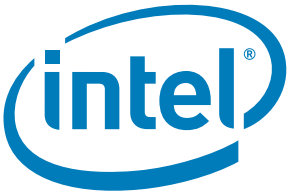
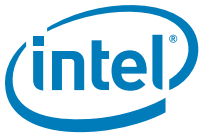


Intel® Core™ i7-900 Mobile Processor Extreme Edition Series, Intel® Core™ i7-800 and i7-700 Mobile Processor Series

Specification Update

July 2014

Document Number: 320767-026US

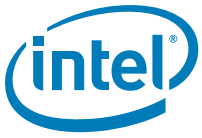


Contents

§ §

Revision History

Revision	Description	Date
001	Initial Release	September 2009
002	January Release • Addition of Errata AAP99-104 • Updated AAP84 and AAP90	January 2010
003	February Release • Added AAP105 and AAP106	February 2010
004	March Release • Added AAP107	March 2010
005	• Updated processor identification table with new SKU info • Added AAP108 and AAP109	April 2010
006	• Added AAP1 in Specification Changes	May 2010
007	• Added AAP110 and AAP111 • Added AAP2 and AAP3 in Specification Changes	June 2010
008	• Removed Item Numbering • Added AAP112, AAP113, AAP114 and AAP115	July 2010
009	• Added AAP116, AAP117 and AAP118 • Added AAP4 in Specification Changes	September 2010
010	• Updated processor identification table with new SKU information • Updated AAP41 • Added AAP119 and AAP120	October 2010
011	• Added AAP121	December 2010
012	• Added AAP122, AAP123 and AAP124	January 2011
013	• Added AAP125 to AAP130	February 2011
014	• Added AAP131	June 2011
015	• Added AAP132	August 2011
016	• Added AAP133	September 2011
017	• Updated Erratum AAP122	October 2011
018	• Added Erratum AAP134	December 2011
019	• Added Erratum AAP135	March 2012
020	• Added Documentation Change AAP1	January 2013
021	• Added Erratum AAP136	May 2013
022	• Added errata AAP137 and AAP138	June 2013
023	• Added errata AAP139 and AAP140	August 2013



Revision	Description	Date
024	• No errata added or deleted • Document standardization	October 2013
025	• No errata added or deleted • Document standardization	December 2013
026	• Updated link to access Intel® 64 and IA-32 Architecture Software Developer's Manual Documentation Changes	July 2014

§ §

Preface

This document is an update to the specifications contained in the Affected 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

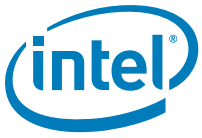
Document Title	Document Number / Location
Intel® Core™ i7-900 Mobile Processor Extreme Edition Series, Intel® Core™ i7-800 and i7-700 Mobile Processor Series Datasheet - Volume 1	320765
Intel® Core™ i7-900 Mobile Processor Extreme Edition Series, Intel® Core™ i7-800 and i7-700 Mobile Processor Series Datasheet - Volume 2	320766

Related Documents

Document Title	Document Number / Location
AP-485, Intel® Processor Identification and the CPUID Instruction	http://www.intel.com/design/processor/applnots/241618.htm
Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1: Basic Architecture *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2A: Instruction Set Reference Manual A-M* *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2B: Instruction Set Reference Manual N-Z* *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3A: System Programming Guide* *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3B: System Programming Guide* *Intel® 64 and IA-32 Intel Architecture Optimization Reference Manual* *Intel® 64 and IA-32 Architectures Software Developer's Manual Documentation Changes* (see note 1)	http://www.intel.com/content/www/us/en/processors/architectures-software-developer-manuals.html
ACPI Specifications	www.acpi.info

Notes:

1. Documentation changes for Intel® 64 and IA-32 Architecture Software Developer's Manual volumes 1, 2A, 2B, 3A, and 3B, and bug fixes are posted in the *Intel® 64 and IA-32 Architecture Software Developer's Manual Documentation Changes*.



Nomenclature

Errata are design defects or errors. These may cause the Intel® Core™ i7-900 Mobile Processor Extreme Edition Series, Intel® Core™ i7-800 and i7-700 Mobile Processor Series 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.

S-Spec Number is a five-digit code used to identify products. Products are differentiated by their unique characteristics, e.g., core speed, L3 cache size, package type, etc. as described in the processor identification information table. Read all notes associated with each S-Spec number.

