Intel® Atom™ Z8000 Processor Series

Specification Update

March 2015

Revision 001
## Contents

Preface ................................................................................................................................. 5  
Summary Tables of Changes ............................................................................................. 7  
Identification Information ............................................................................................... 10  
Component Marking Information .................................................................................... 12  
Errata ................................................................................................................................. 13  
Specification Changes ..................................................................................................... 23  
Specification Clarifications ............................................................................................. 24  
Documentation Changes .................................................................................................. 25  

§
## Revision History

<table>
<thead>
<tr>
<th>Document Number</th>
<th>Revision Number</th>
<th>Description</th>
<th>Revision Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>332067</td>
<td>001</td>
<td>Initial release</td>
<td>March 2015</td>
</tr>
</tbody>
</table>

§
Preface

This document is an update to the specifications contained in the documents listed in the following Affected Documents table. It is a compilation of device and document errata and specification clarifications and changes, and is intended for hardware system manufacturers and for software developers of applications, operating system, and tools.

Information types defined in the Nomenclature section of this document are consolidated into this document and are no longer published in other documents. This document may also contain information that has not been previously published.

Note: Throughout this document Intel® Atom™ Z8000 Processor Series is referred as Processor or SoC.

Affected Documents

<table>
<thead>
<tr>
<th>Document Title</th>
<th>Document Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel® Atom™ Z8000 Processor Series Datasheet (Volume 1 of 2)</td>
<td>332065</td>
</tr>
<tr>
<td>Intel® Atom™ Z8000 Processor Series Datasheet (Volume 2 of 2)</td>
<td>332066</td>
</tr>
</tbody>
</table>

Related Documents

Refer the following documents, which may be beneficial when reading this document or for additional information.

<table>
<thead>
<tr>
<th>Document</th>
<th>Document Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel® 64 and IA-32 Architectures Software Developer’s Manuals</td>
<td></td>
</tr>
<tr>
<td>• Volume 1: Basic Architecture</td>
<td></td>
</tr>
<tr>
<td>• Volume 2A: Instruction Set Reference, A-M</td>
<td></td>
</tr>
<tr>
<td>• Volume 2B: Instruction Set Reference, N-Z</td>
<td></td>
</tr>
<tr>
<td>• Volume 3B: System Programming Guide</td>
<td></td>
</tr>
</tbody>
</table>
Nomenclature

**Errata** are design defects or errors in engineering samples. Errata may cause the processor behavior to deviate from published specifications. Hardware and software designed to be used with any given stepping assumes 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, that is, core speed, L2 cache size, and package type 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, and so forth).
**Summary Tables of Changes**

The following table indicates the Specification Changes, Errata, Specification Clarifications, or Documentation Changes, which apply to the listed steppings. Intel intends to fix some of the errata in a future stepping of the component, and to account for the other outstanding issues through documentation or Specification Changes as noted. This table uses the following notations:

### Codes Used in Summary Table

**Stepping**

- **X**: Erratum, 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 list stepping.

### Status

- **Doc**: Document change or update that 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 is no plan to fix this erratum.

