White Paper
November 2008

Intel® Turbo Boost Technology in Intel® Core™ Microarchitecture (Nehalem) Based Processors

Table of Contents

1.0 Intel® Turbo Boost Technology

Intel® Core™ Microarchitecture (Nehalem) based processors incorporate a new feature: Intel® Turbo Boost technology. Under some configurations and workloads, Intel® Turbo Boost technology enables higher performance through the availability of increased core frequency.

Intel® Turbo Boost technology automatically allows processor cores to run faster than the base operating frequency if the processor is operating below rated power, temperature, and current specification limits. Intel® Turbo Boost technology can be engaged with any number of cores or logical processors enabled and active. This results in increased performance of both multi-threaded and single-threaded workloads.

It is possible for BIOS to contain a set-up option to enable or disable Intel® Turbo Boost technology and it operates under operating system (OS) control by engaging when the OS requests the highest performance state (P0). For ACPI aware operating systems, no changes are required to support Intel® Turbo Boost technology. The maximum frequency is dependent on the number of active cores and varies based on the specific configuration on a per processor number basis. The amount of time the processor spends in the Intel® Turbo Boost technology state will depend on workload and operating environment.

1.1 Overview

Intel® Turbo Boost technology is available only on supported processor versions. With Intel® Turbo Boost technology, the processor is capable of maximizing core frequency while ensuring that it does not exceed its electrical and thermal specifications. This means workloads that are naturally lower in power or lightly threaded may take advantage of headroom in the form of increased core frequency. Continual measurements of temperature, current draw, and power consumption are used to dynamically assess headroom.

1.2 Dependencies / Algorithm

Intel® Turbo Boost technology core frequency upside availability is ultimately constrained by power delivery limits, but within those constraints, it is limited by the following factors:

- The estimated current consumption of the processor
- The estimated power consumption of the processor
- The temperature of the processor

The number of active cores at any given instant dictates the upper limit of Intel® Turbo Boost technology. For this discussion, a core is considered 'active' if it is in the "C0" or "C1" state; cores in the "C3" or "C6" state are considered 'inactive'. The upper limits will vary on a per-processor number basis. For example, one particular processor may allow up to two frequency steps (266.66 MHz) when just one core is active and one frequency step (133.33 MHz) when two or more cores are active. Therefore, higher deep C-state residency ("C3" or "C6") on some cores will generally result in increased core frequency on the active cores.

The upper limits are further constrained by temperature, power, and current. These constraints are managed as a simple closed-loop control system. If measured temperature, power and current are all below factory-configured limits and the OS is requesting P0, the processor automatically steps up core frequency (+133.33 MHz) until it reaches the upper limit dictated by

the number of active cores. When temperature, power or current exceed factory configured limits and you are above the base operating frequency, the processor automatically steps down core frequency (-133.33 MHz) in order to reduce temperature, power and current. The processor then monitors temperature, power, and current and continuously re-evaluates.

Note: When Intel® Turbo Boost technology is requested by the OS, the processor will commonly operate between the max Intel® Turbo Boost technology frequency and the base operating frequency.

All active cores in the processor will operate at the same frequency. Even at frequencies above the base operating frequency, all active cores will run at the same frequency and voltage. Due to the way the BIOS and OS communicate Intel® Turbo Boost technology, software may never detect core clock frequencies above the base operating frequency. This is not reflective of actual core frequency. See Section 2 for more information on accurate methods for calculating core frequency.

2.0 Operating Frequency

Due to the methods used by BIOS to transparently expose Intel® Turbo Boost technology functions to the operating system, software that wishes to calculate (and/or display) the current operating core frequency may need to be updated.

2.1 Problem with Today's Software Reporting Frequency

Most currently available software that calculates frequency relies on the ACPI P-state structures that are provided by the BIOS or calls an application programming interface (API) that relies on these structures. The 'frequency' field in the P0 ACPI _PSS object does not reflect actual Intel® Turbo Boost technology frequencies. Instead, it always shows the P0 frequency as P1 frequency plus 1 MHz. Due to this, software that uses the P-state structures, directly or indirectly, will not properly display Intel® Turbo Boost technology frequencies.

