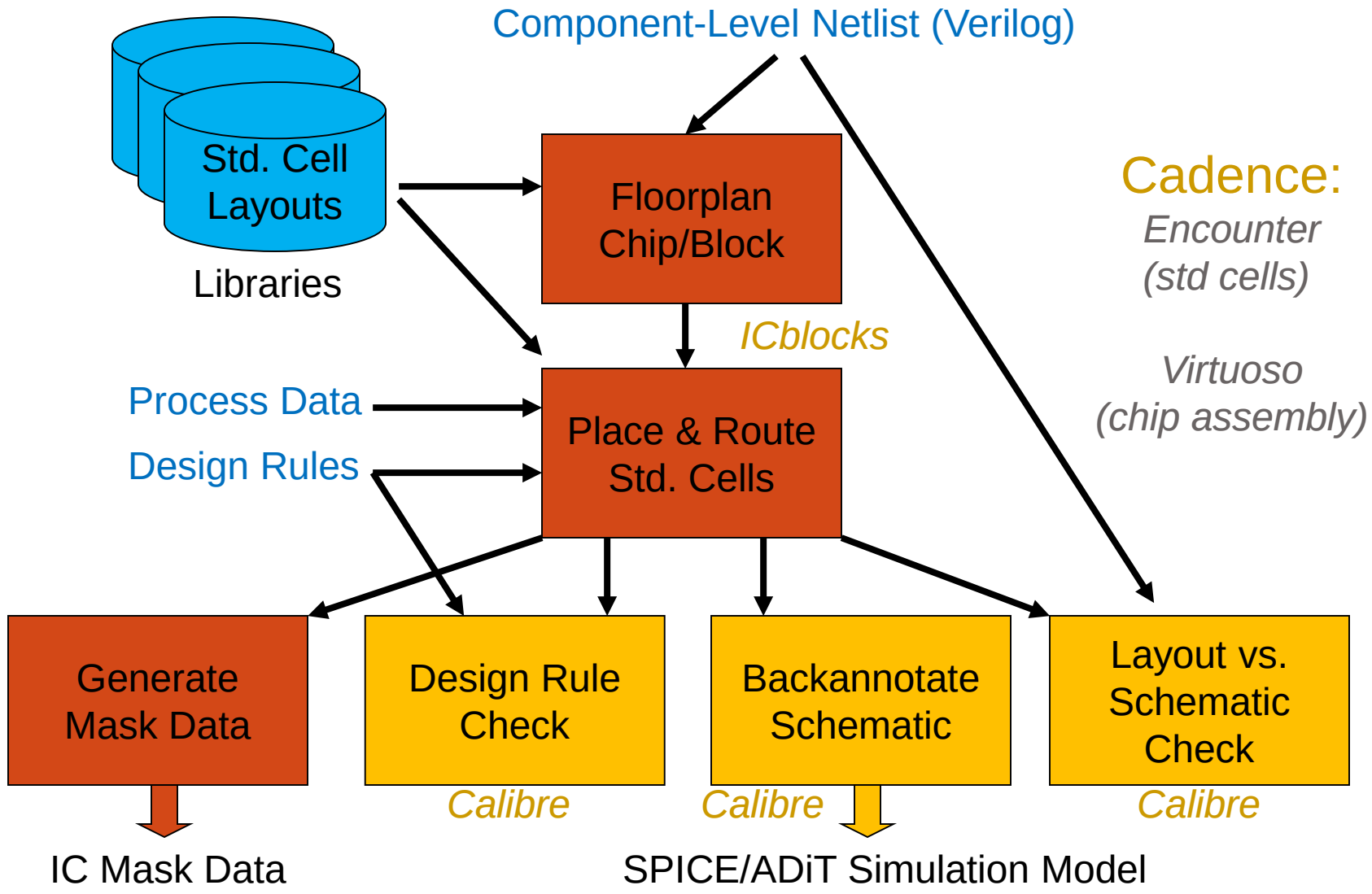
ASIC Physical Design
Post-Layout Verification

ASIC Physical Design (Standard Cell)

(can also do full custom layout)



Cadence setup

- Copy files from /class/ELEC6250/CadenceFiles

  (Replace dot with a period. Example: .cdsenv)

dotcdsenv	to your home directory
dotcdsinit	to your project directory
cds.lib	to your project directory
display.drf	to your project directory
dotsimrc	to your project directory
addpowerv1.txt	to your project directory

- Edit your .bashrc file with the setup information from

  /class/ELEC6250/CadenceFiles/dotbasrch

Import digital block into Virtuoso

- Import GDSII layout information into Virtuoso:
  - Encounter saves: mydesign.gds2
  - Import into a Cadence library
    - File > Import > Stream
  - Results in cell "layout" view
- Import circuit netlist into Virtuoso:
  - Gate-level netlist saved by Encounter: mydesign.v
  - Import netlist into a Cadence Library
    - File > Import > Verilog
  - Results in cell "schematic" and "symbol" views
    - Gates replaced by transistors using "cdslib" components

(Demonstration)

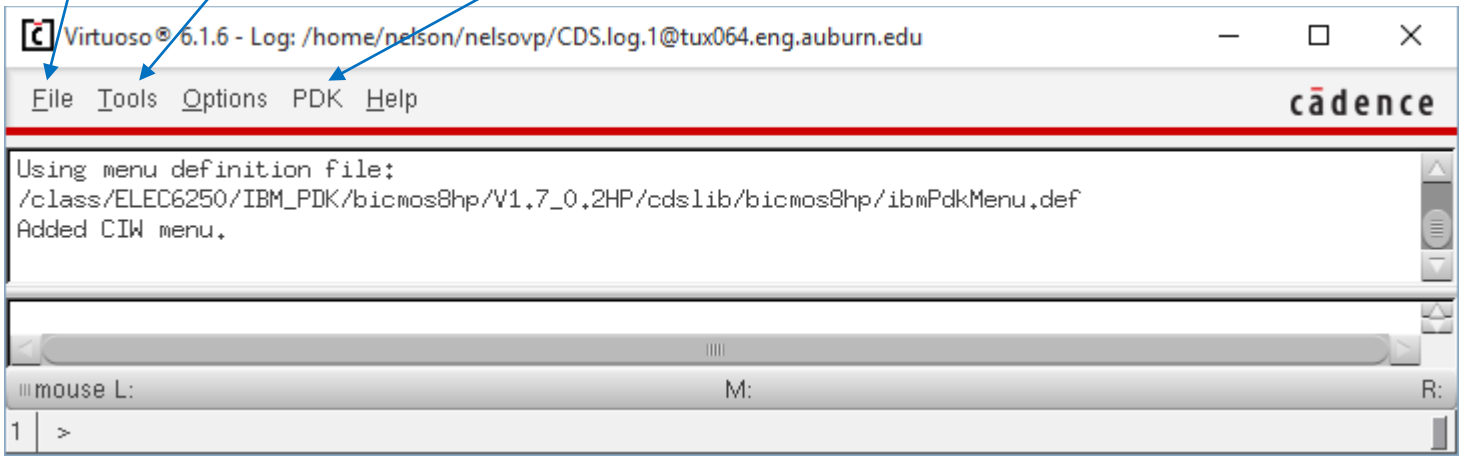
Virtuoso CIW (Command Interpreter Window)