### Row

<table>
<thead>
<tr>
<th>Number</th>
<th>Stepping</th>
<th>Status</th>
<th>Errata Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>C-0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHT1</td>
<td>X</td>
<td>No Fix</td>
<td>IA32_DEBUGCTL_FREEZE_PERFMON_ON_PMI is Incorrectly Cleared by SMI</td>
</tr>
<tr>
<td>CHT2</td>
<td>X</td>
<td>No Fix</td>
<td>Redirection of RSM to Probe Mode May Not Generate an LBR Record</td>
</tr>
<tr>
<td>CHT3</td>
<td>X</td>
<td>No Fix</td>
<td>Unsynchronized Cross-Modifying Code Operations Can Cause Unexpected Instruction Execution Results</td>
</tr>
<tr>
<td>CHT4</td>
<td>X</td>
<td>No Fix</td>
<td>Reported Memory Type May Not Be Used to Access the VMCS and Referenced Data Structures</td>
</tr>
<tr>
<td>Number</td>
<td>Stepping</td>
<td>Status</td>
<td>Errata Title</td>
</tr>
<tr>
<td>--------</td>
<td>----------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>CHT5</td>
<td>X</td>
<td>No Fix</td>
<td>A Page Fault May Not be Generated When the PS bit is set to “1” in a PML4E or PDPTE</td>
</tr>
<tr>
<td>CHT6</td>
<td>X</td>
<td>No Fix</td>
<td>Some Performance Counter Overflows May Not be Logged in IA32_PERF_GLOBAL_STATUS When FREEZE_PERFMON_ON_PMI is Enabled</td>
</tr>
<tr>
<td>CHT7</td>
<td>X</td>
<td>No Fix</td>
<td>CS Limit Violations May Not be Detected After VM Entry</td>
</tr>
<tr>
<td>CHT8</td>
<td>X</td>
<td>No Fix</td>
<td>PEBS Record EventingIP Field May be Incorrect After CS.Base Change</td>
</tr>
<tr>
<td>CHT9</td>
<td>X</td>
<td>No Fix</td>
<td>MOVNTDQA From WC Memory May Pass Earlier Locked Instructions</td>
</tr>
<tr>
<td>CHT10</td>
<td>X</td>
<td>No Fix</td>
<td>Performance Monitor Instructions Retired Event May Not Count Consistently</td>
</tr>
<tr>
<td>CHT11</td>
<td>X</td>
<td>No Fix</td>
<td>LBR Stack And Performance Counter Freeze on PMI May Not Function Correctly</td>
</tr>
<tr>
<td>CHT12</td>
<td>X</td>
<td>No Fix</td>
<td>VM Exit May Set IA32_EFER.NXE When IA32_MISC_ENABLE Bit 34 is Set to 1</td>
</tr>
<tr>
<td>CHT13</td>
<td>X</td>
<td>No Fix</td>
<td>Machine Check Status Overflow Bit May Not be Set</td>
</tr>
<tr>
<td>CHT14</td>
<td>X</td>
<td>No Fix</td>
<td>RTIT Trace May Contain FUP.FAR Packet With Incorrect Address</td>
</tr>
<tr>
<td>CHT15</td>
<td>X</td>
<td>No Fix</td>
<td>RTIT May Delay The PSB by One Packet</td>
</tr>
<tr>
<td>CHT16</td>
<td>X</td>
<td>No Fix</td>
<td>RTIT TraceStop Condition Detected During Buffer Overflow May Not Clear TraceActive</td>
</tr>
<tr>
<td>CHT17</td>
<td>X</td>
<td>No Fix</td>
<td>RTIT FUP.BuffOvf Packet May be Incorrectly Followed by a TIP Packet</td>
</tr>
<tr>
<td>CHT18</td>
<td>X</td>
<td>No Fix</td>
<td>RTIT CYC Packet Payload Values May be Off by 1 Cycle</td>
</tr>
<tr>
<td>CHT19</td>
<td>X</td>
<td>No Fix</td>
<td>The SoC May Not Detect a Battery Charger or May Fail to Connect to a USB Host</td>
</tr>
</tbody>
</table>
## Summary Tables of Changes

<table>
<thead>
<tr>
<th>Number</th>
<th>Stepping</th>
<th>Status</th>
<th>Errata Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>C-0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHT20</td>
<td>X</td>
<td>No Fix</td>
<td>RGB666 Pixel Format Display Panel May Not Operate as Expected</td>
</tr>
<tr>
<td>CHT21</td>
<td>X</td>
<td>No Fix</td>
<td>LPDDR3 tINIT0 JEDEC Specification Violation</td>
</tr>
<tr>
<td>CHT22</td>
<td>X</td>
<td>No Fix</td>
<td>HDMI And DVI Displays May Flicker or Blank Out When Using Certain Pixel Frequencies</td>
</tr>
<tr>
<td>CHT23</td>
<td>X</td>
<td>No Fix</td>
<td>Timing Specification Violation on MIPI DSI Interface</td>
</tr>
<tr>
<td>CHT24</td>
<td>X</td>
<td>No Fix</td>
<td>xHCI USB2.0 Split-Transactions Error Counter Reset Issue</td>
</tr>
<tr>
<td>CHT25</td>
<td>X</td>
<td>No Fix</td>
<td>POPCNT Instruction May Take Longer to Execute Than Expected</td>
</tr>
<tr>
<td>CHT26</td>
<td>X</td>
<td>No Fix</td>
<td>LPSS UART Not Fully Compatible With 16550 UART</td>
</tr>
<tr>
<td>CHT27</td>
<td>X</td>
<td>No Fix</td>
<td>Accessing Undocumented Unimplemented MMIO Space May Cause a System Hang</td>
</tr>
<tr>
<td>CHT28</td>
<td>X</td>
<td>No Fix</td>
<td>USB xHCI Controller May Not Re-Enter D3 State After a USB Wake Event</td>
</tr>
</tbody>
</table>