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 Tables of Changes

The following tables indicate the errata, specification changes, specification clarifications, or documentation changes which apply to the processor. 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. These tables uses the following notations:

Codes Used in Summary Tables

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

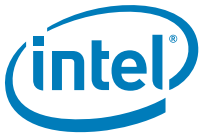
(Page):	Page location of item in this document.

Status

Doc:	Document change or update will be implemented.
Plan Fix:	This erratum may be fixed in a future stepping of the product.
Fixed:	This erratum has been previously fixed.
No Fix:	There are no plans to fix this erratum.

Row

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

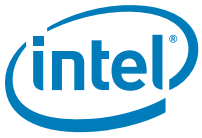


Errata (Sheet 1 of 6)

Number	Steppings	Status	ERRATA
	B-1		
AAP1	X	No Fix	The Processor May Report a #TS Instead of a #GP Fault
AAP2	X	No Fix	REP MOVS/STOS Executing with Fast Strings Enabled and Crossing Page Boundaries with Inconsistent Memory Types May Use an Incorrect Data Size or Lead to Memory-Ordering Violations
AAP3	X	No Fix	Code Segment Limit/Canonical Faults on RSM May Be Serviced before Higher Priority Interrupts/Exceptions and May Push the Wrong Address onto the Stack
AAP4	X	No Fix	Performance Monitor SSE Retired Instructions May Return Incorrect Values
AAP5	X	No Fix	Premature Execution of a Load Operation Prior to Exception Handler Invocation
AAP6	X	No Fix	MOV To/From Debug Registers Causes Debug Exception
AAP7	X	No Fix	Incorrect Address Computed for Last Byte of FXSAVE/FXRSTOR Image Leads to Partial Memory Update
AAP8	X	No Fix	Values for LBR/BTS/BTM Will Be Incorrect after an Exit from SMM
AAP9	X	No Fix	Single Step Interrupts with Floating Point Exception Pending May Be Mishandled
AAP10	X	No Fix	Fault on ENTER Instruction May Result in Unexpected Values on Stack Frame
AAP11	X	No Fix	IRET under Certain Conditions May Cause an Unexpected Alignment Check Exception
AAP12	X	No Fix	General Protection Fault (#GP) for Instructions Greater Than 15 Bytes May Be Preempted
AAP13	X	No Fix	General Protection (#GP) Fault May Not Be Signaled on Data Segment Limit Violation above 4-G Limit
AAP14	X	No Fix	LBR, BTS, BTM May Report a Wrong Address When an Exception/Interrupt Occurs in 64-bit Mode
AAP15	X	No Fix	MONITOR or CLFLUSH on the Local XAPIC's Address Space Results in Hang
AAP16	X	No Fix	Corruption of CS Segment Register during RSM While Transitioning From Real Mode to Protected Mode
AAP17	X	No Fix	Performance Monitoring Events for Read Miss to Level 3 Cache Fill Occupancy Counter May Be Incorrect
AAP18	X	No Fix	A VM Exit on MWAIT May Incorrectly Report the Monitoring Hardware As Armed
AAP19	X	No Fix	Delivery Status of the LINT0 Register of the Local Vector Table May Be Lost
AAP20	X	No Fix	Performance Monitor Event SEGMENT_REG_LOADS Counts Inaccurately
AAP21	X	No Fix	#GP on Segment Selector Descriptor that Straddles Canonical Boundary May Not Provide Correct Exception Error Code
AAP22	X	No Fix	Improper Parity Error Signaled in the IQ Following Reset When a Code Breakpoint Is Set on a #GP Instruction
AAP23	X	No Fix	An Enabled Debug Breakpoint or Single Step Trap May Be Taken after MOV SS/POP SS Instruction If It Is Followed by an Instruction That Signals a Floating Point Exception
AAP24	X	No Fix	IA32_MPERF Counter Stops Counting during On-Demand TM1

Errata (Sheet 2 of 6)

