Intel® 64 and IA32 Architectures Performance Monitoring Events

2017 December
Revision 1.0

Document Number:335279-001

Revision History

Document Number	Revision Number	Description	Date
334525-001	1.0	Initial release of the document	2017 December

Performance Monitoring Events

Glossary

Glossary Items as listed below:

Name	Description
EventSelect	Set the EventSelect bits to the value specified. These bits are defined in Chapter 18.2.1.1 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B.
UMask	Set the UMask bits to the value specified. These bits are defined in Chapter 18.2.1.1 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B.
USR	Set the USR bit to the value specified. This bit is defined in Chapter 18.2.1.1 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B. Unless specified, set the bit according to the desired scope. When set, the counter will count events when the logical processor is operating at privilege level 0. This flag can be used with the USR flag.
OS	Set the OS bit to the value specified. This bit is defined in Chapter 18.2.1.1 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B. Unless specified, set the bit according to the desired scope. When set, the counter will count events when the logical processor is operating at privilege levels 1, 2 or 3. This flag can be used with the OS flag.
EdgeDetect	Set the EdgeDetect bit to the value specified. This bit is defined in Chapter 18.2.1.1 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B. Unless specified, set this bit to 0.
AnyThread	Set the AnyThread bit to the value specified. This bit is defined in Chapter 18.2.1.1 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B. Unless specified, set this bit to 0.
Invert	Set the Invert bit to the value specified. This bit is defined in Chapter 18.2.1.1 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B. Unless specified, set this bit to 0.
CMask	Set the CMask bits to the value specified. These bits are defined in Chapter 18.2.1.1 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B.
MSR_PEBS_FRONTEND	Set the MSR_PEBS_FRONTEND bits to the value specified. These bits are defined in Chapter 18.13.1.4 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B.
MSR_PEBS_LD_LAT_THRESHOLD	Set the MSR_PEBS_LD_LAT_THRESHOLD bits to the value specified. These bits are defined in Chapter 18.8.1.2 and the relevant PEBS sub-sections across the core PMU sections in Chapter 18, Performance Monitoring.

Architectural	This event is architecturally defined as described in Chapter 18.2 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B.
Fixed	This event uses a Fixed-function Performance Counter Register, as defined in Chapter 18.2.2 of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B.
Precise	The Processor Event Based Sampling (PEBS) facility is capable of capturing the exact machine state after the instruction that experienced this event retires, including R/EIP of the next instruction. In some generations, information about the instruction that experienced the event is also available. See Section 18.4.4, "Processor Event Based Sampling (PEBS)," and the relevant PEBS sub-sections across the core PMU sections in Chapter 18, "Performance Monitoring."
Deprecated	In future generations, this event has its name changed or is no longer supported. It remains supported in this generation.

Architectural Performance Monitoring Events

Architectural Performance Monitoring Events

Architectural performance events are introduced in Intel Core Solo and Intel Core Duo processors. They are also supported on processors based on Intel Core microarchitecture. Table below lists pre-defined architectural performance events that can be configured using general-purpose performance counters and associated event-select registers.

Table 1: Architectural Performance Events

Event Name	
Configuration	Description
UnHalted Core Cycles	
EventSel=3CH, UMask=00H	Counts core clock cycles whenever the logical processor is in C0 state (not halted). The frequency of this event varies with state transitions in the core.
UnHalted Reference Cycles	
EventSel=3CH, UMask=01H	Counts at a fixed frequency whenever the logical processor is in C0 state (not halted).
Instructions Retired	
EventSel=C0H, UMask=00H	Counts when the last uop of an instruction retires.
LLC Reference	
EventSel=2EH, UMask=4FH	Accesses to the LLC, in which the data is present (hit) or not present (miss).
LLC Misses	
EventSel=2EH, UMask=41H	Accesses to the LLC in which the data is not present (miss).
Branch Instruction Retired	
EventSel=C4H, UMask=00H	Counts when the last uop of a branch instruction retires.
Branch Misses Retired	
EventSel=C5H, UMask=00H	Counts when the last uop of a branch instruction retires which corrected misprediction of the branch prediction hardware at execution time .

Note - Current implementations count at core crystal clock, TSC, or bus clock frequency. Fixed-function performance counters count only events defined in table below.

Table 1: Architectural Fixed-Function Performance Counter and Pre-defined Performance Events.