### Specification Changes

- None

### Specification Clarifications

- None

### Documentation Changes

- None
Identification Information

Intel® Atom™ Z8000 Processor Series samples on 14-nm process processor signature can be identified by the following registers contents:

Table 1. Processor Signature by Using Programming Interface

<table>
<thead>
<tr>
<th>Reserved</th>
<th>Extended Family&lt;sup&gt;1&lt;/sup&gt;</th>
<th>Extended Model&lt;sup&gt;2&lt;/sup&gt;</th>
<th>Reserved</th>
<th>Processor Type&lt;sup&gt;3&lt;/sup&gt;</th>
<th>Family Code&lt;sup&gt;4&lt;/sup&gt;</th>
<th>Model Number&lt;sup&gt;5&lt;/sup&gt;</th>
<th>Stepping ID&lt;sup&gt;6&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000b</td>
<td>00000000b</td>
<td>0011b</td>
<td>00b</td>
<td>0b</td>
<td>0110b</td>
<td>0101b</td>
<td>0001b</td>
</tr>
</tbody>
</table>

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, Intel® Core™2, or Intel® Atom™ processor series.
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).
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 is accessible through Boundary Scan.
6. The Stepping ID in Bits [3:0] indicates the revision number of that model.

When EAX is initialized to a value of 1, the CPUID instruction returns the Extended Family, Extended Model, Type, Family, Model and Stepping value in the EAX register.

Note: The EDX processor signature value after reset is equivalent to the processor signature output value in the EAX register.
### Table 2. Identification Table for Intel® Atom™ Z8000 Processor Series

<table>
<thead>
<tr>
<th>S-Spec</th>
<th>Stepping</th>
<th>Processor Number</th>
<th>Core Speed</th>
<th>Memory Frequency</th>
<th>Integrated Graphics Core Speed</th>
<th>ISP Speed</th>
<th>H-DID/ H-RID1</th>
<th>G-DID/ G-RID2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Burst Frequency (BFM)</td>
<td>High Frequency Mode (HFM)</td>
<td>Low Frequency Mode (LFM)</td>
<td>Burst Frequency</td>
<td>Base Frequency</td>
<td>Burst Frequency</td>
</tr>
<tr>
<td>SR27M</td>
<td>C-0</td>
<td>Z8700</td>
<td>2.4 GHz</td>
<td>1.6 GHz</td>
<td>480 MHz</td>
<td>LPDDR3 - 1600MT/s</td>
<td>600 MHz</td>
<td>400 MHz</td>
</tr>
<tr>
<td>SR29W</td>
<td>C-0</td>
<td>Z8700</td>
<td>2.4 GHz</td>
<td>1.6 GHz</td>
<td>480 MHz</td>
<td>LPDDR3 - 1600MT/s</td>
<td>600 MHz</td>
<td>400 MHz</td>
</tr>
<tr>
<td>SR27N</td>
<td>C-0</td>
<td>Z8500</td>
<td>2.24 GHz</td>
<td>1.44 GHz</td>
<td>480 MHz</td>
<td>LPDDR3 - 1600MT/s</td>
<td>600 MHz</td>
<td>400 MHz</td>
</tr>
</tbody>
</table>

**NOTES:**

1. H-DID – Host Device ID; H-RID – Host Revision ID (H-RID are last three Bits of H-DID).
2. G-DID – Graphics Device ID; G-RID – Graphics Revision ID (G-RID are last three Bits of G-DID).
Component Marking Information

Processor shipments can be identified by the following component markings and example pictures.

