Addendum— Intel Architecture Software Developer’s Manual

Volume 3: System Programming Guide

Order Number: 243690-001

NOTE: The *Intel Architecture Software Developer’s Manual* consists of the following volumes: *Basic Architecture,* Order Number 243190; Addendum to the *Basic Architecture* (Order Number 243691); *Instruction Set Reference,* Order Number 243191; Addendum to the *Instruction Set Reference,* Order Number 243689; and the *System Programming Guide,* Order Number 243192.
Please refer to all of these volumes when evaluating your design needs.

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.

Intel's Intel Architecture processors (e.g., Pentium®, Pentium® Pro, Pentium® II, and Celeron™ processors) 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

TABLE OF CONTENTS

PAGE

CHAPTER 3
PROTECTED MODE MEMORY MANAGEMENT

CHAPTER 9
MEMORY CACHE CONTROL

CHAPTER 3
PROTECTED MODE MEMORY MANAGEMENT

The following information will be added to Chapter 3 of the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide*, in the next revision.

3.1. 36-BIT PAGE SIZE EXTENSION (PSE)

The 36-bit PSE extends 36-bit physical address support to 4 MB pages while maintaining a 4-byte page-directory entry. This approach provides a simple mechanism for operating system vendors to address physical memory above 4 GB without requiring major design changes, but has practical limitations with respect to demand paging.

The P6 family of processors' physical address extension (PAE) feature provides generic access to a 36-bit physical address space, but requires expansion of the page-directory and page-table entries to an 8-byte format (64 bit), and the addition of a page-directory-pointer table, resulting in another level of indirection to address translation.

For P6 family processors that support the 36-bit PSE feature, the virtual memory architecture is extended to support 4 MB page size granularity in combination with 36-bit physical addressing. Note that some P6 family processors do not support this feature. For information about determining a processor's feature support, refer to the following documents:

- AP-485, *Intel Processor Identification and the CPUID Instruction*
- *Addendum—Intel Architecture Software Developer's Manual, Volume 1: Basic Architecture*

For information about the virtual memory architecture features of P6 family processors, refer to Chapter 3 of the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide*.

3.1.1. Description of the 36-bit PSE Feature

The 36-bit PSE feature (PSE-36) is detected by an operating system by using the CPUID instruction. Specifically, the operating system executes the CPUID instruction with the value 1 in the EAX register and then determines support for the feature by inspecting bit 17 of the EDX register return value (see *Addendum—Intel Architecture Software Developer's Manual, Volume 1: Basic Architecture*). If the PSE-36 feature is supported, an operating system is permitted to utilize the feature, as well as use certain formerly reserved bits. To use the 36-bit PSE feature, the PSE flag must be enabled by the operating system (bit 4 of CR4). Note that a separate control bit in CR4 does not exist to regulate the use of 36-bit 4 MB pages, because this feature becomes the standard for 4 MB pages on processors that support it.

Table 3-1 shows the page size and physical address size obtained from various settings of the page-control flags for the P6 family processors that support the 36-bit PSE feature. Highlighted in bold is the change to this table resulting from the 36-bit PSE feature.

Table 3-1. Paging Modes and Physical Address Size

PG Flag (in CR0)	PAE Flag (in CR4)	PSE Flag (in CR4)	PS Flag (in the PDE)	Page Size	Physical Address Size
0	X	X	X	—	Paging Disabled
1	0	0	X	4 KB	32 bits
1	0	1	0	4 KB	32 bits
1	**0**	**1**	**1**	**4 MB**	**36 bits**
1	1	X	0	4 KB	36 bits
1	1	X	1	2 MB	36 bits

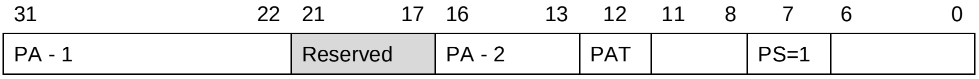
To use the 36-bit PSE feature, the PAE feature must be cleared (as indicated in Table 3-1). However, the 36-bit PSE in no way affects the PAE feature. Existing operating systems and software that use the PAE will continue to have compatible functionality and features with P6 family processors that support 36-bit PSE. Specifically, the Page-Directory Entry (PDE) format when PAE is enabled for 2 MB pages is exactly as depicted in Figure 3-21 of the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide*.