Event Mask Mnemonic	
Fixed-Function Performance Counter	Description
INST_RETIRED.ANY	
Addr=309H, IA32_PERF_FIXED_CTR0	This event counts the number of instructions that retire execution.For instructions that consist of multiple microops, this event counts the retirement of the last micro - op of the instruction.The counter continues counting during hardware interrupts, traps, and inside interrupt handlers .
CPU_CLK_UNHALTED.THREAD /CPU_CLK_UNHALTED.CORE /CPU_CLK_UNHALTED.THREAD_ANY	
Addr=30AH, IA32_PERF_FIXED_CTR1	The CPU_CLK_UNHALTED.THREAD event counts the number of core cycles while the logical processor is not in a halt state. If there is only one logical processor in a processor core, CPU_CLK_UNHALTED.CORE counts the unhalted cycles of the processor core.If there are more than one logical processor in a processor core, CPU_CLK_UNHALTED.THREAD_ANY is supported by programming IA32_FIXED_CTR_CTRL[bit 6]AnyThread = 1. The core frequency may change from time to time due to transitions associated with Enhanced Intel SpeedStep Technology or TM2. For this reason this event may have a changing ratio with regards to time.
CPU_CLK_UNHALTED.REF_TSC	
Addr=30BH, IA32_PERF_FIXED_CTR2	This event counts the number of reference cycles at the TSC rate when the core is not in a halt state and not in a TM stop-clock state. The core enters the halt state when it is running the HLT instruction or the MWAIT instruction. This event is not affected by core frequency changes (e.g., P states) but counts at the same frequency as the time stamp counter. This event can approximate elapsed time while the core was not in a halt state and not in a TM stopclock state.

Performance Monitoring Intel® Core™ Processors

Performance Monitoring Events based on Skylake Microarchitecture - 6th Generation Intel® Core™ Processor and 7th Generation Intel® Core™ Processor

