FPGA SYNTHESIS

Where We've Been, Where We're Going

Exemplar Logic and Xilinx pioneered FPGA synthesis and have been working together for over a decade to advance and improve methodologies.

by Tom Hill, Silicon Vendor Relations Manager, Exemplar Logic, tom.hill@exemplar.com

ASIC synthesis experienced rapid growth in the EDA industry during the early to mid-'90s. However, it was the programmable logic industry, that pioneered chip design using logic synthesis. Pre-dating the introduction of the first commercial ASIC synthesis tools by ten years, PALASM and ABEL were being used to synthesize PAL devices. By the mid-'80s most board designs included a programmable logic device.

Our History

Xilinx, founded in 1984, had a novel idea for a new ASIC device called a field programmable gate array (FPGA). Their architecture differed significantly from the PALs in two ways. First, it employed SRAM technology to achieve reprogrammability, which pushed gate counts far beyond PALs. Second, it offered multiple logic levels on a single device, making it more versatile and more similar to gate arrays than the two-level logic structures of PALs. This second reason spawned a relationship between Exemplar Logic and Xilinx in 1987 that would profoundly change both companies.

In 1985, Ewald Detjens, the founder of Exemplar Logic, was participating in a UC Berkley research project developing algorithms for multiple logic level, interactive synthesis (MIS). Several classmates, wishing to apply these concepts to gate arrays, went on to form the origins of one of today's largest EDA companies. Ewald, a person preferring the road less traveled, was looking for something different when he discovered Xilinx.

Ewald, a person preferring the road less traveled, was looking for something different when he discovered Xilinx.

In 1987 Xilinx introduced its first automated design environment called PPR. Prior to PPR, designers placed and routed their devices by hand using a system called XACT. PPR liberated them from this painstaking process by accepting netlists of cells and automatically performing place and route. For the first time, Xilinx devices could be designed using 3rd party design entry tools and schematic capture was quickly introduced into their design flow. While broadening their appeal to gate array designers currently using schematic capture, Xilinx was failing to effectively reach the market it really wanted: the person already using programmable logic, the PAL designer using PALASM and logic synthesis.

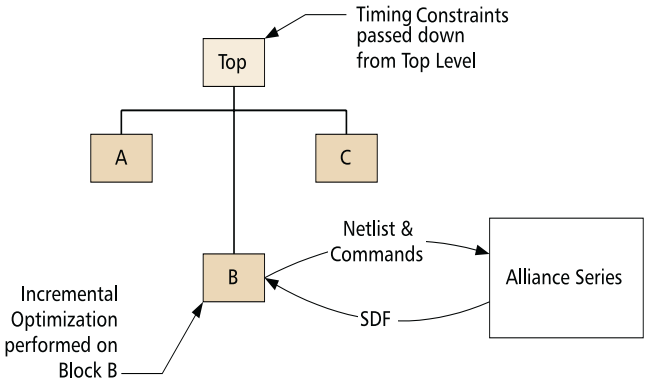
Working closely with some of the original software developers at Xilinx, Exemplar Logic introduced its first two products in 1988. Developed exclusively for Xilinx to target PAL designers, these products were called "PDS2XNF" and "XNFOPT." PDS2XNF converted PALASM files to XNF netlists of simple primitive gates and XNFOPT incorporated the MIS synthesis technology developed at UC Berkley, along with a new invention called LUT mapping, to efficiently optimize PALASM files directly into Xilinx Configurable Logic Blocks (CLBs). Exemplar Logic gave PAL designers easy access to Xilinx technology through logic synthesis and forever changed the face of the programmable logic industry.

From these humble beginnings PLD synthesis has grown to a $50 million a year industry and provides the chief method for designing Xilinx FPGAs. While the tools and devices have grown in sophistication, the design methodology has remained relatively consistent. Today, a single designer can complete most designs using a non-interactive, top-down synthesis approach and "on-board" verification. But will this meet the needs of tomorrow's FPGA designer? The answer is "no." Today's FPGA design methodology needs to evolve to avoid the productivity gap now being experienced by the gate array and standard cell communities where silicon densities have outpaced the designers' ability to utilize them.

The Virtex family represents a major advancement in programmable logic technology providing up to one million system gates of logic.

Our Future

The Virtex family represents a major advancement in programmable logic technology providing up to one million system gates of logic. With this dramatic increase in gate counts, the advantages of preserving design hierarchy are too great to ignore. Initial place and route can be performed early in the design process providing valuable device utilization and performance information while changes, made during final debug, can be limited to a local sub-block, maintaining an efficient design iteration cycle. FPGA synthesis and implementation must become interactive and iterative on a block by block basis to continue to provide an efficient design methodology for FPGAs.

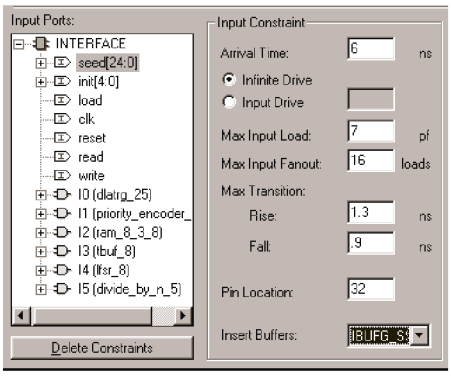


Leonardo Spectrum has been designed, not to hide design hierarchy, but rather to exploit its many advantages. A design hierarchy browser is an integral part of the user interface allowing you to easily access, manipulate, constrain, and swap hierarchical blocks. Incremental design is fully supported at the functional level allowing modules to be corrected at the RTL level, re-synthesized, and re-optimized while preserving all netlist information in the surrounding blocks. Incremental design is also supported at the synthesis level allowing constraints to be "tightened" on sub-blocks and re-optimized to correct timing or area problems discovered during place and route. Block-level design will provide the key to efficient interaction between the synthesis and place and route environments.

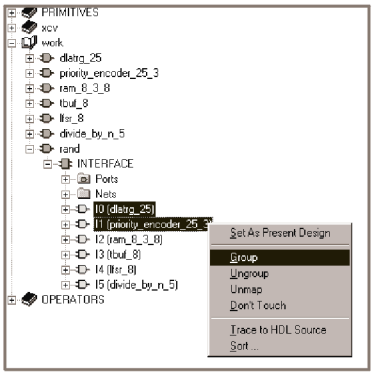
Conclusion

Exemplar Logic and Xilinx pioneered FPGA synthesis and have been working together for over a decade to advance and improve methodologies. Together we are committed to developing new technologies, including improved interactive block-based FPGA design, making programmable logic design the industry's premier design methodology. Σ

Constraint Editor



Hierarchy Browser



Optimization Interface