# HardWire: *Ensuring a Successful FPGA to*

*"Here are some of the critical issues for ensuring a successful FPGA conversion."*

HardWire™ conversion of FPGAs into ASICs gives system designers the power of programmability, greatly accelerating the design phase, while adding the cost-effectiveness of a true ASIC solution, especially for density ranges of 25,000 gates and above.

Since 1991, Xilinx's HardWire service has completed more than 700 conversions of flexible FPGA designs to low-cost ASICs ready for volume production.

Here are some of the critical issues for ensuring a successful FPGA conversion.

## 1. Use Synchronous Design Methods

We recommend that you design your FPGA with the production solution (ASIC) in mind. FPGAs can sometimes "hide" design flaws. These flaws can manifest themselves in the ASIC version when the design speeds up. The most common issues are with asynchronous paths and simultaneous switching output noise.

To ensure consistent timing, your design must be synchronous. Because FPGAs are RAM-based devices, an ASIC conversion will remove all the programmable elements and replace them with metal vias. In almost every case, the device timing speeds up substantially. If a design is asynchronous, the timing relationships may not behave the same in the ASIC as they did in the FPGA, perhaps causing race conditions.

In asynchronous FPGA designs, the small glitches generated by unstable outputs can be filtered by the pass transistors used to control the routing of long nets. However, in the ASIC version, those pass transistors are replaced by metal vias, possibly allowing an unfiltered glitch to propagate throughout the system.

If your design must be asynchronous, it is imperative that you carefully plan the timing relationships on the device itself, and between chips at the system level. Building in generous timing margins can help.

# ASIC Conversion

## 2. Thoroughly Simulate Your FPGA Design

Xilinx does not require functional or timing simulations, prior to FPGA/ASIC conversion. However, the FPGA design destined for an ASIC should be exhaustively simulated.

Unit delay simulation can be thought of as "best-case" simulation, since the logic will usually perform under the actual unit delay. This can set the "timing minimum" pole. The maximum simulation sets the "timing maximum" pole. If there are no functional differences between the maximum and minimum, then the design is likely to be free of timing dependencies.

## 3. Plan Your RAM Usage

RAM on an FPGA device is very efficient. However, RAM on an ASIC can be inefficient if not well planned,; and large RAM blocks may require extensive silicon area. Generally, if a design's RAM requirements exceed 25-35% of the total CLBs used, the die size will increase substantially, perhaps requiring triple-level metal for additional routing. One single-port RAM bit equals roughly four to six gate array gates; one dual port RAM bit can require seven to ten gates. If large amounts of RAM are required, it may be appropriate to leave part off-chip, or in extreme cases, consider a standard cell implementation.

## 4. Pay Attention to FPGA Configuration Modes

FPGA configuration modes are key. In the FPGA, data is stored in the PROM. The PROM is downloaded to the FPGA via one of many configuration modes, allowing the system to "wake up" in an orderly manner. During conversion, the normal configuration mode is "instant on." If the ASIC device is still dependent on other events prior to "waking up," you should implement the configuration mode into the ASIC. Otherwise, system-level timing errors may result or ASICs might appear non-functional. Prior to ASIC conversion, the configuration scheme must be well documented so that configuration logic can be included in the ASIC version.

## 5. Select Your Best Vendor

Finally, select a vendor who can handle the features, density, and volume production of the resulting ASIC. Many ASIC vendors offer a "conversion service" that is nothing more than netlist translation into a third-party library. After the netlist translation, customers must re-simulate and re-validate both timing and functionality.

Xilinx supports full turnkey conversion, using Xilinx-specific tools and technology. The Xilinx-converted ASIC has Xilinx-specific features built into the die to eliminate the mismatch between the FPGA features and the ASIC implementation.

HardWire is the only FPGA conversion method that supports Xilinx devices with state-of-the-art technology and guarantees success. Xilinx design engineers work closely with you to ensure that all considerations have been reviewed. The FPGA is converted to a Xilinx HardWire device using the same fabrication facilities used to make the FPGA, with greater than 90% first-attempt success rates.

When considering an FPGA to ASIC conversion project, it is wise to review all the design considerations, perform the critical system level testing, and most of all choose a vendor who can support your specific requirements. ◆

*"When considering an FPGA to ASIC conversion project, it is wise to review all the design considerations, perform the critical system level testing, and most of all choose a vendor who can support your specific requirements."*