`, `.cdsinit`, and `.cdsenv` from the `/package/cae/cadence/cells/tsmc025` directory into your home directory.
- These files contain the path of the library files and the environment settings.
- Add the line `source /package/cae/setup/ee456.csh` into your `.cshrc` (or `.tcshrc`) file. If you do not have a `.cshrc` file, create one using your favorite text editor. Then either enter `source .cshrc` in any UNIX window or logout and log in again. Note that you only need to do this the first time you start using Cadence.

### 3.0 Online Documentation

Note that this tutorial and the following series cover only the very fundamental concepts of creating CMOS schematics, symbols and layouts, simulating circuits, performing layout verification and parasitic extraction from layout using Cadence. Please refer to the online documentation should you require additional information.

- To access the online documentation, type **cdsdoc** in any UNIX window or click on **help** in the upper right corner of any Cadence window.
- In the online documentation, more detailed information can be found under the *Composer* category. Under *Design Entry*, there are the *Virtuoso Schematic Composer Tutorial* and the *Virtuoso Schematic Composer User Guide* that you may find helpful.

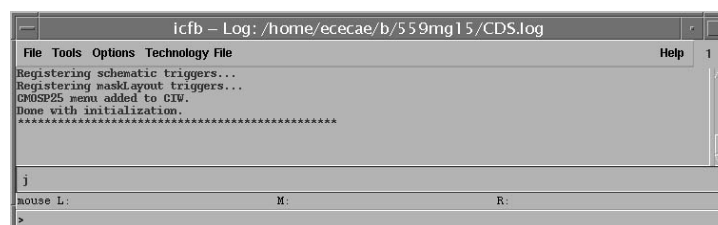
### 4.0 Virtuoso Schematic Composer Basics

The Virtuoso Schematic Composer is used to create the schematic of your design. In the schematic, it will contain devices (transistors) connected together with nets (wire connections).

#### 4.1 Launch Virtuoso Schematic Composer

- Enter **icfb** in a terminal window command prompt. In a minute or so, a window should appear with the title *icms* at the top. This window is known as the *Command Interpreter Window (CIW)*. It is the main control window for the schematic composer software. Various properties can also be changed in this window. Other commands that can be used to launch the CIW, depending on the tools that you require, include:

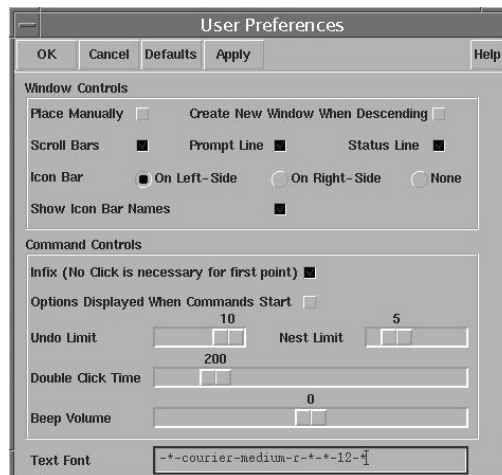
icde – includes the basic IC design entry package  
 icds – includes design and simulation tools  
 icms – a larger set of tools for mixed signal design  
 msfb – supports an IC design flow without place and route  
 icfb – complete front-to-back design set



- The CIW displays a running history of the commands that you execute and their results. It also shows status and error messages from the schematic composer software.
- When commands are run, the CIW will display prompts for action.

## 4.2 Setup User Preferences

- Select **Options -> User Preferences** from the top menu and a window named *User Preferences* appears.
- Check the boxes named *Scroll Bars* and *Infix*. Turning on *Infix* limits the number of mouse clicks required to execute certain commands, as you will learn later in the tutorial.
- Drag the *Undo Limit* slider to 10. Click **OK** to exit.

