

# ***SILICONCHIP TECHNOLOGIES***

P. D. A College Of Engineering, Aiwan-E -  
Shahi Area, Kalaburagi



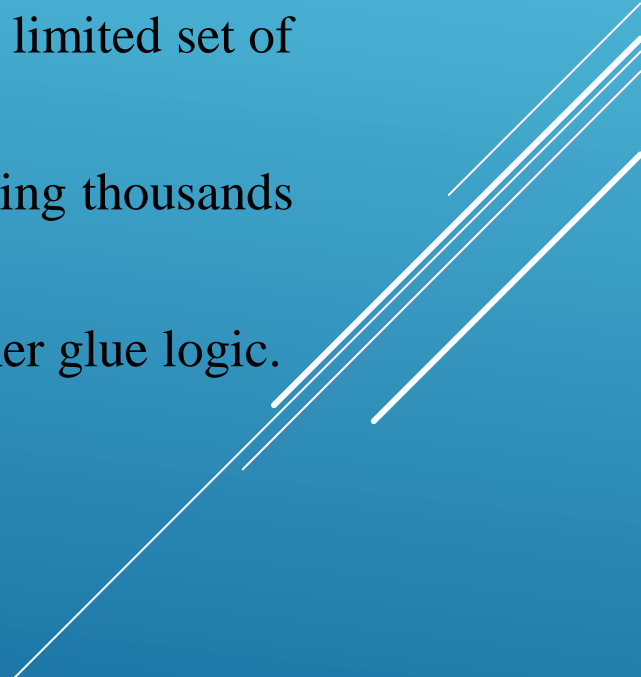
# Contents

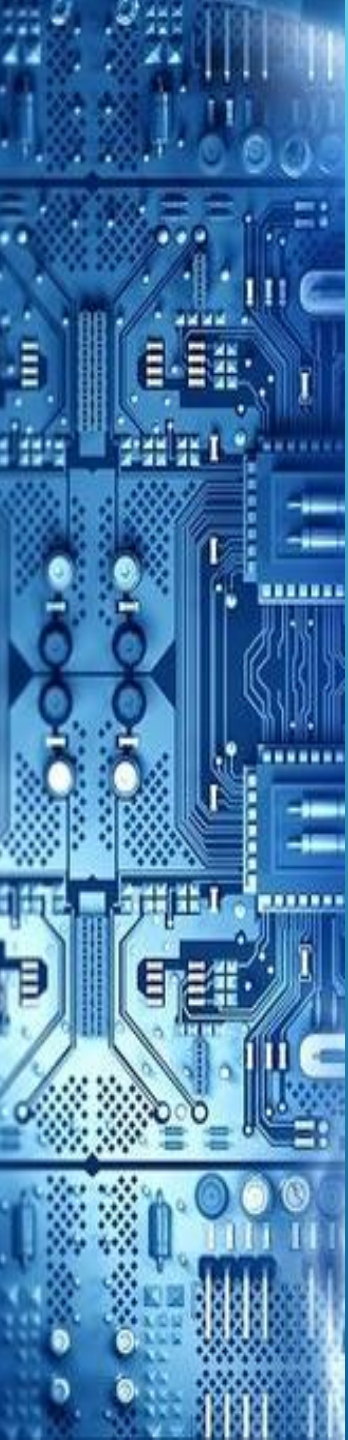
- ▶ Introduction to VLSI
  - ▶ Why VLSI
  - ▶ About Semiconductor Industry
  - ▶ Growth of Semiconductor
  - ▶ About SiliconChip Technologies
  - ▶ Why SiliconChip Technologies ....?
  - ▶ Career Growth
  - ▶ Placement Details
  - ▶ Contact details
- 





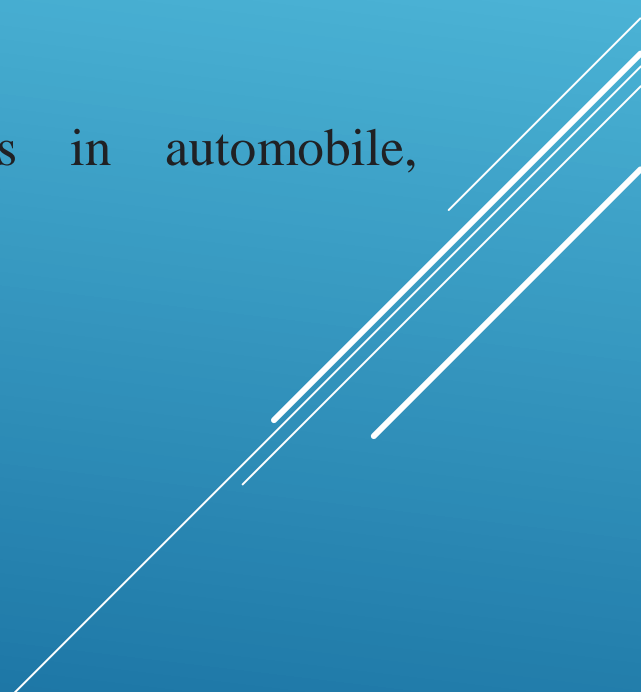
# INTRODUCTION TO VLSI

- ▶ Very large scale integration
  - ▶ VLSI began in the 1970s when complex semiconductor and communication technologies were being developed
  - ▶ Before the introduction of VLSI technology, most ICs had a limited set of functions they could perform.
  - ▶ It is process of creating an integrated circuit (IC) by combining thousands of transistors into a single chip
  - ▶ An electronic circuit consist of a CPU, ROM, RAM and other glue logic.
- 



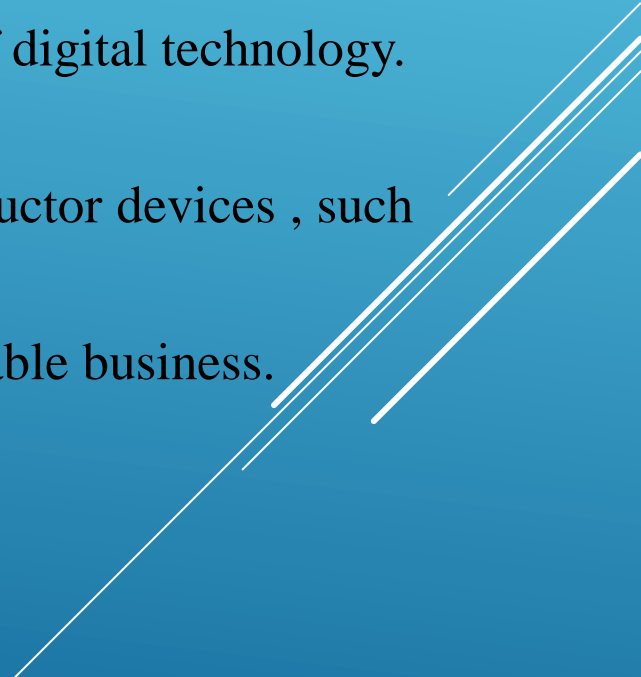
## WHY VLSI

- ▶ VLSI engineers carve the future of automation and industry 4.0.
- ▶ VLSI engineers are responsible for designing the latest and greatest Graphics Processor Units (GPU) which are the backbone for advancements in AI and many other domains.
- ▶ It is a lucrative field that is resilient to automation.
- ▶ There is a huge demand for VLSI design jobs in automobile, telecommunication, mechanical, and so on.





# SEMICONDUCTOR INDUSTRY

- ▶ Semiconductors are materials which have a conductivity between conductors (generally metals) and nonconductors or insulators
  - ▶ The global semiconductor industry is dominated by companies from the United States , Taiwan, South Korea, Japan and the Netherlands.
  - ▶ The semiconductors industry is a key factor in the future of digital technology.
  - ▶ The industry is the aggregate of companies engaged in the design and fabrication of semiconductor and semiconductor devices , such as transistors and integrated circuits.
  - ▶ It formed around 1960, once the fabrication of became a viable business.
- 





# GROWTH OF SEMICONDUCTOR

- Joe Biden, the US President, allocated **\$50 billion** in 2021 for semiconductors.
- Global semiconductor sales surpassed **\$466 billion** in 2020.
- In 2021, the semiconductor industry sold a record **1.15 trillion** semiconductor chips.
- Samsung won the global semiconductor sales crown, with sales of roughly **\$83.09 billion**.
- **70%** of growth in the semiconductor sector is anticipated to be driven by just **3 industries**.
- Sales of semiconductors worldwide increased by **23%** between **Q1 of 2021 to Q1 of 2022**.
- The global sector generated **\$552.9 billion** in revenue in 2021. In 2022, it's predicted to reach **\$600 billion**.
- **Samsung** is the top semiconductor company.
- According to projections, the market would expand at a **CAGR of 9.2%** through the year **2029**.

