Tutorial on Layout Extraction and Post-Layout Simulation

Teaching Assistant: Mohsen Namavar

Winter 2025

Contents

1 Introduction

This tutorial guides students through the process of extracting a layout using Assura Quantus Parasitic Extraction and Post-Layout Simulation Process. The focus will be on extracting the layout without and with RC parasitic elements.Ensure that the schematic and layout have been properly designed, and both DRC and LVS checks have been successfully passed prior to proceeding (Figure1).

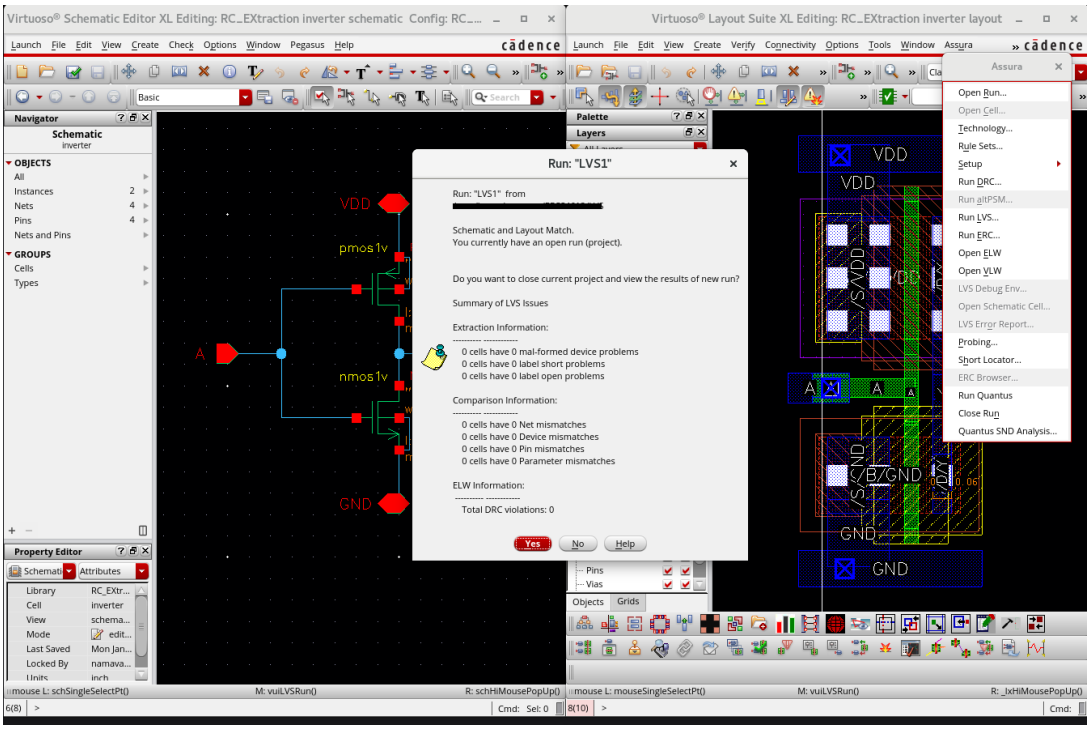

Figure 1: LVS pass message for an inverter

2 Layout Extraction using Assura Quantus Parasitic Extraction

2.1 Introduction

This section guides students through the process of extracting a layout using Assura Quantus Parasitic Extraction. The focus will be on extracting the layout without and with RC parasitic elements. Ensure that the schematic and layout have been properly designed, and both DRC and LVS checks have been successfully passed prior to proceeding.

2.2 Opening Quantus Parasitic Extraction

To begin the parasitic extraction process:

1. In the Virtuoso Layout Suite, navigate to the menu bar and select **Assura → Run Quantus**.

2. This will open the Quantus Parasitic Extraction form, as shown in Figure 2.

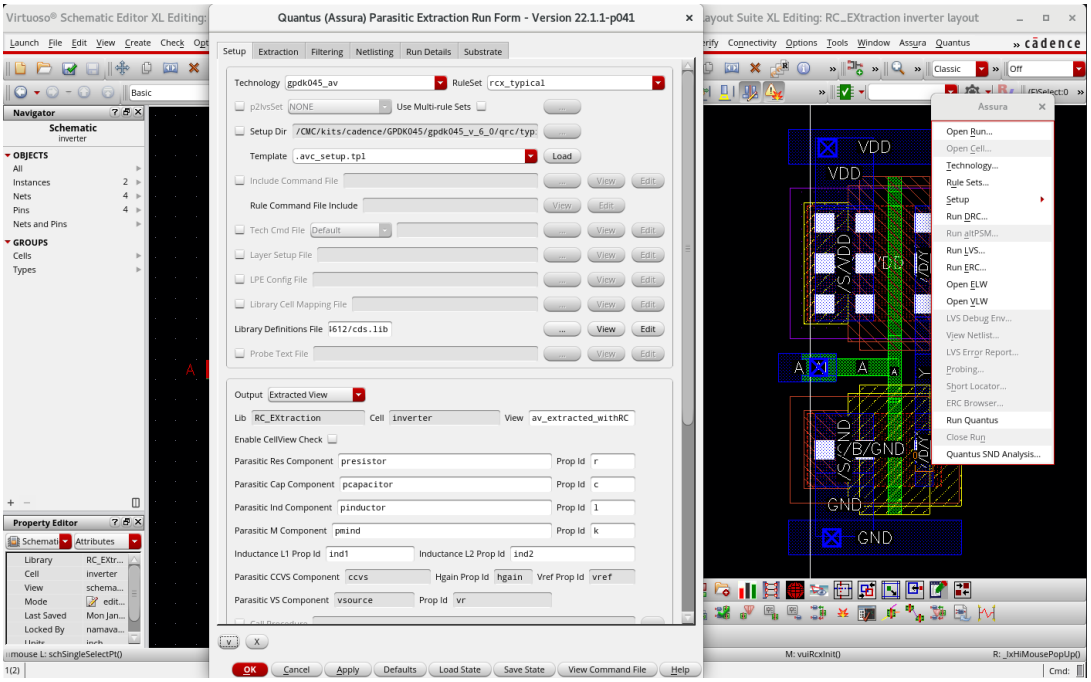

Figure 2: Quantus Parasitic Extraction Setup Tab. Ensure the correct template is selected.

2.3 Configuring the Extraction Settings

The Quantus extraction process described here focuses on extracting the layout without RC parasitic elements and with RC parasitics.

2.3.1 Setup Tab Configuration

In the Quantus setup form:

1. Select the **Technology** as `gpdk045_av`.
2. Set the **Rule Set** to `rcx_typical`.
3. Ensure the correct **Template** is selected. In this case, choose `.avc_setup.tpl`, as highlighted in Figure 2.
4. Set the **Output** to **Extracted View** and specify a unique view name for the output (e.g., `av_extracted_withoutRC` or `av_extracted_withRC`).

2.3.2 Without RC Parasitic Extraction (No RC)

To perform extraction without RC parasitics:

1. Navigate to the **Extraction Tab** and set the **Extraction Type** to **NONE**, as shown in Figure 3.
2. Leave all other settings as default.
3. Click **OK** or **Apply** to start the extraction process.

4. Wait for the process to complete. Upon completion, a confirmation message will appear, as shown in Figure 4.

5. Once the extraction process is complete, verify the creation of the new extracted view in the **Library Manager**, as shown in Figure 5.

