Incorporation into VLSI Design Flow

- **Purpose of Extraction**
- **Transistor Level Design**
- **Cell Based Design**
- **Clock Skew Analysis**
- **Cross-talk Analysis**
- **Reliability Analysis**

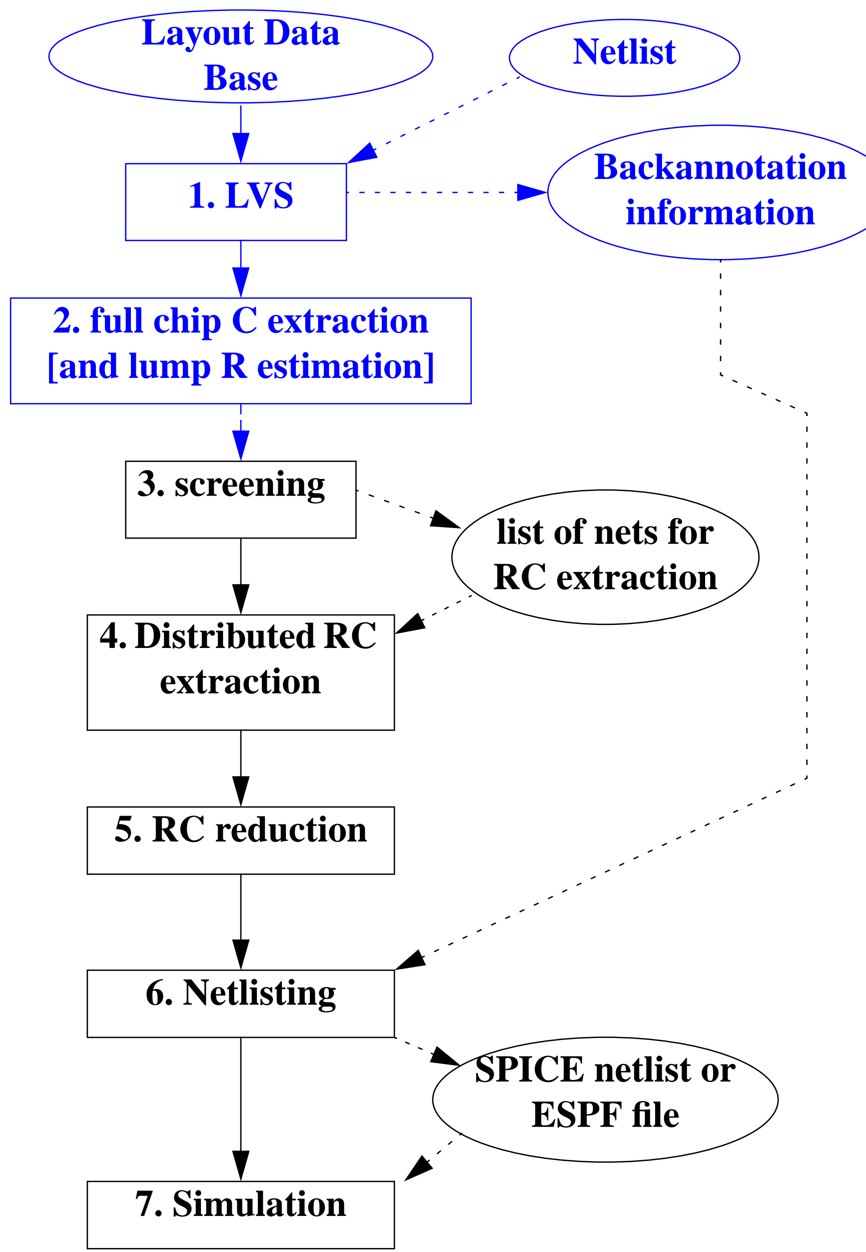
❑ Purpose of Extraction

- Support the simulation needs
  - Extraction by itself is of no interest
- Goals
  - Accurate simulation results (e.g. timing) with parasitic effects
  - Effective usage of computing resource, cpu time, memory, disk space
  - Fit within the simulator's capacity
  - Simple and straight-forward extraction process
  - Provide good trade-off of accuracy versus performance

❑ Transistor Level Design

- A generic transistor level design flow is first discussed with the following assumptions
  - The netlist for the design is available
  - The transistor level simulator can simulate the whole circuit
  - Back-annotation to schematic net-name is required
- Modification of the flow to support:
  - critical path simulation
  - static transistor level timing simulator

Transistor Level Design Flow - Generic



1. LVS step is to generate the layout to schematic name-mapping. This information is used later.

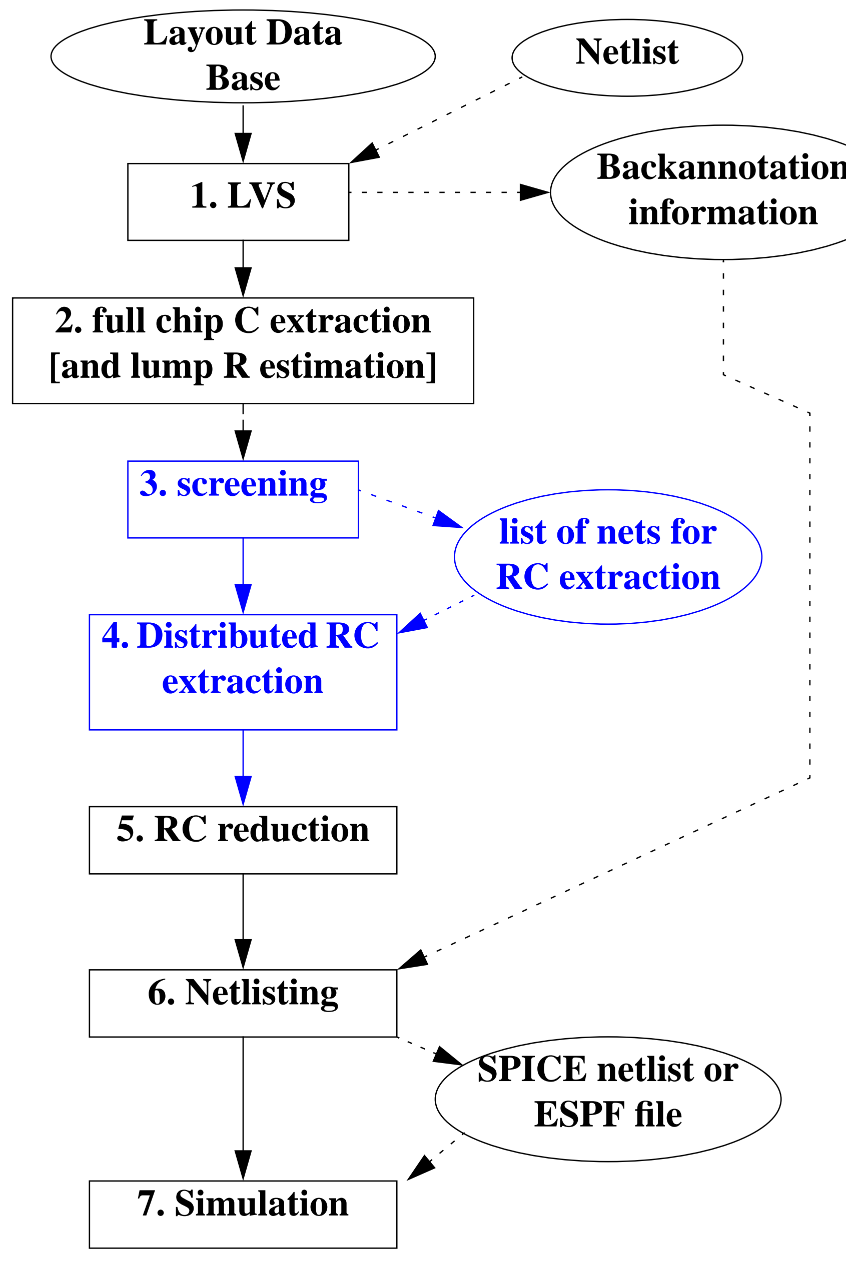
2a. The brute force approach to generate detailed RC will use too much cpu resource and too large an output database.

2b. A screening step will be applied later (in step 3) to identify the nets which needs to be extracted with distributed RC model.

2c. The lump C value for each net is needed for timing simulation. This does not increase the simulation time for most timing simulators (e.g. SPICE).