Number	Steppings	Status	ERRATA
	B-1		
AAP25	X	No Fix	The Memory Controller tTHROT_OPREF Timings May Be Violated during Self Refresh Entry
AAP26	X	No Fix	Processor May Over Count Correctable Cache MESI State Errors
AAP27	X	No Fix	Synchronous Reset of IA32_APERF/IA32_MPERF Counters on Overflow Does Not Work
AAP28	X	No Fix	Disabling Thermal Monitor While Processor Is Hot, Then Re-enabling, May Result in Stuck Core Operating Ratio
AAP29	X	No Fix	OVER Bit for IA32_MCi_STATUS Register May Get Set on Specific Internal Error
AAP30	X	No Fix	Writing the Local Vector Table (LVT) When an Interrupt Is Pending May Cause an Unexpected Interrupt
AAP31	X	No Fix	Faulting Intel® MMX™ Technology Instruction May Incorrectly Update x87 FPU Tag Word
AAP32	X	No Fix	xAPIC Timer May Decrement Too Quickly following an Automatic Reload While in Periodic Mode
AAP33	X	No Fix	Reported Memory Type May Not Be Used to Access the VMCS and Referenced Data Structures
AAP34	X	No Fix	B0-B3 Bits in DR6 for Non-Enabled Breakpoints May Be Incorrectly Set
AAP35	X	No Fix	Core C6 May Clear Previously Logged TLB Errors
AAP36	X	No Fix	Performance Monitor Event MISALIGN_MEM_REF May Over Count
AAP37	X	No Fix	Changing the Memory Type for an In-Use Page Translation May Lead to Memory-Ordering Violations
AAP38	X	No Fix	Running with Write Major Mode Disabled May Lead to a System Hang
AAP39	X	No Fix	Infinite Stream of Interrupts May Occur If an ExtINT Delivery Mode Interrupt Is Received While All Cores in C6
AAP40	X	No Fix	Two xAPIC Timer Event Interrupts May Unexpectedly Occur
AAP41	X	No Fix	EOI Transaction May Not Be Sent If Software Enters Core C6 during an Interrupt Service Routine
AAP42	X	No Fix	FREEZE_WHILE_SMM Does Not Prevent Event from Pending PEBS during SMM
AAP43	X	No Fix	APIC Error “Received Illegal Vector” May Be Lost
AAP44	X	No Fix	DR6 May Contain Incorrect Information When the First Instruction after a MOV SS,r/m or POP SS Is a Store
AAP45	X	No Fix	An Uncorrectable Error Logged in IA32_CR_MC2_STATUS May also Result in a System Hang
AAP46	X	No Fix	IA32_PERF_GLOBAL_CTRL MSR May Be Incorrectly Initialized
AAP47	X	No Fix	Performance Monitor Interrupts Generated from Uncore Fixed Counters (394H) May Be Ignored
AAP48	X	No Fix	Performance Monitor Counter INST_RETIRED.STORES May Count Higher Than Expected
AAP49	X	No Fix	Sleeping Cores May Not Be Woken Up on Logical Cluster Mode Broadcast IPI Using Destination Field Instead of Shorthand
AAP50	X	No Fix	Faulting Executions of FXRSTOR May Update State Inconsistently

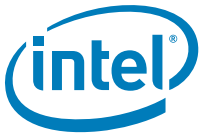


Errata (Sheet 3 of 6)

