EE 266 Digital Synthesis

Drew Hall (drewhall@ucsd.edu)

ECE 266 – Integrated Circuit Laboratory

UC San Diego

Background

For this lab, you will be given synthesis deck

- Based on Cadence Tools (Genus, Innovus)
- Physically aware synthesis

The provided Verilog code contains

- SAR algorithm
- Delay generation for asynchronous clocking

Structure

Folder structure:

- Design → Contains both design and output of the tool
- Scripts → Synthesis deck
  - You don’t need to touch it unless you need to improve it
- TMP → Middle file for the synthesis
- CLEANUP.sh → Remove all the generated files
- RUNME.sh → Synthesis Initiator…

Design Folder

Three base folders

- Config
  - IO, Timing, and Routing configuration
- GDS
  - Final layout will be stored in this folder
- HDL
  - A place for both behavioral and synthesized Verilog code

Two auxiliary folders (mid step verifications)

- Synthesis
- PSynthesis

Config Folder

For the interest of this lab, you need to modify two files

- DesignConfig.cfg
  - To change the used metals for routing the sizing your floorplan
- SARController.io
  - To place the pins of the layout at the best place for your routing

The rest of files

- *.def → Generated based on you IO file and Area in the config
- *.mmmc → Configures the library, corners, etc…
- *.sdc → Configures timing

Upon Synthesis

The synthesized block needs to be imported in virtuoso

- Importing the Verilog netlist for schematic (Design/HDL)
- Importing the GDS file for layout (Design/GDS)

Verilog For Schematic

Convert Verilog code to Schematic (utilizing Verilog In)

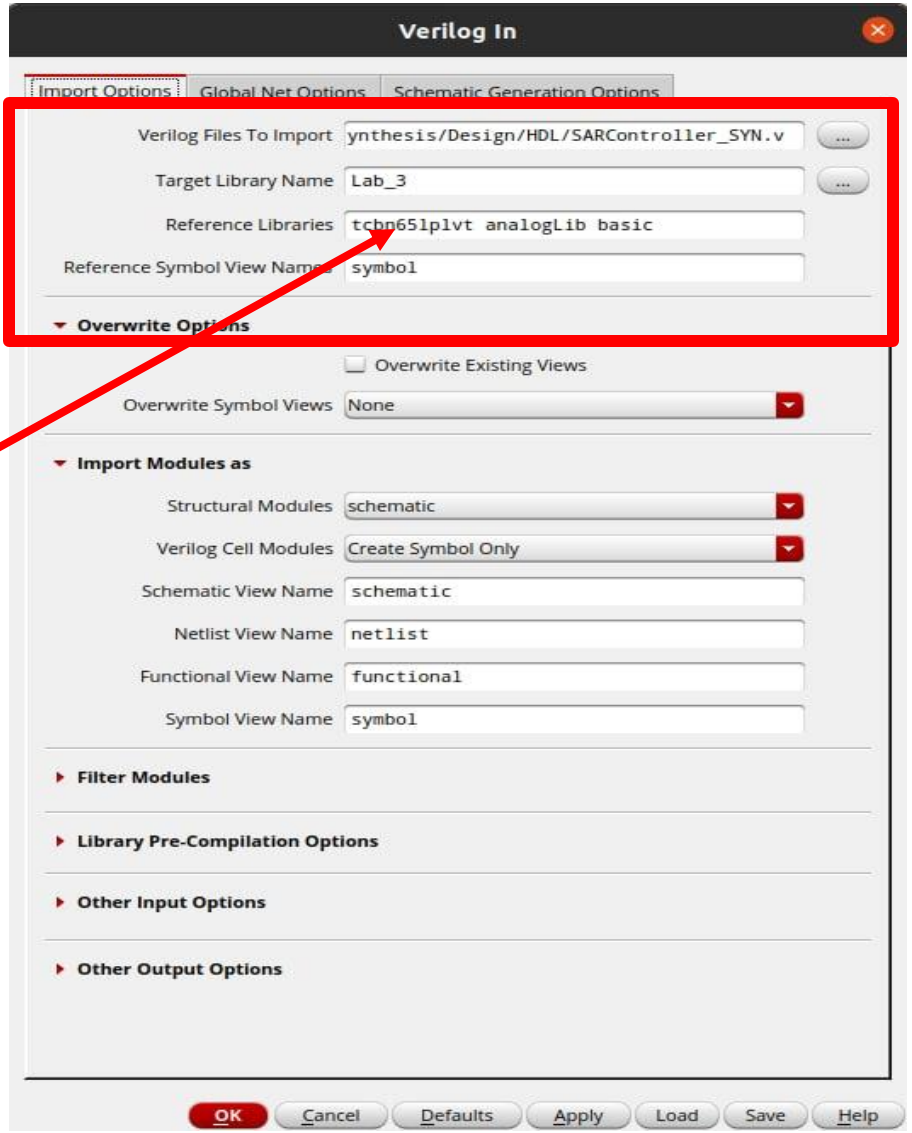
- Rather than converting Verilog to Spice
- SARController_SYN.v for Schematic