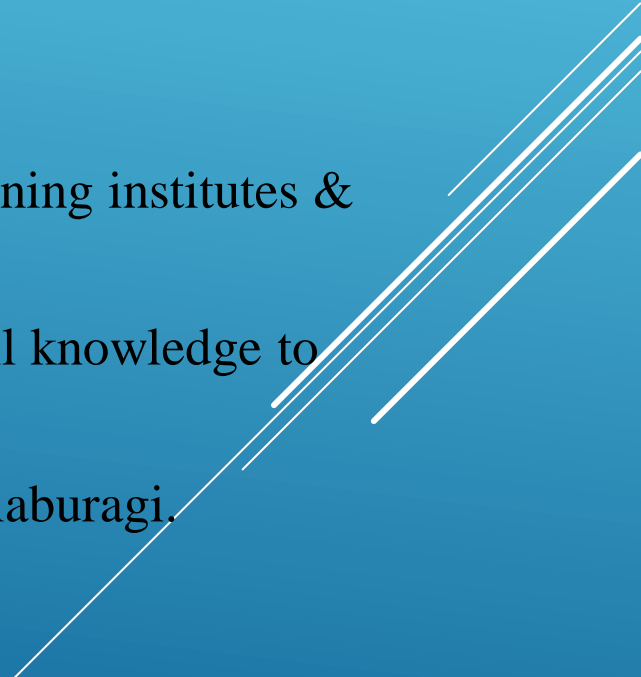


# ABOUT SILICONCHIP TECHNOLOGIES

- SiliconChip Technologies is Known as Best VLSI Training Institute in Bangalore & Kalaburagi offers Industry's best VLSI training curriculum, covering all aspects of Training program.
  - Our Training Institute was set up in 2018, offers industry standard, high quality, affordable training to graduates.
  - Institute is among the very few institute offering quality training in complete spectrum of VLSI flow from RTL design, Functional Verification, Formal Verification, GLS, Synthesis, STA, Physical Design, DFT, Custom Layout and Physical Verification. We also offer courses on AMBA, PCIe, USB, Low power verification and SOC verification, customized for experienced engineers. want to make career in VLSI, and Embedded systems.
  - 100% job oriented VLSI training courses
- 




# WHY SILICONCHIP TECHNOLOGIES

- ▶ We are a startup with energetic working professionals having 10+ years of experience who are dedicated to train the students to excel in their careers.
  - ▶ Offline classes with hands-on lab session with VLSI tools.
  - ▶ Assignments & Projects are assigned to students to read the skill development on weekly basis.
  - ▶ 1:1 mentoring sessions are held by Industry experts.
  - ▶ Courses provided are budget friendly compared to other training institutes & 100% job determined.
  - ▶ Training process is designed to provide technical & soft skill knowledge to have a better job understanding.
  - ▶ We have our training institute in Bangalore and Namma Kalaburagi.
- 





# CAREER GROWTH

- ▶ There is lot of opportunities after completion of VLSI Program. Some of the job opportunities in this domain are Verification Engineer, Design Engineer, Application Engineer, CAD Engineer, etc.
  - ▶ With VLSI training, learners can give their career a new growth. How does VLSI work? Very large-scale integration is the process of making a microcircuit by combining many MOS transistors onto one chip.
  - ▶ It is a microcircuit chips widely adopted, enabling complex semiconductor and telecommunication technologies to be developed.
  - ▶ How much does a VLSI engineer make in India? The average VLSI engineer salary in India is ₹ 900,000 per year or ₹ 361 per hour. Entry-level positions start at ₹ 363,750 per year, while most experienced workers make up to ₹ 2,500,000 per year.
- 

# PLACEMENTS WITH :

**EXINIUS**  
Excellence Everyday



**GRAPHENE**  
Semiconductors



**Google**  
**cādence**<sup>®</sup>

**MMRFI**  
technology pvt ltd  
The "Antenna to Bits" Solutions Company

**HCL**



**SEVITECH**

**intel**<sup>®</sup>

**cerium**  
SYSTEMS  
A Tech Mahindra Company

**Capgemini**

**terminus circuits**  
innovative solutions

**Qualcomm**



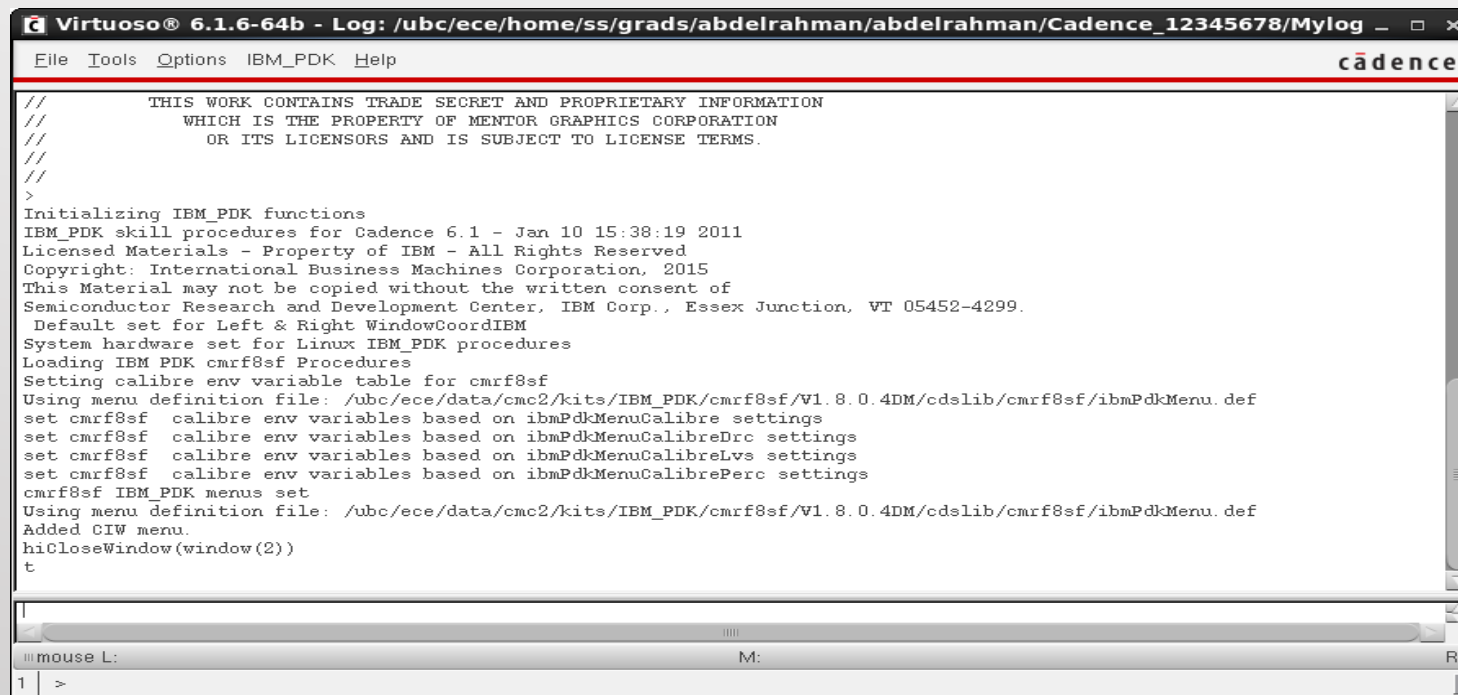
# Cadence Virtuoso – Layout – Inverter (45nm)

- Abdelrahman H. Ahmed. 9/2015 ~
- Virtuoso is a schematic and layout editor software from Cadence.
  - **Environment Setup and starting Cadence Virtuoso**  
The objective of this section is to learn how to get the environment ready for the tool, take care of the licensing issues, and start the tool.
  - **Virtuoso working Directory**  
In your Cadence tools directory, created in “RTL Compiler tutorial” section 1, descend into a folder called “cds”. This folder will be the working directory for the Cadence Virtuoso.