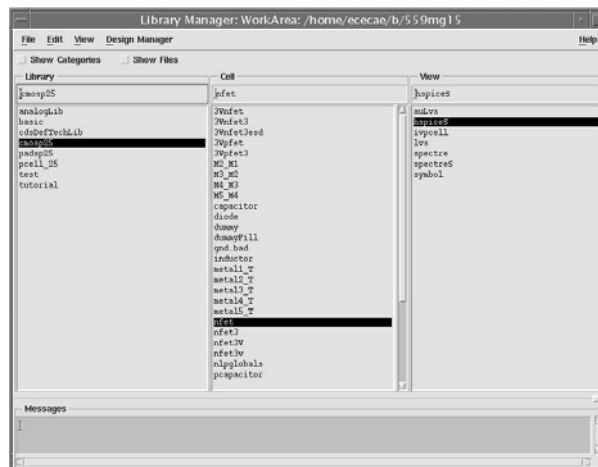


## 4.3 Library Manager

The Library Manager allows you to manage (create, copy, move, delete) libraries. It is recommended that all changes in the libraries be done here to preserve the integrity of all the files associated with your design.

- To run Library Manager, click on **Tools -> Library Manager**.
- Cadence use the term *library* to mean both reference libraries, which contain defined components for a specific technology, and design libraries, in which you create your own designs. The designs are called *cells*.

- Each cell can have multiple representations, such as a symbol or a schematic. These representations are called *cellviews*.
- There should be several libraries already present in the *Library* column. Click on any of the libraries. It will list more cells in the *Cell* column that belongs to this library. You should see a library called *cmosp25*. This will probably be the most often used library for this course, as you will find out later on.
- The cellviews associated with the cell will appear in the *View* column if you click on any of the cells listed.
- The pathnames of these libraries are defined in the *cds.lib* file. The paths can be altered manually with a text editor if required.
- It is also possible to open designs from the library manager. To do this, either right click on the cellview and select *Open* from the menu that appears or simply double click on the cellview name.



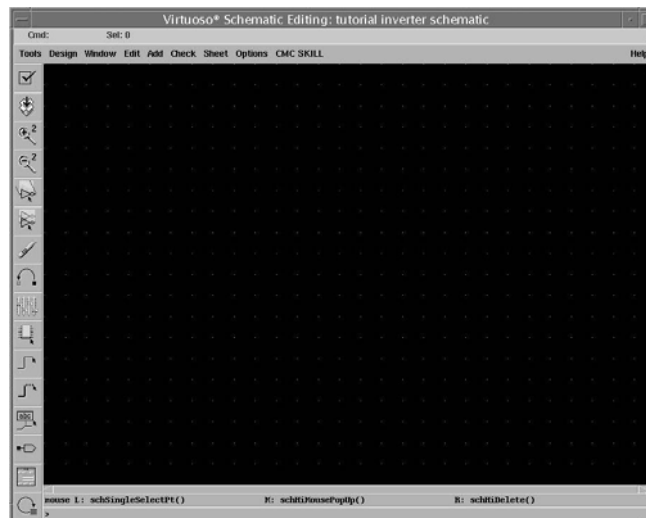
#### 4.4 Creating a New Library

- To create a new library, click on **File -> New -> Library** (you can do this in either the *icfb* window or the Library Manager). A new window appears and type *tutorial* in the *Name* field and click **OK**.
- When another new window (with title *technology file for new library*) appears, click on *Attach to an existing techfile* and click **OK**. The technology file (techfile) stores all the instances, process and rules files required for schematic and layout design.

- A new window should appear asking for a *Technology Library*. Select *cmosp25* and click **OK**.

#### 4.5 Creating a New Cellview

- To create a new cell view, click on **File -> New -> Cellview** (you can do this in either the *icfb* window or the *Library Manager*). A window appears and select *tutorial* for the *Library Name* field, and type in *inverter* for the *Cell Name*. The *View Name* should be *schematic* and the *Tool* field should be *Composer-Schematic*. Click **OK** when done.
- A new window named *Virtuoso Schematic Editing: tutorial inverter schematic* should appear. This is the schematic window or cellview. Note that the last parts of the window name correspond to the library (tutorial) and the cellview (inverter) that you are currently working on.