2d. The lump R estimation is optional, depending on the user's preference in the screening step.

Transistor Level Design Flow - Generic (cont.)



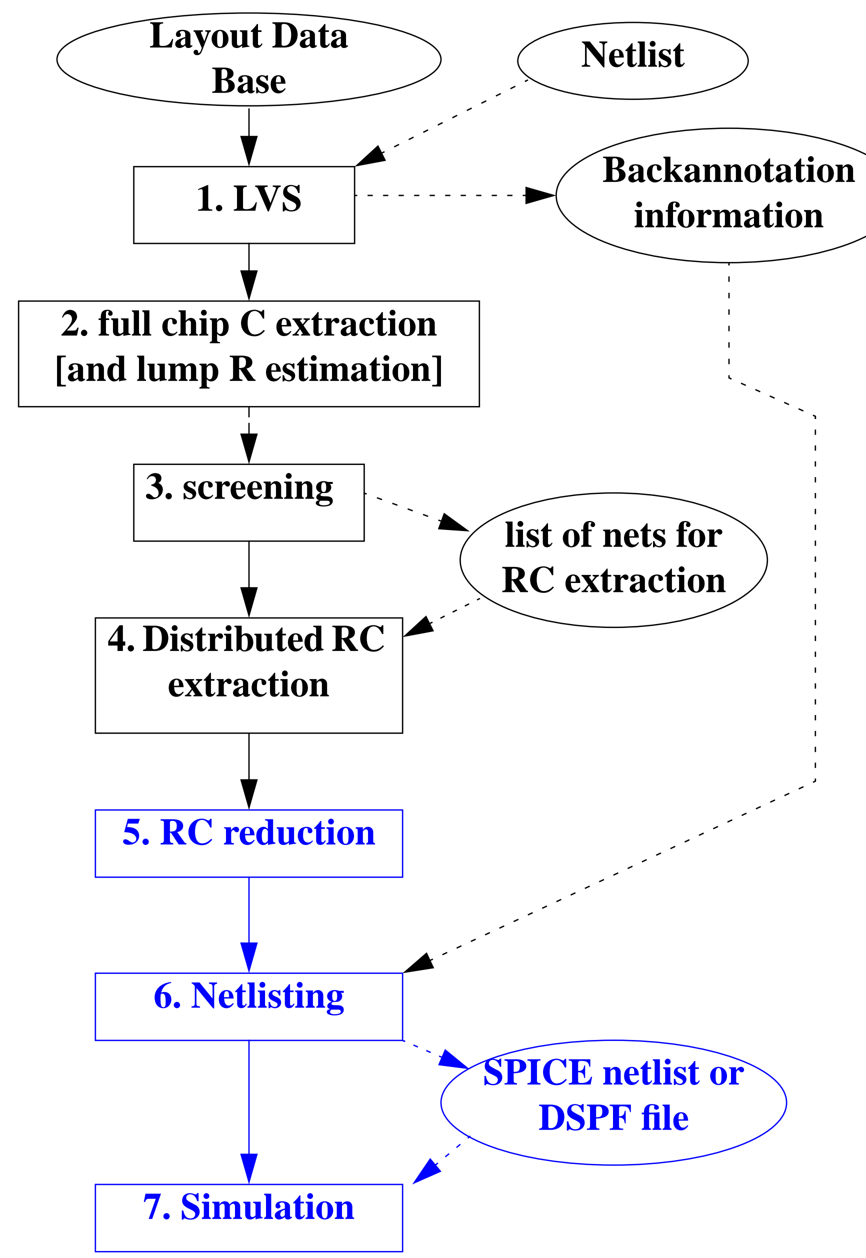
3a. The screening step is used to identify the nets which need to be extracted with distributed RC.

3b. The screening can be based on capacitance value, lump R value, lump RC (estimated time-constant) or lump RC relative to the strength of the driving transistors.

4a. The identified nets will then be extracted for distributed RC.

4b. Typically, less than 5% of the nets are identified. The rest of the nets can be represented by lump C values. Both cpu time and disk space are minimized without causing inaccuracies.

Transistor Level Design Flow - Generic (cont.)



5a. There can be many RCs for each of the distributed RC nets. This can cause capacity and/or slow simulation time. RC reduction can be used to reduce the number of RC elements with small sacrifice in accuracy.

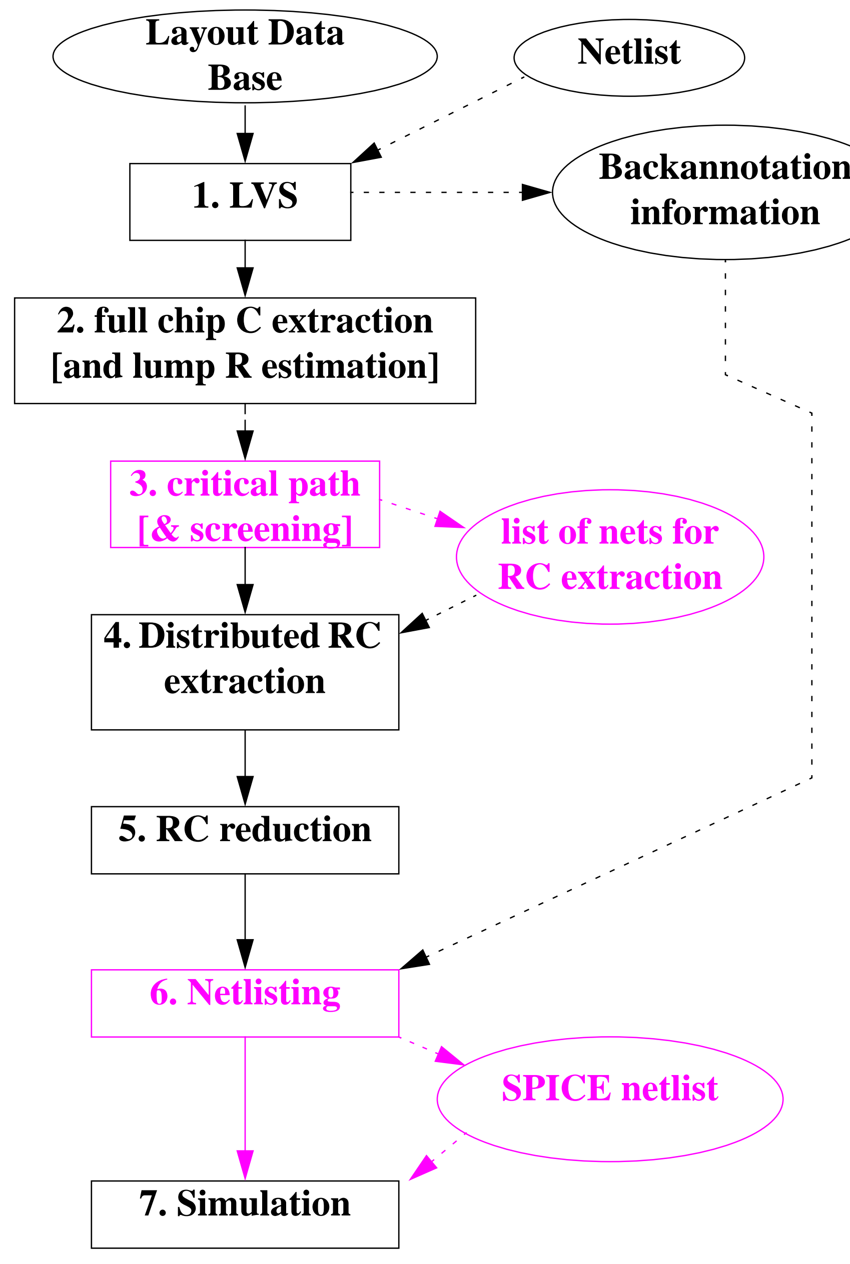
5b. RC reduction of 50% - 95% can be achieved while maintaining accuracy within 5%.

6a. Either SPICE netlist or DSPF file can be the output from the netlisting step.

6b. Standard Parasitic Format (SPF) is a commonly used format. The detailed mode (DSPF) is used for distributed RCs.

6c. Standard Parasitic Exchange Format (SPEF) is undergoing IEEE standardization effort. IT should be available this year.