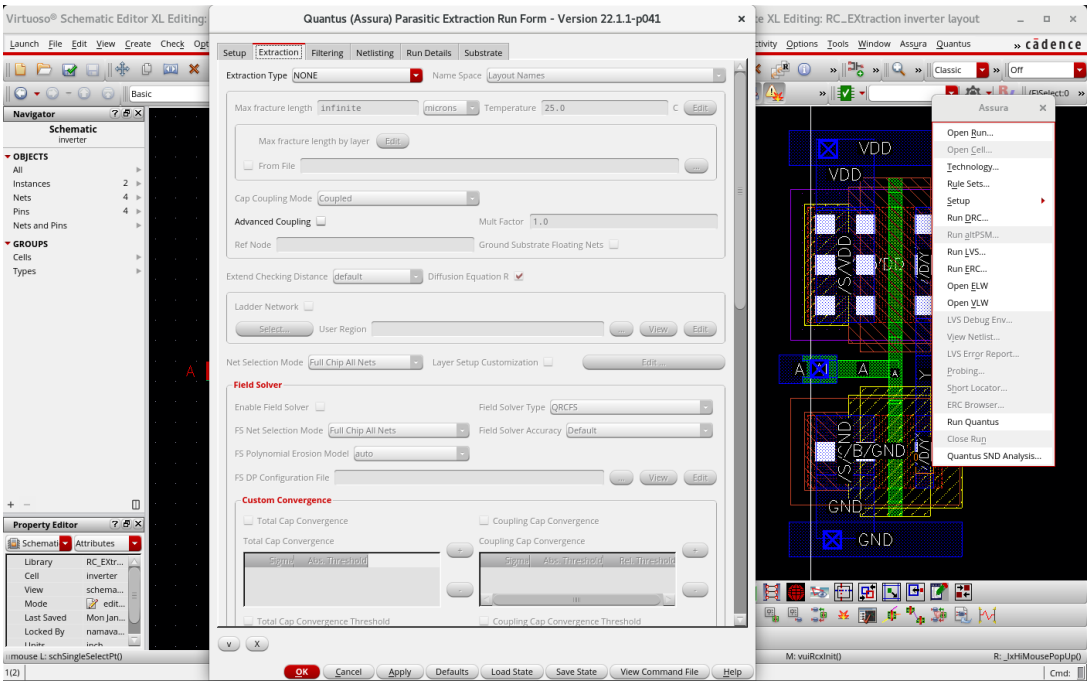

Figure 3: Quantus Extraction Tab Configuration for No RC Parasitic Extraction. Extraction Type is set to NONE.

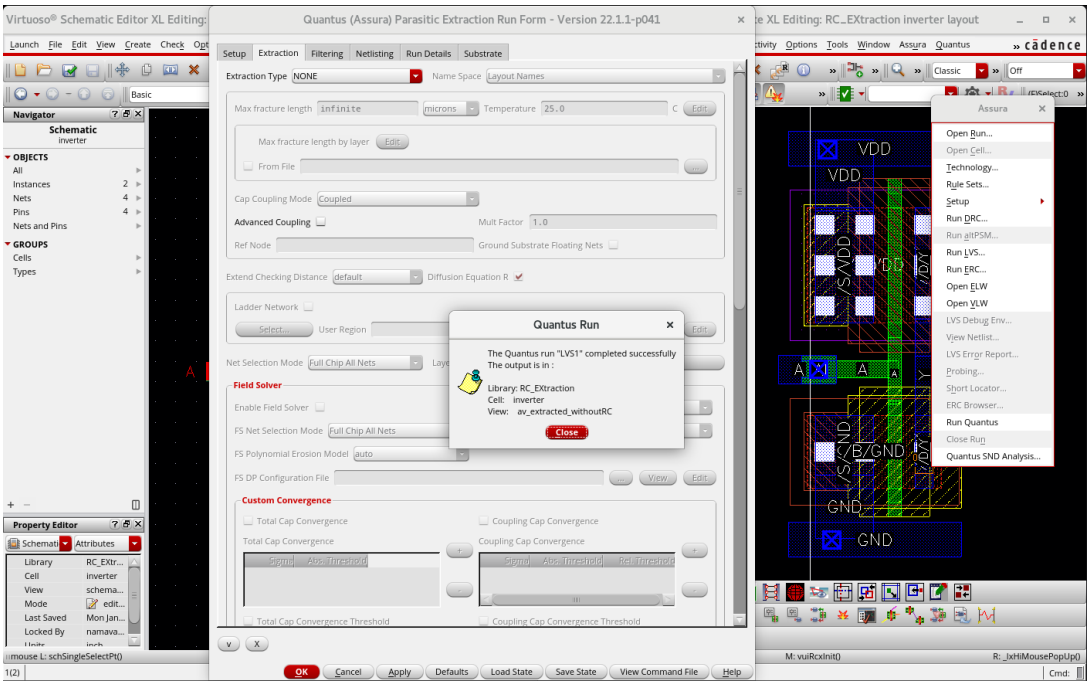

Figure 4: Confirmation of Successful Quantus Extraction for No RC Parasitics.

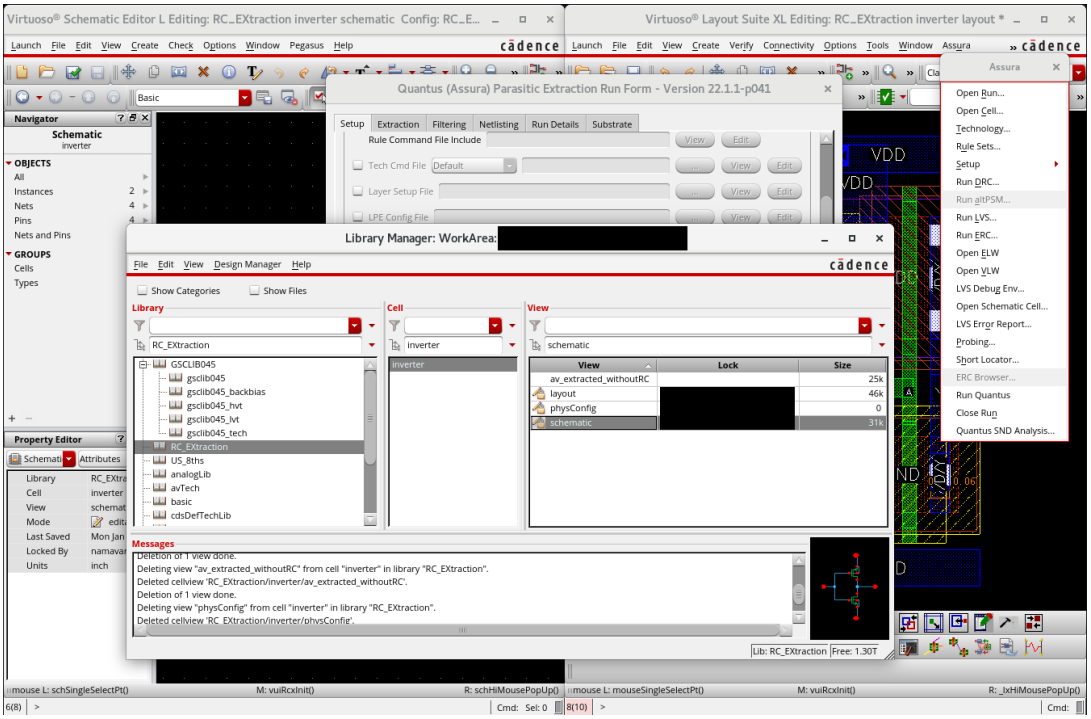

Figure 5: Library Manager showing the new extracted view for the inverter cell. The view name matches the one specified in the Output field.

2.3.3 With RC Parasitic Extraction (RC)

To perform extraction with RC parasitics:

1. Open the Quantus Parasitic Extraction form again by navigating to **Assura** → **Run Quantus**.

2. In the **Setup Tab**, configure the settings as before. Choose a proper and unique name for the **View** in the Output field (e.g., `av_extracted_withRC`). Refer to Figure 7.

3. Navigate to the **Extraction Tab** and set the **Extraction Type** to **RC**. Use **Schematic Names** for the Name Space. Enter the name of the Ref Node corresponding to the ground net in your layout (e.g., if the ground net is named `VSS`, enter `VSS`). Refer to Figure 6.

4. Click **OK** or **Apply** to start the extraction process.

5. Wait for the extraction process to complete. Upon completion, a confirmation message will appear, as shown in Figure 8.

6. Once the extraction process is complete, verify the creation of the new extracted view in the **Library Manager**, as shown in Figure 9.

