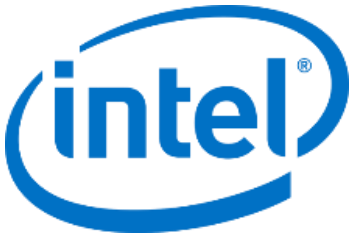

Intel® Pentium® Processor

Invalid Instruction Erratum Overview

Document Consolidation date: August 4, 2015

Legal Disclaimer

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at Intel.com, or from the OEM or retailer.

No computer system can be absolutely secure. Intel does not assume any liability for lost or stolen data or systems or any damages resulting from such losses.

The products described 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.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at intel.com, or from the OEM or retailer.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or visit www.intel.com/design/literature.htm.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.
*Other names and brands may be claimed as the property of others.

Contents

Intel® Pentium® Processor 1

Erratum Overview 4

Technical overview 4

Workaround overview 5

Erratum Technical Description 5

Problem: 5

Implication: 5

Workaround: 6

Software Backgrounder / Workaround for "Invalid Operand with Locked CMPXCHG8B Instruction Erratum (v0.9)" 9

The LOCK Prefix 9

The Erratum 9

The Workarounds 10

Software vendor statements 10

Berkeley Software Design, Inc. 10

IBM 11

Linux 11

Microsoft 11

NCR 12

Novell 12

SCO 12

Sequent 12

SunSoft 12

Unisys 13

How To Contact Us 13

In South & Latin America: 13

In Japan: 13

In Asia-Pacific: 13

In Australia: 14

In Africa, Europe, & Middle East: 14

This applies to: 14

Intel identifies workaround for the "Invalid operand with locked compare exchange 8byte (CMPXCHG8B) instruction" erratum

Erratum Overview

On Friday, November 7th 1997, a number of reports were posted to the Internet implying the possibility of a new erratum on the Pentium® processors and Pentium® processors with MMX™ technology. An erratum is a design defect or error which may cause a product to deviate from published specifications. Based on the Internet reports our engineering team quickly jumped on this issue. Once we were able to reproduce the behavior we confirmed that an erratum does exist which is now named the "Invalid Operand with Locked CMPXCHG8B instruction" erratum. We were also able to identify the following:

- The "Invalid operand with locked CMPXCHG8B instruction" erratum affects the Intel® Pentium® Processor, Pentium® Processor with MMX™ Technology, Pentium OverDrive® Processor and Pentium OverDrive processors with MMX Technology.
- It does not affect the Intel® Pentium® Pro Processor, Pentium® II Processor and Intel486™ and earlier processors.
- This invalid instruction is not in commercial software.
- The erratum only occurs when the processor receives a specific invalid instruction. The result of this erratum is the system may "freeze" and would have to be turned off and rebooted to return to normal operation.
- It is important to note that this erratum will only occur when someone has intentionally created this invalid instruction because they want to freeze the system.
- We have identified a workaround that prevents the system from being "frozen" by this invalid instruction and allows it to continue normal operation. The workaround modifies the execution flow to avoid the system hang after the invalid instruction is received. The workaround can be implemented through the operating system software.

Technical overview

The CMPXCHG8B instruction compares a 64-bit value from internal registers of the processor with a 64-bit value from memory (the destination). It is illegal to use a register as the destination. The result of the CMPXCHG8B instruction is a 64-bit value that will not fit into a 32-bit register. If a register is used as the destination, the processor normally stops execution of the CMPXCH8B instruction, signals this error condition and executes an error handler in software.

This erratum occurs if the CMPXCHG8B instruction is also locked (a special instruction to the processor to allow the completion of the CMPXCHG8B instruction without being interrupted), and an invalid register destination is used. In this case the processor signals the error condition but may not allow the error handler to begin due to the lock on the CMPXCH8B

instruction. As a result, the system hangs and the system must be re-booted to return to normal operation.

This issue does not cause data corruption or physical damage to a user's system. Any data saved to disk in the course of work remains on the disk and will be available for use when the system is re-booted.