# Source the setup file and run Cadence

- In the working directory source the provided Setup file. Sourcing this file will take care of all the needed environment variables, and all the licensing as well. After sourcing the setup file, launch the tool.

After running the previous lines Cadence should open its main window as in Figure 1, also known as **Command Interpreter Window (CIW)**. Read the log in that window to make sure that everything went well with no errors or warnings.



```
Virtuoso® 6.1.6-64b - Log: /ubc/ece/home/ss/grads/abdelrahman/abdelrahman/Cadence_12345678/Mylog _ □ ×
File Tools Options IBM_PDK Help cadence

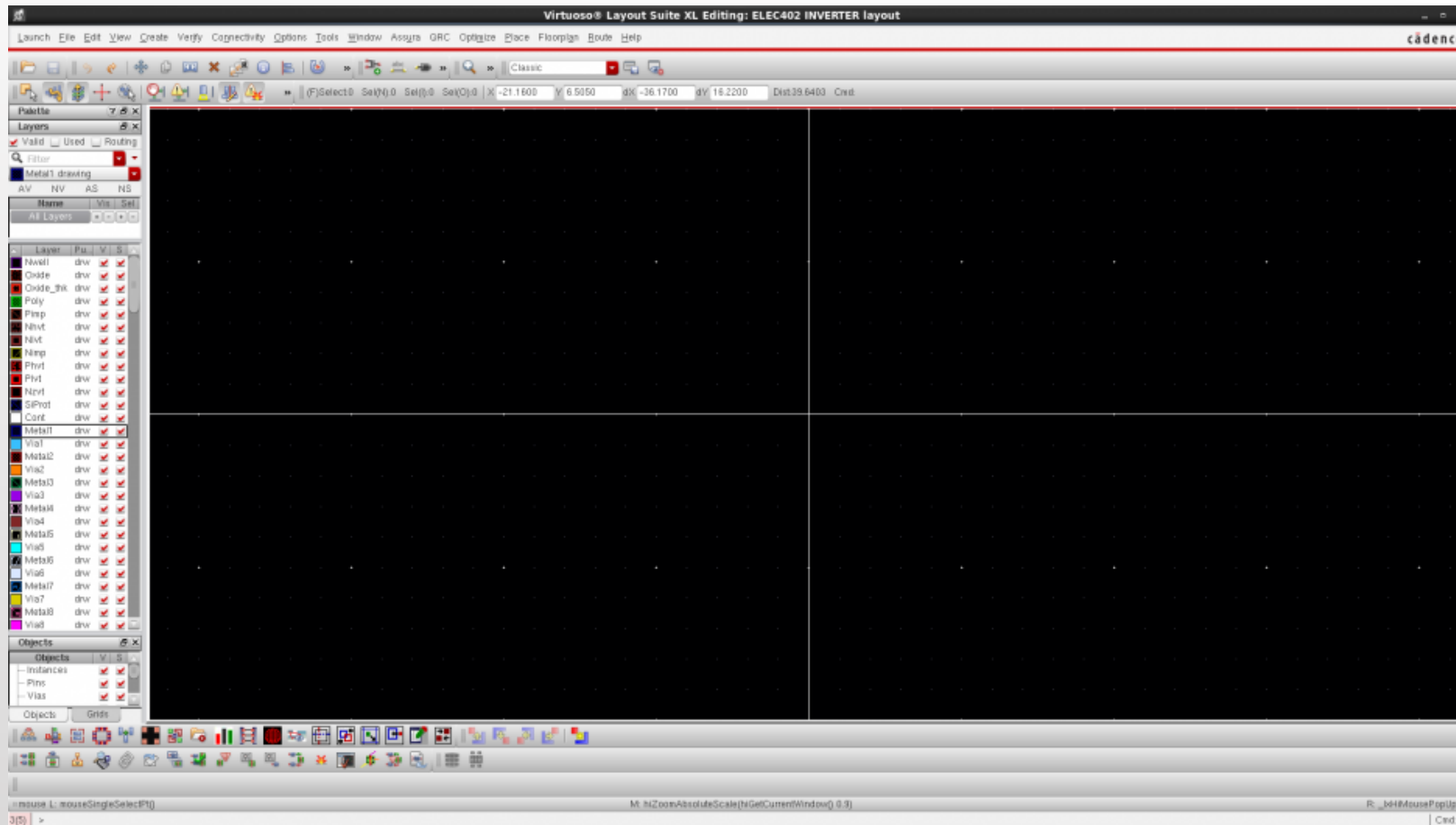
//      THIS WORK CONTAINS TRADE SECRET AND PROPRIETARY INFORMATION
//      WHICH IS THE PROPERTY OF MENTOR GRAPHICS CORPORATION
//      OR ITS LICENSORS AND IS SUBJECT TO LICENSE TERMS.
//
//
>
Initializing IBM_PDK functions
IBM_PDK skill procedures for Cadence 6.1 - Jan 10 15:38:19 2011
Licensed Materials - Property of IBM - All Rights Reserved
Copyright: International Business Machines Corporation, 2015
This Material may not be copied without the written consent of
Semiconductor Research and Development Center, IBM Corp., Essex Junction, VT 05452-4299.
Default set for Left & Right WindowCoordIBM
System hardware set for Linux IBM_PDK procedures
Loading IBM_PDK cmrf8sf Procedures
Setting calibre env variable table for cmrf8sf
Using menu definition file: /ubc/ece/data/cmc2/kits/IBM_PDK/cmrf8sf/V1.8.0.4DM/cdslib/cmrf8sf/ibmPdkMenu.def
set cmrf8sf calibre env variables based on ibmPdkMenuCalibre settings
set cmrf8sf calibre env variables based on ibmPdkMenuCalibreDrc settings
set cmrf8sf calibre env variables based on ibmPdkMenuCalibreLvs settings
set cmrf8sf calibre env variables based on ibmPdkMenuCalibrePerc settings
cmrf8sf IBM_PDK menus set
Using menu definition file: /ubc/ece/data/cmc2/kits/IBM_PDK/cmrf8sf/V1.8.0.4DM/cdslib/cmrf8sf/ibmPdkMenu.def
Added CIW menu.
hiCloseWindow(window(2))
t

|
mouse L: M: R:
1 >
```

Figure 1 Cadence Virtuoso's CIW.



# Starting Virtuoso layout editor



To modify the display and the snap options go to <**Options -> Display**>. Select your preferred options and keep in mind that your minimum snap spacing should match that of the used PDK to avoid **Design Rule Check (DRC)** errors. [Should be multiple of 0.005 for this GPDK]

# Cell layout

- **3.1 Generate used devices from schematic**

In this step you will be generating the layout of sub-cells used in building your cell. In the case of an inverter, the generated sub-cells are 1 NMOS, 1 PMOS, and 4 IO pins. You can instantiate these sub-cells like what you did before in the schematic editor. Another way is to ask virtuoso's assistance in generating the sub-cells. In the layout editor, go to <**Connectivity -> Generate -> All From Source**>. The “Generate Layout” window will open. Please make sure to select the options as shown in Figure 3, and press OK.



Note that the sub-cells have been added to the layout editor window as shown in Figure 4 *left*. The devices are shown as black boxes, and to view the layers inside each sub-cell press <Shift+F> as shown in Figure 4 *right*. To go back to boxes view press <Ctrl+F>. Familiarize yourself with the used layers, and use the Palette on your left-hand-side to assist you.



