

Automatic Layout Generation Using Silicon Ensemble

CheeWe Ng

`cheewe@mtl.mit.edu`

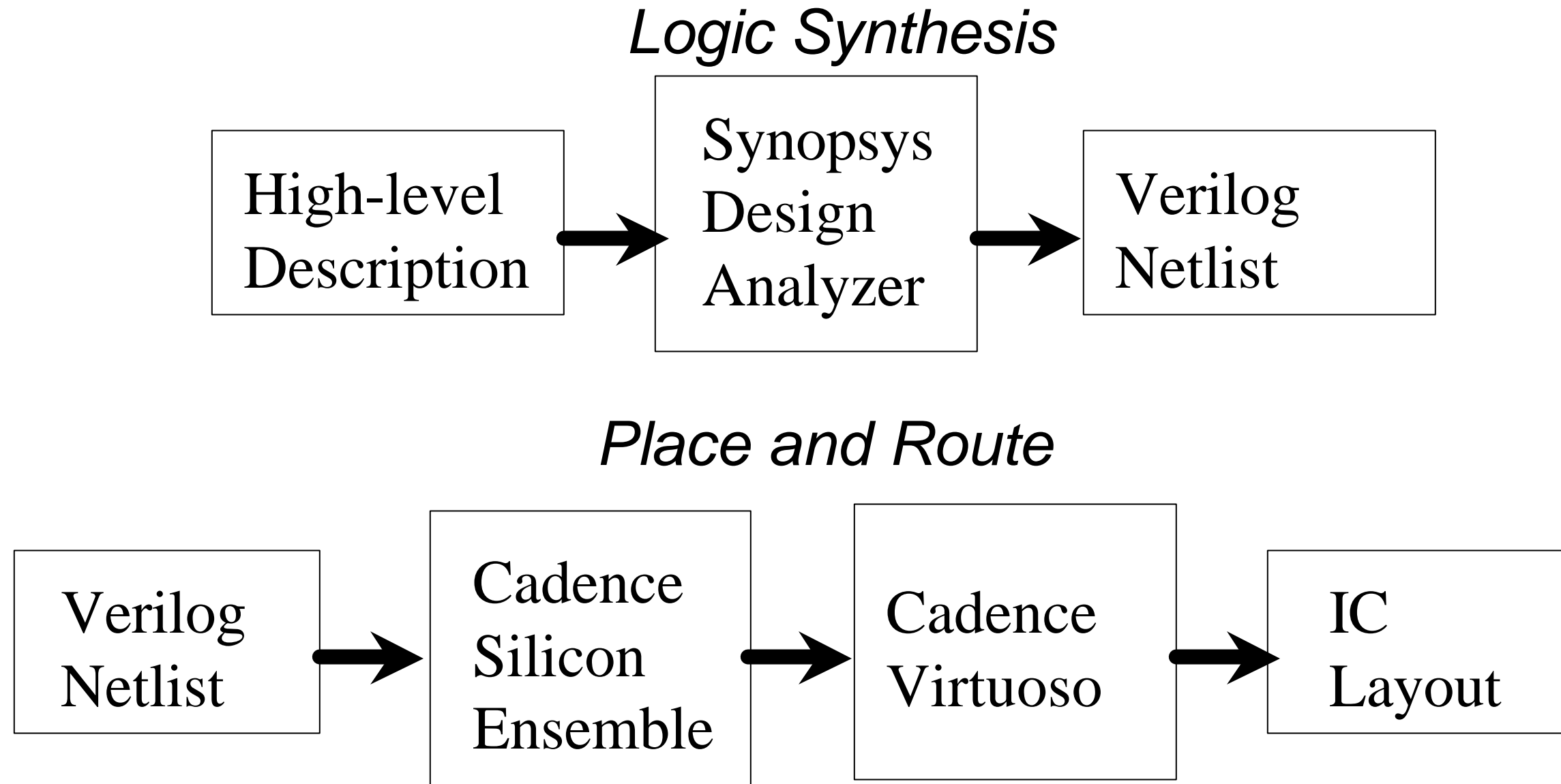
This document can be downloaded from

<http://www-mtl.mit.edu/research/icsystems/research/presentations.html>

Outline

- Overview of Design Flow
From High-level description to IC Layout
- Tutorial Example
- Setting up Tools
- Where to find more information

Overview of Design Flow



We'll come back to this picture to talk about setting up the libraries for these tools.

Outline

- Overview of Design Flow
From High-level description to IC Layout
- **Tutorial Example**
- Setting up Tools
- Where to find more information

Tutorial Example

- 4 bit adder in VHDL

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_signed.all;

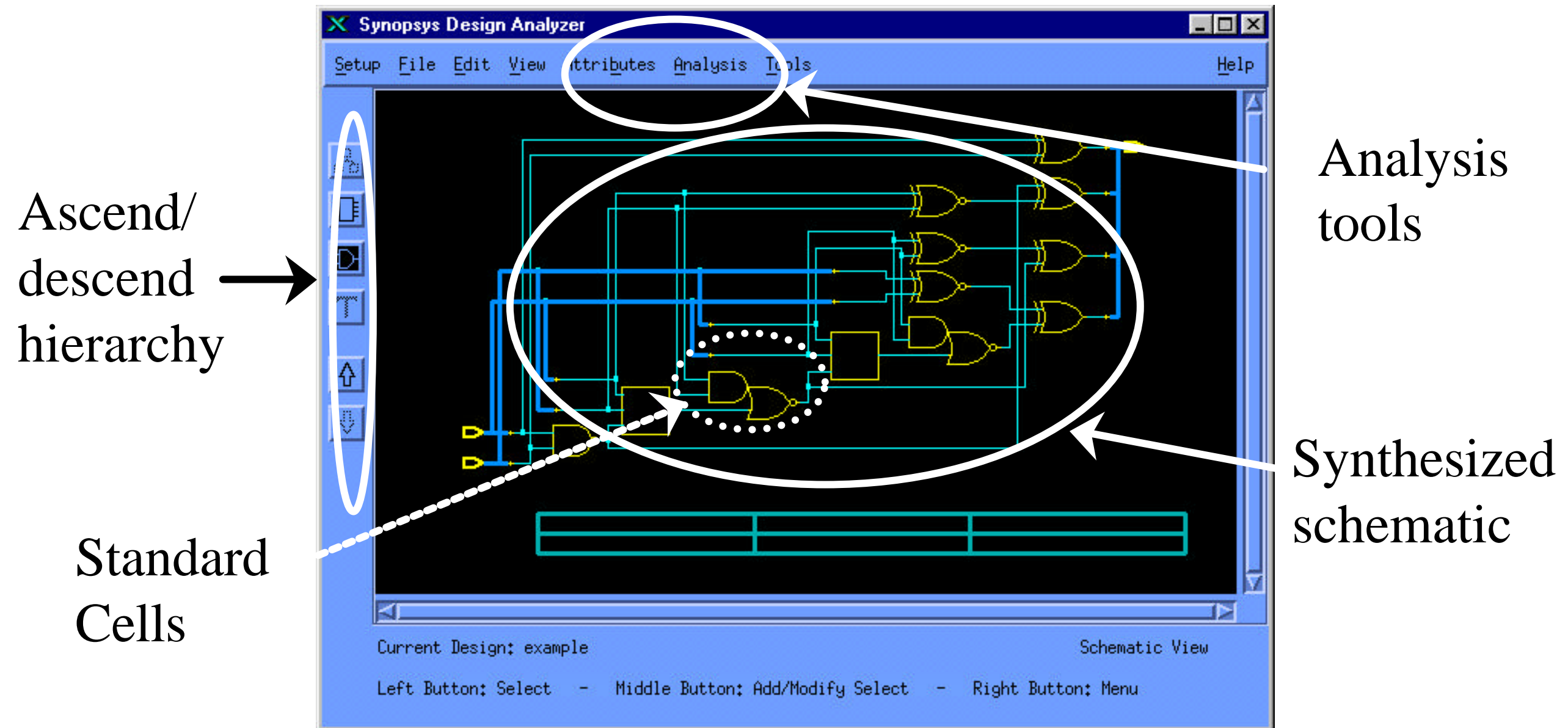
entity example is
  port(A: in  std_logic_vector(3 downto 0);
        B: in  std_logic_vector(3 downto 0);
        Y: out std_logic_vector(3 downto 0));
end example;