Figure 1. Intel® Atom™ Z8000 Processor Series Component Marking Information

Sample Marking Information:
GRP1LINE1: i{M}-{C}YY
GRP2LINE1: FPO12345
**Errata**

<table>
<thead>
<tr>
<th>CHT1</th>
<th>IA32_DEBUGCTL.FREEZE_PERFMON_ON_PMI is Incorrectly Cleared by SMI</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Problem:</strong></td>
<td>FREEZE_PERFMON_ON_PMI (bit 12) in the IA32_DEBUGCTL MSR (1D9H) is erroneously cleared during delivery of an SMI (system-management interrupt).</td>
</tr>
<tr>
<td><strong>Implication:</strong></td>
<td>As a result of this erratum the performance monitoring counters will continue to count after a PMI occurs in SMM (system-management Mode).</td>
</tr>
<tr>
<td><strong>Workaround:</strong></td>
<td>None identified.</td>
</tr>
<tr>
<td><strong>Status:</strong></td>
<td>For the steppings affected, refer the Summary Tables of Changes.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHT2</th>
<th>Redirection of RSM to Probe Mode May Not Generate an LBR Record</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Problem:</strong></td>
<td>A redirection of the RSM instruction to probe mode may not generate the LBR (Last Branch Record) record that would have been generated by a non-redirected RSM instruction.</td>
</tr>
<tr>
<td><strong>Implication:</strong></td>
<td>The LBR stack may be missing a record when redirection of RSM to probe mode is used. The LBR stack will still properly describe the code flow of non-SMM code.</td>
</tr>
<tr>
<td><strong>Workaround:</strong></td>
<td>None identified.</td>
</tr>
<tr>
<td><strong>Status:</strong></td>
<td>For the steppings affected, refer the Summary Tables of Changes.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHT3</th>
<th>Unsynchronized Cross-Modifying Code Operations Can Cause Unexpected Instruction Execution Results</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Problem:</strong></td>
<td>The act of one processor or system bus master writing data into a currently executing code segment of a second processor with the intent of having the second processor execute that data as code is called cross-modifying code (XMC). XMC that does not force the second processor to execute a synchronizing instruction prior to execution of the new code is called unsynchronized XMC. Software using unsynchronized XMC to modify the instruction byte stream of a processor can see unexpected or unpredictable execution behavior from the processor that is executing the modified code.</td>
</tr>
<tr>
<td><strong>Implication:</strong></td>
<td>In this case the phrase &quot;unexpected or unpredictable execution behavior&quot; encompasses the generation of most of the exceptions listed in the Intel Architecture Software Developer's Manual Volume 3: System Programming Guide including a General Protection Fault (GPF) or other unexpected behaviors. In the event that unpredictable execution causes a GPF the application executing the unsynchronized XMC operation would be terminated by the operating system.</td>
</tr>
<tr>
<td><strong>Workaround:</strong></td>
<td>In order to avoid this erratum programmers should use the XMC synchronization algorithm as detailed in the Intel Architecture Software Developer's Manual Volume 3: System Programming Guide Section: Handling Self- and Cross-Modifying Code.</td>
</tr>
</tbody>
</table>
Status: For the steppings affected, refer the Summary Tables of Changes.

CHT4  Reported Memory Type May Not Be Used to Access the VMCS and Referenced Data Structures

Problem: Bits 53:50 of the IA32_VMX_BASIC MSR report the memory type that the processor uses to access the VMCS and data structures referenced by pointers in the VMCS. Due to this erratum, a VMX access to the VMCS or referenced data structures will instead use the memory type that the MTRRs (memory-type range registers) specify for the physical address of the access.

Implication: Bits 53:50 of the IA32_VMX_BASIC MSR report that the WB (write-back) memory type will be used but the processor may use a different memory type.

