Design planning for large SoC implementation at 40nm: Guaranteeing predictable schedule and first-pass silicon success

bhupesh dasila - May 07, 2013

The physical design implementation of large complex deep sub-micron technologies has evolved to a stage where it is essential to consider every aspect of SoC design and implementation during the planning process. The era of a waterfall flow from RTL to GDSII was over long ago. Even the efforts to bridge the gap between the front-end and back-end design process, through tools and flows, are not always sufficient.

Modern SoC development requires a holistic approach and thorough planning starting at the design architecture of the SoC. The ASIC implementation process has to keep pace with the design complexity, performance, and time-to-market, all while ensuring first-time silicon success. These challenges, which are compounded by the advancement in the package technology and the complex foundry requirements at 40nm and lower nodes, make it essential to evaluate the design requirements and the technology limitations early on in the SoC design cycle.

Every team involved in SoC development must work cohesively for optimal planning and strategy. Large, complex designs with high performance requirements benefit from comprehensive planning up-front. These benefits include both predictability and reliability of results.

The most important aspect of planning and strategy is dividing the execution into different phases. These phases include technology, design flow, and SoC evaluation, exploring and identifying physical design challenges early, and finally physical design implementation. The physical design implementation team should thoroughly identify and define the entry and exit criteria at every phase. The learning from one phase should become the foundation for the next phase of the execution.

The definition of various phases and the criteria for entry and exit should align with the RTL design and verification process and their milestones. Hence communication between various groups involved in SoC development is crucial. The key is to thoroughly analyze and explore the design before the final implementation is executed on the fully-verified design to achieve predictable results in quality and schedule.

**Technology, Design Flow, and SoC Evaluation**

When the RTL design and verification is in the architecture stage, the physical design team should be engaging itself in evaluating the technology. This includes a thorough exploration of the process technology to learn its capabilities and limitations. Evaluating the technology libraries, determining
the implementation tools and flows, and capturing the SoC requirements are a few of the fundamental and vital steps, which can be streamlined while RTL is being developed.

**Technology Evaluation**

Library selection and its evaluation are very important for estimating the overall design performance. The designer must know the priority of optimization, such as speed, area or power. This priority will assist in determine what technology best suits the design needs. It also will assist in IP selection.

Different library vendors have a variety of offerings that include a mixture of threshold voltages, multiple channel lengths, and numerous metal stacks. It's important to evaluate and compare all of the offerings that apply to the SoC with respect to the target characteristics. This technology evaluation data can be used by system and RTL designer to write physically aware RTL and start early planning of physical limitations to the design. This also provides a baseline for the physical design team to create estimates of SoC area and power or evaluate the requirements of the SoC. All of these decisions must, of course, align with the business goals.

One of the first library characteristics to evaluate is the delay per stage of the standard cells. This data should be collected for all of the different types of standard cells in each of the threshold voltages that are available. Comparing the advantages of speed of the lower threshold voltage cells versus the higher threshold can assist in determining the mixture Vt cells that should be used in the design in order to obtain the lower power possible.

This data will also assist RTL designers to determine where to insert registers in their design. For very high performance designs that push the technology limits. The number of stages of logic between registers will be maximized. Knowing the number of stages that can be placed between registers will reduce timing closure iterations that require a loop from physical implementation back to RTL.

Figure 1 shows an example of the data collected to determine the delay for one stage. This example results in an average delay of 29ps per stage. The slop of the drawn line gives the corresponding stage delay.
Figure 1: This graph shows an example of the data collected to determine the delay for one stage.

Technology Evaluation (cont.)

Another important technology experiment is to analyze the delay per mm at different metal layers. This data will assist top level floorplanning when determining the maximum distance floorplan blocks can be placed before rebuffering needs to be inserted.

By dividing the clock cycle time by the calculated ps/mm value, a limit can be set to the maximum distanced two floorplan blocks can be placed. This will reduce late top level timing issues due to distance. If for some reason this distance limit cannot be met, it may be necessary to reoute certain nets on the top wide metal layers. In this case, routing guides can be created early to also reduce late timing violations.

If reducing latency in the SoC is a requirement, then placing floorplan blocks within a clock cycle is critical. Knowing this data will also assist in the development of the top level timing budget and eliminate late top level floorplan changes due to timing closure issues.

Figure 2 shows an example of a buffered net using different drive strength cells. The minimum delay achieved in this example using a drive strength of 8 is 195ps/mm.
Figure 2: This figure shows an example of a buffered net using different drive strength cells.

A final example of technology evaluation experiments that should be performed is comparing the area, timing, and power of compiled SRAMs to a synthesized register equivalent. In most technologies there is a crossover point where the area of a synthesized register SRAM configuration is better than the area of a compiled SRAM. Documenting a graph that shows this crossover quickly allows the RTL designers to determine if the best design choice is an instantiated compiled SRAM or synthesized logic. Using the raw area data, the physical implementation team can get a more accurate area estimation early in the RTL development stage for floorplanning purposes.

