White Paper

Using the Intel Flash Memory-Based EPC4, EPC8 & EPC16 Devices

Introduction

Altera® enhanced configuration devices provide single-device, advanced configuration solutions for high-density Altera FPGAs. The core of an enhanced configuration device is divided into two major blocks, a configuration controller and flash memory. The flash memory is used to store configuration data for systems made up of one or more Altera FPGAs. Altera uses a Micron flash memory in the EPC4 device and a Sharp flash memory in the EPC8 and EPC16 devices. Starting in the fourth quarter of 2005, Altera has added Intel flash memory as an additional flash memory source in the EPC4, EPC8, and EPC16 devices.

This document describes the differences between the Micron or Sharp flash memory and the Intel flash memory. It also provides guidelines regarding the transition from the Micron or Sharp flash memory to the Intel Flash memory-based EPC4, EPC8, and EPC16.

Differences Between Intel & Micron Flash-Based EPC4 Devices

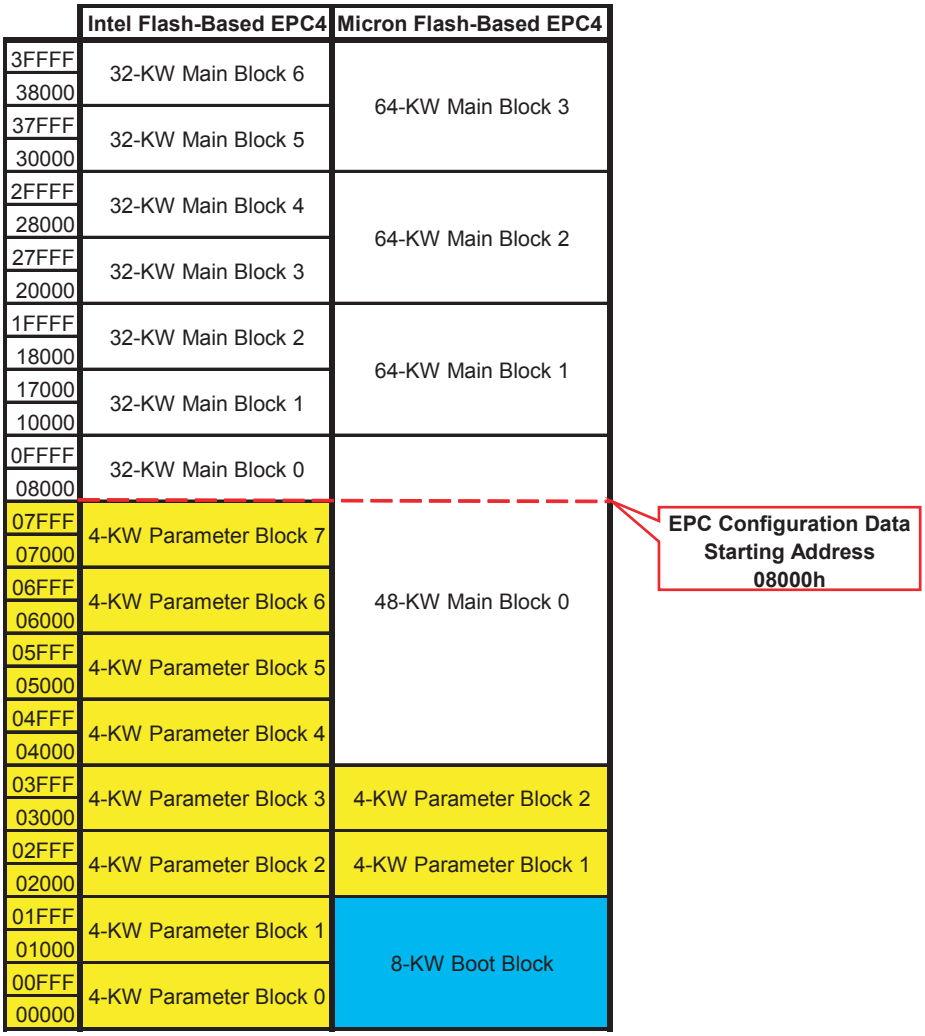
Table 1 lists the differences between the Intel flash memory and the Micron flash memory used in EPC4 devices.

Table 1. Differences Between Intel Flash Memory & Micron Flash Memory in EPC4 Devices

Intel Flash Memory	Micron Flash Memory	Description
No boot block available. Contains: ● Eight 4-KW parameter blocks ● Seven 32-KW main blocks	Contains: ● One 8-KW boot block ● Two 4-KW parameter blocks ● Four 64-KW main blocks	The erase sequence must be changed to accommodate both Intel and Micron flash memory block structures. This is automatically supported by the Altera Quartus® II software beginning with version 5.1 when using programming object files (POF). Other programming files created with Quartus II software beginning with version 5.1, such as serial vector format (SVF), Jam, or Jam byte code (JBC), are also compatible with both structures.