Workaround: Software should ensure that the VMCS and referenced data structures are located at physical addresses that are mapped to WB memory type by the MTRRs.

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

CHT5  A Page Fault May Not be Generated When the PS bit is set to “1” in a PML4E or PDPTE

Problem: On processors supporting Intel® 64 architecture the PS bit (Page Size bit 7) is reserved in PML4Es and PDPTEs. If the translation of the linear address of a memory access encounters a PML4E or a PDPTE with PS set to 1 a page fault should occur. Due to this erratum, PS of such an entry is ignored and no page fault will occur due to its being set.

Implication: Software may not operate properly if it relies on the processor to deliver page faults when reserved bits are set in paging-structure entries.

Workaround: Software should not set bit 7 in any PML4E or PDPTE that has Present Bit (Bit 0) set to “1”.

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

CHT6  Some Performance Counter Overflows May Not be Logged in IA32_PERF_GLOBAL_STATUS When FREEZE_PERFMON_ON_PMI is Enabled

Problem: When enabled, FREEZE_PERFMON_ON_PMI bit 12 in IA32_DEBUGCTL MSR (1D9H) freezes PMCs (performance monitoring counters) on a PMI (Performance Monitoring Interrupt) request by clearing the IA32_PERF_GLOBAL_CTRL MSR (38FH). Due to this erratum, when FREEZE_PERFMON_ON_PMI is enabled and two or more PMCs overflow within a small window of time and PMI is requested, then subsequent PMC overflows may not be logged in IA32_PERF_GLOBAL_STATUS MSR (38EH).

Implication: On a PMI, subsequent PMC overflows may not be logged in IA32_PERF_GLOBAL_STATUS MSR.

Workaround: Re-enabling the PMCs in IA32_PERF_GLOBAL_CTRL will log the overflows that were not previously logged in IA32_PERF_GLOBAL_STATUS.

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

**CHT7  CS Limit Violations May Not be Detected After VM Entry**

**Problem:** The processor may fail to detect a CS limit violation on fetching the first instruction after VM entry if the first byte of that instruction is outside the CS limit but the last byte of the instruction is inside the limit.

**Implication:** The processor may erroneously execute an instruction that should have caused a general protection exception.

**Workaround:** When a VMM emulates a branch instruction it should inject a general protection exception if the instruction’s target EIP is beyond the CS limit.

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

**CHT8  PEBS Record EventingIP Field May be Incorrect After CS.Base Change**

**Problem:** Due to this erratum a PEBS (Precise Event Base Sampling) record generated after an operation which changes CS.Base may contain an incorrect address in the EventingIP field.

**Implication:** Software attempting to identify the instruction which caused the PEBS event may identify the incorrect instruction when non-zero CS.Base is supported and CS.Base is changed. Intel has not observed this erratum to impact the operation of any commercially available system.

**Workaround:** None identified.

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

**CHT9  MOVNTDQA From WC Memory May Pass Earlier Locked Instructions**

**Problem:** An execution of MOVNTDQA that loads from WC (write combining) memory may appear to pass an earlier locked instruction to a different cache line.

**Implication:** Software that expects a lock to fence subsequent MOVNTDQA instructions may not operate properly. If the software does not rely on locked instructions to fence the subsequent execution of MOVNTDQA then this erratum does not apply.

**Workaround:** Software that requires a locked instruction to fence subsequent executions of MOVNTDQA should insert an LFENCE instruction before the first execution of MOVNTDQA following the locked instruction. If there is already a fencing or serializing instruction between the locked instruction and the MOVNTDQA, then an additional LFENCE is not necessary.

**Status:** For the steppings affected, refer the Summary Tables of Changes.
Errata

CHT10 Performance Monitor Instructions Retired Event May Not Count Consistently

Problem: Performance Monitor Instructions Retired (Event C0H; Umask 00H) and the instruction retired fixed counter (IA32_FIXED_CTR0 MSR (309H)) are used to track the number of instructions retired. Due to this erratum, certain situations may cause the counter(s) to increment when no instruction has retired or to not increment when specific instructions have retired.