Number	Steppings	Status	ERRATA
	B-1		
AAP51	X	No Fix	Performance Monitor Event EPT.EPDPE_MISS May Be Counted While EPT Is Disabled
AAP52	X	No Fix	Memory Aliasing of Code Pages May Cause Unpredictable System Behavior
AAP53	X	No Fix	Performance Monitor Counters May Count Incorrectly
AAP54	X	No Fix	Processor Forward Progress Mechanism Interacting with Certain MSR/CSR Writes May Cause Unpredictable System Behavior
AAP55	X	No Fix	Performance Monitor Event Offcore_response_0 (B7H) Does Not Count NT Stores to Local DRAM Correctly
AAP56	X	No Fix	EFLAGS Discrepancy on Page Faults and on EPT-Induced VM Exits after a Translation Change
AAP57	X	No Fix	System May Hang if MC_CHANNEL_{0,1}_MC_DIMM_INIT_CMD.DO_ZQCL Commands Are Not Issued in Increasing Populated DDR3 Rank Order
AAP58	X	No Fix	Package C3/C6 Transitions When Memory 2x Refresh Is Enabled May Result in a System Hang
AAP59	X	No Fix	Back to Back Uncorrected Machine Check Errors May Overwrite IA32_MC3_STATUS.MSCOD
AAP60	X	No Fix	Memory Intensive Workloads with Core C6 Transitions May Cause System Hang
AAP61	X	No Fix	Corrected Errors with a Yellow Error Indication May Be Overwritten by Other Corrected Errors
AAP62	X	No Fix	PSI# Signal May Incorrectly Be Left Asserted
AAP63	X	No Fix	Performance Monitor Events DCACHE_CACHE_LD and DCACHE_CACHE_ST May Overcount
AAP64	X	No Fix	Rapid Core C3/C6 Transitions May Cause Unpredictable System Behavior
AAP65	X	No Fix	Performance Monitor Events INSTR_RETIRED and MEM_INST_RETIRED May Count Inaccurately
AAP66	X	No Fix	A Page Fault May Not Be Generated When the PS Bit Is Set to "1" in a PML4E or PDPTE
AAP67	X	No Fix	CPURESET Bit Does Not Get Cleared
AAP68	X	No Fix	PHOLD Disable in MISCCTRLSTS Register Does Not Work
AAP69	X	No Fix	PCIe PMCSR Power State Field Incorrectly Allows Requesting of the D1 and D2 Power States
AAP70	X	No Fix	Concurrent Updates to a Segment Descriptor May Be Lost
AAP71	X	No Fix	PMIs May Be Lost during Core C6 Transitions
AAP72	X	No Fix	Uncacheable Access to a Monitored Address Range May Prevent Future Triggering of the Monitor Hardware
AAP73	X	No Fix	BIST Results May Be Additionally Reported after a GETSEC[WAKEUP] or INIT-SIPI Sequence
AAP74	X	No Fix	Pending x87 FPU Exceptions (#MF) May Be Signaled Earlier Than Expected
AAP75	X	No Fix	VM Exits Due to "NMI-Window Exiting" May Be Delayed by One Instruction
AAP76	X	No Fix	Malformed PCIe Packet Generated under Heavy Outbound Load
AAP77	X	No Fix	PCIe* Operation in x16 Mode with Inbound Posted Writes May Be Unreliable

Errata (Sheet 4 of 6)

Number	Steppings	Status	ERRATA
	B-1		
AAP78	X	No Fix	Unpredictable PCI Behavior Accessing Non-existent Memory Space
AAP79	X	No Fix	VM Exits Due to EPT Violations Do Not Record Information about Pre-IRET NMI Blocking
AAP80	X	No Fix	Intel® VT-d Receiving Two Identical Interrupt Requests May Corrupt Attributes of Remapped Interrupt or Hang a Subsequent Interrupt-Remap-Cache Invalidation Command
AAP81	X	No Fix	S1 Entry May Cause Cores to Exit C3 or C6 C-State
AAP82	X	No Fix	Multiple Performance Monitor Interrupts Are Possible on Overflow of IA32_FIXED_CTR2
AAP83	X	No Fix	LBRs May Not Be Initialized during Power-On Reset of the Processor
AAP84	X	No Fix	Unexpected Interrupts May Occur on C6 Exit If Using APIC Timer to Generate Interrupts
AAP85	X	No Fix	LBR, BTM or BTS Records May Have Incorrect Branch from Information after an Enhanced Intel SpeedStep® Technology Transition, T-states, C1E, or Adaptive Thermal Throttling
AAP86	X	No Fix	VMX-Preemption Timer Does Not Count Down at the Rate Specified
AAP87	X	No Fix	Multiple Performance Monitor Interrupts Are Possible on Overflow of Fixed Counter 0
AAP88	X	No Fix	SVID and SID of Devices 8 and 16 Only Implement Bits [7:0]
AAP89	X	No Fix	No_Soft_Reset Bit in the PMCSR Does Not Operate As Expected
AAP90	X	No Fix	VM Exits Due to LIDT/LGDT/SIDT/SGDT Do Not Report Correct Operand Size
AAP91	X	No Fix	DPRSLPVR Signal May Be Incorrectly Asserted on Transition Between Low Power C-states
AAP92	X	No Fix	Performance Monitoring Events STORE_BLOCKS.NOT_STA and STORE_BLOCKS.STA May Not Count Events Correctly
AAP93	X	No Fix	Storage of PEBS Record Delayed Following Execution of MOV SS or STI
AAP94	X	No Fix	Performance Monitoring Event FP_MMX_TRANS_TO_MMX May Not Count Some Transitions
AAP95	X	No Fix	INVLPG Following INVEPT or INVVPID May Fail to Flush All Translations for a Large Page
AAP96	X	No Fix	LER MSRs May Be Unreliable
AAP97	X	No Fix	MCi_Status Overflow Bit May Be Incorrectly Set on a Single Instance of a DTLB Error
AAP98	X	No Fix	Debug Exception Flags DR6.B0-B3 Flags May Be Incorrect for Disabled Breakpoints
AAP99	X	No Fix	An Exit From the Core C6-state May Result in the Dropping of an Interrupt
AAP100	X	No Fix	PCIe* Extended Capability Structures May Be Incorrect
AAP101	X	No Fix	PMIs during Core C6 Transitions May Cause the System to Hang
AAP102	X	No Fix	2MB Page Split Lock Accesses Combined with Complex Internal Events May Cause Unpredictable System Behavior
AAP103	X	No Fix	IA32_MC8_CTL2 MSR Is Not Cleared on Processor Warm Reset
AAP104	X	No Fix	The TPM's Locality 1 Address Space Cannot Be Opened



