TUTORIAL 4

GAIN-BASED LOGIC SYNTHESIS

Speakers:

Prabhakar Kadva - IBM T.J. Watson Research Ctr., Yorktown Heights, NY
David Kung - IBM T.J. Watson Research Ctr., Yorktown Heights, NY
Ruchir Puri - IBM T.J. Watson Research Ctr., Yorktown Heights, NY
Leon Stok - IBM T.J. Watson Research Ctr., Yorktown Heights, NY

Background: There exists a large gap between full-custom design and standard ASIC design. Gigahertz micro-processors have been announced while most ASIC parts run at maximum speeds of around 200 Mhz. However, a significant part of this gap can be closed by using the appropriate libraries and synthesis techniques. The same techniques that allow us to synthesize all control logic for gigahertz micro-processors, also helps the time consuming design closure of large complex ASICs. This is especially the case, when decisions early in the process are mistargeted due to the use of misleading wireload models. Delay models parameterized by gain allow predictable pre-placement synthesis without wire-load models and a more rapid evaluation of the effect of changes on the timing, thereby speeding up the combined synthesis and placement process.

Description: Delay models parameterized by gain significantly enhance the design and timing closure problems we are seeing in today’s complex standard cell design methodologies. To use these techniques most effectively the libraries, algorithms and methodology need to be adapted. We will address all three of these and tie them together in a coherent methodology that enhances the predictability of the timing closure process.

We will revisit the circuit basics of standard cells and the creation of static delay models with emphasis on the construction of gain-based delay models. Guidelines for designing practical libraries and their implications will be discussed as well as efficient algorithms for timing analysis, and area and load calculation.

Gain-based delay models open the opportunities to the construction of a different class of synthesis algorithms. Technology mapping in particular can benefit from the load-independence property of these models. Recent algorithms from the literature will be discussed. Buffering and fanout tree construction are other examples of algorithms that can be made much more predictable. Wire and gate sizing will be revisited and shown how they can be done in this environment.

Gain-based synthesis opens the opportunity to synthesize predictably without wire-load models. The actual sizing can be postponed until deep in the physical design phases, resulting in a more reliable timing information and better identification of timing critical regions. A methodology will be described that ties these together.

This tutorial is intended for designers to get an insight in circuits and libraries for which very efficient algorithms exist to synthesize them, for CAD-tool developers to understand state-of-the-art algorithms for rapid design closure on large designs and for their managers to get an insight in the applicability of gain-based synthesis techniques to their specific design problems.