6th Generation Intel® Core™ processors are based on the Skylake microarchitecture. 7th Generation Intel® Core™ processors are based on the Kaby Lake microarchitecture. Performance-monitoring events in the processor core for these processors are listed in the table below.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed	Counts the number of instructions retired from execution. For instructions that consist of multiple micro-ops, Counts the retirement of the last micro-op of the instruction. Counting continues during hardware interrupts, traps, and inside interrupt handlers. Notes: INST_RETIRED.ANY is counted by a designated fixed counter, leaving the four (eight when Hyperthreading is disabled) programmable counters available for other events. INST_RETIRED.ANY_P is counted by a programmable counter and it is an architectural performance event. Counting: Faulting executions of GETSEC/VM entry/VM Exit/MWait will not count as retired instructions.
CPU_CLK_UNHALTED.THREAD	
Architectural, Fixed	Counts the number of core cycles while the thread is not in a halt state. The thread enters the halt state when it is running the HLT instruction. This event is a component in many key event ratios. The core frequency may change from time to time due to transitions associated with Enhanced Intel SpeedStep Technology or TM2. For this reason this event may have a changing ratio with regards to time. When the core frequency is constant, this event can approximate elapsed time while the core was not in the halt state. It is counted on a dedicated fixed counter, leaving the four (eight when Hyperthreading is disabled) programmable counters available for other events.
CPU_CLK_UNHALTED.THREAD_ANY	
AnyThread=1, Architectural, Fixed	Core cycles when at least one thread on the physical core is not in halt state.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
CPU_CLK_UNHALTED.REF_TSC	
Architectural, Fixed	Counts the number of reference cycles when the core is not in a halt state. The core enters the halt state when it is running the HLT instruction or the MWAIT instruction. This event is not affected by core frequency changes (for example, P states, TM2 transitions) but has the same incrementing frequency as the time stamp counter. This event can approximate elapsed time while the core was not in a halt state. This event has a constant ratio with the CPU_CLK_UNHALTED.REF_XCLK event. It is counted on a dedicated fixed counter, leaving the four (eight when Hyperthreading is disabled) programmable counters available for other events. Note: On all current platforms this event stops counting during 'throttling (TM)' states duty off periods the processor is 'halted'. The counter update is done at a lower clock rate then the core clock the overflow status bit for this counter may appear 'sticky'. After the counter has overflowed and software clears the overflow status bit and resets the counter to less than MAX. The reset value to the counter is not clocked immediately so the overflow status bit will flip 'high (1)' and generate another PMI (if enabled) after which the reset value gets clocked into the counter. Therefore, software will get the interrupt, read the overflow status bit '1 for bit 34 while the counter value is less than MAX. Software should ignore this case.
LD_BLOCKS.STORE_FORWARD	
EventSel=03H, UMask=02H	Counts how many times the load operation got the true Block-on-Store blocking code preventing store forwarding. This includes cases when:a. preceding store conflicts with the load (incomplete overlap),b. store forwarding is impossible due to u-arch limitations,c. preceding lock RMW operations are not forwarded,d. store has the no-forward bit set (uncacheable/page-split/masked stores),e. all-blocking stores are used (mostly, fences and port I/O), and others.The most common case is a load blocked due to its address range overlapping with a preceding smaller uncompleted store. Note: This event does not take into account cases of out-of-SW-control (for example, SbTailHit), unknown physical STA, and cases of blocking loads on store due to being non-WB memory type or a lock. These cases are covered by other events. See the table of not supported store forwards in the Optimization Guide.
LD_BLOCKS.NO_SR	
EventSel=03H, UMask=08H	The number of times that split load operations are temporarily blocked because all resources for handling the split accesses are in use.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
LD_BLOCKS_PARTIAL.ADDRESS_ALIAS	
EventSel=07H, UMask=01H	Counts false dependencies in MOB when the partial comparison upon loose net check and dependency was resolved by the Enhanced Loose net mechanism. This may not result in high performance penalties. Loose net checks can fail when loads and stores are 4k aliased.
DTLB_LOAD_MISSES.MISS_CAUSES_A_WALK	
EventSel=08H, UMask=01H	Counts demand data loads that caused a page walk of any page size (4K/2M/4M/1G). This implies it missed in all TLB levels, but the walk need not have completed.
DTLB_LOAD_MISSES.WALK_COMPLETED_4K	
EventSel=08H, UMask=02H	Counts page walks completed due to demand data loads whose address translations missed in the TLB and were mapped to 4K pages. The page walks can end with or without a page fault.
DTLB_LOAD_MISSES.WALK_COMPLETED_2M_4M	
EventSel=08H, UMask=04H	Counts page walks completed due to demand data loads whose address translations missed in the TLB and were mapped to 2M/4M pages. The page walks can end with or without a page fault.
DTLB_LOAD_MISSES.WALK_COMPLETED_1G	
EventSel=08H, UMask=08H	Counts page walks completed due to demand data loads whose address translations missed in the TLB and were mapped to 4K pages. The page walks can end with or without a page fault.
DTLB_LOAD_MISSES.WALK_COMPLETED	
EventSel=08H, UMask=0EH	Counts demand data loads that caused a completed page walk of any page size (4K/2M/4M/1G). This implies it missed in all TLB levels. The page walk can end with or without a fault.
DTLB_LOAD_MISSES.WALK_PENDING	
EventSel=08H, UMask=10H	Counts 1 per cycle for each PMH that is busy with a page walk for a load. EPT page walk duration are excluded in Skylake microarchitecture. .
DTLB_LOAD_MISSES.WALK_ACTIVE	
EventSel=08H, UMask=10H, CMask=1	Counts cycles when at least one PMH (Page Miss Handler) is busy with a page walk for a load.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
DTLB_LOAD_MISSES.STLB_HIT	
EventSel=08H, UMask=20H	Counts loads that miss the DTLB (Data TLB) and hit the STLB (Second level TLB).
INT_MISC.RECOVERY_CYCLES	