- Prior to importing the Verilog in the next step, you need to modify the name of the module
  - Open Design/HDL/SARController_SYN.v and change the module name from SARController to SARController_SYN (just the module name at the top)

Verilog For Schematic (cont.)

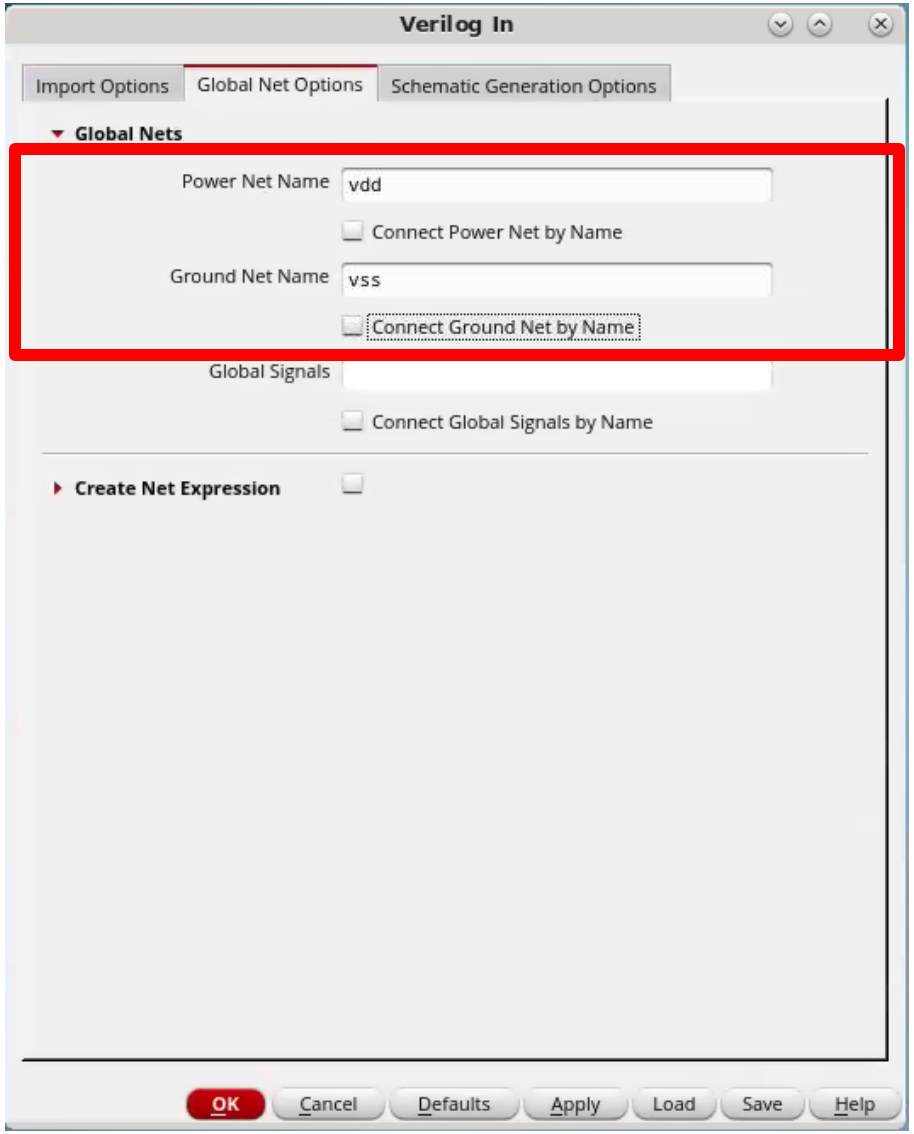
From CIW>File>Import>Verilog

Many configuration can be decided by you like full place and route



For TSMC65 GP, this should be tcbn65gplus

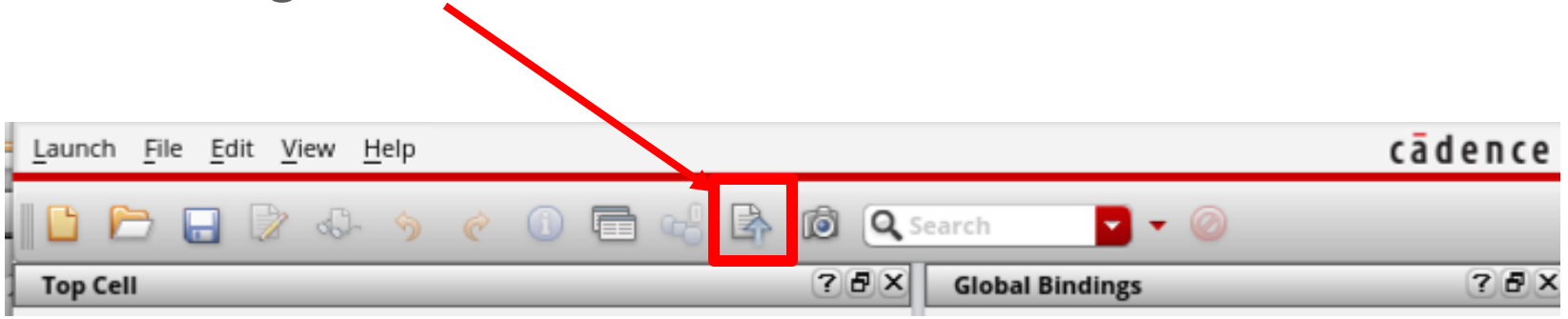
<u>NOTE:</u> This defaults to NOT overwrite the cell, so you need to manually delete SARController_SYN if you are re-importing it.



