WP206 (v1.2) March 1, 2004

The 40% Performance Advantage of Virtex-II Pro FPGAs over Competitive PLDs

By: Hitesh Patel

As programmable logic devices (PLDs) increase in density and complexity, the combination of a feature-rich fabric and sophisticated design tools enables users to realize their performance goals in less time. Shorter design cycle times enable users to lower overall design costs and meet time-to-market requirements. This white paper highlights how the Virtex-II Pro™ FPGA and ISE6 design tool combination provides a 40% performance advantage over the nearest competitor, the Altera Stratix PLD.

© 2004 Xilinx, Inc. All rights reserved. All Xilinx trademarks, registered trademarks, patents, and disclaimers are as listed at http://www.xilinx.com/legal.htm.
All other trademarks and registered trademarks are the property of their respective owners. All specifications are subject to change without notice.

Performance Distribution

The 40% performance average is derived from analyzing 50 customer designs with densities ranging from 200 thousand to 6 million system gates. Virtex-II Pro FPGAs were up to 123% faster than Stratix PLDs.

Figure 1 shows the performance distribution.

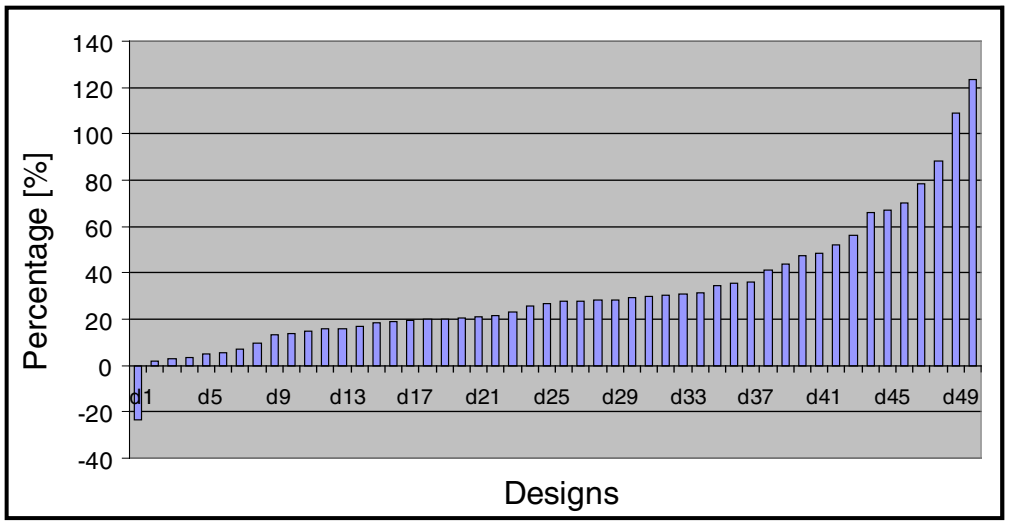

Figure 1: **Virtex-II Pro Performance Advantage vs. Stratix for 50 Customer Designs**

The performance graph in Figure 1 shows that the Virtex-II Pro FPGA outperformed the Stratix PLD in 49 designs out of 50. The one instance where the Virtex-II Pro FPGA was outperformed occurred because our analysis used default settings in synthesis with pipelining "off". Because the specific design had a multiply function on the critical path, the Stratix design had an instantiated pipelined lpm (black-box function generated by Quartus MegaWizard) multiplier. For the Virtex-II Pro design, synthesis inferred the MULT18x18 primitive. By changing pipelining to "on", the synthesis tool inferred a MULT18x18S primitive for the Virtex-II Pro design, resulting in an implementation with faster performance compared to the Stratix design. Thus, comparing equivalent speed grades for real-world designs, Virtex-II Pro FPGAs almost always outperform Stratix PLDs.