Figure 1 shows the memory map for the Intel flash memory versus the Micron flash memory. The configuration data is always programmed at the start address of 0x08000 regardless of the flash memory technology used in EPC4, 8, and 16 devices. You can store the user data in the parameter blocks and the boot block through the external flash interface. The Intel flash-based EPC4 has 18 address input pins providing access to 4 Mbits of the Intel flash memory.

Figure 1. Memory Map for Intel Flash Memory vs. Micron Flash Memory



Differences Between Intel & Sharp Flash-Based EPC8 & EPC16 Devices

Table 2 lists the differences between the Intel flash memory and the Sharp flash memory used in the EPC8 and EPC16 devices.

Table 2. Differences Between Intel Flash Memory & Sharp Flash Memory in EPC8 & EPC16 Devices

Intel Flash Memory	Sharp Flash Memory	Description
RY/BY# pin not available	RY/BY# pin available	This pin is polled to check the status of the Write State Machine during the execution of internal write processes. Altera does not utilize this pin in the JTAG programming mode. This is not supported in the Intel flash-based EPC8 and EPC16 devices. Any customer system that polls the RY/BY# pin through the external flash interface must be changed to read the status register and the Write State Machine through the DQ[7] configuration data pin. *(1)*
Permanent lock-bit not available	Permanent lock-bit available	If the Sharp permanent lock-bit is programmed, then the device is read-only and Program/Erase operations are permanently disabled. This feature is not available in the Intel flash-based EPC8 and EPC16 devices.
Block lock-bits not available	Block lock-bits available	The Sharp block lock-bits have the ability to lock or unlock individual parameter or main memory blocks. This feature is not available in the Intel flash-based EPC8 and EPC16 devices.

Table 2. Differences Between Intel Flash Memory & Sharp Flash Memory in EPC8 & EPC16 Devices

Intel Flash Memory	Sharp Flash Memory	Description
Full Chip Erase command not available	Full Chip Erase command available	The Sharp Full Chip Erase command is used to erase the entire flash memory in one operation. This feature is not available in the Intel flash memory. Any programming algorithm software using this command has to change to use a Block Erase Command for the Intel flash memory-based EPC8 and EPC16 devices. Sharp flash memory-based EPC8 and EPC16 devices support Block Erase commands. This is automatically supported by the Quartus II software version 5.1 and later when using POF files. Other programming file formats created with the Quartus II software beginning with version 5.1, such as SVF, Jam, or JBC, are also compatible.
Conforms to the 100-pin PQFP package Option 1 outlined in the Altera Device Package Information Data Sheet	Conforms to 100-pin PQFP package Option 2 outlined in the Altera Device Package Information Data Sheet	The Intel flash-based EPC8 and EPC16 Q100 package is JEDEC Q100 Compliant. See PCN 0506 for the package drawing differences.

Note:

(1) To read the status register, issue the Read Status Register (70h) command to the Command User Interface (CUI) of the Intel flash memory. The status register bits are output through the DQ[0] to DQ[7] pins with the DQ[7] pin indicating the status of the Write State Machine (WSM): '0' when the WSM is busy and '1' when the WSM is ready.

Programming Support for Intel Flash Memory-Based EPC4, EPC8 & EPC16 Devices

The EPC4, EPC8, and EPC16 devices can be programmed using POF files and HEXOUT, Jam STAPL, JBC, and SVF programming files. The programming file generation and programming support for the Intel flash memory-based EPC4, EPC8, and EPC16 devices is available in the Quartus II programmer version 5.1 and later. The device ID for the Intel flash-based EPC4, EPC8, and EPC16 devices is different from the Micron/Sharp flash-based EPC4, EPC8, and EPC16 devices, as described in Table 3.

Table 3. EPC4, EPC8 & EPC16 Flash Device IDs

Device	Flash Type	Manufacturing Code	Device ID Code
EPC4	Intel	0x0089	0x8891
	Micron	0x0089	0x0071
EPC8	Intel	0x0089	0x8891
	Sharp	0x00B0	0x00ED
EPC16	Intel	0x0089	0x8891
	Sharp	0x00B0	0x00E9

POF and HEXOUT programming files contain only programming data. Therefore, the POF and Hexout files generated for the Micron/Sharp flash-based EPC4, EPC8, and EPC16 devices are compatible with the Intel flash-based devices. However, POF programming for Intel flash-based EPC4, EPC8, and EPC16 devices is only supported in the Quartus II programmer version 5.1 and later. For POF and Hexout programming using third party tools, contact the vendor to obtain the version that supports the Intel flash-based EPC4, EPC8, and EPC16 devices.