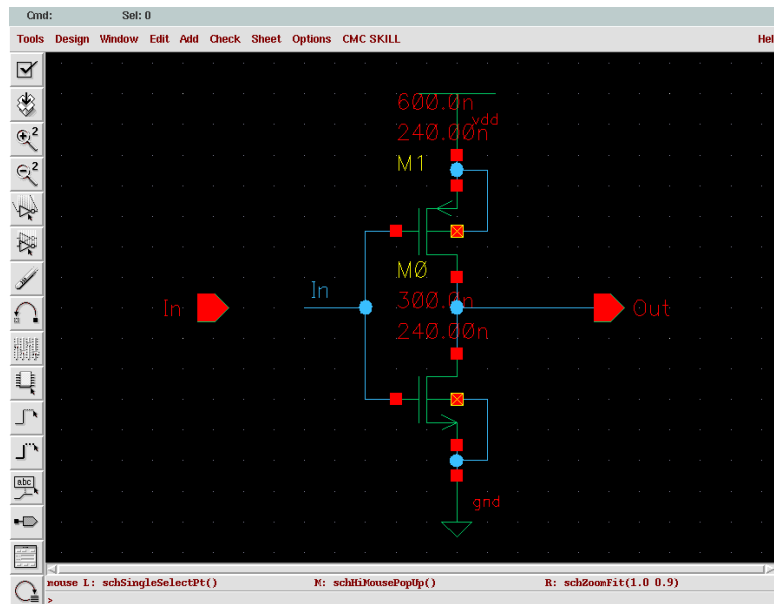


- Take a look at the command menu on top and the icons on the left. Clicking on the top (drop down) menu will reveal more command options. If there is an arrow to the right of the command option in the drop down menu, it means that there are more options under it. Click on the command option to reveal more options.
- Next to certain command options, there are some letters next to it. These are bindkeys (or more commonly known as hot-keys) that invoke the command using simple key presses. They will become handy when you get more familiar with the schematic editor and the bindkeys.

- The icons on the left correspond to several most frequently used commands such as add instance, change instance properties, add wire, zoom in, zoom out, undo, delete etc. By placing the mouse cursor on top of the icon, the name of the icon appears.
- A pop up menu appears when you place the cursor on any empty portion of the schematic and press the middle mouse button.
- The right mouse button repeats the last executed command.

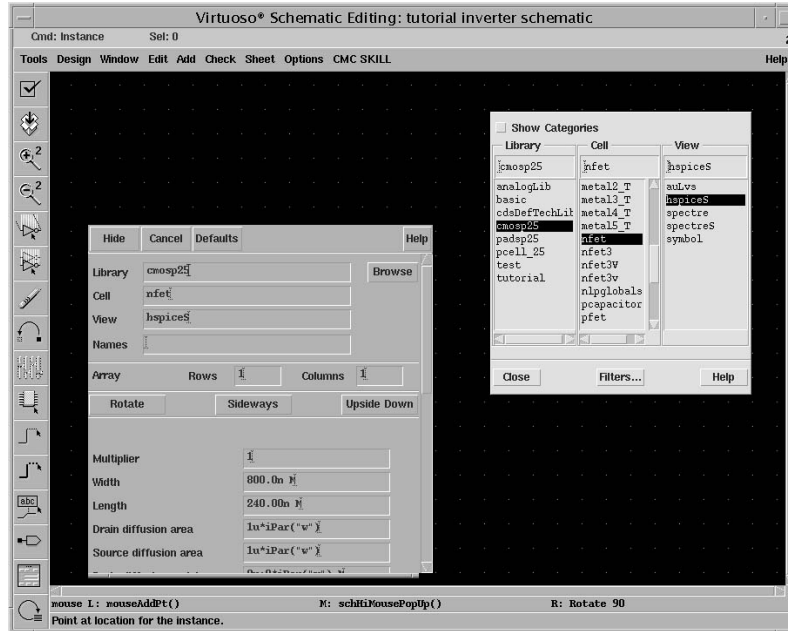
#### 4.6 Drawing a Schematic for a CMOS Inverter

Now you are ready to draw the schematic of a CMOS inverter as shown below in the illustration. From the figure, you can see that the inverter consists of two transistors (one n-type and one p-type), Vdd and Ground. These are known as instances.