The "Invalid Operand with Locked CMPXCHG8B Instruction" is erratum #81 on the Pentium processor errata list. For more information please see: Erratum Technical Description

Workaround overview

We have identified a workaround that can be implemented through the operating system. Basically, the workaround avoids the bus lock condition and allows the processor to execute the error handler. For the full technical description see: Workaround, in the Erratum Technical Description. Software vendors may also want to see the Software Backgrounder for more specific detail.

Intel has been working with industry operating system vendors to assist them in implementing this workaround for their operating systems. We will continue to work with them to implement the workaround in their operating systems. Users should contact their operating system vendor for specific availability of the workaround for that OS. A number of software vendors have already contributed statements with regard to this erratum. See: Software Vendor Statements

For more information see Intel Contacts for the phone number in your region.

Erratum Technical Description Updated Nov. 20 1997

Problem: The CMPXCHG8B instruction compares an 8 byte value in EDX and EAX with an 8 byte value in memory (the destination operand). The only valid destination operands for this instruction are memory operands. If the destination operand is a register the processor should generate an invalid opcode exception, execution of the CMPXCHG8B instruction should be halted and the processor should execute the invalid opcode exception handler. This erratum occurs if the LOCK prefix is used with the CMPXCHG8B instruction with an (invalid) register destination operand. In this case, the processor may not start execution of the invalid opcode exception handler because the bus is locked. This results in a system hang.

Implication: If an (invalid) register destination operand is used with the CMPXCHG8B instruction and the LOCK prefix, the system may hang. No memory data is corrupted and the user can perform a system reset to return to normal operation. Note that the specific invalid code sequence necessary for this erratum to occur is not normally generated in the course of programming nor is such a sequence generated by commercially available software.

This erratum only applies to Pentium® processors, Pentium processors with MMX™ technology, Pentium OverDrive® processors and Pentium OverDrive processors with MMX technology.

Pentium Pro processors, Pentium II Processors and i486™ and earlier processors are not affected.

Workaround: There are two workarounds for this erratum for protected mode operating systems. Both workarounds generate a page fault when the invalid opcode exception occurs. In both cases, the page fault will be serviced before the invalid opcode exception and thus prevent the lock condition from occurring. The implementation details will differ depending on the operating system. Use one of the following:

1. The first part of this workaround sets the first 7 entries (0-6) of the Interrupt Descriptor Table (IDT) in a non-writeable page. When the invalid opcode exception (exception 6) occurs due to the locked CMPXCHG8B instruction with an invalid register destination (and only then), the processor will generate a page fault if it does not have write access to the page containing entry 6 of the IDT. The second part of this workaround modifies the page fault handler to recognize and correctly dispatch the invalid opcode exceptions that are now routed through the page fault handler.

   Part I, IDT page access

   1. Mark the page containing the first seven entries (0-6) of the IDT as read only by setting bit 1 of the page table entry to zero. Also set CR0.WP (bit 16) to 1. Now when the invalid opcode exception occurs on the locked CMPXCHG8B instruction, the processor will trigger a page fault since it does not have write access to the page containing entry 6 of the IDT. This page fault prevents the bus lock condition and gives the OS complete control to process the invalid operand exception as appropriate. Note that exception 6 is the invalid opcode exception, so with this scheme an OS has complete control of any program executing an invalid CMPXCHG8B instruction.
   2. Optional: If updates to entries 7-255 of the IDT occur during the course of normal operation, page faults should be avoided on writes to these IDT entries. These page faults can be avoided by aligning the IDT across a 4KB page boundary such that the first seven entries (0-6) of the IDT are on the first read only page and the remaining entries are on a read/writeable page.

   Part II, Page Fault Handler Modifications

   - Modify the page fault handler to calculate which exception caused the page fault using the fault address in CR2. If the error code on the stack indicates the exception occurred from ring 0 and if the address corresponds to the invalid opcode exception, then pop the error code off the stack and jump to the invalid opcode exception handler. Otherwise continue with the normal page fault handler.

OR

2. This workaround has two parts. First, the Interrupt Descriptor Table (IDT) is aligned such that any invalid opcode exception will cause a page fault (due to the page not being present). Second, the page fault handler is modified to recognize and correctly dispatch the invalid opcode exception and certain other exceptions that are now routed through the page fault handler.