Transistor Level Design Flow - Critical Path



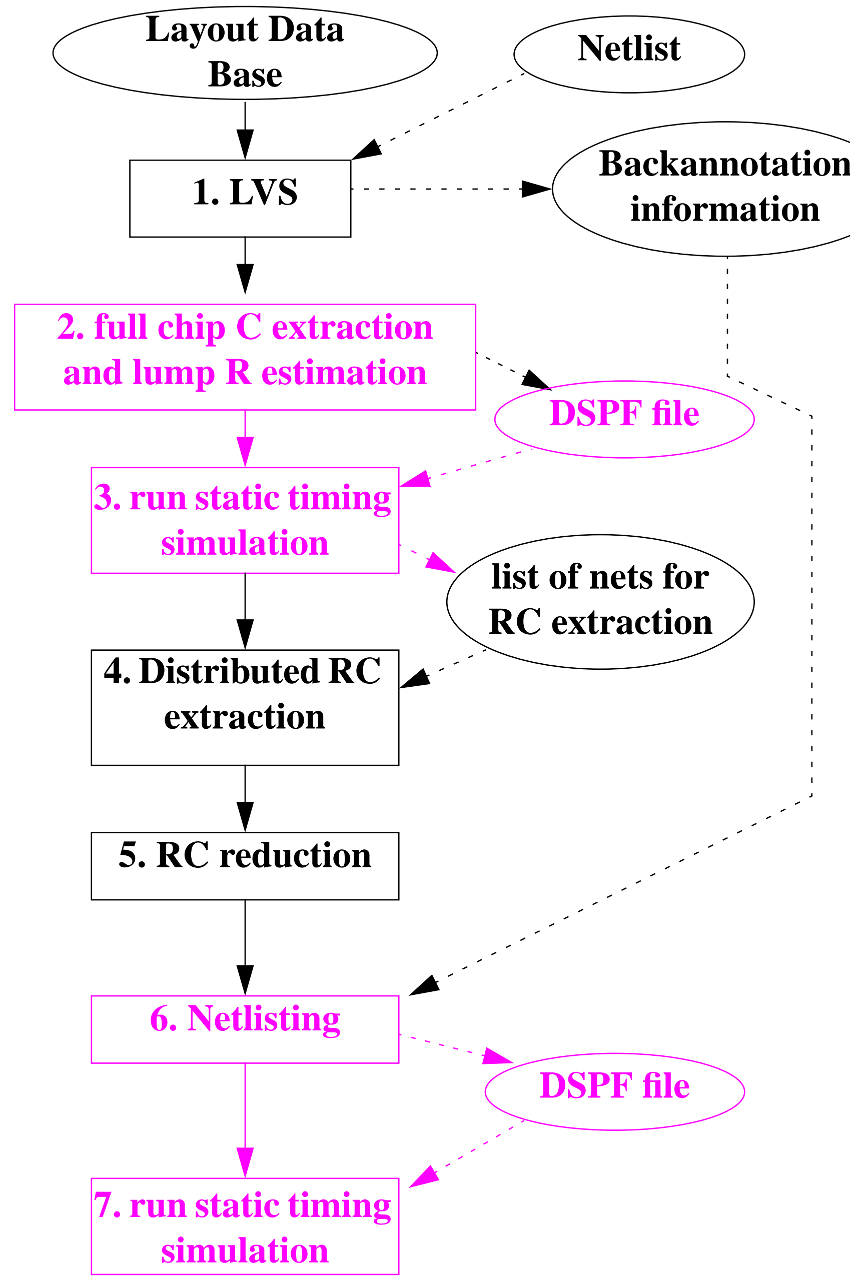
3a. Once the critical paths are identified, the nets on the critical paths can be extracted with distributed RC.

3b. Screening can optionally be performed on the nets along the critical path.

3c. The critical path approach can significantly reduce the number of nets that needs to be extracted for distributed RC.

6. For the critical path flow, SPICE netlist is typically generated since simulation of the other nets are not needed.

❑ Transistor Level Design Flow - Static Timing



2. The lump C and estimated lump RC are placed in a DSPF file

3a. The static timing simulator will use the net capacitance to derive the critical paths.

3b. Typically, twice as many critical paths will be identified.

3c. Based on the lump RC estimate, the delay error for each net in the critical paths is estimated. If the error is larger than a predetermined percentage, the net will be extracted for distributed RC.

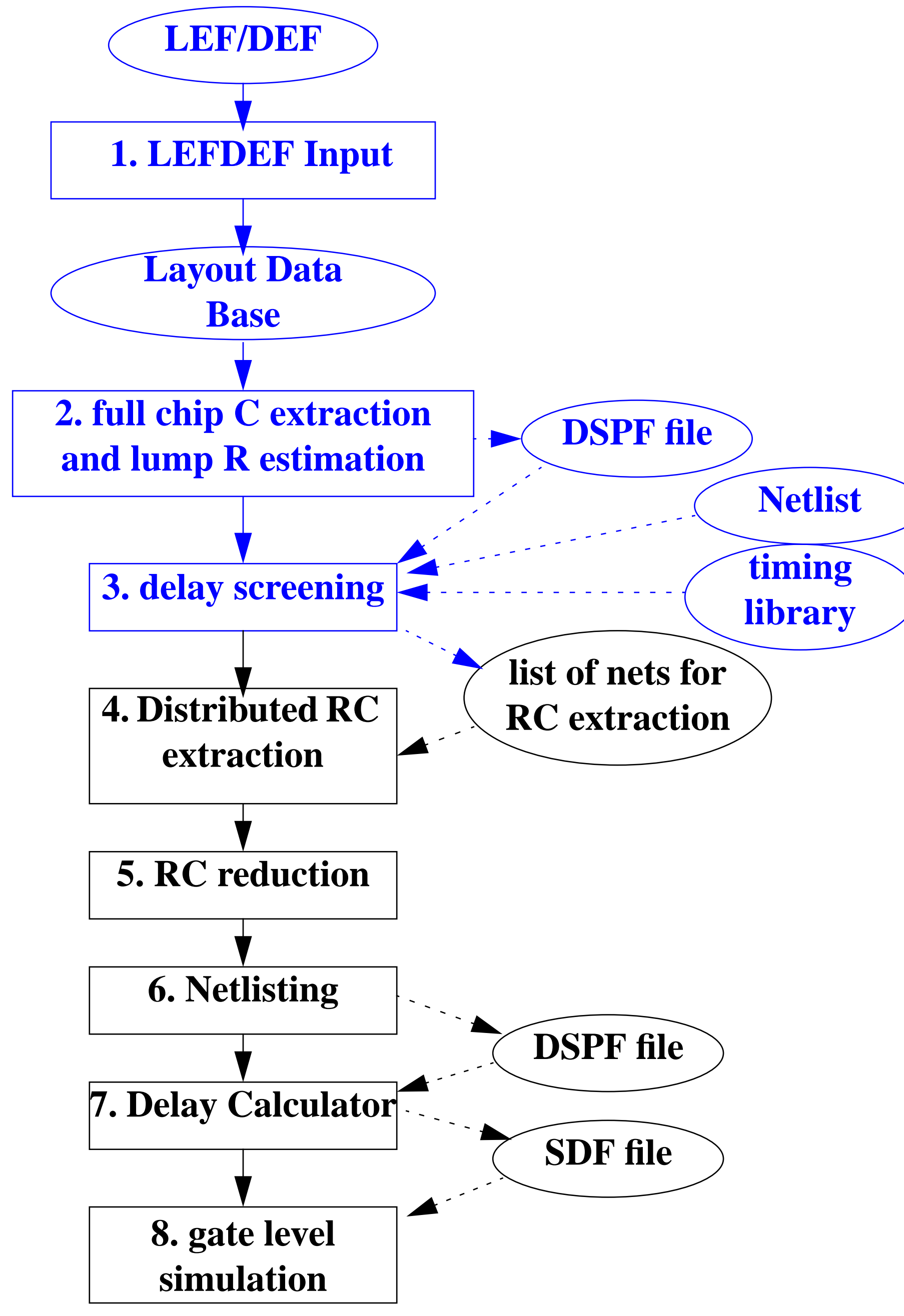
6a. Another DSPF file will be generated as input to the static timing simluator for accurate timing simulation.

6b.Lump C and distributed RC are placed in the DSPF file.