Errata (Sheet 5 of 6)

Number	Steppings	Status	ERRATA
	B-1		
AAP105	X	No Fix	PCIe* Link Bit Errors Present during L0s Entry May Cause the System to Hang during L0s Exit
AAP106	X	No Fix	The Combination of a Page-Split Lock Access and Data Accesses That Are Split across Cacheline Boundaries May Lead to Processor Livelock
AAP107	X	No Fix	FP Data Operand Pointer May Be Incorrectly Calculated after an FP Access Which Wraps a 4-Gbyte Boundary in Code That Uses 32-Bit Address Size in 64-bit Mode
AAP108	X	No Fix	IOTLB Invalidations Not Completing on Intel® VT-d Engine for Integrated High Definition Audio
AAP109	X	No Fix	IO_SMI Indication in SMRAM State Save Area May Be Lost
AAP110	X	No Fix	PCIe* Squelch Detect May be Slow to Respond During L0s Entry and May Cause a Surprise Link Down Condition
AAP111	X	No Fix	TR Corruption Due to Save/Restore x87 FPU Pointers in SMRAM
AAP112	X	No Fix	PCIe* Lanes Returning to The Active Power State May Cause The System to Hang
AAP113	X	No Fix	Performance Monitor Events for Hardware Prefetches Which Miss The L1 Data Cache May be Over Counted
AAP114	X	No Fix	Poisoned Write Caused by an Internal Parity Error Targeting IIO PCI Configuration Registers or MMIO Space will Not be Suppressed
AAP115	X	No Fix	VM Exit May Incorrectly Clear IA32_PERF_GLOBAL_CTRL [34:32]
AAP116	X	No Fix	PCIe* Port's LTSSM May Not Transition Properly in the Presence of TS1 or TS2 Ordered Sets That Have Unexpected Symbols Within those Sets
AAP117	X	No Fix	NTB/RP Link Will Send Extra TS2 Ordered Set During Link Training
AAP118	X	No Fix	PCIe* Ports May Not Enter Slave Loopback Mode From the Configuration LTSSM State
AAP119	X	No Fix	Unexpected DMI and PCIe* Link Retraining and Correctable Errors Reported
AAP120	X	No Fix	QPI Lane May Be Dropped During Full Frequency Deskew Phase of Training
AAP121	X	No Fix	PerfMon Overflow Status Can Not be Cleared After Certain Conditions Have Occurred
AAP122	X	No Fix	An Unexpected Page Fault or EPT Violation May Occur After Another Logical Processor Creates a Valid Translation for a Page
AAP123	X	No Fix	L1 Data Cache Errors May be Logged With Level Set to 1 Instead of 0
AAP124	X	No Fix	Stack Pushes May Not Occur Properly for Events Delivered Immediately After VM Entry to 16-Bit Software
AAP125	X	No Fix	Executing The GETSEC Instruction While Throttling May Result in a Processor Hang
AAP126	X	No Fix	PerfMon Event LOAD_HIT_PRE.SW_PREFETCH May Overcount
AAP127	X	No Fix	Successive Fixed Counter Overflows May be Discarded
AAP128	X	No Fix	#GP May be Signaled When Invalid VEX Prefix Precedes Conditional Branch Instructions
AAP129	X	No Fix	A Logical Processor May Wake From Shutdown State When Branch-Trace Messages or Branch-Trace Stores Are Enabled
AAP130	X	No Fix	Task Switch to a TSS With an Inaccessible LDTR Descriptor May Cause Unexpected Faults

Errata (Sheet 6 of 6)

