

Physical implementation: challenges and solutions

By Olivier Coudert

Senior Technologist

Monterey Design Systems Inc.

E-mail: coudert@mondes.com

Designing a high performance VLSI chip using process technologies at 0.18 μm or below is a tremendous challenge. While market windows continue to shrink, design complexity is rapidly increasing due to the combination of finer geometries, design size, higher clock frequency and lower voltage. As a result, interconnect and signal integrity have emerged as dominant factors to be managed in order to successfully complete designs.

This renders obsolete most of the conventional design flows that have not been designed to address all of these new effects. Simply making incremental improvements to existing tools and methods will not address the full extent of the issues. New technologies and flows are required that have been designed from the ground up for challenges associated with design done at 0.18 μm and below.

This article discusses several critical flow-related issues including the way in which the logical and physical variables are interleaved, and must be considered together during the physical design implementation. It then examines two key issues of logical and physical design. The first is the dependency of the logic and timing variables with respect to the physical aspects, for example, placement and routing. The second issue is the flow itself and its convergence. For instance, what are the tasks that need to be performed, at which moment, and in what sequence?

Challenges in physical design

Accelerating advances in process technology have resulted in a number of key challenges facing designers of multi-million gate chips today: timing closure, signal integrity, design variable interdependence, clock and power routing, design sign-off and design size.

Timing closure continues to be an illusive goal. Today, tim-

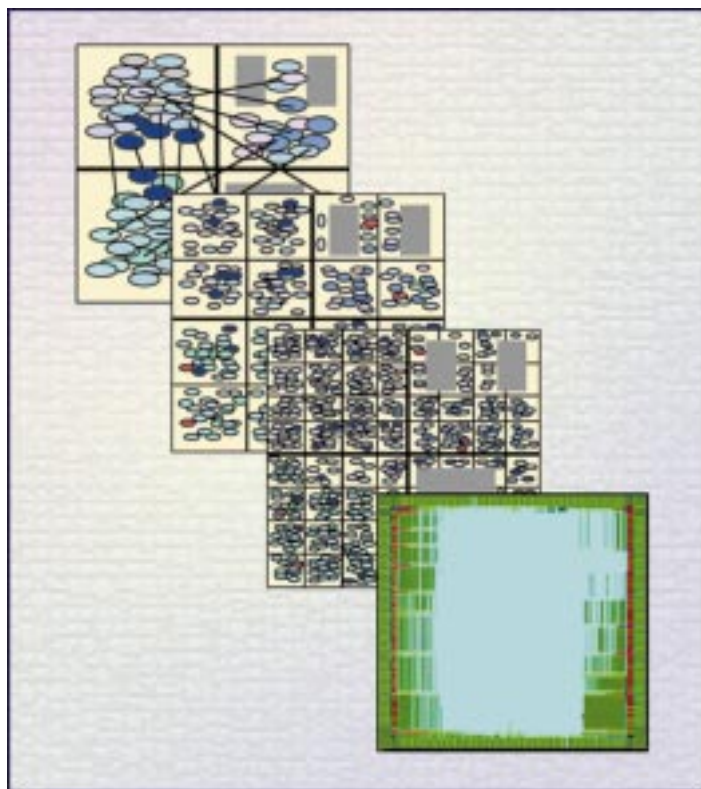


Figure 1: Designers need a flow that makes physical information available throughout the entire design process.

ing closure is very complex and can only be assured by accurately modeling and evaluating the propagation delays of the library cells, the physical placement of the cells and the electrical characteristics of the interconnect.

Signal integrity has emerged as a determining factor to not only the timing, but to the functional integrity of the chip as well. As the ratio of cross-coupling capacitance to inter-layer capacitance increases, the rate of timing and functional violations due to signal integrity problems also rises. Other physical aspects such as antenna effects, electromigration, self-heating, and IR drop must also be analyzed and controlled throughout the chip construction process.