❑ Cell Based Design

- The cell based design flow is presented with the following assumptions
  - The layout is generated by a place-and-route tool.
  - The gate level netlist for the design (e.g. Verilog or VHDL netlist) is available.
  - Timing library for the cells is used.
  - SPICE netlists for the cell library.

Cell Based Design Flow

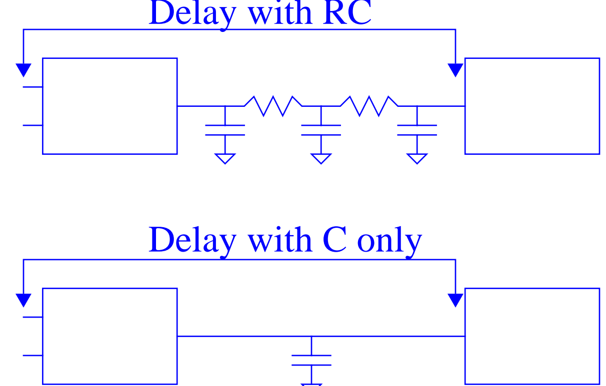


1. LEFDEF input to generate layout data base from LEF/DEF.

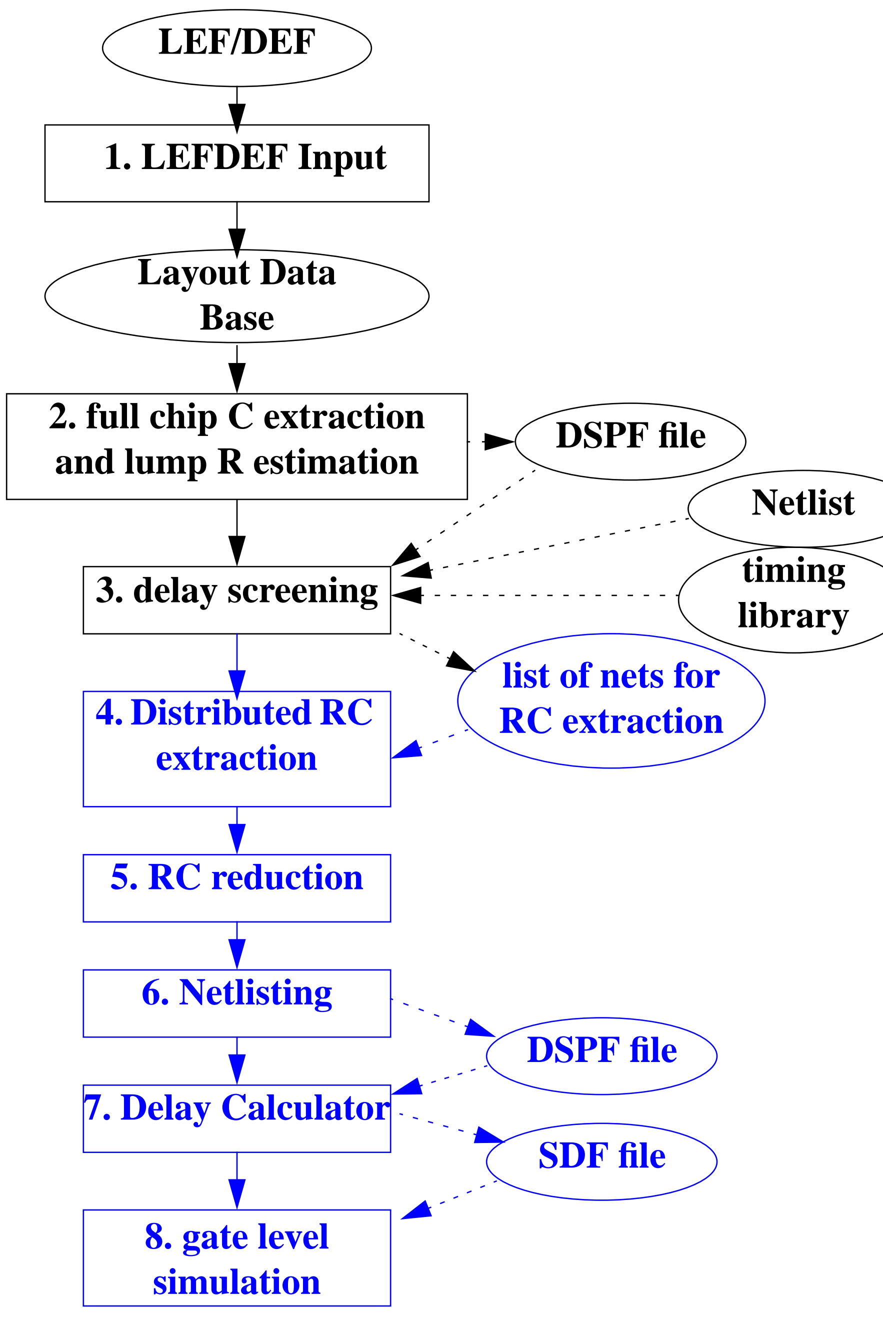
2. The lump C value for each net and the estimated R value are placed in the DSPF file.

3a. Delay screening will identify the nets which will need distributed RC model in order to achieve a pre-defined level of accuracy (e.g. < 2%).

3b. Typically, less than 5% of the nets will need distributed RC model.



Cell Based Design Flow (cont.)



4. The identified nets will be extracted for distributed RC.

5. RC Reduction step is used to reduce the number of RCs per net while maintain acceptable level of accuracy.

6. DSPF file is generated by the Netlister.

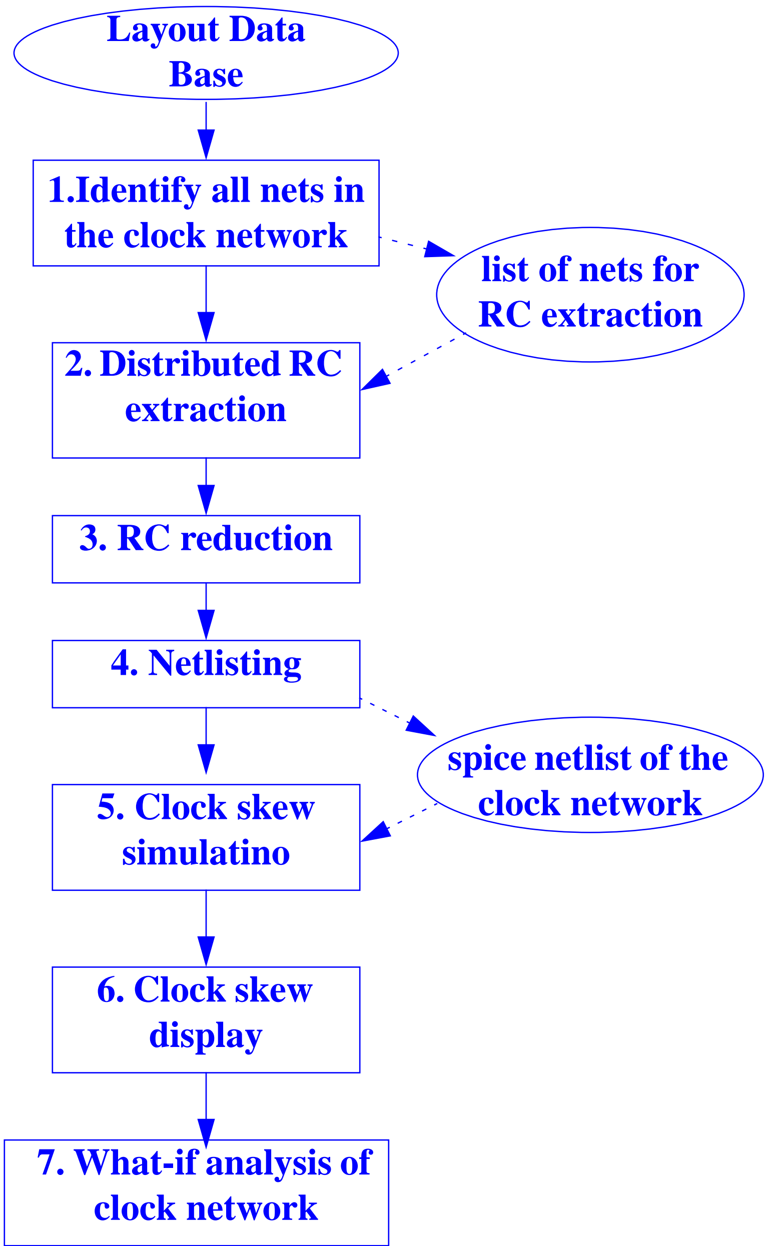
7. Delay calculator is used to derive the delay for each path and then place the output in the Standard Delay Calculator (SDF) file.