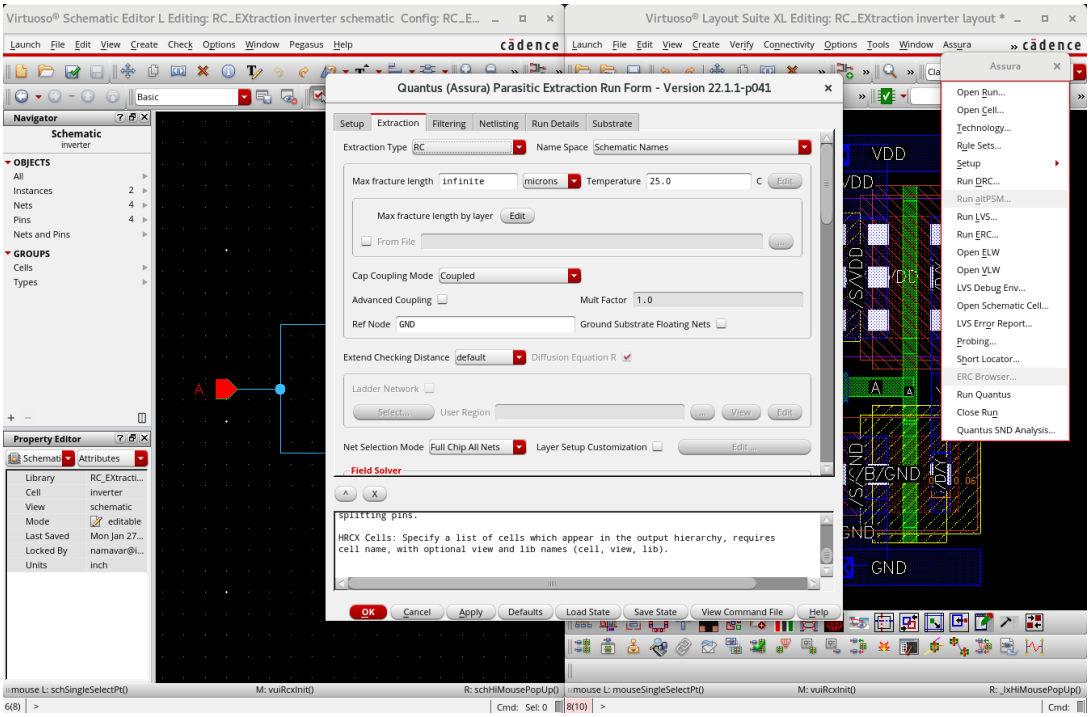

Figure 6: Quantus Extraction Tab Configuration for RC Parasitic Extraction. Set Extraction Type to RC and specify the Ref Node.

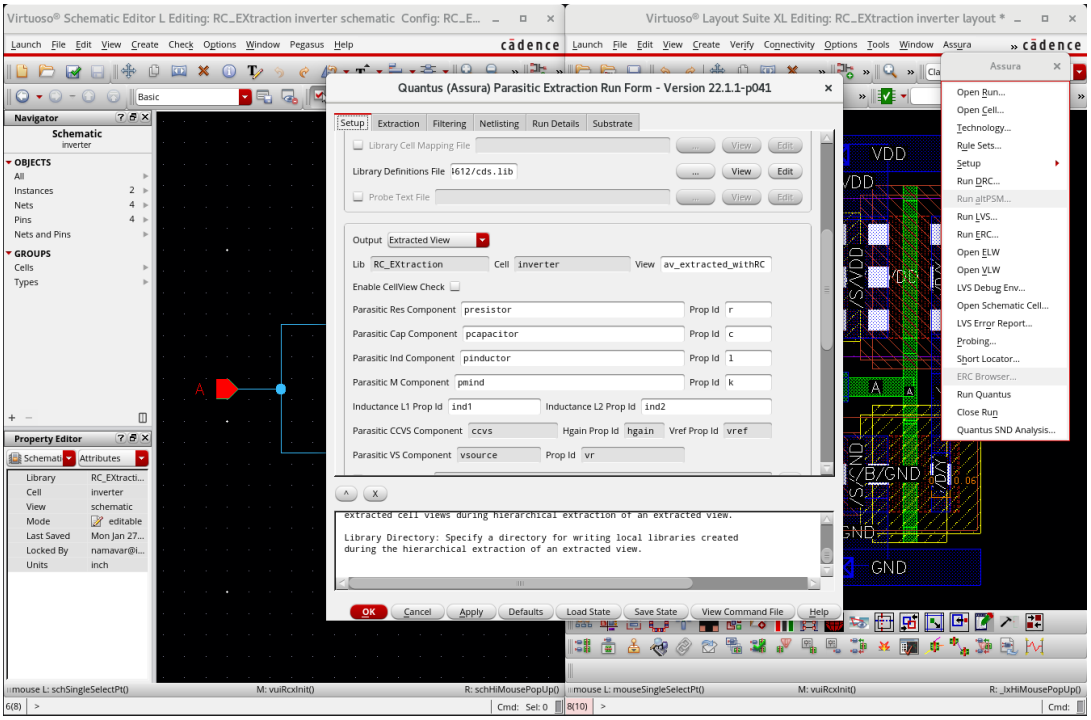

Figure 7: Quantus Setup Tab Configuration for RC Parasitic Extraction. Ensure the proper View name is set.

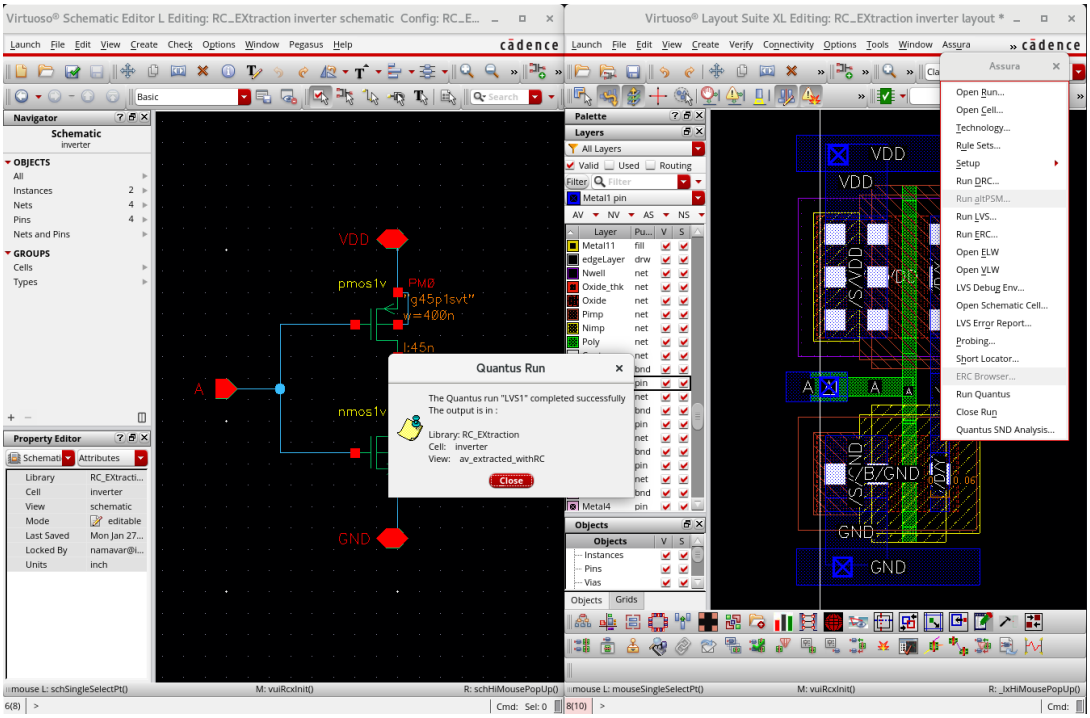

Figure 8: Confirmation of Successful Quantus Extraction for RC Parasitics.

