21285 Core Logic for SA-110 Microprocessor

Specification Update

September 1999

Notice: The SA-110 may contain design defects or errors known as errata. Characterized errata that may cause the SA-110's behavior to deviate from published specifications are documented in this specification update.

Order Number: 278223-003

Information in this document is provided in connection with Intel products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The SA-110 may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an ordering number and are referenced in this document, or other Intel literature may be obtained by calling 1-800-548-4725 or by visiting Intel's website at http://www.intel.com.

Contents

Revision History

Date	Version	Description
10/17/99	003	The 21285AB product data sheet incorrectly references this part’s support of synchronous parity mode for StrongARM™ processors. Since the necessary support for this mode in the SA-110 does not exist, the 21285AB cannot support synchronous parity mode in StrongARM™ processors. Added documentation changes 2 and 3 that describe the operation of bit 13 of the SDRAM timing register.
10/15/98	002	Added documentation change 1. See Summary Table of Changes for list.
07/14/98	001	This is the new Specification Update document. It contains all identified errata published prior to this date.

Preface

As of July, 1996, Intel's Computing Enhancement Group has consolidated available historical device and documentation errata into this new document type called the Specification Update. We have endeavored to include all documented errata in the consolidation process, however, we make no representations or warranties concerning the completeness of the Specification Update.

This document is an update to the specifications contained in the Affected Documents/Related Documents table below. This document is a compilation of device and documentation errata, specification clarifications and changes. It is intended for hardware system manufacturers and software developers of applications, operating systems, or tools.

Information types defined in Nomenclature are consolidated into the specification update and are no longer published in other documents.

This document may also contain information that was not previously published.

Affected Documents/Related Documents

Title	Order
21285 Core Logic for SA-110 Microprocessor Data Sheet	278115-001

Nomenclature

Errata are design defects or errors. These may cause the Product Name's behavior to deviate from published specifications. Hardware and software designed to be used with any given stepping must assume that all errata documented for that stepping are present on all devices.

Specification Changes are modifications to the current published specifications. These changes will be incorporated in any new release of the specification.

Specification Clarifications describe a specification in greater detail or further highlight a specification's impact to a complex design situation. These clarifications will be incorporated in any new release of the specification.

Documentation Changes include typos, errors, or omissions from the current published specifications. These will be incorporated in any new release of the specification.

Note: Errata remain in the specification update throughout the product's lifecycle, or until a particular stepping is no longer commercially available. Under these circumstances, errata removed from the specification update are archived and available upon request. Specification changes, specification clarifications and documentation changes are removed from the specification update when the appropriate changes are made to the appropriate product specification or user documentation (datasheets, manuals, etc.).

Summary Table of Changes

The following table indicates the errata, specification changes, specification clarifications, or documentation changes which apply to the Product Name product. Intel may fix some of the errata in a future stepping of the component, and account for the other outstanding issues through documentation or specification changes as noted. This table uses the following notations:

Codes Used in Summary Table

Stepping

X: Errata exists in the stepping indicated. Specification Change or Clarification that applies to this stepping.

(No mark)

or (Blank box): This erratum is fixed in listed stepping or specification change does not apply to listed stepping.

Page

(Page): Page location of item in this document.

Status

Doc: Document change or update will be implemented.

Fix: This erratum is intended to be fixed in a future step of the component.

Fixed: This erratum has been previously fixed.

NoFix: There are no plans to fix this erratum.

Eval: Plans to fix this erratum are under evaluation.

Row

| Change bar to left of table row indicates this erratum is either new or modified from the previous version of the document.

Errata

No.	Steppings			Page	Status	ERRATA
	S	#	#			
1	X			12	NoFix	Register write from StrongARM™ processor is not ordered with PCI access.
2	X			9	Eval	CSR memory base address register may require special setup.
3	X			10	NoFix	Outbound write flush address region is not supported.

Specification Changes

No.	Steppings		Page	Status	SPECIFICATION CHANGES
	#	#			
					None for this revision of this specification update.

Specification Clarifications

No.	Steppings			Page	Status	SPECIFICATION CLARIFICATIONS
	#	#	#			
						None for this revision of this specification update.

