Altera's 3.3-V ISP-Capable MAX 7000A Devices

In recent years, an increasing number of engineers have moved their designs to a 3.3-V supply voltage environment. See Figure 1. However, because the programmable logic device (PLD) industry lacked an adequate 3.3-V solution that supported in-system programmability (ISP), engineers who wanted the convenience of ISP were forced to provide separate power supplies for their 3.3-V microprocessors and memories and 5.0-V programmable logic devices (PLDs).
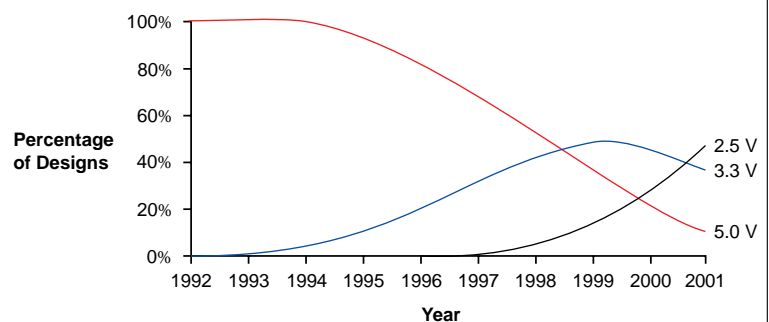
To address this problem, Altera has introduced the first 3.3-V ISP-capable MAX® 7000A device—the EPM7128A device. When using MAX 7000A devices, engineers only need one power supply for all the devices on their 3.3-V printed circuit boards (PCBs). In addition, with the advanced process used to make 3.3-V devices, MAX 7000A devices consume 40% less power than existing MAX 7000 devices in the same package and speed grade. MAX 7000A devices also consume considerably less power than competing devices from field-programmable gate array (FPGA) vendors. See Figure 2 on page 3.

MAX 7000A devices are pin- and function-compatible with MAX 7000S devices and offer fast propagation delays, reduced power consumption, and support for the Jam™ programming and test language. Combined with 5-ns propagation delay times, MAX 7000A devices will clearly provide the fastest solution for your high-volume production design needs.

Benefits of ISP

As time-to-market pressures increase, design engineers continually look for ways to advance the development of system-level products and ensure problem-free manufacturing. The ISP feature available in MAX 7000A devices improves manufacturing efficiency by allowing engineers to program and test their devices in a single step. ISP-capable devices can be easily maintained and upgraded, and allow designers to fix mistakes, add new functions and features, or make changes to accommodate new standard. These features offer tremendous monetary and time savings because system upgrades can be performed in the field. Because ISP allows designers to make changes to devices already soldered onto the PCB, using ISP-capable MAX 7000A devices reduces the cost and quality problems that can arise from bent leads, poor soldering, and electrostatic discharge (ESD).

Figure 1. Designs Moving to 3.3-V Supply Voltage Environment *Note (1)*

Note:
(1) Source: Altera Applications.

Contents

For information about this
newsletter, or to submit questions,
contact:

Printed on recycled paper.

Altera's 3.3-V ISP-Capable
MAX 7000A Devices, continued from page 1

Jam Support Makes ISP Easy

MAX 7000A devices support the Jam programming and test language, a new standard file format for ISP that can program and test electronic systems using the IEEE 1149.1 Joint Test Action Group (JTAG) interface. To achieve the benefits of ISP, the Jam programming and test language addresses the issues designers face when programming PLDs in-system, including proprietary file formats, vendor-specific programming algorithms, large file sizes, and long programming times. Through the use of compression algorithms, short programming pulses, and architecture-independent design, the Jam language provides extremely small file sizes and fast programming times.

MAX 7000A Features

Manufactured on a 0.35-micron, quad-layer-metal EEPROM process, MAX 7000A devices range in density from 32 to 1,024 macrocells and offer 44 to 256 pins. MAX 7000A devices are offered in a variety of packages, including thin quad flat pack (TQFP), plastic quad flat pack (PQFP), ball-grid array (BGA), and plastic J-lead chip carrier (PLCC) packages. MAX 7000A devices offer the same extensive feature set as MAX 7000 devices, including built-in JTAG boundary-scan test (BST) circuitry, programmable slew

Figure 2. MAX 7000A Devices vs. Competing Devices
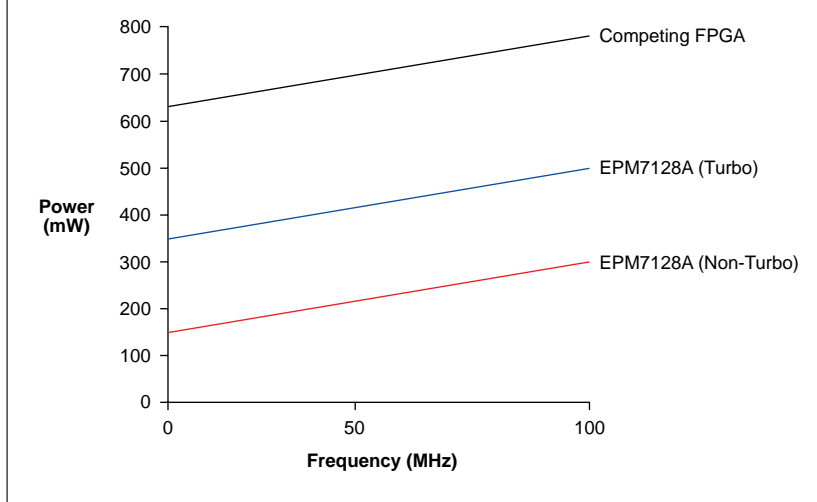
rate control, an open-drain output option, and a programmable power-saving mode. MAX 7000A devices also have advanced pin-locking capability and can be programmed using in-circuit testers, embedded processors, or Altera or third-party programming hardware. In addition, the MultiVolt™ I/O interface allows these devices to interface with 5.0-V, 3.3-V, and 2.5-V devices. Table 1 summarizes MAX 7000A device features.

For more information on MAX 7000A devices, refer to the *MAX 7000A Programmable Logic Family Device Data Sheet* in the *1998 Data Book* or go to the Altera web site at **http://www.altera.com**.

Features	EPM7032A	EPM7064A	EPM7128A	EPM7256A	EPM7384A	EPM7512A	EPM71024A
Usable gates	600	1,250	2,500	5,000	7,500	10,000	20,000
Macrocells	32	64	128	256	384	512	1,024
Maximum user I/O pins	36	68	100	164	212	212	212
t_{PD} (ns)	5	5	5	6	7.5	7.5	7.5
t_{FSU} (ns)	2.5	2.5	2.5	2.5	3	3	3
t_{CO1} (ns)	3.5	3.5	3.5	4	4.5	4.5	4.5
f_{CNT} (ns)	178.6	178.6	178.6	151.5	125	125	125
Packages	44-pin PLCC 44-pin TQFP	44-pin PLCC 44-pin TQFP 84-pin PLCC 100-pin TQFP	84-pin PLCC 100-pin TQFP 144-pin TQFP	100-pin TQFP 144-pin TQFP 208-pin PQFP 256-pin BGA	144-pin TQFP 208-pin PQFP 256-pin BGA	144-pin TQFP 208-pin PQFP 256-pin BGA	208-pin PQFP 256-pin BGA

Table 1. MAX 7000A Device Features

Advin Systems Supports the Jam Programming & Test Language

Advin Systems Inc., a provider of programmable logic device (PLD) programmers, introduced PILOT-JVP, a new programmer that incorporates the Jam™ programming and test language. The PILOT-JVP offers the following features:

- Supports Jam Files (**.jam**) created by the MAX+PLUS II software
- Support devices instantly as they become available from manufacturers
- Programming core software provided by Altera and other PLD manufacturers
- Windows 95 user interface
- True low-voltage support
- Free lifetime software updates via bulletin board service (BBS) and the world-wide web
- Industrial quality

The PILOT-JVP universal programmer contains a Jam Player, which interprets Jam programming files produced by PLD vendors. Because Jam Files contain all programming algorithm instructions, the PILOT-JVP programmer can program devices according to a PLD vendor's exact specification without any chance of error.

The Jam Player software is a built-in feature residing in the PILOT-JVP programmer. A PLD vendor's development tools (e.g., the Altera® MAX+PLUS® II development system) will incorporate all new programming algorithms into the Jam Files automatically. Because new programming algorithms are not required for each new device, the PILOT-JVP programmer can provide instant programming support. Therefore, designers no longer have to implement programming algorithms into programmers.

In addition to full support for Altera PLDs, the PILOT-JVP programmer will support other vendor's PLDs that can be programmed with Jam Files.

For more information on the Jam programming and test language hardware support from Advin Systems or other third-party programmer vendors, go to the Jam web site at **http://www.jamisp.com**.
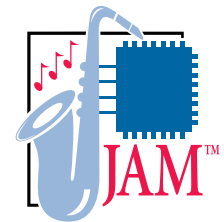
Altera 1998 Data Book Now Available

The Altera® *1998 Data Book* provides comprehensive information about Altera's FLEX® 10K, FLEX 8000, FLEX 6000, MAX 9000, MAX 7000 (including MAX 7000A, which was formerly known as Michelangelo), MAX 5000, Classic™, and Configuration EPROM device families, MAX+PLUS® II development tools, and programming hardware. To request a copy of the *1998 Data Book,* contact Altera Literature Services at (888) 3-ALTERA, or contact your local Altera sales office or sales representative.

For the most up-to-date information about Altera products, go to the Altera web site at **http://www.altera.com**.

FLEX Update

Raphael Update

Altera Announces Revolutionary New Raphael Architecture

Altera's revolutionary new architecture—code-named Raphael—is the industry's first MultiCore™ array programmable logic device (PLD) family. Raphael devices combine look-up table (LUT), product term, and embedded array block (EAB) architectural structures in a single PLD, and they will have more than 15 times the capacity of the largest PLDs on the market today. Altera plans to release the first device based on the Raphael architecture in early 1999. Raphael devices will provide the following features:

■ Fabricated on 0.25-micron, 5-layer-metal process, and moving to 0.18-micron, 6-layer-metal and 0.15-micron, 7-layer-metal processes
■ Densities up to 2 million gates
■ 100-MHz system performance
■ 4-dimensional routing hierarchy
■ Enhanced phase-locked loop (PLL) feature

Raphael devices will combine advanced process technology and architectural innovation to take programmable logic to the next level of density and performance. Figure 1 compares the Raphael architecture to the FLEX® 10K architecture, which is optimized for designs up to 250,000 gates.

Figure 1. Raphael Architecture vs. FLEX 10K Architecture
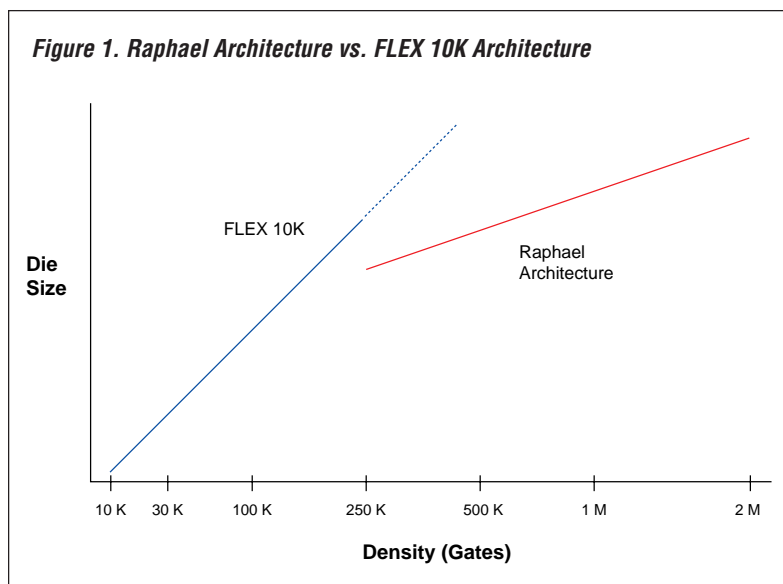
FLEX 10K-1 Shipping

In response to design engineers' increasing need for faster, high-density devices, Altera is now shipping EPF10K50V-1 and EPF10K100A-1 devices. A combination of process improvements and enhancements to the MAX+PLUS® II software have resulted in a 100% performance improvement over the -3 speed grade FLEX 10K devices. These high-density FLEX 10K devices are the fastest devices in the industry.

EPF10K100B Device Coming in Second Quarter 1998

Altera plans to make initial shipments of the EPF10K100B in the second quarter of 1998. Built on a 0.25-μm, 5-layer-metal process, EPF10K100B devices will offer higher performance at a lower cost.

With a 2.5-V core, EPF10K100B devices use less power than existing FLEX 10K devices in the same package and speed grade. Additionally, the MultiVolt™ feature enables the EPF10K100B device to interface with 2.5-V, 3.3-V, or 5.0-V devices.

The EPF10K100B devices will be available initially in 240-pin plastic quad flat pack (PQFP) package followed by the 208-pin PQFP package and space-saving ball-grid array (BGA) packages.

EPF6024A Devices Available

The EPF6024A device—the first 3.3-V member of the FLEX□6000 device family—is now available. As the low-cost alternative to high-volume gate array designs, FLEX 6000 devices are the industry's most cost-effective PLD family. With 24,000 gates, the EPF6024A device extends this low-cost PLD family to higher densities. The EPF6024A device is available now in 208-pin PQFP packages, while devices in 144-pin TQFP, 240-pin PQFP, and 256-pin ball-grid array (BGA) packages are planned. Projected 100,000-unit volume prices start at $9.50 each, which make EPF6024A devices the low-cost gate-array alternative for designs in the 20,000 gate range.

Devices & Tools, continued from page 5

MAX Update

First MAX 9000A Devices Available

With propagation delays as fast as 10 ns, MAX® 9000A devices offer a significant performance enhancement and cost reduction over existing MAX 9000 devices in the same package and speed grade. Production quantities of these high-performance devices are available today. Table 1 summarizes MAX□9000A device availability.