Architecture Features

The basic primitive in the Stratix architecture is called a Logic Element (LE). An LE contains three functional structures: a 4-input look-up table (LUT), a register, and a carry chain. The basic primitive in the Virtex-II Pro architecture not only includes the structures found in an LE, but also provides additional functionalities such as a MUXF function expander, a MULT_AND arithmetic cell, and a more logic-rich carry structure. Furthermore, the Virtex-II Pro LUT can be used as either a 16-bit shift register or a single- or dual-port RAM element. With these additional capabilities in the Virtex-II Pro architecture, users can realize higher design performance. This section describes these additional features and compares them with their equivalent implementations within Stratix PLDs.

MUXF Function Expander

Logic levels in the signal path are one of the primary factors impacting circuit performance in FPGAs. The function expander cell represents a 2:1 mux used to build functions with wider than four inputs without using additional LUT logic levels. For example, when using the Virtex-II Pro MUXF, only four LUTs are required to implement an 8:1 mux in a single LUT

logic level. The Stratix PLD implements the same 8:1 mux using five LUTs in two LUT logic levels. The additional LUT logic level in the Stratix design adds delay to the signal path.

The Virtex-II Pro function expander is not limited to multiplexers and can be used for many other logic functions. For example, a MUXF combined with two LUTs can implement any 5-input function, thereby implementing a full 5-input LUT in a single LUT logic level. A similar Stratix implementation requires two or three LUTs depending on the function and uses two LUT logic levels.

Figure 2 shows an example of a 9-input function. For the Virtex-II Pro architecture, this function is mapped onto two LUTs with one function expander, using a single LUT logic level. For the Stratix implementation, the same function requires three LUTs and two LUT logic levels.

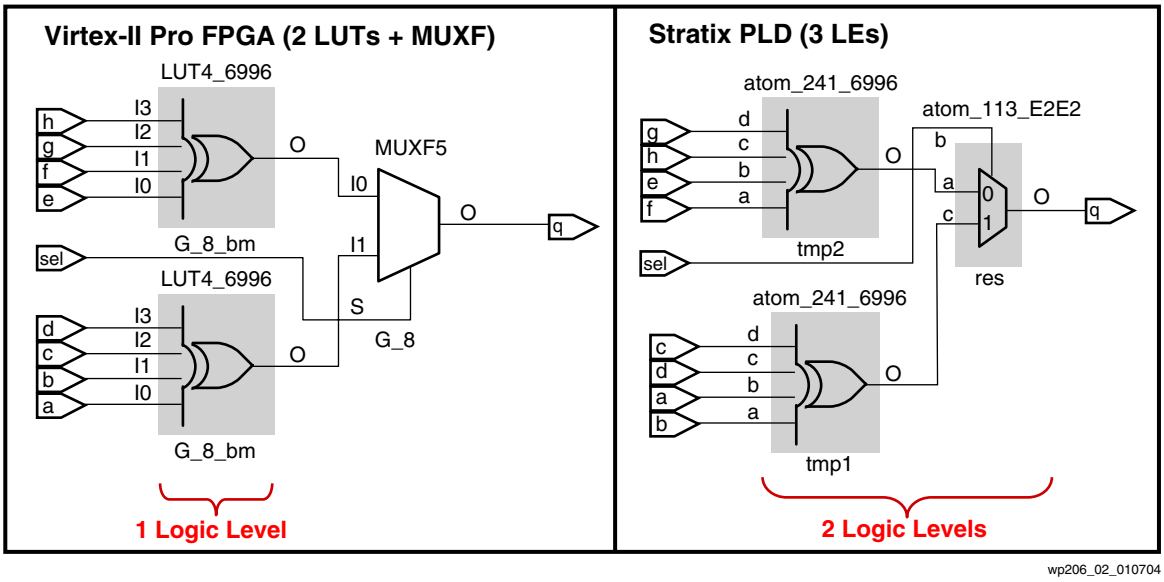

Figure 2: **9-Input Function Mapped to Virtex-II Pro and Stratix Devices**

The MUXFx component is equivalent to having a 5- or 6-input LUT, leading to fewer logic levels and far fewer LUTs consumed (10% on average) than the same function in Stratix devices. This configuration results in higher performance for Virtex-II Pro designs because fewer logic levels generally are required for critical paths and, at the same time, less placement and routing congestion occurs because 10% fewer resources (LUTs) are necessary to build the same functionality.