Implication: A performance counter counting instructions retired may over or under count. The count may not be consistent between multiple executions of the same code.

Workaround: None identified.

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

CHT11 LBR Stack And Performance Counter Freeze on PMI May Not Function Correctly

Problem: When FREEZE_LBRS_ON_PMI flag (bit 11) in IA32_DEBUGCTL MSR (1D9H) is set, the LBR (Last Branch Record) stack is frozen on a hardware PMI (Performance Monitoring Interrupt) request. When FREEZE_PERFMON_ON_PMI flag (bit 12) in IA32_DEBUGCTL MSR is set, a PMI request clears each of the ENABLE fields of the IA32_PERF_GLOBAL_CTRL MSR (38FH) to disable counters. Due to this erratum, when FREEZE_LBRS_ON_PMI and/or FREEZE_PERFMON_ON_PMI is set in IA32_DEBUGCTL MSR and the local APIC is disabled or the PMI LVT is masked, the LBR Stack and/or Performance Counters Freeze on PMI may not function correctly.

Implication: Performance monitoring software may not function properly if the LBR Stack and Performance Counters Freeze on PMI do not operate as expected. Intel has not observed this erratum to impact any commercially available system.

Workaround: None identified.

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

CHT12 VM Exit May Set IA32_EFER.NXE When IA32_MISC_ENABLE Bit 34 is Set to 1

Problem: When “XD Bit Disable” in the IA32_MISC_ENABLE MSR (1A0H) bit 34 is set to 1, it should not be possible to enable the “execute disable” feature by setting IA32_EFER.NXE. Due to this erratum, a VM exit that occurs with the 1-setting of the “load IA32_EFER” VM-exit control may set IA32_EFER.NXE even if IA32_MISC_ENABLE bit 34 is set to 1. This erratum can occur only if IA32_MISC_ENABLE bit 34 was set by guest software in VMX non-root operation.

Implication: Software in VMX root operation may execute with the “execute disable” feature enabled despite the fact that the feature should be disabled by the IA32_MISC_ENABLE MSR. Intel has not observed this erratum with any commercially available software.

Workaround: A virtual-machine monitor should not allow guest software to write to the IA32_MISC_ENABLE MSR.
Errata

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

**CHT13 Machine Check Status Overflow Bit May Not Be Set**

**Problem:** The OVER (error overflow) indication in bit [62] of the IA32_MC0_STATUS MSR (401H) may not be set if IA32_MC0_STATUS.MCACOD (bits [15:0]) held a value of 0x3 (External Error) when a second machine check occurred in the MC0 bank. Additionally, the OVER indication may not be set if the second machine check has an MCACOD value of 0x810, 0x820 or 0x410, regardless of the first error.

**Implication:** Software may not be notified that an overflow of MC0 bank occurred.

**Workaround:** None identified.

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

**CHT14 RTIT Trace May Contain FUP.FAR Packet With Incorrect Address**

**Problem:** The FUP.FAR (Flow Update Packet for Far Transfer) generated by RTIT (Real Time Instruction Trace) on a far transfer instruction should contain the linear address of the first byte of the next sequential instruction after the far transfer instruction. Due to this erratum, far transfer instructions with more than 3 prefixes may incorrectly include an address between the first byte of the far transfer instruction and the last byte of the far transfer instruction.

**Implication:** The RTIT Trace decoder may incorrectly decode the trace due to an incorrect address in the FUP packet.

**Workaround:** The RTIT trace decoder can identify a FUP.FAR in the middle of a far transfer instruction and treat that FUP.FAR as if it was coming from the first byte of the following sequential instruction.

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

**CHT15 RTIT May Delay The PSB by One Packet**

**Problem:** After an RTIT (Real Time Instruction Trace) packet that exceeds the limit specified by Pkt_Mask in RTIT_PACKET_COUNT (MSR 77Ch) bits [17:16], the PSB (Packet Stream Boundary) packet should be sent immediately. Due to this erratum, the PSB packet may be delayed by one packet.

