Performance Monitoring Impact of Intel® Transactional Synchronization Extension Memory

Ordering Issue

March 2019

Revision 1.3

Document Number: 604224

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

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

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

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document. The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

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

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

Contents

Tables

Revision History

Document Number	Revision Number	Description	Date
604224	1.0	• Initial release of the document.	October 2018
604224	1.1	• Update to RTM disable in SGX and SMM modes	October 2018
604224	1.2	• Document RTM retry bit behavior. Perf updates. • PMU counter CPUID enumeration changes. Now number of general purpose counters in CPUID.0xA.EAX[15:08] is not changed.	January 2019
604224	1.3	• Additional details on re-enabling Intel TSX in Linux	March 2019

1 Introduction

This whitepaper describes Intel® Transactional Synchronization Extension (Intel® TSX) and Performance Monitoring Unit (PMU) behavior due to the updated microcode for Intel® Xeon® Processor E3 v5 and v6 Family (code name Skylake, Kaby Lake), Intel® Xeon® D (code name Skylake-D), Intel® Xeon® Scalable Processor and 6th, 7th, and 8th Generation Intel® Core™ i7 and i5 (code name Skylake, Kaby Lake, Coffee Lake and Whiskey Lake)

Intel TSX is a technology to enable hardware transactional memory. Intel TSX provides two software interfaces – Hardware Lock Elision (HLE) and Restricted Transactional Memory (RTM). HLE is an instruction prefix-based interface designed to be backward compatible with processors without Intel TSX support. RTM is a new instruction set interface using the XBEGIN and XEND instructions. For more details on Intel TSX please see http://www.intel.com/software/tsx.

The PMU measures performance events using performance counters. With the updated microcode described in the Intel TSX Memory Ordering Issue disclosure, general purpose (GP) counters will be available to the PMU driver but the fourth performance counter may contain unexpected values. The microcode update will also disable the HLE instruction prefix of Intel TSX and force all RTM transactions to abort when operating in Intel SGX mode or SMM mode.

Intel does not expect this microcode update to affect users who do not use the PMU, or who only use updated PMU drivers and tools. However, we recommend that PMU driver developers and performance tool developers follow the guidance in this document. Some advanced users of performance monitoring (Perfmon) may need to change their collection scripts and methodologies. The purpose of this whitepaper is to enable Perfmon users and tool developers to understand and, if necessary, work around the implications of these errata.

1.1 Implications for Users

The microcode update (CPUID.07H.EDX[bit 13]=1) is not expected to impact users who do not utilize Perfmon or HLE. We recommend that all users who do use Perfmon to update the PMU profiling tools to the latest version. Refer to Appendix B Guidance on Specific Profilers and the PMU tools documentation for more information on specific tools. No further action is required for PMU users who do not use groups.

Performance tools often use event multiplexing to collect data using more events than the number of available GP counters in the CPU. In a typical user scenario, such as Microarchitecture Exploration Analysis Type in Intel® VTune™ Amplifier, there are predefined tool configurations, profiles, or scripts that can specify the event groupings. The primary impact to users in this scenario is that the GP counter collection groups would be split into three events each instead of four events.[i] Updated versions of these profiling tools (see Appendix B Guidance on Specific Profilers) automatically handles this change.

Some advanced users who choose to develop their own event groupings in collection methodologies or scripts will need to modify their input to ensure they only utilize the number of available counters for each counter grouping, or use the method described in Appendix A Enabling FORCE_ABORT_RTM Mode to use all Four Counters.

1.2 Implications for PMU Drivers and Performance Tools

For more details on the PMU, refer to the Software Developer's Manual (http://www.intel.com/sdm) Volume 3, Chapter 18 "Performance Monitoring".

When Restricted Transactional Memory (RTM) is supported (CPUID.07H.EBX.RTM [bit 11] = 1) and CPUID.07H.EDX[bit 13]=1 and TSX_FORCE_ABORT[RTM_FORCE_ABORT]=0 (described later in this document), then Performance Monitor Unit (PMU) general purpose counter 3 (IA32_PMC3, MSR C4H and IA32_A_PMC3, MSR 4C4H) may contain unexpected values. Specifically, IA32_PMC3 (MSR C4H), IA32_PERF_GLOBAL_CTRL[3] (MSR 38FH) and IA32_PERFEVTSEL3 (MSR 189H) may contain unexpected values, which also affects IA32_A_PMC3 (MSR 4C4H) and IA32_PERF_GLOBAL_INUSE[3] (MSR 392H). PMU driver should avoid using general purpose counter 3. General purpose counters beyond 3, if reported in CPUID.(EAX=0xA).EAX[15:08], can still be used. Using counter 3 will result in nondeterministic counting, especially in the presence of RTM transactions; however, this should not crash the PMU driver.

When supporting event multiplexing, the PMU driver needs to split the event list into the correct configuration and groups based on the number of available GP counters. Also, any tool configurations or scripts which have hard-coded specific groups of counters must be changed to support the possibility of having fewer counters available.

New versions of the PMU driver tools can add an option to gain use of all GP counters by enabling FORCE_ABORT_RTM mode during the measurement (see Appendix A Enabling FORCE_ABORT_RTM Mode to use all Four Counters and Appendix B Guidance on Specific Profilers).