8. Gate level simulation performed with delays specified in SDF.

❑ Clock Skew Analysis

- The clock network is very important in determining the performance of the design.
- In order to minimize clock skew (clock skew is defined as the difference in time between the fastest clock edge and the slowest clock edge for the same clock period), clock tree (and sometime clock mesh) structures are used.
- Since buffers are used, a simple method to identify all the nets in the clock network is needed.
- The clock network should be displayed such that the clock skew values can be superimposed on the layout.
- A what-if analysis environment should be provided for user to move the clock network and observe the effect on clock skew.

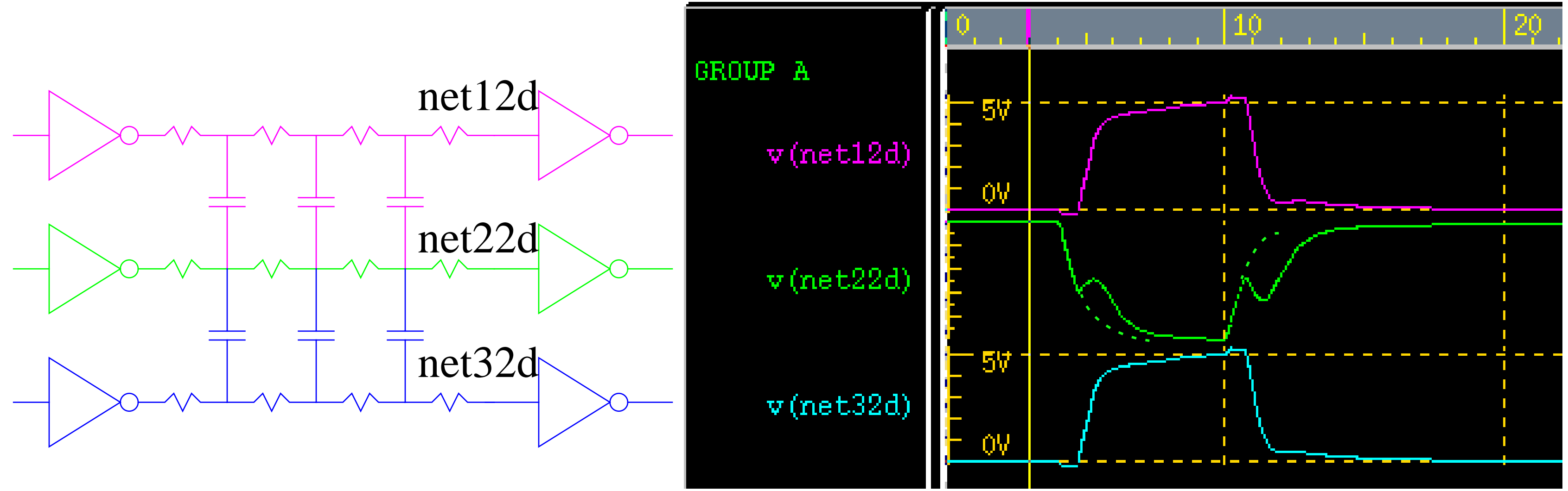
Clock Skew Analysis Flow



1. Trace all nets of a clock network from pad through all buffers and gated buffers.
2. All nets in the clock network will be extracted with distributed RC.
3. RC reduction is used to minimize the size of the circuit.
4. Spice netlist will be generated by the Netlister.
5. The clock network will first be simulated and then the clock skew values will be derived from the simulation results.
6. The clock skew value will be displayed on top of the layout.
7. What-if analysis environment for the user to study the effects of changing the clock network on clock skew.

❑ Cross-talk Analysis

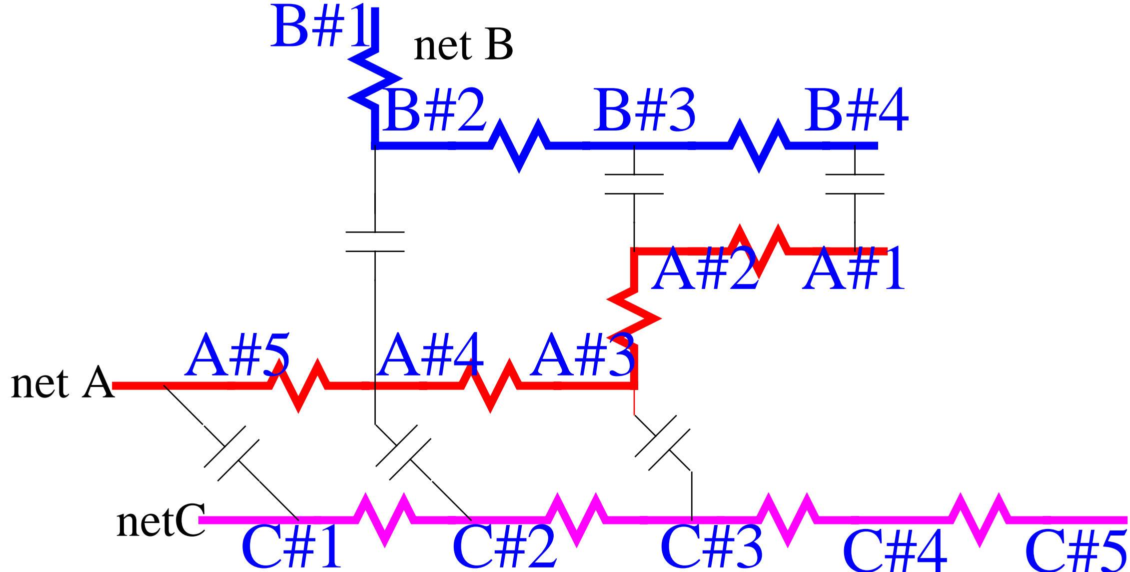
- Cross-talk can significantly affect the performance of a design



- Cross-talk is mainly due to cross-coupling of the nets.
- net22d is the victim net
- net12d and net32d are the aggressor nets
- Delay cannot be modeled by converting the CC capacitance to grounded capacitance with twice the capacitance value
- Delay in the victim net can sometimes be improved by slowing down the transition for the victim net.

❑ Cross-talk Analysis - Simulation

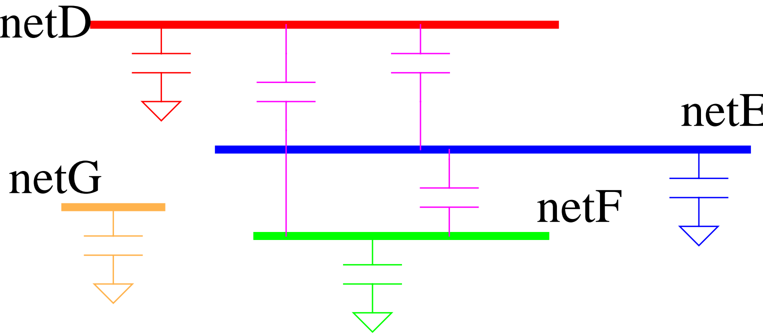
- Extract distributed cross-coupling capacitances



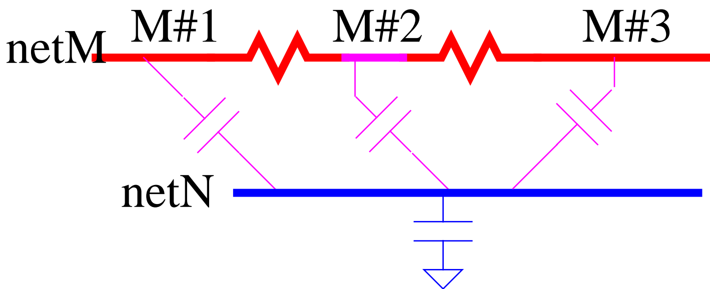
- Simulator to identify the appropriate simulation model
  - signal edges do not interact => model with grounded capacitance
  - signal edges interact slightly => model with Miller equivalent capacitance
  - signal edges interact strongly => simulate the cross-coupling capacitances directly