NOTE: Operating Systems may also be impacted and may not report accurate frequency.

The algorithm below can be used to estimate the current operating frequency of a processor. The algorithm will calculate the average frequency of the active cores over a 1-second window. The time window can be adjusted if increased monitoring is desired.

2.2 Frequency Algorithm

The following are the general steps to implement the frequency-monitoring algorithm.

1. Compute and save the reference frequency. Bits 15:8 of the PLATFORM_INFO MSR (0CEH) are the "Base Operating Ratio." Multiply by the bus clock frequency (BCLK) to get the base operating frequency. The standard bus clock frequency is 133.33 MHz.
2. Enable fixed Architectural Performance Monitor counters 1 and 2 in the Global Performance Counter Control IA32_PERF_GLOBAL_CTRL (38FH) and the Fixed-Function Performance Counter Control IA32_FIXED_CTR_CTL (38DH). Repeat this step for each logical processor in the system. Fixed Counter 1 (CPU_CLK_UNHALTED.CORE) counts the number of core cycles while the core is not in a halted state. Fixed Counter 2 (CPU_CLK_UNHALTED.REF) counts the number of reference (base operating frequency) cycles while the core is not in a halted state.
3. Configure an auto-rearming timer with 1-second duration using an OS API.
4. Repeat steps 5 through 10 until the application exits.
5. Wait for the 1-second timer to expire.
6. Read the Fixed-Function Performance Counter 1 IA32_FIXED_CTR1 (30AH) and the Fixed-Function Performance Counter 2 IA32_FIXED_CTR2 (30BH). Repeat this step for each logical processor in the system.
7. Compute the number of unhalted core cycles and unhalted reference cycles that have expired since the last iteration by subtracting the previously sampled values from the currently sampled values.
8. Compute the actual frequency value for each logical processor as follows: $F_{current}$ = Base Operating Frequency * (Unhalted Core Cycles / Unhalted Ref Cycles)

9. Update with GUI to display the newly computed values (for accurate display, choose the maximum frequency of all logical processors).
10. Save Unhalted Core Cycles and Unhalted Ref Cycles for use in the next iteration.

NOTE: Step 9 is required due to the fact that idle cores will be woken up to read their frequency and that will change the results of those idle cores. For displaying the frequency of the active cores, ignore these lower results.

2.3 Determining Maximum Possible Intel® Turbo Boost Technology Upside

It is possible for software to read the maximum possible Intel® Turbo Boost technology frequency via MSR 1ADH.

On Intel® Core i7 processors: Bits 31:24 of this MSR contain the max ratio with 4 active cores, bits 23:16 of this MSR contain the max ratio with 3 active cores, bits 15:8 of this MSR contain the max ratio with 2 active cores and bits 7:0 contain the max ratio with 1 active core. This ratio needs to be multiplied by bus clock frequency (BCLK) to get the actual frequency. The standard bus clock frequency is 133.33 MHz.

NOTE: This MSR is NOT architectural, which means its use is restricted to BIOS and other platform-specific software. This MSR is not guaranteed to exist on future products or be in the same MSR location if it is available.

2.4 Displaying Active Cores

It is possible for software to calculate the percentage of time a core is in C0. This may be accomplished with the following algorithm:

1. Enable fixed Architectural Performance Monitor counter 2 in the Global Performance Counter Control IA32_PERF_GLOBAL_CTRL (3BFH) and the Fixed-Function Performance Counter Control IA32_FIXED_CTR_CTL (38DH). Repeat this step for each logical processor in the system. Fixed counter 2 (CPU_CLK_UNHALTED.REF) counts the number of reference cycles while the core is not in a halt state.
2. Configure an auto-rearming timer with 1-second duration using an OS API.
3. Repeat steps 4 – 7 until the application exits.
4. Read the Fixed-Function Performance Counter 2 IA32_FIXED_CTR2 (30BH) and the Time Stamp Counter (10H).
5. Compute the number of unhalted reference cycles and Time Stamp Counter cycles that have expired since the last iteration by subtracting the previously sampled values from the currently sampled values.
6. Calculate the percent of time in C0: %C0_time = Unhalted Ref Cycles/TSC.
7. Save Unhalted Ref Cycles and TSC for use in the next iteration.

