

Summary

This application brief presents benchmarks that demonstrate the superior pin-locking capability of the Xilinx XC9500 CPLDs. These benchmarks are based on typical applications and demonstrate the benefits of a highly routable switch matrix and wide function block fan-in when iterating pin-locked designs. The Xilinx results are compared to other vendors' CPLDs using their production fitters, proving that the Xilinx XC9500 family is the industry's best pin-locking CPLD.

Xilinx Family

XC9500

Introduction

The Xilinx XC9500 CPLD family provides the most advanced, most reliable pin-locking capability in the industry. This important feature allows designers to maintain pinouts after making design changes, eliminating costly, time consuming PC board re-work. CPLDs that do not have adequate pin-locking capability usually require new pinouts even after minor design changes, leaving no room for error and no possibility for field upgrades or field customization. Now, with the XC9500 family, designers can save time and money because they no longer need to modify PC boards every time they make a design change. In addition, this reliable pin-locking capability allows designers to use the in-system programmability features of the XC9500 family to upgrade or modify systems in the field.

This application brief demonstrates the advanced pin-locking features of the XC9500 family and provides pin-locking performance comparisons for competing devices.

Pin-Locking Issues

In most CPLDs, each I/O pin is driven directly by a macrocell through an I/O block as shown in Figure 1. When the design is pin-locked, the fitter is forced to map logic into specific macrocells to maintain the pinout. If the device architecture is limited, with inadequate routing in the central switch matrix, the fitter may not be able to place and route the design when the pins are locked.

Some CPLDs use an output routing pool in an attempt to compensate for their primary routing deficiencies. However, output routing pools introduce additional delays and do not prevent the fitter from having to consume logic resources as routing feedthroughs, impacting both design performance and resource utilization.

Logic requirements also affect the ability of the fitter to place and route the design when the pinout is locked. Slow speed designs with simple, narrow logic functions requiring few inputs, feedbacks, and product terms are inherently easier to pinlock than high speed designs with wide fan-in and product term intensive logic functions.

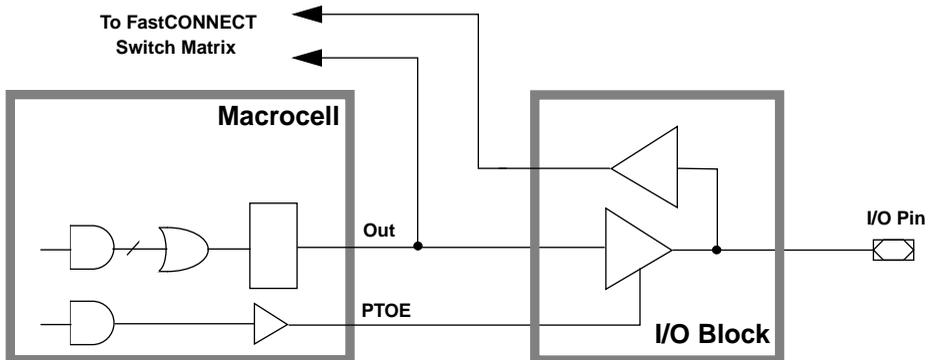


Figure 1: Simplified XC9500 I/O Architecture

The Keys to Reliable Pin-Locking

To address these pin-locking issues, Xilinx XC9500 CPLDs feature abundant routing resources, wide function block fan-in, and flexible product term allocation. The XC9500 fitter also optimizes the initial placement to maximize the design's pin-locking capability. Each of these factors is described as follows.

Routing Resources

Routability is a primary requirement for reliable pin-locking. The routing resources of a CPLD determine how much of the logic block resources (inputs, product terms, and registers) can be used to accommodate design changes after the pins are locked in a design. In a fully routable CPLD, buried logic can be moved without regard to routing restrictions, freeing function block resources that may be needed by the logic that drives the I/O pins.

The XC9500 family provides the most routing resources of any CPLD family currently available. The FastFLASH technology used in the XC9500 family uses smaller cell sizes than other technologies and therefore more routing switches can be packed into the same area. As a result, all devices in the XC9500 family are 100% routable; if there are enough function block resources to implement the design, it will route.