Figure 3 shows an example of this graph for a 1-port, 32-bit-wide SRAM.
Figure 3: Using the raw area data, the physical implementation team can get a more accurate area estimation early in the RTL development stage for floorplanning purposes. This shows an example of this graph for a 1-port, 32-bit-wide SRAM.

**RTL Design Rules**

Using the data collected in the technology evaluation experiments will not only assist with the selection of the optimal technology for the SoC, it also starts the documentation of RTL design rules. These design rules will assist the RTL designers to write code that is targeted to the physical capabilities of the selected technology. Early physical information translated into design rules will reduce late surprises in physical implementation due to timing issues, area and power estimations, and dft requirements.

Along with the results of the three experiments already discussed, technology RTL design rules should include the following:

- Standard cell drive strength capabilities
- Standard cell power characteristics
- Standard cell dont_use list for synthesis and physical design
- Floorplan block IO registering requirements – ideally register all IO
- Clock gating requirements
- SoC frequency and power targets
- Floorplan block area targets

Most designs require some up-front preparation to achieve high coverage from DFT techniques. One
of the simplest and biggest implications that can be applied to the RTL design rules is that the test signal 'Scan_Mode' is only present in gate-level netlists. This is a step that will save a tremendous amount of work in the back end when DFT is being inserted. Some of the basic points to be taken care of while coding the RTL are:

- In the case of internally-generated clocks, the roots of the new clocks should have a new MUX added to them, and the original clock should be fed through it. The MUX's select pin should be controlled by 'Scan_Mode'.
- Clock gating cells should include and disable Scan_Mode
- During scan, no clock will feed into the data input of an FF – Clocks cannot influence the data content of any FF during scan.
- All sets and resets should be controlled by a primary I/O pin.
- Latches should be driven into a transparent state during Scan_Mode.
- All tri-state bus drivers need to be driven from a fully decoded set of FFs. All tri-state buffers feeding into a single node need to have one, and only one, output enabled during the scan shift and capture phases.
- No combinational logic loops in the design -- If there are any combinational logic loops in the design, they should be broken by a Scan_Mode signal.

**Design Flow**

**Design Flow**

Since every SoC has its custom requirements it is important for designers to streamline the design tools and flows in the early development phase to meet the SoC requirements and align with the logistics, business targets and schedule. The large and complex designs at technology nodes 40nm and below require designers to consider every aspect of the design exchange and to place adequate hooks and checks at every stage.

For example, pushing the physical information from the synthesis stage down to physical design is a must for high-performance design. Figure 4 shows some of the hooks and checks that should be used.
Figure 4: Pushing the physical information from the synthesis stage down to physical design is a must for high-performance design. This design flow diagram shows some of the hooks and checks that should be used.

DFT is another parallel implementation, which has to be merged smoothly in the mainstream implementation flow. DFT can be implemented in the RTL or in the netlist, and this aspect has to be assessed and streamlined at the initial planning phase.
Tools from different tool vendors may have different capabilities, which may push designers to employ tools from various vendors. For example, one vendor’s synthesis tool may compile faster with better results while another vendor’s place and route tool may have an exceptional router. Although, logistics may also push designers to a specific set of tools.

Hence, in the early RTL development phase itself, designers should exercise taking some part of the design, like a complex block, through the entire implementation process. This will help in streamlining the custom flow, generating the various kinds of statistics needed for early design performance analysis, and it will help in establishing the miscorrelation at various stages, moving down from RTL to GDSII. For example, miscorrelation in timing from physical synthesis to post layout STA may require much analysis to adjust tool setting and margin to eliminate the miscorrelation.

It is a common practice to introduce a multi-phase design implementation flow. However, the key is in the definition of every phase and the entry-exit criteria for every phase. Another key point is to align the implementation phases with the RTL design and verification development.

The graph below explains the miscorrelation at every stage moving from physical synthesis to signoff.

**Stagewise correlation**

![Stage correlation graph](image)

**Soc Evaluation**

Using the technology characterization data and taking a block through the design flow will help assess the effort required to meet the Soc requirements. For area sensitive designs, using early area analysis will help develop an accurate floorplan. For very fast designs, reducing tool correlation and applying the technology evaluation data at the RTL stage will assist in quicker timing closure and ease the flow through the physical design tools.

**Summary**

Large SoCs in a smaller geometry technology increase the design complexity multifold. The traditional waterfall approach of SoC implementation can no longer guarantee a predictable schedule and reliable silicon. Upfront and thorough analysis, in every aspect of SoC development, is needed for today's SoC designs. This analysis can then be used in detailed floorplanning, die size estimation, and finally implementation of the physical design.

In Part 2: [Die size and power estimations](#).

**Acknowledgements**
With special thanks to Kris Kuula for her substantial contributions to this article.

More about Bhupesh Dasila