

Benchmarks Again Demonstrate XC9500 Pinlocking Capabilities

A new set of XC9500 pinlocking benchmarks focuses on two commonly-used comparator functions — magnitude and equality compares. Magnitude comparators are often used in digital peak detector circuits in audio and video applications. Equality comparators are a key part of page mode DRAM memory controllers.

As with previously-published CPLD benchmarks (*XCell 22*, page 18), the comparator benchmarks illustrate the CPLD device's capability to accommodate design changes with fixed pinouts while maintaining an acceptable level of design performance; in other words, not only must the iterated design route when the pinout is maintained, it must do so with minimal impact on design performance.

The following two sets of benchmark data show the relative pinlocking performance of the XC9500 CPLDs and a competitor's CPLD family. The designs are ABEL-HDL implementations of benchmarks suggested by the competitor, and are targeted at devices of the competitor's choosing. ABEL compiler and fitter options were chosen to maximize design performance, per each vendor's recommendations.

Magnitude Compare Benchmark

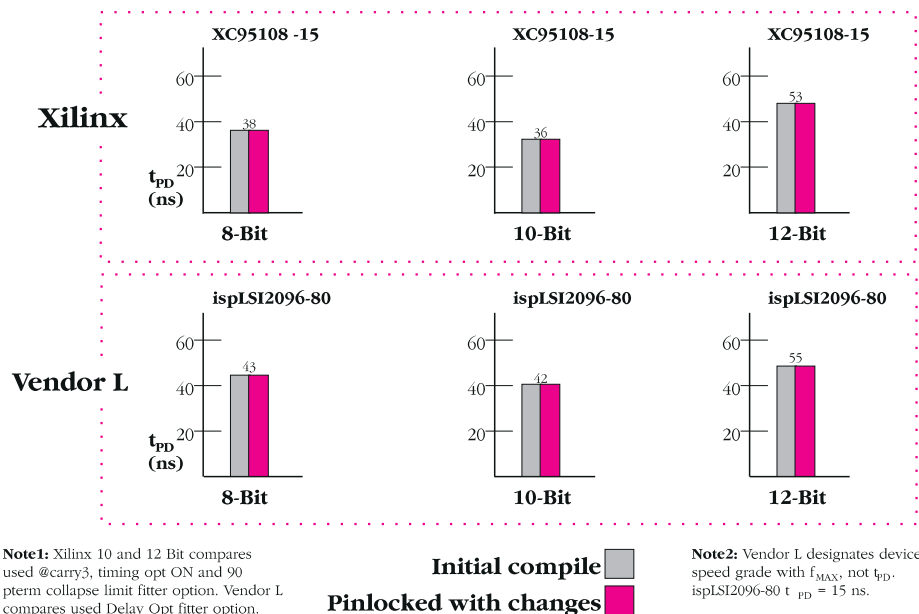
This benchmark contains a single 8-, 10-, or 12-bit magnitude comparator and is intended to measure the effect of routing resources and function block fan-in on the CPLD's pinlocking performance. A typical design change involves the correction of an error in which one of the input data paths was ordered incorrectly.

The benchmark results in **Figure 1** demonstrate that both the XC95108-15 and the ispLSI2096-80 were able to accommodate the design change without any impact on design performance. However, it should be noted that the XC95108-15 was faster overall than the other device.

Equality Compare Benchmark

This benchmark contains a single 8-, 10-, or 12-bit equality comparator and is intended

Figure 1: Magnitude Compare Performance



to measure the effect of routing resources and function block fan-in on the CPLD's pinlocking performance. A typical design change involves the correction of an error in which one of the input data paths was ordered incorrectly.

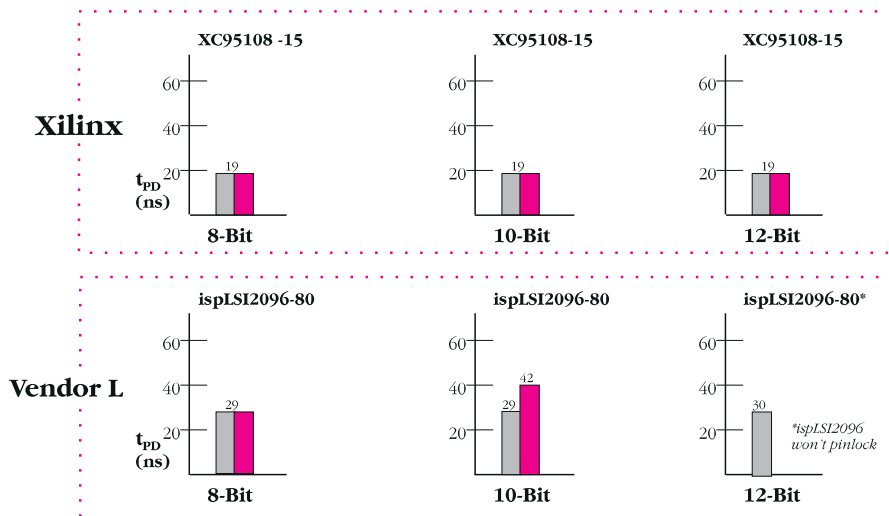
The benchmark results in **Figure 2** demonstrate the superiority of the XC9500 CPLD architecture. Not only was the XC95108 able to accommodate the design changes without any impact on design performance, that level of performance was significantly higher than the competitor's device. Vendor L's device

Continued on the next page

XC9500 Benchmarking

Continued from
the previous page

Figure 2: Equality
Compare Performance



Note 1: Xilinx compares didn't use @carry. Vendor L compares used @carry 4 to reduce run time and Delay Opt fitter option.

Initial Compile
Pinlocked with Changes

Note2: Vendor L designates device speed grade with f_{MAX} , not t_{PD} . ispLSI2096-80 $t_{PD} = 15$ ns.

used multi-level logic to implement the 8-bit equality compare; a 16-input, 16 product-term logic function that is implemented in a single pass in the XC9500 CPLD.

Vendor L's routing resources were stressed while attempting to pinlock the 10-bit equality compare; a 45% performance degradation was incurred after the design iteration. While attempting to pinlock the 12-bit equality

compare, the performance didn't just degrade, the device completely failed to route.

Conclusions

These benchmarks reconfirm the superior pin-locking performance of the XC9500 CPLD family. The wide function-block fan-in enables pinlocking of wide high-speed logic functions while at the same time delivering higher performance than competitive devices. Furthermore, because logic feedthroughs are

not needed for routing, there is no performance degradation due to routing congestion. This timing consistency is as important as routing ability for maintaining pinlocked designs.

The XC9500 CPLD devices deliver high performance and feature the industry's best pin-locking capability, eliminating the need for PCB modifications due to design changes. This feature not only shortens design cycles and decreases design costs but also facilitates the use of in-system programmability to upgrade or modify systems in the field. ♦