APPLICATION NOTE

Designing Intel® StrataFlash™ Memory into Intel® Architecture

July 1998

Order Number: 292222-001

CONTENTS

PAGE

ABSTRACT **5**

1.0 INTRODUCTION **5**

2.0 INTERFACING INTEL® StrataFlash™ MEMORY TO THE ISA BUS **7**

2.1 Paging Intel® StrataFlash™ Memory into the Expansion ROM Area 7

2.2 Connecting the Intel® StrataFlash™ Memory as Linear Memory to ISA Bus 8

3.0 PCI BUS WITH AN ADD-IN CARD **9**

4.0 CONCLUSION **9**

APPENDIX A: Additional Information **10**

FIGURES

Figure 1. Flash Memory Interfaces into Intel® Architecture 6

Figure 2. Paging Intel® StrataFlash™ Memory into the Expansion ROM Area 7

Figure 3. Connecting Intel® StrataFlash™ Memory to the ISA Bus as Linear Memory 9

REVISION HISTORY

Date of Revision	Version	Description
07/10/98	-001	Original version

ABSTRACT

Intel® StrataFlash™ memory provides a low cost, high density memory solution for disk-less systems. One of the most common types uses Intel® microprocessors and chip-sets. Intel® Architecture provides a number of interfaces for flash memory. Depending on the design of the interface, the flash memory could store the BIOS, operating system, data parameters, or a combination of these. This paper will discuss a number of ways to interface Intel StrataFlash memory to Intel Architecture. Specifically, this paper will discuss interfacing Intel StrataFlash memory to the ISA bus and PCI bus.

1.0 INTRODUCTION

There are many ways to connect flash memory to Intel Architecture to provide a rugged, nonvolatile storage media. Figure 1 shows six ways to interface flash memory to Intel Architecture:

1. ISA bus where the flash memory is embedded on the circuit board.
2. ISA bus by means of an add-in card
3. ISA bus by means of a SIMM
4. PCI bus by means of an add-in card
5. ISA/PCI bus by means of the PCMCIA interface (PC card form-factor)

This paper will explore the ISA bus and PCI bus options for Intel StrataFlash memory.

