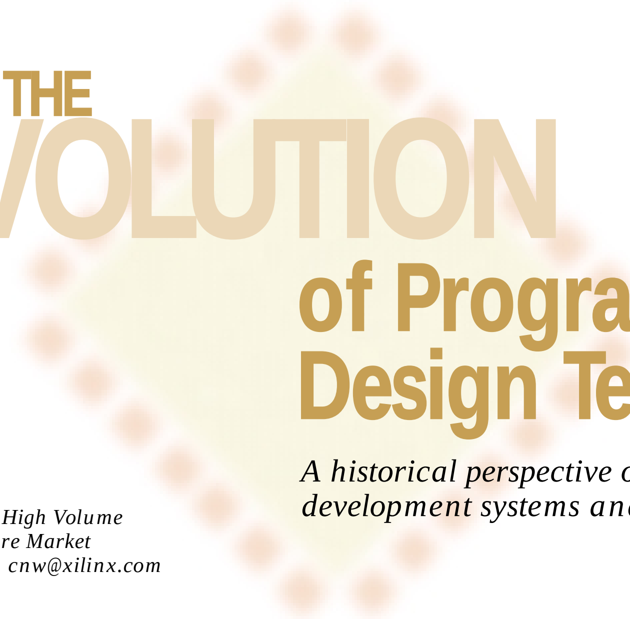


THE EVOLUTION of Programmable Logic Design Technology

A historical perspective on the evolution of Xilinx development systems and design methods.

by Craig Willert, High Volume Solutions, Software Market Manager, Xilinx, cnw@xilinx.com

With the introduction of the first programmable LCA (Logic Cell Array) in 1985, Xilinx changed the course of electronic design. This innovative new technology allowed engineers to design a single chip that performed the equivalent function of a typical printed circuit board. While this creative use of silicon has been heralded as a breakthrough technology that created a new industry, many insiders will tell you that it was the methodical delivery of innovations in the development systems tools that has enabled so many engineers to take advantage of programmable logic.

Development System Timeline:

Year	Key Features	Associated Design Flow
1984	Xilinx Design Editor	Physical Design Editor/ PIP Poking
1986	XNF2LCA1.0, APR1.0, XACTOR 2	Schematic based design, automatic logical to physical design translation, automatic place and route, in-circuit debugging
1988	XNFBA	
1990	XMAKE, Guided Design,	Automatic Design Implementation, Map-then-Merge, XDM
1992	Unified Libraries, XBLOX, Hard Macros	XACT 5 – Batch XACT 6 – Windows (DM/FE)
1994	XACT Performance, RPMs, DM/FE, Floorplanner	
1995	Foundation Series introduction, Software Localization (Japanese)	
1996	MAP, PAR, MPPR, Implementation Engine, EDIF, VHDL, and Verilog standards, LogiBlox, PCI LogiCore, ITA, JTAG, FPGA Express, Core Generator, Constraints Editor, Floorplanner (reintroduced), Foundation UPM	

—1985—

Introduction of the XC2000 Series

When Xilinx introduced its first Logic Cell Array in 1985, the Electronic Design Automation (EDA) industry was focused on delivering schematic capture and gate-level simulation tools to simplify the process of logic design. Although it was considered important to provide an easy to use design solution, the first priority of Xilinx was to enable its customers to gain access to the full complexity of its programmable logic devices. Xilinx first developed a physical design tool – XACT 1.0. This tool, which later became known as XDE (Xilinx Device Editor), gave designers the ability to view, edit, and highlight all of the logic and routing resources within a device. Furthermore, XDE featured advanced "high-level" design capabilities such as Boolean expression, Karnaugh Map, and truth table design (graphically cross-coupled, no less).

...the first priority of Xilinx was to enable its customers to gain access to the full complexity of its programmable logic devices.

The use of XDE ushered in the era of a design methodology that may be characterized as "PIP-poking" (Physical Interconnect Point). XDE's introduction was also responsible for the development of a new breed of designers – "Xilinx experts," who not only were unique in their understanding of the Xilinx design methodology, but also in their detailed understanding of Xilinx programmable logic devices.

Continued on the following page

—1986—

From PIPs to XNF

As the use of programmable logic devices became popular, and it became clear that the Programmable Logic business was viable, Xilinx turned its attention towards improving the LCA design methodology so that designers could take better advantage of the time-to-market benefits that programmable logic offers. To tackle this problem, Xilinx developed a set of design tools that would handle the translation of a design from its logical form (represented as a schematic) to the physical structures that exist within the Logic Cell Array. This spurred the creation of the "PIN2LCA, XNF2LCA, and APR programs.

PIN2LCA

The first qualified "front-to-back" design environment for Xilinx logic devices was created to work with Futurenet, a leading schematic capture tool at the time. Through the first "EDA Alliance," Xilinx and Futurenet were able to create an automatic logical netlist (logic gates and registers) to physical netlist (configured CLBs and IOBs) translation tool (PIN2LCA). By allowing an engineer to focus on designing their required circuitry using schematic capture rather than the Xilinx Design Editor, the use of programmable logic became a viable design methodology for many more designers.