No matter which 36-bit addressing feature is used (PAE or 36-bit PSE), the linear address space of the processor remains at 32 bits. Applications must partition the address space of their work loads across multiple operating system processes to take advantage of the additional physical memory provided in the system.

The 36-bit PSE feature extends the PDE format of the Intel Architecture for 4 MB pages and 32-bit addresses by utilizing bits 16-13 (formerly reserved bits that were required to be zero) to extend the physical address without requiring an 8-byte page-directory entry. Therefore, with the 36-bit PSE feature, a page directory can contain up to 1024 entries, each pointing to a 4 MB page that can exist anywhere in the 36-bit physical address space of the processor.

Figure 3-1 shows the difference between PDE formats for 4 MB pages on P6 family processors that support the 36-bit PSE feature compared to P6 family processors that **do not** support the 36-bit PSE feature (i.e., 32-bit addressing).

Page Directory Entry format for processors that support 36-bit addressing for 4 MB pages



Page Directory Entry format for processors that support 32-bit addressing for 4 MB pages

31 22	21 12	11 8	7	6 0
Base Page Address	Reserved		PS=1	

NOTES:

1. PA-2 = Bits 35-32 of the base physical address for the 4 MB page (correspond to bits 16-13)

2. PA-1 = Bits 31-22 of the base physical address for the 4 MB page

3. PAT = Bit 12 used as the Most Significant Bit of the index into Page Attribute Table (PAT); see Section 10.2.

4. PS = Bit 7 is the Page Size Bit—indicates 4 MB page (must be set to 1)

5. Reserved = Bits 21-17 are reserved for future expansion

6. No change in format or meaning of bits 11-8 and 6-0; refer to the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide*, Figure 3-15, for details.

Figure 3-1. 36-Bit PSE Page-Directory Format

Figure 3-2 shows the linear address mapping to 4 MB pages when the 36-bit PSE is enabled. The base physical address of the 4 MB page is contained in the PDE. PA-2 (bits 13-16) is used to provide the upper four bits (bits 32-35) of the 36-bit physical address. PA-1 (bits 22-31) continues to provide the next ten bits (bits 22-31) of the physical address for the 4 MB page. The offset into the page is provided by the lower 22 bits of the linear address. This scheme eliminates the second level of indirection caused by the use of 4 KB page tables.

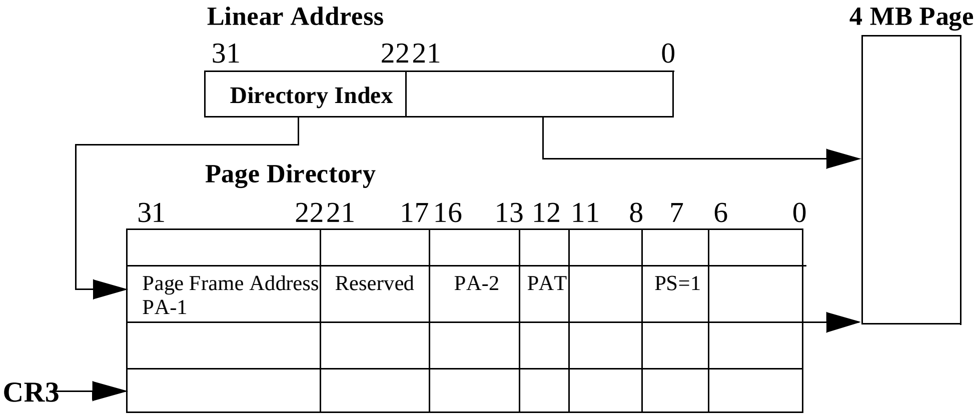


Figure 3-2. Page Size Extension Linear to Physical Translation

The PSE-36 feature is transparent to existing operating systems that utilize 4 MB pages because unused bits in PA-2 are currently enforced as zero by Intel processors. The feature requires 4 MB pages aligned on a 4 MB boundary and 4 MB of physically contiguous memory. Therefore, the ten bits of PA-1 are sufficient to specify the base physical address of any 4 MB page below 4GB. An operating system easily can support addresses greater than 4 GB simply by providing the upper 4 bits of the physical address in PA-2 when creating a PDE for a 4 MB page.