While there are many design variable interdependencies to be dealt with, perhaps the most important is the trade-off between routability, timing, and power consumption. Optimizing for any one may cause problems with the others. An open objective function that can consider multiple objectives simultaneously is required to address these complex interde-

pendencies.

Clock and power networks consume massive amounts of routing resources. Their construction and analysis must be started early and they must be tailored for an individual chip's requirements. Clock trees are frequently inserted after detailed placement is complete. Power networks are frequently pre-determined based on statistical or empirical estimates. However, these methods are showing increasing rates of non-convergence and are no longer adequate.

Design sign-off lost its meaning in the mid-1990s. On a multi-million gate chip, it is not possible to sign-off on a netlist after completing logic design with a reasonable expectation that the physical implementation will meet all design requirements. Physical design is now as important as logic design in determining whether the requirements are achievable.

Design size has exceeded the limitations of gate-level design tools. It is not possible to design and implement a multi-million gate chip as an indivisible whole. The chip must be planned at a high level, parti-

tioned into smaller pieces, and then completed using lower level tools.

Different approaches

A number of different approaches have been proposed for meeting the challenges described above. These include: using custom wireload models in lieu of statistical wireload models, designing smaller blocks, then assembling them, constant delay synthesis and placement aware synthesis.

Custom wireload models are generated after place-and-route and are fed back to synthesis to fix timing violations. However, re-synthesis results in a different netlist and may cause new timing problems resulting in a non-convergent loop.

Designing smaller blocks containing several hundred thousand gates each and then assembling them at the chip level requires top-down mapping of design constraints to the individual blocks, then bottom-up refinement as the definition of the blocks solidifies. Done manually, this process is very time-consuming and may result in unacceptable increases in die size.

Constant delay synthesis is attractive because of its elegance and simplicity. A fixed delay (i.e., fixed gain) is assigned to every logical stage so that timing constraints are met. However, constant delay synthesis assumes continuous (non-discrete) sizing. Mapping the result onto a discrete library will result in sub-optimal results.

Placement aware synthesis is a trend that is gaining acceptance. However, combining logic synthesis together with placement does not help unless accurate routing information is included. Timing constraints may result in an overly congested, un-routable placement. This approach is a step in the right direction, but does not go far enough in the integration of the logical and physical domains.

Complete design flow

Physical design must start early—at the same time as logic design. Initial estimates of

physical parameters will be very coarse, but will provide a useful baseline that can be used to measure progress towards the required goals. As more accurate information becomes available about the physical characteristics of the chip, the new information is incrementally updated and evaluated to track progress towards design convergence.

Physical design must incorporate hierarchy in order to address the issues of design size, design reuse, turnaround time and ultimately convergence. Automated operations are needed to view, manage, and analyze physical hierarchy, and to manage the complex interrelationships between the hierarchical subsystem blocks.

Physical design must provide feedback and metrics during the entire flow. At any point in the flow, it is necessary to be able to determine where the design is relative to all requirements, and then to use that information to drive the downstream operations as the design progresses. The process must be able to quickly converge to the point where the correlation between the current estimate and the final implementation can be bounded sufficiently to sign-off on the design prior to final implementation.

Physical design must be a convergent process where the chip is optimized for all design requirements simultaneously and is continually refined until the final implementation is complete. It cannot remain a sequential and iterative process of implementation, verification and repair that may never converge.

The physical design process starts with chip-level hierarchical design planning. Chip-level planning may start very early, before the sub-system blocks have any structural, physical, or timing definition. Chip-level requirements are allocated and

budgeted to sub-system level blocks and are used to drive the implementation of those blocks. As the blocks gain structural, physical, and timing definition, new models for the blocks are abstracted and incrementally updated. At the chip-level, timing and area slack are automatically re-allocated to accommodate the new models. In this way, the chip-level plan

design requirements. This “check-and-repair” approach worked well for process technologies of 0.25 μm and larger. However, at 0.25 μm and below, the sequential approach is very unwieldy and may take many iterations to converge.

