

# Summary of RTL-to-GDSII tool flow using Synopsys DC and Cadence SoC Encounter

Arun C.

December 16, 2010

## 1 Overview

The reader is assumed to be familiar with the IC design methodology and steps. If not please go through the same provided in the wiki.

In order to use this flow, the following tools must be installed:

- Synopsys design compiler (2006-06).
- CADENCE SoC encounter (8.1).
- Mentor-Graphics Modelsim RTL simulator (SE-64 6.2). ModelSim may be replaced by any simulator which produces a VCD file.

A standard-cell library is also needed in order to run the flow. The current flow uses the OSU standard-cell library (for the 180nm TSMC process).

This flow will work correctly only for designs with a single clock.

The toplevel directory structure is organised as follows.

1. rtl: Contains the rtl code.
2. LIB: All the libraries are copied/linked here.
3. synthesis: Contains the synthesis flow.
4. pnr: Contains the PnR flow.

In each individual directory, a corresponding readMe is available.

The steps in the back-end design flow from RTL to GDSII are as follows:

- Synthesis: RTL ( behavioural code ) is converted to structural netlist. Inputs for this stage are rtl, design constraints and synthesis libraries.
- Placement and Routing (PnR): The netlist is the logical description of the circuit. Physical implementation of the logical netlist is carried out in PnR stage. Following steps are done as part of this stage.
  1. Floorplan: Overall chip dimensions are designed in this stage. IOPins/IOCells/PADS are placed, memories and other macros if available are placed.
  2. Powerplan: Designs how power is provided to the chip. Metal stripes/meshes are drawn on the metal layers, which supply power to the bottom base-layer devices through power VIAs. The power-plan should be done based on maximum IR drop ( voltage supply drop that can be tolerated), Electromigration limits of the metal, Routability of the signal nets ( Power stripes block the signal routes ) and to an extent target power consumption of the chip.
  3. Placement: In this stage the design is placed, and the stage performs optimizations based on timing, transition and capacitances ( typical DRV, design rule viol, parameters ), power and signal routability. In this stage, the clock insertion delay is assumed to be ideal. It is based on the values "we" input to the tool or 0, not on actual "achieved" values. Trial routing is done after every stage to ensure the routability of the design and to get a more accurate idea of parasitics. Similarly timing, area, routability and design-rule-violation analysis (DRV) are analysed in each stage.
  4. Clock-tree synthesis (CTS): Clock insertion delay is balanced across all the sequential elements in the design. An ideal clock tree has 0 insertion delay and 0 skew. But the implemented one will have differences in the clock latency and skew among the flops. Optimizations and analyses are done similar to the placement stage based on these clock non ideal timing and ensure that all these are within limits. Other signals similar to clock such as synchronous RESET, SCANMODE etc also need to be synthesised in similar way, if required.
  5. Routing: The design is actually routed in this stage. All those signals which cannot be routed will be reported as geometry/connectivity

violations. If the routing does not conform to antenna rules, these also are reported as violations. One more round of optimizations and analysis similar to placement and CTS is done with the actual extracted route parasitics. ( RC values ).

6. Filler insertion and final database generation: During the VLSI manufacturing process, it is important that the surface of the wafer is as planar as possible after each processing step. Hence the gaps in between the standard cells are filled with "filler cells" that do not have any functionality. These cells just extend the bulk region of the silicon and improve the silicon planarity.

The design is RC extracted ( the format used is DSPF). Design is saved in GDSII form ( contains metal level mask information, only polygons and texts ), DEF ( contains the physical data, such as the co-ordinates of a particular signal connection (layer,thickness)) and the verilog/vhdl netlist corresponding to the final layout. Final reports are generated and analysed. Also the design is checked for geometry/connectivity/antenna violations which may occur depending on the way the design is routed. If present these are corrected and fixed again.

- Simulation and VCD Generation: The user must write a Testbench which provides a set of inputs for the rtl and monitors the output. The final netlist from the PnR stage is used and the node activity is written out in the VCD (value change dump) format. This step is called gate-level-simulation (GLS)
- Power calculation based on VCD (done using Encounter): The VCD-file generated in the simulation step along with the final Encounter database is used to generate the final power numbers. Inputs to this stage are the VCD file and the Encounter database (contains the netlist, placement, routing, parasitics information etc.). If the parasitics information is not present in the database, we need to re-extract the database.