architecture behavior of example is
begin
  Y<=A+B;
end;
```

Step 1: Design Analyzer

- At the prompt (assuming the path and license environment variable is set), type `design_analyzer`
- Read in the VHDL file by using the `File` → `Read` menu
- Open the command window by using the `Setup` → `Command Window` menu
- At the command window prompt, type `compile`

Design Analyzer



Step 1 (continued)

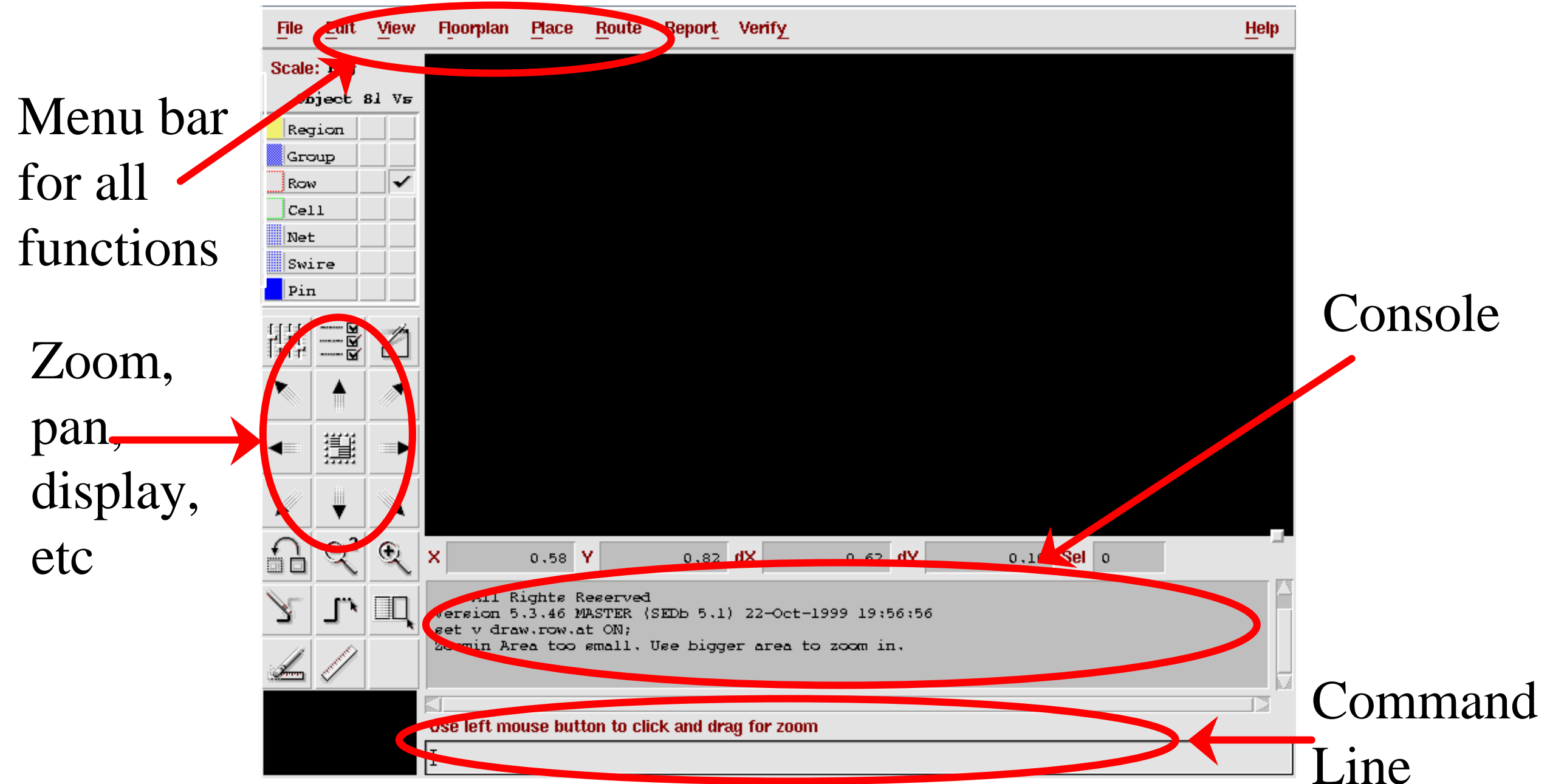
- Next, save your design as a Verilog netlist by using the File → Save As menu.
- The Verilog netlist will look like:

```
module example ( A, B, Y );
input  [3:0] A;
input  [3:0] B;
output [3:0] Y;
    wire n59, n60, n61, n62, n63, n64, n65, n66;
    AOI2BB1X1 U7 ( .A0N(A[2]), .A1N(B[2]), .B0(n60), .Y(n59) );
    AOI21X1 U8 ( .A0(A[2]), .A1(B[2]), .B0(n59), .Y(n61) );
    XOR2X1 U9 ( .A(n62), .B(n61), .Y(Y[3]) );
    XOR2X1 U10 ( .A(n63), .B(n60), .Y(Y[2]) );
    ...
endmodule
```

Standard Cells

Step 2: Silicon Ensemble

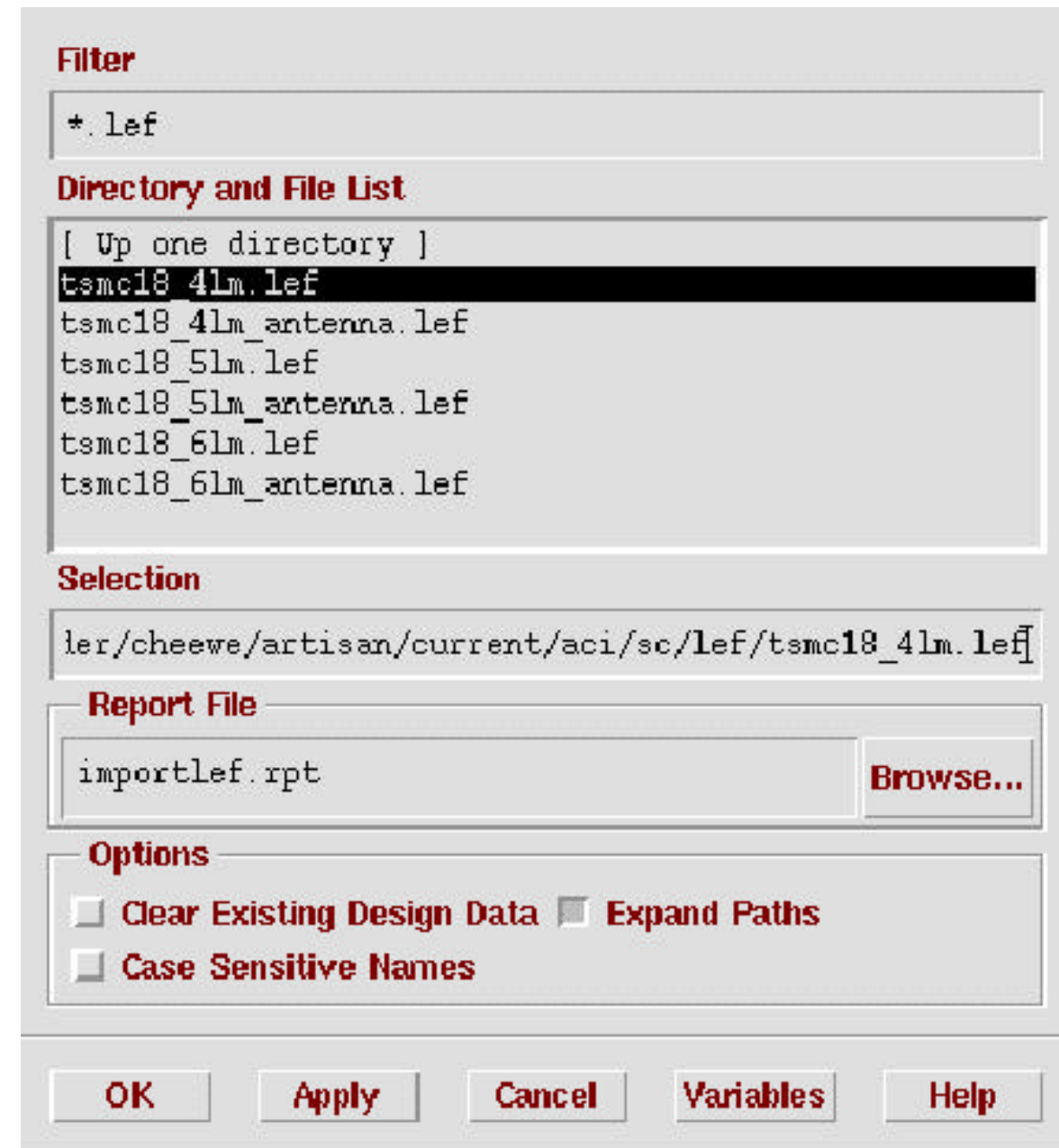
- Run silicon ensemble by typing `seultra`



Silicon Ensemble (Import LEF)

- Import “.lef” files that tells SE where the pins of each cell are located:
File → Import LEF

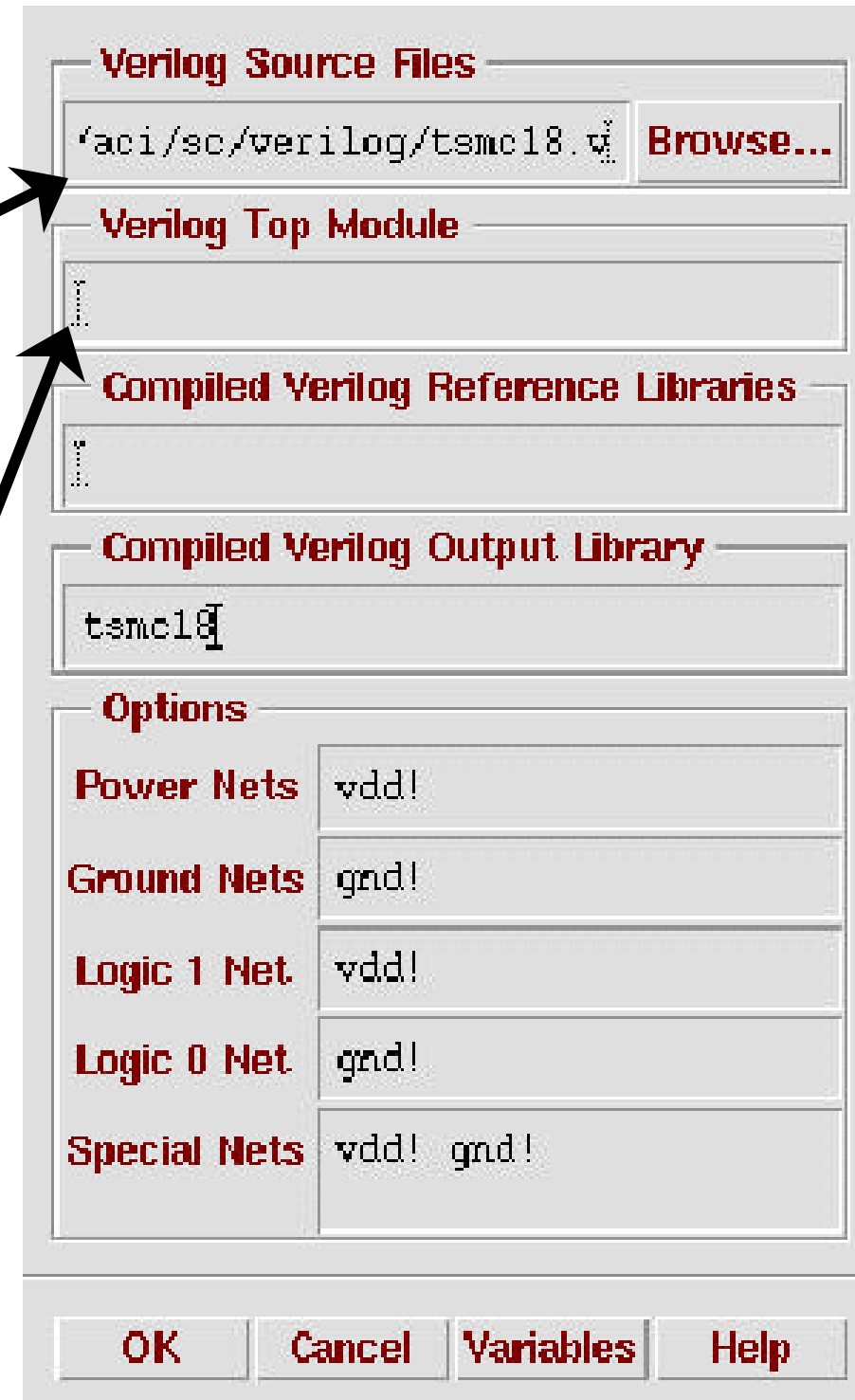
These are provided by the cell-library provider, e.g. Artisan Components



Anantha, Charlie, and Harry's groups can look at
/vader/cheewe/artisan/current/aci/sc/

Step 2 (continued)

- Import Verilog netlist from logic synthesis
 - Need to import a module declaration of all the standard-cells in Verilog as well.
 - Be sure to specify the name of the top module, e.g. “example”



The screenshot shows a dialog box for Verilog compilation with the following sections:

- Verilog Source Files:** A text field containing the path `'aci/sc/verilog/tsmc18.v` and a **Browse...** button.
- Verilog Top Module:** An empty text field.
- Compiled Verilog Reference Libraries:** An empty text field.
- Compiled Verilog Output Library:** A text field containing `tsmc18`.
- Options:** A table of options:

Power Nets	vdd!
Ground Nets	gnd!
Logic 1 Net	vdd!
Logic 0 Net	gnd!
Special Nets	vdd! gnd!

At the bottom of the dialog are four buttons: **OK**, **Cancel**, **Variables**, and **Help**. Two black arrows point from the text in the list on the left to the **Verilog Source Files** and **Verilog Top Module** fields in the dialog.

Step 2 (continued)

- Floorplan → Initialize Floorplan menu

IO To Core Distance:
Region for Vdd and Gnd rings

Design Statistics			
Number of:			
Cells	12	Blocks	0
IO Pads	0	IO Pins	12
Corner Pads	0	Nets	22
Area (Square Microns)			
Cells	256.133		
Blocks	0.000		
IOs	0.000		

IO To Core Distance

Left / Right microns 10.0000

Top/Bottom microns 10.0000

Core Area Parameters

Row Utilization(%) 85.0

Row Spacing tracks 0

Block Halo Per Side microns 1.0000

Flip Every Other Row Abut Rows

Calculate Expected Results

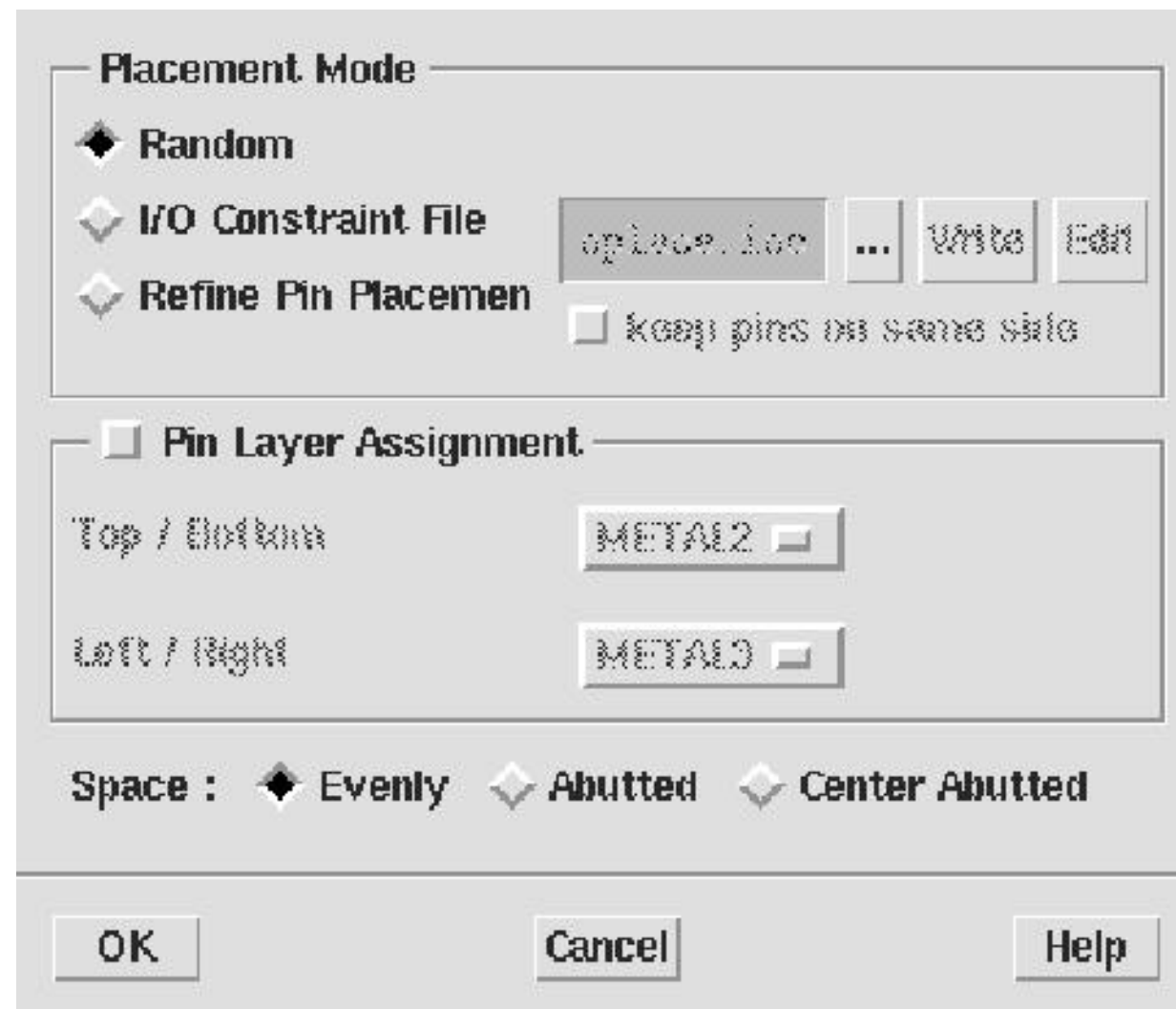
