EDN®

JULY **24**

Issue 15 / 2008

www.edn.com

VOICE OF THE ENGINEER

MAGNETIC MEASUREMENT
TOOLS ATTRACT ATTENTION

HDL-DESIGN CHALLENGES AND PHILOSOPHIES FOR
REAL-WORLD ASIC IMPLEMENTATIONS

SPREAD-SPECTRUM-CLOCK GENERATORS REDUCE EMI AND SIGNAL-INTEGRITY PROBLEMS

MEASUREMENT

PROTOTYPING WITH FPGAs WORKS BEST IF YOU DO IT WITH THE FINAL ASIC IN MIND.

HDL-design challenges and philosophies
for real-world ASIC implementations

BY JESSE CHEN • SILVUS TECHNOLOGIES

New requirements for the MAC (medium-access control) and PHY (physical-layer interface) of a wireless-communications system can pose significant challenges for system designers looking to quickly get from development to production. This situation holds especially true as the demands for wireless connectivity and increased data rates continue to grow rapidly. The migration to new standards, such as 802.11n and WiMax (worldwide interoperability for microwave), requires designers to add new features, which necessitates the addition of a significant number of resources to design validation, testing, and integration of already-large, complex designs. FPGAs are excellent, cost-effective resources when it comes to initial design validation. Designers may spend a considerable amount of time prototyping not only new functional blocks, but also the entire wireless system on an FPGA platform.

When a design cycle reaches the point at which a custom-silicon option is necessary for real-world performance and reliability testing, designers must translate the FPGA design into an ASIC implementation for the end product to be viable. This situation ultimately leads to a key question for any SOC (system-on-chip) developer looking to add cutting-edge wireless functions to a design: How can you make the transition from a rock-solid FPGA design to a viable ASIC as swift and as painless as possible? The answer lies in an HDL (hardware-description-language)-design philosophy that keeps in mind from the outset the needs of the ASIC engineers: speed, area, and power efficiency.

FPGAs AND ASICs

Often, wireless-system designers who are developing their prototypes for validation on an FPGA are perfectly content with designs that run at speeds significantly lower than the target application requires. They expect a significant speed increase when porting the FPGA design to an ASIC implementation. This expectation is not unreasonable; users have reported FPGA designs containing logic, memory, and DSP blocks to experience an approximately three-fold speed increase when porting them to a standard-cell ASIC of the same technology—that is, the same minimum feature size (**Reference 1**). However, you need not be content with passively

AT A GLANCE

↘ An HDL (hardware-description-language)-design philosophy must keep in mind the needs of ASIC engineers: speed, area, and power efficiency.

↘ You need not be content with passively hoping that faster blocks will give you a faster system.

↘ Any performance increase in the design due to migration from an FPGA to an ASIC directly increases the design-speed margin.

↘ ASIC designers have turned to multiple-supply and multiple-threshold design techniques.

↘ "Siliconization"-design and HDL-coding guidelines ensure swift and seamless porting of FPGA designs to ASIC designs.

hoping that faster blocks will give you a faster system. When porting a typical FPGA design, including logic, memory, and DSP blocks, to a standard-cell ASIC of the same technology, one example showed a 20-fold increase in area performance, a twofold to threefold increase in speed, and a ninefold increase in dynamic power.