Pin-locking restricts the fitter's capability to place design resources and therefore good routability is crucial. With adequate routability, the constraints imposed by fixed pinouts can be overcome.

Function Block Fan-In Capability

Wide function block fan-in is another important requirement for pin-locking. Since CPLDs are typically used for high speed signal-intensive logic functions, wide function block fan-in is a requirement for implementing functions in a single logic level. The number of available function block inputs affects the fitter's ability to add more signals to any logic that must remain in that function block (because it drives I/O pins). Wide fan-in also helps the fitter implement that logic in a single pass through the device.

Each XC9500 function block has 36 inputs from the switch matrix. Other vendors' in-system programmable CPLDs have as few as 16 inputs.

Product Term Allocation

Product term allocation is important to pin-locking because it allows design changes that increase the product term requirement. All XC9500 devices allocate individual product terms from anywhere in the function block to the macrocell that needs them, accommodating logic changes when the design is pin-locked.

In the XC9500 family, up to 90 product terms can be allocated to any macrocell in the function block. This is in con-

trast to other vendors' CPLDs that restrict the product term availability (from 5 to 32 pterms) on the basis of macrocell location in the function block.

Fitter Strategy

Fitter software is a key component of any successful CPLD pin-locking solution. It must work in conjunction with the device architecture, spreading the outputs to accommodate design changes when the design is pin-locked.

The XC9500 fitter is optimized to take full advantage of the hardware resources of the XC9500 family. The fitting algorithms that determine how to place and route the design make full use of the abundant routing and product term allocation resources within an XC9500 device to give unparalleled pin-locking performance. The Xilinx fitter is capable of intelligently utilizing all available device resources to retain pinouts and still maintain the required performance, even after significant design changes.

Pin-Locking Benchmarks

The following benchmark data shows the relative pin-locking performance of Xilinx, Altera, Lattice, and AMD CPLDs. These benchmarks are based on typical applications such as address decoders, datapath designs, and address counters, in which reliable pin-locking is crucial. They illustrate the CPLD's capability to accommodate design changes while maintaining an acceptable level of design performance, because not only must the iterated design re-route when the pinout is maintained, it must do so with minimal impact on design performance. Therefore, all of the benchmark data presented in this application brief is normalized to the design performance achieved when the fitters are free to choose the pinouts without restrictions.

Synario™ was used for design entry to support retargeting to multiple CPLD vendors using identical ABEL code. The following fitters from the CPLD vendors were used to implement the benchmark designs:

- XABEL-CPLD v6.1 for Xilinx
- pDS+ v2.2 for Lattice
- MAX+2 v6.2 for Altera
- MACH Device Kit v2.3 for AMD

Each design was initially compiled by allowing the fitter to freely choose the pinout. After changes were made to the design, it was re-compiled using the previously assigned pinout. Design performance was measured using t_{PD} and external f_{MAX} as true measures of system performance, where external f_{MAX} is defined as $1/(t_{CO} + t_{SU})$.

Software and Device Availability

Not all of the other vendors' announced devices were supported by their software and therefore not all of their device densities and packages could be evaluated, as indicated in the following charts. Updated benchmarks will be published when available.

Address Decoder Benchmark

This benchmark design, shown in Figure 2 and Figure 3, measures the effect of routing resources and function block fan-in on the CPLD's pin-locking capability. The design contains two 16, 32, or 36 bit buses which are decoded to generate two chip select outputs. A typical design change, involving the correction of a typographic error in which the outputs are decoded incorrectly, is illustrated in Figure 4.

The benchmark results in Figure 11 demonstrate that both the Xilinx XC9500 family and the Altera EPM7000S devices were able to accommodate the design changes without impact on design performance. The Lattice devices maintained the same pinout with a significant (up to 60%) performance penalty. Since the Lattice devices have 16 input logic blocks, the performance degradation of the 16-bit address decoder can be attributed to poor routing resources while the performance of the 32 and 36 bit decoders is degraded by both poor routing and narrow logic block fan-in.