Number	Steppings	Status	ERRATA
	B-1		
AAP131	X	No Fix	VM Entries That Return From SMM Using VMLAUNCH May Not Update The Launch State of the VMCS
AAP132	X	No Fix	VM Entry May Clear Bytes 81H-83H on Virtual-APIC Page When "Use TPR Shadow" Is 0
AAP133	X	No Fix	A First Level Data Cache Parity Error May Result in Unexpected Behavior
AAP134	X	No Fix	Intel® Trusted Execution Technology ACM Revocation
AAP135	X	No Fix	An Event May Intervene Before a System Management Interrupt That Results from IN or INS
AAP136	X	No Fix	The Corrected Error Count Overflow Bit in IA32_ MC0_STATUS is Not Updated After a UC Error is Logged
AAP137	X	No Fix	The Upper 32 Bits of CR3 May be Incorrectly Used With 32-Bit Paging
AAP138	X	No Fix	EPT Violations May Report Bits 11:0 of Guest Linear Address Incorrectly
AAP139	X	No Fix	SMRAM State-Save Area Above the 4GB Boundary May Cause Unpredictable System Behavior
AAP140	X	No Fix	Virtual-APIC Page Accesses With 32-Bit PAE Paging May Cause a System Crash

Specification Changes

Number	SPECIFICATION CHANGES
AAP1	Update to Datasheet - Volume 2 to Uncore Revision Identification Register
AAP2	Update to Datasheet - Volume 2 to PCI Express Device Control Register 2
AAP3	Update to Datasheet - Volume 2 to Completion Timeout Control Register
AAP4	Update to Datasheet - Volume 1 to Table 35 and Table 41

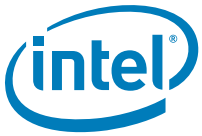
Specification Clarifications

Number	SPECIFICATION CLARIFICATIONS
	None for this revision of this specification update.

Documentation Changes

Number	DOCUMENTATION CHANGES
AAP1	On-Demand Clock Modulation Feature Clarification

§ §



Identification Information

Component Identification using Programming Interface

The Intel® Core™ i7-900 Mobile Processor Extreme Edition Series, Intel® Core™ i7-800 and i7-700 Mobile Processor Series stepping can be identified by the following processor signatures:

Reserved	Extended Family[1]	Extended Model[2]	Reserved	Processor Type[3]	Family Code[4]	Model Number[5]	Stepping ID[6]
31:28	**27:20**	**19:16**	**15:14**	**13:12**	**11:8**	**7:4**	**3:0**
	00000000b	0001b		00b	0110	1110b	xxxxb

Notes:

1. The Extended Family, bits [27:20] are used in conjunction with the Family Code, specified in bits [11:8], to indicate whether the processor belongs to the Intel386®, Intel486®, Pentium®, Pentium Pro®, Pentium® 4, or Intel® Core™ processor family.
2. The Extended Model, bits [19:16] in conjunction with the Model Number, specified in bits [7:4], are used to identify the model of the processor within the processor's family.
3. The Processor Type, specified in bits [13:12] indicates whether the processor is an original OEM processor, an OverDrive® processor, or a dual processor (capable of being used in a dual processor system).
4. The Family Code corresponds to bits [11:8] of the EDX register after RESET, bits [11:8] of the EAX register after the CPUID instruction is executed with a 1 in the EAX register, and the generation field of the Device ID register accessible through Boundary Scan.
5. The Model Number corresponds to bits [7:4] of the EDX register after RESET, bits [7:4] of the EAX register after the CPUID instruction is executed with a 1 in the EAX register, and the model field of the Device ID register accessible through Boundary Scan.
6. The Stepping ID in bits [3:0] indicates the revision number of that model. See Table 1 for the processor stepping ID number in the CPUID information.

When EAX is initialized to a value of '1', the CPUID instruction returns the *Extended Family, Extended Model, Processor Type, Family Code, Model Number and Stepping ID* value in the EAX register. Note that the EDX processor signature value after reset is equivalent to the processor signature output value in the EAX register.

Cache and TLB descriptor parameters are provided in the EAX, EBX, ECX and EDX registers after the CPUID instruction is executed with a 2 in the EAX register.