Aspect Ratio: 1.00 Width: 37.359 microns, Height: 37.359 microns.
Core row utilization = 97.59%.
Chip Area = 1395.695 sq. microns.
IO to Core Distance (microns): X: 10.000 Y: 10.000
Number of Standard Cell Rows = 3.
Design is core-limited.

OK Apply Cancel Variables Help

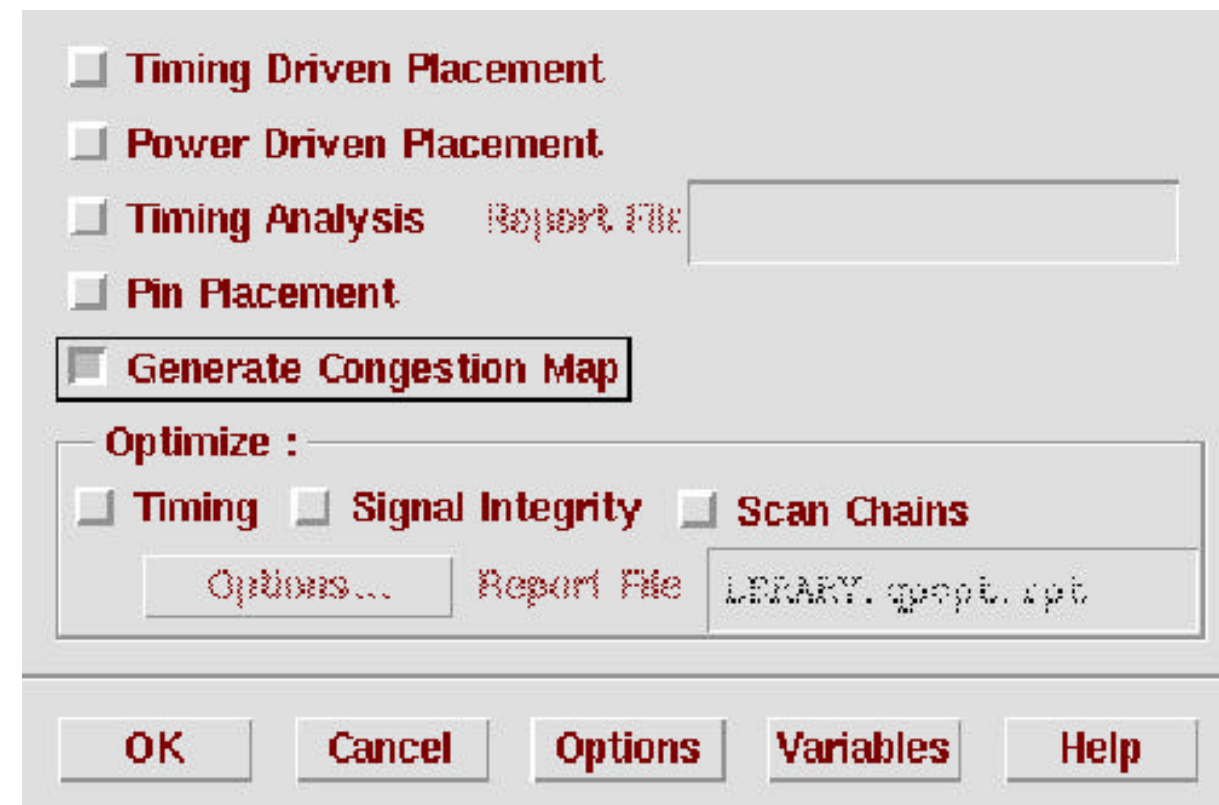
Select
“Flip
Every
Other Row

Step 2 (continued)

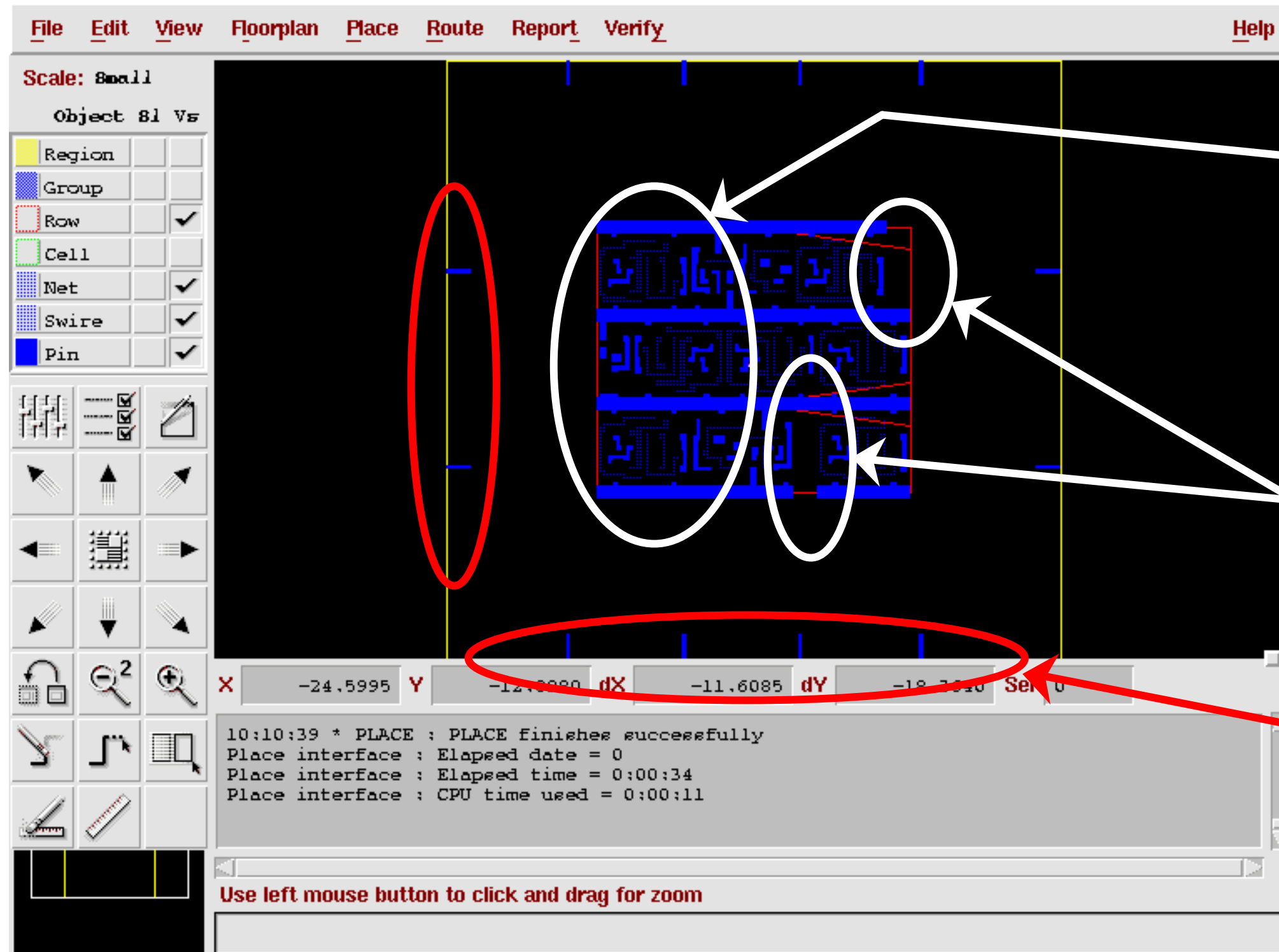
- Place → Place IOs



- Place → Place Cells



Silicon Ensemble (after placement)



Placed cells with Vdd and Gnd Rails

Unused Slots

Placed IO Pins

Step 2 (continued)

- Add filler cells:
- Place → Filler Cells
→ Add

FILL1 is a standard-cell
with no transistors

You can deselect all
the options

Model: FILL1
Prefix: Fill

Overlaps OK

Placement

Freendcap Postendcap

North South East West

Flip North Flip South Flip East Flip West

Pin

	Pin	Net
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		

Special Pins

	Special Pin	Special Net
<input type="checkbox"/>		
<input type="checkbox"/>		
<input type="checkbox"/>		

Area

X1	-17.1600	Y1	-16.8000
X2	17.8200	Y2	17.3600

OK Cancel Variables Help

Step 2 (continued)

- Add power and ground rings:
 - Route → Plan Power and Click “Add Rings”