Both power and speed are critical metrics of a wireless-communications system. Companies often design—and heavily optimize—802.11n PHY and MAC IP (intellectual property) to run on an FPGA at the target speed that the final ASIC implementation requires: clock speeds in excess of 100 MHz for the critical, fastest signal-processing clock domain in the PHY. This optimization benefits both the IP designer and the customer by allowing validation of the design on a real-world platform with real-world situations. When the vendor enables true over-the-air data transmission and interoperability with consumer off-the-shelf hardware, there is no longer a need to rely on the promised, though unknown, performance increase that an ASIC offers. This relief increases confidence in the validity of the design in
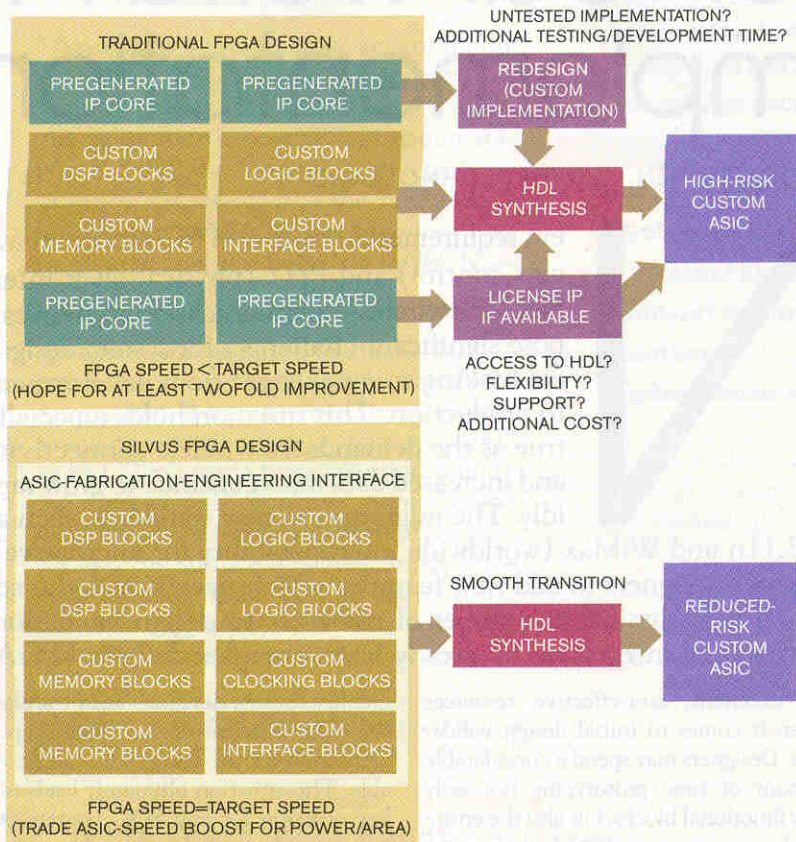

Figure 1 Companies such as Silvus Technologies have established a set of "siliconization"-design and HDL-coding guidelines to ensure the swift and seamless porting of an FPGA design to an ASIC.
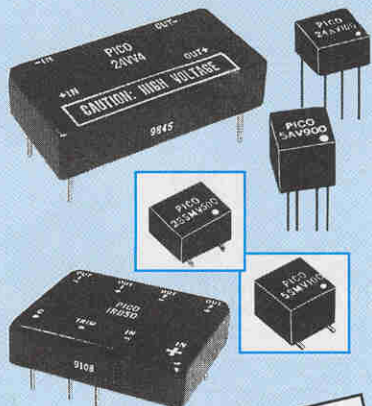
addition to reducing tapeout risk, potentially saving approximately $1 million to $3 million in total design cost per iteration.

Perhaps an even more important consequence of an FPGA design that runs at the speed of its target application is that any performance increase in the design due to migration from an FPGA to an ASIC directly increases the design-speed margin. Instead of using ASIC technology to "catch up" to the target speed, a chip designer looking to integrate the MAC and PHY blocks into a larger system now possesses a generous performance margin to play with to maximize power efficiency, which is crucial to the viability of a wireless system. The unrelenting scale-down of modern semiconductor-fabrication technology deep into the submicron world has turned circuit-leakage power into a major design consideration. Transistor-leakage currents in the form of subthreshold conduction between source and drain and both gate-oxide and junction tunneling have become major contributors to overall system-power dissipation: approximately 10% for 0.13-micron technology and rapidly increasing with each generation (**Reference 2**).

To combat this now-infamous implementation challenge, ASIC designers have turned to multiple-supply and multiple-threshold design techniques. In their advanced technology families, the leading foundries have allowed the designer access to double- and even triple-gate-oxide technologies. This access enables a mixed-threshold design, inside which the designer may choose from multiple types of transistors to minimize leakage. Designers may use thin-gate-oxide devices with higher speeds and lower switching thresholds to significantly improve the critical-path performance of the system. Although these devices generate more static-power dissipation in the form of leakage currents, they usually make up a small percentage of the overall design. The designer then places thick-gate-oxide devices with lower speeds and higher switching thresholds everywhere else in the design to maximize power savings—a step that can translate into tremendous energy savings. Unfortunately, such a mixed-threshold design requires careful analysis because modifications to the current critical path can lead to

the emergence of new ones. An FPGA design that requires no speed increase to function at maximum performance can hand its generous speed margin as an ASIC implementation to the SOC-integration engineer, who may simply be able to synthesize it with all thick-oxide devices and realize the maximum energy savings with minimum additional optimization effort.

DISCIPLINED HDL CODING

To ensure the swift and seamless porting of an FPGA design to an ASIC, designers must adopt a disciplined HDL-coding methodology. Some vendors have established a set of "siliconization"-design and HDL-coding guidelines. Strict adherence to these guidelines has yielded a unified design that can target an FPGA just as easily as it can an ASIC.