The AMD MACH 5 devices exhibited a 33% performance degradation in the higher pin count packages when the designs were pin-locked. This degradation resulted from segment delays incurred during re-routing (but not incurred during the initial design compilation). Additionally, the MACH 5 software was unable to route the 36-bit wide decoder during the initial compile. This can be attributed to poor fitter performance, inadequate routing resources, or both.

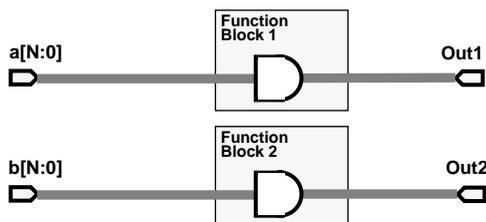


Figure 2: Address Decoder

```

MODULE SWAP
TITLE 'DECODER'
//inputs
a15..a0 pin; "A bus
b15..b0 pin; "B bus

//variables
a_bus = [a15..a0];
b_bus = [b15..b0];

//outputs
out1 pin istype 'com';
out2 pin istype 'com';

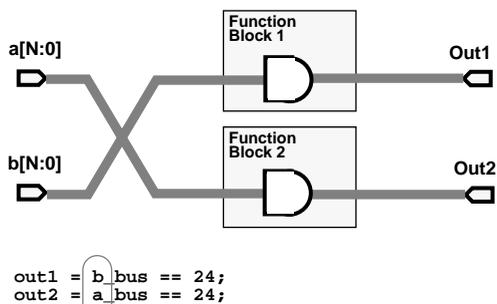
equations

out1 = a_bus == 24;
out2 = b_bus == 24;

END

```

Figure 3: Address Decoder Code



Note: In this example, due to a typo, the wrong address bits are decoded and therefore the bus must be rerouted.

Figure 4: Address Decoder Design Iteration

Datapath Benchmark

This benchmark design, shown in Figure 5 and Figure 6, measures the affect of routing resources on the CPLD's pin-locking capability. This design contains a single 16, 32, or 36 bit wide data bus. A typical design change involving the reordering of data bits is illustrated in Figure 7.

The benchmark results shown in Figure 12 show that the Xilinx XC9500, AMD MACH 5, and Altera EPM7000S devices were able to accommodate the design changes without impact on design performance. Both the Lattice ispLSI1000 and ispLSI2000 devices sacrificed performance (up to 80%) to reroute the design when pinlocked. Since only one logic block input was required for each output, this performance degradation can be attributed to poor routing resources, or fitter performance, or both, but cannot be attributed to logic block fan-in.

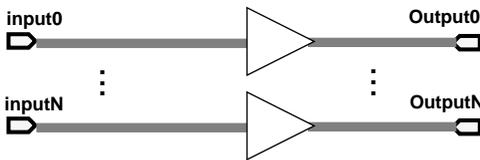


Figure 5: Data Path

```

MODULE REORDER
TITLE 'Datapath test'

//inputs
input15..input0 pin; "inputs

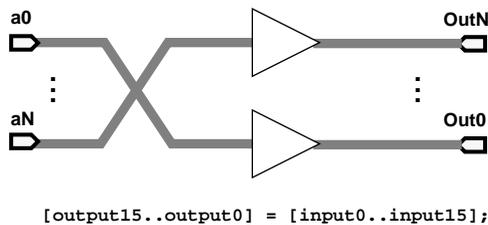
//outputs
output15..output0 pin istype 'com'; "outputs

equations

[output15..output0] = [input0..input15];

END
    
```

Figure 6: Data Path Code



Note: In this example, due to a typo, the data bits are ordered incorrectly therefore they must be reordered.

Figure 7: Datapath Design Iteration

Address Counter Benchmark

This benchmark design shown in Figure 8 and Figure 9, measures the effect of routing resources and function block fan-in on the CPLD's pin-locking capability when macrocell feedbacks and other high fan-out signals are involved. The design contains two 16, 24, or 32 bit loadable address counters loaded from separate buses but with common clock and hold signals. A typical design change correcting initial count load value is illustrated in Figure 10.