Cadence libraries and tools are accessed from the CIW

Import/Export designs

Access libraries

BICMOS8HP PDK Items



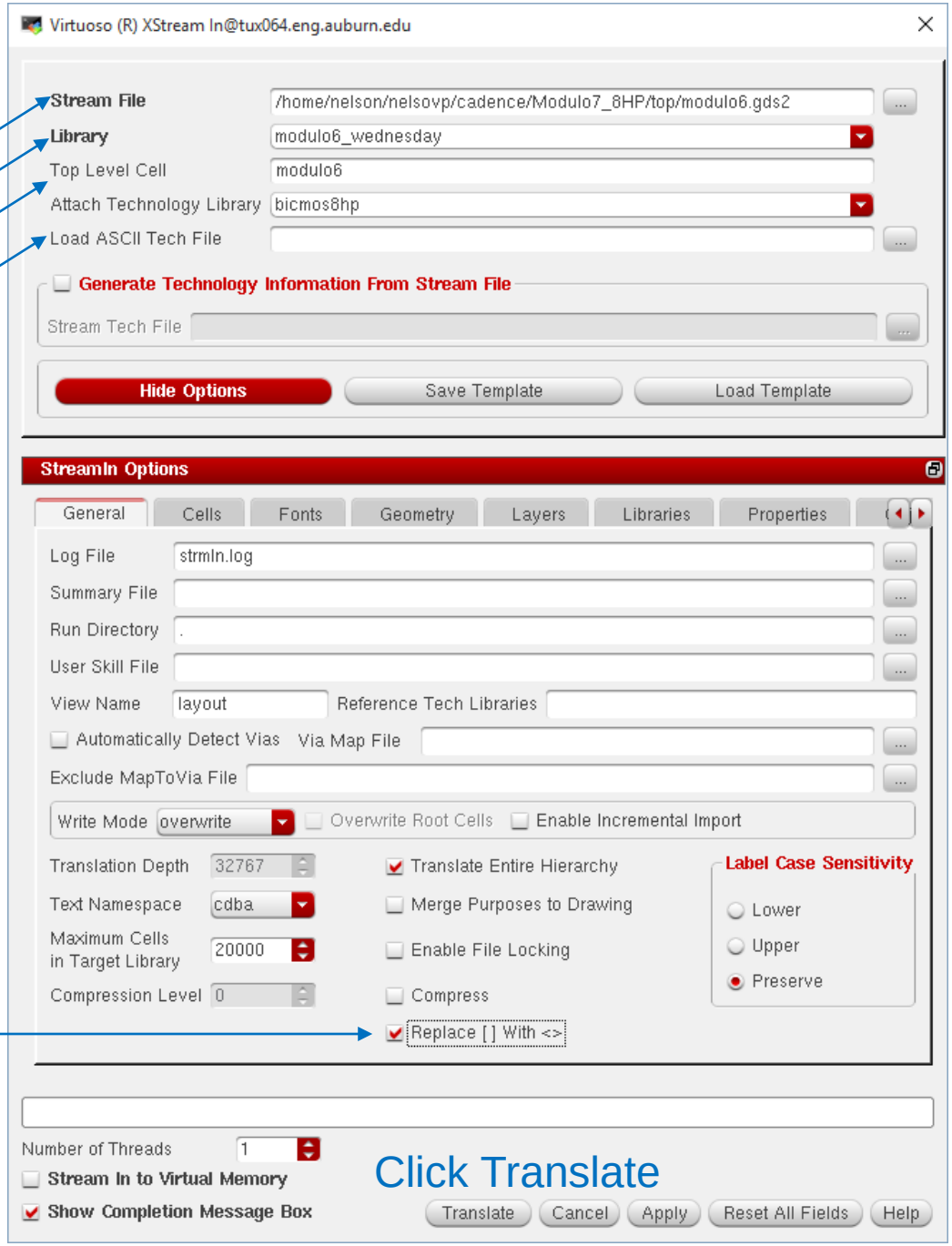
File > Import > Stream

GDSII file from Encounter

My library for this cell

Name of top design cell

Technology library



Replace Verilog [ ] with < >

Click Translate

Importing the Verilog netlist

- Verilog netlists saved by Synopsys Design Compiler and Cadence Encounter do not contain ports or definitions of power and ground connections.
- Manually add power/ground connections by executing the following perl script from a linux command line.

  perl addpowerv1.txt design.v design_vg.v

  where:

  addpowerv1.txt is provided with the setup files

  design.v is the netlist generated by Encounter

  design_vg.v is the netlist with VDD/GND added

Generated Verilog netlist

```
module modulo6 (VDD, VSS,
        CLEARbar,
        L_Cbar,
        CLK,
        I,
        Q);
  inout VDD, VSS;
  input CLEARbar;
  input L_Cbar;
  input CLK;
  input [2:0] I;
  output [2:0] Q;
```

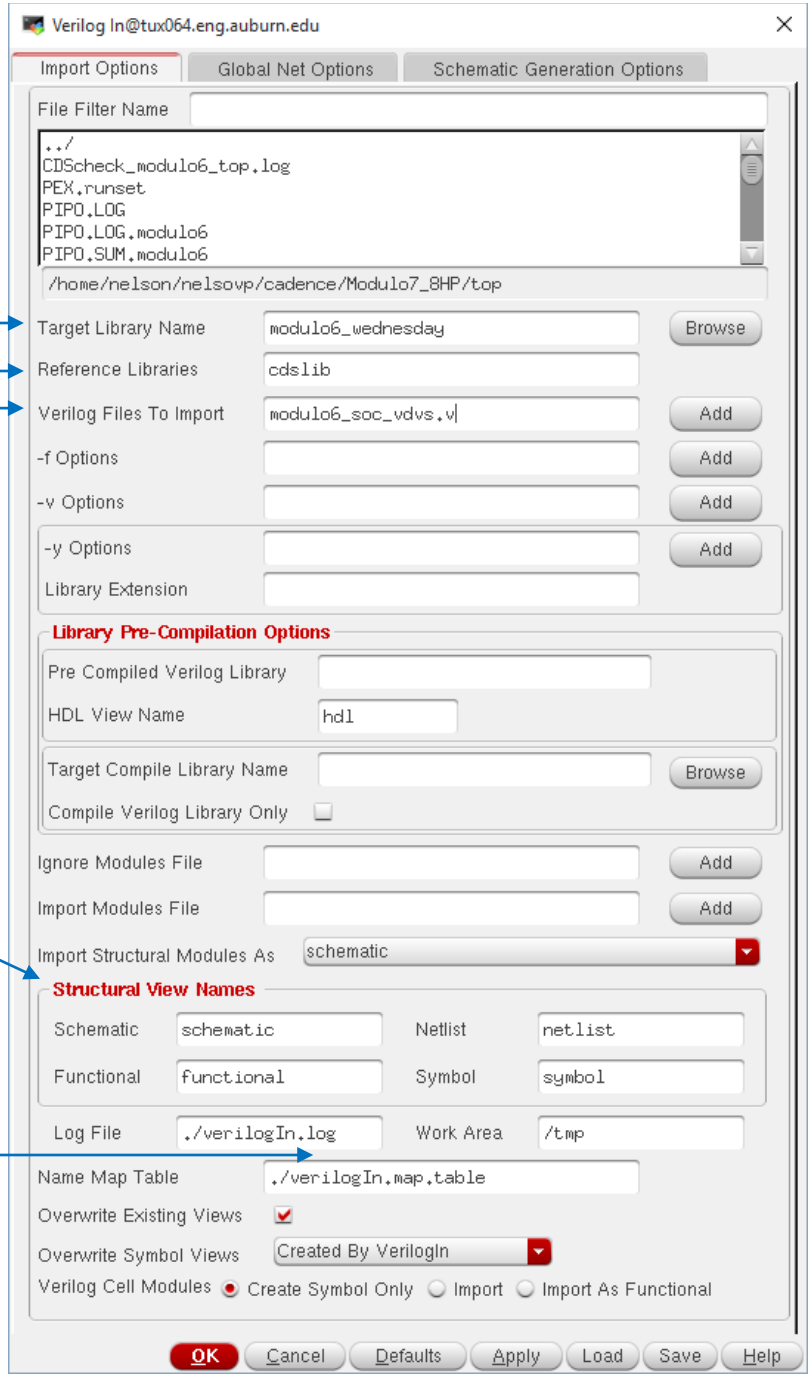
File > Import > Verilog

My library for this cell
Reference tech library*
Verilog file(s)

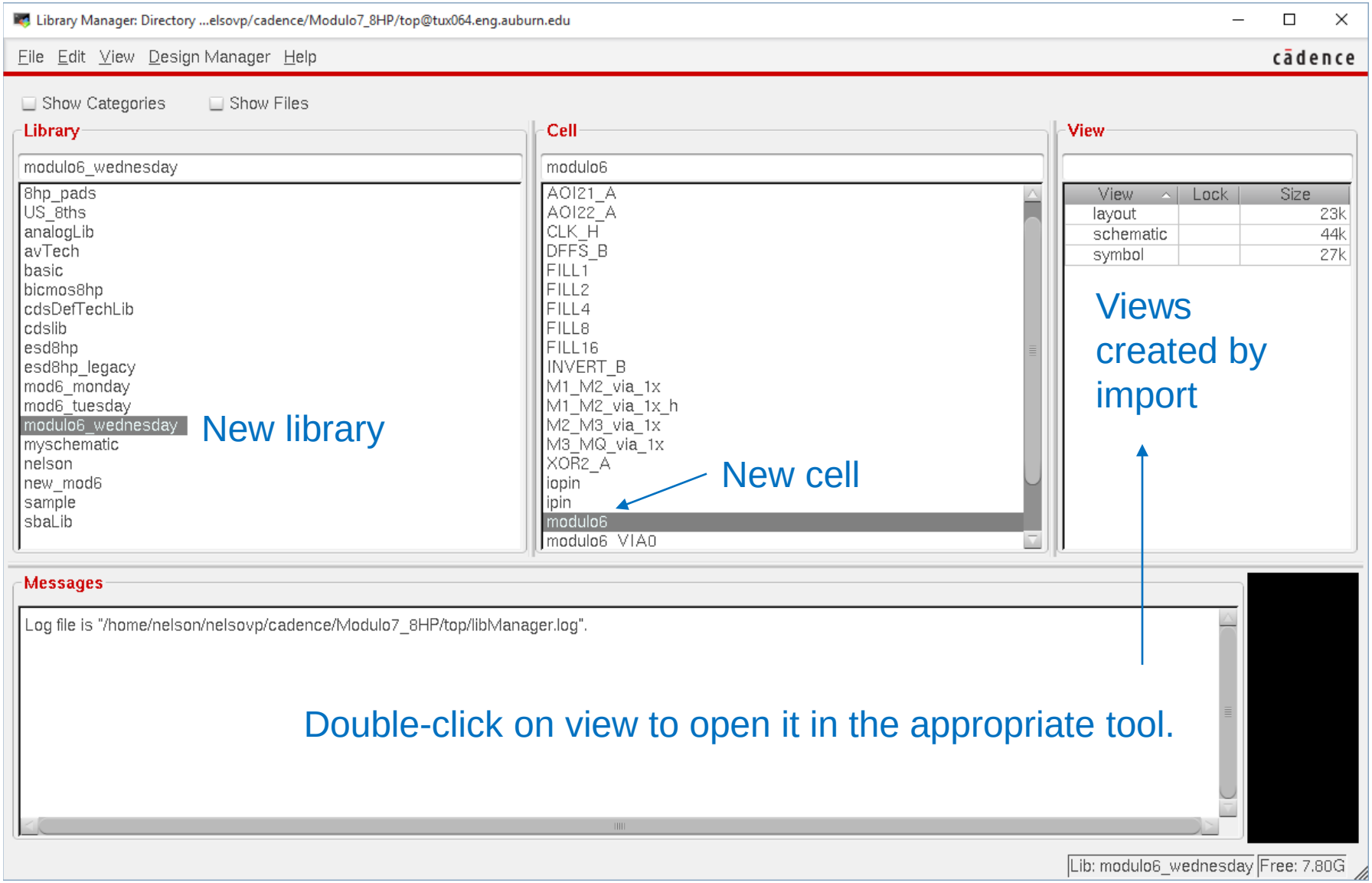
* Contains tech file for "311" bicmos8hp