   Part I, IDT Alignment:

   a. Align the Interrupt Descriptor Table (IDT) such that it spans a 4KB page boundary by placing the first entry starting 56 bytes from the end of the first 4KB page. This places the first seven entries (0-6) on the first 4KB page, and the remaining entries on the second page.

   b. The page containing the first seven entries of the IDT must not have a mapping in the OS page tables. This will cause any of exceptions 0-6 to generate a page not present fault. A page fault prevents the bus lock condition and gives the OS complete control to process these exceptions as appropriate. Note that exception 6 is the invalid opcode exception, so with this scheme an OS has complete control of any program executing an invalid CMPXCHG8B instruction.

   Part II, Page Fault Handler Modifications:

   a. Recognize accesses to the first page of the IDT by testing the fault address in CR2. Page not present faults on other addresses can be processed normally.

   b. For page not present faults on the first page of the IDT, the OS must recognize and dispatch the exception which caused the page not present fault. Before proceeding, test the fault address in CR2 to determine if it is in the address range corresponding to exceptions 0-6.

   c. Calculate which exception caused the page not present fault from the fault address in CR2.

   d. Depending on the operating system, certain privilege level checks may be required, along with adjustments to the interrupt stack.

   e. Jump to the normal handler for the appropriate exception.

Both workarounds should only be implemented on Intel processors that return Family=5 via the CPUID instruction.

75/90/100/120/133/150/166/200 MHz Pentium® Processors and 120/133/150/166/200/233 MHz Pentium Processors with MMX™ Technology

No.	B1	B3	B5	C2	mA1	cB1	mcB1	cC0	mA4	mcC0	E0	xA3	mxA3	xB1	mxB1	myA0	ERRATA
81	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	Invalid operand with locked CMPXCHG8B instruction

60 MHz and 66 MHz Pentium® Processors

No.	B1	C1	D1	ERRATA
52	X	X	X	Invalid operand with locked CMPXCHG8B instruction

63/83/120/125/133/150/166 MHz Pentium® OverDrive® Processors and 125/150/166/180/200 MHz Pentium OverDrive Processors with MMX™ Technology

No.	B1	B2	C0	tA0	aC0	oxA3	oxB1	ERRATA
70	X	X	X	X	X	X	X	Invalid operand with locked CMPXCHG8B instruction

Software Backgrounder / Workaround for "Invalid Operand with Locked CMPXCHG8B Instruction Erratum (v0.9)"

Updated Nov. 20 1997

The LOCK Prefix

Some types of programs perform computations that require data accesses to have a specific ordering. These types of programs most commonly include operating systems, database engines, and applications for multiple processors. To ensure the ordering of data accesses, these programs use *synchronization*. Synchronization may be done by either software or hardware methods, but most programs use hardware synchronization for efficiency. Hardware synchronization usually involves reading and updating a memory location, with the hardware ensuring that the sequence is done in one operation. Such a combined operation is called a *locked* access. Intel processors support locked accesses by an instruction feature called the *lock prefix.* This feature tells the processor that an instruction that updates memory is to be processed as a locked access to memory.

The CMPXCHG8B Instruction

Beginning with the Pentium® processor, Intel processors have provided special hardware support for synchronization using the CMPXCHG8B (compare and exchange 8 bytes) instruction. This instruction compares a specified memory location with processor registers, and conditionally updates the 8-byte memory location. When used with the lock prefix, this instruction provides very flexible hardware support for synchronization.

The Erratum

In Pentium processors, Pentium processors with MMX™ technology, Pentium OverDrive® processors, and Pentium OverDrive processors with MMX technology there is an erratum that affects the lock prefix on a CMPXCHG8B instruction with a register destination. This erratum does not affect the Pentium® Pro processor, Pentium® II processor, or the Intel486™ and earlier processors. The documented use of the CMPXCHG8B instruction requires an 8-byte memory destination; attempting to use a CMPXCHG8B to update a 4-byte processor register is a program error. A computer?s operating system typically processes program errors through error handling routines. The erratum may cause an unexpected system freeze, preventing the program error from being processed by the error handling routine.