Table 1. MAX 9000A Device Availability							
Device	**t_PD (ns)**	**Availability**	**84-Pin PLCC**	**208-Pin RQFP**	**240-Pin RQFP**	**356-Pin BGA**
EPM9320A	10	Now	✓	✓		✓
EPM9480A	10	Q3 1998		✓	✓	
EPM9560A	10	Now		✓	✓	✓

Industrial-Temperature MAX 7000S Devices Now Available

Industrial-temperature MAX 7000S devices are now available in devices with 64 to 256 macrocells. For example, -7□speed grade EPM7256S devices and -10□speed grade EPM7128S, EPM7192S, and EPM7256S devices are available as industrial-temperature-range devices.

In-Circuit Test Support for MAX 7000S Devices

Altera offers full support for designers using in-circuit testers to program MAX 7000S devices. To use in-circuit test equipment efficiently, designers should use fixed-programming-algorithm devices. The ordering codes for these devices are appended with an "F." For example, -10 speed grade EPM7128S devices in 100-pin TQFP packages intended for use with in-circuit testers have the ordering code EPM7128STC100-10F. If you wish to use in-circuit testers with MAX□7000S devices, contact your local Altera sales representative about ordering fixed-programming-algorithm devices.

Faster MAX 7000S Devices

At any macrocell count, MAX 7000S devices are the fastest product term-based PLDs in the world. MAX□7000S devices now offer even faster performance—as fast as 5 ns. Table 2 describes package options, speed grades, and availability for MAX□7000S devices.

Table 2. MAX 7000S Device Availability			
Device	**Package**	**Speed Grade**	**Availability**
EPM7032S	44-pin PLCC	-6, -7, -10	March 1998
	44-pin TQFP	-6, -7, -10	March 1998
EPM7064S	44-pin PLCC	-5, -6, -7, -10	Now
	44-pin TQFP	-5, -6, -7, -10	Now
	84-pin PLCC	-6, -7, -10	Now
	100-pin TQFP	-6, -7, -10	Now
EPM7128S	84-pin PLCC	-6, -7, -10, -15	Now
	100-pin TQFP	-6, -7, -10, -15	Now
	100-pin PQFP	-6, -7, -10, -15	Now
	160-pin PQFP	-6, -7, -10, -15	Now
EPM7160S	84-pin PLCC	-7, -10, -15	May 1998
	100-pin TQFP	-7, -10, -15	May 1998
	160-pin PQFP	-7, -10, -15	May 1998
EPM7192S	160-pin PQFP	-7, -10, -15	Now
EPM7256S	208-pin RQFP	-7, -10, -15	Now
	208-pin PQFP	-7, -10, -15	Now

MAX 7000 Product Transitions

Altera has migrated existing MAX 7000 devices from a 0.65-micron process to a 0.5-micron process. Table 3 outlines the process migration schedule and lists the advisories and product change notices associated with this migration.

Table 3. MAX 7000 Migration Schedule		Note (1)	
Device	**Reference** *Note (2)*	**Date**	**Process**
EPM7032	PCN9703 ADV9708	August 1998	0.5-micron
EPM7064	PCN9703 ADV9708	Complete	0.5-micron
EPM7064S	PCN9703 ADV9708	Complete	0.5-micron
EPM7128E EPM7128S	PCN9703 ADV9708	Complete	0.5-micron
EPM7160	PCN9703 ADV9708	October 1998	0.5-micron
EPM7192E EPM7192S	ADV9708 ADV9708	April 1998	0.5-micron
EPM7256S EPM7256E	PCN9703 ADV9708	Complete	0.5-micron

Notes:
(1) This process transition will not result in any changes to data sheet parameters or ordering codes.
(2) Go to the Altera web site at **http://www.altera.com** for advisories and process change notices.

MAX+PLUS II Updates

MegaWizard Plug-Ins

MegaWizard™ Plug-Ins provide a tool-independent way to parameterize megafunctions. This new feature is available in MAX+PLUS II version 8.2 and higher or as a stand alone tool for use with third-party EDA design interfaces. MegaWizard Plug-Ins are currently available for the functions from library of parameterized modules (LPM).

For more information on the MegaWizard Plug-Ins , see "Parameterizing Megafunctions with MegaWizard Plug-Ins" on page 12.

Verilog HDL Synthesis Now Available as a Standard Feature

Altera now offers Verilog HDL synthesis as a standard feature in MAX+PLUS II version 8.2 and higher. With this new option, you can choose between VHDL or Verilog HDL text-based HDL design.

These entry-level HDL tools allow you to create a VHDL Design File (**.vhd**) or a Verilog Design File (**.v**), respectively, with the MAX+PLUS II Text Editor or any standard editor and compile it directly with the MAX+PLUS II software.

The MAX+PLUS II software supports most Verilog HDL synthesis constructs. Currently, Verilog HDL synthesis is available for Windows 95 or Windows NT platforms. VHDL or Verilog HDL is available at no additional cost. Contact Altera Applications at (800) 800-EPLD for more information.

New Device Support in MAX+PLUS II Version 8.2

MAX+PLUS II version 8.2 includes support for several new Altera devices. Full compilation, simulation, timing analysis, and programming configuration support is available for newly released devices in the FLEX 10K, FLEX 6000, and MAX 9000 families. Compilation, simulation, and timing analysis support is available for other newly released devices in the FLEX 10KA, FLEX 6000, MAX 9000, and MAX 7000 (including MAX 7000S and MAX 7000A) families. For more information on specific devices supported by MAX+PLUS II version 8.2, go to MAX+PLUS II Help.

Timing-Driven Compilation Improvements

Altera has significantly enhanced timing-driven compilation in MAX+PLUS II version 8.2 by improving critical path estimation and selection, and adding dynamic weight allocation during partitioning. On average, timing-driven compilation in MAX+PLUS II version 8.2 improves design performance by 25%. Compared to a similar design compiled in MAX+PLUS II version 8.1, a 15% improvement in design performance can be achieved in MAX+PLUS II version 8.2.

Timing-driven compilation in MAX+PLUS II version 8.2 also shows consistent results during fitting optimization. Even when timing constraints change, timing-driven compilation consistently tries to optimize fitting to meet performance requirements.

Improvements in design performance, however, are dependent on device utilization. Table 4 shows how to estimate design performance improvements when you are compiling with timing-driven compilation.

Table 1. Estimating Timing-Driven Compilation	
When Device Utilization is…	**Expected Performance Improvement is…**
Less than 50%	30%
More than 50%	15%-20%

When timing-driven compilation is used, the MAX+PLUS II software works harder to fit the design and meet all timing requirements. This increases the average compilation time by a factor of 10.

For more information on timing-driven compilation, go to MAX+PLUS II version 8.2 Help, see *Technical Brief 36 (Timing-Driven Compilation Improvements in MAX+PLUS II Version 8.2)*, or contact Altera Applications at (800) 800-EPLD.

Techniques to Improve FLEX 10K EAB Performance

When implementing designs in FLEX® 10K devices, critical timing paths sometimes drive through an embedded array block (EAB). This article summarizes three techniques that you can use to improve the performance of data paths involving an EAB. These techniques are useful for any design using FLEX□10K EABs.

Place the Source & Destination Paths in a Clique

Placing the source and destination logic elements (LEs) in a clique with the EAB eliminates any possible column delay, because the source LE, destination LE, and EAB are on the same row. This technique works well for memory blocks implemented with one EAB, e.g, a 256×8 memory block. However, for designs that use a larger memory block (requiring multiple EABs), it is better to use the following technique.

Use Multiple EABs

To create a memory block larger than 256×8, you must use multiple EABs. For example, a 256×16 memory block is generated from two EABs. The address and write enable inputs are shared, because the same address is used for both EABs. When using memory functions from the library of parameterized modules (LPM), the MAX+PLUS® II software will automatically split the memory blocks into multiple EABs when necessary. Figure 1 shows how two EABs are combined to make a 256×16 memory block.

Because the two EABs in Figure 1 are located on separate rows, the address and write enable signals must transmit through a column interconnect to drive into one of the EABs. If the critical path in the design is either the address or

write-enable signal to the EAB, the column interconnect delay can slow the design's performance. To avoid this situation, you can speed up the design by manually splitting the memory block into EAB-sized blocks (256×8) and duplicate the address and write enable logic. Then, place the EABs and the appropriate write enable, address, and data logic in a clique. By duplicating and placing the appropriate logic in a clique, the design will not need to use the column interconnect when driving into the EAB.

Figure 1. Creating a 256×16 Memory Block from Two 256×8 EABs
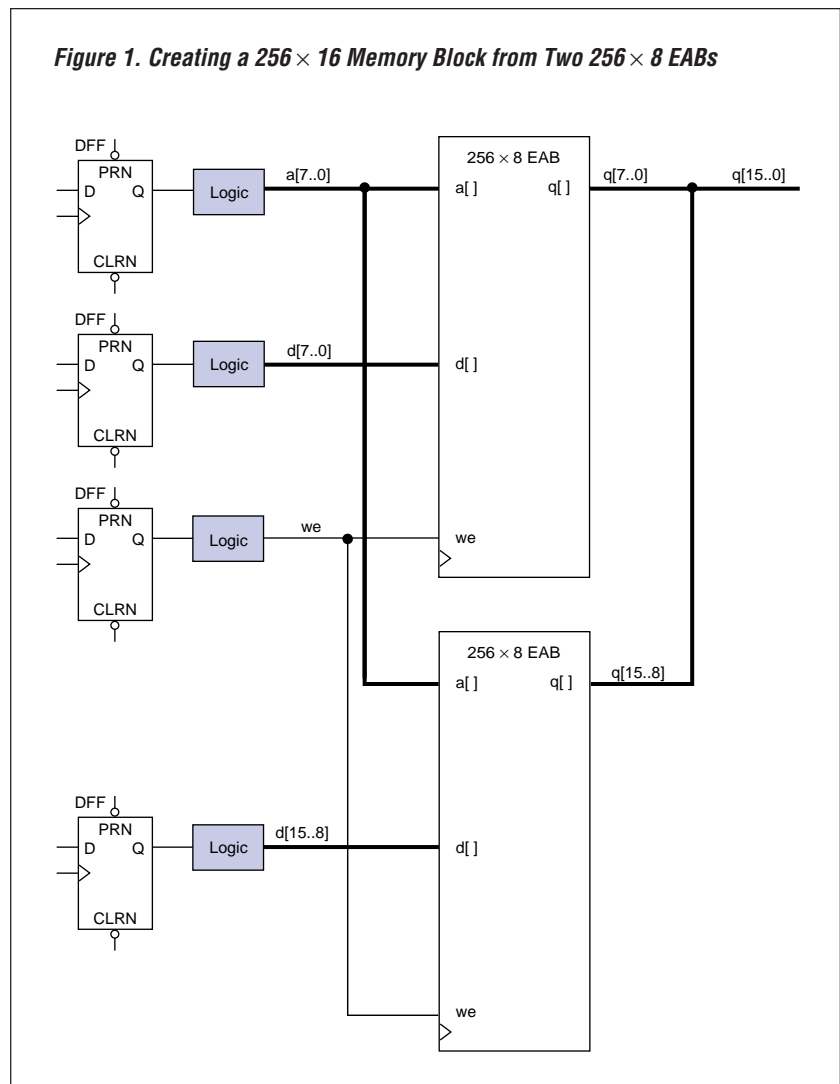
Because each bit of the data logic drives only one of the EABs (see Figure 1), you do not need to duplicate the data logic. Using cliques is enough to ensure that the signal does not need to use the column interconnect to drive into the EAB. This duplication technique is useful for any memory block that uses more than one EAB. Figure 2 shows an example of duplicating the appropriate logic to create a 256 × 16 memory block.

When duplicating logic, make sure that the registers driving the duplicated logic never violate the setup or hold times on the duplicate registers. If the setup or hold times are violated on one but not the other register, the two registers will have different values, resulting in incorrect system behavior. When the logic driving the duplicated register is synchronous, the system will not violate the setup and hold times, unless the system exceeds the maximum operating frequency given by the MAX+PLUS II Timing Analyzer.

When using this technique, it is important to consider how the MAX+PLUS II software will respond to duplicated logic. Because the MAX+PLUS II Compiler tries to minimize the size of the design, it will detect duplicate logic and remove it. However, you can

prevent the software from removing the duplicated logic by using one of three methods:

- To prevent the MAX+PLUS II Compiler from removing all duplicate logic in the design, choose **Global Project Logic Synthesis** (Assign menu), choose **Define Synthesis Style**, choose **Advanced Options**, and turn off *Register Optimization*. Choose **OK** twice to save your changes.

- To prevent the MAX+PLUS II Compiler from removing the duplicate logic on a register-by-register basis, you can turn off the *Register Optimization* option for each register. After selecting the registers, select **Logic Options** (Assign menu), choose **Individual Logic Options**, choose **Advanced Options**, and turn off *Register Optimization.* Choose **OK** twice to save your changes.