The Intel® Core™ i7-900 Mobile Processor Extreme Edition Series, Intel® Core™ i7-800 and i7-700 Mobile Processor Series can be identified by the following register contents:

Processor Stepping	Vendor ID[1]	Device ID[2]	Revision ID[3]
B-1	8086h	D132h	11h

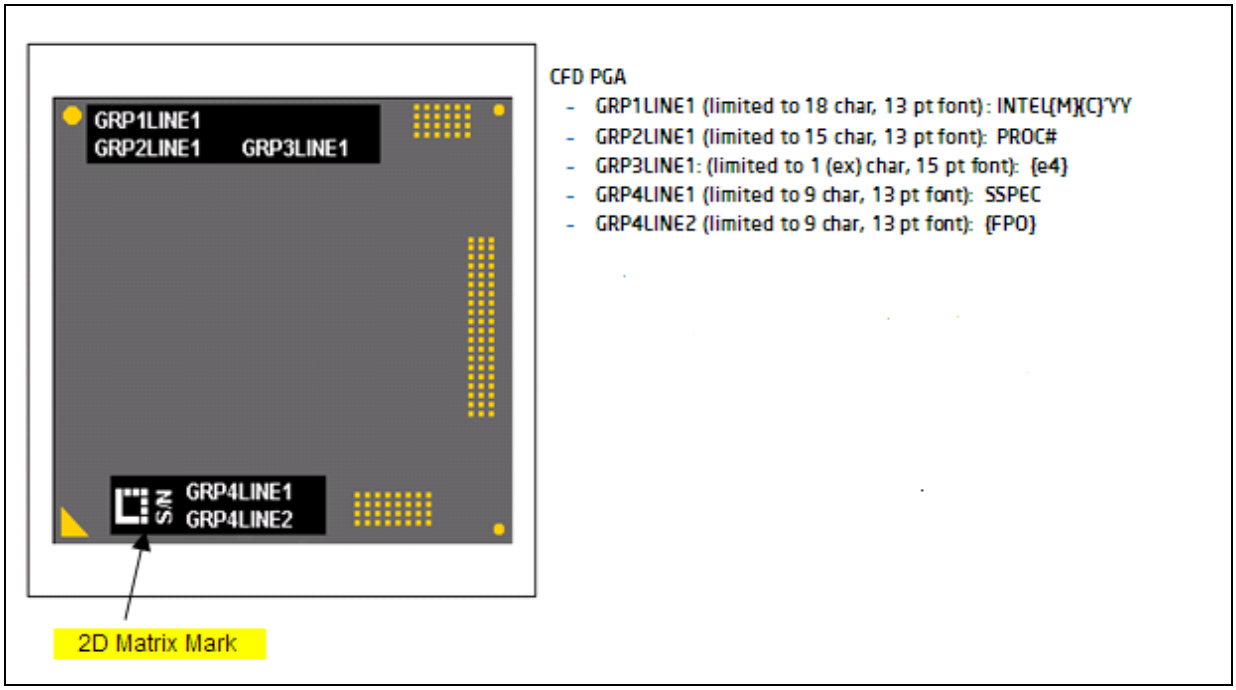
Notes:

1. The Vendor ID corresponds to Bits 15:0 of the Vendor ID Register located at Offset 00–01h in the PCI Function 0 configuration space.
2. The Device ID corresponds to Bits 15:0 of the Device ID Register located at Device 0 Offset 02–03h in the PCI Function 0 configuration space.
3. The Revision Number corresponds to Bits 7:0 of the Revision ID Register located at Offset 08h in the PCI Function 0 configuration space.

Component Marking Information

The processor stepping can be identified by the following component markings:

Figure 1. Intel® Core™ i7-900 Mobile Processor Extreme Edition Series, Intel® Core™ i7-800 and i7-700 Mobile Processor Series Component Markings