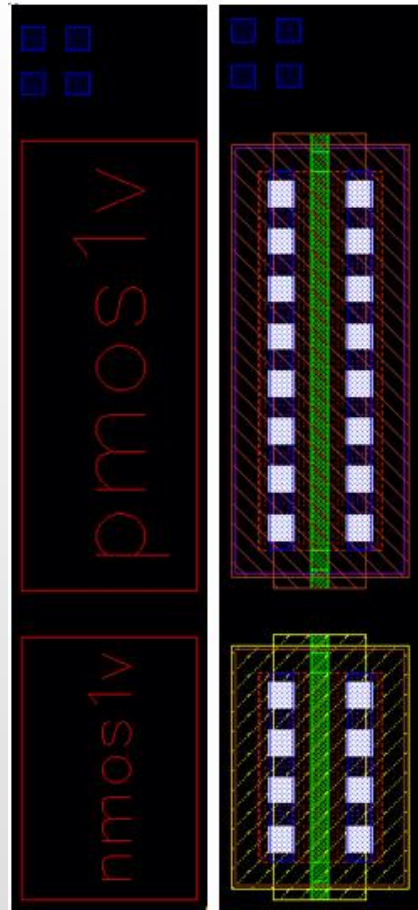


Figure 4 Left (Sub-cells with boxes view). Right (Sub-cells with all layers visible)

## 3.2 Floorplan and route

- Start moving stuff around to match your floorplan. Note that virtuoso assists you by showing virtual connections between nodes as shown in Figure 5. Also, when selecting a sub-cell in the layout editor, note that it will be automatically selected in the schematic editor. To draw a ruler press <K>. To erase all rulers press <Shift+K>.

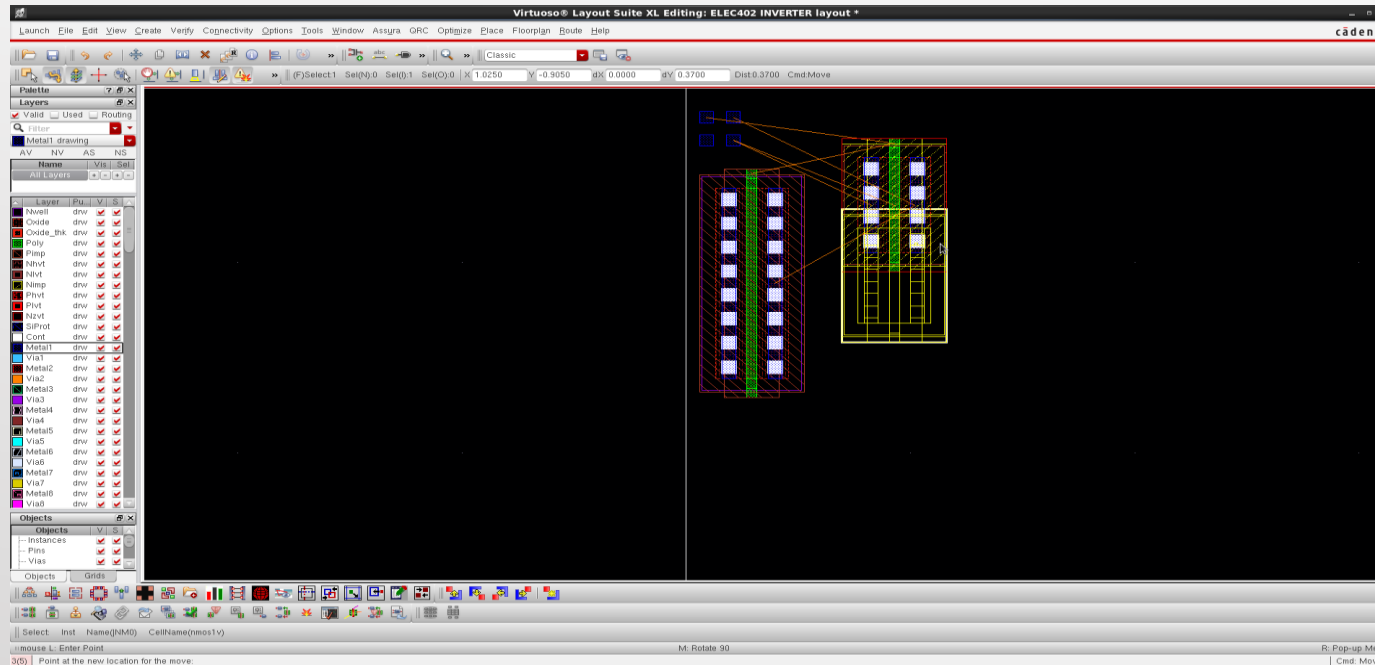


Figure 5 Moving sub-cells

Finalize the placement and the routing according to your project requirements. Keep in mind that you always want to minimize parasitics generated by connections. Figure 6 shows layout of an inverter.



- Notes:

For a connection to switch between two layers a Via has to be used. To create a Via press <O>. Set the needed options, and make sure that the ‘Via Definition’ field corresponds to the needed transition. For example, to switch between the Poly layer and Metal 1 select [M1\_PO].

For reliability considerations, a minimum of two Via cuts should be used at every transitions.

To avoid latch up problems, bulk connections should be used. For NMOS use [*M1\_PIMP Via*] and connect it to GND, and for PMOS use [*M1\_NWELL Via*] and connect it to VDD.

Using “text-drw” layer add labels to all your layout ports as shown in Figure 6

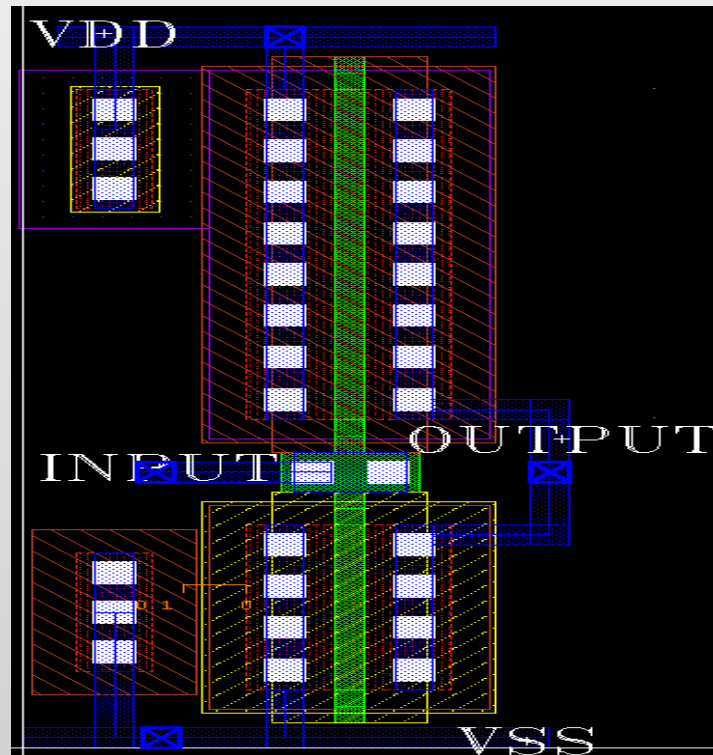


Figure 6 CMOS inverter full layout.

# Verification and checks

In this section you will be checking that your layout satisfies the fabrication constraints through the **Design Rule Check (DRC)**. Also, you will be checking that both the layout and the schematic match through the **Layout Versus Schematic check (LVS)**. For both checks Cadence's **Physical Verification System (PVS)** will be used.

In the layout editor window go to <**Launch -> Plugins -> PVS**> which will add a new menu for PVS.

## **4.1 Design Rule Check (DRC)**

To start the DRC tool go to <**PVS -> Run DRC**>. In the DRC window shown in Figure 7 go to <**File -> Load Presets**> then browse to “virtuoso.drc\_preset” and press **Submit**.

# Design Rule Check(DRC)

- To start the DRC tool go to <**PVS -> Run DRC**>. In the DRC window shown in Figure 7 go to <**File -> Load Presets**> then browse to “virtuoso.drc\_preset” and press **Submit**.

**PVS 14.12-64b DRC Run Submission Form**

File Preferences Toolbars Help

cadence

**Layout**

Library: ELEC402  
Cell: INVERTER  
View: layout

Create GDSII: PVSDRC/INVERTER.gds

Layer Name Table: /ubc/ece/data/cmc2/kits/AMSKIT616\_GPDk/tech/gpdk045\_v\_4\_0/gpdk045/gp...  
Cell Name Table: ...  
Object Name Table: ...  
Label Map Table: ...