2.5 Displaying Time Spent In C3

It is possible for software to calculate the percentage of time a core is in C3. This is done with the following algorithm:

1. Configure an auto-rearming timer with 1-second duration using an OS API.
2. Repeat steps 3-6 until the application exits.
3. Read the CORE_C3_RESIDENCY (3FCH) MSR and the Time Stamp Counter (10H).
4. Compute the number of C3 reference cycles and Time Stamp Counter cycles that have expired since the last iteration by subtracting the previously sampled values from the currently sampled values.
5. Calculate the percent of time in C3: %C3_time = Core_C3_Residency/TSC.

6. Save Unhalted Ref Cycles and TSC for use in the next iteration.

NOTE: The C3 residency MSR is NOT architectural which means its use is restricted to BIOS and other platform-specific software. This MSR is not guaranteed to exist on future products or be in the same MSR locations if it is available.

2.6 Displaying Time Spent In C6

It is possible for software to calculate the percentage of time a core is in C6. This is done with the following algorithm:

1. Configure an auto-rearming timer with 1-second duration using an OS API.
2. Repeat steps 3-6 until the application exits.
3. Read the CORE_C6_RESIDENCY (3FDH) MSR and the Time Stamp Counter (10H).
4. Compute the number of C6 reference cycles and Time Stamp Counter cycles that have expired since the last iteration by subtracting the previously sampled values from the currently sampled values.
5. Calculate the percent of time in C6: %C6_time = Core_C6_Residency/TSC.

NOTE: The C6 residency MSR is NOT architectural which means its use is restricted to BIOS and other platform-specific software. This MSR is not guaranteed to exist on future products or be in the same MSR locations if it is available.

3.0 C-State Definitions

Intel® processors based on *Nehalem* microarchitecture support core C0, C1, C3, and C6. C0 and C1 are always supported; the availability of the remaining C-states may vary by processor number. Any core within the processor can go into any C-state independent of the state of the other cores.

Note: The behavior in a particular C-state on Intel® processors based on *Nehalem* microarchitecture may be different from C-states referred to with the same number on previous products. One reason for this is the platform changes, including the move to the Intel® Quick Path Interconnect, which results in the elimination of the STPCLK#, SLP#, and DPSLP# signals.

3.1 C0

C0 is defined as the active state. While in C0, instructions are being executed by the core. For Intel® Turbo Boost technology, a core in C0 is considered an active core.

3.2 C1

C1 is defined as the halt state. While in C1, no instructions are being executed. For Intel® Turbo Boost technology, a core in C1 is considered an active core. This idle state is generally classified as "ACPI C1."

3.3 C3

While in C3 the core PLLs are turned off, and all the core caches are flushed. For Intel® Turbo Boost technology, a core in C3 is considered an inactive core. This idle state may be classified as "ACPI C2" or "ACPI C3" depending on processor number, BIOS, and operating system.

3.4 C6

While in C6, the core PLLs are turned off, the core caches are flushed and the core state is saved to the Last Level Cache. Power Gates are used to reduce power consumption to close to zero. For Intel® Turbo Boost technology, a core in C6 is considered an inactive core. This idle state may be classified as "ACPI C2" or "ACPI C3" depending on processor number, BIOS, and operating system.

Legal Disclaimer

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.

UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

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. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document 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.

Intel® Turbo Boost Technology requires a PC with a processor with Intel Turbo Boost Technology capability. Intel Turbo Boost Technology performance varies depending on hardware, software and overall system configuration. Check with your PC manufacturer on whether your system delivers Intel Turbo Boost Technology. For more information, see www.intel.com.

*Other names and brands may be claimed as the property of others