3.1.2. Fault Detection

There are several conditions that can cause P6 family processors that support this feature to generate a page fault (PF) fault that are related to the use of, or switching between, various memory management features:

- If the PSE feature is enabled, a nonzero value in any of the remaining reserved bits (17-21) of a 4 MB PDE causes a page fault, with the reserved bit (bit 3) set in the error code.
- If the PAE feature is enabled and set to use 2 MB pages (that is, 8-byte page-directory table entries are being used), a nonzero value in any of the reserved bits 13-20 causes a page fault, with the reserved bit (bit 3) set in the error code. Note that bit 12 is now being used to support the Page Attribute Table feature (see Section 9.1 of this addendum).

CHAPTER 9
MEMORY CACHE CONTROL

The following information will be added to Chapter 9 of the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide*, in the next revision.

9.1. PAGE ATTRIBUTE TABLE (PAT)

The Page Attribute Table (PAT) is an extension to Intel's 32-bit processor virtual memory architecture for certain P6 family processors. Specifically, the PAT is an extension of the page-table format, which allows the specification of memory types to regions of physical memory based on linear address mappings. The PAT provides the equivalent functionality of an unlimited number of Memory Type Range Registers (MTRRs).

Using the PAT in conjunction with the MTRRs of the P6 family of processors extends the memory type information present in the current Intel Architecture page-table format. It combines the extendable and programmable qualities of the MTRRs with the flexibility of the page tables, allowing operating systems or applications to select the best memory type for their needs. The ability to apply the best memory type in a flexible way enables higher levels of performance.

NOTE: In multiple processor systems, the operating system(s) must maintain MTRR consistency between all the processors in the system. The P6 family processors provide no hardware support for maintaining this consistency. In general, all processors must have the same MTRR values.

9.1.1. Background

The P6 family of processors support the assignment of specific memory types to physical addresses. Memory type support is provided through the use of Memory Type Range Registers (MTRRs). Currently there are two interacting mechanisms that work together to set the effective memory type: the MTRRs and the page tables (refer to the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide).*

The MTRRs define the memory types for physical address ranges. MTRRs have specific alignment and length requirements for the memory regions they describe. Therefore, they are useful for statically describing memory types for physical ranges, and are typically set up by the system BIOS. However, they are incapable of describing memory types for the dynamic, linearly addressed data structures of programs. The MTRRs are an expandable and programmable way to encode memory types, but are inflexible because they can only apply those memory types to physical address ranges.

The page tables allow memory types to be assigned dynamically to linearly addressed pages of memory. This gives the operating system the maximum amount of flexibility in applying memory types to any data structure. However, the page tables only offer three of the five basic P6 processor family memory type encodings: Write-back (WB), Write-through (WT) and Uncached (UC). The PAT extends the existing page-table format to enable the specification of additional memory types.

9.1.2. Detecting Support for the PAT Feature

The page attribute table (PAT) feature is detected by an operating system through the use of the CPUID instruction. Specifically, the operating system executes the CPUID instruction with the value 1 in the EAX register, and then determines support for the feature by inspecting bit 16 of the EDX register return value. If the PAT is supported, an operating system is permitted to utilize the model specific register (MSR) specified for programming the PAT, as well as make use of the PAT-index bit (PAT*i*), which was formerly a reserved bit in the page tables.

Note that there is not a separate flag or control bit in any of the control registers that enables the use of this feature. The PAT is always enabled on all processors that support it, and the table lookup always occurs whenever paging is enabled and for all paging modes (e.g., PSE, PAE).

9.1.3. Technical Description of the PAT

The Page Attribute Table is a Model Specific Register (MSR) at address 277H (for information about the MSRs, refer to Table B-1 in the *Intel Architecture Software Developer’s Manual, Volume 3: System Programming Guide*). The model specific register address for the PAT is defined and will remain at the same address on future Intel processors that support this feature. Figure 9-1 shows the format of the 64-bit register containing the PAT.

31 27	26 24	23 19	18 16	15 11	10 8	7 3	2 0
Rsvd	PA3	Rsvd	PA 2	Rsvd	PA 1	Rsvd	PA 0

63 59	58 56	55 51	50 48	47 43	42 40	39 35	34 32
Rsvd	PA7	Rsvd	PA6	Rsvd	PA5	Rsvd	PA4

NOTES:

1. PA0-7 = Specifies the eight page attribute locations contained within the PAT
2. Rsvd = Most significant bits for each Page Attribute are reserved for future expansion

Figure 9-1. Page Attribute Table Model Specific Register

Each of the eight page attribute fields can contain any of the available memory type encodings, or indexes, as specified in Table 9-1 in Section 9.1.4.