SRL Shift Register LUT

A single LUT in the shift register mode (SRL) can implement a selectable 16-bit shift register. The same shift register in a Stratix PLD is implemented either using 16 flip-flops and up to 10 LUTs or in a memory block (a less flexible method). In the Stratix PLD, if the shift register cannot be implemented in a memory block, then it is implemented using 16 LEs, which creates added routing congestion that might impact design performance. If the Stratix shift register requires variable tap selection, logic levels are added on the output path, resulting in much slower operation.

MULT_AND Primitive

The Virtex-II Pro MULT_AND primitive is used commonly in soft multiplication applications. The flexibility of the FPGA fabric allows some 5-input functions to be mapped onto a single LUT. For example, loadable up and down counters implemented with the MULT_AND primitive utilize only one LUT per bit instead of two LUTs per bit as in the Stratix PLD. Due to the fewer logic levels and fewer required LUTs to implement this function, Virtex-II Pro FPGAs can achieve up to 30% increased performance.

SelectRAM Components (LUT-based RAMs)

In the Virtex-II Pro architecture, a LUT can be configured as a single- or dual-port RAM, resulting in very fast read and write accesses for smaller data storage and buffering applications. In the Stratix PLD, the smallest RAM configuration (the M512 blocks) offers much slower RAM operation and less flexible dual-port access while at the same time requiring greater latency for reads. The maximum read speeds for the M512 RAMs are 266 MHz for one clock cycle reads and 320 MHz with two clock cycle latency. The Virtex-II Pro SelectRAM™ components provide up to 360 MHz read operation with a single clock latency and offer asynchronous read capability for low latency design requirements. Because small RAMs are often used in implementations such as data storage for small FIFOs, coefficient storage for DSP filters, and buffers for packet processing, having maximum performance in this structure often can enable designers to meet their system performance requirements.

Block RAMs

Because most designs typically utilize a majority of available RAM memory on the device, many times Stratix users are forced to use the MegaRAM memory blocks to create their desired functionality. Xilinx evaluated the read/write performance of Virtex-II Pro block RAM configured to the same width (4k x 144) and depth (64k x 8) as the Stratix MegaRAM memory. The results, as presented in Table 1, show that for the deep and wide configurations with one clock delay, the memory read time performance in the Virtex-II Pro FPGA is approximately 40% (shown in red) and 95% (shown in blue) faster, respectively, than the Stratix PLD.

Table 1: **Virtex-II Pro (-7) and Stratix (-5) Block RAM Performances**

Configuration	Clock Delays	Write Speed		Read Speed	
		Stratix PLD [MHz]	Virtex-II Pro FPGA [MHz]	Stratix PLD [MHz]	Virtex-II Pro FPGA [MHz]
Deep Single Port Memory (64k x 8)	1	287	282	199	282
	2	287	282	287	282
Wide Single Port Memory (4k x 144)	1	255	284	145	282
	2	255	287	255	287

The wide MegaRAM configuration has approximately 300 signals that must connect to the relatively small footprint of the memory block, leading to registers and logic competing for the optimal placement locations of a few sites in the array closest to these memory pins. The additional routing congestion of these signals impacts overall memory performance. Because the Virtex-II Pro configuration was created using smaller RAMs spread out over a greater area of the chip, a more optimal placement and routing is achieved, resulting in higher performance.

Multiply-Accumulate (MAC) Function

The Stratix PLD contains a dedicated DSP block with MAC function that often is assumed to be able to outperform the same function created in a Virtex-II Pro device. Figure 3 highlights the maximum performance with latency for two MAC sizes: 9 x 9 and 18 x 18. This analysis shows that the MAC function for the Virtex-II Pro FPGA outperforms the MAC function for the Stratix PLD.