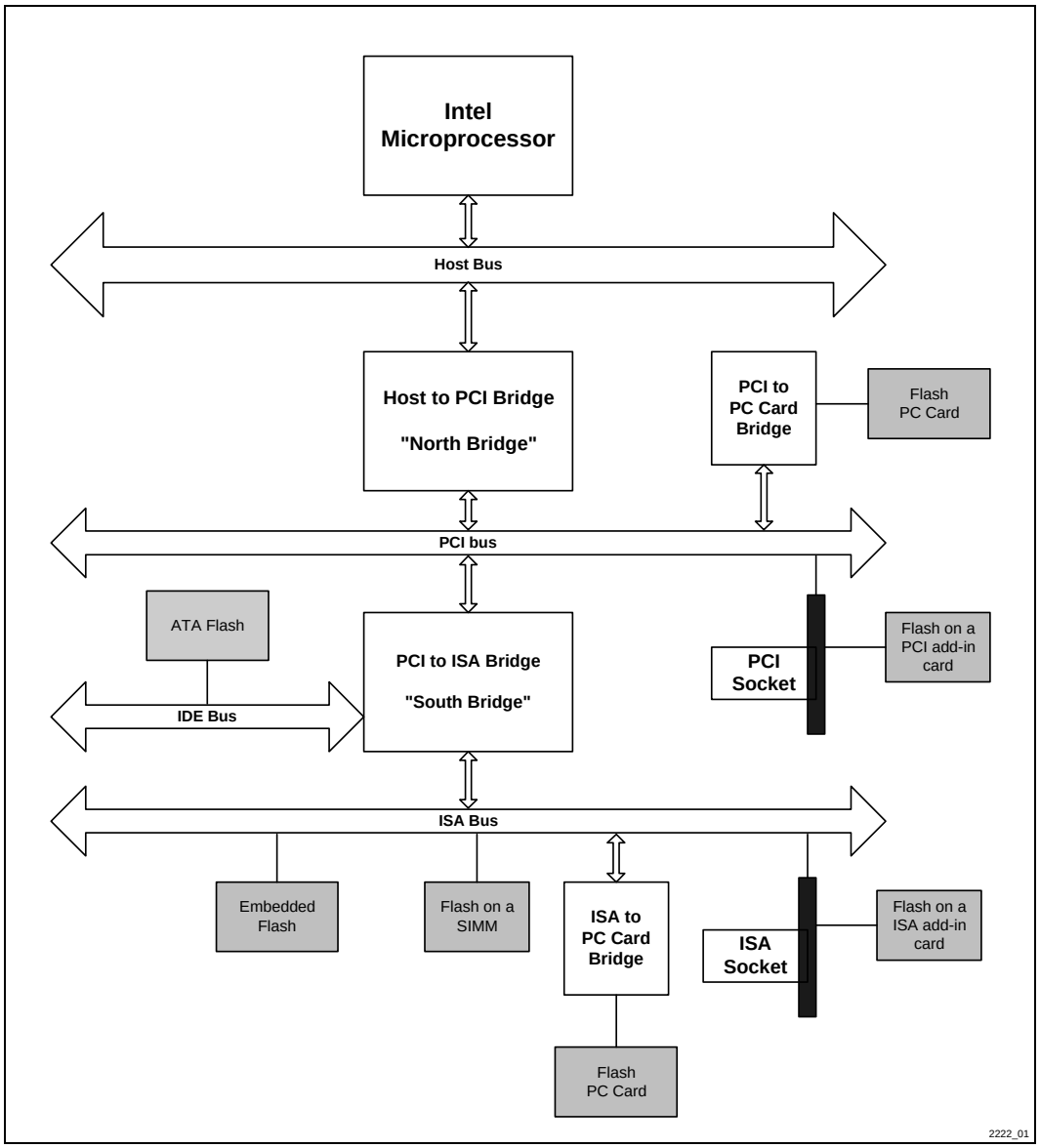

Figure 1. Flash Memory Interfaces into Intel® Architecture

2.0 INTERFACING INTEL® StrataFlash™ MEMORY TO THE ISA BUS

The method used to connect and control flash memory in Intel Architecture depends on the chipset and BIOS used to control the bus. There are two basic approaches: paging flash memory into the expansion ROM area or connecting flash memory as linear memory to the ISA bus.

2.1 Paging Intel® StrataFlash™ Memory into the Expansion ROM Area

Those systems that cannot access extended memory on the ISA bus can still use large arrays of flash. The ISA address range from 0x0C0000 to 0x0DFFFF was designated for expansion ROMs. These ROMs were included with expansion boards. Prior to booting off a disk, the BIOS scanned this address range to see if a device needed initialization or could be bootable.

This technique breaks the flash up into 16-KB to 128-KB pages and locates them in this area. It includes a page register, but some chipsets have discrete I/O outputs that can be used for this purpose. Figure 2 shows one implementation of this technique. If a page register is used, it must be mapped into the system I/O space upon the boot sequence. The design in Figure 2 uses 128Kbyte pages because that matches the block size of the Intel StrataFlash memory. This is not necessary, but it makes it easier to manipulate the flash. When the RESETDRV signal is asserted, it will abort any read or write operations in progress.

Obviously, much of the complexity is implemented in the box labeled “Control Logic.” This will likely be a simple PLD that controls access to the page register (74x273) and generates the CE_0 # for the flash memory. It uses address inputs to select a location in the I/O space and generates a clock input to the page register to latch the page address. It also includes the logic to clear the page register upon a system reset.