The benchmark results shown in Figure 13 demonstrate the superiority of the Xilinx pin-locking capability vs. Altera Lattice, and AMD. All Xilinx XC9500 devices were able to accommodate the design changes without impact on design performance. When the Altera EPM7000, EPM7000E and in-system-programmable EPM7000S routing resources were stressed, performance didn't just degrade, the devices completely failed to route. The Lattice ispLSI2000 devices used several layers of logic in the initial design, with correspondingly low f_{MAX} . This enabled the fitter to reroute the design using alternate routing paths, with less performance degradation (20%) than designs initially using only one logic level.

The MACH 5 devices were able to accommodate the design changes without incurring additional time delays for the 16- and 24-bit address counters. This was possible because segment delays were incurred during the initial design compilation and not just during the re-route. However, they completely failed to route the 32-bit wide counters during the initial design compilation. This can be attributed to poor fitter performance, inadequate routing resources, or both.

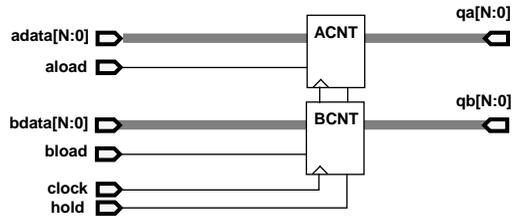


Figure 8: Address Counter

```

MODULE CNTSWAP
TITLE 'Counter Swapping'

//inputs
clock pin; "clock
hold pin; "counter hold
ain15..ain0, aload pin; a data bus
bin15..bin0, bload pin; b data bus

//outputs
qa15..qa0, qb15..qb0, pin istype 'reg';

//variables
account = [qa15..qa0]; adata = [ain15..ain0];
bcount = [qb15..qb0]; bdata = [bin15..bin0];
equations
account := adata & aload
# account & !aload & hold
# (account + 1) & !aload & !hold;
account.clk = clock;

bcount := bdata & bload
# bcount & !bload & hold
# (bcount + 1) & !bload & !hold;
bcount.clk = clock;

END