❑ Cross-talk Analysis - Modeling

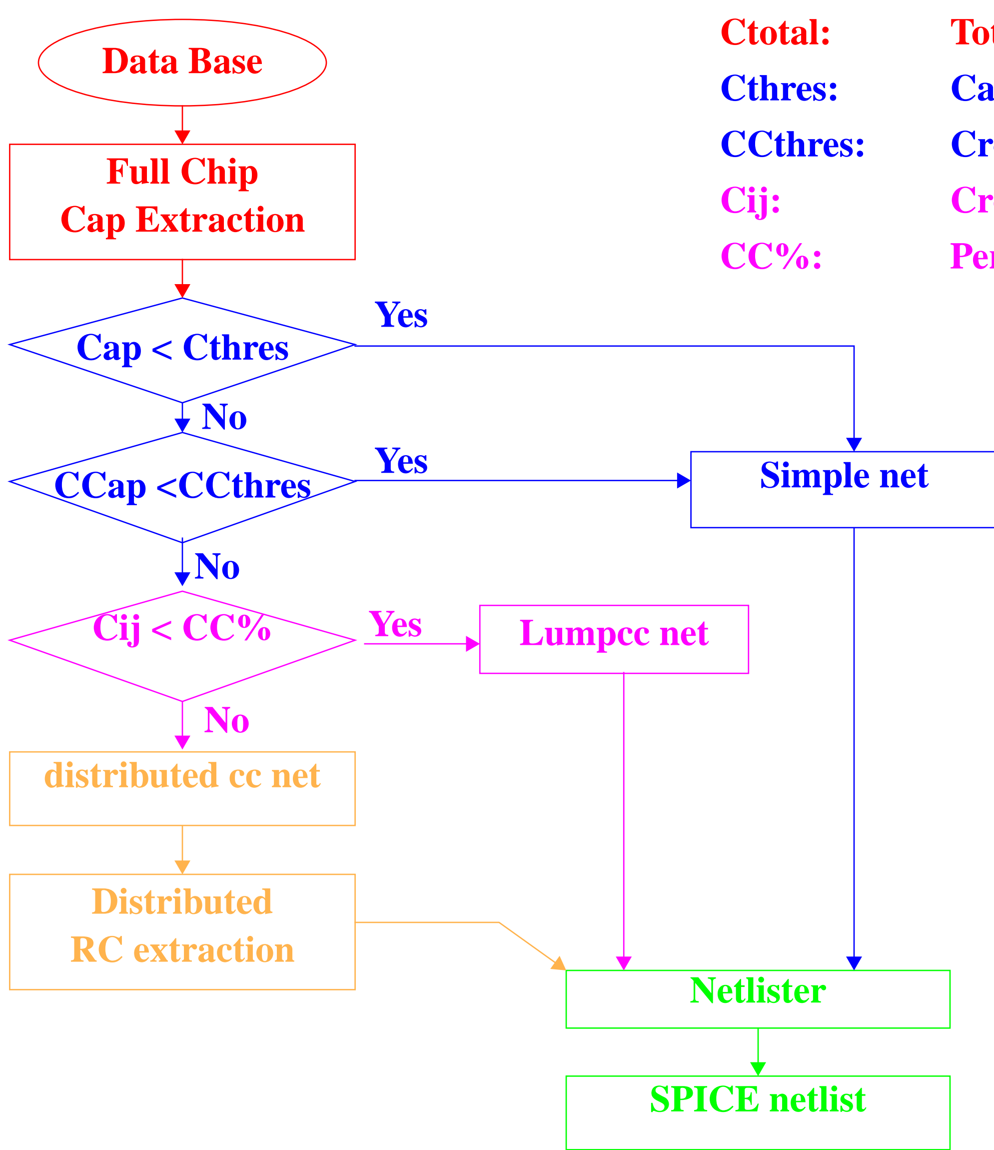
- Distributed cross-coupling extraction of all nets can consume huge amount of disk
- Some nets can be represented by lump cc or simple nets:



- Nets between distributed cc net and lump cc nets:



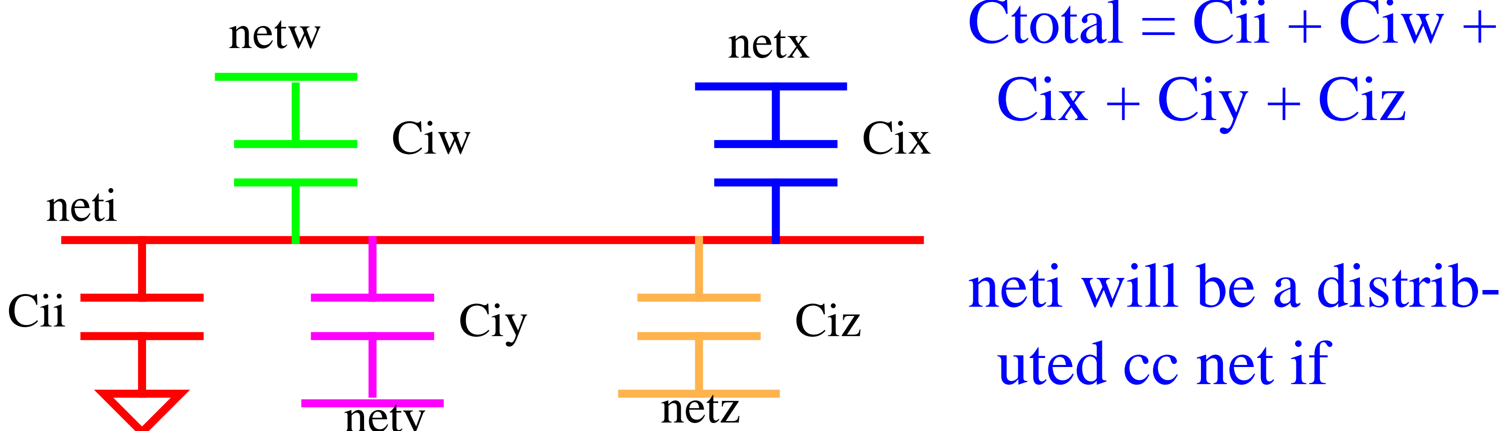
Cross-talk Analysis - Extraction Flow



Ctotal:	Total capacitance of neti
Cthres:	Cap Threshold
CCthres:	Cross-Cap Threshold to each net
Cij:	Cross-Cap between neti & netj
CC%:	Percentage threshold of Cij/Ctotal

❑ Cross-talk Analysis - Extraction Filtering

- Cap < Cthres - total capacitance of the net is less than Cthres
- CCap < CCthres - every cross-capacitance of the net to the other nets is less than CCthres
- Cij < CC% - the percentage of every cross-coupling capacitance of net i to the other nets over the total capacitance is less than CC%



Ctotal = Cii + Ciw + Cix + Ciy + Ciz

neti will be a distributed cc net if

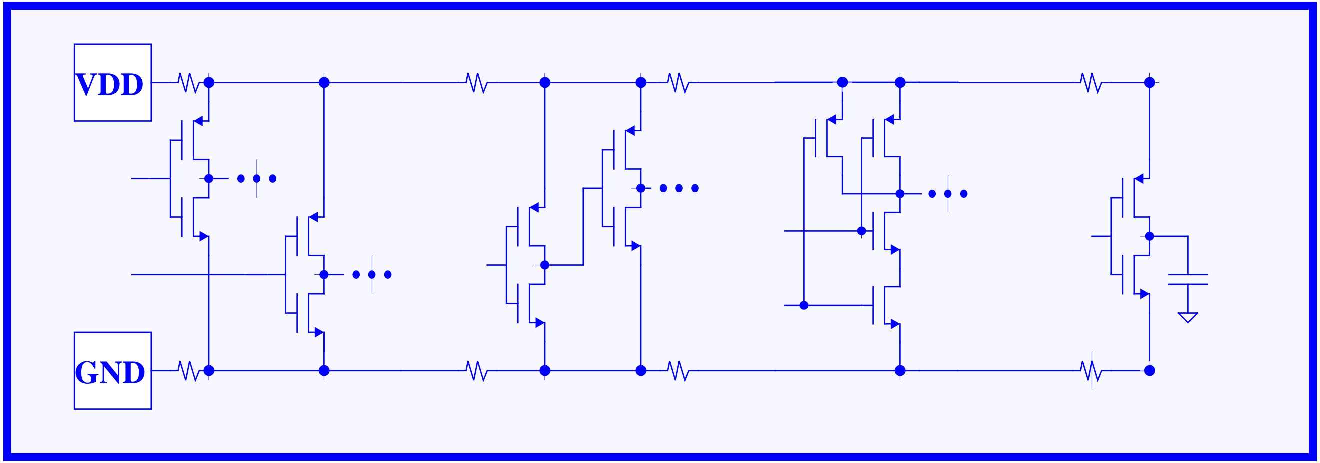
❑ Reliability Issues

- Electromigration
  - IC failure caused by metal fatigue in the power lines due to high current density
- Voltage Drops
  - Decreased supply voltage caused by the current flowing through the resistance of the power supply lines
- Power Distribution
  - Nonuniform power distribution, particularly high power density, can unreliable chip operation.