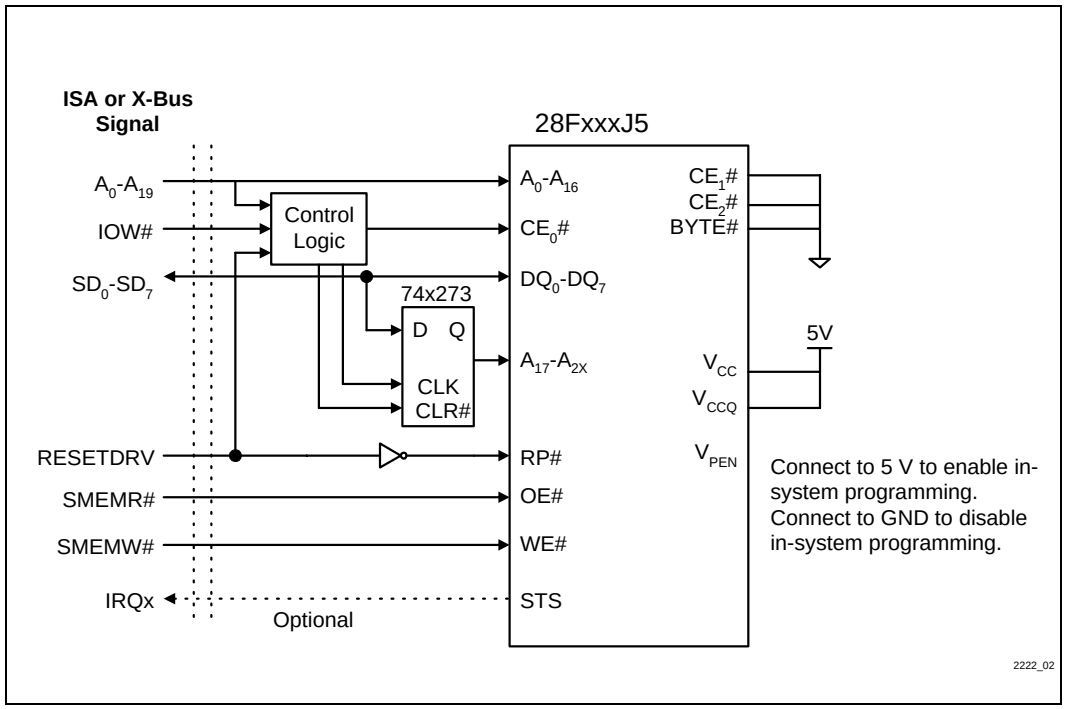

Figure 2. Paging Intel® StrataFlash™ Memory into the Expansion ROM Area

With a little additional functionality, the flash memory can also store the BIOS. When the system access 0x0E0000 through 0x0FFFFF(BIOS area), the PLD could clear the address register and enable the flash memory. This means that both 0x0C0000 through 0x0DFFFF and 0x0E0000 through 0x0FFFFF would be mapped into flash memory block 0 since clearing the latch would output all 0's to the flash memory address inputs A_{17} - A_{2x}. The upper 64 Kbytes of block 0 would be the boot code, the lower could be the expansion ROM code. After the BIOS has been shadowed, the system will no longer read from 0x0E0000 through 0x0FFFFF, so the latch will not be cleared.

When the system begins reading the expansion ROM area, it will see the lower 64 K in block 0. That area would have the code to copy the rest of the flash into DRAM.

2.2 Connecting the Intel® StrataFlash™ Memory as Linear Memory to ISA Bus

The ISA bus was designed to be a memory and I/O expansion bus, so ISA signals are compatible with Intel StrataFlash memory. Since current systems use PCI for expansion, the only thing on the ISA bus may be the flash memory. Therefore, there may be no need to buffer the ISA signals. This discussion assumes:

1. Intel StrataFlash memory is the only memory connected to the ISA bus
2. If there are I/O devices connected to the bus, they do not present a significant DC or Capacitive load to the flash memory.

Based on these assumptions, the flash will be connected directly the bus.

With the exception of the high order address and RESET signals, all ISA signal are functionally and parametrically compatible with Intel StrataFlash memory. Both are TTL compatible. Intel StrataFlash memory meets 16-bit memory timings. The discrepancies are:

- The ISA RESETDRV signal is active high, while the flash memory RP# input is active low.
- The high order addresses LA_{17}–LA_{23} are not stable throughout an entire memory operation and must be latched. There is a signal on the ISA, BALE, that can be used for this purpose.
- In order to simplify the interface, only 16-bit memory accesses are supported. The flash memory software driver must perform only word reads and writes.
- Finally, if no buffering separates the X-bus from ISA, and if there is a separate BIOS component, some decode logic may be needed to prevent contention between the size of the Intel StrataFlash memory and the BIOS.

Figure 3 shows a typical connection between the flash memory and the ISA bus.

This technique will work as long as the system directs extended memory (0x100000 to 0xFFFFF) operations to ISA. It has the benefit that the data stored in flash memory is always available, so seldom used data can reside in flash memory and be accessed as needed. However, typical PC chipset/BIOS combinations direct these extended memory accesses to DRAM. This technique also limits the size of the flash memory to 16 Mbytes. The flash memory is linearly mapped above the "top-of-memory" location within the system's memory map. The ISA bus would claim the flash memory access since the PCI bus did not and then pass it on to the flash memory. The ISA memory limit of 16 Mbytes is what limits the flash memory.

The scenarios previously described for interfacing Intel StrataFlash memory to the ISA bus can be implemented in the form of add-in cards, SIMMs, or components embedded directly on the circuit board.