#### Adding Instances

- To add a transistor to your schematic, click on the **Instance** icon. A new window named *Add instance* appears. There are several other ways to bring up this window. One way is to press the middle mouse button within the cellview, then select **Add Instance** when the pop-up menu appears. The other way is to select **Add -> Instance** from the top menu. The last and probably the most convenient way is to use hot-key (in this case, press 'i' on the keyboard).
- In this new window, click on the **Browse** button to browse available libraries. Another new window named *Library Browser* appears, very similar to the Library Manager window.



- In the new window, click on the *cmosp25* library, then select the *nfet* cell and choose *hspiceS* for the cellview. Now move the mouse cursor back to the schematic window and you will find a symbol representing an n-type transistor attached to the cursor.
- You can rotate or flip the instance (sideways or upside down) by clicking the **Rotate**, **Sideways** and **Upside Down** buttons in the window before placement.
- If you have accidentally chosen the wrong instance, press the **ESC** key or click the **Cancel** button on the *Add Instance* window.
- Move the cursor to a desired location on the schematic window and click the mouse button to put the transistor in place. If you find out that the windows are blocking your view, simply click and drag on the title bar of the window to move them away and continue with the placement.
- After the placement of the n-type transistor, it will continue to prompt you to add another instance (same instance by default). This will allow you to place multiple instances onto your cellview. Return to the *Library Browser* and under the same library, select *pfet* cell and choose *hspiceS* for the cellview. Place a p-type transistor in the schematic window.
- As an exercise, continue to add the remaining instances (Vdd and Gnd) into your cellview from the *analogLib* library.

- Press the `ESC` key to stop adding more instances.
- You can simply click and drag the instance around to re-position them on your cellview.

### Editing Instance Properties

- To modify the properties of an instance, such as the width and length of the n-type transistor, select the n-type transistor and click the **Property** icon on the left or selecting **Edit -> Properties -> Objects** from the drop down menu (hot-key is 'q').
- A new window named *Edit Object Properties* appears. The *Library Name*, *Cell Name* and *View Name* are displayed near the middle section. Ensure that these values correspond to the right instance before modifying the properties.

Property	Value	Display
Library Name	cmosp25	off
Cell Name	nfet	off
View Name	hspiceS	off
Instance Name	M1	value

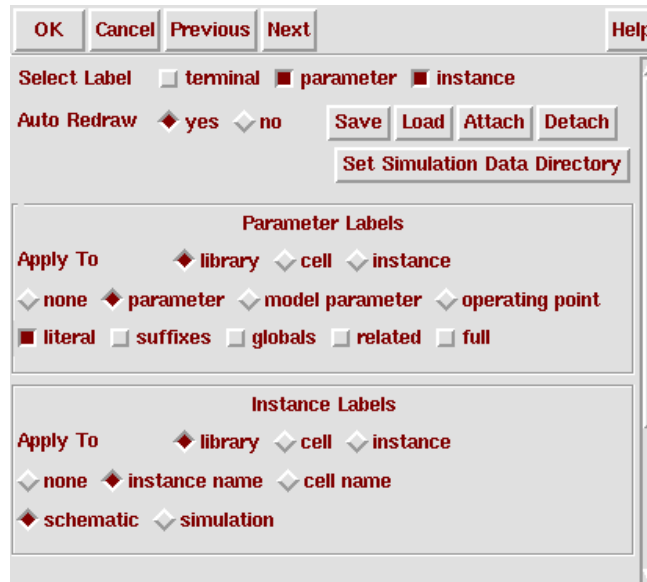
CDF Parameter	Value	Display
Multiplier	1	off
Width	300.0n	value
Length	240.00n	value
Drain diffusion area	1u*1Par("w")	off
Source diffusion area	1u*1Par("w")	off
Drain diffusion periphery	2u+2*1Par("w")	off
Source diffusion periphery	2u+2*1Par("w")	off
Drain diffusion res squares	1u/1Par("w")	off
Source diffusion res squares	1u/1Par("w")	off
Drain diffusion length		off
Source diffusion length		off
Device initially off	<input type="checkbox"/>	off
Drain source initial voltage		off
Gate source initial voltage		off
Bulk source initial voltage		off
Temp rise from ambient		off
Estimated operating region	triode	off
Bulk node connection	S	off
Additional drain resistance		off
Additional source resistance		off
Temperature difference		off
Source/drain selector		off
model name	nch	off