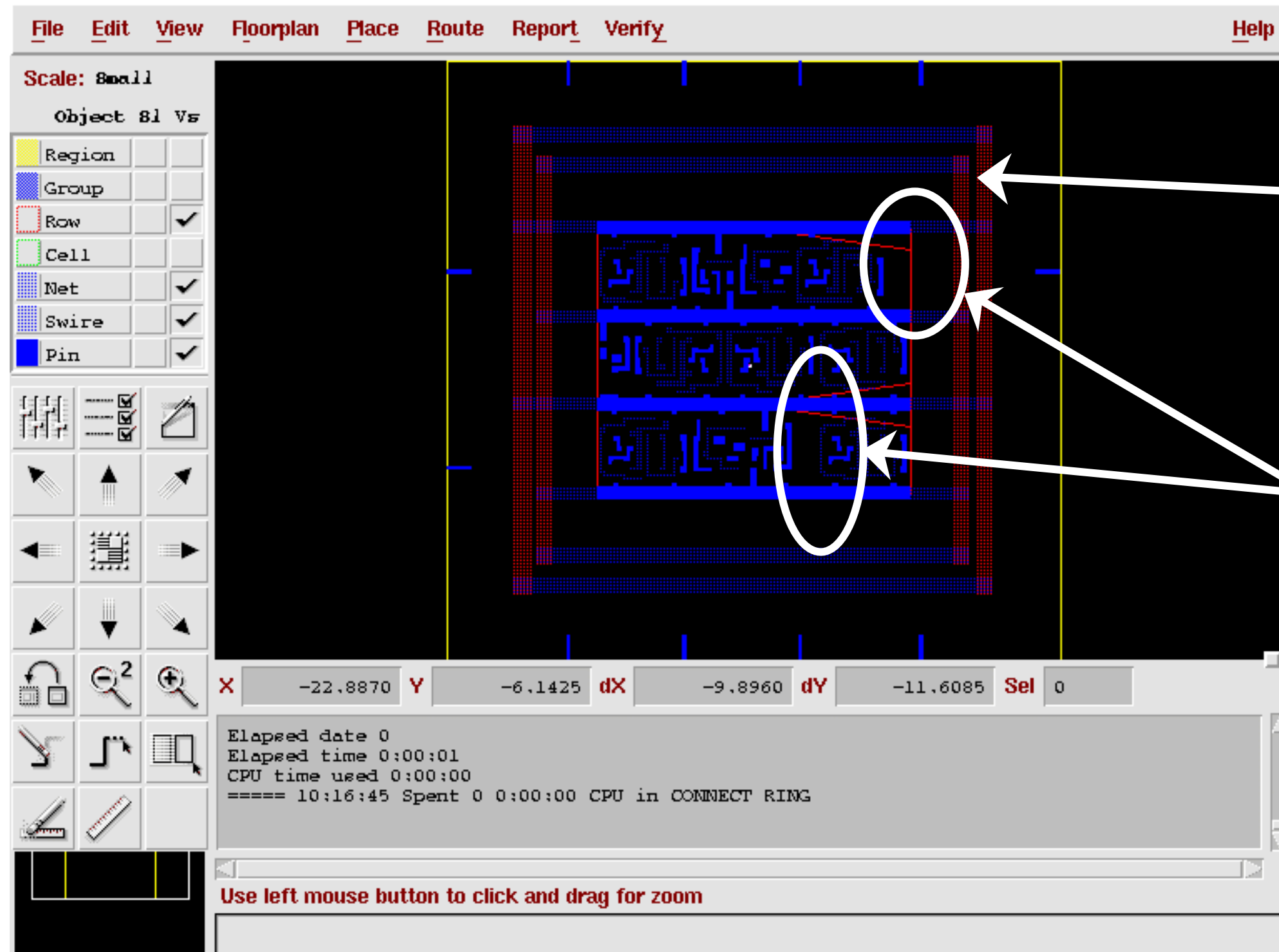
Ring Width
& position

Ring	Layer	Core Ring Width	Core Ring Spacing	Block Ring Width
Horizontal	METAL1	11.0000	Center	0.0000
Vertical	METAL2	11.0000	Center	0.0000

Ring	Layer	Core Ring Width	Core Ring Spacing	Block Ring Width
Horizontal	METAL1	0.0000	Center	0.0000
Vertical	METAL2	0.0000	Center	0.0000

- Route → Connect Rings”

Silicon Ensemble (w/ rings and filler cells)

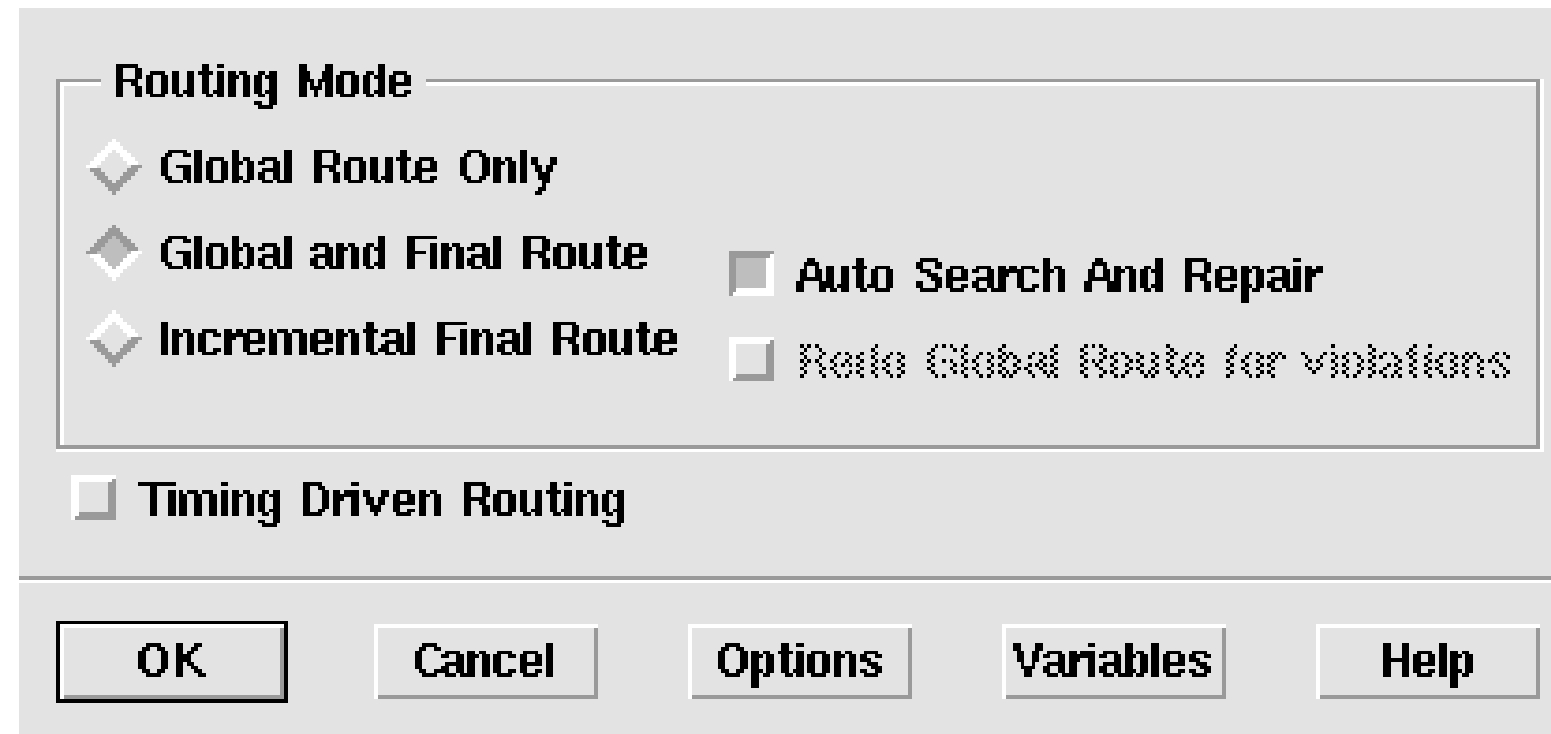


Vdd and
Gnd Rings

Filler Cells

Step 2 (continued)

- Routing: Route \rightarrow WRoute

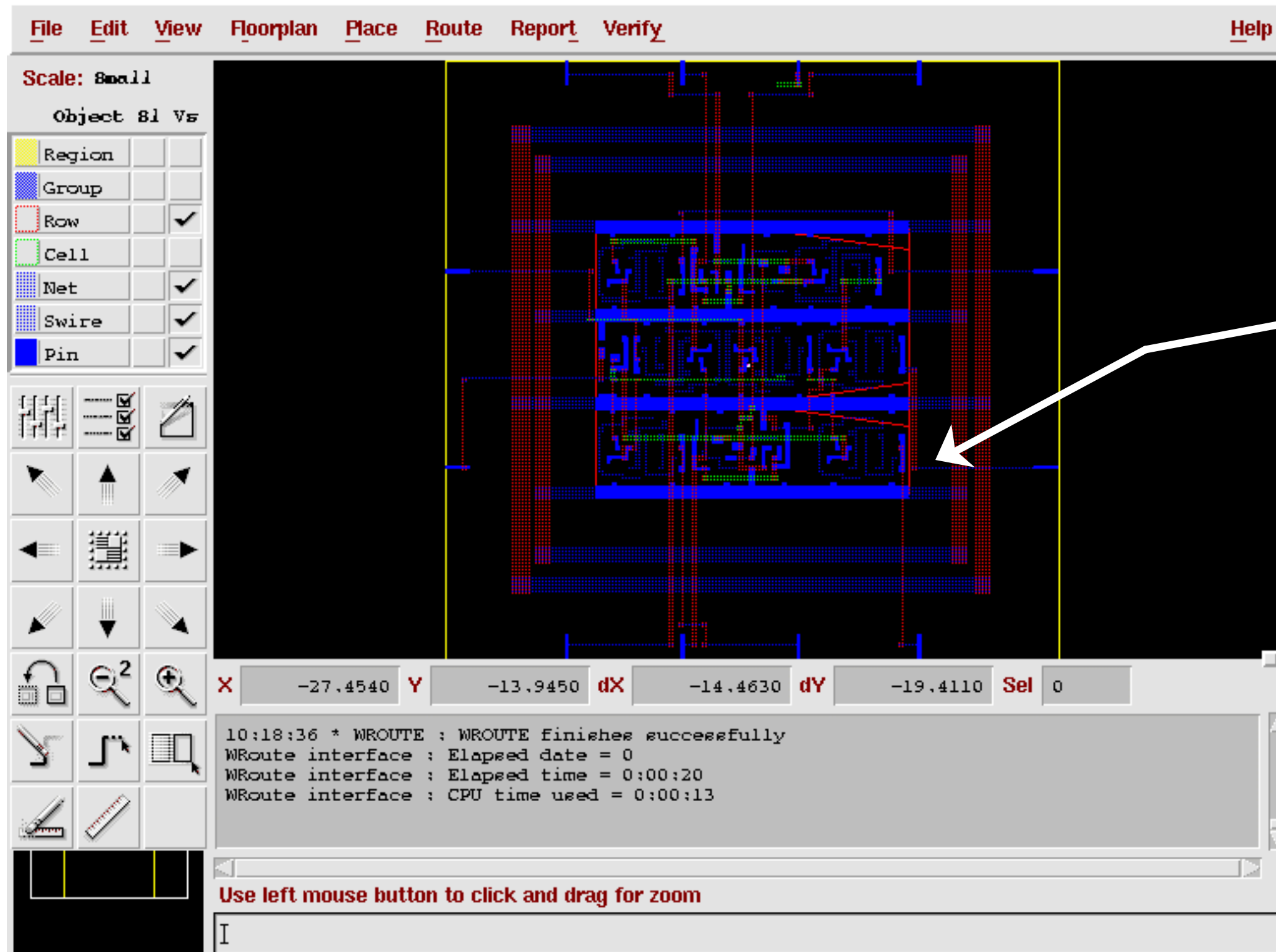


The image shows a dialog box titled "Routing Mode" with the following options:

- Global Route Only
- Global and Final Route
- Incremental Final Route
- Auto Search And Repair
- Route Global Route for violations
- Timing Driven Routing

At the bottom of the dialog box are five buttons: OK, Cancel, Options, Variables, and Help.

Silicon Ensemble (placed & routed)



Routing

Step 2 (finally...)

- Export to GDSII (stream format) to be imported into Virtuoso: File → Export



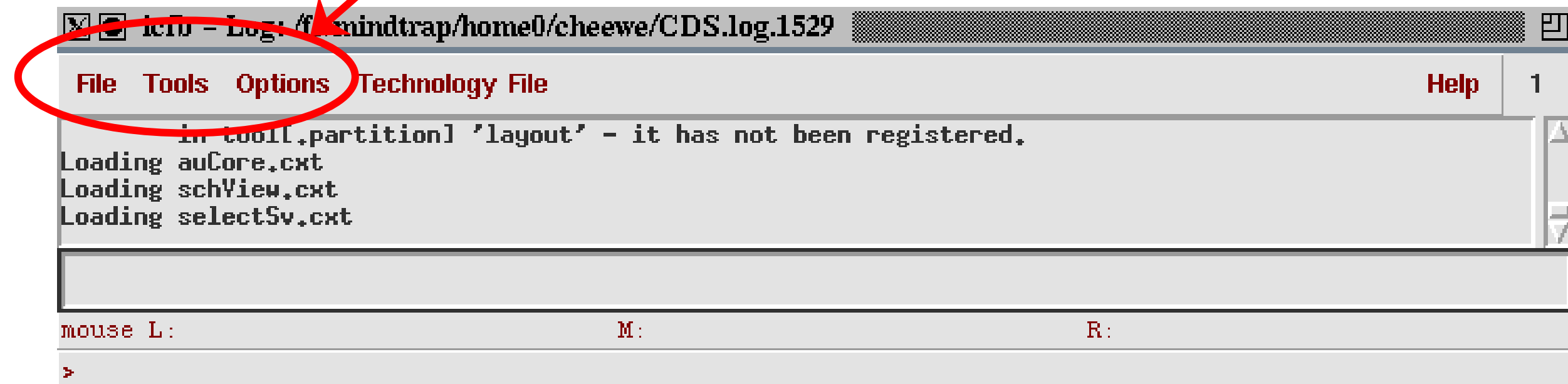
Map file
containing
the layer
numbers
for gds2