Convert Pin to: ☒ Geometry ☐ Text ☐ Geometry+Text ☐ Ignore

Hierarchy Depth Limit: 32 Maximum Vertices In Path/Polygon: 2048

☐ Area to Check on Layout  
☐ Flatten Input Hierarchy  
☒ Abort on Layout Error

Start DRC DE ☒

Cancel Apply Submit

Form is filled according autorun flow setting: /ubc/ece/home/ss/grads/abdelrahman/abdelrahman/Cadence\_12345678/cds/PVSDRC



After running DRC “DRC Debug Environment” window will open Figure 8. In that window you will see all the violations in your layout along with brief explanations. Work on your errors and **ReRun** from the “PVSDRC” window, Figure 9, until you get a clean design with zero errors

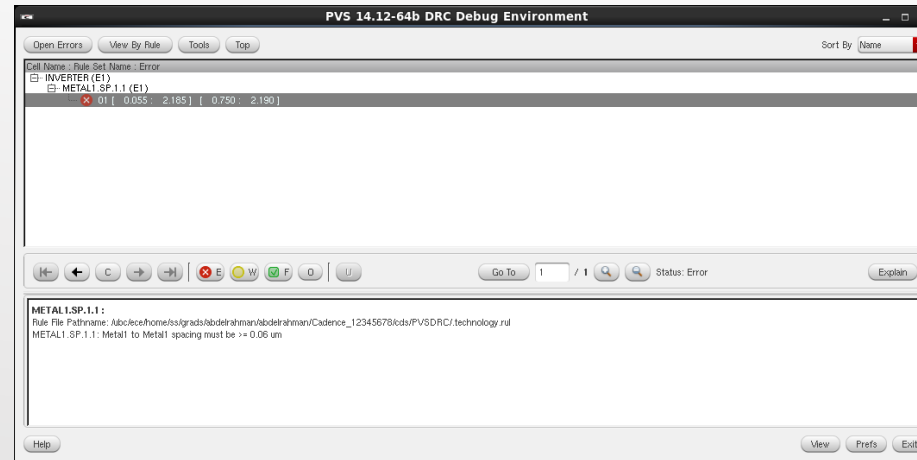


Figure 8 DRC Debug Environment.

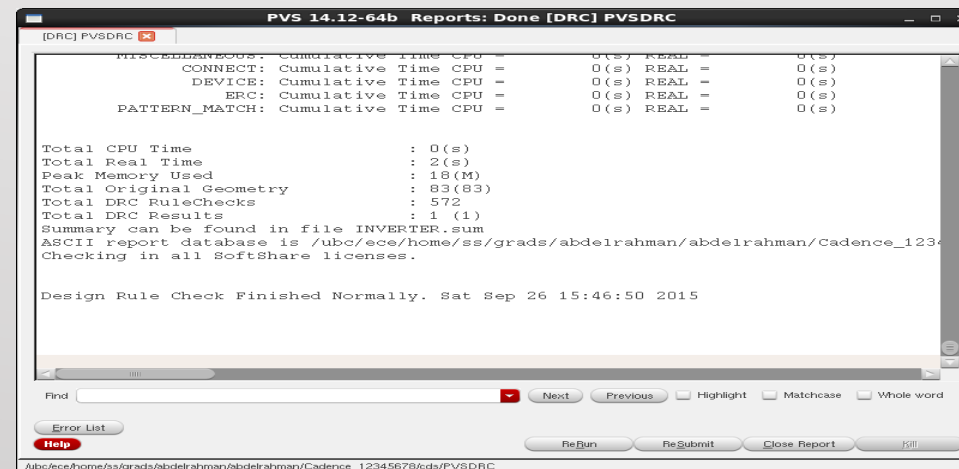


Figure 9 PVSDRC report window.

# Layout Versus Schematic (LVS)

To start the LVS tool go to <**PVS -> Run LVS**>. In the LVS window shown in Figure 10 go to <**File -> Load Presets**> then browse to “virtuoso.lvs\_preset”. Go to Output tab and make sure that “Create Quantus QRC Input Data” is checked, and press **Submit**.

**PVS 14.12-64b LVS Run Submission Form**

File Preferences Toolbars Help

cadence

**Output**

**H-Cell Settings**

☐ Automatch

☐ HCell

☐ GenHier Cells

☒ Run ERC Checks

**ERC Report**

Name: INVERTER.sum

Limit: 1000

☒ Replace File ☐ Append To File ☐ Statistics by Cell

Output Format: ASCII PVS/LVS/INVERTER.erc\_errors.ascii

Output Errors Hierarchically: Rules Definition

☐ Use Waivers for ERC Checks

**LVS Report**

Name: INVERTER.rep

Options: -filtered\_devices SET

Limit: 50 Mismatched Nets Limit: 100

**Additional Output**

☒ Create Quantus QRC Input Data

QRCDataDir: svdb

Keep Layers: None

Start LVS DE ☒

Cancel Apply Submit

Latest Loaded Presets File: /abc/eca/home/ss/grads/abdelrahman/abdelrahman/Cadence\_12345678/virtuoso6.lvs\_preset

Figure 10 LVS run window.

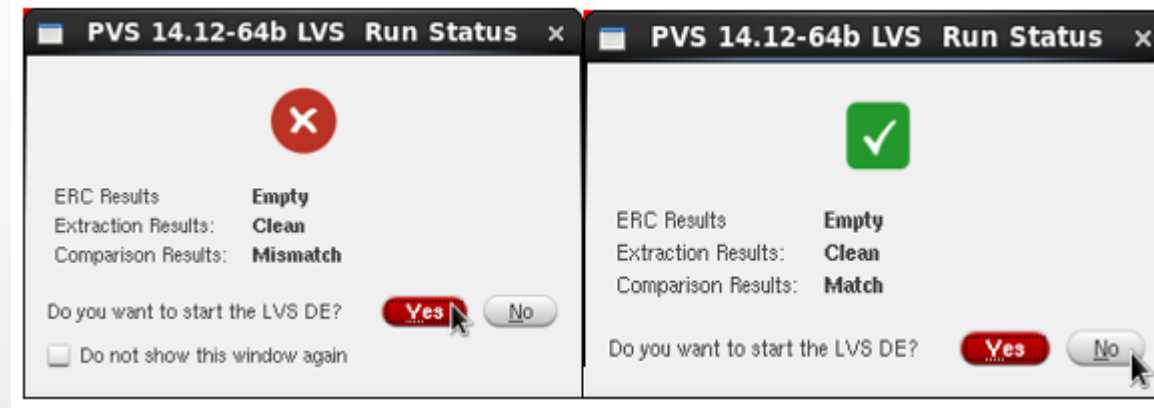


Figure 11 LVS Run status: [Left-Mismatch] and [Right-Match]