The affected form of the instruction is not contained in any operating system or other application known to Intel, nor is there any reasonable purpose for a software tool to generate it. Hence user software should not be affected. However, it is possible for a malicious program to use this instruction to cause a system freeze. The system freeze will not affect data that a user has already saved to disk. When the system is restarted all saved data will still be available.

The Workarounds

Intel has developed two workarounds for this erratum that can be incorporated by operating systems vendors. Both workarounds take advantage of the memory management support provided by Intel processors. The first workaround takes advantage of the fact that locating the error handling routine normally only involves a read from memory. The workaround marks the memory accessed by an affected instruction to prohibit writing. When an affected instruction is processed and the processor attempts to locate the error handling routine, it appears to be writing this memory. The attempt to write causes a *page fault*. While processing the page fault the program error is dispatched to the error handling routine, and the operating system continues normally.

The second workaround also relies upon a page fault being processed before the program error handling routine. The page fault prevents the memory bus lock caused by the lock prefix. The workaround marks the memory used to find the error handling routine as not present in memory. When an affected instruction is processed the processor attempts to read memory to find the error handling routine. Since that memory is marked not present, the processor is made to encounter a page fault. While processing the page fault the program error is dispatched to the error handling routine as expected. The operating system then continues normally.

If you are an operating system vendor and would like further information about the erratum or the workaround, call Intel Customer Support. Please identify yourself as an operating system vendor.

Software vendor statements

The following companies have issued statements on the status of their products with regard to the "Invalid Operand with Locked CMPXCHG8B Instruction" erratum.

BSDI
NCR
SunSoft
IBM
Novell
Unisys
Linux
SCO
Microsoft
Sequent

Berkeley Software Design, Inc.

"BSDI has worked closely with Intel since they contacted us about this erratum. We were able to develop a workaround for BSD/OS very quickly, and Intel's assistance was invaluable in this

process. BSDI is confident that the software workaround solves this problem for our customers. Patches for our current (3.0 and 3.1) releases, as well as for the previous release, are available for download now:
BSD/OS 3.1 and 3.0
ftp://ftp.bsdi.com/bsdi/patches/patches-3.1/M310-001

BSD/OS 2.1
ftp://ftp.bsdi.com/bsdi/patches/patches-2.1/K210-030

General information about BSDI can be found at http://www.bsdi.com or by calling 800-ITS-BSD8 (800-487-2738)."

Mike Karels, VP Engineering

IBM

In response to the invalid instruction erratum confirmed Monday (11/10/97) by Intel, IBM and Intel are working together to deliver a software workaround for OS/2 users to the processor erratum. The erratum, under certain user-definable conditions, can affect Intel Pentium® Processor and Pentium Processor with MMX™ Technology systems running any operating system, including IBM's OS/2. Once testing is completed, the workaround will be made available to OS/2 customers.

Linux

"We have been in discussion with the Intel engineering team with regard to the Pentium® Processor Invalid Instruction Erratum, including the software work-around solution. The software workaround is in place in kernel versions starting at 2.0.32 and 2.1.64, available for download at
ftp://ftp.kernel.org/pub/linux/kernel/
and various mirrors of the Linux kernel development site."

Linus Torvalds
Linux

Microsoft

"Microsoft has worked closely with Intel to understand and characterize the effects of the recently uncovered Pentium® processor erratum, and we're in the process of studying the implementation of potential workarounds in order to meet the needs of our customers," said Moshe Dunie Vice-President Windows Operating Systems Division at Microsoft. "Since this erratum can only be exploited by a program that was developed with malicious intent and deliberately uses this illegal instruction, following common-sense computing practices, such as not downloading or running executables from unknown sources, can protect a user from this problem."
For more information, please visit:
http://support.microsoft.com/support/kb/articles/q163/8/52.asp

NCR

NCR is working closely with Intel to determine the impact of this problem on our customers' systems and develop corrective procedures. Our goal is to minimize any disruption necessary to correct the problem. Any NCR customers who have questions on how this will affect their systems should call their customer support representatives through normal channels.