Creates schematic and symbol views

Replace Verilog [ ] with < >

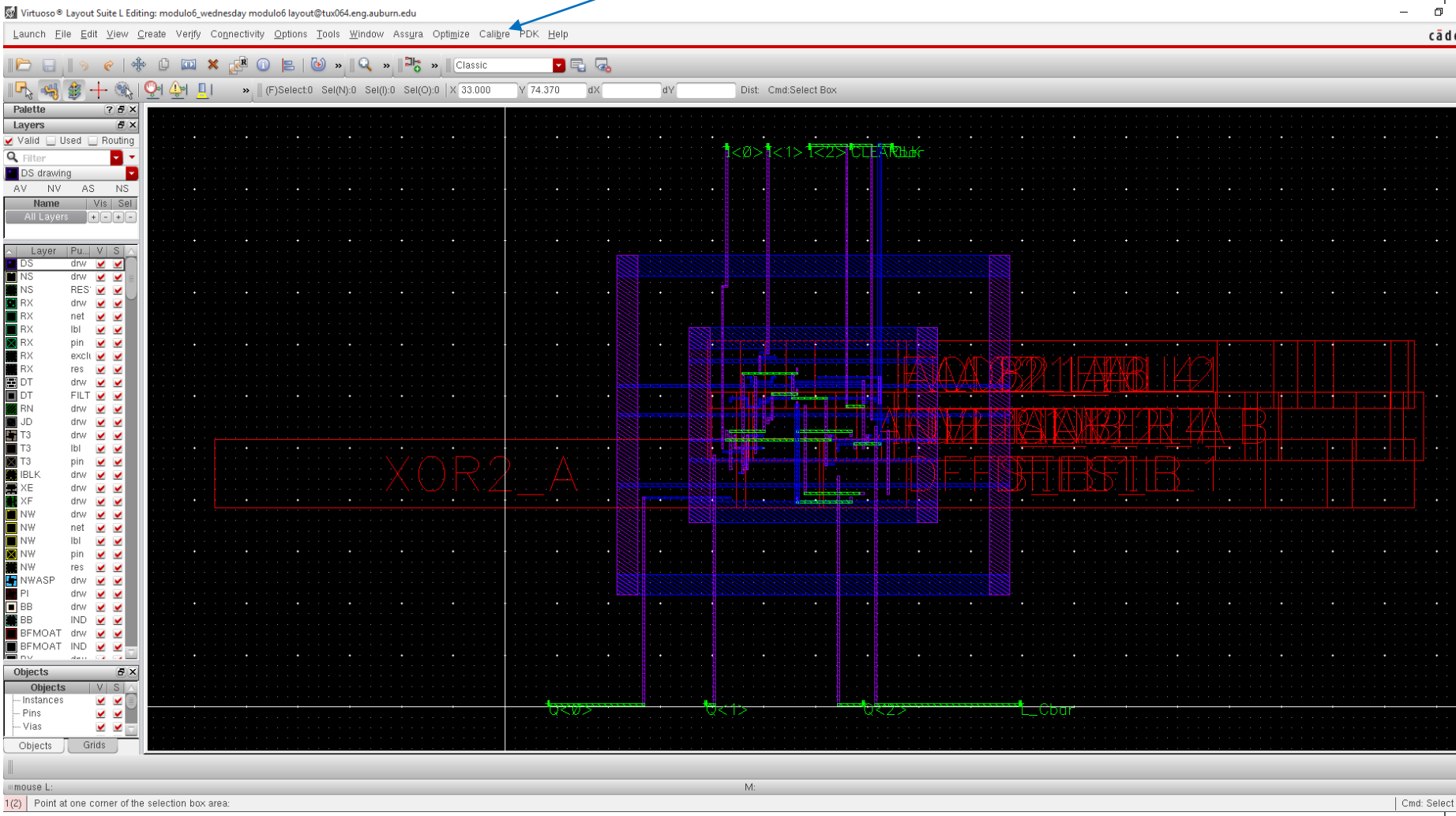


Library Manager

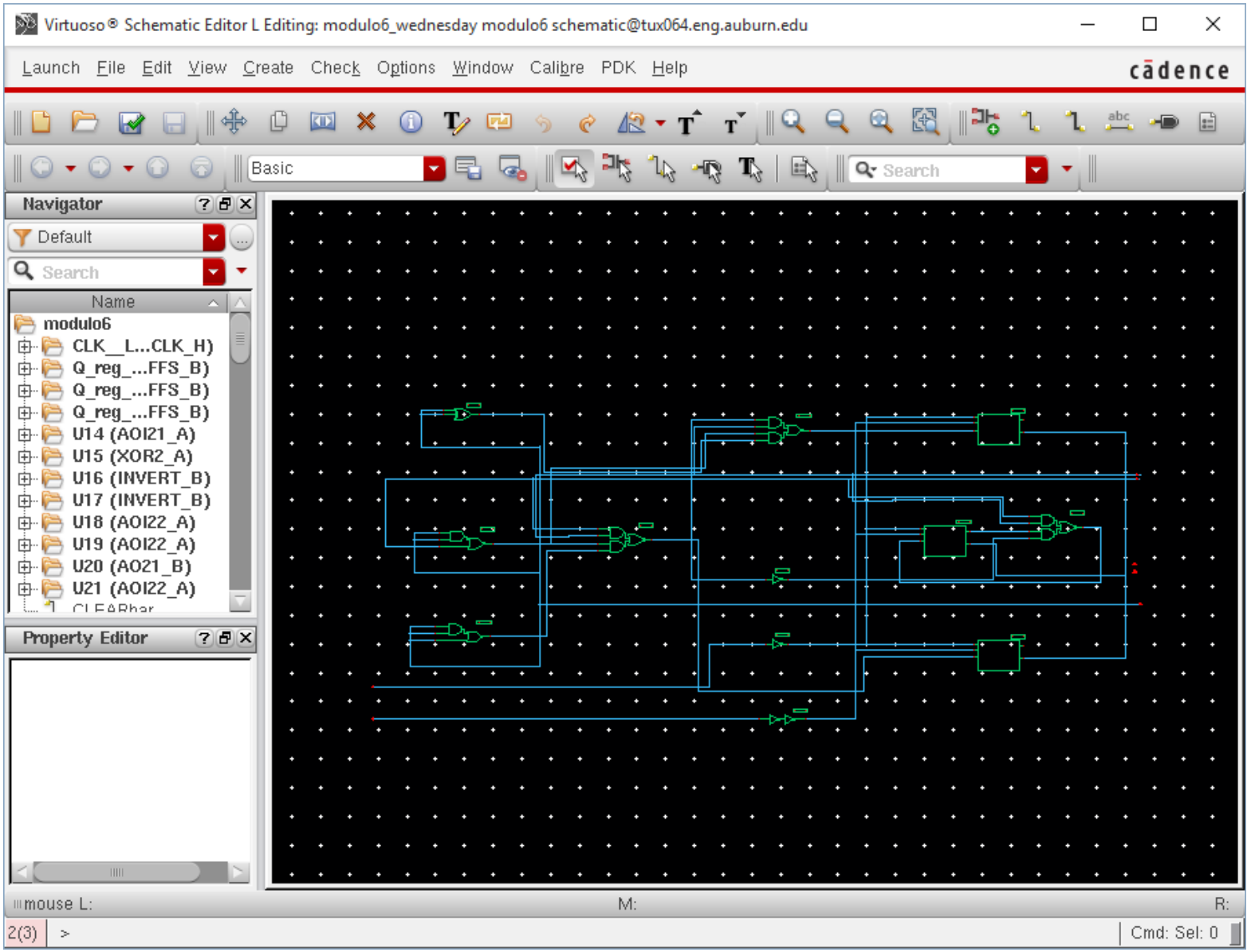


Layout view of “modulo6”