❑ Reliability Analysis - Simulation Issues

- **Traditional simulation tools are not adequate**

  - Supplies are assumed to be perfect or constant
  - Simplified power model is inadequate

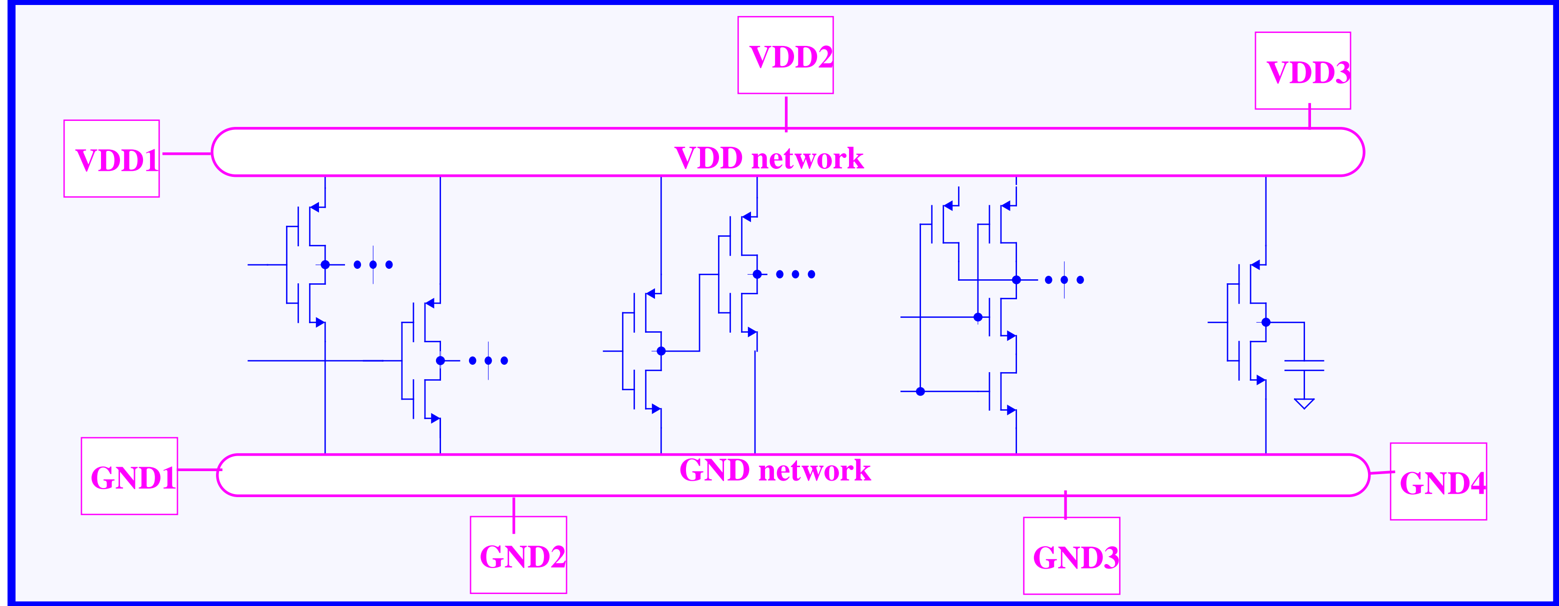


  - Power networks are very complex

❑ Reliability Analysis - Simulation Issues (cont.)

- **The following is needed:**

  - Realistic model for supply network, including physical information
  - Separation of the supply network from the transistor network



❑ Reliability Analysis - Simulation Issues (cont.)

- Transistor currents from the transistor level simulation are applied as current source to the power network

I(M123)

- Simulation and analysis of supply network for voltage drop and electromigration
- Reporting of voltage drop and electromigration violations and display of power distribution
- Meaningful presentation of simulation results

❑ Reliability Analysis - Simulation Implementation

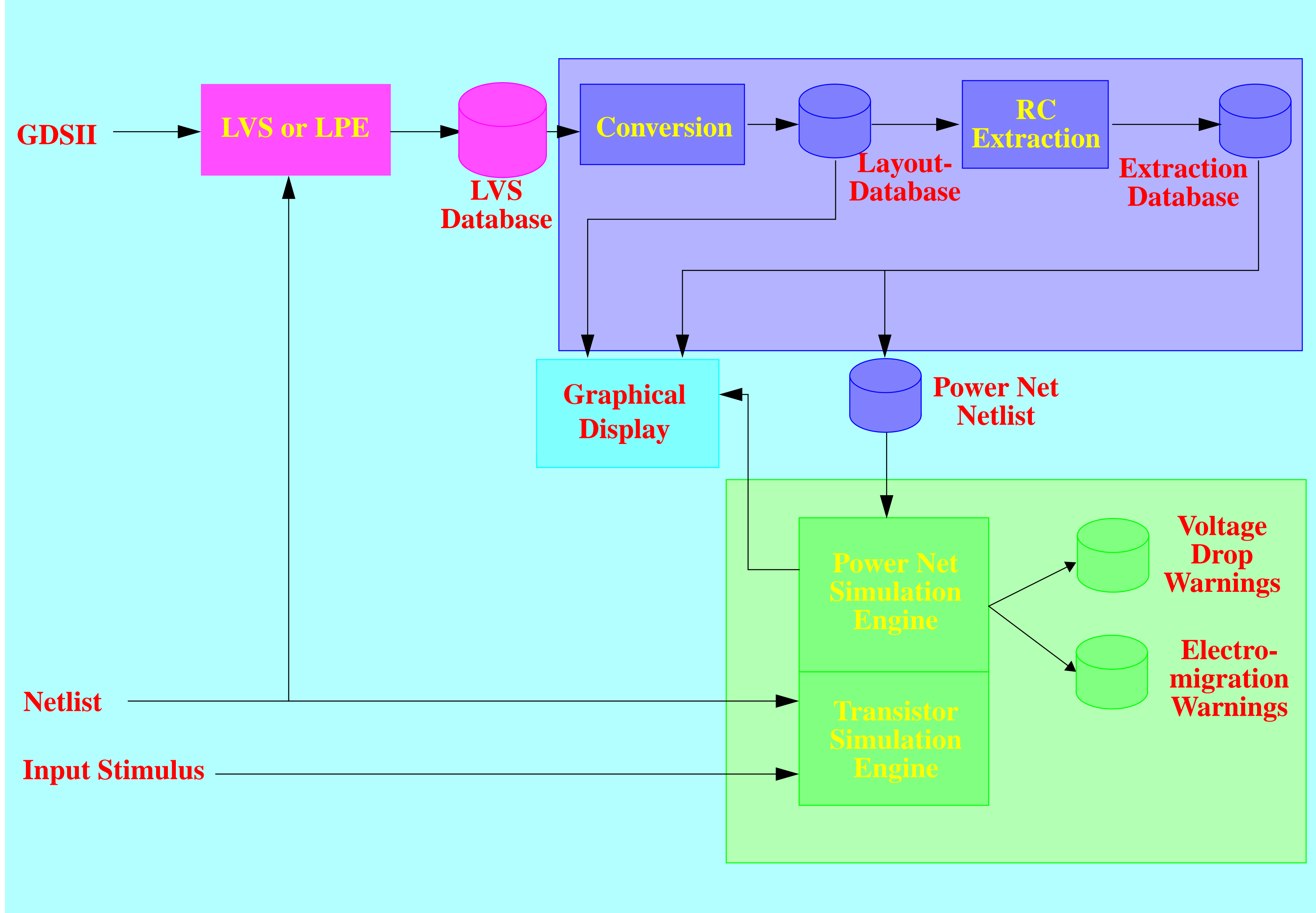
- **Transistor Simulation Engine**: simulate transistor circuit and obtain accurate current information