Table 1-1. Affected Products if Intel Intel® Transactional Synchronization Extension (Intel® TSX) is Supported

Family-Model	Stepping	Processor Families /Processor Number Series
06_55H	<=5	Intel® Xeon® Processor Scalable Family based on Skylake microarchitecture

Family-Model	Stepping	Processor Families / Processor Number Series
06_4EH, 06_5EH	All	6th generation Intel Core processors and Intel Xeon processor E3-1500m v5 product family and E3- 1200 v5 product family based on Skylake microarchitecture
06_8EH, 06_9EH	All	7th/8th generation Intel® Core™ processors based on Kaby/Coffee Lake microarchitecture
06_9E	All	Coffee Lake
06_8E	All	Whiskey Lake(ULT)

1.3 Implications for Intel® TSX Library Developers

Libraries using RTM transactions often check the return (or abort) value of _xbegin() to decide when and how often to retry transactions. Normally it is beneficial to retry transactions on a conflict abort. For a normal conflict abort the _XBEGIN_CONFLICT and _XBEGIN_RETRY bits are in the abort value. With CPUID.07H.EDX[bit 13]=1, it is possible to see conflict aborts that only have the _XBEGIN_CONFLICT bit set. These should be handled like normal conflicts for best performance.

With CPUID.07H.EDX[bit 13] =1, the correct implementation of lock elision for aborts that only have the XBEGIN_CONFLICT bit set:

```
for (retry = 0; retry < conflict_retry_count; retry++) {

    if ((abort_val = _xbegin()) == _XBEGIN_START) {

          … execute critical region …

    } else if (abort_val & _XBEGIN_CONFLICT) {

          /* wait on lock */
          continue; /* retry on conflict */

   }

}
```

For other non-XBEGIN aborts the retry bit should still be taken into account.

For more details on TSX please see the Intel Software Developer’s manual volume 1 chapter 16 (Programming with TSX) and the Intel Optimization Manual Chapter 16 (TSX Optimizations). Both available from http://www.intel.com/sdm. For generic TSX resources please see http://www.intel.com/software/tsx.

A Enabling FORCE_ABORT_RTM Mode to use all Four Counters

It is possible for the PMU driver to opt-in to use all GP counters by enabling FORCE_ABORT_RTM mode. This requires setting bit 0 (FORCE_ABORT_RTM) in the TSX_FORCE_ABORT (0x10f) MSR for each logical CPU that is affected. The driver should only access this MSR when CPUID 7.EDX[13] is set.

When FORCE_ABORT_RTM is enabled all RTM transactions on the logical CPU will forcefully abort, which can potentially impact performance of Intel TSX-enabled software, but the general purpose counter 3 will report correct values.

Application functionality should not be impacted because software that uses RTM is required to implement valid, non-transactional fallback paths for potential aborts, which are already exercised. When FORCE_ABORT_RTM mode is disabled, the RTM transactions will be allowed to commit again.

FORCE_ABORT_RTM mode does not change the CPUID feature enumeration for RTM or HLE.

FORCE_ABORT_RTM mode should always be disabled when the measurement session is finished to prevent applications that use RTM from experiencing performance impacts.

Table A-2. Description of TSX_FORCE_ABORT_MSR

Register address		Register Name / Bit fields	Bit Description	Comment
Hex	Dec			
10f	271	TSX_FORCE_ABORT		MSR existence enumerated by CPUID 7:0 EDX[13]
		0	RTM_FORCE_ABORT: When set to 1 all RTM transactions abort with EAX code 0 while the bit it set. Counter 3 becomes usable.	
		1:63	Reserved	

§

B Guidance on Specific Profilers

B.1 Linux* perf

Linux* perf is a PMU profiler integrated into the Linux kernel. With old or unpatched kernel versions, when Intel TSX is used, measurements using general purpose counter 4 (PMC3) may report incorrect values. This can be resolved with a kernel update.

The updated kernel exposes a new API as a "/sys/devices/cpu/allow_tsx_force_abort" sys file that takes the values of 0 or 1. If the value is set by the administrator to 0 then, the system will only support three general purpose counters for use with perf on affected systems with Intel TSX. In this case RTM transactions will not be forced to abort.

However; when the value of allow_tsx_force_abort is 1 then RTM transactions will force abort only while anyone on the system is running a perf session under the following 2 scenarios (exclusively):

1) Using PMC3 while profiling the process using Intel TSX instructions. Only root or the user running the program using can do this.

2) Using PMC3 while profiling the global system that includes the kernel and all processes running. This is done by using perf with the -a option. With default perf permission settings, a user needs to be root to use the –a option.

The default setting of the upstream kernel is to set allow_tsx_force_abort to 1. However, the Linux distribution may choose to set allow_tsx_force_abort to 0. When set to 1, Intel TSX users need to be aware of the possibility of performance variation due to Intel TSX instructions aborting associated with the use of perf under the 2 scenarios outlined above. To avoid this, system administrator can set allow_tsx_force_abort to 0, or avoid using perf in the scenarios above.

When allow_tsx_force_abort is set to 0 then any perf command lines defining groups with 4 generic counter events will need to be updated to use at most 3 generic events per group.

Tools that generate Linux perf groups (such as pmu-tools or Intel® VTune™) will also need to be updated.

B.2 Processor Counter Monitor

Processor Counter Monitor (PCM) is an application programming interface (API) and a set of tools based on the API used to monitor performance and energy metrics of Intel® Core™, Intel® Xeon®, Intel® Atom™ and Intel® Xeon Phi™ processors.

When ALLCTR (`force_RTM_abort`) mode is disabled, these PCM versions automatically limit the number of metrics and events being collected simultaneously. These versions also support enabling ALLCTR mode (`force_RTM_abort`) using a command line switch (see the output of `pcm.x --help`). Users can also control the modes programmatically with the `enableForceRTMAbortMode()` and `disableForceRTMAbortMode()` API calls.

§

[i] Eight counters when hyper-threading is disabled