- To prevent the MAX+PLUS II Compiler from removing the duplicated logic from the subdesign, select **Hierarchy Display** (MAX+PLUS II menu), select the subdesign, select **Logic Options** (Assign menu), choose **Define Synthesis Style**, choose

Figure 2. Duplicating Address & Write Enable Logic
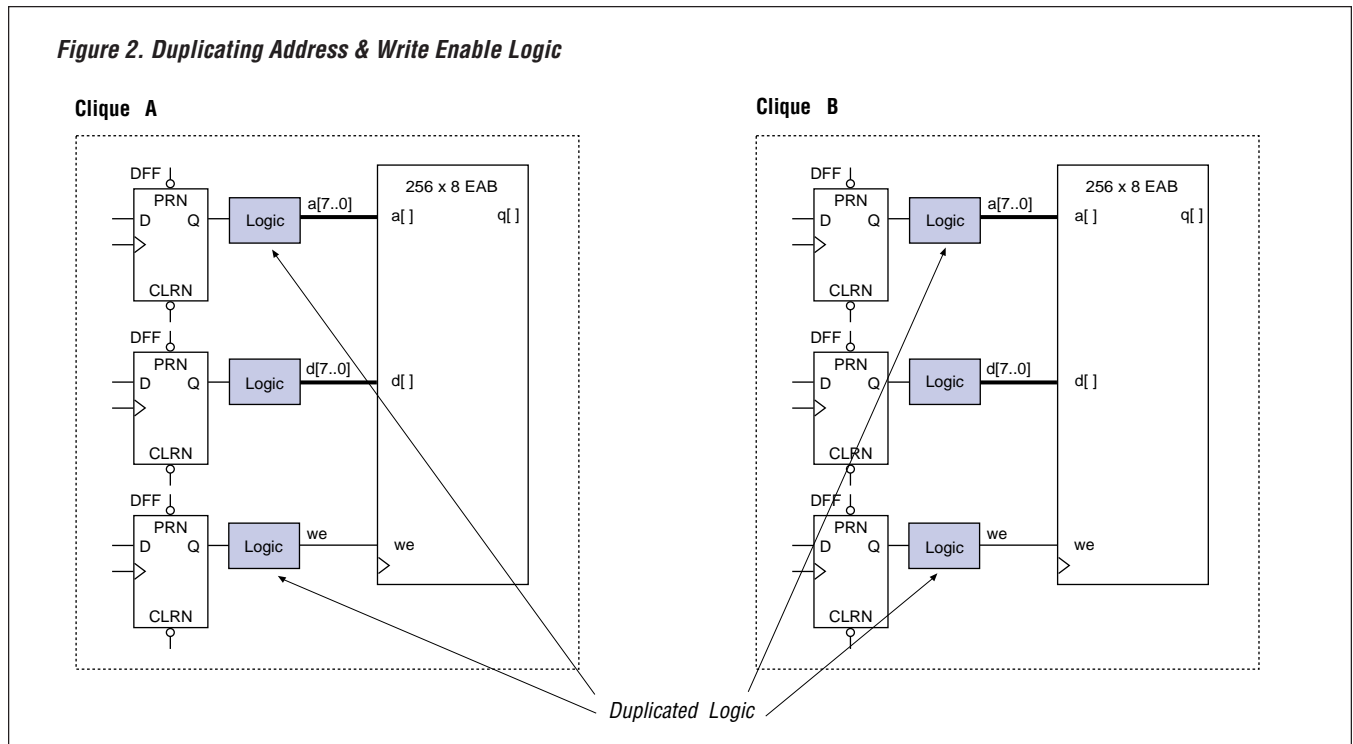
Techniques to Improve FLEX 10K EAB Performance, continued from page 9

Advanced Options, and turn off *Register Optimization.* Choose **OK** twice to save your changes.

By turning off the *Register Optimization* option off on a register-by-register basis, you preserve the necessary duplicate registers while allowing the Compiler to remove any unnecessary registers.

Use the Output Registers in the EAB

Each EAB has a bank of optional registers on the output port of the EAB. You can use these registers to add a pipeline stage to the design, increasing system performance at the cost of one latency cycle. For many applications, e.g., digital signal processing (DSP), the increased system performance outweighs the latency cycle cost.

To demonstrate this technique, consider a relatively simple design in which both the address and write enable ports of a 256×16 memory block are controlled by one of two registers. A multiplexer determines which register controls the address and write enable ports. The output of the memory block is then multiplexed with other logic and drives a register. See Figure 3.

Table 1 shows the registered performance obtained using varying levels of pipelining and duplication when compiling an EPF10K50V-1 device with the sample design in Figure 3. All performance figures were obtained using MAX+PLUS II version 8.2.

For more information about techniques to improve FLEX 10K EAB performance, call Altera Applications at (800) 800-EPLD.

Figure 3. Example Design

Table 1. Registered Performance Results
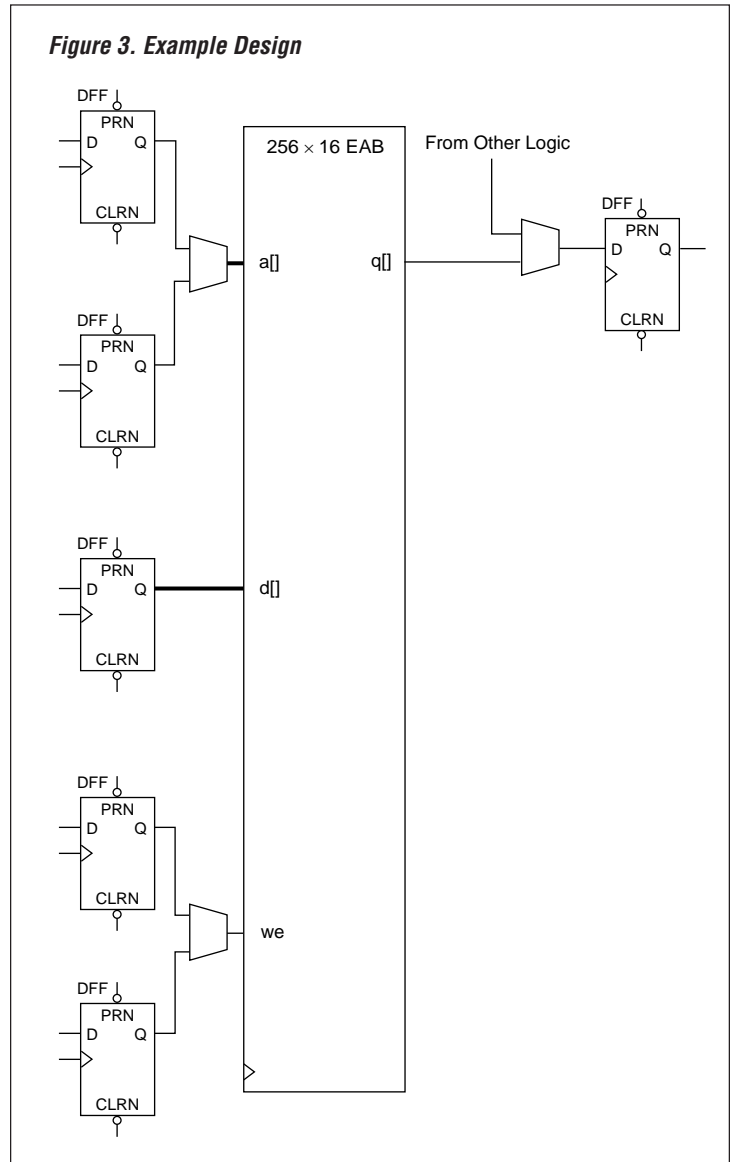
Optimization	f_{MAX} (MHz)	Critical Path Source	Critical Path Route	Critical Path Destination
None	79	Input registers of EAB	Through EAB memory and the output multiplexer	Output register
Pipeline with output registers	116	WE register	Through the multiplexer and column interconnect	EAB WE register
Pipeline with output registers and duplicate control logic	120	WE register	Through the multiplexer	EAB WE register

New Data Communication Megafunctions Available

AMPP SM
ALTERA MEGAFUNCTION PARTNERS PROGRAM

To speed up your data-communications designs, two megafunctions are available from Altera Megafunction Partners Program (AMPP℠) partners. These megafunctions, which are targeted for specific Altera device architectures, permit you to focus more time on improving your overall design. Both megafunctions are optimized by the AMPP partner and verified by Altera.

High-Level Data Link Controller (HDLC)

Integrated Silicon Systems (ISS) is now offering a High-Level Data Link Controller (HDLC) megafunction that creates data packets for point-to-point communications. Compared to a standard digital signal processing (DSP) approach, implementing HDLC in Altera® FLEX® devices provides a faster data rate as well as shorter time-to-market. This HDLC megafunction is a building block for X.25, frame-relay, and ISDN (both B and D channels) implementations. The function features:

- Flag insertion
- Multi-channel operation
- Either zero-insertion transparency for serial data stream output, or octet-insertion transparency for byte parallel data stream output

The single channel mode offers 50-MHz operation and supports data rates of up to 400 Mbps, while consuming only 296 logic elements (LEs) of an EPF10K30 device. ISS is currently optimizing this function for Altera's FLEX 6000 device family. For pricing and customization information, contact ISS directly at:

Integrated Silicon Systems, Ltd
50 Malone Road
Belfast, BT9 5BS
Northern Ireland
Tel. (44) 1232-664-664
Fax (44) 1232-669-664
http://www.ISS-DSP.com

Speed-Matching Asynchronous FIFO

The Speedbridge megafunction, from SIS Microelectronics, Inc., offers a speed-matching first-in first-out (FIFO) buffer that can reliably transfer data across an asynchronous interface. An asynchronous FIFO function like the Speedbridge megafunction can help minimize overall system development and debugging time. This megafunction, which is designed for high-speed data transfer, is a key module of 10/100 Ethernet MAC, as well as many high-speed networking protocol interfaces. The read and write ports of the Speedbridge have independent, continuously running clocks and synchronous enables for accessing their respective functions.

SIS Microelectronics will customize the FIFO width and depth according to design requirements; one very efficient implementation is to use a shallower Speedbridge megafunction for asynchronous communication together with a parameterized FIFO function (e.g., the Altera `csfifo` function) for data buffering.

Solution Brief 13 (Speedbridge Megafunction) provides additional performance information on the Speedbridge megafunction for the Altera FLEX□10K and FLEX 8000 device families. This document is available on Altera's web site at **http://www.altera.com**.

SIS Microelectronics also plans to offer a parameterized version of this function, which is user-controlled and enabled by the Altera MegaWizard™ Plug-In Manager. For more information on the Altera MegaWizard Plug-In Manager, refer to "Parameterizing Megafunctions with MegaWizard Plug-Ins" on page□12.

For pricing and customization information, contact SIS Microelectronics directly:

SIS Microelectronics, Inc.
1831 LeftHand Circle Suite E
P.O. Box 1432
Longmont, CO 80501
Tel. (303) 776-1667
Fax (303) 776-5947

Parameterizing Megafunctions with MegaWizard Plug-Ins

Altera now provides MegaWizard™ Plug-Ins, which parameterize megafunctions without requiring the designer having to use a third-party tool. You can use this new feature in MAX+PLUS® II version 8.2 or as a stand-alone tool with third-party EDA design interfaces. MegaWizard Plug-Ins provide maximum flexibility and allow you to customize megafunctions without changing your design's source code. Unlike existing megafunction-parameterizing techniques that force you into a proprietary design flow, MegaWizard Plug-Ins allow you to integrate a parameterized megafunction in *any* hardware description language (HDL) or netlist file using *any* EDA tool. Figure 1 shows the design flow for using MegaWizard Plug-Ins with parameterized megafunctions.

Figure 1. Design Flow for Using MegaWizard Plug-Ins
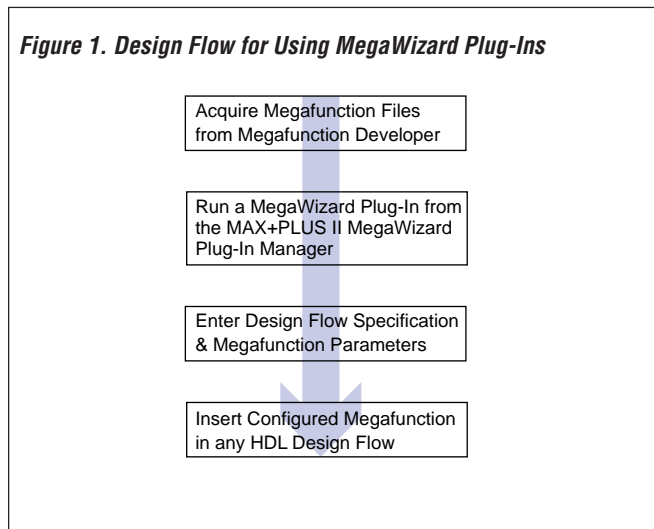
MAX+PLUS II Design Support

MegaWizard Plug-Ins are currently available for library of parameterizable modules (LPM) functions, which are industry-standard sets of configurable functions including arithmetic, memory, counter, and multiplexer functions. LPM functions are supported by Altera and major EDA tool vendors such as Cadence, Exemplar Logic, Mentor Graphics, Synopsys, Synplicity, and Viewlogic.

In addition, MegaWizard Plug-Ins are being developed for Altera® MegaCore™ functions, including fast Fourier transform (FFT), digital signal processing (DSP), and peripheral component interconnect (PCI)

megafunctions. The Altera Megafunction Partners Program (AMPP) vendors are also parameterizing their megafunctions with MegaWizard Plug-Ins. For information on which AMPP megafunctions are parameterized with MegaWizard Plug-Ins, go to the Altera web site at **http://www.altera.com**.

Using the MegaWizard Plug-In

To create a parameterized megafunction using MegaWizard Plug-Ins in MAX+PLUS II version 8.2, follow these steps:

1. Start the MegaWizard Plug-In Manager by choosing either the:

    ■   **Megafunction Plug-In Manager** command (File menu) in any MAX+PLUS II software application.
    ■   **MegaWizard Plug-In Manager** button in the **Enter Symbol** dialog box (Symbol menu) in the MAX+PLUS II Graphic Editor.

2. In the **MegaWizard Plug-In Manager** window, specify that you wish to edit or create a new custom megafunction variation. A custom megafunction variation is a design file based on Altera- or AMPP-provided megafunctions, including LPM functions. Choose **Next**. See Figure 2.

Figure 2. MegaWizard Plug-In Manager Window
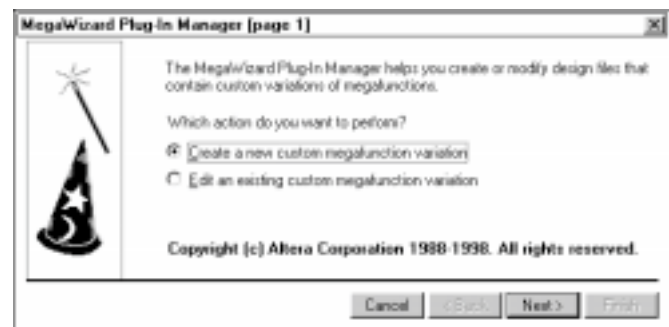
3. The MegaWizard Plug-In Manager will guide you through a series of windows. In these windows, you should specify the HDL you want to use (i.e., VHDL, Verilog HDL, or Altera Hardware Description Language (AHDL)), add your desired parameters, and select boundary conditions such as I/O pins. Figure 3 shows an example.

Figure 3. Selecting a Megafunction for Customization

4. Choose **Finish** to save your selections. The MegaWizard Plug-Ins create a new custom megafunction variation, which can be instantiated with the rest of a design.

You can now instantiate the custom megafunction variation in a design file. For more information on MegaWizard Plug-Ins, contact Altera Applications at (800) 800-EPLD or go to MAX+PLUS II version 8.2 Help.