```
13 POLY1;  
16 METAL1;  
17 VIA12;  
18 METAL2;  
27 VIA23;  
28 METAL3;  
29 VIA34;  
31 METAL4;  
40 NAME METAL1;  
41 NAME METAL2;  
42 NAME METAL3;  
43 NAME METAL4;  
62 CELL;
```

Step 3: Cadence Virtuoso

- Run Design Framework by typing `icfb`

Menu to run Library Manager and to import GDSII from Silicon Ensemble



```
icfb - Log: /home/mindtrap/home0/cheewe/CDS.log.1529
File  Tools  Options  Technology File  Help  1
in [tool.partition] 'layout' - it has not been registered.
Loading auCore.cxt
Loading schView.cxt
Loading selectSv.cxt

mouse L:           M:           R:
v
```

Step 3 (continued)

- Import GDSII file from Silicon Ensemble:
File → Import-Stream

User-Defined Data

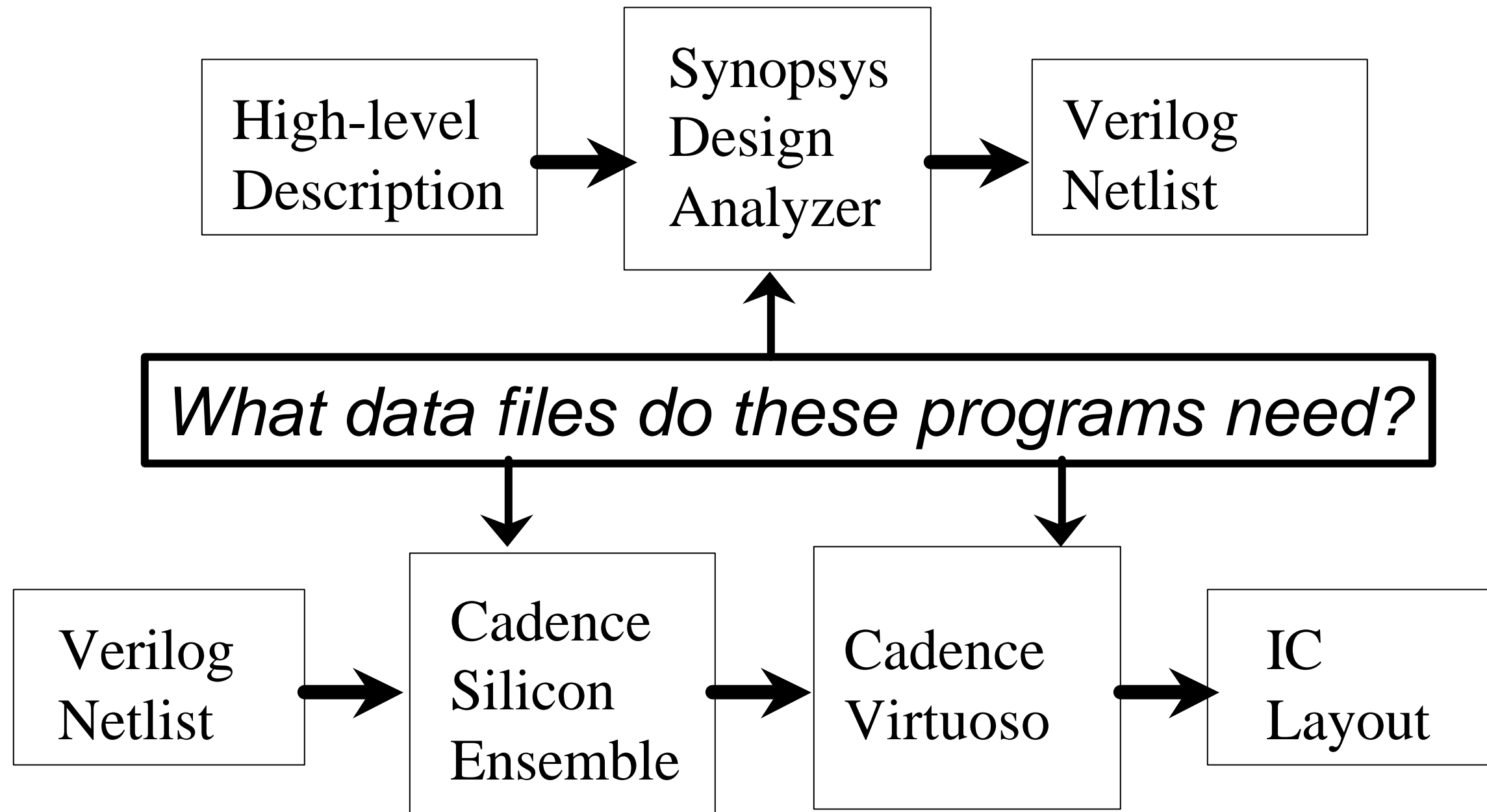
Files from the foundry,
e.g. TSMC

Anantha, Charlie, and Harry's groups can look at
/vader/cheewe/tsmc/Virtuoso

Outline

- Overview of Design Flow
From High-level description to IC Layout
- Tutorial Example
- **Setting up Tools**
- Where to find more information

Setting Up Tools



Design Analyzer

- Needs
 - DesignWare (comes with Design Analyzer)
 - Standard-cell “back-end views”
- These files are usually named
*.db, *.sldb, *.lib

Setting up Design Analyzer

- Need a `.synopsys_dc.setup` in the directory you run design analyzer.