**Implication:** The PSB packet may be delayed by one packet.

**Workaround:** None identified.

**Status:** For the steppings affected, refer the Summary Tables of Changes.
Errata

CHT16  RTIT TraceStop Condition Detected During Buffer Overflow May Not Clear TraceActive

Problem: If an RTIT (Real Time Instruction Trace) TraceStop condition is detected while RTIT_STATUS.Buffer_Overflow MSR (769H) bit 3 is set, the processor may not clear RTIT_CTL.TraceActive MSR (768H) bit 13, and tracing will continue after the overflow resolves. Such a case will be evident if the TraceStop packet is inserted before overflow is resolved, as indicated by the FUP.BuffOvf (Flow Update Packet for Buffer Overflow) packet.

Implication: The RTIT trace will continue tracing beyond the intended stop point.

Workaround: None identified.

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

CHT17  RTIT FUP.BuffOvf Packet May be Incorrectly Followed by a TIP Packet

Problem: When RTIT (Real Time Instruction Trace) suffers an internal buffer overflow, packet generation stops temporarily, after which a FUP.BuffOvf (Flow Update Packet for Buffer Overflow) is sent to indicate the LIP that follows the instruction upon which tracing resumes. In some cases, however, this packet will be immediately followed by a FUP.TIP (Flow Update Packet for Target IP) which was generated by a branch instruction that executed during the overflow. The IP payload of this FUP.TIP will be the LIP of the instruction upon which tracing resumes.

Implication: The spurious FUP.TIP packet may cause the RTIT trace decoder to fail.

Workaround: The RTIT trace decoder should ignore any FUP.TIP packet that immediately follows a FUP.BuffOvf whose IP matches the IP payload of the FUP.BuffOvf.

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

CHT18  RTIT CYC Packet Payload Values May be Off by 1 Cycle

Problem: When RTIT (Real Time Instruction Trace) is enabled with RTIT_CTL.Cyc_Acc MSR (768H) bit 1 set to 1, all CYC (Cycle Count) packets have a payload value that is one less than the number of cycles that have actually passed. Note that for CYC packets with a payload value of 0, the correct value may be 0 or 1.

Implication: The trace decoder will produce inaccurate performance data when using CYC packets to track software performance.

Workaround: As a partial workaround, the trace decoder should add 1 to the payload value of any CYC packet with a non-zero payload.

Status: For the steppings affected, refer the Summary Tables of Changes.
CHT19  The SoC May Not Detect a Battery Charger or May Fail to Connect to a USB Host

Problem: During power-on, when the SoC is used in device mode instead of host mode, the USB D+ line may have a 2 µsec glitch to 3.3 V.

Implication: Due to this erratum, the platform may not detect a battery charger (and hence not charge the battery) or the SoC may not successfully connect to an attached USB host.

Workaround: Power the SoC on before connecting to its USB port. Alternatively, manually disconnecting and re-connecting the USB cable restores operation after the erratum has occurred.

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

CHT20  RGB666 Pixel Format Display Panel May Not Operate as Expected

Problem: Due to this erratum, the RGB666 format support on the SOC has restrictions on the horizontal resolution. For single link MIPI* DSI (Display Serial Interface), the horizontal resolution must be evenly divisible by 4. For dual link MIPI DSI, one-half the horizontal resolution plus the overlapping pixels must be evenly divisible by 4.

Implication: Due to this erratum, the RGB666 panel may not operate as expected.

Workaround: For dual link panels with overlap, choose the overlap so that one-half the horizontal resolution plus the overlapping pixels is evenly divisible by 4. For single link panels the horizontal resolution must be evenly divisible by 4.

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

CHT21  LPDDR3 tINIT0 Duration May be Longer Than Specification Requirement

Problem: JEDEC Standard JESD209-3 requires a maximum power ramp duration tINIT0 of 20ms. Due to this erratum, the SoC may not comply with the tINIT0 specification.

Implication: Intel has not observed this erratum to impact the functionality or performance of any commercially available LPDDR3 parts. Intel has obtained waivers from vendors who provide commonly used LPDDR3 DRAM parts.