Now Available: New Versions of Altera Digital Library & In-System Programmability CD-ROMs

Altera has recently updated the *Altera Digital Library CD-ROM* and *In-System Programmability CD-ROM*, which are now available from Altera Literature Services.

The *Altera Digital Library CD-ROM* version 4 contains all current technical literature, including:

■ Information on all Altera® device families

■ All current data sheets and application notes

■ All current megafunction information, including the *AMPP Catalog*, solution briefs, and detailed technical information on the Microperipheral MegaCore™ Function Library

■ All literature on Altera's support for in-system programmability (ISP) and the Jam programming and test language

■ Information on target applications, including digital signal processing (DSP), bus interface, broadband communications, and networking applications

The *In-System Programmability CD-ROM* version 4 contains all current technical literature on the ISP feature available in Altera MAX® 9000 (including MAX 9000A), MAX 7000S, and MAX 7000A devices. The CD-ROM also includes information about the Jam Programming & Test Language.

To receive a copy of either CD-ROM, contact Altera Literature Services at (888) 3-ALTERA or send e-mail to lit_req@altera.com.

For the most up-to-date information about Altera products and services, go to the Altera web site at **http://www.altera.com**. For more information about the Jam programming and test language, go to the Jam web site at **http://www.jamisp.com**.

FLEX Devices: A Process Generation Ahead

As programmable logic architectures migrate to more advanced process technologies, the inherent benefits of utilizing a continuous interconnect over segmented interconnect become more apparent. By examining the implementation of digital circuitry in silicon, this article explains how programmable logic devices (PLDs) using continuous routing structures benefit from improvements in process technology.

To implement a digital circuit in silicon, two main process steps are required: diffusion and metalization. Diffusion is used to place transistors in silicon, while metalization is used to connect the transistors to form a circuit. To fully benefit from the lower cost and higher performance possible with process improvements, the device architecture must be adaptable to the effects of die shrinkage, i.e., smaller transistors and multi-layered metal.

The continuous interconnect of FLEX® devices is fully adaptable to the effects of die shrinkage, and therefore offers maximum die-size reduction. For example,

because the dedicated interconnect consists of device-wide metal lines that enable signals to traverse across a row or column of a FLEX device without using transistors (see Figure 1), these metal lines can be efficiently mapped to multiple metalization layers. This efficient use of multi-layer metal results in maximum die size reduction and the lowest possible cost.

In contrast, the segmented interconnect of field-programmable gate arrays (FPGAs) consists of short metal segments that are connected with programmable switch matrixes (PSMs). A PSM consists of programmable interconnect points (PIPs). A typical PSM contains 8 PIPs. Each PIP contains 6 pass transistors and 6 SRAM bits, which control the pass transistors. Each SRAM bit uses 5 transistors. Therefore, each PIP requires 36 transistors. See Figure 2. The segmented architecture cannot achieve the maximum die area reduction from process improvements because transistors must be diffused into silicon and cannot efficiently use additional layers of metal. Thus, for a given process technology, a

Figure 1. Dedicated Interconnect Consisting of Device-Wide Metal Lines

Metal Interconnect (Row)

Metal Interconnect (Column)

LAB 1

LAB 2

LAB 3

LAB 4

LAB 5

LAB 6

LAB 7

LAB 8

No Transistors within Interconnect Channel
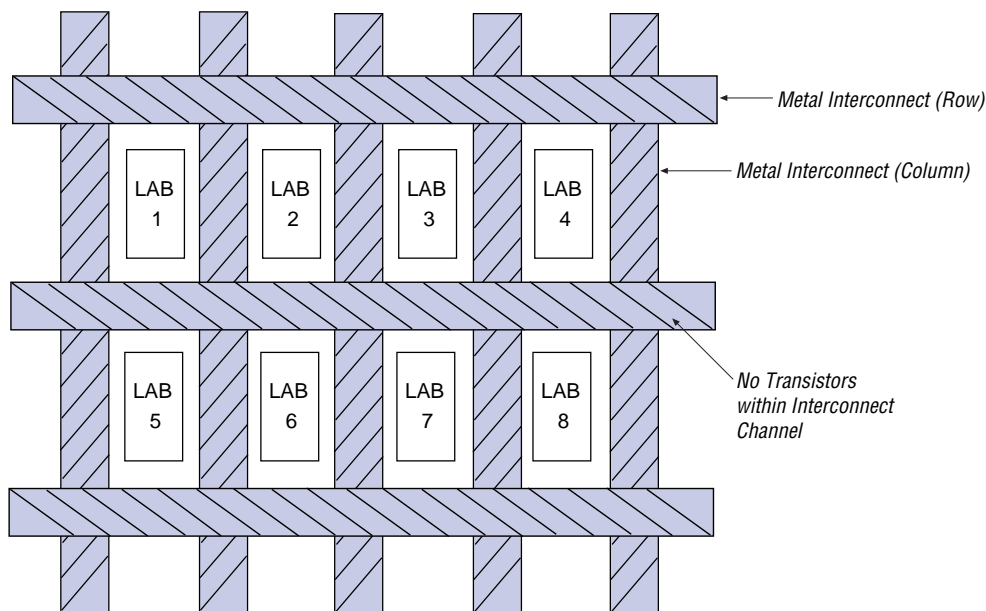
Figure 2. PSM Used to Connect Horizontal and Vertical Segments of the Interconnect
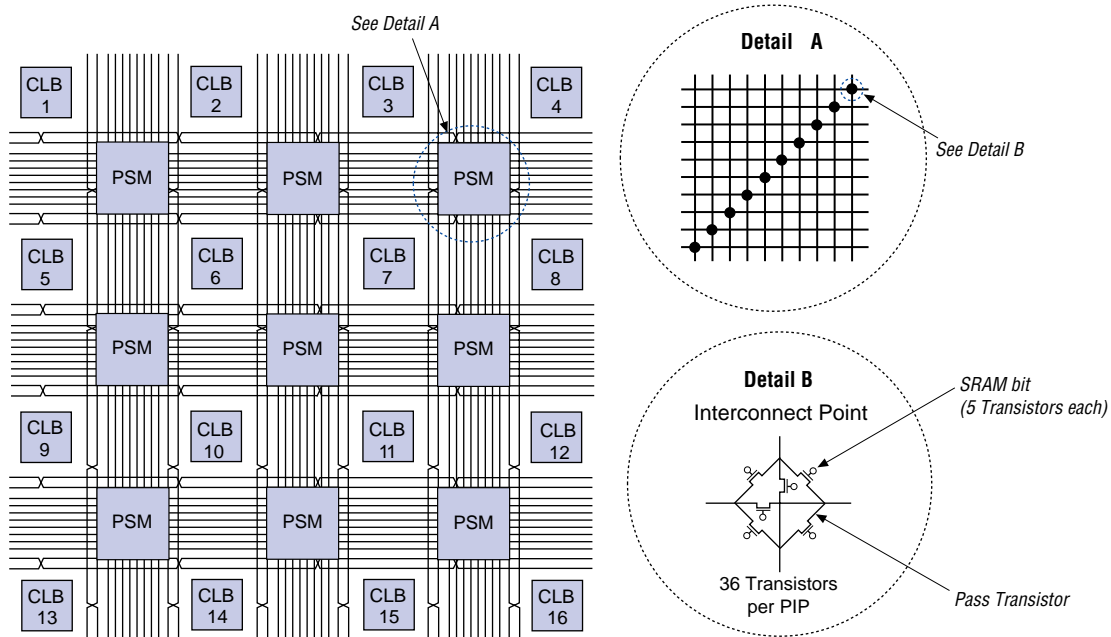
continuous interconnect structure will produce a smaller die size and lower cost than a segmented interconnect structure.

The fundamental difference between continuous and segmented interconnects becomes apparent when comparing actual die sizes. For example, the die size of the 0.35-micron competing FPGA device is almost twice as large as the die size of the 0.35-micron Altera EPF10K100A device—even though the two devices have comparable resources. In fact, the 0.35-micron competing FPGA device is closer in die size to the

0.5-micron EPF10K100 device than to the 0.35-micron EPF10K100A device. Table 1 provides die size comparisons between a 0.35-micron competing FPGA device and two process generations of the EPF10K100 device.

With each new process generation, the metal-intensive continuous interconnect of FLEX devices takes full advantage of multiple metal layers for maximum die-size reduction. However, die size reduction of competing FPGA devices is limited by the burden of transistors in the segmented interconnect path.

Table 1. Die Comparison of EPF10K100 Devices vs. Xilinx XC4062XL Device

Device	Process Geometry (Micron)	Density	Normalized Die Size *Note (1)*
XC4062XL	0.35	4,608 equivalent LEs, no EABs	1.91
EPF10K100GC503-4	0.5	4,992 LEs plus 24,576 EAB bits	2.38
EPF10K100ARC240-3	0.35	4,992 LEs plus 24,576 EAB bits	1.00

Note:
(1) Source: Altera Applications.

Configuring FLEX 10KA & FLEX 6000A Devices

Sections of this article appeared in the News & Views Fourth Quarter 1997 issue under the title "Configuring FLEX 10KA Devices" and contained inaccurate information. The corrected information as well as additional information about configuring FLEX 6000A devices appears in the article below.

Designers can configure FLEX® 10KA and FLEX 6000A devices using passive serial (PS) or JTAG configuration mode with support from the BitBlaster™ serial or the ByteBlaster™ parallel port download cable. The download cables support passive serial (PS) or JTAG configuration modes by channeling configuration data to FLEX 10KA or FLEX 6000A devices on system circuit boards that have a 5.0-V power plane. By downloading design changes directly to a device, you can easily prototype the devices and accomplish multiple design iterations in quick succession. FLEX 10KA (including EPF10K50V and EPF10K130V devices) and FLEX 6000A devices can operate with 5.0-V systems, allowing you to configure them with the 5.0-V BitBlaster or ByteBlaster download cable.

This article discusses PS and JTAG configuration of FLEX 10KA and FLEX 6000A devices with the BitBlaster or ByteBlaster download cable, and how a system circuit board should be wired to support these device configurations.

PS or JTAG Configuration

Before configuring a FLEX 10KA or FLEX 6000A device with a BitBlaster or ByteBlaster download cable, the system circuit board should be properly wired to support PS or JTAG configuration. The BitBlaster and ByteBlaster download cables have 5.0-V TTL input and output pins and drive out 5.0-V TTL-compatible signals with a required high-level output voltage (V_{OH}) of 2.4 V. FLEX 10KA and FLEX 6000A devices have MultiVolt™ I/O pins that accept 5.0-V inputs and drive output voltage levels compatible with a TTL V_{OH} of 2.4 V. Before configuring a FLEX 10KA or FLEX 6000A device with the BitBlaster or ByteBlaster download cable, you should connect their V_{CCINT} and V_{CCIO} pins to the 3.3-V power plane. However, you should not connect the V_{CCIO} pin to the 2.5-V power plane, because 2.5-V output pins cannot drive 5.0-V-TTL logic levels, which are required by the BitBlaster or ByteBlaster download cable.

PS Configuration

FLEX 10KA and FLEX 6000A devices can be configured using a PS configuration scheme. Any pin can be pulled up to the 3.3-V or 5.0-V supply. See Figures 1 and 2, respectively.

Figure 1. PS Configuration of FLEX 10KA Devices Using the ByteBlaster Download Cable
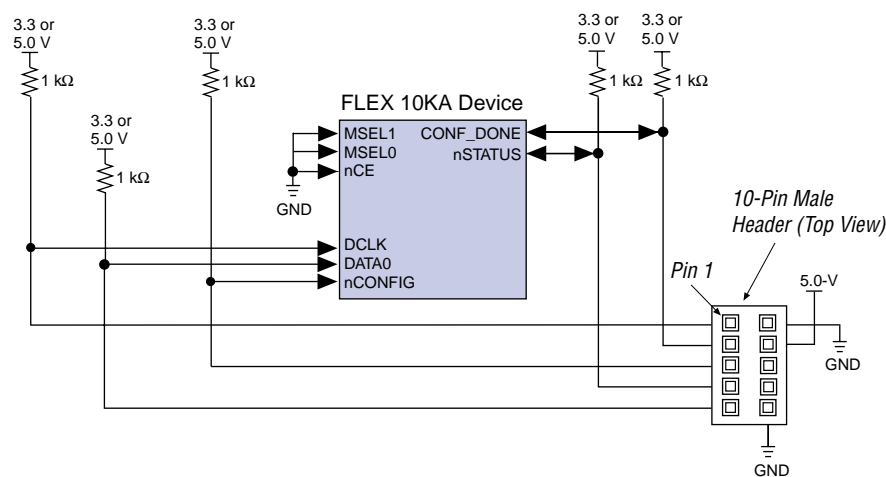
IEEE 1149.1 (JTAG) Configuration

FLEX 10KA devices can be configured using the IEEE 1149.1 (JTAG) interface. Any pin can be pulled up to the 3.3-V or 5.0-V supply. See Figure 3.

For more information on FLEX 10KA devices and FLEX 6000A devices, see the *FLEX 10K Embedded*

Programmable Logic Family Data Sheet and the *FLEX 6000 Programmable Logic Device Family Data Sheet*, respectively, in the **1998 Data Book**. For more information on the BitBlaster or ByteBlaster download cables, see the *BitBlaster Serial Download Cable Data Sheet* and the *ByteBlaster Parallel Port Download Cable Data Sheet*, respectively, in the **1998 Data Book**.