- **Extractor**: accurate extraction of power network

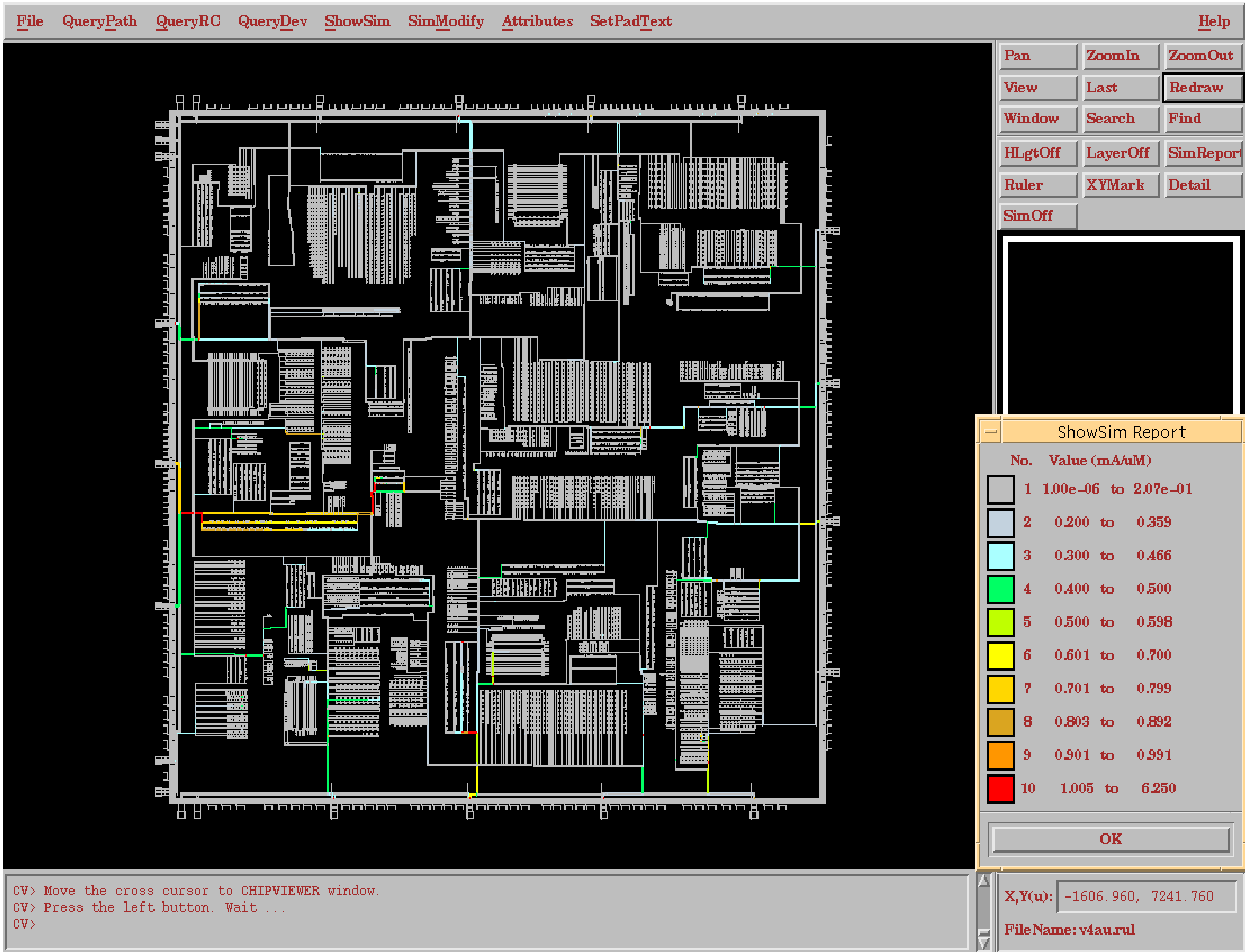
- **Power Net Simulation Engine**: simulate the power network and to perform reliability analysis

- **Display**: graphical display of extracted data and simulated results

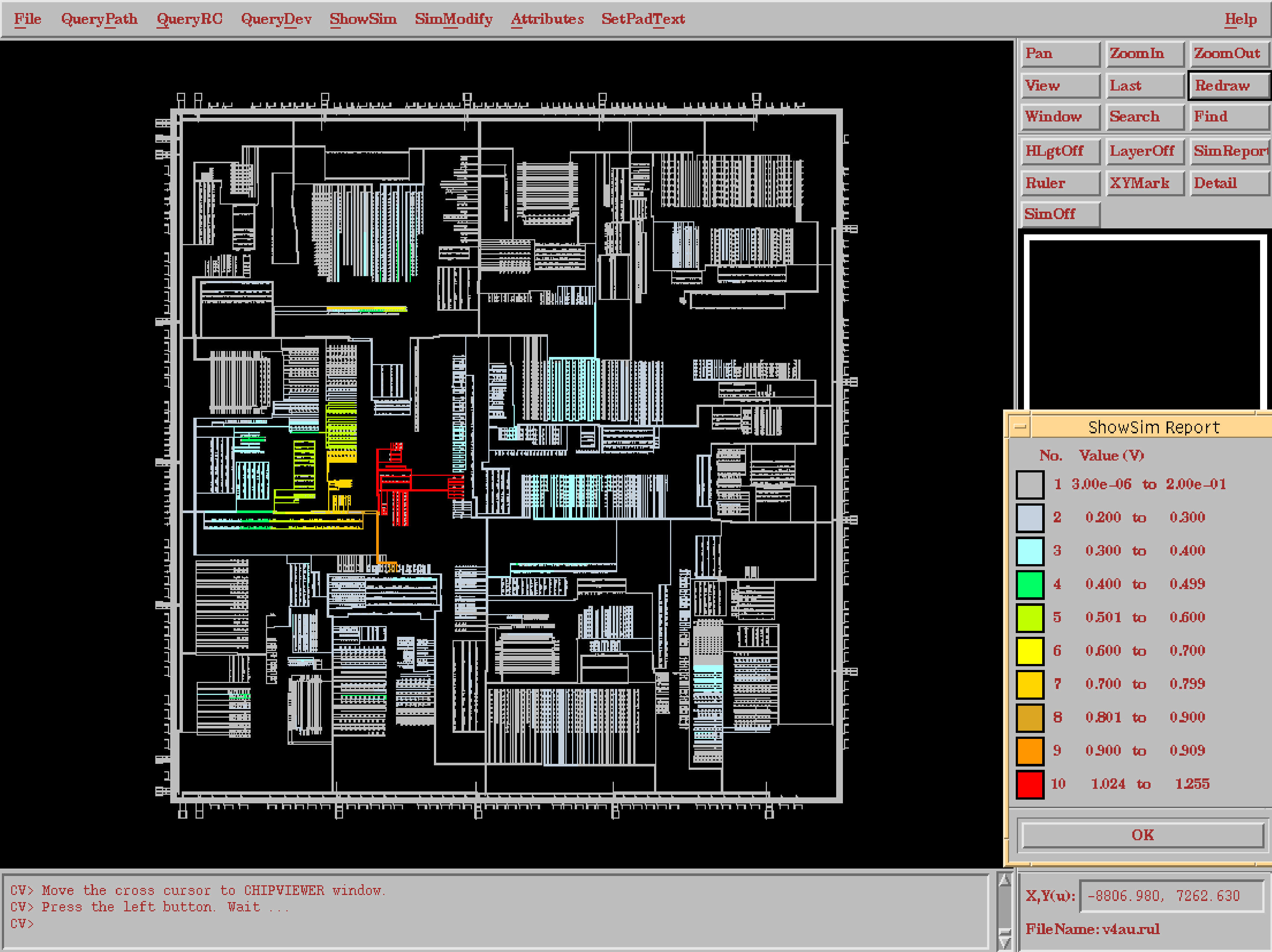
Reliability Analysis - Simulation Data Flow



• Reliability Analysis - VDD Current Density Display



• Reliability Analysis - VDD Voltage Drop Display



• Reliability Analysis- VDD Power Distribution