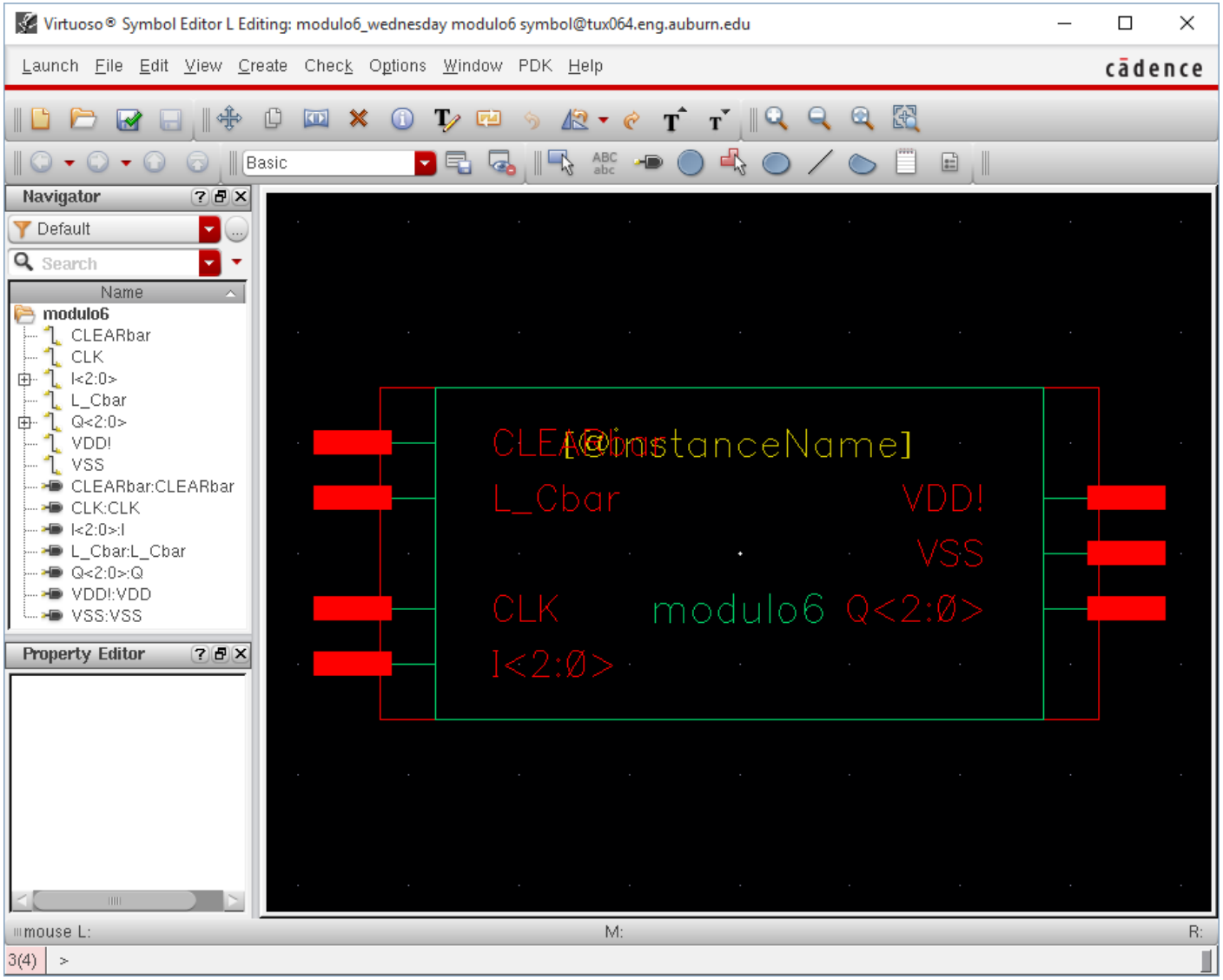
Calibre LVS/DRC/PEX



Schematic view of “modulo6”



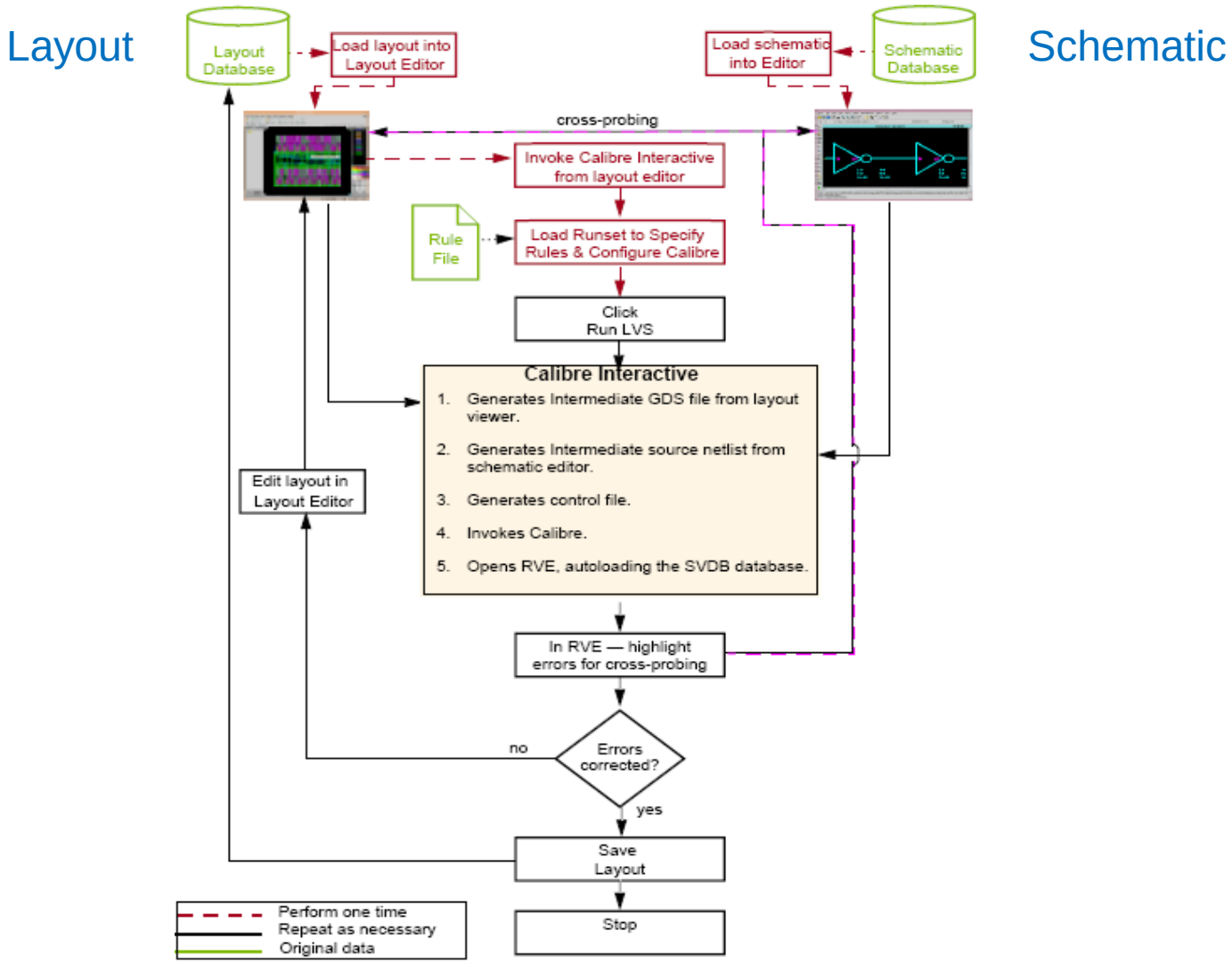
Symbol view of “modulo6”



Verify correctness of layout

- Open layout in Virtuoso
- Verify with Calibre or Assura tools
  1. LVS (layout vs. schematic)
     - Extract netlist from layout
     - Compare extracted netlist to imported netlist
  2. DRC (design rule check)
     - Checks all layout levels
     - Errors should be fixed as appropriate
  3. PEX (parameter extraction)
     - Extract netlist from layout, including R/C parameters
     - Simulate netlist to verify functionality and timing

Calibre Layout-vs-Schematic (LVS) Check



Layout vs schematic check
(Calibre Interactive LVS)

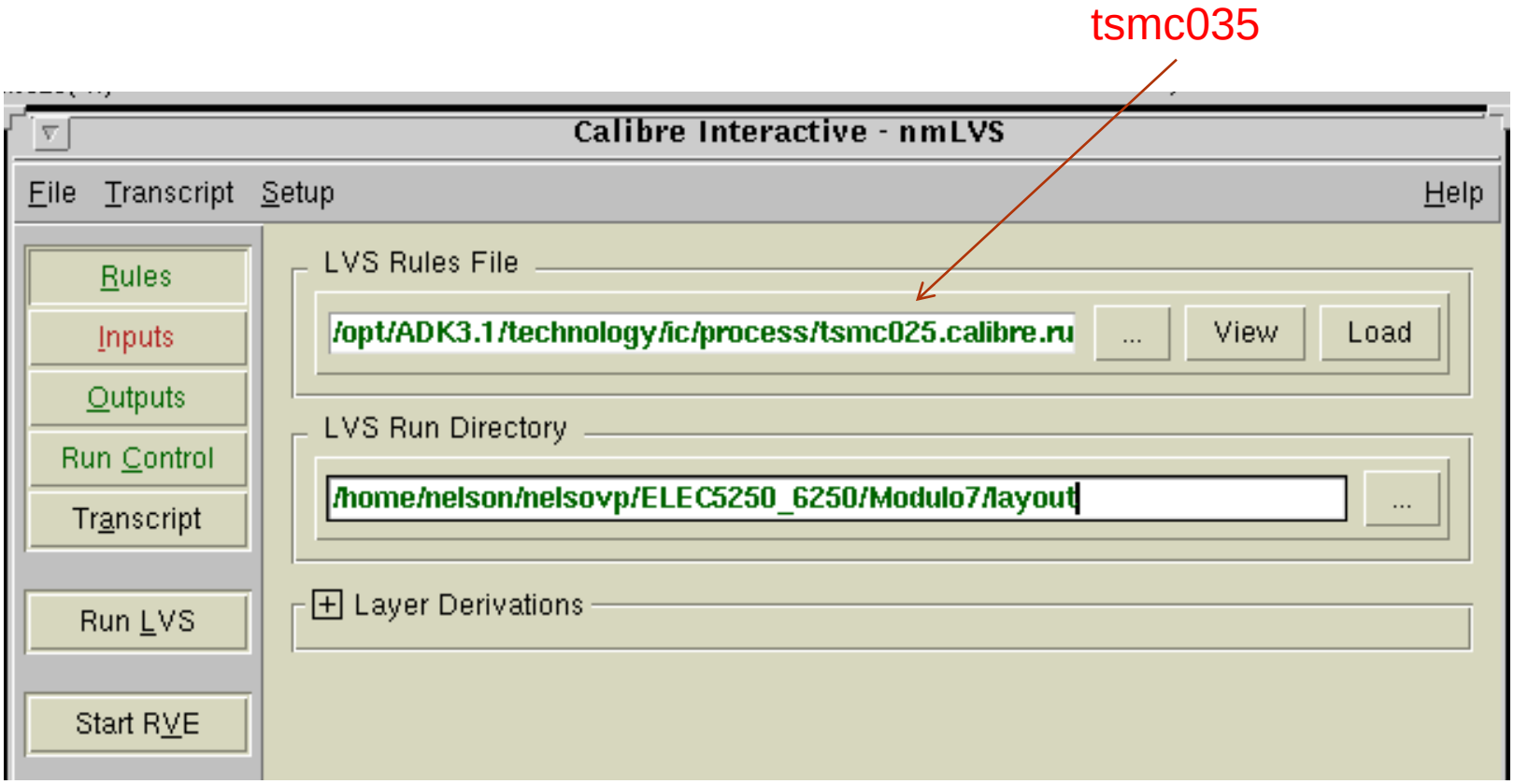
- Compares extracted transistor-level netlist vs. netlist generated from Verilog gate-level netlist
- From Layout GXL menu: *Calibre > Run LVS*