Xilinx Netlist Format and XNF2LCA

In addition to working with Futurenet to develop a proprietary design flow, Xilinx recognized the need to work with other EDA companies. The Xilinx Netlist Format (XNF) was introduced in order to provide a standard for transferring a design database from any schematic package to the Xilinx design environment. Of course, to leverage this netlist standard, Xilinx had to develop the netlist translation program XNF2LCA, completing the transformation of a design from the logical database to the physical (LCA) database.

Through the publishing of this standard, and the creation of the XNF2LCA program, Xilinx ensured that any EDA company that could create XNF files would be in a position to offer advanced design methods for Xilinx devices.

—1987—

Introduction of the XC3000 Series

With the introduction of the XC3000 family, the density and complexity of the Xilinx programmable logic devices increased dramatically. Recognizing the associated growth in complexity of the programmable logic design methodology, Xilinx introduced a number of innovations to help ease the computing burden. The innovations included Automatic Place and Route, Map-then-Merge, XBLOX, Guided Design methodologies, and the XMAKE program.

Automatic Place and Route

To complete the automation of the layout of Xilinx logic devices, Xilinx developed the industry's first Automatic Place and Route (APR) program. APR 1.0 simplified the programmable logic design process by relieving the engineer of the time consuming tasks of identifying which CLBs within a LCA would contain the design's logic, and interconnecting them to other CLBs and IOBs. With the delivery of this program, the Xilinx design process went from being manual and interactive to being automated and batch oriented.

Map-then-Merge

Acknowledging the limitations of the performance of the typical engineer's computing platform, Xilinx patented a modular design method which is known as the Map-then-Merge approach. Taking clues from the source design's hierarchy, the Map-then-Merge approach divides the technology mapping task into smaller units, thus increasing the algorithmic throughput, and improving the processing time for designing complex programmable logic circuits. Furthermore, by separating the XNFMAP program from the XNFMERGE program, designers could discover the cause of any design errors more easily.

With the introduction of the XC3000 family, the density and complexity of the Xilinx programmable logic devices increased dramatically.

XBLOX

In an era where design reuse, and the inclusion of 3rd party Intellectual Property (IP or Cores) is at an all time high, it is interesting to review the XBLOX design tool that Xilinx offered its customers so long ago. In what was one of the industry's first high-level design methodologies, Xilinx patented an innovative mechanism for performing parameterized design using schematic capture. XBLOX allowed engineers to create a design with "n-width" buses, which could be specified in a single location within the schematic. The bus width parameter would then be propagated through the design, based on the designer's specifications. The propagation (or forward annoation) of the bus width parameter simplified the process of modifying a design's functionality by localizing this type of change to one design element.

XBLOX also included the ability to specify area or speed optimization of its functions on a "core-by-core" basis. The

In the era of schematic design and batch mode processing, Xilinx offered both interactive-based (GUI) processing and command line access to the algorithms that were required to layout Xilinx FPGAs.

XBLOX compiler was rerun at the time of layout, whenever design iteration called for a modification in the XBLOX specification or target device.

Guided Design

Attempting to accelerate the process of iterating on a design, Xilinx accelerated the place and route process by delivering the Guided Design capability. Guided Design allows an engineer to leverage the results from an earlier place and route run in the layout of his next version of an FPGA. While not effective in all design flows, this initial guided design technology proved to be very effective in accelerating the placement and routing of most FPGA design's when only small (iterative in nature) changes were required to a schematic.

XMAKE

In the era of schematic design and batch mode processing, Xilinx offered both interactive-based (GUI) processing and command line access to the algorithms that were required to layout Xilinx FPGAs. However, unless a designer was sitting in front of a design flow chart, the sequence of commands was not always easy to remember. To further simplify the programmable logic design process, Xilinx introduced the XMAKE program. XMAKE was designed to perform date stamp analysis, and to automatically run any out of date step in the design flow.

—1992—

Advanced Design Specifications

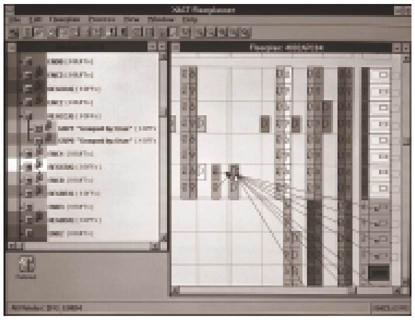
With the mainstream use of Programmable Logic Devices came a growing hunger for performance and efficiency in the use of silicon. To help its customers take advantage of the density and performance of Xilinx Logic devices, Xilinx introduced a set of tools that allowed an engineer to provide more information about a design to the layout tools. Armed with the new information provided by TimeSpecs, Relationally Placed Macros (RPM), and the Floorplanner, the layout tools were more prepared to focus on the true design requirements.