Documentation Changes

No.	Document Revision	Page	Status	DOCUMENTATION CHANGES
1	278115-001	13	Doc	Section 7.3.25, Page 7-43, Table.
2	278115-001	14	Doc	Section 7.3.12, Page 7-32, Table
3	278115-001	14	Doc	Section 4.1.3, Page 4-4, Text

Identification Information

Markings

21285 Revision E

- This document contains the errata for the 21285-AB core logic for the SA-110 microprocessor. The 21285-AB device revision affected by these errata can be identified as order number 21285-AB Rev. E and is also labeled as DC1065E.
- These errata describe problems associated with this chip or its documentation and offer workarounds, where available, that allow system designers and driver developers to use the 21285-AB successfully.
- Future revisions, if required, will increment the last letter (e.g., DC1065*F*).
- The 21285-AB can be identified in a PCI system by reading the REV_ID value of 04h or higher, and is identifiable on the packaging as DC1065E.
- While Intel believes the information included in this publication to be correct as of the date of publication, it is subject to change without notice.

Related Information

None for this revision of the specification update.

Errata

1. Register Write from StrongARM™ Processor is Not Ordered with PCI Access

Problem: This problem exists for parts with REV_ID 4 or lower. There are four CSRs that control various aspects of StrongARM™ processor to PCI accesses:

1. PCI Address Extension – offset 140h
2. Pre-fetchable Memory Range – offset 144h
3. ARM DAC Address – offsets 200h and 204h
4. DAC Pre-fetch Control – offset 208h

Note: ARM DAC Address and DAC Prefetch Control only exist in chips with REV_ID 4 or higher.

Implication: A typical program sequence is comprised of a store to one of the registers to modify a parameter, followed by a load or a store to PCI space. Ideally, the PCI access should use the new value in the register. However, under some specific timing condition (listed in the sequence that follows), the PCI access occurs before the register write, and then uses the old value in the register in controlling the PCI access.

For this to happen, the following sequence of events must occur with a specific timing pattern:

1. Host processor write (via PCI bus) to a 21285 CSR
2. StrongARM™ processor write to one of the 21285 CSRs listed above.
3. DMA channel read of a descriptor from SDRAM.
4. StrongARM™ processor read or write to PCI.

Workaround: There are four possible workarounds. Numbers 2, 3, and 4 might already be in place, depending on the nature of the application. In a case where neither 2, 3, or 4 applies, workaround number 1 is required.

1. Any writes to the four CSRs listed in the problem statement should be followed by another register access prior to the PCI access. Accessing a register guarantees that the register write occurs before the PCI access. (The second register access could consist of reading the register just written, writing the register a second time with the same data, or reading or writing any other register). The second register access does not need to be done for any registers other than the four CSRs listed in the problem statement.
2. If the application does not use DMA channels, then no workaround is needed.

Note: The I_2O message unit at PCI offsets 0x40h and 0x44h is *not* considered a CSR for this purpose.

3. If the host processor does not perform writes to 21285 CSRs, then no workaround is needed.
4. If the DMA channels are programmed via register mode, rather than descriptor list mode, the problem will not occur. In register mode the DMA transfer parameters are written directly into DMA registers rather than into local memory to be fetched by the channel. See Section 6.2.1 and 7.3.5 of the 21285 Data Sheet

Status: NoFix.

2. CSR Memory Base Address Register may Require Special Setup.

Problem: This problem exists for parts with REV_ID 4 or lower. This problem exists for systems which have PCI masters which must access SDRAM memory through the CSR memory base address register (Offset 10h) with the PCI memory read multiple command. Memory read multiple accesses from PCI to SDRAM memory are done as delayed reads. The initial SDRAM memory address is generated by using the CSR base address mask register and the CSR base address offset register. When the PCI master has repeated the read command and the 21285 has detected that the master has consumed the 17th Dword, the 21285 will start streaming by prefetching another 16 Dwords. The prefetch operation requires that another SDRAM address be generated. The 21285 incorrectly uses the SDRAM base address mask register (offset 100h) and the SDRAM base address offset register (offset 104h) rather than the CSR BAR mask and offset registers to generate the prefetch SDRAM address.

Implication: The 21285 incorrectly uses the SDRAM base address mask register (offset 100h) and the SDRAM base address offset register (offset 104h) rather than the CSR BAR mask and offset registers to generate the prefetch SDRAM address.