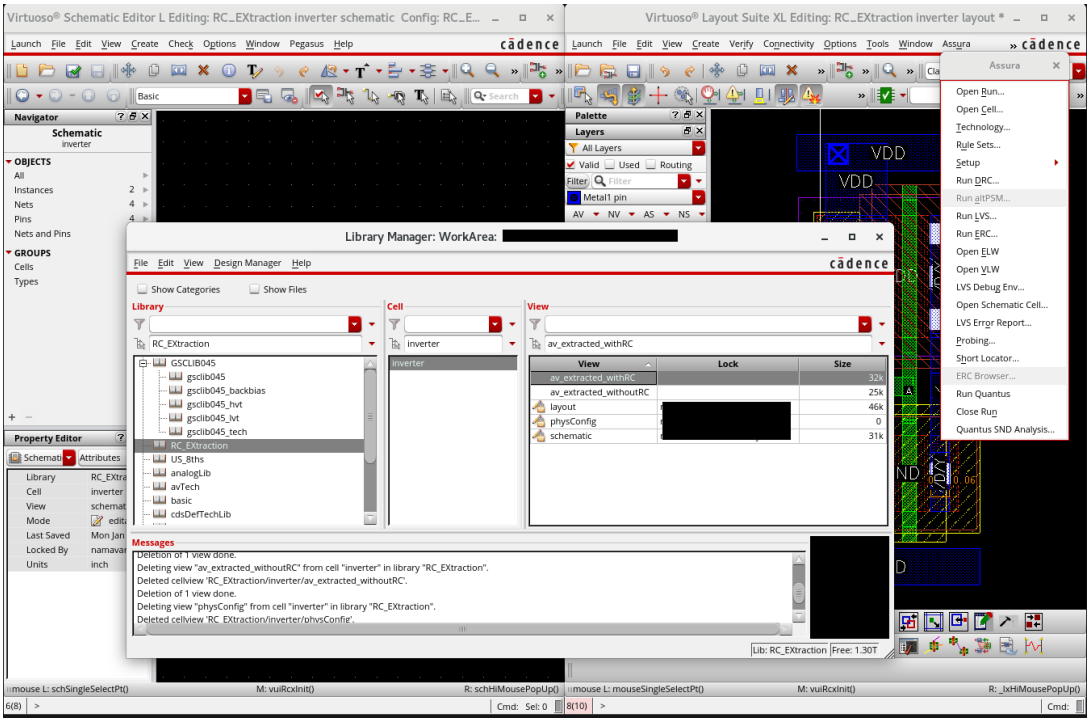

Figure 9: Library Manager showing the new extracted view for the inverter cell with RC parasitics. The view name matches the one specified in the Output field.

2.3.4 Comparing Extracted Views

After completing the extraction processes, you can open the two extracted views (with and without RC parasitics) side by side to compare the results. The left window shows

the view with RC parasitics, and the right window shows the view without RC parasitics, as illustrated in Figure 10.

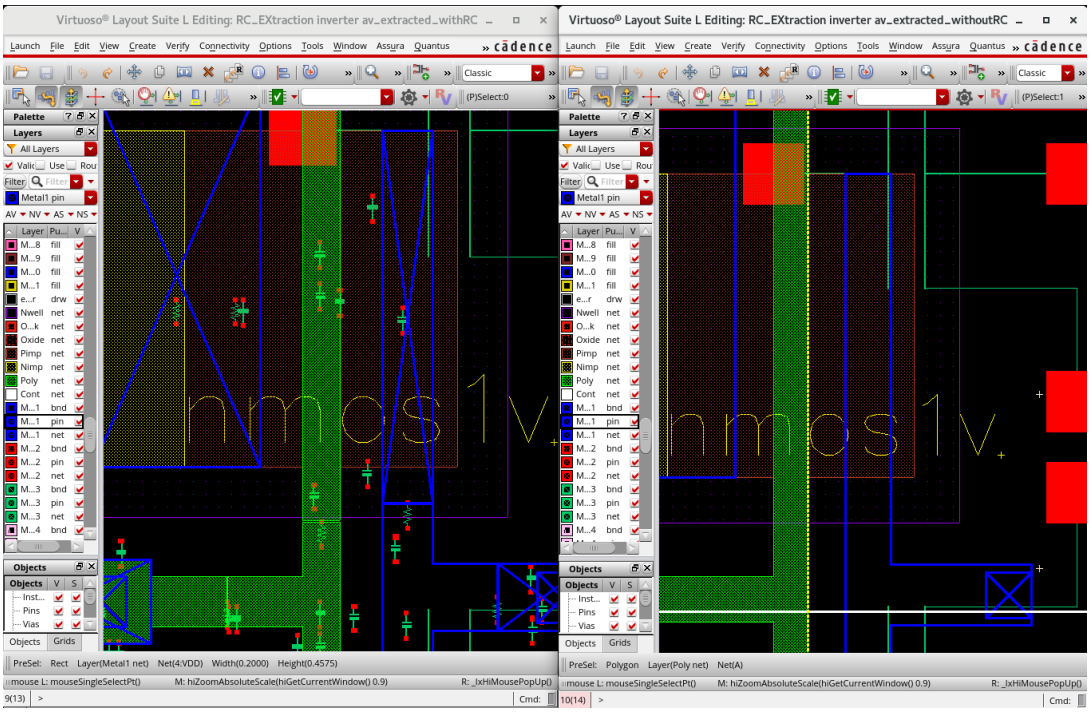

Figure 10: Comparison of Extracted Views: Left - with RC parasitics; Right - without RC parasitics. The left view includes parasitic elements, while the right view contains only NMOS and PMOS.

3 Post-Layout Simulation

3.1 Creating the Testbench Schematic

Before running post-layout simulation, a testbench schematic must be created. The testbench is used to apply test stimuli and analyze the extracted design.

1. Open the **Library Manager**.
2. Select the appropriate library (e.g., `RC_Extraction`).
3. Click on **File** → **New** → **Cell View** to create a new schematic.
4. Name the testbench as `logicname_tb`, replacing `logicname` with the actual design name (e.g., `inverter_tb`).
5. Set the **View** to `schematic` and the **Type** to `schematic`.
6. Click **OK** to proceed.

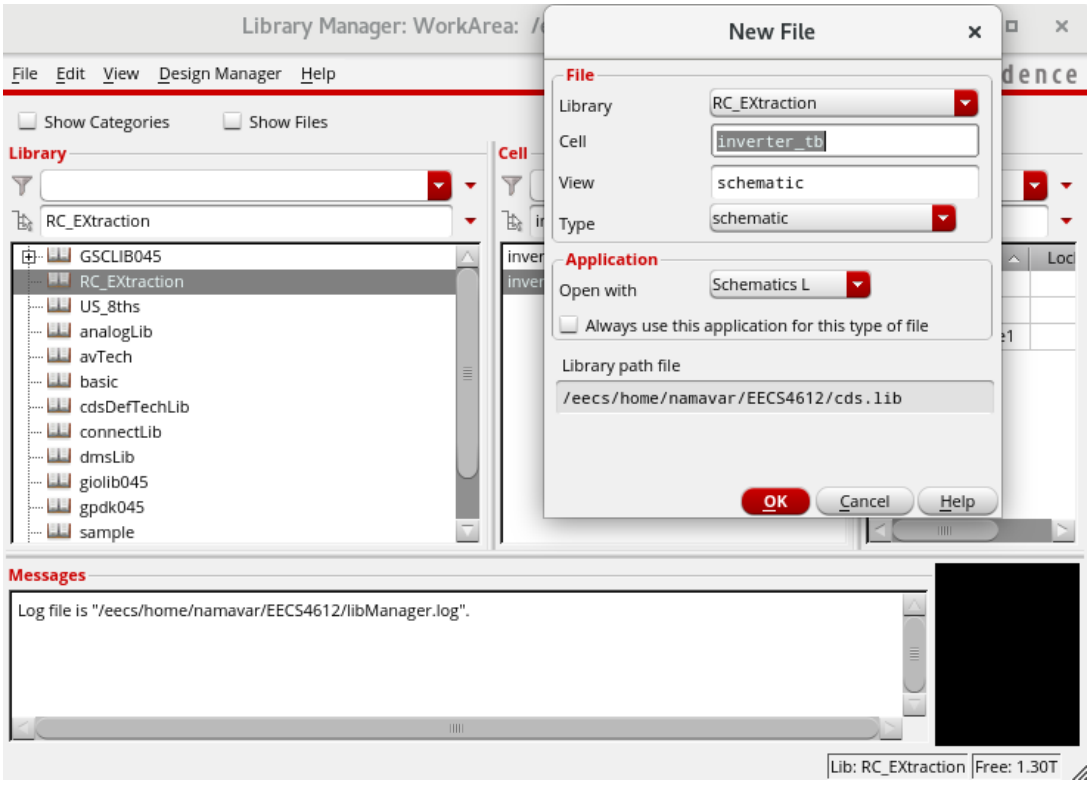

Figure 11: Creating a new schematic for the testbench. The testbench should be named `logicname_tb`.

