intel.

80C186 AND 80C188 EMBEDDED MICROPROCESSORS

SPECIFICATION UPDATE

Release Date: July, 1996

Order Number 272894-001

The 80C186 and 80C188 Embedded Microprocessors may contain design defects or errors known as errata. Characterized errata that may cause the 80C186 and 80C188 Embedded Microprocessor's behavior to deviate from published specifications are documented in this specification update.

CONTENTS

REVISION HISTORY

Date of Revision	Version	Description
07/01/96	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 Semiconductor Products 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 is the first release of the 80C186 and 80C188 Specification Update. 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 additional information that was not previously published.

Affected Documents/Related Documents

Title	Order
80C186/188 80C186XL/188XL Microprocessor User's Manual	272164-003

Nomenclature

Errata are design defects or errors. These may cause the published (component, board, system) behavior to deviate from published specifications. Hardware and software designed to be used with any component, board, and system must consider all errata documented.

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 changes 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 80C186 and 80C188 product. The 80C186 and 80C188 devices have an A and B stepping. All information applies to both the A and B steppings of the part. 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

Steps

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

I	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
	C186	C188	#			
9600001	X	X		7	NoFix	Non-Contiguous Interrupt Acknowledge Cycles
9600002	X	X		7	NoFix	ERROR# Processing During FWAIT Instructions
9600003	X	X		9	NoFix	Interrupt Status Register
9600004	X	X		9	NoFix	Bus Preemption Bug
9600005		X		10	NoFix	80C188 RFSH# Pin Errata

Specification Changes

No.	Steppings			Page	Status	SPECIFICATION CHANGES
	C186	C188	#			
001	X	X		11	NoFix	Input High Voltage Requirements On SRDY And ARDY

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
				None for this revision of this specification update

IDENTIFICATION INFORMATION

Markings

Product identifier on part:

80C186

80C188

intel®

ERRATA

9600001.   *Non-Contiguous Interrupt Acknowledge Cycles*

PROBLEM: A set of conditions exist that prevent the 80C186 from correctly issuing two back-to-back interrupt acknowledge cycles in response to an interrupt input. In systems using the 80130 master interrupt controller, the failure can be noticeable. The conditions required to cause the failure are as follows:

1) The 80C186 integrated interrupt controller is configured in SLAVE

    (iRMX) mode

2) An 80130 is used as the master interrupt controller

3) The interrupt acknowledge cycles are preceded by a LOCKed instruction

4) DMA or HOLD activity is active.

IMPLICATION: If an interrupt arrives during a LOCKed bus cycle, a loss of synchronization can occur and LOCK# may not be asserted between the 1st and 2Nd INTA# pulses. Without LOCK# being active, the BIU will give up the bus (after the first INTA#) in response to a DMA request or HOLD. Some devices, such as the 80130, will operate incorrectly if INTA bus cycles are not run contiguously. The problem typically results in an incorrect vector being read by the 80C186, thus program execution is disrupted. This problem has not been detected when using a device such as the 8259A.

WORKAROUND: Block DMA or HOLD requests between INTA# cycles with the circuitry shown in Figure 61 of AP-186, March 1987 printing (210973-004).

STATUS: No Fix. Refer to Summary Table of Changes to determine the affected stepping(s).

9600002.   *ERROR# Processing During FWAIT Instructions*

PROBLEM: During the execution of a FWAIT instruction, the 80C186 does not test the ERROR# input pin.

IMPLICATION: This presents a problem when the FWAIT instruction is used to suspend program execution so that the result of the previous numerics instruction can be used immediately. Since FWAIT does not check for errors, the error may not be detected until the next numerics instruction is executed.

This is unlike the operation of other microprocessors, such as the 80286 and 80386, who check for ERROR# during FWAIT instructions. If an error is detected, an interrupt type 16 is automatically generated. Doing this allows for normal error processing. It should be noted that the problem is only with the FWAIT instruction, other numerics instructions correctly test for ERROR# prior to execution, and if present, generate an interrupt type 16.

WORKAROUND: At this time, there is no one solution to the FWAIT ERROR# test problem. High-level languages (especially those written for the 80186/8087), often insert an FWAIT instruction to control program flow. In the case where one numerics instruction is followed by another, the occurrence of ERROR# will be correctly detected (the next numerics instruction will detect the error). However, if program flow requires the result of a previous numerics instruction to terminate correctly, then the execution of an FWAIT instruction will not function correctly. There are, however, several ways to get around this.

1) Follow an FWAIT instruction by an FNOP instruction will allow the error to be detected, since FNOP does check the ERROR# input prior to execution. If an error did occur, when the FNOP instruction is executed by the 80C186, and interrupt type 16 will be generated. This solution, however, is not built into our high-level language compilers.