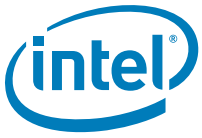


Table 1. Processor Identification

S-Spec Number	Processor Number	Stepping	Processor Signature	Core Frequency (GHz) / DDR3 (MHz)	Max Intel® Turbo Boost Technology Frequency (GHz)[1]	LFM Frequency (GHz)	Shared L3 Cache Size (MB)	Notes
SLBLW	i7-920XM	B-1	106E5h	2.00 / 1333	4 core: 2.26 3 core: 2.26 2 core: 3.06 1 core: 3.20	1.200	8	2, 3, 4, 5
SLBLX	i7-820QM	B-1	106E5h	1.73 / 1333	4 core: 2.00 3 core: 2.00 2 core: 2.80 1 core: 3.06	1.200	8	2, 3, 4, 5
SLBLY	i7-720QM	B-1	106E5h	1.60 / 1333	4 core: 1.73 3 core: 1.73 2 core: 2.40 1 core: 2.80	0.933	6	2, 3, 4, 5
SLBSC	i7-940XM	B-1	106E5h	2.13 / 1333	4 core: 2.40 3 core: 2.40 2 core: 3.20 1 core: 3.33	1.200	8	2, 3, 4, 5, 6
SLBMP	i7-840QM	B-1	106E5h	1.86 / 1333	4 core: 2.00 3 core: 2.00 2 core: 2.93 1 core: 3.20	1.200	8	2, 3, 4, 5, 6
SLBQG	i7-740QM	B-1	106E5h	1.73 / 1333	4 core: 1.86 3 core: 1.86 2 core: 2.53 1 core: 2.93	0.933	6	2, 3, 4, 5, 6

Notes:

1. This column indicates maximum Intel® Turbo Boost Technology frequency (GHz) for 4, 3, 2, or 1 cores active respectively.
2. Intel® Hyper-Threading Technology enabled.
3. Intel® Trusted Execution Technology (Intel® TXT) enabled.
4. Intel® Virtualization Technology (Intel® VT) for IA-32, Intel® 64 and Intel® Architecture (Intel® VT-x) enabled. Intel® Virtualization Technology for Directed I/O (Intel® VT-d) enabled.
5. The core frequency reported in the processor brand string is rounded to 2 decimal digits. (For example, core frequency of 3.06666, repeating 6, is reported as @3.07 in brand string. Core frequency of 1.7333, is reported as @1.73 in brand string.)

§ §

Errata

AAP1. The Processor May Report a #TS Instead of a #GP Fault

Problem: A jump to a busy TSS (Task-State Segment) may cause a #TS (invalid TSS exception) instead of a #GP fault (general protection exception).

Implication: Operation systems that access a busy TSS may get invalid TSS fault instead of a #GP fault. Intel has not observed this erratum with any commercially-available software.

Workaround: None identified.

Status: For the steppings affected, see the Summary Tables of Changes.

AAP2. REP MOVS/STOS Executing with Fast Strings Enabled and Crossing Page Boundaries with Inconsistent Memory Types May Use an Incorrect Data Size or Lead to Memory-Ordering Violations

Problem: Under certain conditions as described in the Software Developers Manual section "Out-of-Order Stores For String Operations in Pentium 4, Intel Xeon, and P6 Family Processors" the processor performs REP MOVS or REP STOS as fast strings. Due to this erratum fast string REP MOVS/REP STOS instructions that cross page boundaries from WB/WC memory types to UC/WP/WT memory types, may start using an incorrect data size or may observe memory ordering violations.

Implication: Upon crossing the page boundary the following may occur, dependent on the new page memory type:

- UC the data size of each write will now always be 8 bytes, as opposed to the original data size.
- WP the data size of each write will now always be 8 bytes, as opposed to the original data size and there may be a memory ordering violation.
- WT there may be a memory ordering violation.

Workaround: Software should avoid crossing page boundaries from WB or WC memory type to UC, WP or WT memory type within a single REP MOVS or REP STOS instruction that will execute with fast strings enabled.

Status: For the steppings affected, see the Summary Tables of Changes.

AAP3. Code Segment Limit/Canonical Faults on RSM May Be Serviced before Higher Priority Interrupts/Exceptions and May Push the Wrong Address onto the Stack

Problem: Normally, when the processor encounters a Segment Limit or Canonical Fault due to code execution, a #GP (General Protection Exception) fault is generated after all higher priority Interrupts and exceptions are serviced. Due to this erratum, if RSM (Resume from System Management Mode) returns to execution flow that results in a Code Segment Limit or Canonical Fault, the #GP fault may be serviced before a higher priority Interrupt or Exception (e.g., NMI (Non-Maskable Interrupt), Debug break(#DB), Machine Check (#MC), etc.). If the RSM attempts to return to a non-canonical address, the address pushed onto the stack for this #GP fault may not match the non-canonical address that caused the fault.

Implication: Operating systems may observe a #GP fault being serviced before higher priority Interrupts and Exceptions. Intel has not observed this erratum on any commercially-available software.

Workaround: None identified.

Status: For the steppings affected, see the Summary Tables of Changes.