EventSel=0DH, UMask=01H	Core cycles the Resource allocator was stalled due to recovery from an earlier branch misprediction or machine clear event.
INT_MISC.RECOVERY_CYCLES_ANY	
EventSel=0DH, UMask=01H, AnyThread=1	Core cycles the allocator was stalled due to recovery from earlier clear event for any thread running on the physical core (e.g. misprediction or memory nuke).
INT_MISC.CLEAR_RESTEER_CYCLES	
EventSel=0DH, UMask=80H	Cycles the issue-stage is waiting for front-end to fetch from resteered path following branch misprediction or machine clear events.
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=01H	Counts the number of uops that the Resource Allocation Table (RAT) issues to the Reservation Station (RS).
UOPS_ISSUED.STALL_CYCLES	
EventSel=0EH, UMask=01H, Invert=1, CMask=1	Counts cycles during which the Resource Allocation Table (RAT) does not issue any Uops to the reservation station (RS) for the current thread.
UOPS_ISSUED.VECTOR_WIDTH_MISMATCH	
EventSel=0EH, UMask=02H	Counts the number of Blend Uops issued by the Resource Allocation Table (RAT) to the reservation station (RS) in order to preserve upper bits of vector registers. Starting with the Skylake microarchitecture, these Blend uops are needed since every Intel SSE instruction executed in Dirty Upper State needs to preserve bits 128-255 of the destination register. For more information, refer to “Mixing Intel AVX and Intel SSE Code” section of the Optimization Guide.
UOPS_ISSUED.SLOW_LEA	
EventSel=0EH, UMask=20H	Number of slow LEA uops being allocated. A uop is generally considered SlowLea if it has 3 sources (e.g. 2 sources + immediate) regardless if as a result of LEA instruction or not.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
ARITH.DIVIDER_ACTIVE	
EventSel=14H, UMask=01H, CMask=1	Cycles when divide unit is busy executing divide or square root operations. Accounts for integer and floating-point operations.
L2_RQSTS.DEMAND_DATA_RD_MISS	
EventSel=24H, UMask=21H	Counts the number of demand Data Read requests that miss L2 cache. Only not rejected loads are counted.
L2_RQSTS.RFO_MISS	
EventSel=24H, UMask=22H	Counts the RFO (Read-for-Ownership) requests that miss L2 cache.
L2_RQSTS.CODE_RD_MISS	
EventSel=24H, UMask=24H	Counts L2 cache misses when fetching instructions.
L2_RQSTS.ALL_DEMAND_MISS	
EventSel=24H, UMask=27H	Demand requests that miss L2 cache.
L2_RQSTS.PF_MISS	
EventSel=24H, UMask=38H	Counts requests from the L1/L2/L3 hardware prefetchers or Load software prefetches that miss L2 cache.
L2_RQSTS.MISS	
EventSel=24H, UMask=3FH	All requests that miss L2 cache.
L2_RQSTS.DEMAND_DATA_RD_HIT	
EventSel=24H, UMask=41H	Counts the number of demand Data Read requests that hit L2 cache. Only non rejected loads are counted.
L2_RQSTS.RFO_HIT	
EventSel=24H, UMask=42H	Counts the RFO (Read-for-Ownership) requests that hit L2 cache.
L2_RQSTS.CODE_RD_HIT	
EventSel=24H, UMask=44H	Counts L2 cache hits when fetching instructions, code reads.
L2_RQSTS.PF_HIT	
EventSel=24H, UMask=D8H	Counts requests from the L1/L2/L3 hardware prefetchers or Load software prefetches that hit L2 cache.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
L2_RQSTS.ALL_DEMAND_DATA_RD	
EventSel=24H, UMask=E1H	Counts the number of demand Data Read requests (including requests from L1D hardware prefetchers). These loads may hit or miss L2 cache. Only non rejected loads are counted.
L2_RQSTS.ALL_RFO	
EventSel=24H, UMask=E2H	Counts the total number of RFO (read for ownership) requests to L2 cache. L2 RFO requests include both L1D demand RFO misses as well as L1D RFO prefetches.
L2_RQSTS.ALL_CODE_RD	
EventSel=24H, UMask=E4H	Counts the total number of L2 code requests.
L2_RQSTS.ALL_DEMAND_REFERENCES	
EventSel=24H, UMask=E7H	Demand requests to L2 cache.
L2_RQSTS.ALL_PF	
EventSel=24H, UMask=F8H	Counts the total number of requests from the L2 hardware prefetchers.
L2_RQSTS.REFERENCES	
EventSel=24H, UMask=FFH	All L2 requests.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	Counts core-originated cacheable requests that miss the L3 cache (Longest Latency cache). Requests include data and code reads, Reads-for-Ownership (RFOs), speculative accesses and hardware prefetches from L1 and L2. It does not include all misses to the L3. .
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	Counts core-originated cacheable requests to the L3 cache (Longest Latency cache). Requests include data and code reads, Reads-for-Ownership (RFOs), speculative accesses and hardware prefetches from L1 and L2. It does not include all accesses to the L3. .
SW_PREFETCH_ACCESS.NTA	
EventSel=32H, UMask=01H	Number of PREFETCHNTA instructions executed.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
SW_PREFETCH_ACCESS.T0	
EventSel=32H, UMask=02H	Number of PREFETCHT0 instructions executed.
SW_PREFETCH_ACCESS.T1_T2	
EventSel=32H, UMask=04H	Number of PREFETCHT1 or PREFETCHT2 instructions executed.
SW_PREFETCH_ACCESS.PREFETCHW	
EventSel=32H, UMask=08H	Number of PREFETCHW instructions executed.
CPU_CLK_UNHALTED.THREAD_P	
EventSel=3CH, UMask=00H, Architectural	This is an architectural event that counts the number of thread cycles while the thread is not in a halt state. The thread enters the halt state when it is running the HLT instruction. The core frequency may change from time to time due to power or thermal throttling. For this reason, this event may have a changing ratio with regards to wall clock time.
CPU_CLK_UNHALTED.THREAD_P_ANY	
EventSel=3CH, UMask=00H, AnyThread=1, Architectural	Core cycles when at least one thread on the physical core is not in halt state.
CPU_CLK_UNHALTED.RING0_TRANS	
EventSel=3CH, UMask=00H, USR=0,OS=1, EdgeDetect=1, CMask=1, Architectural	Counts when the Current Privilege Level (CPL) transitions from ring 1, 2 or 3 to ring 0 (Kernel).
CPU_CLK_THREAD_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	Core crystal clock cycles when the thread is unhalted. **Note:Also defined at CPU_CLK_UNHALTED.REF_XCLK.*
CPU_CLK_THREAD_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Core crystal clock cycles when at least one thread on the physical core is unhalted. **Note:Also defined at CPU_CLK_UNHALTED.REF_XCLK_ANY.*
CPU_CLK_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	Core crystal clock cycles when the thread is unhalted. **Note:Also defined at CPU_CLK_THREAD_UNHALTED.REF_XCLK.*
CPU_CLK_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Core crystal clock cycles when at least one thread on the physical core is unhalted. **Note:Also defined at CPU_CLK_THREAD_UNHALTED.REF_XCLK_ANY.*