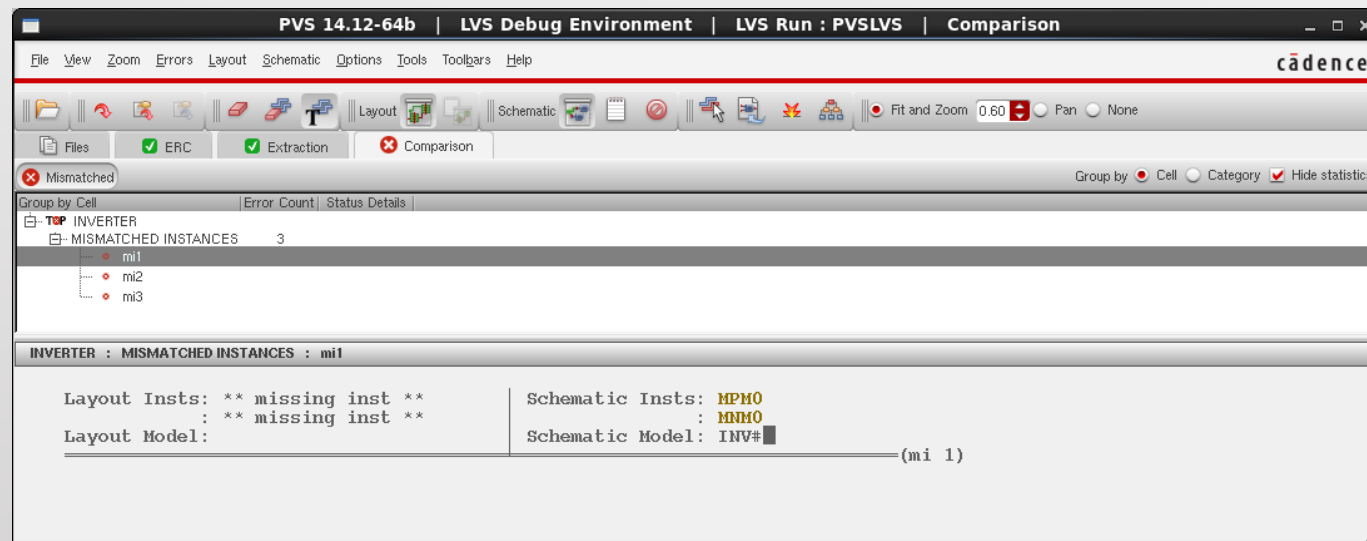


Figure 12 LVS Debug Environment



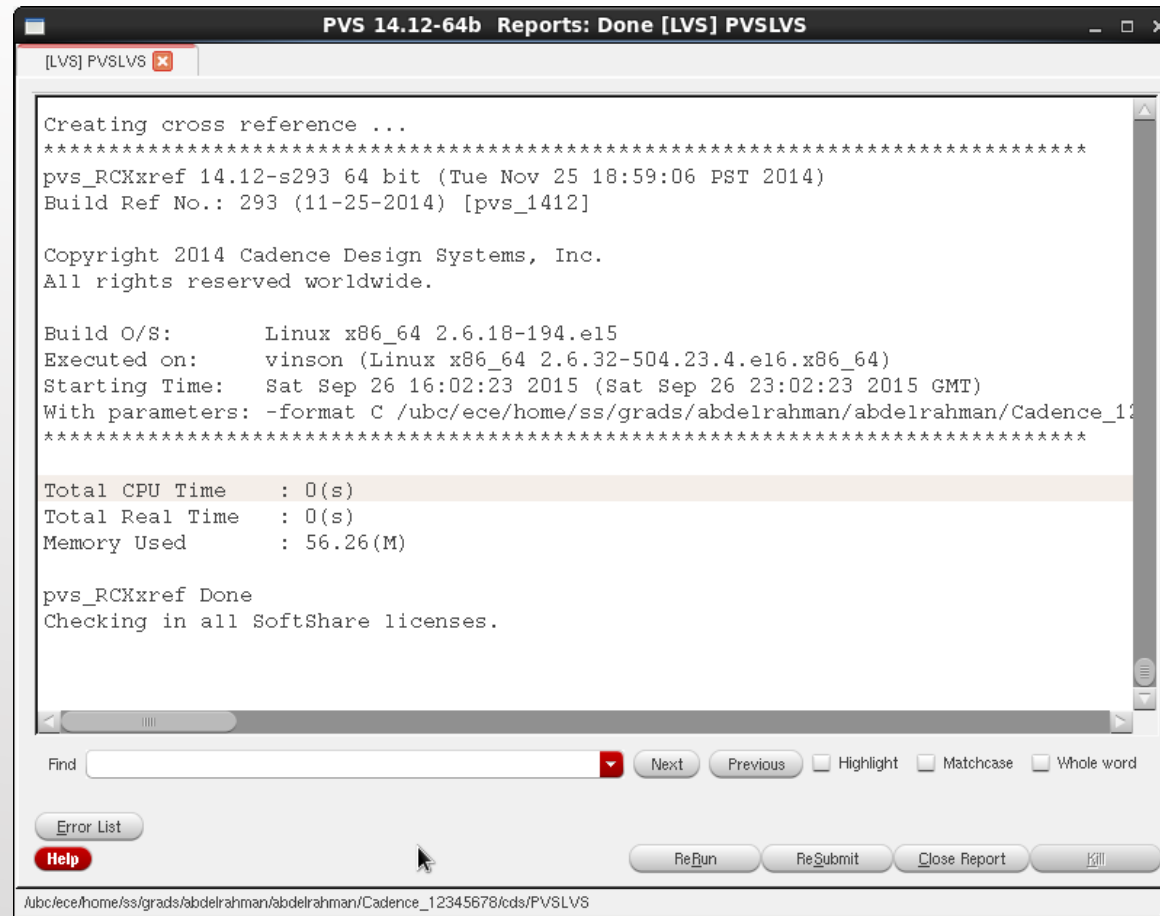


Figure 13 PVSLVS report window.

# Parasitic Extraction and Post Layout Simulation

- **5.1 Run QRC extraction**

After passing LVS your layout will be ready for extraction. In the layout editor window go to <**QRC -> Run PVS-QRC**>. In the “QRC (PVS) interface” window, make sure that the cell name and the technology fields are right, and press OK. In the “Parasitic Extraction Run Form” make sure to set the fields as shown in Figure 14 to Figure 16. Press Ok and wait for the tool to do the extraction and give you the completion notification shown in Figure 17.

The screenshot shows the 'QRC (PVS) Parasitic Extraction Run Form' with the 'Setup' tab selected. The form contains various configuration options for parasitic extraction. Key settings include:

- Technology:** gpdK045\_pvs
- RuleSet:** rcx\_typical
- Setup Dir:** /CMC/kits/AMSKIT616\_GPDK/tech/gpdK045/pvs/./qrc/ty
- Output:** Extracted View
- Lib:** ELEC402
- Cell:** INVERTER
- View:** av\_extracted
- Parasitic Res Component:** presistor
- Parasitic Cap Component:** pcapacitor
- Parasitic Ind Component:** pinductor
- Parasitic M Component:** pmind
- Inductance L1 Prop Id:** ind1
- Inductance L2 Prop Id:** ind2
- Parasitic CCVS Component:** ccvs
- Parasitic VS Component:** vsource
- Substrate Extract:** ☐
- Extract MOS Diffusion AP:** ☐
- Substrate Profile:** NONE
- Extract MOS Diffusion Res:** ☐
- Add LVS MOS Diffusion Res:** ☐
- Extract MOS Diffusion High:** NONE

At the bottom, there is a text area for 'Library Directory' and a row of buttons: OK, Cancel, Defaults, Apply, Load State, Save State, View Command File, and Help.

Figure 14 Parasitic Extraction Run Form (Setup).

QRC (PVS) Parasitic Extraction Run Form

Setup Extraction Filtering Netlisting Run Details Substrate

Extraction Type RC Name Space Schematic Names

Max fracture length infinite microns Temperature 25.0 C Edit

Cap Coupling Mode Coupled Ref Node VSS

Mult Factor 1.0 Diffusion Equation R

Ladder Network

Select... User Region View Edit

Net Selection Type Full Chip All Nets QRCFS Extraction Mode NONE

Layer Setup Customization Edit... QRCFS High

Resistance Mesh Edit...

Select... R Mesh User Region View Edit

Non-Manhattan Resistance Accuracy default

From File

SelfFromSch

SelfFromLay

Change LithoBias Direction

Litho Config File View Edit

Contour Directory

HRCX Cells: Specify a list of cells which appear in the output hierarchy, requires cell name, with optional view and lib names (cell, view, lib).

OK Cancel Defaults Apply Load State Save State View Command File Help

Figure 15 Parasitic Extraction Run Form  
(Extraction)



QRC (PVS) Parasitic Extraction Run Form

Setup Extraction **Filtering** Netlisting Run Details Substrate

Dangling R	<input type="checkbox"/>	MinR	<input type="text" value="0.001"/>	
Merge ParallelR	<input type="checkbox"/>	MinC	<input type="text" value="0.01"/>	tF <input type="text" value="0.1"/> %
Reduce Parasitics	<input type="checkbox"/>	Decoupled To Substrate	<input type="checkbox"/>	
Reduction Control	<input type="text" value=""/>	Filter Size	<input type="text" value="2.0"/>	
Exclude Self Capacitance	<input checked="" type="checkbox"/>	M Factor R	<input type="text" value=""/>	
Exclude Floating Nets	<input type="checkbox"/>	M Factor Keep R	<input type="checkbox"/>	M Factor W <input type="checkbox"/>
Exclude Float Limit	<input type="text" value="2000"/>	Max Merged Via Size	<input type="text" value="auto"/>	Microns
Max Fracture Via Count	<input type="text" value="auto"/>	<input type="checkbox"/> Max Merged Via Count	<input type="text" value="1"/>	
		Array Vias Spacing	<input type="text" value="auto"/>	Microns

M Factor Exclude File  ...