For deep submicron designs, simultaneous optimization of all design requirements provides a cleaner and more

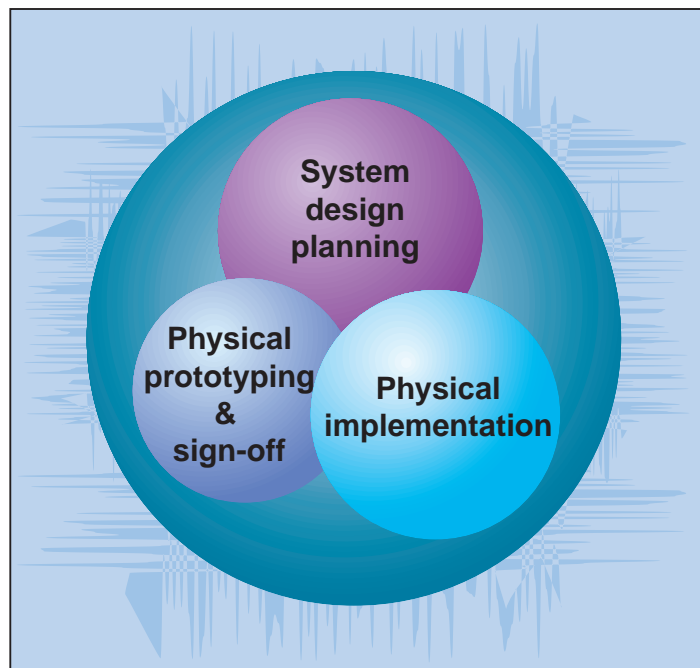


Figure 2: Continuous refinement is the process of starting with a very coarse model and continuously refining it until the final result is achieved.

starts with a very coarse model and is continuously refined until the final implementation is achieved.

Block implementation is composed of cell placement, logic optimization, clock tree synthesis, power routing and signal routing. Historically, these operations were performed sequentially and at the end of the process, the implementation would be checked to make sure that all design requirements were met. If problems were discovered, the block would be sent back to an earlier step and the process would be repeated as many times as needed in order to achieve all

predictable design flow. Cell placement, logic optimization and routing must be done concurrently with full awareness of the power network and clock tree requirements.

Simultaneous optimization also makes possible the construction of a “physical prototype.” As the implementation is continuously refined, it arrives at a stage where the physical parameters can be bounded. The prototype is assured to correlate with the final implementation within a specified range. At this point, the design may be signed-off, well in advance of the completion of the final implementation.

In order to achieve simultaneous optimization, cell placement must use an open objective function that can incorporate multiple terms for routability, timing, wire length and crosstalk. Logic optimization must be available at any point during the process to perform cell sizing, buffering, resynthesis, and technology mapping. Power and clock networks must be fully visible at all times. Routing must be performed in conjunction with real-time signal integrity tools to avoid problems with crosstalk, IR drop, electromigration, and inductance. Only by performing all of these operations concurrently, and with full knowledge of the power and clock networks, can convergence be assured.

With every process generation, there are more transistors, higher clock frequencies and additional physical effects to consider. This article discussed the challenges that need to be addressed during physical implementation. A flow based on planning, placement, and routing refinement (including clock tree, power routing, and scan chain) with an open cost function, together with logic synthesis and optimization, meets these challenges. It enables early estimation and optimization of congestion, timing and physical effects at a point in the flow when the design has enough flexibility to accommodate perturbations produced by the optimization procedures.

The physical prototype concept has been introduced to restore the timing sign-off lost in the mid-1990s. The interconnect-centric deep submicron era raises new issues, where placement, routing and logic optimization are tightly interdependent. Further, with the introduction of hierarchy to handle multi-million gate designs the need for a new full-chip design system emerges.