Figure 2. PS Configuration of FLEX 6000A Devices Using the ByteBlaster Download Cable
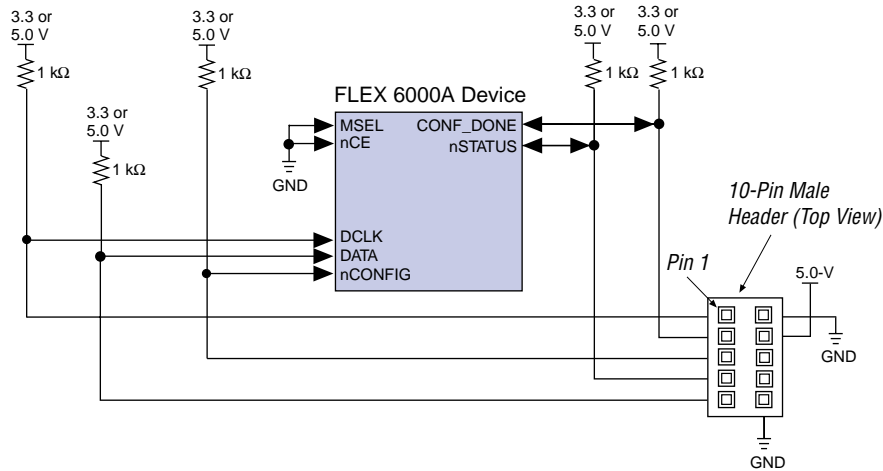
Figure 3. FLEX 10KA JTAG Configuration Using a ByteBlaster Download Cable
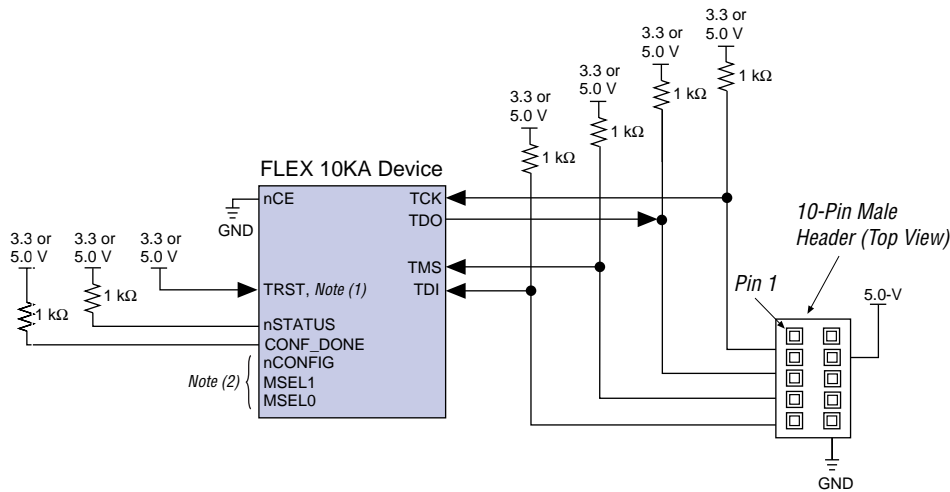
Notes:
(1) Because FLEX 10KA devices in 144-pin TQFP packages do not have TRST pin, you can ignore this connection when using these devices.
(2) The nCONFIG, MSEL0, and MSEL1 pins should be connected to support a FLEX configuration scheme. If only the JTAG configuration is used, connect nCONFIG to V$_{CC}$ (either 3.3 V or 5.0 V), and MSEL0 and MSEL1 to ground.

Questions
& A N S W E R S

Q *Which version of the MAX+PLUS® II software supports -5 speed grade EPM7064S devices?*

A -5 speed grade EPM7064S devices are fully supported in MAX+PLUS II version 8.2. To design for this device with MAX+PLUS II version 8.1, you must also install the MAX+PLUS II version 8.14 update, which is available via the Altera web site at **http://www.altera.com**.

Q *Will my device work properly when operating at the absolute maximum junction temperature?*

A Absolute maximum ratings are specified for reliability purposes, not functionality. The device is not guaranteed to function properly at the absolute maximum junction temperature. The functional operation of Altera devices is not specified at the absolute maximum ratings, but is guaranteed within the range defined by a device's recommended operating conditions.

For more information on absolute maximum ratings, refer to the *Operating Requirements for Altera Devices Data Sheet*, available in the *1998 Data Book* or from the Altera web site at **http://www.altera.com**.

Q *How do I generate a pin-out and Programmer Object File (.pof) for EPF6024AQC208 devices in MAX+PLUS II version 8.2?*

A To generate a pin-out and POF for EPF6024AQC208 devices in MAX+PLUS II version 8.2, you must enter a password in the MAX+PLUS II Programmer by performing the following steps:

1. Choose **Device** (Assign menu).

2. In the **Device** dialog box, select *FLEX 6000* in the *Device Family* drop-down list box. Choose **OK**.

3. Choose **Select Device** (Options menu) in the MAX+PLUS II Programmer and select *FLEX 6000* in the *Device Family* drop-down list box.

4. Select *EPF6024AQC208* (any speed grade) in the *Available Devices* drop-down list box.

5. Choose **Enable**.

7. Enter the password 4R0EYZ (with a zero, not the letter O) in the *Password* box.

8. Choose **Add**. The password should appear in the *Existing Passwords* box.

9. Choose **OK** to save your changes.

The MAX+PLUS II software now fully supports EPF6024AQC208 devices.

Q *Is it OK to toggle the* DATA, DATA0, *and* DCLK *signals once a FLEX® 10K or FLEX 6000 device has entered user mode?*

A DATA, DATA0, and DCLK are dedicated inputs that are only used during configuration. Once configuration is complete and the device has entered user mode, these pins can be toggled without affecting the configuration data.

Q *How do I use the* LPM_HINT *parameter to use multiple Altera-specific parameters in a VHDL design?*

A In a VHDL design, you can use multiple parameters by using the LPM_HINT parameter. In a Generic Map Statement, use a comma to separate the parameters, which must be enclosed in quotation marks. For example:

```
inst_1: lpm_mult
   GENERIC MAP (
     lpm_widtha => 8,
     lpm_widthb => 8,
       lpm_widthp => 16,
       lpm_widths => 1,
       lpm_hint => "USE_EAB= ON,
         INPUT_A_IS_CONSTANT= YES")
.
.
.
```

For more information on the LPM_HINT parameter, go to MAX+PLUS II Help.

Q *Why does the Compiler generate a "No fit found" message for my FLEX 10K project when I try to feed a global signal with any logic element (LE) except the first one in a logic array block (LAB)?*

A In FLEX 10K devices, a subset of the LEs can drive global signals; check the data sheet for details. If you have no logic cell assignments, the MAX+PLUS II

software automatically places logic driving a global signal in an appropriate position (i.e., the first LE in an LAB). In this case, making an illegal assignment causes the MAX+PLUS II Compiler to generate a "no fit" message.

Q *What is the required rise time for* nCONFIG?

A When nCONFIG is driven by a signal, the input signal rise time must meet the 40-ns input rise time specification. When the device's V_{CC} reaches steady state, the nCONFIG signal rise time should not exceed this specification.

When nCONFIG is tied to V_{CC}, the nCONFIG rise time must meet the 100 ms V_{CC} rise time requirement. The slower rise time is allowed because nCONFIG sees a V_{IH} at a lower voltage level than is required for the device to exit Power On Reset (POR). Therefore, when the device exits POR, it will recognize nCONFIG as a logic high.

Q *How many times can you reprogram a windowed EPROM-based device?*

A Altera windowed EPROM-based devices can be programmed and erased at least 25 times.

For more information on windowed EPROM-based devices, refer to the *Operating Requirements for Altera Devices Data Sheet*, available in the *1998 Data Book* or from the Altera web site at **http://www.altera.com**.

Q *Why does boundary-scan testing fail immediately after I power up my FLEX devices?*

A As stated in *Application Note 39 (Boundary-Scan Testing in Altera Devices)*, boundary-scan testing for FLEX devices is supported before and after, but not during configuration. Therefore, if you are testing before configuration, you must hold nCONFIG low to prevent configuration from starting.

Q *Do I need a new authorization code to run the MAX+PLUS II software when I upgrade my operating system (e.g., from Windows 95 to Windows NT)?*

A Depending on the type of MAX+PLUS II software guard you have, you may need to obtain a new authorization code when upgrading operating systems. To identify the software guard type, choose **Authorization Code** (Options menu) in the MAX+PLUS II software. The software guard

identification number is displayed in the *Method* box in the **Authorization Code** dialog box. If your software guard begins with a "T," you do not need to obtain a new authorization code. However, if your guard number begins with a "D," "E," or "S," you must obtain a new authorization code. For "D" guards, contact Altera Customer Service at (800) SOS-EPLD. For "E" or "S" guards, go to the Altera web site at **http://www.altera.com**.

Q *Can I program a MAX 9000A device with a MAX 9000 Programmer Object File (.pof)?*

A Yes. However, MAX 9000A devices offer some enhanced features that are not available in MAX 9000 devices. When a MAX 9000 POF is used to program a MAX 9000A device, these enhanced features are disabled automatically.

Q *When will MAX+PLUS II support for Windows 3.1 and Windows 3.11 end?*

A MAX+PLUS II version 8.3 for PCs and higher will support Windows 95 and Windows NT operating systems. The Altera Stand-Alone Programmer (ASAP2) for device programming will be supported on Windows 3.1 and Windows 3.11 until the end of 1998.

Q *How can I include a VHDL design in my Altera Hardware Description Language (AHDL) design?*

A To include a VHDL design in an AHDL design, perform the following steps:

1. Compile your VHDL design.

2. Choose **Create Default Include File** (File menu) to create an Include File with the extension **.inc**.

3. Use an Include Statement in the beginning of your AHDL Text Design File (**.tdf**) to reference the Include File you just created:

    ```
    INCLUDE "<design name>.inc";
    ```

4. Instantiate the VHDL subdesign with a Variable Declaration or in-line logic function reference in your TDF.

Customer
Application

Altera Devices Speed Alcatel's Development Process

The Alcatel design team liked Altera devices because they permit vertical migration and allow easy testing.

The electronic market is continuously expanding, forcing designers to get their products to market faster. This time-to-market demand, coupled with increasing design complexity, has forced designers to re-examine their development process; traditional simulation and verification methods no longer meet a project's system and deadline requirements.

Designers at Alcatel Telecom, an industry leader in telecommunications equipment, recently faced this challenge. Alcatel engineers needed to speed the development process of their 1570 BB Cablephone, an optical communications system that transmits and distributes broadband services (i.e., cable TV and sound radio) as well as narrowband services (i.e., POTS, ISDN, 2 MBps G.703, and leased lines), via optical fibers. Designers of the 1570 BB Cablephone needed a flexible hardware solution for performing system integration and manufacturing tests.

To perform these functions, designers of the 1570 BB Cablephone created two development boards, the Hardware Simulator and Development Kit, which contain Altera® FLEX® 10K, FLEX 8000, and MAX® 7000 devices. The Alcatel design team liked Altera devices because they permit vertical migration and allow easy testing. In addition, Altera devices are supported by the easy-to-use MAX+PLUS® II development system, which provides seamless integration with the Synopsys Design Compiler that is already used by Alcatel engineers.

Hardware Simulator

Alcatel designers needed a way to test their design before building the ASIC-based hardware. The Hardware Simulator enabled the design team to build a real-time hardware model of the signal processing function and complete hardware tests. The Hardware Simulator is described below:

- 233 mm × 210 mm development board with 200,000 gates (See Figure 1)
- Four EPF10K50 devices
- One EPM7032 device
- Two slots for standard SIMM memory modules
- Two voltage-controlled oscillators (48,640 MHz and 16,384 MHz)
- On-board power-up reset circuit
- Area for user applications

To achieve the highest performance, Alcatel engineers divided the Hardware Simulator's functional modules and implemented them in four EPF10K50RC240-4 devices. Because these devices are pin-compatible with FLEX 10K devices in other speed grades and densities (e.g., EPF10K70 and EPF10K100 devices), the engineers can increase the density of the Hardware Simulator to 400,000 gates without

Figure 1. 233 mm × 210 mm Development Board with 200,000 Gates
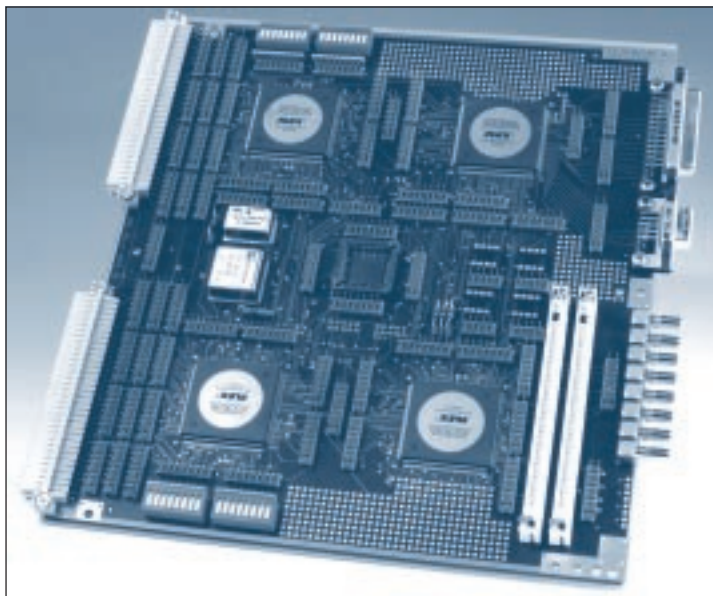
redesigning the board. The EPM7032LC44 device provides the clock tree and can be used for the division and distribution of the system clock.

The Alcatel designers were able to use the Hardware Simulator prototyping board and the MAX+PLUS II development system to determine which signal processing algorithm to use in the ASIC. The designers tested several signal processing algorithms written in MATLAB and VHDL and created several co-simulations. The prototyping board offers the capability to implement emulation models, which enables real-time signal processing. The emulated signal processing algorithm became part of the new ASIC generation.

Development Kit

Alcatel Telecom needed a way to perform system integration tests while the ASIC hardware was still in production. The Development Kit enabled the engineers to perform system integration tests within a real environment under real-time conditions—all long before the ASIC hardware was available. The designers were able to convert some of the ASIC functions written in VHDL, implement the functions in EPF81188 devices, and perform the system integration within a matter of days. The Development Kit contains the following items:

■ 233 mm × 160 mm development board with 20,000 gates (See Figure 2)
■ Two EPF81188 devices
■ Two voltage-controlled oscillators (both 16,384 MHz)
■ On-board power-on reset circuit
■ Area for user applications

With the emulation model based on the prototyping board, the designers were able to completely debug their software design. Nine weeks later, the ASIC hardware was available and regression tests of the software were performed without difficulties.

Late in the system integration cycle, Alcatel discovered some problems with one ASIC on the PCB; the implemented phase-locked loop (PLL) circuit seemed to be causing problems, but software simulations could not detect the cause. To find the problem, the designers used the Development Kit prototyping board to create a hardware implementation of the PLL circuit. It took only two days for the designers to convert the ASIC design, implement it in the EPF81188 devices, and find the cause of the problem. Alcatel was then able to create a workaround and incorporate it into the design.