Enter Exclude Reduce Parasitics Nets:

From File ☐

Enter Power Nets:

From File ☐

Enter Ground Nets:

From File ☐

Ground Nets: Specify Ground Nets that are not defined as global by LVS extraction for exclusion from parasitic resistance extraction.

Figure 16 Parasitic Extraction Run Form (Filtering).

## 5.2 Display parasitic associated with nodes

- After extracting the layout note that a new cell view named “av\_extracted” was added in the Library manager. To show the summation of capacitance associated with each node go to the schematic editor window and go to <**Launch -> Plugins -> Parasitics**>. In the Parasitics menu go to Setup. In the “Setup Parasitics” window, make sure that all the fields are set similar to Figure 18 and press OK.

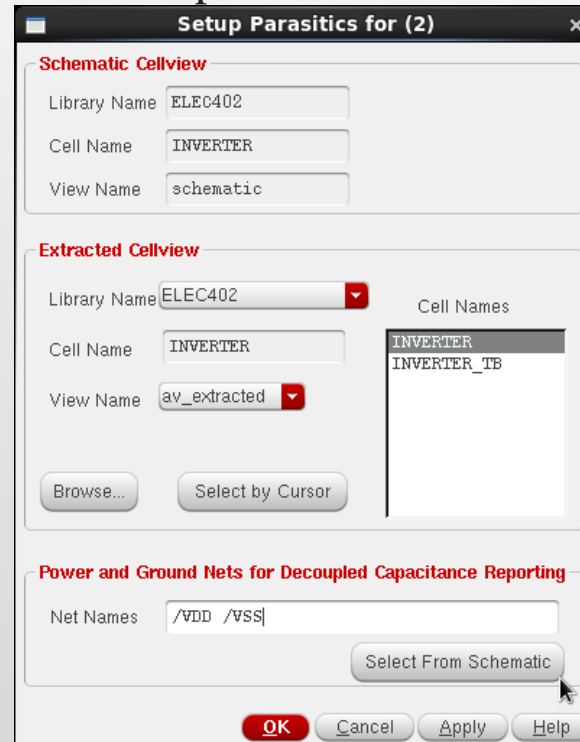


Figure 18 Setup Parasitics

Go to <**Parasitics -> Show Parasitics**>. Note that the tool added the summation of capacitance associated with each node to the schematic as shown in Figure 19.

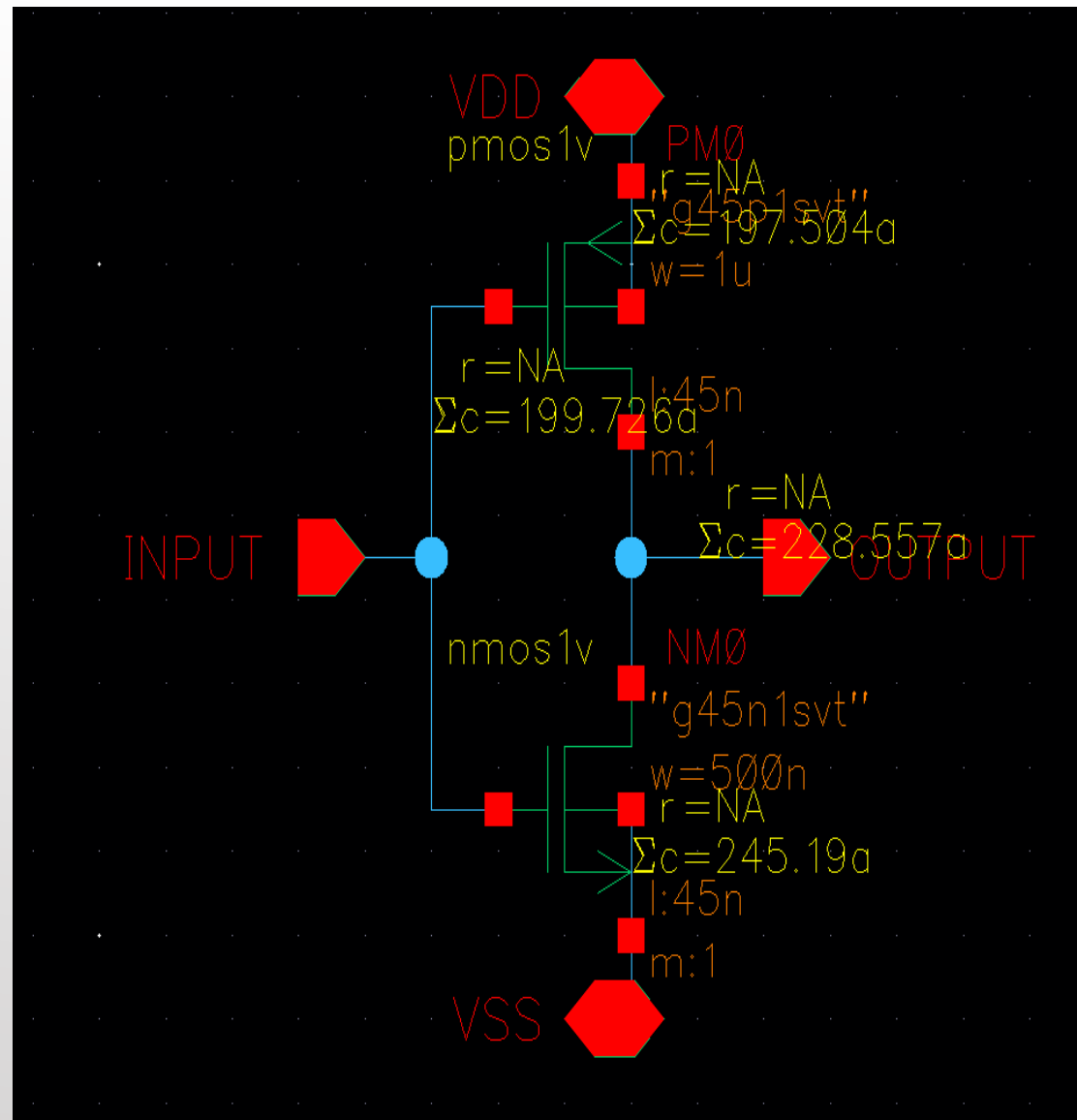


Figure 19 CMOS inverter with the annotated parasitic capacitance.



# Post-layout Simulation

- After extracting the layout all the simulation done in “[Cadence Virtuoso – Schematic & Simulations – Inverter \(45nm\)](#)” tutorial should be repeated to include the parasitics’ effect. To do so, you will have to let the simulator know that you want to use the extracted view coupled with the TB netlist, which is done through the “config” view. From the library manager create a new cell view for your TB with the Type field set to “config” as shown in Figure 20. The “New Configuration” window will open. Modify the fields, use the AMS template, and add the “av\_extracted” to the View List as shown in Figure 21.

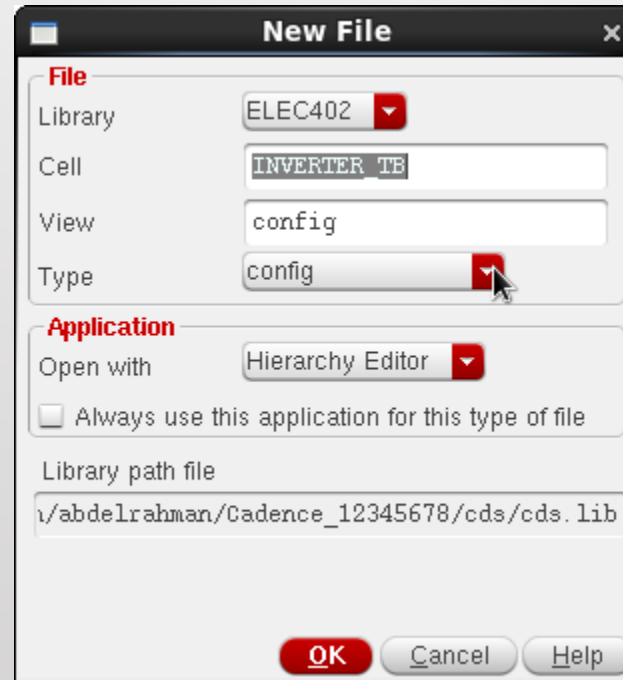


Figure 20 Create config view for the TB.