- For the *Instance Name* field, it can be changed to any value for easy identification between instances.
- For the *CDF Parameters* change the *Width* to  $0.3\mu$  and the *Length* to  $0.24\mu$  ( $\mu$  represents micrometers and  $n$  represents nanometers). Use the `Tab` key or mouse to move between fields and do not press the `Enter` key unless you are done.
- Change the *model name* to `nch`. This is the name that corresponds to the model that will be used to simulate your schematic. For a p-type transistor, type in `pch`.
- To change the properties for another instance, click on the instance and the *Edit Object Properties* window will be updated to the new instance.
- As an exercise, change the properties of the p-type transistor such that the width is  $0.6\mu$  and the length is  $0.24\mu$ .
- After done editing, click **OK** or press `Enter` to quit.
- Note that other instance properties can be edited in the same manner.

### Displaying Instance Properties

- It is possible to make the instance properties that you specified above visible in the schematic. This provides a quick view to the property values that you entered for each instance.
- To change the display options, click on **Edit -> Component Display**. A new window will appear.
- In the schematic, click on a component that you want to display values for.
- In the *Component Display* window, check the boxes as indicated in the figure below.



- Note that when you check the *library* box, it will cause all symbols from the same library to display the same way so you don't have to repeat this step for each symbol. Also, checking on the *Instance Name* box under *Instance Labels* will cause the reference name for the symbol to appear.

### Deleting Instances

- To delete an instance, click on the instance to be deleted. A box around the instance indicates that the instance is selected. Click the delete icon on the left side of the schematic window or press the `Delete` key to delete.
- To delete multiple instances, you can first select multiple instances by pressing the `shift` key while selecting the instances with the mouse or by holding the left mouse button down and dragging a box, then execute the delete command as above.
- Another way is to execute the delete command, then select the instances that are to be deleted. Note that the delete command will remain active until you cancel the command by pressing the `ESC` key. This is also true for most other commands.

### Adding Pins

- To add pins (used to connect your current design to external circuits), either click on the **Pin** icon, or select **Add -> Pin** from the drop down menu (hot-key is 'p').

- A new window named *Add Pin* appears. Type *In* for the *Pin Names* field. Select *input* for the *Direction* tab. Leave other settings at their default state.
- Move the mouse cursor back to the schematic window and you will find a symbol representing a pin attached to the cursor, similar to the placing of an instance.
- You can rotate or flip the symbol (sideways or upside down) by clicking the **Rotate**, **Sideways** and **Upside Down** buttons in the window before placement.
- As an exercise, add an output pin named *Out*.