(Demonstrate)

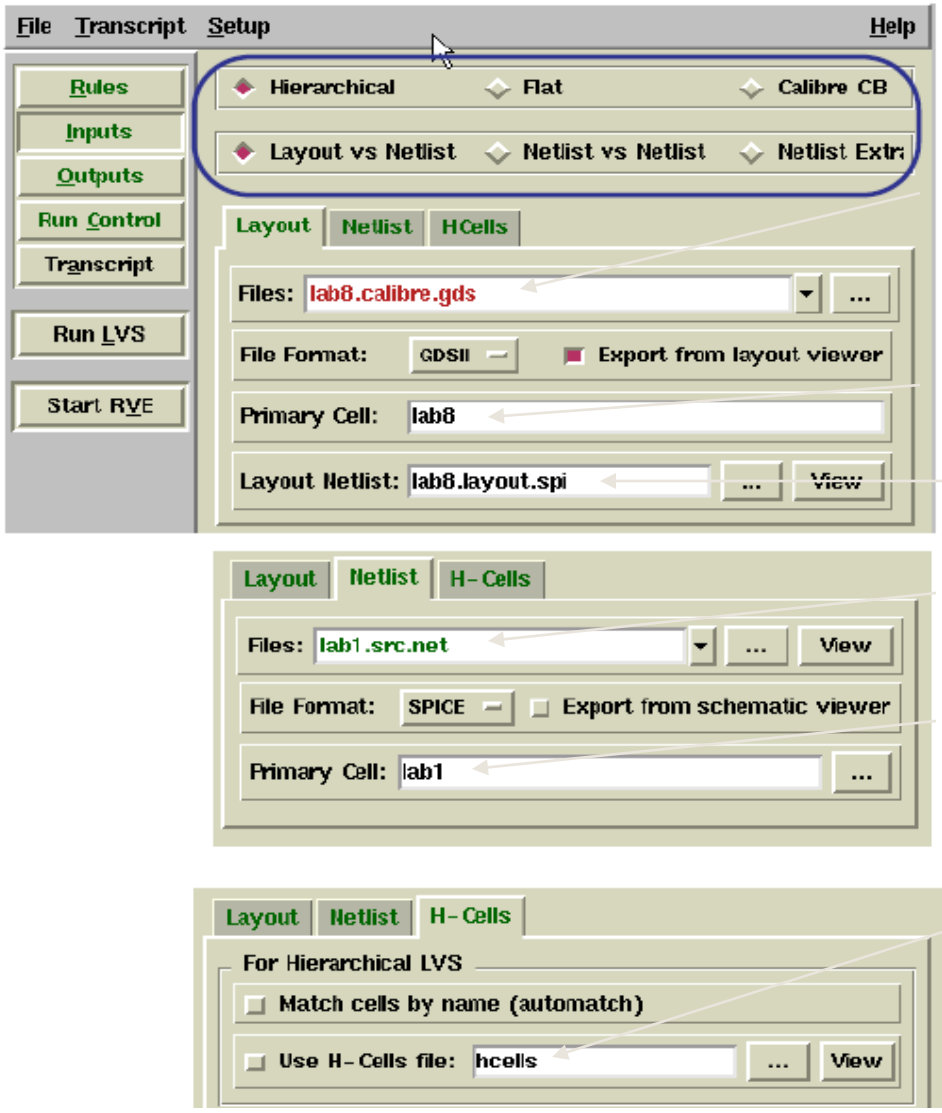
Mentor Graphics LVS

- Rules: *$ADK/technology/ic/process/tsmc035.calibre.rules*
- Inputs/Layout: *will be generated by Calibre*
- Inputs/Netlist: *count4.src.net* (created in DA-IC)
  Top-level cell: *count4* (schematic name)
- Inputs/H-cells (hierarchical cells): *$ADK/technology/adk.hcell*
- Extracted file: *count4.lay.net*

Load rules file



Calibre inputs



Layout to be extracted by Calibre (GDSII format)