```

Figure 9: Address Counter Code

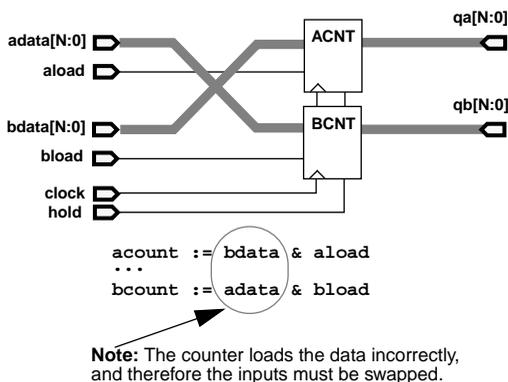


Figure 10: Address Counter Design Iteration

Conclusion

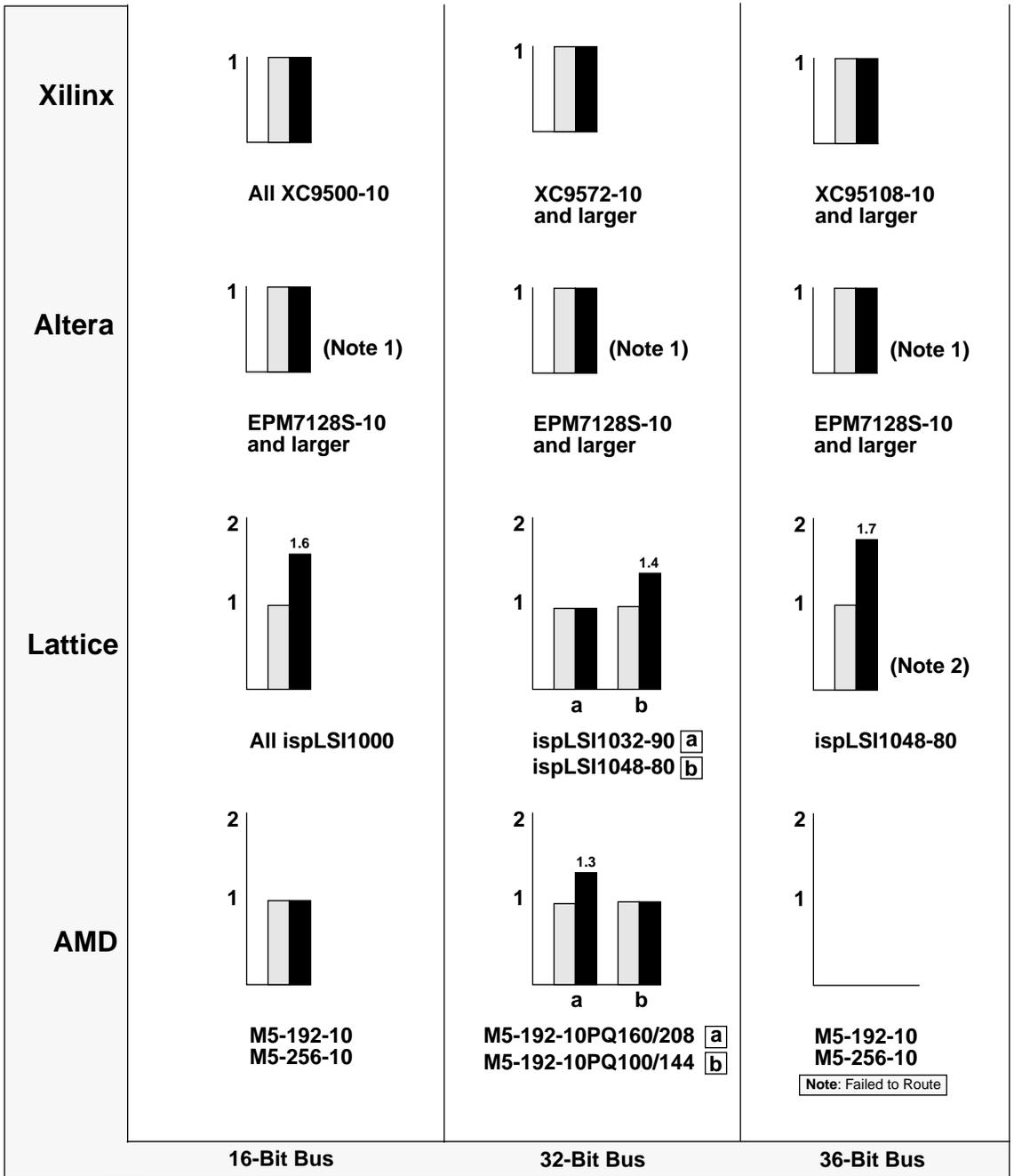
The benchmark results show the superior pin-locking performance of the Xilinx XC9500 family. This performance is consistent across all devices and package types. The wide function block fan-in enables pin-locking of wide, high speed logic functions. And, because feedthroughs are not needed for routing, there is no performance degradation due to routing congestion. This timing consistency is as important as routing ability for maintaining pin-locked designs.

Altera MAX7000, 7000E, and 7000S devices exhibit pin-locking problems due to sparse routing resources. This occurs when many macrocell feedbacks are used and these macrocells drive output pins. The problem is made worse in higher pin count versions of these Altera devices.

The current Altera software does not use logic feedthroughs to resolve routing congestion. Instead, when routing congestion occurs, the design fails to route. This failure can lead to unnecessary PC board re-work to accommodate the design change.

Lattice ispLSI devices suffer from poor routing resources and narrow function block fan-in. The Lattice fitter does use logic resources as feedthroughs in an effort to completely route the design. However, the impact on performance and utilization is significant, even for these very simple designs. In some cases t_{PD} slows as much as 80% and macrocell count increases 25%. The Lattice ispLSI devices employ a poor pin-locking architecture.