9.1.4. Accessing the PAT

Access to the memory types that have been programmed into the PAT register fields is accomplished with a 3-bit index consisting of the PAT*i*, PCD, and PWT bits. Table 9-1 shows how the PAT register fields are indexed. The last column of the table shows which memory type the processor assigns to each PAT field at processor reset and initialization. These initial values provide complete backward compatibility with previous Intel processors and existing software that use the previously existing page-table memory types and MTRRs.

Table 9-1. PAT Indexing and Values After Reset

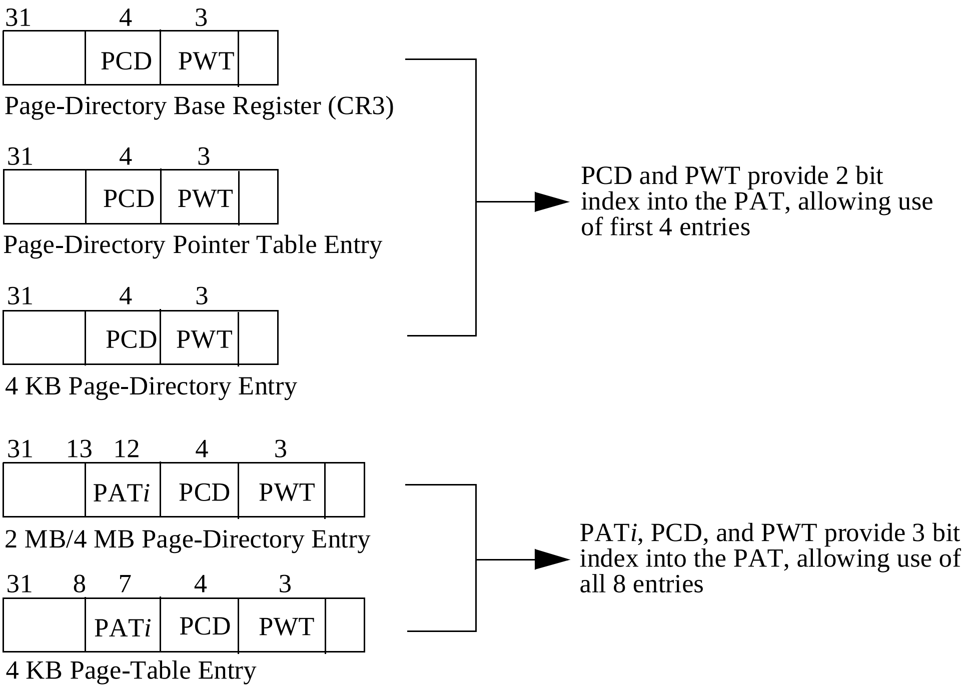
PAT*i*[1]	PCD	PWT	PAT Entry	Memory Type at Reset
0	0	0	0	WB
0	0	1	1	WT
0	1	0	2	UC-[2]
0	1	1	3	UC[3]
1	0	0	4	WB
1	0	1	5	WT
1	1	0	6	UC-[2]
1	1	1	7	UC[3]

NOTES:

1. PAT*i* bit is defined as bit 7 for 4 KB PTEs, bit 12 for PDEs mapping 2 MB/4 MB pages.
2. UC- is the page encoding PCD, PWT = 10 on P6 family processors that do not support this feature. UC- in the page table is overridden by WC in the MTRRs.
3. UC is the page encoding PCD, PWT = 11 on P6 family processors that do not support this feature. UC in the page-table overrides WC in the MTRRs.

In P6 family processors that do not support the PAT, the PCD and PWT bits are used to determine the page-table memory types of a given physical page. The PAT feature redefines these two bits and combines them with a newly defined PAT-index bit (PAT*i*) in the page-directory and page-table entries. These three bits create an index into the 8-entry Page Attribute Table. The memory type from the PAT is used in place of PCD and PWT for computing the effective memory type.