After the system design was complete, Alcatel needed to build electronic tools to test the PCB assemblies. The Development Kit prototyping board provided an easy way to build test equipment because Altera-based prototyping boards are much cheaper and more flexible than common signal generators. The Development Kit became part of several electronic test tools

It took only two days for the designers to convert the ASIC design, implement it in the EPF81188 devices, and find the cause of the problem.

Contact Information:
Lothar Brodbeck
AS/EC1
Alcatel Telecom
Lorenzstraße 10
70435 Stuttgart
Germany
E-mail: lothar.brodbeck
@ks.sel.alcatel.de

Figure 2. 233 mm × 160 mm Development Board with 20,000 Gates
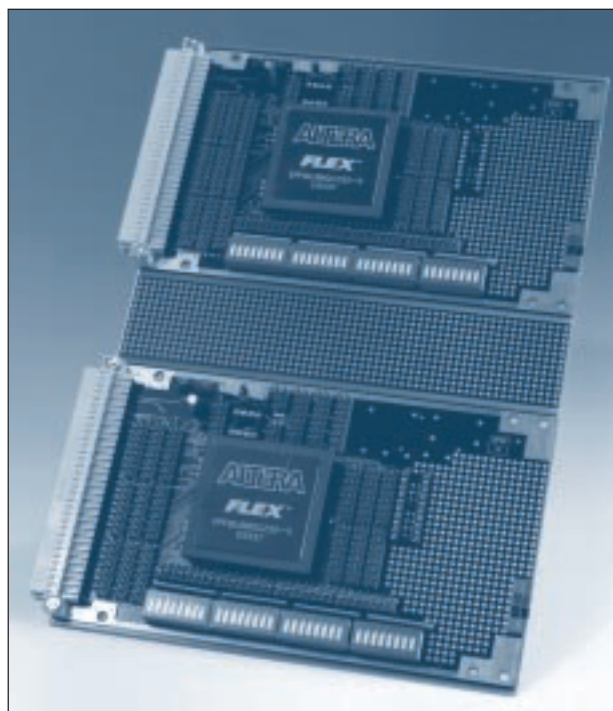
Customer Applications, continued from page 21

and was used during the development process to test the PCB assemblies of the Alcatel 1570 BB Cablephone. Later in the manufacturing phase, the Development Kit was used in the automatic test system for manufacturing tests.

Conclusion

Altera devices provided an easy way for Alcatel designers to perform system integration and manufacturing tests. While the ASIC design and hardware are still extremely useful in speeding the development process of the 1570 BB Cablephone, Alcatel is planning to sell both development boards on the market.

Altera Megafunctions

Altera's megafunction program provides solutions to your application-specific needs. These solutions leverage Altera® MegaCore™ functions and Altera Megafunctions Partners Program (AMPP℠) functions to provide integrated solutions that deliver significant time-to-market benefits. The complete megafunction solution includes both the logic core and the documentation that is critical to get these megafunctions working in-system. Altera megafunctions focus on markets such as digital signal processing (DSP), peripheral component interconnect (PCI), and wireless and broadband communications. For more information, go to the Altera web site at **http://www.altera.com**.

Altera megafunction solutions include:

- FLEX DSP
    - DSP Building Blocks
    - DSP Imaging Functions
    - DSP Wireless & Broadband Communications
    - DSP Error Correction & Encode/Decode

- Bus Interfaces
    - Peripheral Component Interconnect (PCI)
    - Universal Serial Bus (USB)
    - Controller Area Network (CAN) Bus
    - I²C Interface
    - IEEE 1494 (Firewire)

- Communications
    - Data Communications & Telecommunications
    - Asynchronous Transfer Mode (ATM)

Discontinued Devices Update

In recent months, Altera has announced that various products will be discontinued (see the table below). Altera distributes advisories (ADVs) and product discontinuance notices (PDNs) that provide information on discontinued devices. To obtain a copy of a specific ADV or PDN, contact your local Altera sales representative. Selected ADVs, PDNs, and a complete listing of discontinued devices are also available on Altera's world-wide web site at **http://www.altera.com**. Rochester Electronics, an after-market supplier, offers support for many discontinued Altera products. Contact Rochester Electronics at (508) 462-9332 or go to their web site at **http://www.rocelec.com**.

Discontinued Device Ordering Codes

Device Family	Device	Last Order Date	Last Shipment Date	Reference
FLASHlogic	EPX880 and EPX8160 (all packages, temperature grades, and speed grades)	6/30/97	6/30/98	PDN 9625
MAX 7000	EPM7256SRC208-12	12/31/97	3/31/98	PDN 9713

Implementing the SFIFO Function in FLEX 10K EABs

The MAX+PLUS® II software provides a synchronous, first-in first-out (FIFO) buffer that can act as a cycle-shared, logic-cell-based, or interleaved-memory FIFO function. This article explains how to use the interleaved-memory FIFO function. With an interleaved-memory FIFO, simultaneous read and write transactions can occur between two embedded array blocks (EABs), improving the design's performance. Figure 1 shows the symbol for the synchronous FIFO (sfifo) function.

Figure 1. sfifo Symbol


```
                                    LPM_NUMWORDS=256
                                    LPM_WIDTH=width
                                    USE_EAB="ON"

                    SFIFO

          data[ ]              q[ ]
          wreq[ ]             empty
          rreq                 full
          clock
```
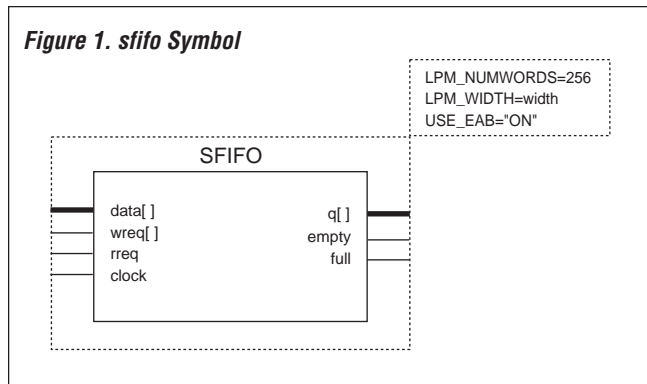
Interleaved-memory FIFO transactions are different from random access memory (RAM) transactions because:

■ The transactions occur in consecutive locations (i.e., read or write transactions occur in memory blocks 1, 2, 3, and so on).
■ The data words are stored in multiple memory blocks.

With consecutive memory access, the sfifo function writes to one FLEX® 10K EAB while reading from another. Thus, the sfifo function does not require a doubled clock, as cycle-shared FIFOs do.

The sfifo function uses 2 EABs per 8 bits of width and can store up to 512 words before requiring more EABs (e.g., either a 256×8 or a 512×8 FIFO requires just two EABs). Therefore, the sfifo function provides high performance for both small and large FIFO applications. For example, a 512×8 FIFO compiled for a -1 speed grade EPF10K50V device operates at 69 MHz.

The sfifo function operates just like any other FIFO (i.e., write transactions are valid when the empty flag is high, and read transactions are valid when the full flag is high). The sfifo function can be entered in a schematic, Verilog HDL, VHDL, or Altera Hardware Description Language (AHDL) design and compiled with the MAX+PLUS II software.

Tables 1 and 2 list the ports and parameters of the sfifo function. To use the sfifo function as an interleaved memory FIFO, do not use the clockx2 port and set the USE_EAB parameter high.

Table 1. sfifo Function Ports

Name	Function
aclr	Asynchronous clear
clock	System clock
clockx2	Doubled system clock. When using the sfifo function as an interleaved memory FIFO, do not use this port.
data[]	Data input
rreq	Read request
wreq	Write request
threshlevel[]	Threshold level which controls threshold output. It can be used to signify when the FIFO is almost full or almost empty.
q[]	Data output
empty	Empty flag
full	Full flag
usedw[]	Number of data words currently stored in the FIFO.
threshold	FIFO is storing more words than the threshlevel[] input.

Table 2. sfifo Function Parameters

Name	Function
LPM_WIDTH	Width of the data bus
LPM_NUMWORDS	Depth of the FIFO
USE_EAB	Can be set high to use the sfifo function as an interleaved-memory FIFO.

Design Tips
Altera Applications

Using VHDL If Statements Efficiently

This article is the first of a four-part series that discusses ways to achieve optimal logic cell utilization with VHDL in the MAX+PLUS® II software. Upcoming issues of News & Views will feature the following topics:

- *State machine coding styles*
- *Using arithmetic operators with MAX+PLUS II VHDL*
- *Importance of hierarchical instantiation*

By using design techniques discussed in this article, you will be able to achieve optimal logic cell usage from a synthesized design easily. For example, using If Statements efficiently can dramatically improve a design's performance.

Using If Statements is a good way to optimize logic cell utilization. Because If Statements are used in Process Statements, they are decoded sequentially to achieve priority encoding. However, sometimes priority encoding should be reconsidered.

The following two examples show sample VHDL code and the resulting MAX+PLUS II Report File (**.rpt**) equation. Example 1 shows an inefficient use of If Statements; Example 2 provides a solution for improving the code.

In Example 1, the `ELSIF` clause causes `output3` to be dependent on decoding all states, which requires much more complex logic.

Example 1

```
PROCESS(current_state,x,y,z)
BEGIN
   output1 <= "000";
   output2 <= "000";
   output3 <= "000";
   IF current_state = (s1 OR s3 OR
      s4) THEN output1 <= x;
   ELSIF current_state = (s0 OR s2 OR
      s5) THEN output2 <= y;
   ELSIF current_state = (s6 OR s7 OR
      s8) THEN output3 <= z;
   END IF;
END PROCESS;
```

The MAX+PLUS II Report File equation for Example 1 is shown below:

```
output3 = z and ((s6 or s7 or s8) and
   (not(s0) and not(s1) and not(s2)
   and not(s3) and not(s4) and not(s5)))
```

This equation assumes that the state machine is implemented as a one-hot state machine, which is generally more efficient for FLEX® devices.

In Example 2, the If Statements are separated to remove the priority encoding on the mutually exclusive outputs. The result is a much smaller logic equation that represents a reduction of logic resource usage. The net result is increased routability and dramatically improved performance.

Example 2

```
PROCESS(current_state,x,y,z)
BEGIN
   output1 <= "000";
   output2 <= "000";
   output3 <= "000";
   IF current_state = (s1 OR s3 OR
      s4) THEN output1 <= x;
   END IF;
   IF current_state = (s0 OR s2 OR s5)
      THEN output2 <= y;
   END IF;
   IF current_state = (s6 OR s7 OR s8)
      THEN output3 <= z;
   END IF;
END PROCESS;
```

The MAX+PLUS II Report File Equation for Example 2 is shown below:

```
output3 = z & (s6 or s7 or s8)
```

For more information on achieving optimal logic cell utilization with VHDL code, go to MAX+PLUS II Help or contact Altera Applications at (800) 800-EPLD.

Altera Announces APEX & ASCEND Programs

Altera recently announced alliances with Exemplar Logic and Synplicity, Inc. to develop design software and methodologies for programmable logic devices (PLDs). The Altera Partnership with Exemplar Logic (APEX℠) and Altera & Synplicity Cooperative Engineering Development (ASCEND℠) programs are based on sharing engineering and marketing resources. The programs will develop easy-to-use design tools that can address PLD designs with 1,000,000 or more logic gates, with the goal of enhancing designer productivity.

Increasing Productivity & Performance

As part of the APEX program, Altera and Exemplar Logic will focus on the following goals:

- Improving designer productivity
- Maximizing device performance
- Developing and marketing state-of-the-art design software and methodologies for the next generation of Altera PLDs

The companies will also develop ways of enhancing integration between Exemplar Logic's synthesis and place-and-route tools and Altera's MAX+PLUS® II development software. By improving placement efficiency, Altera and Exemplar Logic will increase design performance and automation and enhance optimization engines.

Creating Superior Design Features

As part of the ASCEND program, Altera and Synplicity will focus on the following goals:

- Creating superior synthesis and high-level design methodologies
- Developing improvements for such features as floorplanning, timing constraints and analysis, integration of megafunctions, and the library of parameterized modules (LPM)
- Increasing the quality of design implementation results

The companies will also focus on improving the optimization level between Synplicity's flagship synthesis engine, Synplify, and the MAX+PLUS II development software.

Conclusion

The APEX and ASCEND programs are the next step in Altera's commitment to developing advanced design tools, which began last year when a five-year strategic agreement was signed with Synopsys. These programs are a part of Altera's ongoing effort to create alliances with leading third-party design tool vendors in order to give designers state-of-the-art tools to enhance productivity when they are using high-density PLDs.

New Altera Publications

New Altera publications are available from Altera Literature Services. Individual documents are available on Altera's world-wide web site at **http://www.altera.com**. Document part numbers are shown in parentheses.

- *1998 Data Book* (A-DB-0198-01)
- *FLEX 10K PCI Prototype Board Data Sheet* (A-DS-PCI-01)
- *Altera Digital Library CD-ROM* version 3 (P-CD-ADL-03)
- *In-System Programmability CD-ROM* version 4 (M-CD-ISP-04)

- *MAX+PLUS II & ACCESS Partner EDA Tool Compatibility Guide* (P25-05870-00)
- *SB 31: I²C Megafunctions* (A-SB-031-01)
- *TB 33: Evaluating MAX 7000S Device Utilization & Fitting* (M-TB-033-01)
- *TB 34: MAX 7000S Power Consumption* (M-TB-034-02)
- *TB 35: Download the PLS_WEB MAX+PLUS II Software for Free* (M-TB-035-01)
- *TB 36: Timing-Driven Compilation Improvements in MAX+PLUS II Version 8.2* (M-TB-036-01)
- *TB 37: Power Consumption in FLEX 10K Devices* (M-TB-037-01)
- *TB 38: FLEX 10KA-1 Devices: The Fastest High-Density Devices Available*

Third-Party Programming Support

Data I/O and BP Microsystems provide programming hardware support for selected Altera devices. Algorithms are supplied via Data I/O's Keep Current Express-Bulletin Board Service (KCE-BBS) and BP Microsystems' BBS. Programming support for Configuration EPROM, MAX® 9000, and MAX 7000 devices is shown in the table below. All information is subject to change.