Simulating with the Synthesized Verilog Code

- After importing the Verilog schematic, it will create a symbol.
  - But… this doesn't match the symbol we made! So the pins won't line up!
  - Open Library Manager, go to SARController->symbol and copy this into SARController_SYN (overwriting the symbol the tool autogenerated)

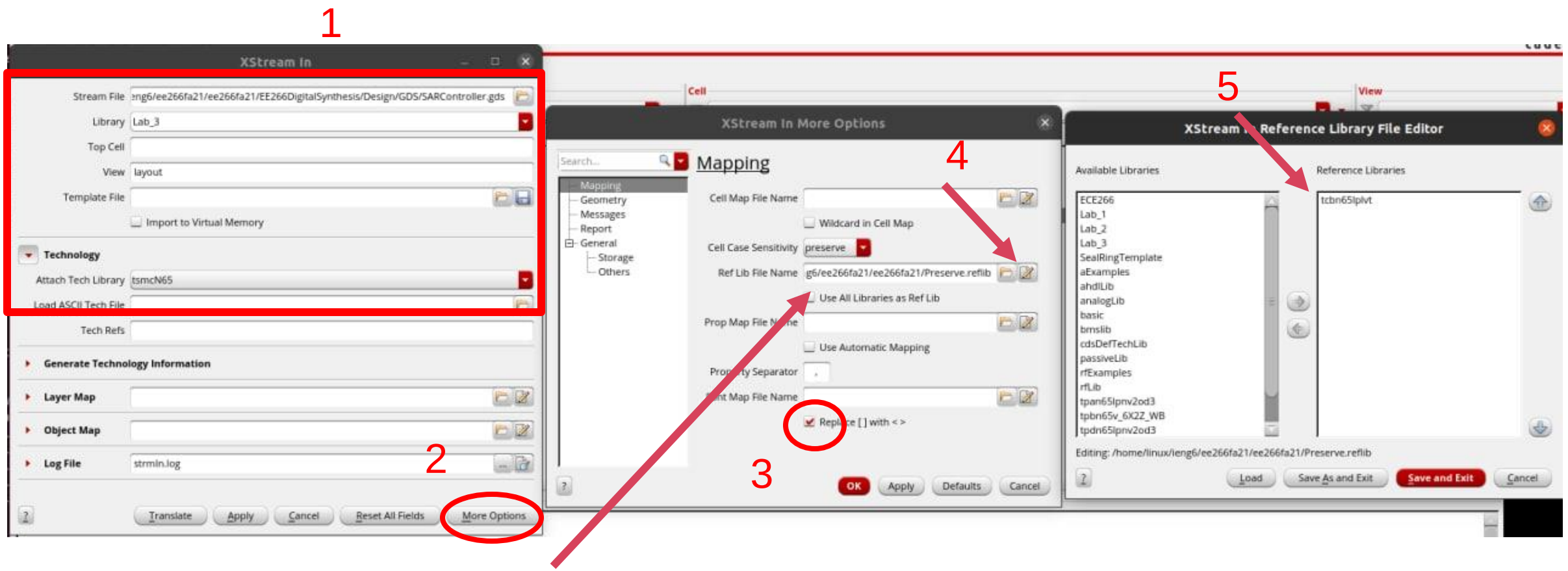
- To use the new code, open the SAR_ADC schematic and click on SARController. Change the properties (hit q) to update the cell name to SARController_SYN.
- Refresh your Config View and then simulate with ADE L