The bit used for PAT*i* differs depending upon the level of the paging hierarchy. PAT*i* is bit 7 for page-table entries, and bit 12 for page-directory entries that map to large pages. Reserved bit faults are disabled for nonzero values for PAT*i*, but remain present for all other reserved bits. This is true for 4 KB/2 MB pages when PAE is enabled. The PAT index scheme for each level of the paging hierarchy is shown in Figure 9-2.


31
4
3
PCD
PWT
Page-Directory Base Register (CR3)
31
4
3
PCD
PWT
Page-Directory Pointer Table Entry
31
4
3
PCD
PWT
4 KB Page-Directory Entry
PCD and PWT provide 2 bit index into the PAT, allowing use of first 4 entries
31
13
12
4
3
PATi
PCD
PWT
2 MB/4 MB Page-Directory Entry
31
8
7
4
3
PATi
PCD
PWT
4 KB Page-Table Entry
PATi, PCD, and PWT provide 3 bit index into the PAT, allowing use of all 8 entries

NOTE:

This figure only shows the format of the lower 32 bits of the PDE, PDEPTR, and PTEs when in PAE mode (refer to Figure 3-21 of the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide*). Additionally, the formats shown in this figure are not meant to accurately represent the entire structure, but only the labeled bits.

Figure 9-2. Page Attribute Table Index Scheme for Paging Hierarchy

Figure 9-2 shows that the PAT bit is not defined in CR3, the Page-Directory-Pointer Tables when PAE is enabled, or the Page Directory when it doesn't describe a large page. In these cases, only PCD and PWT are used to index into the PAT, limiting the operating system to using only the first 4 entries of PAT for describing the memory attributes of the paging hierarchy. Note that all 8 PAT entries are available for describing a 4 KB/2 MB/4 MB page.

The memory type as now defined by PAT interacts with the MTRR memory type to determine the effective memory type as outlined in Table 9-2. Compare this to Table 9-5 in Chapter 9 of the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide*.

Table 9-2. Effective Memory Type Depending on MTRRs and PAT

PAT Memory Type	MTRR Memory Type	Effective Memory Type
UC-	WB, WT	UC_PAGE
	WC	WC
	UC	UC_MTRR
	WP	Undefined
UC	WB, WT, WP, WC	UC_PAGE
	UC	UC_MTRR
WC	X	WC
WT	WB, WT	WT
	UC	UC_MTRR
	WC	Undefined
	WP	Undefined
WP	WB, WP	WP
	UC	UC_MTRR
	WC, WT	Undefined
WB	WB	WB
	UC	UC_MTRR
	WC	WC
	WT	WT
	WP	WP

NOTES:

- This table assumes that the CD and NW flags in register CR0 are set to 0. If CR0.CD = 1, then the effective memory type returned is UC, regardless of what is indicated in the table. However, this does not force strict ordering. To ensure strict ordering, the MTRRs also must be disabled.
- The effective memory types in the gray areas are implementation dependent and may be different between implementations of Intel Architecture processors.
- UC_MTRR indicates that the UC attribute came from the MTRRs and the processor(s) are not required to snoop their caches since the data could never have been cached. This is preferred for performance reasons.
- UC_PAGE indicates that the UC attribute came from the page tables and processors are required to check their caches because the data may be cached due to page aliasing, which is not recommended.

- UC- is the page encoding PCD, PWT = 10 on P6 family processors that do not support this feature. UC- in the PTE/PDE is overridden by WC in the MTRRs.
- UC is the page encoding PCD, PWT = 11 on P6 family processors that do not support this feature. UC in the PTE/PDE overrides WC in the MTRRs.

Whenever the MTRRs are disabled, via bit 11 (E) in the MTRRDefType register, the effective memory type is UC for all memory ranges.

An operating system can program the PAT and select the 8 most useful attribute combinations. The PAT allows an operating system to offer performance-enhancing memory types to applications.

The page attribute for addresses containing a page directory or page table supports only the first four entries in the PAT, since a PAT-index bit is not defined for these mappings. The page attribute is determined by using the two-bit value specified by PCD and PWT in CR3 (for page directory) or the page-directory entry (for page tables). The same applies to Page-Directory-Pointer Tables when PAE is enabled.

9.1.5. Programming the PAT

The Page Attribute Table is read/write accessible to software operating at ring 0 through the use of the `rdmsr` and `wrmsr` instructions. Accesses are directed to the PAT through use of model specific register address 277H. Refer to Figure 9-1 for the format of the 64-bit register containing the PAT.