Layout top cell name

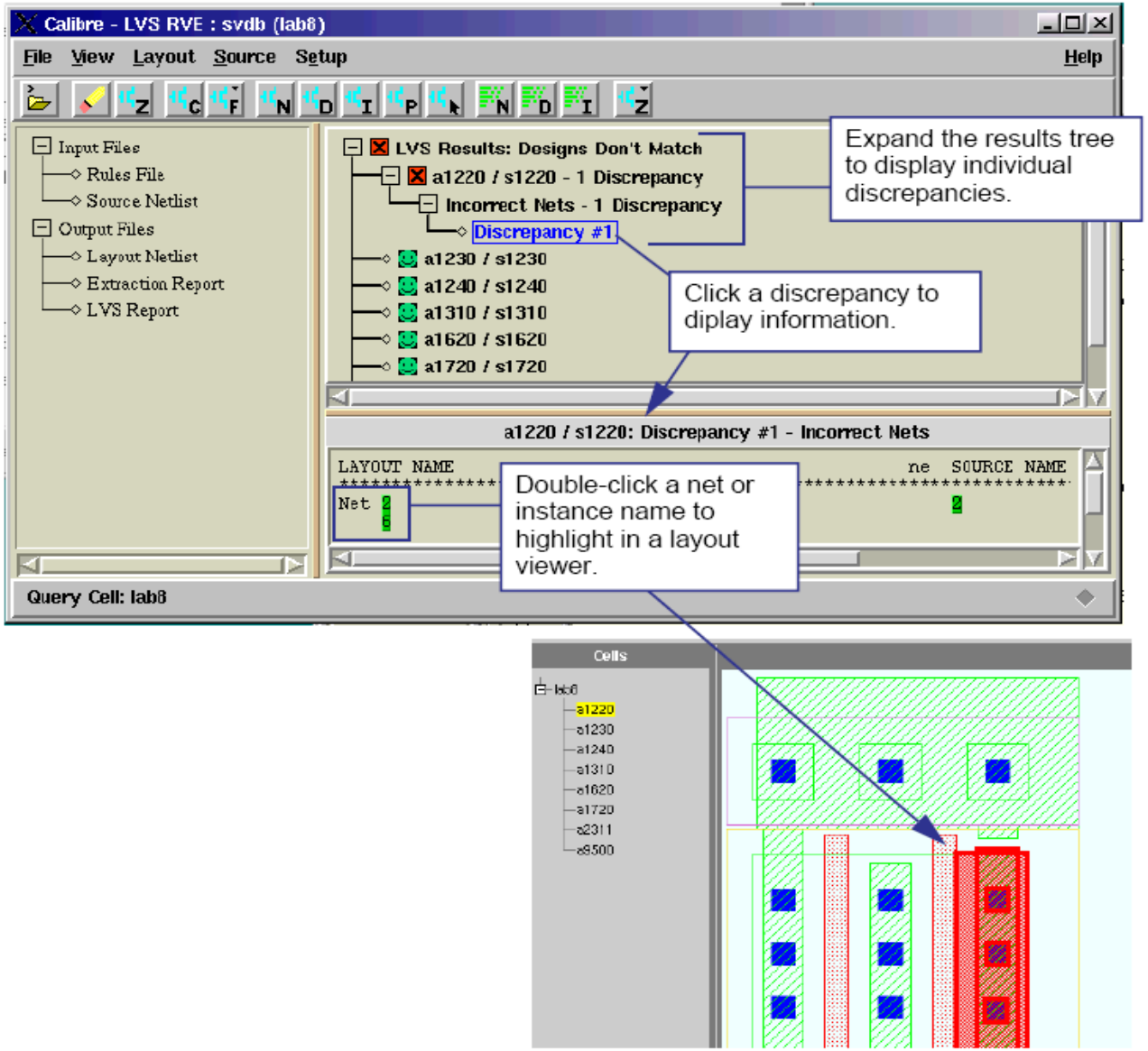
Extracted layout netlist

Source netlist created in DA-IC

Schematic name

Hierarchical cells file:
$ADK/technology/adk.hcell

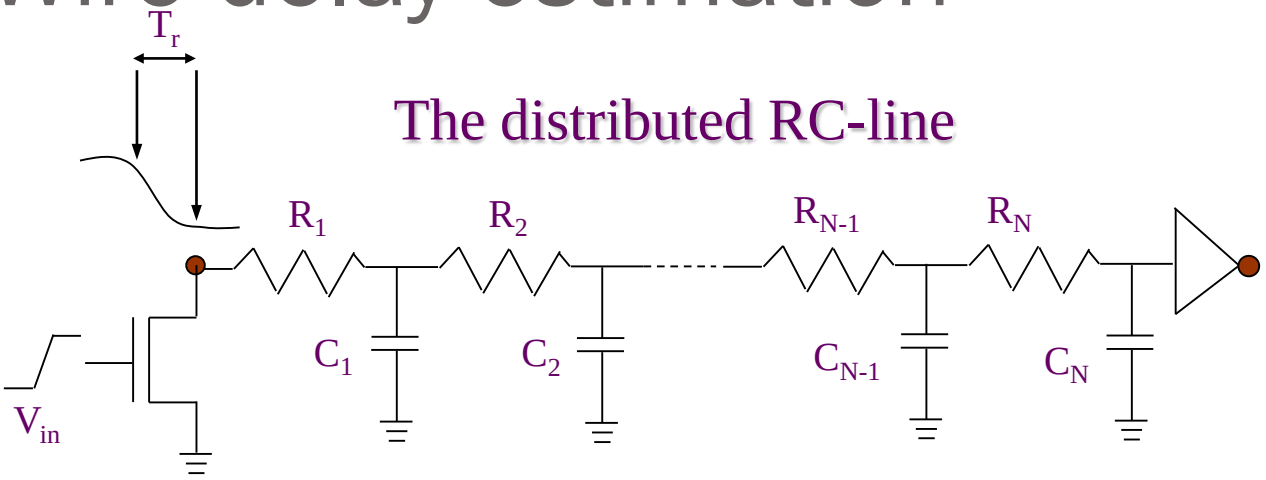
Calibre RVE to probe LVS results



Post-layout functional/timing verification (Calibre PEX)

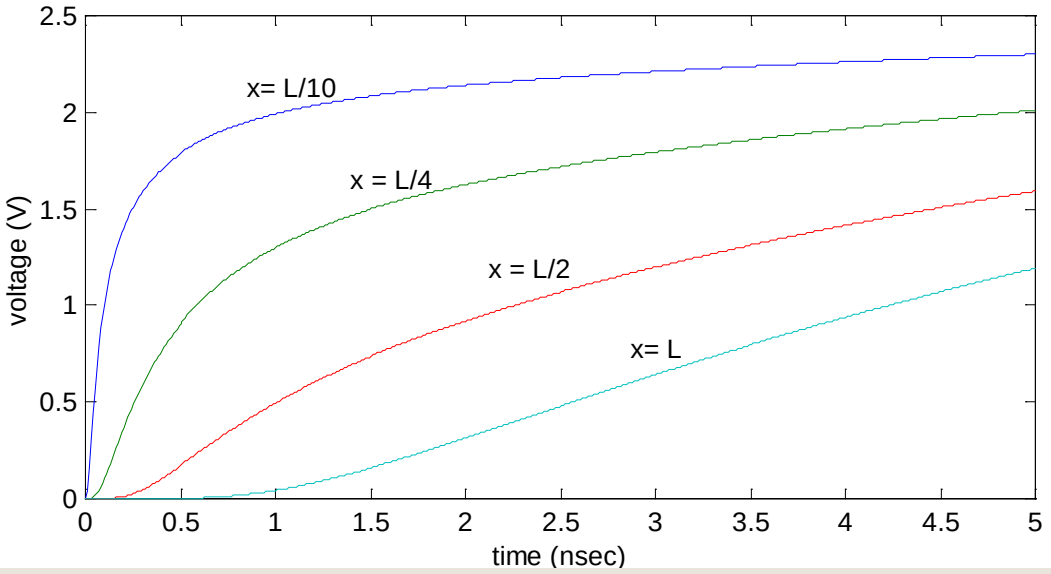
- Purpose: timing analysis & functional verification of the final design
  - analyze netlist extracted from layout
    - parasitic wire capacitance
    - parasitic wire to wire capacitance
    - net and via resistance
  - perform netlist & parameter extraction with *Calibre PEX*
  - simulate in *ADiT*, *Eldo, Spectre, PSPICE, HSPICE, etc.*

Wire delay estimation

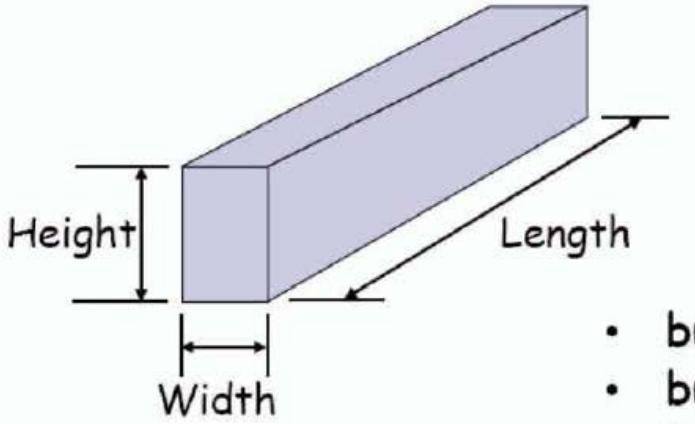


Diffused signal propagation

Delay ~ L^2



Wire resistance



$$\text{resistance} = \frac{(\text{length} \times \text{resistivity})}{(\text{height} \times \text{width})}$$

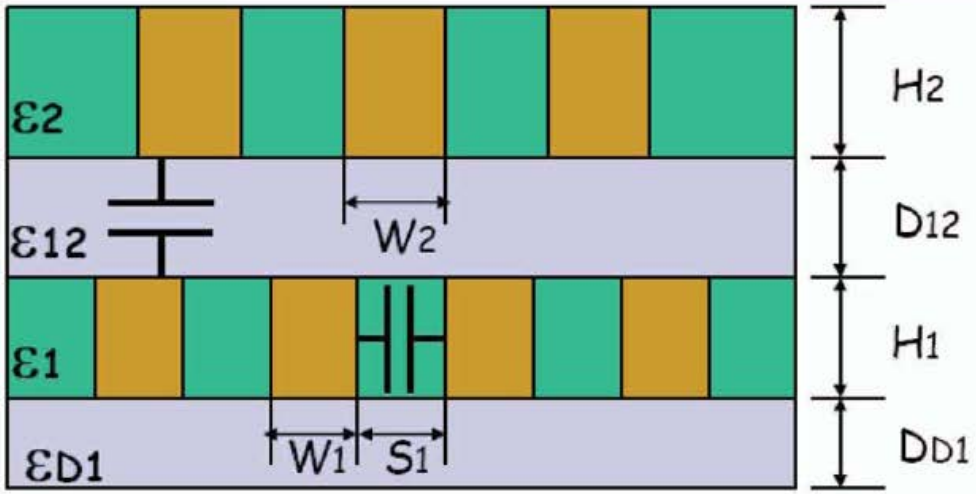
- bulk aluminum $2.8 \times 10^{-8}\ \Omega$-m
- bulk copper $1.7 \times 10^{-8}\ \Omega$-m
- bulk silver $1.6 \times 10^{-8}\ \Omega$-m

- Height (Thickness) fixed in given manufacturing process
- Resistances quoted as Ω/square
- TSMC 0.18μm 6 Aluminum metal layers
  - M1-5 0.08 Ω/square (0.5 μm x 1mm wire = 160 Ω)
  - M6 0.03 Ω/square (0.5 μm x 1mm wire = 60 Ω)

Courtesy of Arvind and Krste Asanovic. Used with permission.

Cite as: Vladimir Stojanovic, course materials for 6.973 Communication System Design, Spring 2006.
MIT OpenCourseWare (http://ocw.mit.edu/), Massachusetts Institute of Technology.
Downloaded on [DD Month YYYY].

Wire capacitance



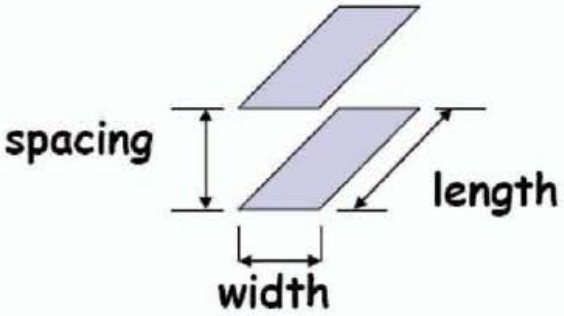
- Capacitance depends on geometry of surrounding wires and relative permittivity, ε_r, of insulating dielectric
  - silicon dioxide, SiO_2 $\varepsilon_r = 3.9$
  - silicon flouride, SiOF $\varepsilon_r = 3.1$
  - SiLK™ polymer, $\varepsilon_r = 2.6$
- Can have different materials between wires and between layers, and also different materials on higher layers

Courtesy of Arvind and Krste Asanovic. Used with permission.

Cite as: Vladimir Stojanovic, course materials for 6.973 Communication System Design, Spring 2006.
MIT OpenCourseWare (http://ocw.mit.edu/), Massachusetts Institute of Technology.
Downloaded on [DD Month YYYY].