Third-Party Programming Hardware Support		
Device	**Data I/O** (1)	**BP Microsystems** (2)
EPC1064	✓	✓
EPC1213	✓	✓
EPC1	✓	✓
EPC1441	Note (3)	✓
EPM7032	✓	✓
EPM7064	✓	✓
EPM7064S	✓	✓
EPM7096	✓	✓
EPM7128E	✓	✓
EPM7128S	✓	✓
EPM7160E	✓	✓
EPM7192E	✓	✓
EPM7192S	✓	✓
EPM7256E	✓	✓
EPM7256S	✓	✓
EPM9320	✓	✓
EPM9320A	✓	✓
EPM9400	✓	✓
EPM9480	✓	✓
EPM9560	✓	✓
EPM9560A	✓	✓

Notes:
(1) These devices are supported by Data I/O 3900 version 5.6 and UniSite version 5.6 programmers.
(2) These devices are supported by BP Microsystems programmers version 3.29.
(3) Data I/O plans to support EPC1441 devices. Contact Data I/O directly for more information.

Current Software Versions

The latest versions of Altera software products are shown below:

- MAX+PLUS® II version 8.2
  (PC, Sun SPARCstation, HP 9000 Series 700/800, and IBM RISC System/6000 platforms)

Programming Hardware Support

The following tables contain the latest programming hardware information for Altera devices. For correct programming, use the software version shown in "Current Software Versions" below. See Table 1.

Table 1. Altera Programming Adapters (Part 1 of 2)		Note (1)
Device	**Package**	**Adapter**
EPC1064 (2), EPC1064V (2), EPC1441 (3)	DIP, J-lead TQFP	PLMJ1213 PLMT1064
EPC1 (3), EPC1213, (2)	DIP J-lead	PLMJ1213 PLMJ1213
EPM9320A	J-lead (84-pin)	PLMJ9320-84
	RQFP (208-pin)	PLMR9000-208NC (4)
	RQFP (240-pin)	PLMR9000-240NC (4)
EPM9320	PGA	PLMG9000-280
	J-lead (84-pin)	PLMJ9320-84
	RQFP (208-pin)	PLMR9000-208
EPM9400	J-lead (84-pin)	PLMJ9400-84
	RQFP (208-pin)	PLMR9000-208
	RQFP (240-pin)	PLMR9000-240
EPM9480	RQFP (208-pin)	PLMR9000-208
	RQFP (240-pin)	PLMR9000-240
EPM9560A	RQFP (208-pin)	PLMR9000-208NC (4)
	RQFP (240-pin)	PLMR9000-240NC (4)
EPM9560	PGA (280-pin)	PLMG9000-280
	RQFP (208-pin)	PLMR9000-208
	RQFP (240-pin)	PLMR9000-240
	RQFP (304-pin)	PLMR9000-304
EPM7032S	J-lead (44-pin)	PLMJ7000-44
	TQFP (44-pin)	PLMT7000-44
EPM7032, EPM7032V	J-lead (44-pin)	PLMJ7000-44
	PQFP (44-pin)	PLMQ7000-44
	TQFP (44-pin)	PLMT7000-44
EPM7064S	J-lead (44-pin)	PLMJ7000-44
	TQFP (44-pin)	PLMT7000-44
	TQFP (100-pin)	PLMT7000-100NC (4)
EPM7064	J-lead (44-pin)	PLMJ7000-4
	TQFP (44-pin)	PLMT7000-44
	J-lead (68-pin)	PLMJ7000-68
	J-lead (84-pin)	PLMJ7000-84
	PQFP (100-pin)	PLMQ7000-100
EPM7096	J-lead (68-pin)	PLMJ7000-68
	J-lead (84-pin)	PLMJ7000-84
	PQFP (100-pin)	PLMQ7000-100
EPM7128S	J-lead (84-pin)	PLMJ7000-84
	PQFP (100-pin)	PLMQ7000-100NC (4)
	TQFP (100-pin)	PLMT7000-100NC (4)
	PQFP (160-pin)	PLMQ7128/160-160NC
EPM7128, EPM7128E	J-lead (84-pin)	PLMJ7000-84
	PQFP (100-pin)	PLMQ7000-100
	PQFP (160-pin)	PLMQ7128/7160-160

Table 1. Altera Programming Adapters (Part 2 of 2)   *Note (1)*

Device	Package	Adapter
EPM7160S	J-lead PQFP (100-pin) PQFP (160-pin)	PLMJ7000-84 PLMQ7000-100NC *(4)* PLMQ7128/7160-160NC *(4)*
EPM7160E	J-lead PQFP (100-pin) PQFP (160-pin)	PLMJ7000-84 PLMQ7000-100 PLMQ7128/7160-160
EPM7192S *(4)*	PQFP (160-pin)	PLMQ7192/256-160NC
EPM7192E	PGA (160-pin) PQFP (160-pin)	PLMG7192-160 PLMQ7192/7256-160
EPM7256S *(4)*	RQFP (208-pin)	PLMQ7256-208NC
EPM7256E	PQFP (160-pin) PGA (192-pin) RQFP (208-pin)	PLMQ7192/7256-160 PLMG7256-192 PLMR7256-208

Notes:
(1) Refer to the Altera *1998 Data Book* for device adapter information on MAX 5000 and Classic devices. Altera offers an adapter exchange program for 0.8-micron EPM5032, EPM5064, and EPM5130 programming adapters.
(2) FLEX 8000 Configuration EPROM.
(3) FLEX 10K, FLEX 8000, or FLEX 6000 Configuration EPROM.
(4) These devices are not shipped in carriers.

Table 2 provides programming and configuration compatibility information for the BitBlaster™ serial and ByteBlaster™ parallel port download cables.

Table 2. BitBlaster & ByteBlaster Cable Compatibility

Device	Package	Hardware
FLEX 10K FLEX 10KA	All packages	PL-BITBLASTER PL-BYTEBLASTER
FLEX 8000	All packages	PL-BITBLASTER PL-BYTEBLASTER
FLEX 6000	All packages	PL-BITBLASTER PL-BYTEBLASTER
MAX 9000 MAX 9000A	All packages	PL-BITBLASTER PL-BYTEBLASTER
MAX 7000S MAX 7000A	All packages	PL-BITBLASTER PL-BYTEBLASTER

Altera Device Selection Guide

All current information for the Altera FLEX 10K, FLEX 8000, FLEX 6000, MAX 9000, and MAX 7000 devices is listed here. Information on other Altera products is located in the Altera *1998 Data Book*. Contact Altera or your local sales office for current product availability.

FLEX 10K Devices	(Part 1 of 2)							
DEVICE	**GATES**	**PIN/PACKAGE OPTIONS**	**I/O PINS** *Note (1)*	**TEMP.**	**SPEED GRADE**	**FLIPFLOPS**	**LOGIC ELEMENTS**	**RAM BITS**
EPF10K10	10,000	84-Pin PLCC, 144-Pin TQFP, 208-Pin QFP	59, 102, 134	C	-3, -4	720	576	6,144
EPF10K10	10,000	144-Pin TQFP, 208-Pin QFP	102, 134	I	-4	720	576	6,144
EPF10K10A	10,000	144-Pin TQFP, 208-Pin QFP	102, 134	C	-1, -2, -3	720	576	6,144
EPF10K10A	10,000	144-Pin TQFP, 208-Pin QFP	102, 134	I	-3	720	576	6,144
EPF10K20	20,000	144-Pin TQFP, 208-Pin QFP, 240-Pin QFP	102, 147, 189	C	-3, -4	1,344	1,152	12,288
EPF10K20	20,000	144-Pin TQFP, 208-Pin QFP, 240-Pin QFP	102, 147, 189	I	-4	1,344	1,152	12,288
EPF10K30	30,000	208-Pin QFP, 240-Pin QFP, 356-Pin BGA	147, 189, 246	C	-3, -4	1,968	1,728	12,288
EPF10K30	30,000	208-Pin QFP, 240-Pin QFP	147, 189	I	-4	1,968	1,728	12,288
EPF10K30A	30,000	144-Pin TQFP, 208-Pin QFP, 240-Pin QFP, 256-Pin BGA	102, 147, 189, 189	C	-3, -4	1,968	1,728	12,288
EPF10K30A	30,000	144-Pin TQFP, 208-Pin QFP, 240-Pin QFP, 256-Pin BGA	102, 147, 189, 189	I	-4	1,968	1,728	12,288
EPF10K40	40,000	208-Pin QFP, 240-Pin QFP	147, 189	C	-3, -4	2,576	2,304	16,384
EPF10K50	50,000	240-Pin QFP, 356-Pin BGA, 403-Pin PGA	189, 274, 310	C	-3, -4	3,184	2,880	20,480
EPF10K50	50,000	240-Pin QFP	189	I	-4	3,184	2,880	20,480
EPF10K50V	50,000	240-Pin QFP, 356-Pin BGA	189, 274	C	-1, -2, -3, -4	3,184	2,880	20,480
EPF10K50V	50,000	240-Pin QFP, 356-Pin BGA	189, 274	I	-4	3,184	2,880	20,480
EPF10K70	70,000	240-Pin QFP, 503-Pin QFP	189, 358	C	-3, -4	4,096	3,744	18,432
EPF10K70	70,000	240-Pin QFP	189	C	-2	4,096	3,744	18,432
EPF10K100	100,000	503-Pin QFP	406	C	-3, -4	5,392	4,992	24,576

Altera Device Selection Guide, continued from page 22

FLEX 10K Devices (Part 2 of 2)								
DEVICE	**GATES**	**PIN/PACKAGE OPTIONS**	**I/O PINS** *Note (1)*	**TEMP.**	**SPEED GRADE**	**FLIPFLOPS**	**LOGIC ELEMENTS**	**RAM BITS**
EPF10K100A	100,000	240-Pin QFP, 356-Pin BGA, 600-Pin BGA	189, 274, 406	C	-1, -2, -3	5,392	4,992	24,576
EPF10K100A	100,000	240-Pin QFP, 356-Pin BGA, 600-Pin BGA	189, 274, 406	I	-3	5,392	4,992	24,576
EPF10K130V	130,000	599-Pin BGA, 600-Pin BGA	470, 470	C	-2, -3, -4	7,120	6,656	32,768
EPF10K250A	250,000	599-Pin BGA, 600-Pin BGA	470, 470	C	-1, -2, -3	12,624	12,160	40,960
EPF10K250A	250,000	600-Pin BGA	470	I	-3	12,624	12,160	40,960

Notes:

(1) Six I/O pins are dedicated inputs.

FLEX 8000 Devices							
DEVICE	**GATES**	**PIN/PACKAGE OPTIONS**	**I/O PINS** *(1)*	**TEMP.**	**SPEED GRADE**	**FLIPFLOPS**	**LOGIC ELEMENTS**
EPF8282A	2,500	84-Pin PLCC, 100-Pin TQFP	68, 78	C	A-2	282	208
		84-Pin PLCC, 100-Pin TQFP	68, 78	C, I	A-3		
		84-Pin PLCC, 100-Pin TQFP	68, 78	C, I	A-4		
EPF8282AV *(2)*	2,500	100-Pin TQFP	78	C	A-4	282	208
EPF8452A	4,000	160-Pin PQFP	120	C	A-2	452	336
		84-Pin PLCC, 100-Pin TQFP, 160-Pin PGA/PQFP	68, 120	C, I	A-3		
		84-Pin PLCC, 100-Pin TQFP, 160-Pin PGA/PQFP	68, 120	C, I	A-4		
EPF8636A	6,000	208-Pin PQFP	136	C	A-2	636	504
		84-Pin PLCC, 160-Pin PQFP, 192-Pin PGA, 208-Pin PQFP	68, 118, 136	C	A-3		
		84-Pin PLCC, 160-Pin PQFP, 192-Pin PGA, 208-Pin PQFP	68, 118, 136	C, I	A-4		
EPF8820A	8,000	144-Pin TQFP, 160-Pin PQFP, 192-Pin PGA, 208-Pin PQFP	112, 120, 152	C	A-2	820	672
		144-Pin TQFP, 160-Pin PQFP, 192-Pin PGA, 208-Pin PQFP, 225-Pin BGA	112, 120, 152	C	A-3		
		144-Pin TQFP, 160-Pin PQFP, 192-Pin PGA, 208-Pin PQFP, 225-Pin BGA	112, 120, 152	C, I	A-4		
EPF81188A	12,000	208-Pin PQFP, 232-Pin PGA, 240-Pin PQFP	148, 184	C	A-2	1,188	1,008
		208-Pin PQFP, 232-Pin PGA, 240-Pin PQFP	148, 184	C, I	A-3		
		208-Pin PQFP, 232-Pin PGA, 240-Pin PQFP	148, 184	C, I	A-4		
EPF81500A	16,000	240-Pin PQFP, 280-Pin PGA, 304-Pin RQFP	181, 208	C	A-2	1,500	1,296
		240-Pin PQFP, 280-Pin PGA, 304-Pin RQFP	181, 208	C, I	A-3		
		240-Pin PQFP, 280-Pin PGA, 304-Pin RQFP	181, 208	C	A-4		

Notes:

(1)   Four I/O pins are dedicated inputs.
(2)   A "V" indicates a 3.3-V voltage supply.