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
CPU_CLK_THREAD_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Core crystal clock cycles when this thread is unhalted and the other thread is halted.
CPU_CLK_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Core crystal clock cycles when this thread is unhalted and the other thread is halted.
L1D_PEND_MISS.PENDING	
EventSel=48H, UMask=01H	Counts duration of L1D miss outstanding, that is each cycle number of Fill Buffers (FB) outstanding required by Demand Reads. FB either is held by demand loads, or it is held by non-demand loads and gets hit at least once by demand. The valid outstanding interval is defined until the FB deallocation by one of the following ways: from FB allocation, if FB is allocated by demand from the demand Hit FB, if it is allocated by hardware or software prefetch.Note: In the L1D, a Demand Read contains cacheable or noncacheable demand loads, including ones causing cache-line splits and reads due to page walks resulted from any request type.
L1D_PEND_MISS.PENDING_CYCLES	
EventSel=48H, UMask=01H, CMask=1	Counts duration of L1D miss outstanding in cycles.
L1D_PEND_MISS.PENDING_CYCLES_ANY	
EventSel=48H, UMask=01H, AnyThread=1, CMask=1	Cycles with L1D load Misses outstanding from any thread on physical core.
L1D_PEND_MISS.FB_FULL	
EventSel=48H, UMask=02H	Number of times a request needed a FB (Fill Buffer) entry but there was no entry available for it. A request includes cacheable/uncacheable demands that are load, store or SW prefetch instructions.
DTLB_STORE_MISSES.MISS_CAUSES_A_WALK	
EventSel=49H, UMask=01H	Counts demand data stores that caused a page walk of any page size (4K/2M/4M/1G). This implies it missed in all TLB levels, but the walk need not have completed.
DTLB_STORE_MISSES.WALK_COMPLETED_4K	
EventSel=49H, UMask=02H	Counts page walks completed due to demand data stores whose address translations missed in the TLB and were mapped to 4K pages. The page walks can end with or without a page fault.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
DTLB_STORE_MISSES.WALK_COMPLETED_2M_4M	
EventSel=49H, UMask=04H	Counts page walks completed due to demand data stores whose address translations missed in the TLB and were mapped to 2M/4M pages. The page walks can end with or without a page fault.
DTLB_STORE_MISSES.WALK_COMPLETED_1G	
EventSel=49H, UMask=08H	Counts page walks completed due to demand data stores whose address translations missed in the TLB and were mapped to 1G pages. The page walks can end with or without a page fault.
DTLB_STORE_MISSES.WALK_COMPLETED	
EventSel=49H, UMask=0EH	Counts demand data stores that caused a completed page walk of any page size (4K/2M/4M/1G). This implies it missed in all TLB levels. The page walk can end with or without a fault.
DTLB_STORE_MISSES.WALK_PENDING	
EventSel=49H, UMask=10H	Counts 1 per cycle for each PMH that is busy with a page walk for a store. EPT page walk duration are excluded in Skylake microarchitecture. .
DTLB_STORE_MISSES.WALK_ACTIVE	
EventSel=49H, UMask=10H, CMask=1	Counts cycles when at least one PMH (Page Miss Handler) is busy with a page walk for a store.
DTLB_STORE_MISSES.STLB_HIT	
EventSel=49H, UMask=20H	Stores that miss the DTLB (Data TLB) and hit the STLB (2nd Level TLB).
LOAD_HIT_PRE.SW_PF	
EventSel=4CH, UMask=01H	Counts all not software-prefetch load dispatches that hit the fill buffer (FB) allocated for the software prefetch. It can also be incremented by some lock instructions. So it should only be used with profiling so that the locks can be excluded by ASM (Assembly File) inspection of the nearby instructions.
EPT.WALK_PENDING	
EventSel=4FH, UMask=10H	Counts cycles for each PMH (Page Miss Handler) that is busy with an EPT (Extended Page Table) walk for any request type.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
L1D.REPLACEMENT	
EventSel=51H, UMask=01H	Counts L1D data line replacements including opportunistic replacements, and replacements that require stall-for-replace or block-for-replace.
TX_MEM.ABORT_CONFLICT	
EventSel=54H, UMask=01H	Number of times a TSX line had a cache conflict.
TX_MEM.ABORT_CAPACITY	
EventSel=54H, UMask=02H	Number of times a transactional abort was signaled due to a data capacity limitation for transactional reads or writes.
TX_MEM.ABORT_HLE_STORE_TO_ELIDED_LOCK	
EventSel=54H, UMask=04H	Number of times a TSX Abort was triggered due to a non-release/commit store to lock.
TX_MEM.ABORT_HLE_ELISION_BUFFER_NOT_EMPTY	
EventSel=54H, UMask=08H	Number of times a TSX Abort was triggered due to commit but Lock Buffer not empty.
TX_MEM.ABORT_HLE_ELISION_BUFFER_MISMATCH	
EventSel=54H, UMask=10H	Number of times a TSX Abort was triggered due to release/commit but data and address mismatch.