```
...
search_path= search_path + ${synopsys_root}/libraries/syn+
${synopsys_root}/dw/sim_ver]+
"/u/vader/cheewe/artisan/current/aci/sc/synopsys"+
"/u/vader/cheewe/artisan/current/aci/sc/vhdl"
target_library=typical.db
symbol_library=typical.db
synthetic_library=dw_foundation.sldb
# additions from DesignWare Foundation Quick Reference
link_library={typical.db, dw_foundation.sldb}
synlib_wait_for_design_license={"DesignWare-Foundation"}
...
```

Standard-cell Library

DesignWare

Setting up Silicon Ensemble

- Need a `se.ini` in the directory Silicon Ensemble is launched:

```
# from /u/vader/cheewe/artisan/aci/sc/lef/README
# Silicon Ensemble floorplan variables - required for TSMC .18
set v plan.rgrid.M1offset          560 ;
set v plan.rgrid.M2offset          660 ;
set v plan.rgrid.M3offset          560 ;

set v groute.Allow.OffGrid.PinAccess false ;
set v froute.Allow.OffGrid.PinAccess false ;
set v froute.Avoid.OffGrid.Blockage true ;
set v froute.Build.OffGrid.SPins    false ;

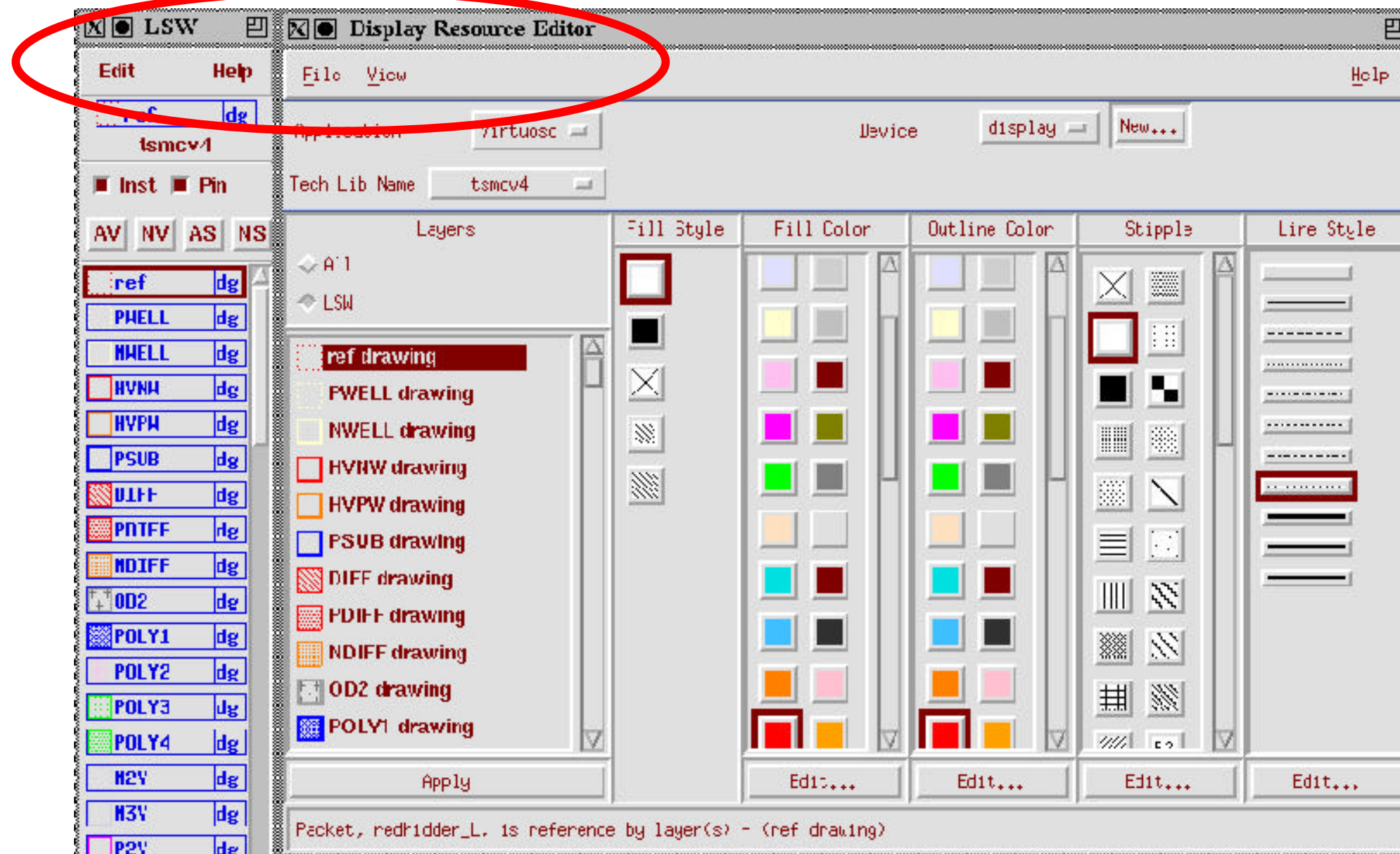
# so that pins are labeled with [] which is compatible with
# spice and verilog
SET VAR INPUT.VERILOG.BUS.DELIM "[ ]" ;
```

Setting up Design Framework

- Need to create a library that has the “front-end” layout views of all the standard-cells, and the `display.drf` file:
 - Import the GDSII layout views from the cell provider into a new library, like previously described.
 - Set up the display resource, like you would with any technology.

Cadence Display Resource

- LSW: File → Display Resource Editor
File → Load



Outline

- Overview of Design Flow
From High-level description to IC Layout
- Tutorial Example
- Setting up Tools
- **Where to find more information**

Where to find more information?

- **Synopsys SOLD (Anantha Group)**

`acroread`

`/usr/synopsys/current/doc/online/synth/
dctut/toc.pdf`

- **Cadence Openbook: (MTL Users)**

`openbook -f`

`/amd/mtlcad/cadence/DES4.3/doc/ASICpnr/
silref/silrefTOC.obk`