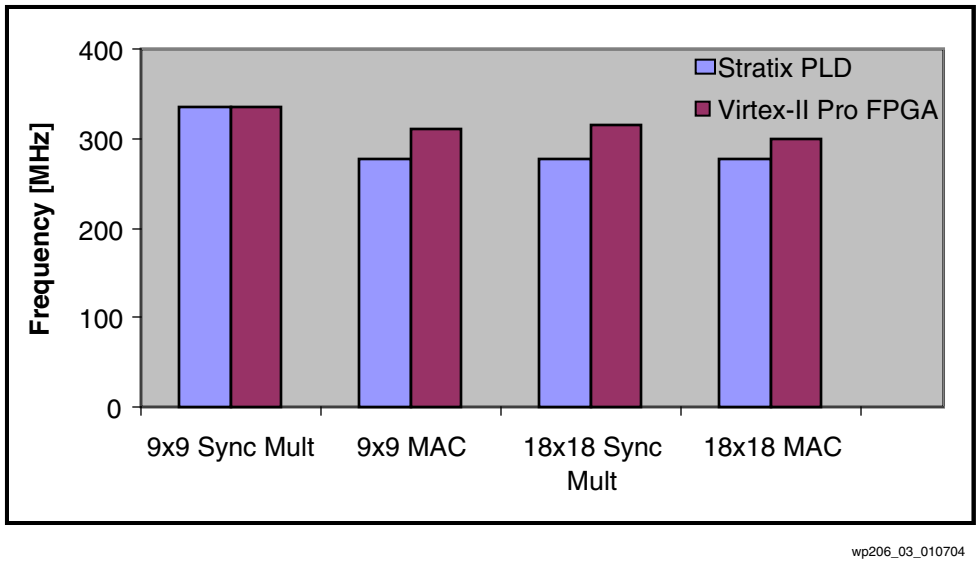

Figure 3: **Virtex-II Pro (-7) and Stratix (-5) MAC Performances**

Software Features

The FPGA fabric feature set continues to offer capabilities that improve design performance and reduce area. In order for users to realize these benefits, the software tools, both synthesis and place and route, need to use these architecture capabilities.

Synthesis

FPGA centric synthesis tools constantly look for new optimization techniques that go beyond mere LUT mapping. These synthesis tools can extract known functions (such as arithmetic functions, memories, and multiplexers) by parsing the RTL code and then automatically mapping these functions to features on the target architecture. Synthesis mappings to the MUXF, MULT_AND, and SRL primitives are examples of synthesis tools providing architecture-specific mapping to reduce logic levels on the critical paths and to reduce place-and-route congestion, thereby improving overall design performance. Synthesis tools also automatically infer either the LUT RAM or block RAM based on the coding style and the size of memory being used. For example, Synplify may infer fast LUT RAMs for up to 2k bits of memory.

As FPGAs implement more advanced submicron technologies, routing delays become more predominant, and design performance is highly influenced by cell placement. Hence Xilinx provides detailed timing estimates to enable the synthesis tools to not only select the best architecture element for the implementation, but also to improve timing predictability between post-synthesis and post-layout. This close technical collaboration ensures that synthesis optimization is focused on the path critical to placement and routing.

Place and Route

A study done by researchers at UCLA[1] showed that timing driven placement algorithms for FPGAs can be on average 30% away from optimal results. The study also found that Xilinx synthesis tools outperform other tools in the industry. On average, the delay generated by the ISE placer was only 8.3% below optimal and only 4.1% below after routing. The ISE place-and-route software delivers near optimal results due to the years of fundamental research Xilinx has done on the segmented routing architecture. To illustrate this advantage, Xilinx compiled the "blowfish" encryption algorithm, an open source design, using ISE6 and Quartus 3.0 targeting Virtex-II Pro (-7) FPGAs and Stratix (-5) PLDs, respectively. Figure 4 represents the breakdown of logic and route delays for the critical path. This analysis shows that ISE placement technology provides near optimal placement, resulting in an 80:20 logic to route delay ratio for Virtex-II Pro devices. The Stratix implementation using Quartus leads to a 50:50 logic to route delay ratio. As a result, the Virtex-II Pro FPGA design is 2x faster than the Stratix design.