3.2 Building the Testbench Circuit

After creating the testbench schematic, the next step is to instantiate the logic cell (e.g., inverter) and set up the circuit for simulation.

1. Instantiate the logic cell (e.g., inverter) three times within the testbench schematic.
2. Ensure that the **VDD** and **GND** pins of all instances are properly connected.
3. Use a **VDC** source from the `analogLib` library to apply **VDD = 1.1V**.
4. Use a **VPULSE** source from the `analogLib` library to apply an input pulse to the instances.
5. Connect a **50fF** capacitor to the output of each instance.
6. Assign appropriate labels to the output nodes using the labeling feature to distinguish them.

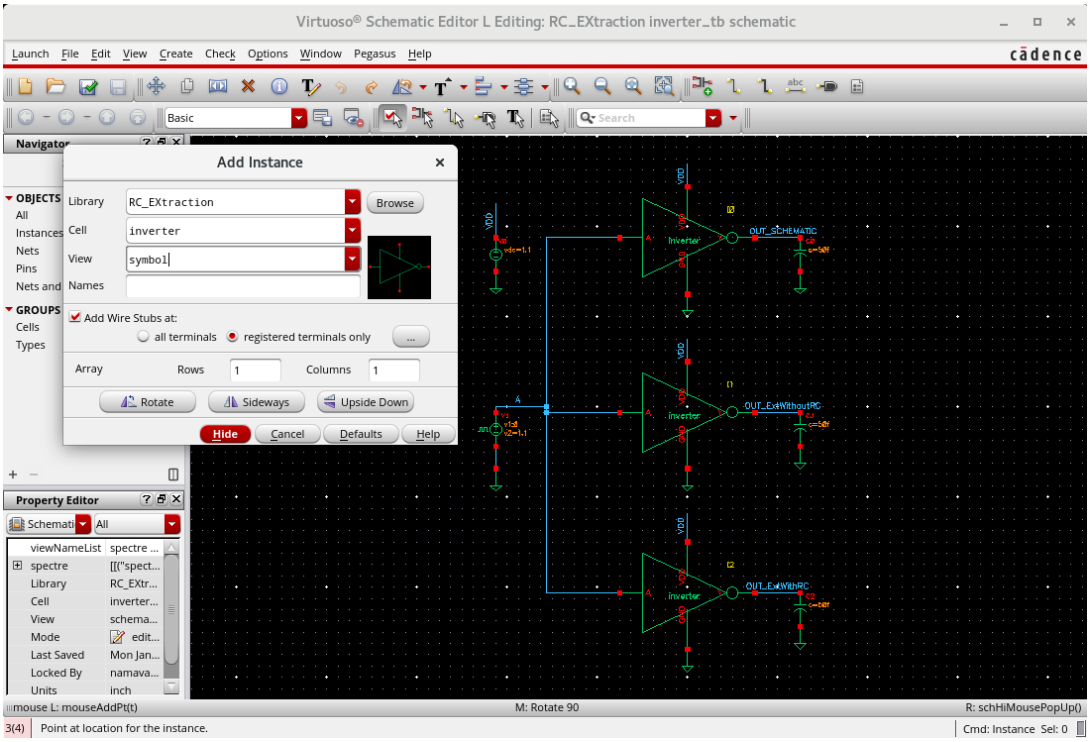

Figure 12: Testbench schematic with instantiated logic cells, power connections, input stimulus, and output capacitors.

3.3 Creating the Configuration View

After building the testbench schematic, a configuration view must be created to define which extracted view will be used in the simulation.

1. Open the **Library Manager**.
2. Select the testbench cell (e.g., `inverter_tb`) from the appropriate library.
3. Click on **File** → **New** → **Cell View**.
4. Name the view as `config` and set the **Type** to `config`.
5. Select **Hierarchy Editor** as the application to open the view.
6. Click **OK** to proceed.

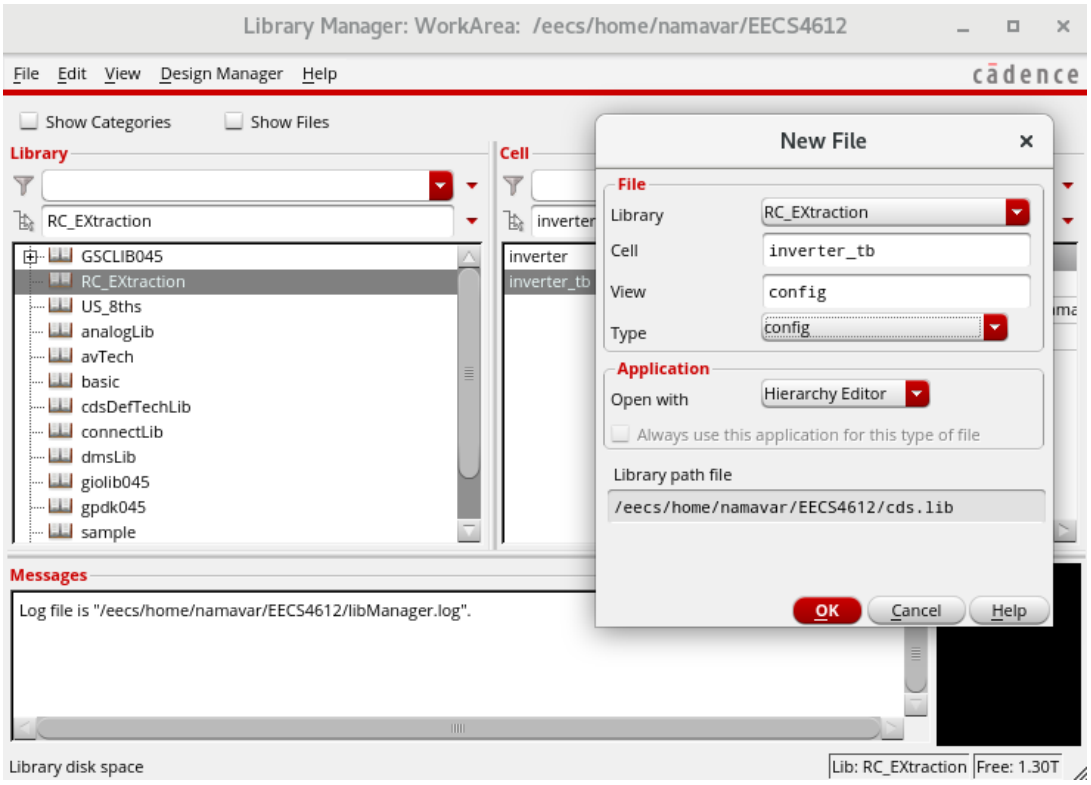

Figure 13: Creating a new configuration view for the testbench.

3.4 Creating the Configuration View

After building the testbench schematic, a configuration view must be created to define which extracted view will be used in the simulation.

1. Open the **Library Manager**.
2. Select the testbench cell (e.g., `inverter_tb`) from the appropriate library.
3. Click on **File** → **New** → **Cell View**.
4. Name the view as `config` and set the **Type** to `config`.
5. Select **Hierarchy Editor** as the application to open the view.
6. Click **OK** to proceed.