MAX 9000 Devices					
DEVICE	**MACRO-CELLS**	**PIN/PACKAGE OPTIONS**	**I/O PINS** *(1)*	**TEMP.**	**SPEED GRADE**
EPM9320	320	84-Pin PLCC, 208-Pin RQFP, 280-Pin PGA, 356-Pin BGA	60, 132, 168	C	-15
	320	84-Pin PLCC, 208-Pin RQFP, 280-Pin PGA, 356-Pin BGA	60, 132, 168	C, I	-20
EPM9320A	320	84-Pin PLCC, 208-Pin RQFP, 356-Pin BGA	60, 132, 168	C	-10
	320	84-Pin PLCC, 208-Pin RQFP, 356-Pin BGA	60, 132, 168	C, I	-15
EPM9400	400	84-Pin PLCC, 208-Pin RQFP, 240-Pin RQFP	59, 139, 159	C	-15, -20
EPM9480	480	208-Pin RQFP, 240-Pin RQFP	146, 175	C	-15, -20
EPM9480A	480	208-Pin RQFP, 240-Pin RQFP	146, 175	C	-10, -12
	480	208-Pin RQFP, 240-Pin RQFP	146, 175	C, I	-15
EPM9560	560	208-Pin RQFP, 240-Pin RQFP, 280-Pin PGA, 304-Pin RQFP, 356-Pin BGA	153, 191, 216	C	-15
	560	208-Pin RQFP, 240-Pin RQFP, 280-Pin PGA, 304-Pin RQFP, 356-Pin BGA	153, 191, 216	C,I	-20
EPM9560A	560	208-Pin RQFP, 240-Pin RQFP, 304-Pin RQFP, 356-Pin BGA	153, 191, 216	C	-10, -12
	560	208-Pin RQFP, 240-Pin RQFP, 304-Pin RQFP, 356-Pin BGA	153, 191, 216	C,I	-15

Note:

(1)   Four I/O pins are dedicated inputs.

FLEX 6000 Devices

DEVICE	GATES	PIN/PACKAGE OPTIONS	I/O PINS (1)	TEMP.	SPEED GRADE	FLIPFLOPS	LOGIC ELEMENTS
EPF6016	16,000	100-Pin TQFP, 144-Pin TQFP, 208-Pin QFP, 240-Pin PQFP, 256-Pin BGA	117, 171, 199, 204	C	-2	1,320	1,320
		100-Pin TQFP, 144-Pin TQFP, 208-Pin QFP, 240-Pin PQFP, 256-Pin BGA	117, 171, 199, 204	C, I	-3	1,320	1,320
EPF6016A (2)	16,000	100-Pin TQFP, 144-Pin TQFP, 208-Pin QFP, 240-Pin PQFP, 256-Pin BGA	81, 117, 171	C	-2	1,320	1,320
		100-Pin TQFP, 144-Pin TQFP, 208-Pin QFP, 240-Pin PQFP, 256-Pin BGA	81, 117, 171	C, I	-3	1,320	1,320
EPF6024A (2)	24,000	140-Pin TQFP, 208-Pin QFP, 240-Pin PQFP, 256-Pin BGA	117, 171, 199, 214	C	-2	1,960	1,960
		140-Pin TQFP, 208-Pin QFP, 240-Pin PQFP, 256-Pin BGA	117, 171, 199, 214	C, I	-3	1,960	1,960

Notes:
(1)   Four I/O pins are dedicated inputs.
(2)   An "A" indicates a 3.3-V voltage supply.

MAX 7000 Devices

DEVICE	MACRO-CELLS	PIN/PACKAGE OPTIONS	I/O PINS	TEMP.	SPEED GRADE	t_{PD} (ns)	f_{CNT} (MHz)
EPM7032, EPM7032S	32	44-Pin PLCC/TQFP	36	C	-6	6	150
EPM7032, EPM7032S	32	44-Pin PLCC/TQFP	36	C	-7	7.5	125
EPM7032, EPM7032S	32	44-Pin PLCC/TQFP	36	C, I (1)	-10	10	100
EPM7032	32	44-Pin PLCC/TQFP	36	C, I	-12	12	90.9
EPM7032	32	44-Pin PLCC/TQFP	36	C, I	-15	15	76.9
EPM7032V (2)	32	44-Pin PLCC/TQFP	36	C	-12	12	90.9
EPM7032V (2)	32	44-Pin PLCC/TQFP	36	C	-15	15	76.9
EPM7032V (2)	32	44-Pin PLCC/TQFP	36	C, I	-20	20	62.5
EPM7064S	64	44-Pin PLCC/TQFP	36	C	-5	5	178.6
EPM7064, EPM7064S	64	44-Pin PLCC/TQFP, 68-Pin PLCC (1), 84-Pin PLCC, 100-Pin PQFP (1)/TQFP (2)	36, 52, 68	C	-6	6	150
EPM7064, EPM7064S	64	44-Pin PLCC/TQFP, 68-Pin PLCC (1), 84-Pin PLCC, 100-Pin PQFP (1)/TQFP (2)	36, 52, 68	C, I (2)	-7	7.5	125
EPM7064, EPM7064S	64	44-Pin PLCC/TQFP, 68-Pin PLCC (1), 84-Pin PLCC, 100-Pin PQFP (1)/TQFP (2)	36, 52, 68	C, I (1)	-10	10	100
EPM7064	64	44-Pin PLCC/TQFP, 68-Pin PLCC, 84-Pin PLCC, 100-Pin PQFP	36, 52, 68	C	-12	12	90.9
EPM7064	64	44-Pin PLCC/TQFP, 68-Pin PLCC, 84-Pin PLCC, 100-Pin PQFP	36, 52, 68	C, I	-15	15	76.9
EPM7096	96	68-Pin PLCC (1), 84-Pin PLCC, 100-Pin PQFP/TQFP	52, 64, 76	C	-7	7.5	125
EPM7096	96	68-Pin PLCC, 84-Pin PLCC, 100-Pin PQFP/TQFP	52, 64, 76	C, I	-10	10	100
EPM7096	96	68-Pin PLCC, 84-Pin PLCC, 100-Pin PQFP	52, 64, 76	C, I	-12	12	90.9
EPM7096	96	68-Pin PLCC, 84-Pin PLCC, 100-Pin PQFP	52, 64, 76	C	-15	15	76.9
EPM7128S	128	84-Pin PLCC, 100-Pin PQFP, 160-Pin PQFP	68, 84, 100	C	-6	6	150
EPM7128E, EPM7128S	128	84-Pin PLCC, 100-Pin PQFP/TQFP (2), 160-Pin PQFP	68, 84, 100	C, I (2)	-7	7.5	125
EPM7128E, EPM7128S	128	84-Pin PLCC, 100-Pin PQFP/TQFP (2), 160-Pin PQFP	68, 84, 100	C	-10(P)	10	100
EPM7128E	128	84-Pin PLCC, 100-Pin PQFP, 160-Pin PQFP	68, 84, 100	C, I	-12	12	90.9
EPM7128E, EPM7128S	128	84-Pin PLCC, 100-Pin PQFP/TQFP (2), 160-Pin PQFP	68, 84, 100	C, I	-15	15	76.9
EPM7128E	128	84-Pin PLCC, 100-Pin PQFP, 160-Pin PQFP	68, 84, 100	C	-20	20	62.5
EPM7160S	160	84-Pin PLCC, 100-Pin PQFP/TQFP, 160-Pin PQFP	64, 84, 104	C, I (2)	-7	7.5	125
EPM7160E, EPM7160S	160	84-Pin PLCC, 100-Pin PQFP/TQFP (2), 160-Pin PQFP	64, 84, 104	C	-10(P)	10	100
EPM7160E	160	84-Pin PLCC, 100-Pin PQFP, 160-Pin PQFP	68, 84, 100	C, I	-12	12	90.9
EPM7160E, EPM7160S	160	84-Pin PLCC, 100-Pin PQFP/TQFP (2), 160-Pin PQFP	64, 84, 104	C, I (1)	-15	15	76.9
EPM7160E	160	84-Pin PLCC, 100-Pin PQFP, 160-Pin PQFP	68, 84, 100	C	-20	20	62.5
EPM7192S	192	160-Pin PQFP	124	C, I (2)	-7	7.5	125
EPM7192S	192	160-Pin PQFP	124	C	-10	10	100
EPM7192E	192	160-Pin PQFP/PGA	124	C, I (2)	-12(P)	12	90.9
EPM7192E, EPM7192S	192	160-Pin PQFP/PGA (1)	124	C, I	-15	15	76.9
EPM7192E	192	160-Pin PQFP/PGA	124	C	-20 (1)	20	62.5
EPM7256S	256	208-Pin RQFP	132, 164	C	-7	7.5	125
EPM7256S	256	208-Pin RQFP	132, 164	C	-10	10	100
EPM7256E	256	160-Pin PQFP, 192-Pin PGA, 208-Pin RQFP	132, 164	C, I (2)	-12(P)	12	90.9
EPM7256E, EPM7256S	256	160-Pin PQFP (1), 192-Pin PGA (1), 208-Pin RQFP (2)	132, 164	C, I	-15	15	76.9
EPM7256E	256	160-Pin PQFP, 192-Pin PGA, 208-Pin RQFP , 208-Pin RQFP (2)	132, 164		-20 (1)	20	62.5

Notes:
(1)   Not available in MAX 7000S devices
(2)   Available in MAX 7000S devices only.

Altera Device Selection Guide, continued from page 29

MAX 7000A Devices								
DEVICE	MACRO-CELLS	PIN/PACKAGE OPTIONS	I/O PINS	TEMP.	SPEED GRADE	t_{PD} (ns)	f_{CNT} (MHz)
EPM7032A	32	44-Pin PLCC/TQFP	36	C	-5	5	178.6
EPM7032A	32	44-Pin PLCC/TQFP	36	C, I	-7	7.5	125
EPM7032A	32	44-Pin PLCC/TQFP	36	C	-10	10	100
EPM7064A	64	44-Pin PLCC/TQFP, 84-Pin PLCC, 100-Pin TQFP	36, 68	C	-5	5	178.6
EPM7064A	64	44-Pin PLCC/TQFP, 84-Pin PLCC, 100-Pin TQFP	36, 68	C, I	-7	7.5	125
EPM7064A	64	44-Pin PLCC/TQFP, 84-Pin PLCC, 100-Pin TQFP	36, 68	C	-10	10	100
EPM7128A	128	84-Pin PLCC, 100-Pin TQFP, 144-Pin TQFP	68, 84, 100	C	-5	5	178.6
EPM7128A	128	84-Pin PLCC, 100-Pin TQFP, 144-Pin TQFP	68, 84, 100	C	-6	6	150
EPM7128A	128	84-Pin PLCC, 100-Pin TQFP, 144-Pin TQFP	68, 84, 100	C	-7	7.5	125
EPM7128A	128	84-Pin PLCC, 100-Pin TQFP, 144-Pin TQFP	68, 84, 100	C, I	-10	10	100
EPM7256A	256	100-Pin TQFP, 144-Pin TQFP, 208-Pin PQFP, 256-Pin BGA	84, 120, 164	C	-6	6	150
EPM7256A	256	100-Pin TQFP, 144-Pin TQFP, 208-Pin PQFP, 256-Pin BGA	84, 120, 164	C	-7	7.5	125
EPM7256A	256	100-Pin TQFP, 144-Pin TQFP, 208-Pin PQFP, 256-Pin BGA	84, 120, 164	C, I	-10	10	100
EPM7384A	384	144-Pin TQFP, 208-Pin PQFP, 256-Pin BGA	120, 176, 212	C	-7	7.5	125
EPM7384A	384	144-Pin TQFP, 208-Pin PQFP, 256-Pin BGA	120, 176, 212	C, I	-10	10	100
EPM7384A	384	144-Pin TQFP, 208-Pin PQFP, 256-Pin BGA	120, 176, 212	C	-15	15	76.9
EPM7512A	512	144-Pin TQFP, 208-Pin PQFP, 256-Pin BGA	120, 176, 212	C	-7	7.5	125
EPM7512A	512	144-Pin TQFP, 208-Pin PQFP, 256-Pin BGA	120, 176, 212	C, I	-10	10	100
EPM7512A	512	144-Pin TQFP, 208-Pin PQFP, 256-Pin BGA	120, 176, 212	C	-15	15	76.9
EPM71024A	1,024	208-Pin PQFP, 256-Pin BGA	176, 212	C	-7	7.5	125
EPM71024A	1,024	208-Pin PQFP, 256-Pin BGA	176, 212	C	-10	10	100
EPM71024A	1,024	208-Pin PQFP, 256-Pin BGA	176, 212	C, I	-15	15	76.9

How to Access Altera

Getting information and services from Altera is now easier than ever. The table below lists some of the ways you can reach Altera:

Information Type	Access	U.S. & Canada	All Other Locations
Literature *(1)*	Altera Literature Services	(888) 3-ALTERA lit_req@altera.com	(408) 544-7144 *(2)* lit_req@altera.com
	World-Wide Web	http://www.altera.com	http://www.altera.com
Non-Technical Customer Service	Telephone Hotline	(800) SOS-EPLD	(408) 544-7000
	Fax	(408) 544-6403	(408) 544-6403
Technical Support	Telephone Hotline (6 a.m. to 6 p.m. Pacific Time)	(800) 800-EPLD (408) 544-7000	(408) 544-7000 *(2)*
	Fax	(408) 544-6401	(408) 544-6401 *(2)*
	Electronic Mail	sos@altera.com	sos@altera.com
	FTP Site	ftp.altera.com	ftp.altera.com
General Product Information	Telephone	(408) 544-7104	(408) 544-7104 *(2)*
	World-Wide Web	http://www.altera.com	http://www.altera.com

Notes:
(1) For MAX+PLUS II software manuals, contact Altera Customer Service or your local distributor.
(2) You can also contact your local Altera sales office or sales representative. See the Altera web site for a listing.

Fax Response Form

☐ **I would like a subscription to _News & Views_.**

☐ **I would like to have my design featured in _News & Views_.**

☐ **Please correct my address.**

Your Name: _____

Organization: _____

Street Address: _____

City, State, ZIP: _____

Phone: _____

E-Mail: _____

Tell Us What You Think

Please take a moment to help us improve _News & Views_ by rating the usefulness of the following sections. Your answers will help shape the content of future issues.

		Not Useful				Very Useful
1.	Devices & Tools	1	2	3	4	5
2.	New Altera Publications	1	2	3	4	5
3.	Questions & Answers	1	2	3	4	5
4.	Technical "How To" Articles	1	2	3	4	5
5.	Information on Altera's	1	2	3	4	5
	EDA Partners & Interface Support	1	2	3	4	5
6.	Customer Applications	1	2	3	4	5
7.	Current Software Versions	1	2	3	4	5
8.	Altera News	1	2	3	4	5

Please write your comments about _News & Views_ in the space below (use additional pages if necessary). Which subjects are not getting enough coverage? What questions do you still have? What new features would you like to see?

Please fax or mail a copy of this page to:

Altera Literature Distribution Services
Altera Corporation
101 Innovation Drive
San Jose, CA 95134
Fax: (408) 544-7809
E-mail: n_v@altera.com