TX_MEM.ABORT_HLE_ELISION_BUFFER_UNSUPPORTED_ALIGNMENT	
EventSel=54H, UMask=20H	Number of times a TSX Abort was triggered due to attempting an unsupported alignment from Lock Buffer.
TX_MEM.HLE_ELISION_BUFFER_FULL	
EventSel=54H, UMask=40H	Number of times we could not allocate Lock Buffer.
TX_EXEC.MISC1	
EventSel=5DH, UMask=01H	Counts the number of times a class of instructions that may cause a transactional abort was executed. Since this is the count of execution, it may not always cause a transactional abort.
TX_EXEC.MISC2	
EventSel=5DH, UMask=02H	Unfriendly TSX abort triggered by a vzeroupper instruction.
TX_EXEC.MISC3	
EventSel=5DH, UMask=04H	Unfriendly TSX abort triggered by a nest count that is too deep.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
TX_EXEC.MISC4	
EventSel=5DH, UMask=08H	RTM region detected inside HLE.
TX_EXEC.MISC5	
EventSel=5DH, UMask=10H	Counts the number of times an HLE XACQUIRE instruction was executed inside an RTM transactional region.
RS_EVENTS.EMPTY_CYCLES	
EventSel=5EH, UMask=01H	Counts cycles during which the reservation station (RS) is empty for the thread.; Note: In ST-mode, not active thread should drive 0. This is usually caused by severely costly branch mispredictions, or allocator/FE issues.
RS_EVENTS.EMPTY_END	
EventSel=5EH, UMask=01H, EdgeDetect=1, Invert=1, CMask=1	Counts end of periods where the Reservation Station (RS) was empty. Could be useful to precisely locate front-end Latency Bound issues.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD	
EventSel=60H, UMask=01H	Counts the number of offcore outstanding Demand Data Read transactions in the super queue (SQ) every cycle. A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor. See the corresponding Umask under OFFCORE_REQUESTS.Note: A prefetch promoted to Demand is counted from the promotion point.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_DATA_RD	
EventSel=60H, UMask=01H, CMask=1	Counts cycles when offcore outstanding Demand Data Read transactions are present in the super queue (SQ). A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor (SQ de-allocation).
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD_GE_6	
EventSel=60H, UMask=01H, CMask=6	Cycles with at least 6 offcore outstanding Demand Data Read transactions in uncore queue.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_CODE_RD	
EventSel=60H, UMask=02H	Counts the number of offcore outstanding Code Reads transactions in the super queue every cycle. The 'Offcore outstanding' state of the transaction lasts from the L2 miss until the sending transaction completion to requestor (SQ deallocation). See the corresponding Umask under OFFCORE_REQUESTS.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_CODE_RD	
EventSel=60H, UMask=02H, CMask=1	Counts the number of offcore outstanding Code Reads transactions in the super queue every cycle. The 'Offcore outstanding' state of the transaction lasts from the L2 miss until the sending transaction completion to requestor (SQ deallocation). See the corresponding Umask under OFFCORE_REQUESTS.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_RFO	
EventSel=60H, UMask=04H	Counts the number of offcore outstanding RFO (store) transactions in the super queue (SQ) every cycle. A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor (SQ de-allocation). See corresponding Umask under OFFCORE_REQUESTS.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_RFO	
EventSel=60H, UMask=04H, CMask=1	Counts the number of offcore outstanding demand rfo Reads transactions in the super queue every cycle. The 'Offcore outstanding' state of the transaction lasts from the L2 miss until the sending transaction completion to requestor (SQ deallocation). See the corresponding Umask under OFFCORE_REQUESTS.
OFFCORE_REQUESTS_OUTSTANDING.ALL_DATA_RD	
EventSel=60H, UMask=08H	Counts the number of offcore outstanding cacheable Core Data Read transactions in the super queue every cycle. A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor (SQ de-allocation). See corresponding Umask under OFFCORE_REQUESTS.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DATA_RD	
EventSel=60H, UMask=08H, CMask=1	Counts cycles when offcore outstanding cacheable Core Data Read transactions are present in the super queue. A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor (SQ de-allocation). See corresponding Umask under OFFCORE_REQUESTS.
OFFCORE_REQUESTS_OUTSTANDING.L3_MISS_DEMAND_DATA_RD	
EventSel=60H, UMask=10H	Counts number of Offcore outstanding Demand Data Read requests that miss L3 cache in the superQ every cycle.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_L3_MISS_DEMAND_DATA_RD	