Workaround: The problem will ***not*** occur if any one of the following conditions are met:

1. The CSR memory base address register (offset 10h) is not used to access SDRAM memory with PCI memory read multiple commands. Note that accesses through the SDRAM base address register (offset 18h) work properly.
2. The PCI bus masters, which generate memory read multiples through the CSR memory BAR, always read 32 Dwords (128 bytes) or less. The first 32 Dwords read from SDRAM will be correct.
3. If the application must use the CSR memory BAR to access SDRAM memory, the SDRAM base address mask and offset registers can be programmed to exactly match the values in the CSR memory base address mask and offset registers. This ensures that the prefetch SDRAM address is generated correctly.

Status: Eval.

3. Outbound Write Flush Address Region is not Supported

Problem: This problem exists for parts with REV_ID 4 or lower. The outbound write flush address region is not supported.

Implication: StrongARM™ processor accesses to the address region from 78000000h through 78ffffffh may not work properly. In some cases, the 21285 could cause StrongARM™ processor to hang.

Workaround: There are no workarounds for this problem.

Status: NoFix.

Specification Changes

None for this revision of this specification update.

Specification Clarifications

None for this revision of this specification update.

Documentation Changes

1. Section 7.3.25, Page 7-43, X-Bus I/O Strobe Mask Register—Offset 14CH.

The values in the definition of I/O device *n* strobe mask assert, deassert are interchanged. The correct definition is:

Dword Bit	Name	R/W	Description
7:0	I/O device 0 strobe mask	R/W	Strobe mask shifted out to **xior_l** or **xiow_l** (for read or write respectively) during an access to X-Bus device in **xcs_l[0]** range. When 0: Strobe asserts. When 1: Strobe deasserts. Reset value: 0.
15:8	I/O device 1 strobe mask	R/W	Strobe mask shifted out to **xior_l** or **xiow_l** during an access to X-Bus device in **xcs_l[1]** range. When 0: Strobe asserts. When 1: Strobe deasserts. Reset value: 0.
23:16	I/O device 2 strobe mask	R/W	Strobe mask shifted out to **xior_l** or **xiow_l** during an access to X-Bus device in **xcs_l[2]** range. When 0: Strobe asserts. When 1: Strobe deasserts. Reset value: 0.
31:24	I/O device *n* strobe mask	R/W	Strobe mask shifted out to **xior_l** or **xiow_l** during an access to X-Bus device in range. When 0: Strobe asserts. When1: Strobe deasserts. Reset value: 0.

2. Section 7.3.12, Page 7-32, SDRAM Timing Register—Offset 10Ch

Change the bit 13 description that reads:

Dword Bit	Name	R/W	Description
13	SA-110 Prime	R/W	Indicates that the SA-110 chip will drive parity information when it is performing a write. This bit should not be written to 1 if Parity Enable (bit [12] of this register) is not a 1. Reset value: 0.

To read:

Dword Bit	Name	R/W	Description
13	SA-110 Prime	R/W	Indicates that the SA-110 chip will drive parity information when it is performing a write. **NOTE: Do not write a 1 to this bit. The required support in "SA-110 Prime" does not exist, so this bit should never be written to 1.** Reset value: 0.

3. Section 4.1.3, Page 4-8, Parity

Change the 3rd paragraph that reads:

During a write from the SA-110, the operation depends on bit [13] of the SDRAM timing register.

- If a 0, data is received by the 21285 and flows through an internal parity generator onto **parity**. Therefore, it is necessary to run a parity-enabled memory subsystem slower than a comparable parity-disabled memory to allow for the parity computation.
- If a 1, the SA-110 provides the parity information for the write. The 21285 leaves its parity pins tristate.

To read:

During a write from the SA-110, the operation depends on bit[13] of the SDRAM timing register. Since the bit can never be written to 1, data, when received by the 21285 always flows through an internal parity generator onto **parity**. Therefore, it is necessary to run a parity-enabled memory subsystem slower than a comparable parity-disabled memory to allow for the parity computation.

Support, Products, and Documentation

If you need general information or support call **1-800-628-8686** or visit the Intel’s website at:

http://www.intel.com

Copies of documents that have an ordering number and are referenced in this document, or other Intel literature may be obtained by calling **1-800-548-4725** or by visiting Intel’s website for developers at:

http://developer.intel.com

ARM POWERED™

StrongARM