The initial rule involves the use of IP in the FPGA. The FPGA designer may wish to save initial development time by using pregenerated IP cores or HDL constructs, often from the FPGA vendor, which are not synthesizable with an ASIC-vendor library. These practices can indeed save time—on the FPGA—but the designer should strictly avoid them; they will quickly lead to FPGA designs that require significant redesign and patchwork after FPGA validation but before the design becomes acceptable for an ASIC implementation (**Figure 1**).

Designers should be sure to instantiate any vendor- or customer-specific components, such as memory modules or I/O buffers, in the highest top-level design entity. For example, use a pad-ring wrapper to interface all I/Os with the desired package contacts. This wrapper can include all tristate I/O buffers and associated registers with all input, output, and tristate-enabled lines tunneling up from the rest of the design. With all these FPGA-specific components at the top level, you can use global or generic constructs to select among components for an FPGA implementation or vendor-specific components for an ASIC target. You can use a similar strategy with memory modules. In addition to providing for easy vendor customization, this strategy gives the ASIC engineer the freedom to designate, for example, a FIFO-depth threshold above which synthesis selects an SRAM- instead of a register-based implementation. These types of

design methodologies may initially complicate a modular design, though, because design often requires the components of interest at the lowest level of the hierarchy. However, the benefit is worth the hassle because it allows the vendor to easily configure, drop in, and wire these components—which you should not synthesize from behavioral code—without error-prone modification of individual lower-level submodules.

CLOCKS, FLIP-FLOPS, AND RESETS

Playing around with multiple clock domains and clock gating yields additional power savings, but designers must again be careful of how the HDL is written. The dynamic-power dissipation of a block is roughly proportional to the frequency of its input clock, so one simple power-saving approach is to use slower clocks in blocks that do not require processing to occur as quickly. Such a scheme necessitates the use of a clock generator/manager circuit that, again, you should pull to the top level so it is visible to the vendor. Use clock gating to kill dynamic-power consumption of temporarily unneeded blocks, such as those in the transmitting chain when in receiving mode and vice versa, as well as local clock-tree buffers. Use clock multiplexers to accomplish this task—also instantiated on the top level. This methodology for the use of clock managers and multiplexers resonates with the theme of top-level visibility—for designer flexibility, vendor-specific components, and, in this case, clock-tree analysis.

As with any other good ASIC design, all flip-flops should trigger on the rising edge and use a global asynchronous reset. This approach will enable ASIC-scan-chain testing without logic changes. Avoid locally generated asynchronous resets, but when they are absolutely necessary, multiplex them with a scan-enable signal—which will select the global reset during a scan test—so that the flip-flops in question won't be untestable. Carefully monitor synthesis reports for inferred latches; they are unreliable and untestable via scan chains. Pull the global asynchronous reset, as well as any local resets and scan-chain-enable logic, to the top level so it is visible to the vendor for reset-tree analysis (references 3 and 4).

ELIMINATE SURPRISES

ASIC development is a tricky business. Poor design-for-ASIC practices and convoluted development flow only add to the risk of potential setbacks and additional cost. The key to success is to take nothing for granted. A design that works at the target application speed on an FPGA frees the ASIC engineer to work his magic with power and area savings. When it comes to complex systems involving multiple types of functional blocks, proper design and HDL-coding doctrine and the discipline to follow them can significantly reduce the risk at tapeout. You invariably lower the number of refab and test iterations and ultimately reduce development cost and time to market. An SOC designer looking to quickly drop a wireless system into his chip inevitably has to turn to an outside IP provider who has had the time and resources necessary to develop such a complex system. When it comes down to a choice between an IP option that a designer has thrown together on an FPGA without proper regard for the ASIC transition and a design that he has meticulously prepared, streamlined, and built for a single-step transition to ASIC, the choice is clear.EDN

AUTHOR'S BIOGRAPHY

Jesse Chen is an FPGA/ASIC-hardware design engineer at Silvus Technologies. He received a master's degree in electrical engineering of integrated circuits and systems from the University of California—Los Angeles and is still involved in cutting-edge research at UCLA's WISR (Wireless Integrated Systems Research) Group. You can reach Chen at jesse@silvuscom.com.

REFERENCES

[1] Kuon, I, and J Rose, "Measuring the gap between FPGAs and ASICs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Volume 26, Issue 2, February 2007, pg 203.

[2] Su, H, et al, "Full chip leakage estimation considering power supply and temperature variations," *Proceedings of the 2003 International Symposium on Low Power Electronics and Design*, 2003.

[3] *The International Technology Roadmap for Semiconductors*, Semiconductor Industry Association, 2001.

[4] Sirisantana, N, and K Roy, "Low-power design using multiple channel lengths and oxide thicknesses," *Design & Test of Computers*, IEEE, January/February 2004.