Jam STAPL, JBC, and SVF programming files contain both programming data and algorithm instructions. Therefore, the EPC8 and EPC16 Jam STAPL, JBC, and SVF files generated using the Quartus II software version 5.0 SP1 and earlier, and the MAX+PLUS® II software, are not compatible with the Intel flash memory-based EPC4, EPC8, and EPC16 devices. However, new SVF, Jam STAPL, and JBC programming files generated using Quartus II software version 5.1 and later can program Micron, Sharp, and Intel flash memory devices. You must regenerate the Quartus II version 5.0 SP1 or earlier and MAX+PLUS II Jam STAPL, JBC, and SVF programming files using the Quartus II programmer beginning with version 5.1.

If you have the original design POF programming file, follow the Jam STAPL, JBC, and SVF file generation procedures within the Quartus II programmer beginning with version 5.1.

If you do not have the original POF file, Altera provides a utility that converts your single-device chain EPC4, EPC8, and EPC16 Jam STAPL, JBC, or SVF programming file to the POF format. Subsequently, that POF file must be taken to the Quartus II programmer version 5.1 and later for conversion into Jam STAPL, JBC, or SVF files. The utility is available with this white paper.

For multi-device chain Jam STAPL, JBC, or SVF files that do not have the original POF files, contact Altera Applications via MySupport on the Altera website. For more details, see the EPC Programming File Conversion Utility section of this document.

Table 4 summarizes the programming support for Intel flash memory-based EPC4, EPC8, and EPC16 devices.

Table 4. Programming Support For Intel Flash Memory-Based EPC4, EPC8 & EPC16 Devices

Programming File Type	Programming File Generation Support Availability	Device Programming Support Availability *(1)(2)*
POF	No regeneration needed *(3)*	Quartus II 5.1 and later
HEXOUT	No regeneration needed *(3)*	*(1)*
Jam STAPL	Quartus II software version 5.1 and later Backward compatible *(4)*	Quartus II software version 5.1 and later
JBC	Quartus II software version 5.1 and later Backward compatible *(4)*	Quartus II software version 5.1 and later
SVF	Quartus II software version 5.1 and later Backward compatible *(4)*	Not applicable

Notes:
(1) Contact the third-party vendor for support availability.
(2) The MAX+PLUS II software does not support the Intel-based EPC4, EPC8, and EPC16 devices. Programming will fail when using MAX+PLUS II software with these devices.
(3) POF and HEXOUT programming files generated with the Quartus II software version 5.0 SP1 and earlier or the MAX+PLUS II software can be used to program the Intel-based devices, the Micron flash-based EPC4 device, or the Sharp flash-based EPC8/EPC16 devices.
(4) Jam STAPL, JBC, and SVF programming files generated with the Quartus II software version 5.1 and later can be used to program the Intel-based devices, the Micron flash-based EPC4 device, or the Sharp flash-based EPC8/EPC16 devices.

EPC Programming File Conversion Utility

Altera provides a utility to convert single-device chain EPC4, EPC8, and EPC16 Jam, JBC, and SVF programming files to POF format. You can download the latest conversion utility from Altera's white paper web page.

To run this conversion utility, type the following command at a command prompt:

```
convert_epc_file <input filename> <output filename.pof>
```

The input filename should have a Jam, JBC, or SVF extension.

Conclusion

The enhanced configuration device is a single-chip configuration solution that contains 4-, 8-, and 16-Mbit flash memory for configuration data storage.

This white paper documents the differences between the Micron or Sharp flash memory against the Intel flash memory. This document helps designers to perform the migration from current Micron-based EPC4 device or Sharp-based EPC8 and EPC16 devices to the new Intel flash memory-based EPC4, EPC8, and EPC16 devices.

Additional Resources

- EPC Programming File Conversion Utility:
  www.altera.com/literature/wp/convert_epc_file.zip

- Use PCN 0506 along with the device date code information to differentiate the flash memory type in an EPC4, EPC8, or EPC16 device:
  www.altera.com/literature/pcn/pcn0506.pdf
- Intel Flash Memory (28F160B3) Data Sheet:
  http://www.intel.com/design/flcomp/datashts/290580.htm
- Altera Device Package Information Data Sheet:
  www.altera.com/literature/ds/dspkg.pdf
- Altera's MySupport web page:
  https://www.altera.com/mysupport/

101 Innovation Drive
San Jose, CA 95134
(408) 544-7000
http://www.altera.com

Copyright © 2005 Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.