### Adding Wires

- To add wires to connect the instances together, click on the *Wire (narrow)* icon on the left. Alternatively, click the middle mouse button within the cellview and select *Wire (narrow)* in the pop up. You might have noticed that there is a similar icon named *Wire (wide)*. It is used to create buses. More information regarding creating buses will be discussed further in the next tutorial.
- A new window named *Add Wire* appears. Leave the *Draw Mode* as *route* unless you absolutely need to draw non-rectilinear wires.
- Click on the wire starting point (for example, at the red boxes indicating an instance pin). Move the mouse cursor and click again for each wire segment.
- You might notice that as you move the mouse cursor, a small diamond shape appears over the connection object closest to the pointer. To end the wire, press *s* to snap to the nearest object that shows a diamond shape. Another way is to click a schematic pin, an instance pin, or another wire or double-click on a new wire endpoint.
- Finish up the schematic by connecting the instances together with wires.
- You should now have a schematic similar to the one in the diagram.

### Naming Nets

- You might have noticed that in the schematic, the input pin is not connected to the schematic. By naming the net (or wire), it is possible to indicate that there is an electrical connection between other nets or pins. This could help in reducing the amount of wire cluttering that makes the schematic hard to read.
- To name a net, select **Add -> Wire Name** from the drop down menu or select **Add Name** from the pop up menu. A new window named *Add Wire Name*

appears. Type in `In` for the *Names* field and move the mouse cursor to the net that is to be named *In*.

- It is possible to add several names to different nets at one time. To do this, simply type the names separated by a space in the *Names* field and the names will be added in order when you click on the various nets.

## 4.7 Checking and Saving Your Schematic

Once you think that your schematic is complete, you will need to run a check on it. This check only checks for very rudimentary problems (such as unconnected pins or dangling wires) and for many more subtle or obscure problems that may cause trouble later on in programs that try to use your schematic.

- To check your schematic, click on the **Check and Save** icon. The results of the check are displayed in the CIW. If everything goes well, you should see a message in the CIW:

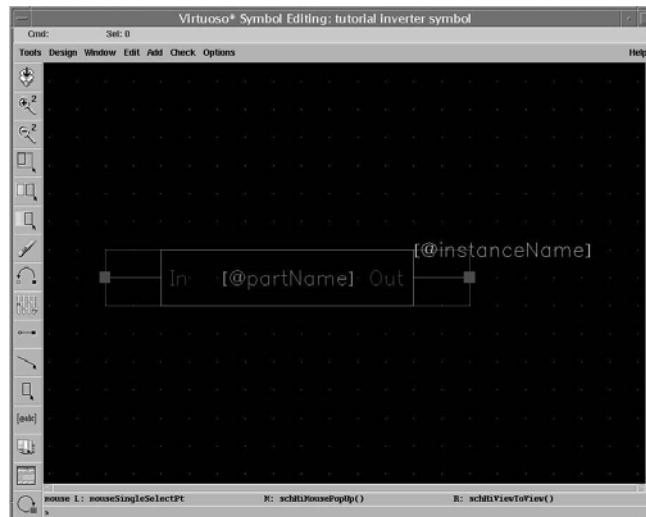
```
Schematic check completed with no errors.
"tutorial schematic inverter" saved.
```

- As an exercise, delete a wire segment from your schematic and do a check. A window named *Schematic Check* will appear indicating the number of errors and warnings found. Click **Close**.
- In the cellview, flashing markers highlight the areas that are causing the errors or warnings.
- To understand the cause of errors, select the drop down menu item **Check -> Find Marker**. A new window appears showing the list of errors and warnings.
- Clicking on a particular error and warning from the list will highlight the markers that correspond to that particular error or warning highlighted on the schematic. Click **Delete** to remove the marker from the cellview.
- There might be situations, for example, where you do not need to use a certain pin on an instance and would just want to leave it "floating". The checker will indicate a warning that the pin is floating. To ignore the warning and to prevent it from appearing repeatedly, select the warning from the list and click **Ignore**. That particular warning will not appear in the next check. To make the warning reappear, click **Restore All** in the same window.
- Fix the schematic, do a check and save it.

## 5.0 Creating and Editing a Symbol for Your Schematic

At this point, you should have a schematic of a CMOS inverter that passes the check without errors. You will now generate a symbol for the inverter.