2) A circuit can be constructed to force an interrupt during the execution of an FWAIT instruction when an error occurs. However, since there are no numerics chips available as of yet, this solution has not been tested and verified. Simply, the circuit would force BUSY to remain high when the ERROR# signal is activated (normally, BUSY will go low 3 clocks after ERROR# is asserted). Then, by routing an inverted ERROR# signal to one of the interrupt input lines, an interrupt will be generated and detected during the execution of the FWAIT instruction (interrupts are serviced while waiting on BUSY). The reason why BUSY is forced high is to allow enough time for the 80C186 to detect and respond to the interrupt while still executing the FWAIT instruction. Otherwise, the interrupt may not be detected at the right point to correctly identify the error. An OR gate can be used to force a busy indication based on BUSY or ERROR (inverted ERROR#).

STATUS: No Fix. Refer to Summary Table of Changes to determine the affected stepping(s).

9600003.   Interrupt Status Register

PROBLEM: A timer interrupt request occurring during a write operation to the register may be ignored or redirected to the wrong interrupt vector. All instructions capable of affecting the register are implicated: OUT, MOV, AND, OR, etc. The problem affects all 80186 NMOS and CMOS devices.

IMPLICATION: The upper bit in the interrupt request register is the DHLT bit. Its main purpose is to provide a means for the non-maskable interrupt to stop DMA. The lowest 3 bits indicate timer interrupt requests from each of the integrated timer/counters. If a write of 8000H or 0000H is made to the register to turn DMA activity on or off while one of the timer interrupt bits is changing, the expected interrupt response may not occur, e.g., an expected Timer 0 interrupt may incorrectly use a Timer 2 vector, or a Timer 2 interrupt may be missed entirely.

WORKAROUND: Any scheme which avoids writing the interrupt status register while timer interrupts are active:

1) Disable DMA by using the individual DMA control registers instead of the DHLT bit.

2) Temporarily disable all interrupts while altering the Interrupt Status Register using the CLI instruction.

3) Temporarily mask off timer interrupts through the timer control register, the mask register, or the priority mask register.

STATUS: No Fix. Refer to Summary Table of Changes to determine the affected stepping(s).

9600004.   Bus Preemption Bug

PROBLEM: An internal conflict between the HOLD/HLDA protocol and the DRAM refresh unit can lock up the 80C186/80C188 bus controller in many applications. This problem is a logic bug and affects all speeds.

There are three necessary conditions:

1) 80C186 HOLD in progress

2) Pending non-pipelined effective address (EA) calculation

3) Pending DRAM refresh cycle

IMPLICATION: The processor has a provision to regain the bus from an auxiliary master in order to run refresh cycles which depends on the temporary deassertion or HOLD by the auxiliary master. Under ordinary circumstances, HOLD must be deasserted only one clock in order for the refresh cycle to take place.

When a non-pipelined EA calculation is also pending, the usual priority scheme in the bus controller is disturbed. This disturbance gives the EA calculation and its associated data cycles precedence over refresh. HOLD must be deasserted additional clocks (well into the data cycles) to allow the refresh cycle to start, or else the bus controller will lock up, never running the refresh cycle and never returning HLDA to the auxiliary master.

WORKAROUND: The general workaround technique is to delay HOLD reassertion until at least two clocks before the T4 state of the last possible data cycle. That moment occurs in T2 unless there are wait states in the last data cycle; delay HOLD reassertion an additional clock for each wait state. Note that there can be as many as two data cycles preceding the refresh cycle. Details of the bus preemption problem and example workarounds may be found in Tech Bit MIC-02, January 27, 1989.

STATUS: No Fix. Refer to Summary Table of Changes to determine the affected stepping(s).

9600005.   80C188 RFSH# Pin Errata

PROBLEM: The RFSH# pin (#64) goes active and inactive on T4 states rather than T1 states as indicated in the data sheet. BHE#, the same pin on the 80C186, correctly changes states in T1.

IMPLICATION: RFSH# is only active during DRAM refresh cycles, so no fix is necessary unless the DRAM refresh unit is activated.

WORKAROUND: In most 80C188 DRAM controller applications, workarounds are not necessary because RFSH# is suitably qualified with other control signals such as latched addresses, chip selects, or S2:0#. If a workaround is necessary, it should be possible to simply delay RFSH# one clock.

STATUS: No Fix. Refer to Summary Table of Changes to determine the affected stepping(s).

SPECIFICATION CHANGES

001.   Input High Voltage Requirements On SRDY And ARDY

PROBLEM: The minimum Vih specification for ARDY and SRDY are higher than those for other pins.

IMPLICATION: As a result, there is less noise margin when interfacing to TTL devices at low voltage. The voltage specifications for these two pins is .2VCC + 1.1 volts, which equates to 2.0 volts at 4.5VCC and 2.2 volts at 5.5VCC. Normal Vih is .2VCC + .9 volts.

WORKAROUND: Pull-up resistors should be added to these pins when interfacing to TTL logic if timing is critical, or if there is a risk of wide VCC variation on the board (worst case is TTL device operating at 4.5 volts and 80C186 operating at 5.5 volts). When interfacing to CMOS logic, there should be no problems due to an increased minimum Voh specification of CMOS devices.

STATUS: There are no plans to correct this. Refer to Summary Table of Changes to determine the affected stepping(s).

SPECIFICATION CLARIFICATIONS

None for this revision of this specification update

DOCUMENTATION CHANGES

None for this revision of this specification update