EventSel=60H, UMask=10H, CMask=1	Cycles with at least 1 Demand Data Read requests who miss L3 cache in the superQ.
OFFCORE_REQUESTS_OUTSTANDING.L3_MISS_DEMAND_DATA_RD_GE_6	
EventSel=60H, UMask=10H, CMask=6	Cycles with at least 6 Demand Data Read requests that miss L3 cache in the superQ.
IDQ.MITE_UOPS	
EventSel=79H, UMask=04H	Counts the number of uops delivered to Instruction Decode Queue (IDQ) from the MITE path. Counting includes uops that may 'bypass' the IDQ. This also means that uops are not being delivered from the Decode Stream Buffer (DSB).
IDQ.MITE_CYCLES	
EventSel=79H, UMask=04H, CMask=1	Counts cycles during which uops are being delivered to Instruction Decode Queue (IDQ) from the MITE path. Counting includes uops that may 'bypass' the IDQ.
IDQ.DSB_UOPS	
EventSel=79H, UMask=08H	Counts the number of uops delivered to Instruction Decode Queue (IDQ) from the Decode Stream Buffer (DSB) path. Counting includes uops that may 'bypass' the IDQ.
IDQ.DSB_CYCLES	
EventSel=79H, UMask=08H, CMask=1	Counts cycles during which uops are being delivered to Instruction Decode Queue (IDQ) from the Decode Stream Buffer (DSB) path. Counting includes uops that may 'bypass' the IDQ.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
IDQ.MS_DSB_CYCLES	
EventSel=79H, UMask=10H, CMask=1	Counts cycles during which uops initiated by Decode Stream Buffer (DSB) are being delivered to Instruction Decode Queue (IDQ) while the Microcode Sequencer (MS) is busy. Counting includes uops that may 'bypass' the IDQ.
IDQ.ALL_DSB_CYCLES_4_UOPS	
EventSel=79H, UMask=18H, CMask=4	Counts the number of cycles 4 uops were delivered to Instruction Decode Queue (IDQ) from the Decode Stream Buffer (DSB) path. Count includes uops that may 'bypass' the IDQ.
IDQ.ALL_DSB_CYCLES_ANY_UOPS	
EventSel=79H, UMask=18H, CMask=1	Counts the number of cycles uops were delivered to Instruction Decode Queue (IDQ) from the Decode Stream Buffer (DSB) path. Count includes uops that may 'bypass' the IDQ.
IDQ.MS_MITE_UOPS	
EventSel=79H, UMask=20H	Counts the number of uops initiated by MITE and delivered to Instruction Decode Queue (IDQ) while the Microcode Sequencer (MS) is busy. Counting includes uops that may 'bypass' the IDQ.
IDQ.ALL_MITE_CYCLES_4_UOPS	
EventSel=79H, UMask=24H, CMask=4	Counts the number of cycles 4 uops were delivered to the Instruction Decode Queue (IDQ) from the MITE (legacy decode pipeline) path. Counting includes uops that may 'bypass' the IDQ. During these cycles uops are not being delivered from the Decode Stream Buffer (DSB).
IDQ.ALL_MITE_CYCLES_ANY_UOPS	
EventSel=79H, UMask=24H, CMask=1	Counts the number of cycles uops were delivered to the Instruction Decode Queue (IDQ) from the MITE (legacy decode pipeline) path. Counting includes uops that may 'bypass' the IDQ. During these cycles uops are not being delivered from the Decode Stream Buffer (DSB).
IDQ.MS_CYCLES	
EventSel=79H, UMask=30H, CMask=1	Counts cycles during which uops are being delivered to Instruction Decode Queue (IDQ) while the Microcode Sequencer (MS) is busy. Counting includes uops that may 'bypass' the IDQ. Uops maybe initiated by Decode Stream Buffer (DSB) or MITE.
IDQ.MS_SWITCHES	
EventSel=79H, UMask=30H, EdgeDetect=1, CMask=1	Number of switches from DSB (Decode Stream Buffer) or MITE (legacy decode pipeline) to the Microcode Sequencer.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
IDQ.MS_UOPS	
EventSel=79H, UMask=30H	Counts the total number of uops delivered by the Microcode Sequencer (MS). Any instruction over 4 uops will be delivered by the MS. Some instructions such as transcendentals may additionally generate uops from the MS.
ICACHE_16B.IFDATA_STALL	
EventSel=80H, UMask=04H	Cycles where a code line fetch is stalled due to an L1 instruction cache miss. The legacy decode pipeline works at a 16 Byte granularity.
ICACHE_64B.IFTAG_HIT	
EventSel=83H, UMask=01H	Instruction fetch tag lookups that hit in the instruction cache (L1I). Counts at 64-byte cache-line granularity.
ICACHE_64B.IFTAG_MISS	
EventSel=83H, UMask=02H	Instruction fetch tag lookups that miss in the instruction cache (L1I). Counts at 64-byte cache-line granularity.
ICACHE_64B.IFTAG_STALL	
EventSel=83H, UMask=04H	Cycles where a code fetch is stalled due to L1 instruction cache tag miss.
ITLB_MISSES.MISS_CAUSES_A_WALK	
EventSel=85H, UMask=01H	Counts page walks of any page size (4K/2M/4M/1G) caused by a code fetch. This implies it missed in the ITLB and further levels of TLB, but the walk need not have completed.
ITLB_MISSES.WALK_COMPLETED_4K	
EventSel=85H, UMask=02H	Counts completed page walks (4K page size) caused by a code fetch. This implies it missed in the ITLB and further levels of TLB. The page walk can end with or without a fault.
ITLB_MISSES.WALK_COMPLETED_2M_4M	