3.5 Setting Up the Configuration View

Once the configuration view is created, the Virtuoso Hierarchy Editor window will pop up (see Figure 14).

1. Press the **Use Template** button (Figuer 15).
2. Select **Spectre** in the template window and press **OK**.
3. Set the **View** field to `schematic`.

4. Press **OK** to finalize the configuration (Figuer 16).

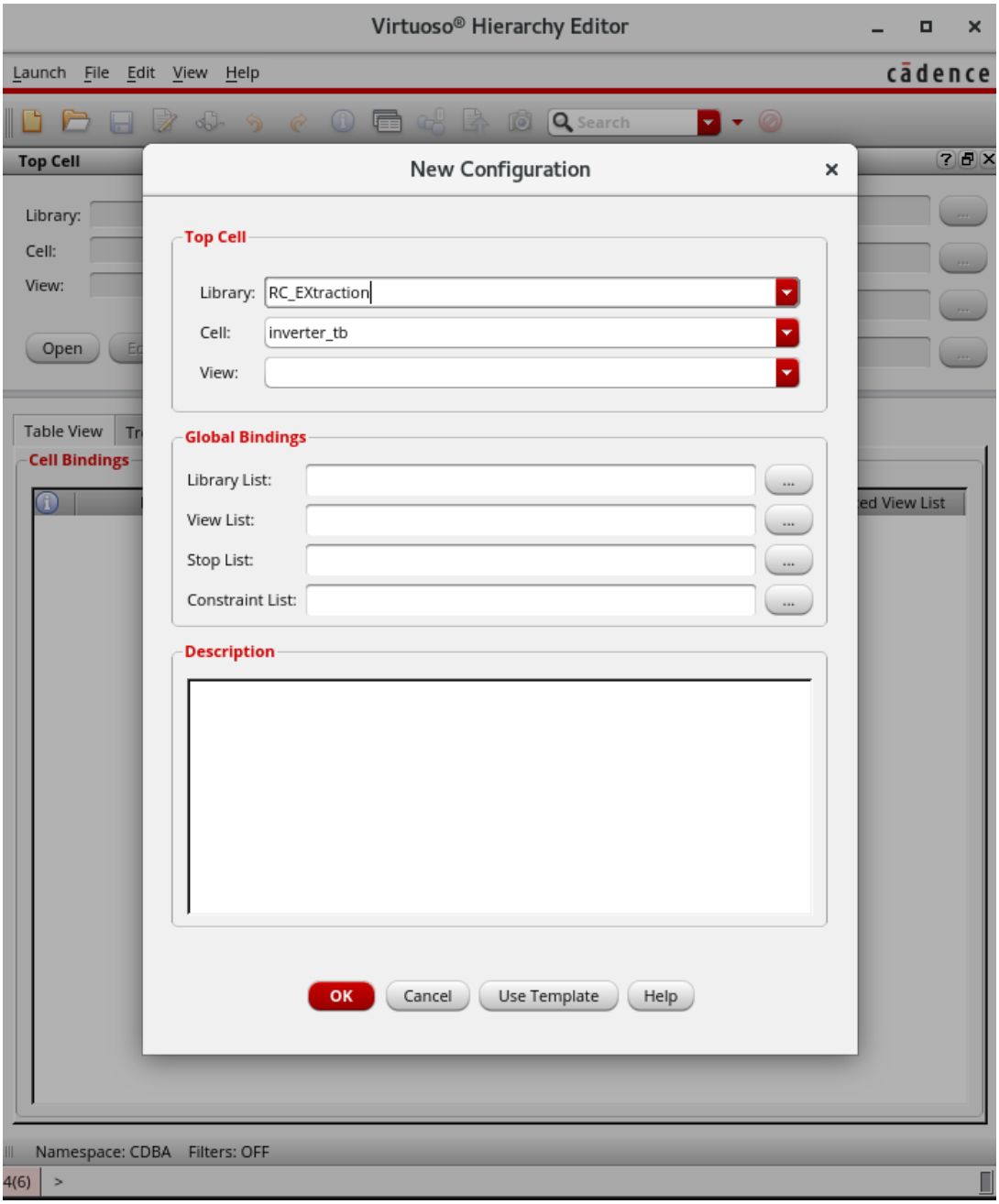

Figure 14: Virtuoso Hierarchy Editor: Setting up the configuration view.

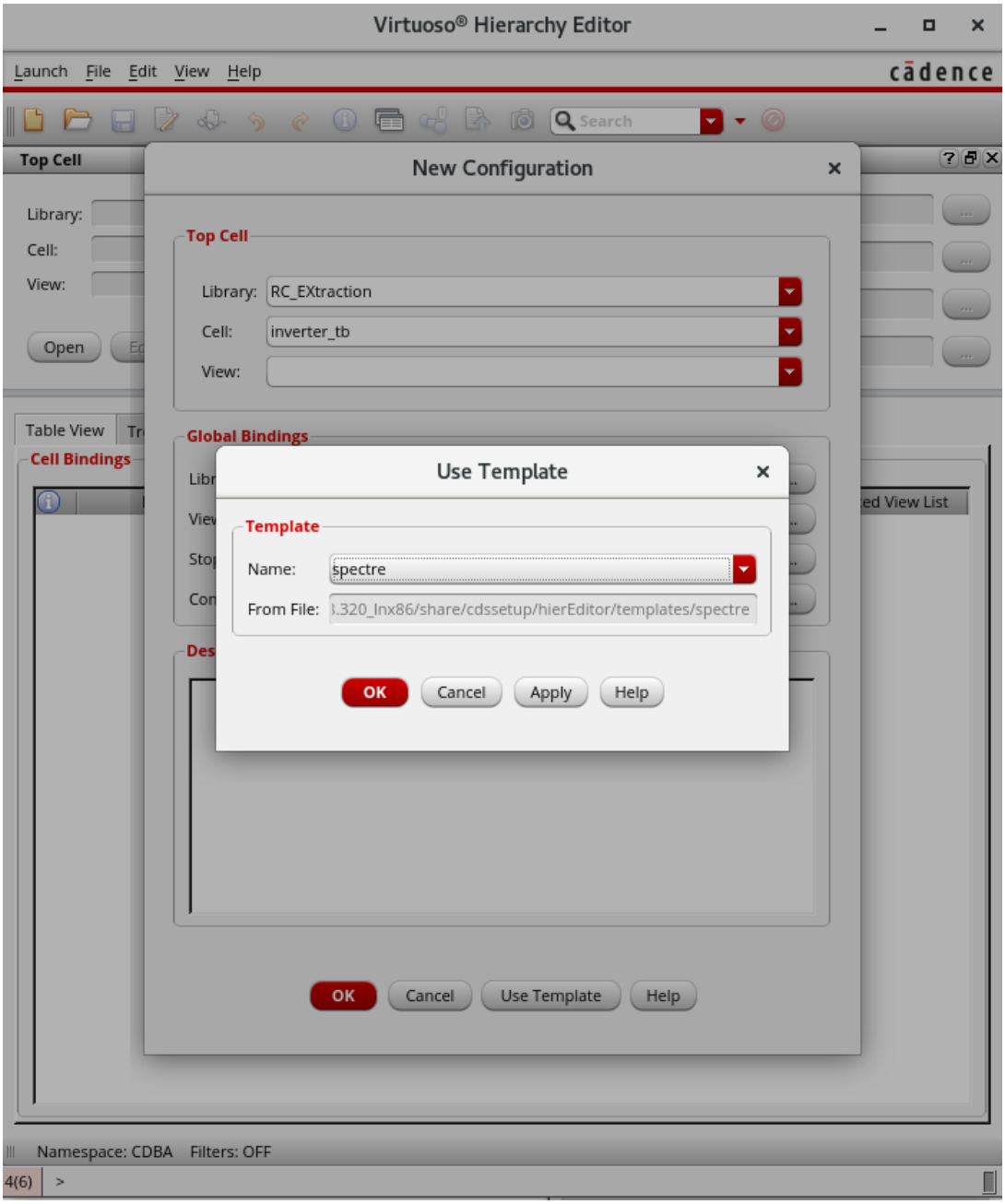

Figure 15: Selecting the Spectre template for the configuration.