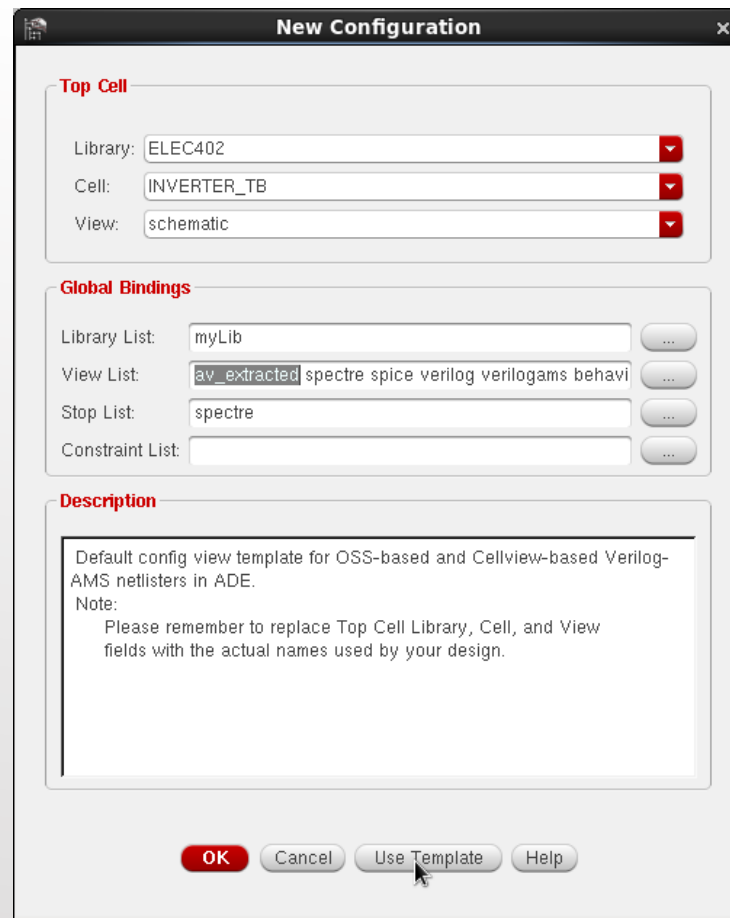


Figure 21 New Configuration.

In Virtuoso Hierarchy Editor, set the View to use for your cell to be “av\_extracted” and press “Recompute the hierarchy” as shown in Figure 22. Press Open to open virtuoso schematic editor with in the config view and repeat all the needed simulations. **How did the results change?!**

- **Note:** Always make sure that the word “Config” exists in the schematic editor window’s name, and the ADE window as well as show in Figure 23

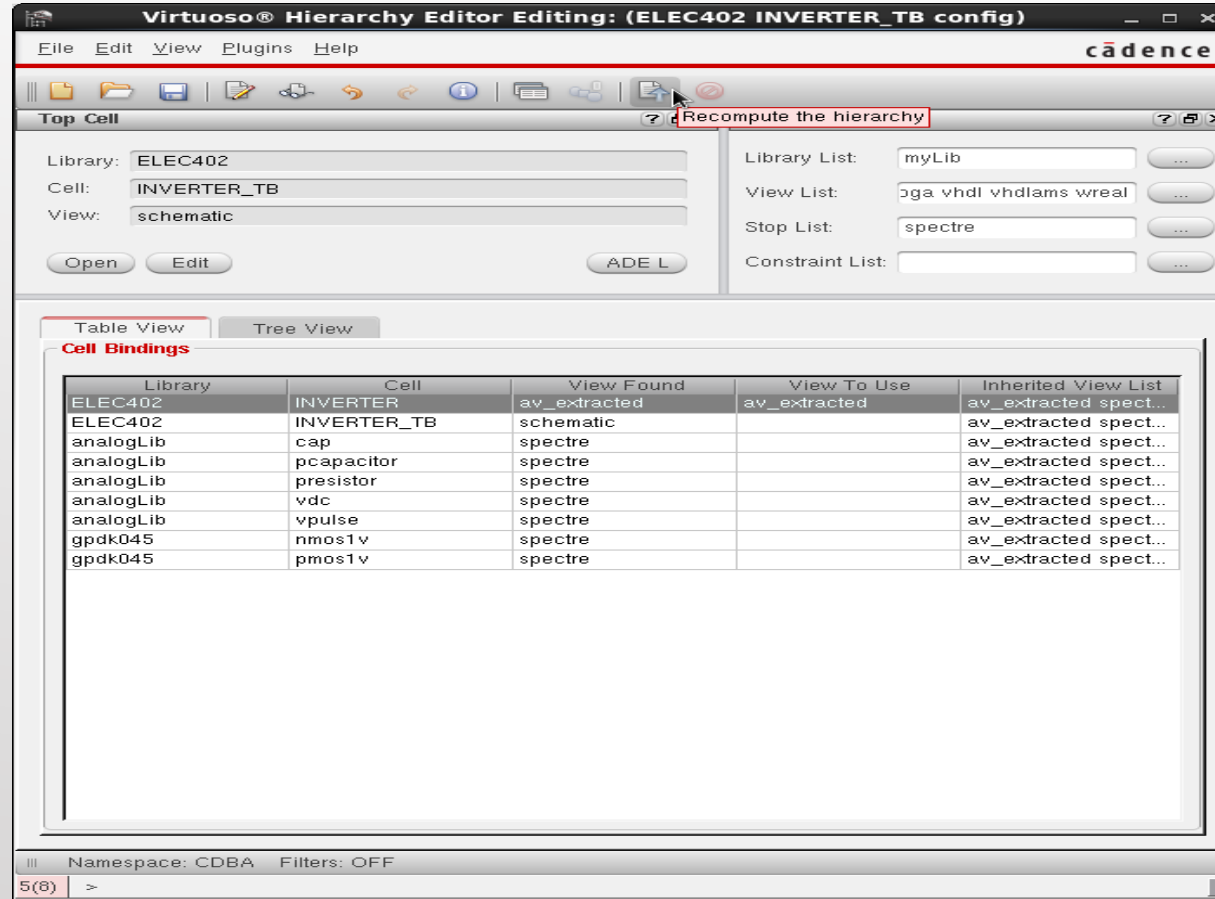


Figure 22 Virtuoso Hierarchy Editor.

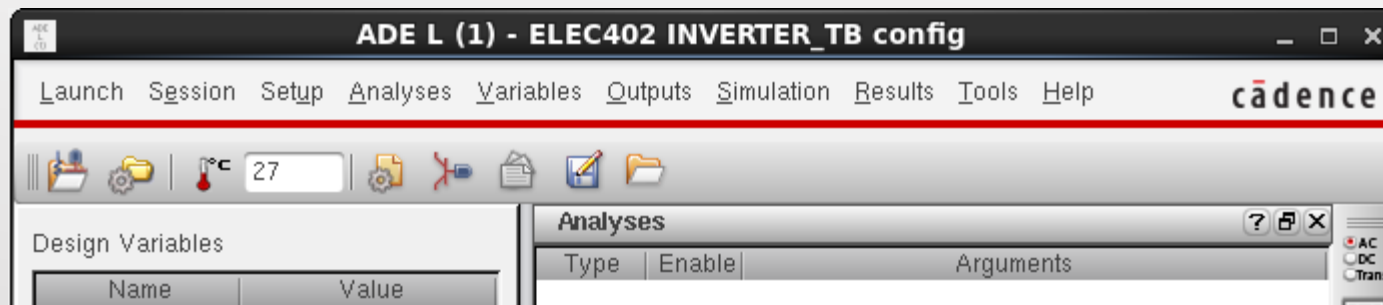


Figure 23 ADE in “config” view.



## Contact details :



+91 8792662546



training@siliconchip.in



[http://www.siliconchip.in\](http://www.siliconchip.in/)



#25/A, 2nd floor, Next to St. Theresa School, Kamadenu Layout, B.  
Narayanapura, Bengaluru, Karnataka

THANK YOU