Capacitance scaling

$$\text{parallel plate capacitance} \propto \frac{\text{width}}{\text{spacing}} \times \text{length}$$

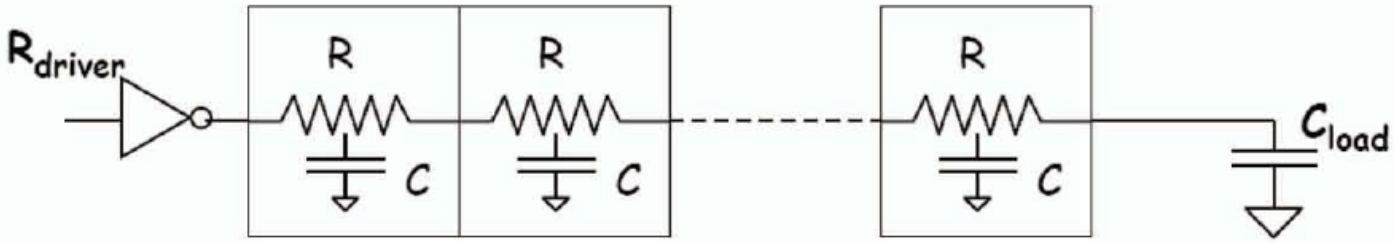


- Capacitance/unit length ~constant with feature size scaling (width and spacing scale together)
  - Isolated wire sees approx. 100 fF/mm
  - With close neighbors about 160 fF/mm
- Need to use capacitance extractor to get accurate values

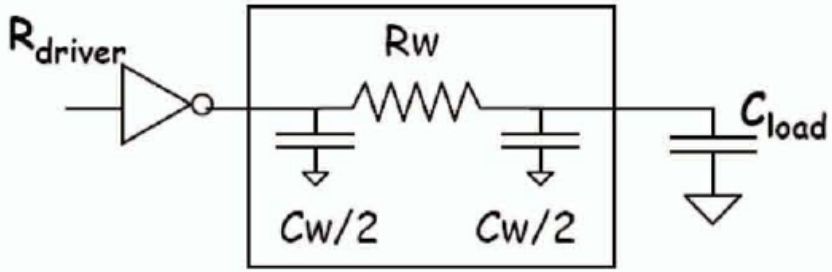
Courtesy of Arvind and Krste Asanovic. Used with permission.

Cite as: Vladimir Stojanovic, course materials for 6.973 Communication System Design, Spring 2006.
MIT OpenCourseWare (http://ocw.mit.edu/), Massachusetts Institute of Technology.
Downloaded on [DD Month YYYY].

Wire delay models



- Wire has distributed R and C per unit length
  - wire delay increases quadratically with length
  - edge rate also degrades quadratically with length
- Simple lumped Π model gives reasonable approximation
  - Rw is lumped resistance of wire
  - Cw is lumped capacitance (put half at each end)



$$\text{Delay} = \text{Rdriver} \times \frac{\text{Cw}}{2} + (\text{Rdriver} + \text{Rw}) \times \left(\frac{\text{Cw}}{2} + \text{Cload} \right)$$

Courtesy of Arvind and Krste Asanovic. Used with permission.

Cite as: Vladimir Stojanovic, course materials for 6.973 Communication System Design, Spring 2006.
MIT OpenCourseWare (http://ocw.mit.edu/), Massachusetts Institute of Technology.
Downloaded on [DD Month YYYY].

Wire delay example

- In 0.18μm TSMC, 5x minimum inverter with effective resistance of 3 kΩ, driving FO4 load (25fF)
- Delay = Rdriver x Cload = 75 ps

- Now add 1mm M1 wire, 0.25μm wide
  - Rw = 320Ω wire + 22Ω vias = 344Ω
  - Cw = 160fF

$$\text{Delay} = \text{Rdriver} \times \frac{Cw}{2} + (\text{Rdriver} + \text{Rw}) \times \left(\frac{Cw}{2} + \text{Cload} \right)$$

$$= 3\text{k}\Omega \times 80\text{fF} + (3\text{k}\Omega + 344\Omega) \times (80\text{fF} + 25\text{fF})$$

$$= 591\text{ps}$$

Courtesy of Arvind and Krste Asanovic. Used with permission.

Cite as: Vladimir Stojanovic, course materials for 6.973 Communication System Design, Spring 2006.
MIT OpenCourseWare (http://ocw.mit.edu/), Massachusetts Institute of Technology.
Downloaded on [DD Month YYYY].

Parameter extraction with Calibre PEX

- Extract SPICE netlist, including parasitic RC
  - *Transistor-level*, gate-level, or hierarchical extraction
- With the layout cell open:
  - In the menu bar: *Calibre>Run PEX*
  - Input options: *similar to Calibre LVS*
  - Extraction options (Outputs tab):
    - *choose "Transistor level"*
    - *choose one of:*
      - *C: lumped C + coupling cap's*
      - *RC: distributed RC*
      - *RCC: distributed RC + coupling cap's*
  - Click "Run PEX"
  - Output files: modulo5.sp - main SPICE model (transistors)
    modulo5.sp.pex - extracted R/C (lumped)
    modulo5.sp.MODULO5.pxi - extracted C (coupling)

Extracted file – top level

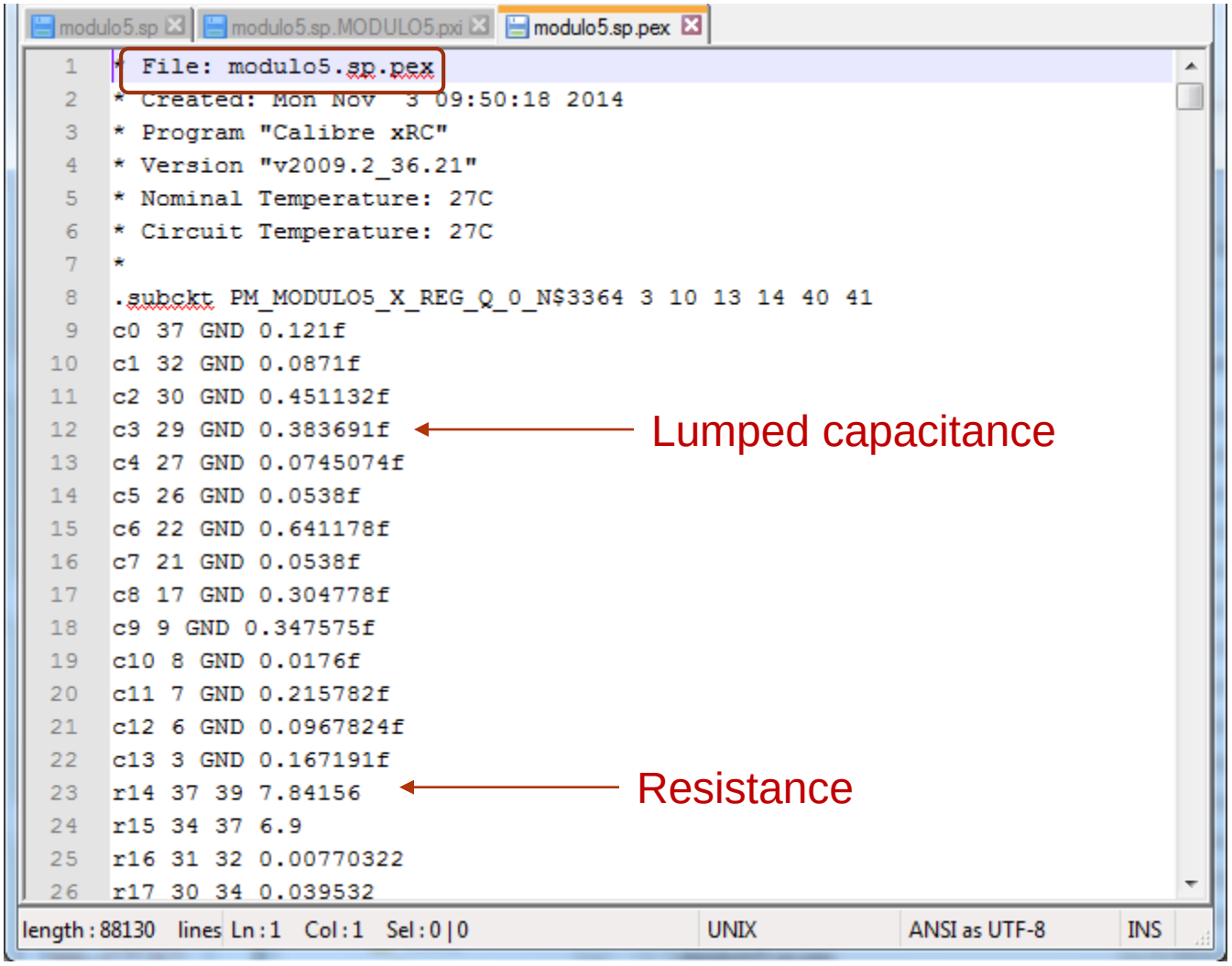
```
1  * File: modulo5.sp
2  * Created: Mon Nov  3 09:50:18 2014
3  * Program "Calibre xRC"
4  * Version "v2009.2_36.21"
5  *
6  .include "modulo5.sp.pex"
7  .global GND VDD
8  .subckt modulo5  I[0] I[1] I[2] Q[0] Q[1] Q[2] L_CBAR CLK CLEARBAR
9  *
10 * VDD   VDD
11 * GND   GND
12 * L_CBAR    L_CBAR
13 * Q[2]  Q[2]
14 * CLK   CLK
15 * Q[1]  Q[1]
16 * CLEARBAR  CLEARBAR
17 * I[0]  I[0]
18 * Q[0]  Q[0]
19 * I[2]  I[2]
20 * I[1]  I[1]
21 mX_IX3_M_I$212 N_NX2_X_ix3_M_I$212_d N_X_IX3_N$5_X_ix3_M_I$212_g
22 + N_GND_X_ix3_M_I$212_s N_GND_X_reg_Q_0_M_I$1901_b N L=4e-07 W=1e-06
23 mX_IX3_M_I$211 N_NX2_X_ix3_M_I$211_d N_X_IX3_N$5_X_ix3_M_I$211_g
24 + N_VDD_X_ix3_M_I$211_s N_VDD_X_reg_Q_1_M_I$1899_b P L=4e-07 W=1.8e-06
25 mX_IX3_M_I$5 N_X_IX3_N$5_X_ix3_M_I$5_d N_Q[0]_X_ix3_M_I$5_g
   N_GND_X_ix3_M_I$5_s
26 + N_GND_X_reg_Q_0_M_I$1901_b N L=4e-07 W=1e-06
27 mX_IX3_M_I$4 N_X_IX3_N$5_X_ix3_M_I$4_d N_Q[1]_X_ix3_M_I$4_g
   N_GND_X_ix3_M_I$4_s
28 + N_GND_X_reg_Q_0_M_I$1901_b N L=4e-07 W=1e-06
29 mX_IX3_M_I$3 N_X_IX3_N$5_X_ix3_M_I$3_d N_Q[1]_X_ix3_M_I$3_g
30 + N_X_IX3_N$1_X_ix3_M_I$3_s N_VDD_X_reg_Q_1_M_I$1899_b P L=4e-07 W=2.6e-06
31 mX_IX3_M_I$2 N_X_IX3_N$1_X_ix3_M_I$2_d N_Q[0]_X_ix3_M_I$2_g
```