GDS For Layout

- From CIW>File>Import>Stream
  - Import the
    - SARController.gds for layout
  - Make sure
    - You selected the Tech Library (tsmcN65)
    - From “More Options” menu
      - Preserve the “tcbn65gplus” library cells
    - Otherwise, your layout wouldn’t have the correct cells.

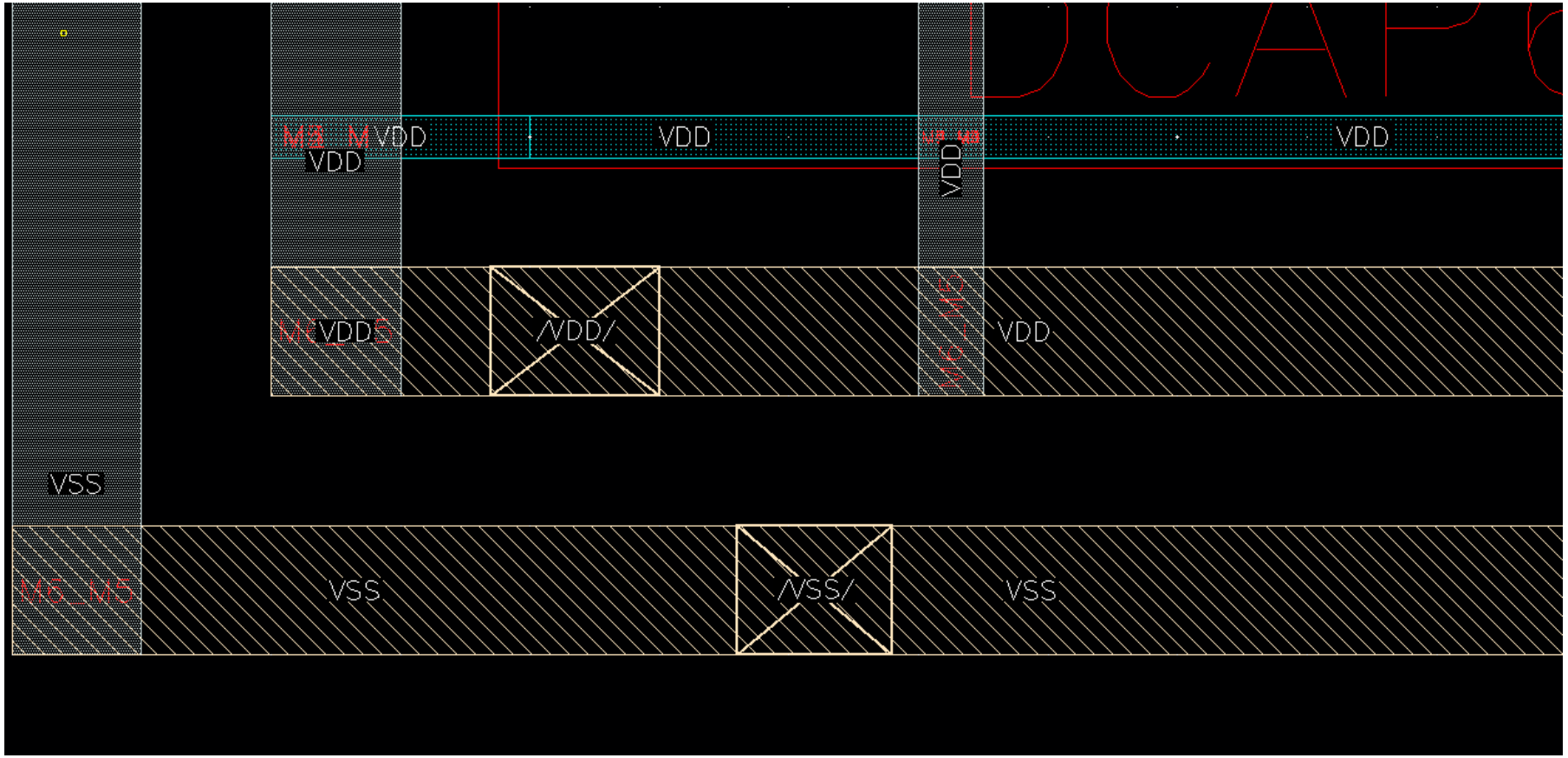
GDS For Layout (cont.)