The PAT implementation on processors that support the feature defines only the 3 least significant bits for page attributes. These bits are used to specify the memory type with the same encoding as used for the P6 family MTRRs as shown in Table 9-2 (from the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide,* Table 9-6). Processors that support the PAT feature modify those encodings slightly, in that encoding 0 is UC and encoding 7 is UC-, as indicated in the Table 9-3. Encoding 7 remains undefined for the fixed and variable MTRRs, and any attempt to write an undefined memory type encoding continues to generate a GP fault. Attempting to write an undefined memory type encoding into the PAT generates a GP fault.

Table 9-3. PAT Memory Types and Their Properties

Mnemonic	Encoding	Cacheable	Writeback Cacheable	Allows Speculative Reads	Memory Ordering Model
Uncacheable (UC)	0	No	No	No	Strong Ordering
Write Combining (WC)	1	No	No	Yes	Weak Ordering
Write-through (WT)	4	Yes	No	Yes	Speculative Processor Ordering
Write-protect (WP)	5	Yes for reads, no for writes	No	Yes	Speculative Processor Ordering
Write-back (WB)	6	Yes	Yes	Yes	Speculative Processor Ordering
Uncached (UC-)	7	No	No	No	Strong Ordered, but can be overridden by WC in the MTRRs
Reserved	2, 3, 87-255				

The operating system is responsible for ensuring that changes to a PAT entry occur in a manner that maintains the consistency of the processor caches and translation lookaside buffers (TLB). This is accomplished by following the procedure as specified in the *Intel Architecture Software Developer's Manual, Volume 3: System Programming Guide*, for changing the value of an MTRR. It involves a specific sequence of operations that includes flushing the processor(s) caches and TLBs. An operating system must ensure that the PAT of all processors in a multi-processing system have the same values.

The PAT allows any memory type to be specified in the page tables, and therefore it is possible to have a single physical page mapped by two different linear addresses with differing memory types. This practice is strongly discouraged by Intel and should be avoided as it may lead to undefined results. In particular, a WC page must never be aliased to a cacheable page because WC writes may not check the processor caches. When remapping a page that was previously mapped as a cacheable memory type to a WC page, an operating system can avoid this type of aliasing by:

- Removing the previous mapping to a cacheable memory type in the page tables; that is, make them not present.
- Flushing the TLBs of processors that may have used the mapping, even speculatively.
- Creating a new mapping to the same physical address with a new memory type, for instance, WC.
- Flushing the caches on all processors that may have used the mapping previously.

Operating systems that use a Page Directory as a Page Table and enable Page Size Extensions must carefully scrutinize the use of the PAT*i* index bit for the 4 KB Page-Table Entries. The

PAT*i* index bit for a PTE (bit 7) corresponds to the page size bit in a PDE. Therefore, the operating system can only utilize PAT entries PA0-3 when setting the caching type for a page table that is also used as a page directory. If the operating system attempts to use PAT entries PA4-7 when using this memory as a page table, it effectively sets the PS bit for the access to this memory as a page directory.

UNITED STATES, Intel Corporation
2200 Mission College Blvd., P.O. Box 58119, Santa Clara, CA 95052-8119
Tel: +1 408 765-8080

JAPAN, Intel Japan K.K.
5-6 Tokodai, Tsukuba-shi, Ibaraki-ken 300-26
Tel: + 81-29847-8522

FRANCE, Intel Corporation S.A.R.L.
1, Quai de Grenelle, 75015 Paris
Tel: +33 1-45717171

UNITED KINGDOM, Intel Corporation (U.K.) Ltd.
Pipers Way, Swindon, Wiltshire, England SN3 1RJ
Tel: +44 1-793-641440

GERMANY, Intel GmbH
Dornacher Strasse 1
85622 Feldkirchen/ Muenchen
Tel: +49 89/99143-0

HONG KONG, Intel Semiconductor Ltd.
32/F Two Pacific Place, 88 Queensway, Central
Tel: +852 2844-4555

CANADA, Intel Semiconductor of Canada, Ltd.
190 Attwell Drive, Suite 500
Rexdale, Ontario M9W 6H8
Tel: +416 675-2438