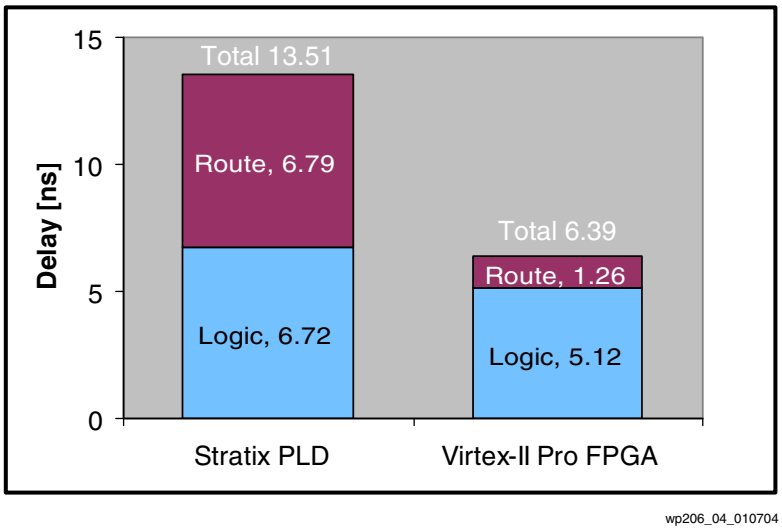

Figure 4: **Logic vs. Route Delay on the Critical Path for "Blowfish" Design**

The *Timing Driven Map technology*, new in the ISE6 software, is just one example of Xilinx's years of expertise in placement and routing for segmented architectures. This technology enables the placer to iterate between map and place, as shown in Figure 5, such that the placer can provide the mapper with suggested slice-level primitive mapping. This iterative loop leads to near optimal slice mapping and placement, resulting in improved timing because the router now can pick the best route with fewer conflicts for the same routing resources.

1. "Optimality and Stability Study of Timing-Driven Placement Algorithms" by Jason Cong, Michail Romesis, and Min Xie, UCLA, Technical Report #030030.

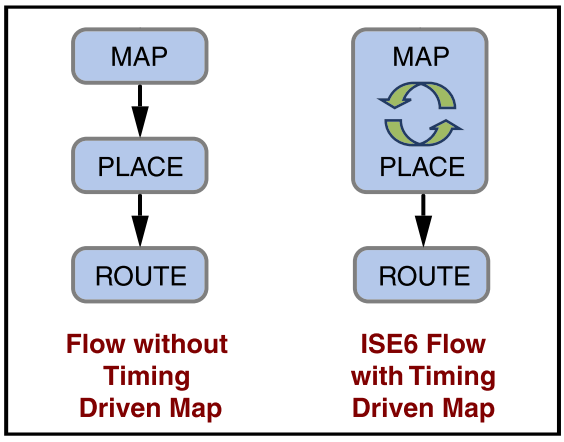

Figure 5: **ISE6 Timing Driven Map Flow**

In summary, users can maximize design performance with Xilinx software technology support of Virtex-II Pro architecture features and near optimal placement technology.

Conclusion

The advanced architecture features of Virtex-II Pro FPGAs (such as MUXFs, SRLs, MULT_ANDs, fast SelectRAM and block RAM components, and fast dedicated multipliers) contribute significantly to the FPGAs' superior performance. The 40% average performance advantage of Virtex-II Pro FPGAs over Stratix PLDs is due to the combination of the following attributes:

- an advanced architecture,
- the capability of the synthesis tools to access these architecture specific features, and
- the place-and-route software's ability to deliver near optimal placement for a segmented architecture.

In most cases, the fastest Stratix PLD speed grade must be used to reach the performance of the slowest speed grade for a Virtex-II Pro FPGA. The performance in the faster speed grades of Virtex-II Pro FPGAs cannot be matched by Stratix PLDs in any speed grade. Virtex-II Pro devices have reached a higher performance level not matched by any FPGA in the industry today.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
01/08/04	1.0	Initial Xilinx release.
01/14/04	1.1	Revised Figure 5. In **Place and Route** section, changed terminology from pack to map.
03/01/04	1.2	Updated performance advantage from 38% to 40%.