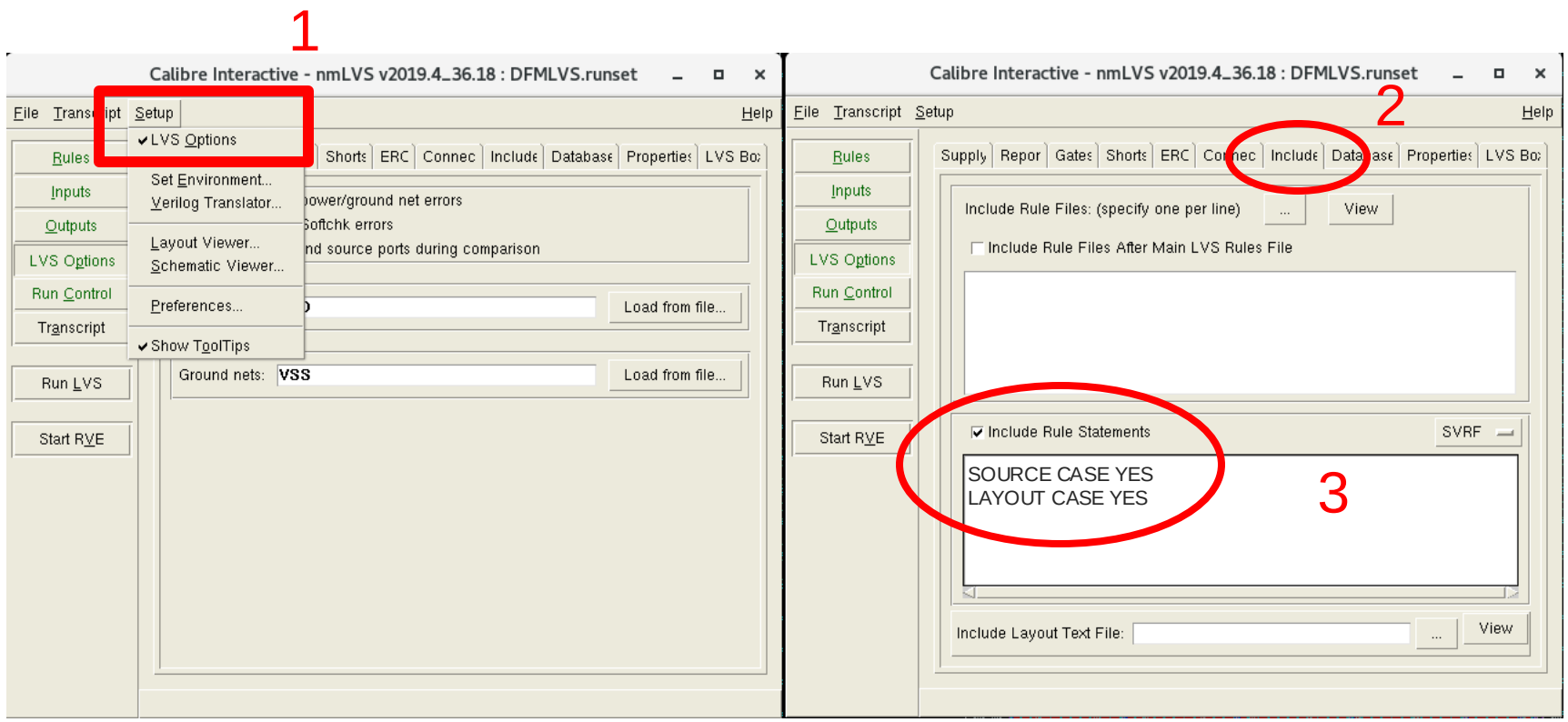
Once done you can save the preserve configuration for later use. I saved it as “Preserve.reflib”

Power Ports

- After importing the GDS, you need to manually add the power ports to the layout



Case Setting for LVS



If your pin names are written in noncapitalized format and run in LVS errors due to this cause, follow the above setup to add case insensitive rules to LVS.