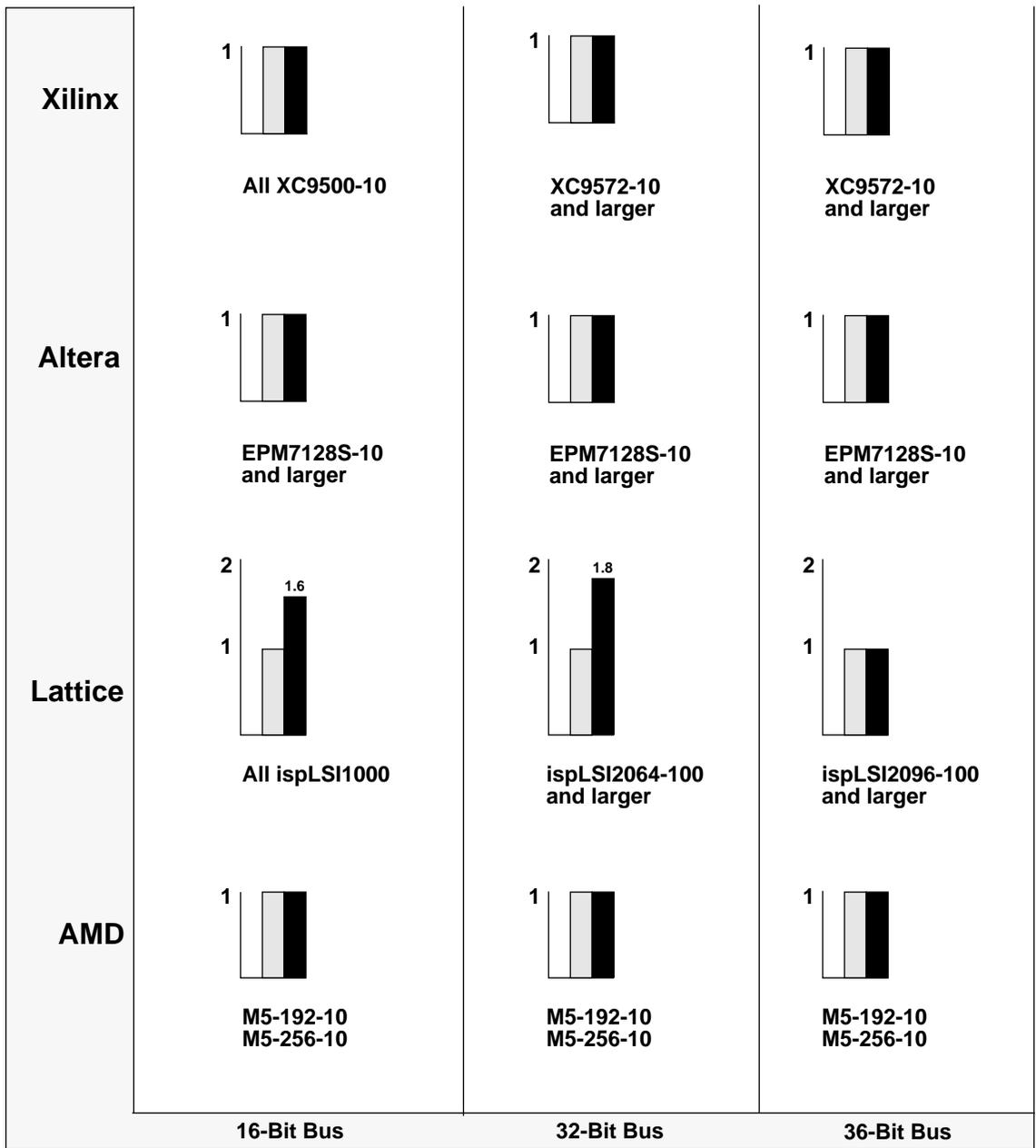
The AMD MACH 5 devices appear to suffer from a combination of inadequate routing resources and poor fitter performance. Narrow functions always re-routed after pin-locking, but with some performance degradation caused by segment delays. However, re-routing of wide functions is the strongest test of the affect of routing resources on pin-locking; in these tests, the AMD MACH 5 failed completely because it could not route the designs, even during the initial design compilation.



t_{PD} Performance — Initial Compile Before Pin-Locking
t_{PD} Performance — After Pin-Locking, with changes

Note 1: Lower density 7KS not avail., or would not generate pinout.
Note 2: Not enough I/O for design, using ispLSI1032.