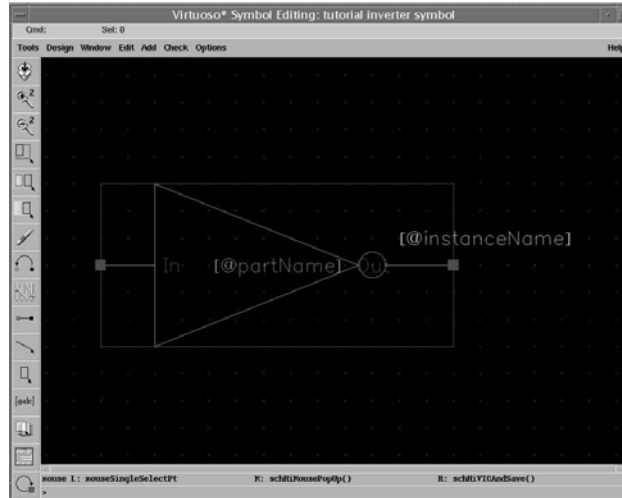
- From your cellview of your inverter schematic, select **Design -> Create Cellview -> From Cellview** from the drop down menu. A new window named *Cellview From Cellview* appears. The window shows entries for the active design, so *Library Name* should be set to *tutorial*, *Cell name* to *inverter*, and *From View Name* to *schematic*. Check to make sure that *To View Name* is set to *symbol* and all the above information is correct, then click **OK**.
- The Symbol Editor window appears, displaying the *inverter* symbol generated by the schematic composer software. This symbol is functional and can be readily used, but the shape is a simple rectangle and not the conventional inverter symbol.



- Note that the icons on the left are slightly different than the ones in the Schematic Editor window.
- To draw the new shape, select **Add -> Shape -> Polygon**. Point to the first point of the polygon, followed by the next point until you end with your original starting point and form a triangle. Add a small circle at the apex of the triangle at the output end by selecting **Add -> Shape -> Circle**. Point and click to define the center of the circle and move to size the circle by moving the mouse.
- Carefully delete the inner box. Do not delete the outside box that runs through the pin squares.
- Select the top and bottom sides of the outside box and move them out to the points on the triangle (click and drag), so that the entire triangle is inside the box.

This box determines the boundaries of the symbol (i.e., determines what area you click on the schematic to select the symbol)

- The final result should look like this:



- Check and save the symbol by using the **Design -> Check and Save** command from the drop down menu.
- If you take a look at the *inverter* cell in the *tutorial* library, you can see that it has now a schematic cellview and a symbol cellview.
- Close the Symbol Editor.

Now you have completed a discrete CMOS inverter design, with its own symbol, and the instance parameters as specified. The next step is to perform a simulation on the inverter to verify that it works and to evaluate its performance.

## 6.0 Printing from Cadence

- Before you start printing, copy the `display.drf` file from the `tsmc025` directory (`/package/cae/cadence/cells/tsmc025`) to your home directory. It contains color mappings that are needed to produce a readable B/W print.
- To print out any type of cellview, select **Design -> Plot -> Submit**. In the new window that appears, turn off the *header* option unless you want one. If you are creating a plot file for the color plotter, always turn this off.

- Click on the **Plot Options** box. In the *Plot Options* window, turn on the *Center Plot* option and choose an appropriate paper size:

For the laser printers, choose A.

For the DesignJet, B = 11x17, C = 17x22, D = 22x34, E=34x44.

D is the largest size possible for 24" wide paper; E is the largest for 36" paper.

Make sure the scale settings are set to fit within the paper size.

- If plotting for the Design jet, choose the plot to file option and give the file to your TA to plot.
- Disable the *mail log to* option, or make sure that the username is one you want the mail sent to.
- If there are other printers you want to use that are not in the printer list in Cadence, enter `plotconfig` in a UNIX prompt window and choose the *Personal* option.