Workaround: None identified.

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

CHT22  HDMI And DVI Displays May Flicker or Blank Out When Using Certain Pixel Frequencies

Problem: Due to this erratum, HDMI (High-Definition Multimedia Interface) and DVI (Digital Visual Interface) ports may send data out at an incorrect rate, that is different than the one requested when using certain pixel frequencies.

Implication: When this erratum occurs, panels may flicker or blank out. The impacted pixel frequencies are: 218.25 MHz, 218.70 MHz, 220.50 MHz, 221.20 MHz, 229.50 MHz, 233.793 MHz and 234.00 MHz.

Workaround: Select a video mode that does not use an affected pixel frequency.

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

CHT23  MIPI* DSI Interface Timing Marginality

Problem: MIPI D-PHY Specification v1.1 Section 9.1.1 requires minimum tr (rise time) and tf (fall time) of 150ps for data rates of less than 1Gbps. Due to this erratum, the SoC may exhibit rise time and fall time marginality on a MIPI DSI interface with an 80 ohm or 100 ohm impedance.

Implication: EMI compliance tests on a MIPI DSI interface with one of the listed impedance values may not pass. Intel has not observed any functional, performance, or regulatory failures resulting from this erratum.

Workaround: None identified.

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

CHT24  xHCI USB2.0 Split-Transactions Error Counter Reset Issue

Problem: The xHCI controller may not reset its split transaction error counter if a high-speed USB hub propagates a mal-formed bit from a low-speed or full-speed USB device exhibiting non-USB specification compliant signal quality.

Implication: The implication is device dependent.
• Full Speed and Low Speed devices behind the hub may be re enumerated and may cause a device to not function as expected.

Workaround: A driver workaround has been identified. Contact your Intel representative for further details.

Status: For the steppings affected, refer the Summary Tables of Changes.
**Errata**

**CHT25**  **POPCNT Instruction May Take Longer to Execute Than Expected**

**Problem:** POPCNT instruction execution with a 32 or 64 bit operand may be delayed until previous non-dependent instructions have executed.

**Implication:** Software using the POPCNT instruction may experience lower performance than expected.

**Workaround:** None identified.

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

**CHT26**  **LPSS UART Not Fully Compatible With 16550 UART**

**Problem:** Stick Parity bit, LCR[5], (Line Control Register, HSUART0_BAR0, Offset 0CH; bit [5] for HSUART0 and HSUART1_BAR0, Offset 0CH; bit [5] for HSUART1) does not follow the 16550 specified behavior, instead the parity bit is always logic 0.

**Implication:** LPSS (Low Power Sub-system) UARTs are not fully 16550 compatible and may cause an error when connected to a UART device that requires the Stick Parity feature.

**Workaround:** Do not use Stick Parity mode of UART.

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

**CHT27**  **Accessing Undocumented Unimplemented MMIO Space May Cause a System Hang**

**Problem:** Access to undocumented unimplemented MMIO space should result in a software error. Due to this erratum, an access to undocumented unimplemented MMIO space may not complete.

**Implication:** When this erratum occurs, the system may hang.

**Workaround:** Do not access to undocumented unimplemented MMIO space.

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

**CHT28**  **USB xHCI Controller May Not Re-Enter D3 State After a USB Wake Event**

**Problem:** After processing a USB wake event, the USB xHCI controller may not reenter D3 state.

**Implication:** When this erratum occurs, the affected USB xHCI controller may not recognize subsequent USB wake events. When this erratum occurs, PME status bit [15] of register Power Management Control/Status (PM_CS) (Bus 0; Device 20; Function 20; Offset 74H) remains at 1.

**Workaround:** The software driver should set PMCTRL[28] (Bus 0; Device 14; Function 0; Offset 80A4H) after the xHCI controller enters D0 state following an exit from D3 state.

**Status:** For the steppings affected, refer the Summary Tables of Changes.
There are no specification changes in this revision of the Specification Update.

§
There are no specification clarifications in this revision of the Specification Update.

§
There are no documentation changes in this revision of the Specification Update.