Include extracted R/C

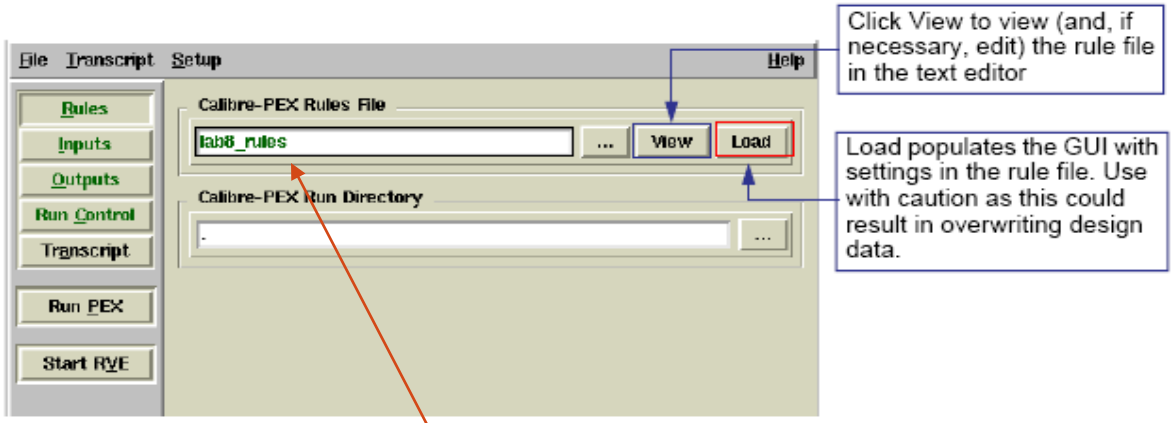
N transistor

source drain bulk gate

Extracted file – extracted R/C



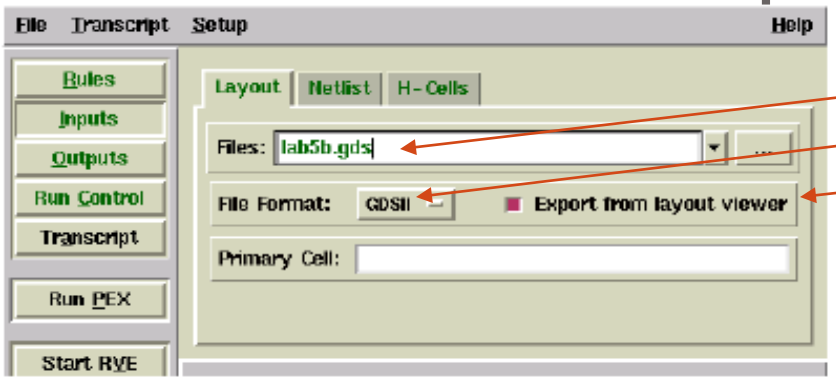
Calibre PEX inputs



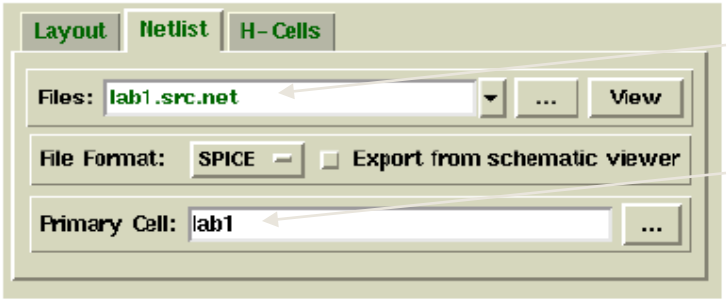
Specify rules file:

$ADK/technology/ic/process/tsmc035.calibre.rules

Calibre PEX inputs

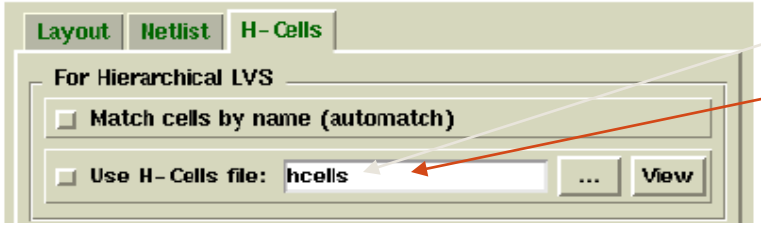


Name of layout file (count4.gds)
GDSII file format
Check to generate new layout file
Name of top cell (count4)



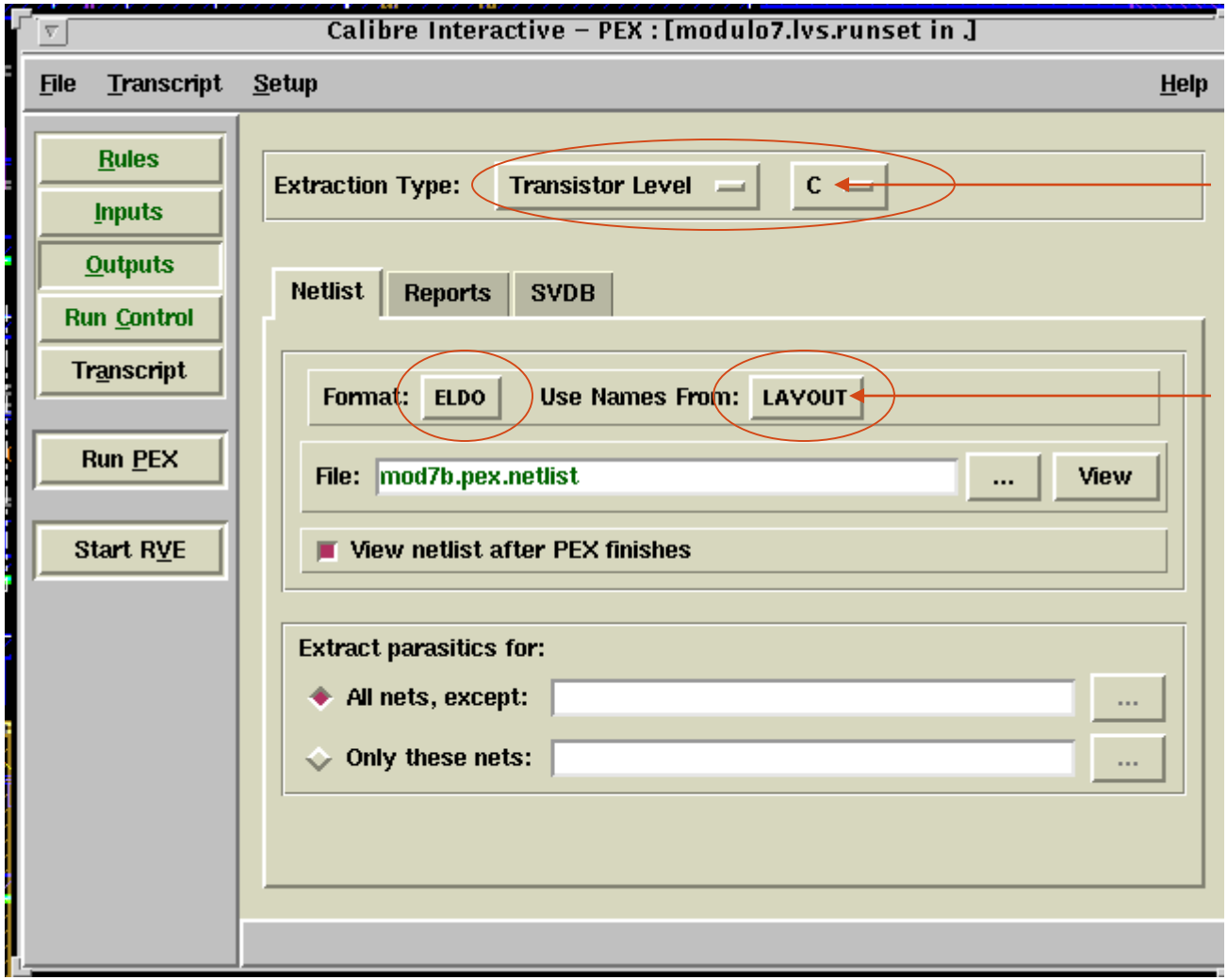
Source(SPICE) netlist created in DA-IC

Top-level cell name in SPICE netlist



Hierarchical cells file:
$ADK/technology/adk.hcell

Calibre PEX netlist output



Lumped capacitance

Use net names from LAYOUT

Designate GND and VDD nets