Novell

"Novell's network operating system NetWare/IntranetWare is not affected by the invalid instruction erratum found in the Pentium® processor. NetWare/IntranetWare requires proper authentication to run NLM's and applications on the server. Due to this secure access, NetWare/IntranetWare is not susceptible to NLM's or applications that would use the invalid opcode. For further information, please contact Novell at 1-801-861-5533 or www.novell.com."

Tom Oldroyd
Senior Marketing Manager
Novell Inc.

SCO

"SCO has been informed of the latest erratum for the Pentium® Processors by Intel. SCO and Intel are working together to determine the extent to which this erratum effects the SCO Operating Environments (SCO OpenServer and SCO UnixWare) and to develop any necessary fixes or supplements to resolve such issues and maintain the integrity of the SCO environments. SCO will release these as soon as they are available. For further information contact your SCO reseller or visit SCO's website at www.sco.com/support/."

Doug Michels
CTO and EVP, Products
SCO

Sequent

"Sequent is working closely with Intel to understand any issues raised by the Pentium® Processor erratum. Once we have full insight into these issues we will take appropriate measures to ensure the continued success of our customers."
Kevin Joyce
Director of Product Marketing
Sequent Computer Systems, Inc.

SunSoft

SunSoft has not yet received any eports from its customers of problems related to the Pentium® Processor Invalid Instruction Erratum. SunSoft is working closely with Intel to understand the problem and its impact, if any, to the Solaris operating environment. For an update and the list of available Solaris patches and their locations, please go to: http://sunsolve.sun.com/sunsolve/secbulletins/security-alert-161.txt

Unisys

"Unisys is actively working with Intel to incorporate the work-around for the Pentium® Processor Invalid Instruction erratum into our Operating Environments. We will provide updates to our customers as they are available. For further information, please contact Unisys at http://www.unisys.com or by phone at 215-986-4788."

Martin Krempasky
Dir of Public Relations.

How To Contact Us

In United States, Canada, Guam, Puerto Rico, and the Virgin Islands:

- For all Pentium® processor replacement inquiries call:
  1-800-628-8686 or 1-916-356-7599

In South & Latin America:

- Argentina: 01-345-2143 or 2145 then 210/341-3690-9949 then 1-210/341-3690
  Colombia 9-801/53796
  Brazil: 0008165500006
  Chile: 800/241-400 then 210/341-3690-9587# then 1 then 1-210/341-3690
  Jamaica/Caribbean: 1-800/533-5064 then 210/341-3690-5347 then 1-210/341-3690
  Mexico: 9-580/038-01523

In Japan:

- +81-298-47-1841--Outside Japan
  0120-868686 Japan Local

In Asia-Pacific:

- China
  --Beijing:.....86 10 6238 5130
  --Guangzhou:.....86 20 8332 3333
  --Mongolia:.....86 10 2385130
  --Shanghai:.....86 21 6485 2828
  Hong Kong:.....852 2844 4555
  India:.....91 80 555 0940
  Indonesia:.....6221 577 1930
  Malaysia:.....603 469 6677
  New Zealand:.....61 2 9937 5829
  Philippines:.....63 2 636 2191
  Singapore:.....65 735 3811
  South Korea:.....82 2 767 2500

Taiwan:.....886 2 514 4200 or 886 2 716 9660
Thailand:.....662 654 0654

In Australia:

- 61 2 9937 5829

In Africa, Europe, & Middle East:

Country	**Language**	**Number**
Deutschland	Deutsch	0800 181 89210
España	Español	900 99 4414
France	Français	0800 908179
Italia	Italiano	800 010475
United Kingdom	English	0800 374838
United States of America	English	1-800-628-8686
All Other Countries	English	+44 (0) 870 5673011
After 28 April 2001 "All Other Countries"		
number changes to:		**+44 (0) 870 5673011**

This applies to:

Intel® Pentium® Processor Intel® Pentium® Processor with MMX™ Technology	OverDrive® Processors