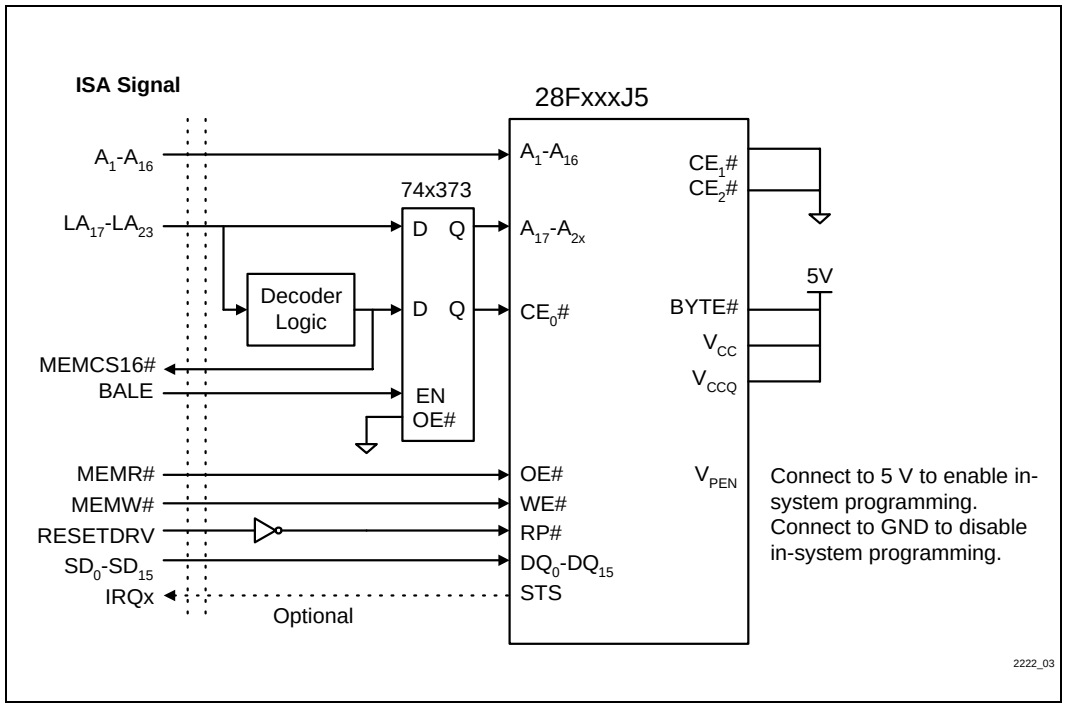

Figure 3. Connecting Intel® StrataFlash™ Memory to the ISA Bus as Linear Memory

3.0 PCI BUS WITH AN ADD-IN CARD

There already exists an application note that discusses how to implement a flash memory PCI add-in card. This is application note is *AP-758 Flash Memory PCI Add-In Card for Embedded Systems* (order number 273121). This design considers the 28F0xxS5 byte-wide flash memory. Instead of using these components, the design can be adapted to use Intel StrataFlash memory. This would result in fewer components on the board as well as enabling higher densities. This PCI add-in card designs uses a FPGA on the card to form a PCI to linear flash memory bridge. This FPGA considers all the requirements that are necessary for the add-in card to be PCI-compliant. VHDL code is available for the FPGA; however, it must be modified to work with Intel® StrataFlash memory.

It should be noted that the programming commands and algorithm of the 28F0xxS5 is slightly different than the 28Fxx0J5. The main difference is that the 28Fxx0J5 uses a 32-byte write buffer for programming and the 28F0xxS5 does not. The 32-byte write buffer requires a different programming algorithm and some additional commands. See the Intel StrataFlash memory datasheet for the programming algorithm and commands.

4.0 CONCLUSION

Intel StrataFlash memory provides a high density, low cost mass storage solution. The two optimal ways to interface Intel StrataFlash memory to Intel Architecture is by either paging it into the expansion ROM area via the ISA bus or interfacing it to the PCI bus via a PCI to linear flash bridge component.

APPENDIX A
ADDITIONAL INFORMATION

Order Number	Document/Tool
210830	*Flash Memory Databook*
290606	*Intel® StrataFlash™ Memory Technology 32 and 64 Mbit* datasheet
273121	*AP-758 Flash Memory PCI Add-In Card for Embedded Systems*
292204	*Common Flash Interface (CFI) and Command Sets*

NOTES:

1. Please call the Intel Literature Center at (800) 548-4725 to request Intel documentation. International customers should contact their local Intel or distribution sales office.
2. Visit Intel's World Wide Web home page at http://www.Intel.com for technical documentation and tools.