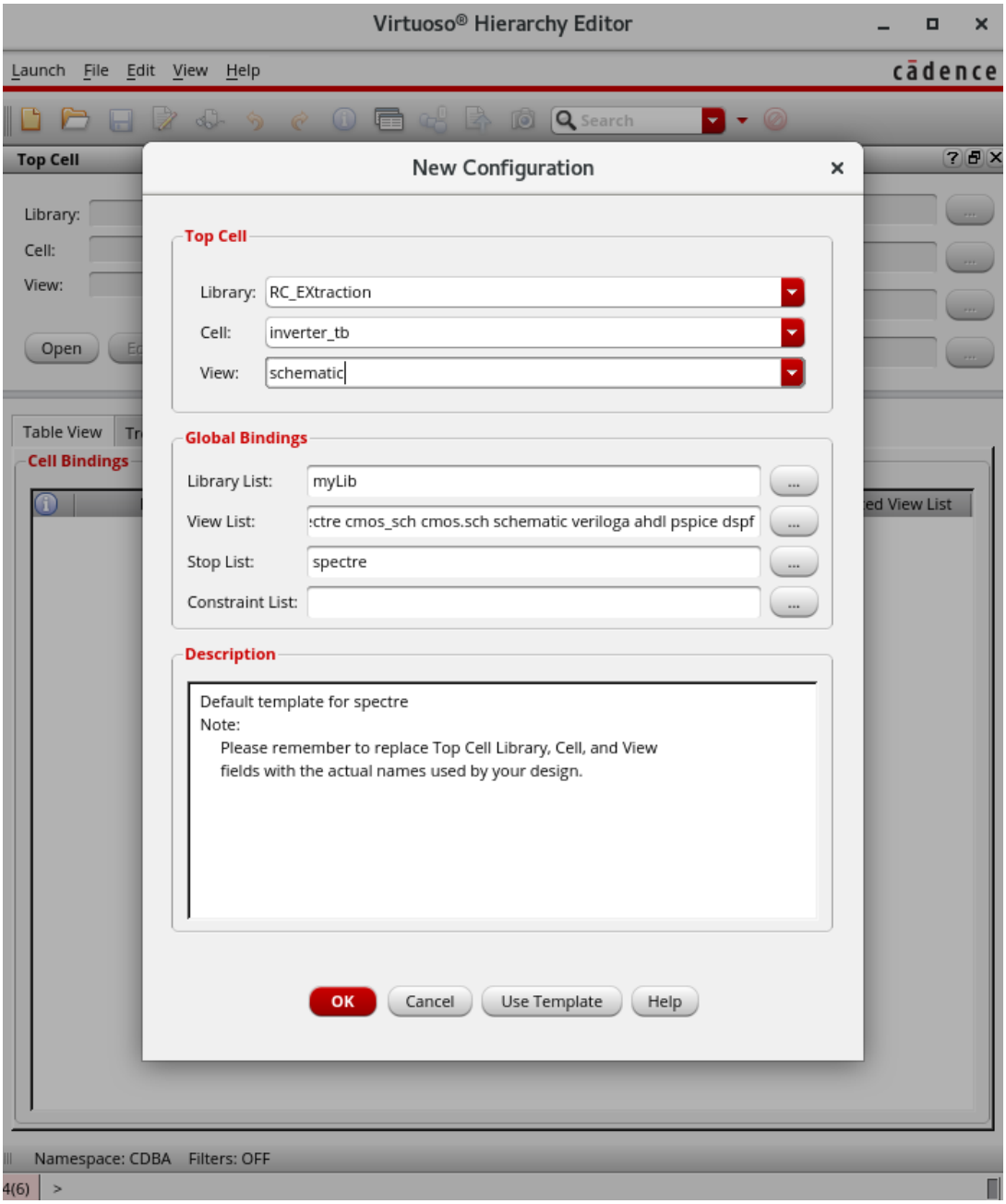

Figure 16: Finalizing the configuration by setting the view to schematic.

3.6 Using the Tree View Tab

After configuring the hierarchy, navigate to the **Tree View** tab.

1. Open the testbench schematic alongside the Hierarchy Editor window.
2. Select an instance in the **Tree View** tab. The corresponding instance in the testbench schematic will be highlighted automatically.
3. Right-click on each instance, then navigate to **Set Instance View** and select the appropriate view for each instance (Figure 18).

Important Note

The inverter was instantiated three times:

- One for schematic simulation.
- One for post-layout simulation without parasitic elements.
- One for post-layout simulation with parasitic elements.

4. Press **Save** and close the Virtuoso Hierarchy Editor window.

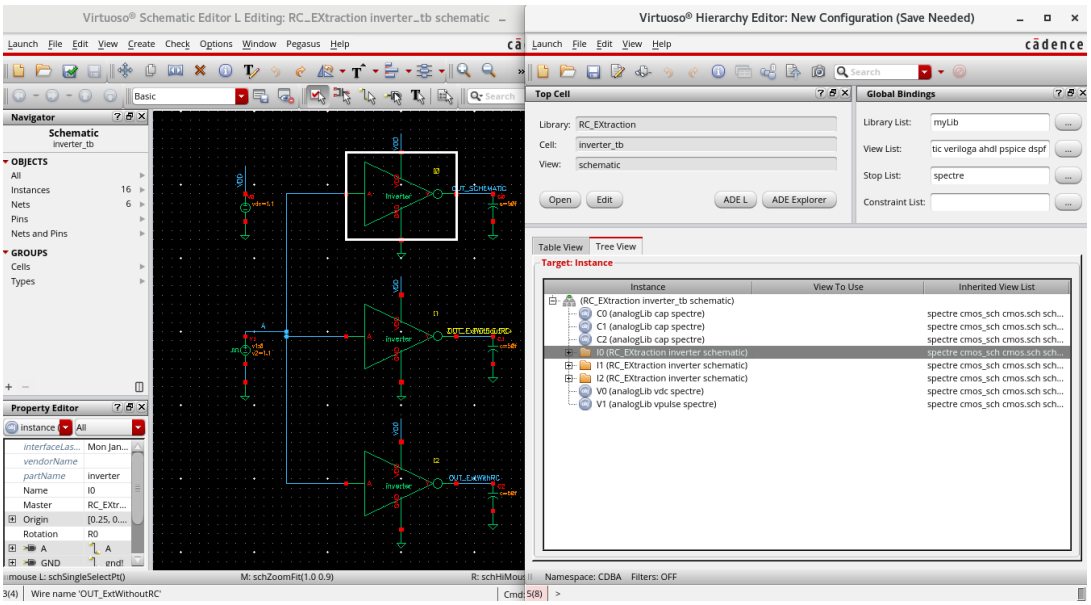

Figure 17: Tree View tab in Virtuoso Hierarchy Editor showing instance selection.

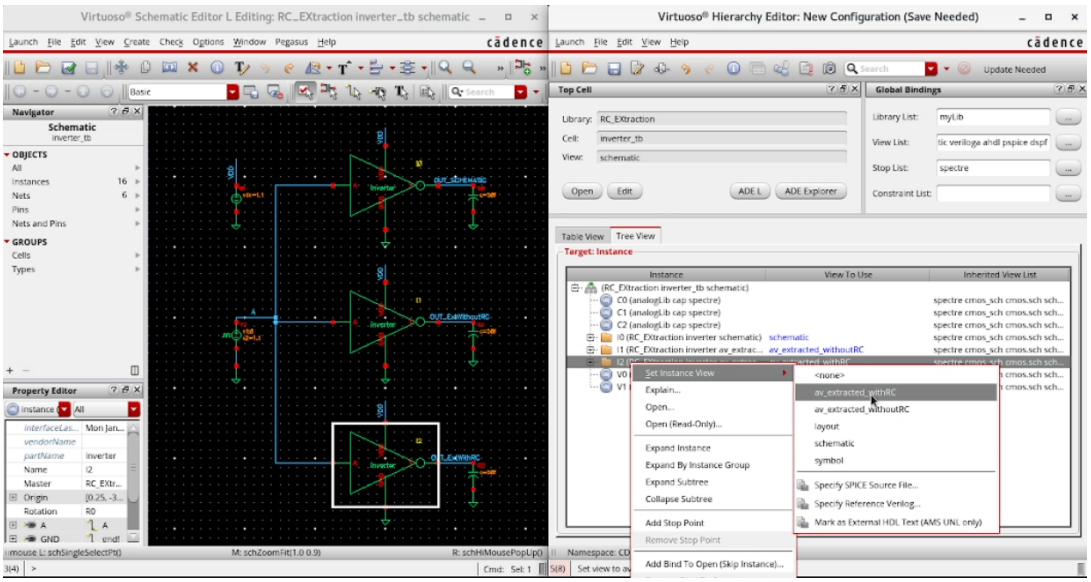

Figure 18: Set view to use for simulation.