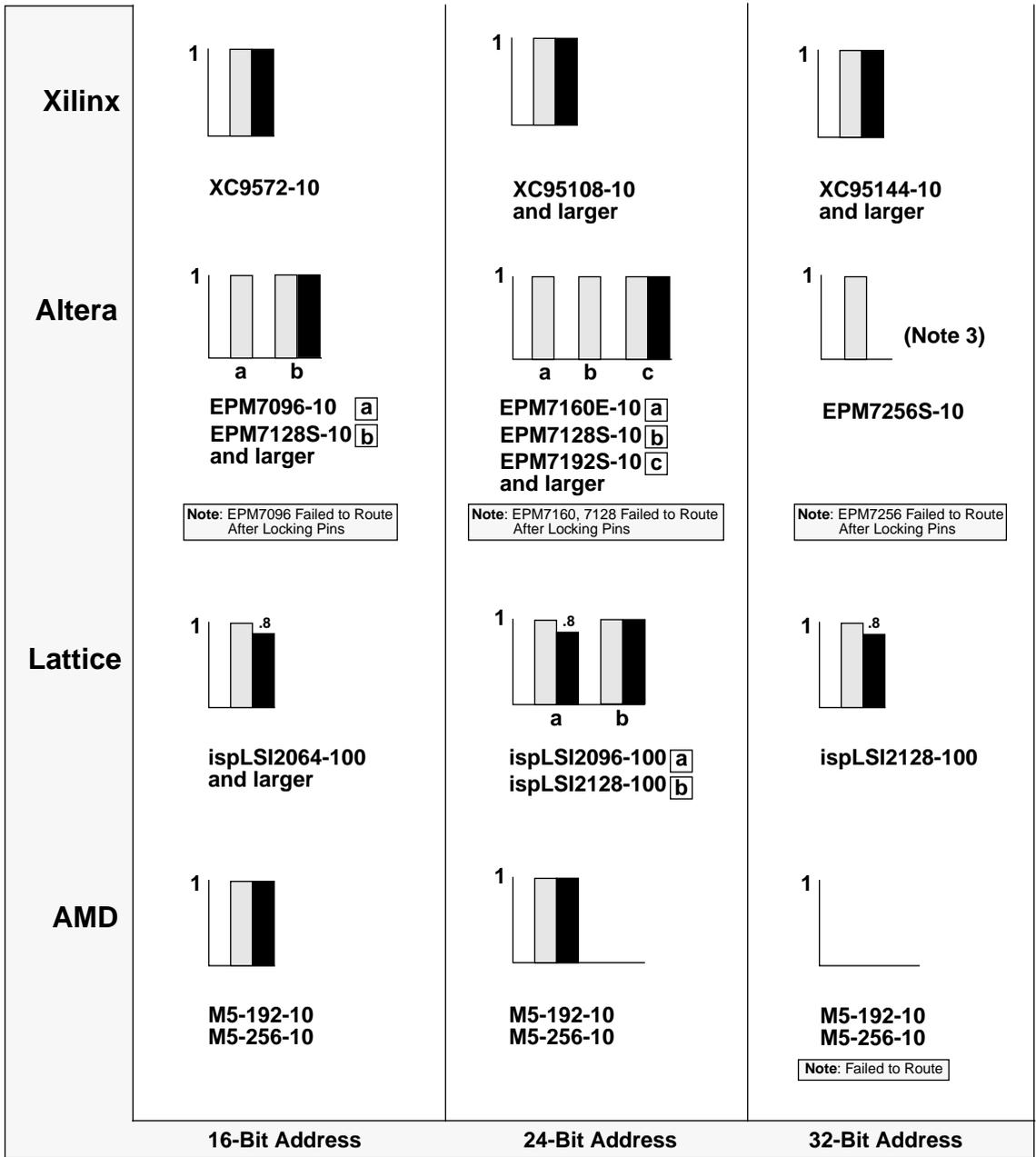
Figure 11: Address Decoder Pin-Locking Performance



2

t_{PD} Performance — Initial Compile Before Pin-Locking
t_{PD} Performance — After Pin-Locking, with changes

Figure 12: Datapath Pin-Locking Performance



f_{max} Performance — Initial Compile Before Pin-Locking 
 f_{max} Performance — After Pin-Locking, with changes 

Note 3: Not enough I/O for design, using EPM7160 and EPM7192.

Figure 13: Address Counter Pin-Locking Performance