XAPP140 (v1.0) February 26, 2001

Physical Synthesis

Author: Hamid Agah

Why is Physical Synthesis Necessary?

In the domain of deep submicron (DSM) and nanometer ASIC technologies (180 nm and below), the traditional separation between logical (synthesis) and physical (place and route) design methods often causes a problem—designs cannot meet their realistic timing objectives; creating the well known “timing closure problem.” Timing closure is now considered the biggest area of difficulty for ASIC performance-oriented designs. The underlying reason is that circuit delays are dominated by net delays, which are influenced by the placement of the cells. The traditional fanout-based wireload models, for estimating interconnect delay during synthesis, are considered inaccurate and are the key factor causing the lack of timing predictability between post synthesis and post layout results. It is evident that synthesis and placement technologies must merge to create properly placed and routable designs that meet realistic performance goals.

What is Physical Synthesis?

Physical Synthesis refers to the ability of creating a properly placed-and-routed circuit from the RTL code; a circuit that meets the realistic performance goals of the design in one pass. In cases where there are very aggressive performance goals or tight physical constraints, a second pass may be needed to achieve the desired performance goals. A properly placed-and-routed design meets the design rules of the target silicon technology and is routable by a detailed router. The silicon technology determines the realistic performance goals of the design.

What Does It Do in an ASIC Environment?

Physical Synthesis tools, such as Physical Compiler from Synopsys or PKS from Cadence (Physically Knowledgeable Synthesis) replaces the typical Synthesis tools (such as Design Compiler from Synopsys) that many ASIC designers are using today. Users typically start with a design-planning tool such as the Synopsys Chip Architect or the Cadence LDP and decide on:

- Physical area allocated to each synthesizable module in the design
- Physical location of each synthesizable module
- Physical locations of RAM, ROM, hard IP, and other non-synthesizable blocks in the design
- Pad (I/O) locations

The next step is to perform the top-level routing and the timing analysis, based on the chip-level timing constraints. Based on the analysis result, users adjust the physical port location on synthesizable modules, and adjust the location and orientation of non-synthesizable blocks to derive a realistic timing budget for each synthesizable module in the design.

At this point, all the necessary information is available for every synthesizable module in the design so the user can proceed with the physical synthesis step. For each synthesizable module, the Physical Compiler takes in:

- RTL code
- Timing constraints that are derived from the design planning step
- Physical constraints (area, port locations, etc.) that are derived from the design planning step
- Synthesis and physical libraries

The Physical Compiler produces a netlist and physical information (such as the proper placement of all the cells) that meets the timing goal of the particular module. The next step is to perform the detailed routing on the circuit and timing analysis to insure that the fully placed-and-routed module meets its performance goal. Cadence and Avant! are the only EDA vendors that offer proven detailed routers that are trusted by major ASIC vendors (such as NEC, LSI, TI, etc.)

Physical synthesis requires thousands of strategies to be evaluated by the software while the circuit is being properly placed-and-routed and timed. Detailed routing takes a very long time and it is not suitable during the synthesis process. Therefore, following placement, the Physical Compiler performs an estimated routing to assess the net delay and the impact on the module's timing objectives. It must also decide on the next synthesis strategy if the timing goal is not being met. The single most important step is the correlation between routability analysis (obtained from the congestion map) of the Physical Synthesis tool and the final detailed router. Without such a correlation, the placement that is created by the Physical Synthesis tool may be unusable by the detailed router, nullifying the result of the Physical Synthesis tool.

What Are the Pitfalls?

Cost: Synopsys currently charges $200K for a single license of its Physical Compiler. A design planner or floorplanner (such as the Synopsys Chip Architect) is needed to create the necessary information for the Physical Compiler and costs about $150K. The total cost of setting up the Physical Synthesis environment is about $350K.

Interoperability: The success of a physical synthesis tool is highly dependent on the routability of the placed circuits that it produces. If the Physical Synthesis vendor also provides a proven detailed router, then there is high degree of certainty that placed circuit can be routed by the vendor's detailed router. Synopsys' Physical Compiler is the most successful physical compiler so far, but it relies on Cadence and Avant! detailed routers to complete the physical implementation of the design. Synopsys does not yet provide a proven detailed router. If Cadence and Avant! change the algorithms of their routers they may not work properly with the Synopsys tools.

Layout Expertise: The users require extensive training to become comfortable with physical design concepts and this uses additional resource at additional cost.

How Does Xilinx Address the Timing Closure Problem for Virtex and Spartan II FPGAs?

Xilinx addresses the timing closure issue for Virtex™ and Spartan™-II architectures by using a three-step process of "Predict, Control and Improve" to implement designs that can meet their realistic timing objectives with a minimum number of iterations.

Predict: In the Xilinx FPGA architectures, the interconnect timing is less variable than an ASIC. This characteristic makes it possible to create interconnect models that are not based on fanout. The models can be used during the synthesis process to estimate the interconnect timing with a high degree of predictability with respect to the placed and routed design. Xilinx has partnered with leading FPGA EDA vendors to offer synthesis tools that are aware of the Xilinx FPGA architectural details. They use our accurate interconnect modeling to produce netlists with timing within 20% of placed-and-routed designs. The next two steps close the remaining performance gap.

Control: Guiding the Xilinx timing driven implementation tools with realistic timing constraints, high quality netlists, and accurate physical constraint are the keys to closing the performance gap that may exist between the synthesized netlist and the placed and routed design. Xilinx worked closely with Synplicity to develop Amplify, a design planner and physical optimization tool for FPGAs. Amplify can improve the netlist quality through physical optimization techniques such as moving registers across physical boundaries to increase performance. It can also provide accurate area constraints and critical path physical grouping to the Xilinx implementation tools. It is still possible to have failing paths remaining after the place- and-route. The next step addresses the remaining failing paths.

Improve: Xilinx, in partnership with leading FPGA EDA vendors, has developed a tight interface between the synthesis tools and the Xilinx implementation tools that allows re-

optimization of failing paths and ECOing new circuits into the Xilinx implementation tools. In a majority of cases this capability can enable timing closure in two passes.

Conclusion

Physical Synthesis is rapidly becoming a requirement to close the timing gap for DSM (0.18µ and below) ASIC designs. The key reason is inaccurate timing estimation during synthesis due to unpredictable interconnect timing. Xilinx has successfully applied the three-step process of "Predict, Control and Improve" to close the timing for Virtex and Spartan-II architectures (0.22µ to 0.18µ). The key success factors are accurate timing estimation during synthesis, true timing driven place-and-route, and re-optimization of failing paths. Process technology continues to shrink for upcoming Xilinx architectures enabling designers to implement more complex and higher performance designs. Xilinx will continue to provide the right solution to the timing closure issue for future generations of FPGA architectures.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
02/26/01	1.0	Initial Xilinx release.