4 Post-Layout Simulation

After extracting the layout and configuring the testbench, the next step is to perform a post-layout simulation to analyze the behavior of the circuit w/o parasitic elements.

4.1 Opening ADE and Setting Up the Simulation

To set up the simulation in Virtuoso Analog Design Environment (ADE L):

1. Open the
   textbfLibrary Manager and navigate to the testbench cell (e.g., `inverter_tb`).
2. Open the `config` view of the testbench by double-clicking it.
3. When prompted, select **Yes** for both configuration options to ensure the correct hierarchical view is used.
4. In the testbench schematic window, open ADE L by selecting
   textbfLaunch → ADE L from the menu.

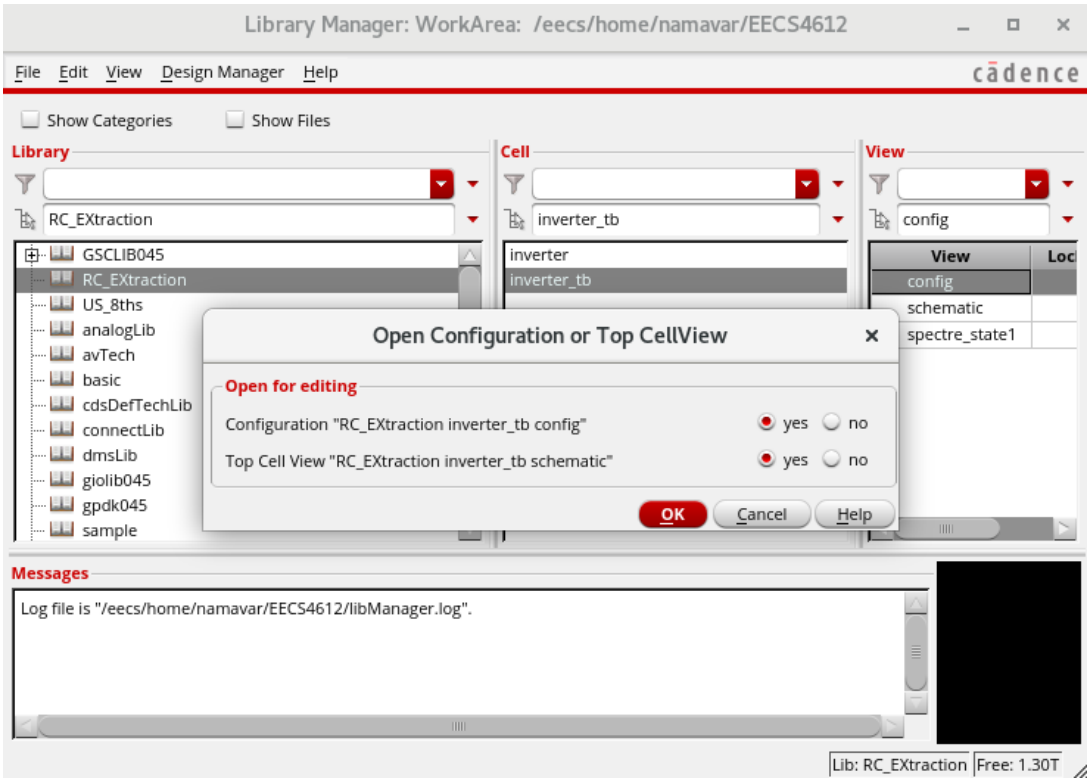

Figure 19: Opening the testbench `config` view and selecting the correct configuration settings.

4.2 Configuring the Simulation in ADE L

Once ADE L is open, configure the required analyses and output signals for post-layout verification.

1. In the
   textbfAnalyses section, add:
   - **Transient Analysis (tran):** Set the stop time to `50n` and use a conservative accuracy mode.
   - **DC Analysis (dc):** Enable it by selecting the appropriate setting.
2. In the
   textbfOutputs section, add the following signals for plotting:
   - `OUT_SCHEMATIC` - Output of the schematic simulation.
   - `OUT_ExtWithoutRC` - Output of the extracted view without parasitic RC elements.
   - `OUT_ExtWithRC` - Output of the extracted view with parasitic RC elements.
   - `A` - The input signal applied to all instances.
3. Ensure that the signals are marked for plotting by checking the **Plot** box next to each signal.

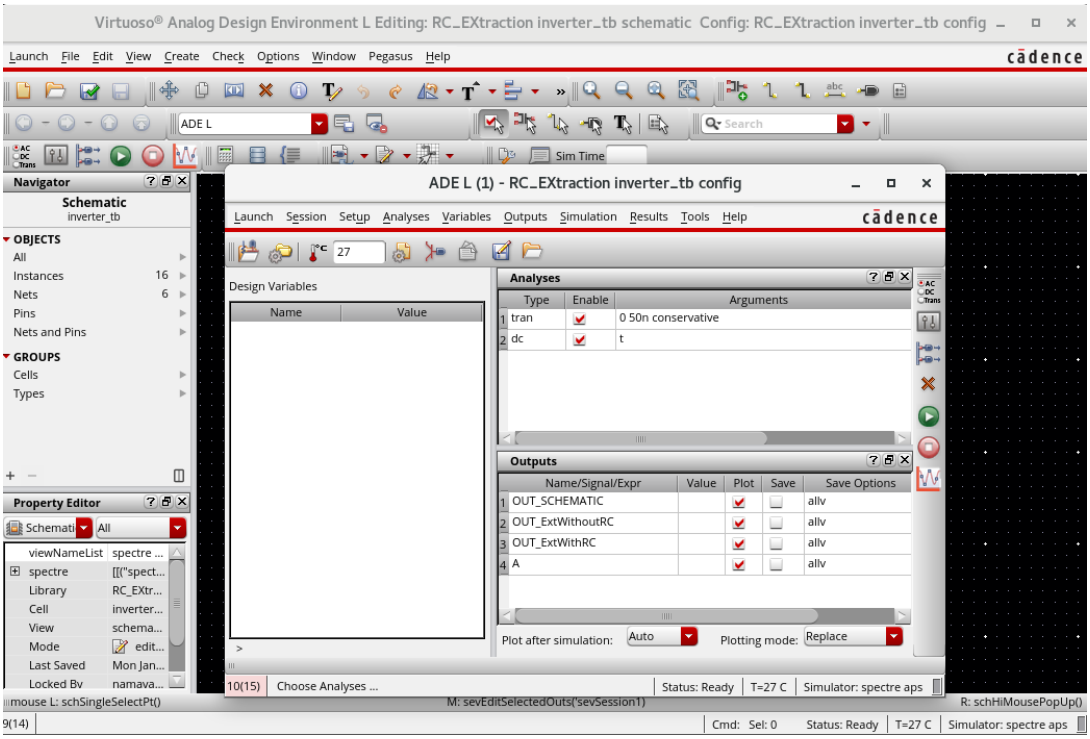

Figure 20: Configuring the ADE L window for post-layout simulation. Transient and DC analyses are added, and key signals are selected for plotting.

Once these steps are completed, the setup is ready for simulation. The next step will be to run the simulation and analyze the results.