XACT Performance

Prior to the introduction of XACT Performance and TimeSpecs, skilled programmable logic designers manually placed and routed a design, or played tricks with a signal's "net criticality" in order to improve a circuit's performance. XACT Performance changed all of this by allowing design engineers to specify the circuit's performance criteria using Timespecs. Given this new information, the XACT Performance place and route tools were able to algorithmically evaluate the suitability of various layouts. While this processing required more CPU time, batch mode processing freed the designer to attend to the other requirements of his job, thus improving design time and time to market.

Floorplanner

Xilinx introduced another revolutionary capability to the programmable logic design market with the release of the XACT Floorplanner. The Floorplanner enabled a designer to impart his expert knowledge of the design as guidance to the layout process, thus simplifying the increasingly complex problem of logic placement. Advanced programmable logic designers found that by "seeding" the placement of a small portion of a design, they could improve circuit performance by up to 25%, and cut place and route run times in half. Furthermore, the Floorplanner proved to be an excellent tool for analyzing a design's logic and routing density, providing insight into the suitability of a design for future enhancements.



—1994—

Graphical User Interfaces

With the introduction of powerful Pentium processors, and the Windows operating systems, the use of personal computers as engineering workstations became common. To foster the adoption of its operating system, Microsoft established standards for GUIs that would enable first time users of an application to feel comfortable in learning to use any new product which was designed using these standards.

Continued on the following page

With ambitions of expanding the market for programmable logic, Xilinx developed a GUI for the XACT implementation tools that adhered to MFC standards. In addition to creating a GUI which simplified the means by which the tools were run (Xilinx Flow Engine), Xilinx also added a variety of design management capabilities to this tool set, simplifying the process of managing a design's source and object files (Xilinx Design Manager).

—1995—

Xilinx Acquires NeoCAD

Xilinx acquired the advanced FPGA implementation technology and resources of NeoCAD to more rapidly deliver state of the art support for a growing number of devices. The result of merging the technology and resources of these two companies has been the creation of a powerful, fast, timing-driven set of implementation tools for both CPLDs and FPGAs.

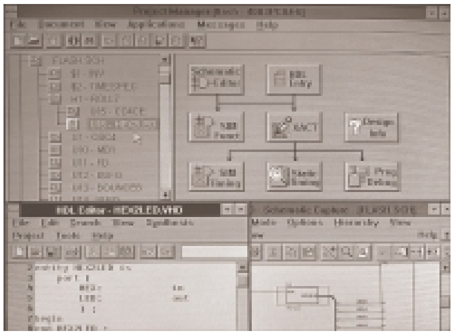
Some key software milestones that have occurred directly as a result of the acquisition:

- The first release of software support for new Xilinx architectures and devices in advance of sample silicon.
- The development of the first FPGA that has been optimized for fast and efficient layout, by customers using push-button design flows.
- The creation of the industry's best standards-based interface for working with 3rd party EDA providers, including EDIF, VHDL, Vital, Verilog, and SDF.
- The development of advanced performance driven algorithms that now run ten times faster than previous releases of XACT and NeoCAD Foundry implementation tools.
- The first localization of programmable logic tools for the Japanese market.

—1996—

Ready-to-Use PLD Design Tools

Shortly after Xilinx and NeoCAD joined forces, Xilinx introduced its new suite of tools for the design of programmable logic devices — the Foundation Series.™ The Foundation Series software included all of the basic necessities required of an easy-to-use design package, plus some capabilities that only the most advanced EDA companies were offering. The product was so well received that within the first two years, Xilinx had shipped over 10,000 seats.



To improve the Foundation Series quality of results, Xilinx signed an OEM agreement with Synopsys, Inc. for the inclusion of its FPGA Express.™ FPGA Express strengthened the Foundation Series by enabling engineers to design with VHDL, Verilog, or a mixture of the two languages — a trend that has become more popular as device densities have skyrocketed, causing more people to turn to design reuse methodologies.

—1997—

The Xilinx CORE Generator™

As a result of the close working relationships with its foundries and the inherent advantages of using programmable logic as a process leader, Xilinx devices leapt to the forefront of process technology. The immediate benefit of this process leadership role to Xilinx and its customers was rapid advancements in device densities and performance. These benefits positioned Xilinx as a viable alternative for the heart of many state of the art electronic systems. As such, developers of emerging standards began turning to Xilinx as a potential provider of qualified solutions. This in turn spurred the creation of the Xilinx LogiCore and Alliance Core programs. Rather than merely being a conduit for marketing qualified VHDL and Verilog designs, Xilinx core programs focussed on developing a flexible, parameterized core delivery capability, where hard, soft, and firm cores are all made available to customers through a tightly integrated design interface – the Xilinx CORE Generator.

Conclusion

Xilinx has made key advancements in both device and software technologies, and our programmable logic solutions will continue to lead the industry for years to come.