EventSel=85H, UMask=04H	Counts code misses in all ITLB levels that caused a completed page walk (2M and 4M page sizes). The page walk can end with or without a fault.
ITLB_MISSES.WALK_COMPLETED_1G	
EventSel=85H, UMask=08H	Counts store misses in all DTLB levels that cause a completed page walk (1G page size). The page walk can end with or without a fault.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
ITLB_MISSES.WALK_COMPLETED	
EventSel=85H, UMask=0EH	Counts completed page walks (2M and 4M page sizes) caused by a code fetch. This implies it missed in the ITLB and further levels of TLB. The page walk can end with or without a fault.
ITLB_MISSES.WALK_PENDING	
EventSel=85H, UMask=10H	Counts 1 per cycle for each PMH (Page Miss Handler) that is busy with a page walk for an instruction fetch request. EPT page walk duration are excluded in Skylake michroarchitecture. .
ITLB_MISSES.WALK_ACTIVE	
EventSel=85H, UMask=10H, CMask=1	Cycles when at least one PMH is busy with a page walk for code (instruction fetch) request. EPT page walk duration are excluded in Skylake microarchitecture.
ITLB_MISSES.STLB_HIT	
EventSel=85H, UMask=20H	Instruction fetch requests that miss the ITLB and hit the STLB.
ILD_STALL.LCP	
EventSel=87H, UMask=01H	Counts cycles that the Instruction Length decoder (ILD) stalls occurred due to dynamically changing prefix length of the decoded instruction (by operand size prefix instruction 0x66, address size prefix instruction 0x67 or REX.W for Intel64). Count is proportional to the number of prefixes in a 16B-line. This may result in a three-cycle penalty for each LCP (Length changing prefix) in a 16-byte chunk.
IDQ_UOPS_NOT_DELIVERED.CORE	
EventSel=9CH, UMask=01H	Counts the number of uops not delivered to Resource Allocation Table (RAT) per thread adding “4 – x” when Resource Allocation Table (RAT) is not stalled and Instruction Decode Queue (IDQ) delivers x uops to Resource Allocation Table (RAT) (where x belongs to {0,1,2,3}). Counting does not cover cases when: a. IDQ-Resource Allocation Table (RAT) pipe serves the other thread. b. Resource Allocation Table (RAT) is stalled for the thread (including uop drops and clear BE conditions). c. Instruction Decode Queue (IDQ) delivers four uops.
IDQ_UOPS_NOT_DELIVERED.CYCLES_0_UOPS_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=4	Counts, on the per-thread basis, cycles when no uops are delivered to Resource Allocation Table (RAT). IDQ_Uops_Not_Delivered.core =4.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_1_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=3	Counts, on the per-thread basis, cycles when less than 1 uop is delivered to Resource Allocation Table (RAT). IDQ_Uops_Not_Delivered.core >= 3.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_2_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=2	Cycles with less than 2 uops delivered by the front-end.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_3_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=1	Cycles with less than 3 uops delivered by the front-end.
IDQ_UOPS_NOT_DELIVERED.CYCLES_FE_WAS_OK	
EventSel=9CH, UMask=01H, Invert=1, CMask=1	Counts cycles FE delivered 4 uops or Resource Allocation Table (RAT) was stalling FE.
UOPS_DISPATCHED_PORT.PORT_0	
EventSel=A1H, UMask=01H	Counts, on the per-thread basis, cycles during which at least one uop is dispatched from the Reservation Station (RS) to port 0.
UOPS_DISPATCHED_PORT.PORT_1	
EventSel=A1H, UMask=02H	Counts, on the per-thread basis, cycles during which at least one uop is dispatched from the Reservation Station (RS) to port 1.
UOPS_DISPATCHED_PORT.PORT_2	
EventSel=A1H, UMask=04H	Counts, on the per-thread basis, cycles during which at least one uop is dispatched from the Reservation Station (RS) to port 2.
UOPS_DISPATCHED_PORT.PORT_3	
EventSel=A1H, UMask=08H	Counts, on the per-thread basis, cycles during which at least one uop is dispatched from the Reservation Station (RS) to port 3.
UOPS_DISPATCHED_PORT.PORT_4	
EventSel=A1H, UMask=10H	Counts, on the per-thread basis, cycles during which at least one uop is dispatched from the Reservation Station (RS) to port 4.
UOPS_DISPATCHED_PORT.PORT_5	
EventSel=A1H, UMask=20H	Counts, on the per-thread basis, cycles during which at least one uop is dispatched from the Reservation Station (RS) to port 5.
UOPS_DISPATCHED_PORT.PORT_6	
EventSel=A1H, UMask=40H	Counts, on the per-thread basis, cycles during which at least one uop is dispatched from the Reservation Station (RS) to port 6.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
UOPS_DISPATCHED_PORT.PORT_7	
EventSel=A1H, UMask=80H	Counts, on the per-thread basis, cycles during which at least one uop is dispatched from the Reservation Station (RS) to port 7.
RESOURCE_STALLS.ANY	
EventSel=A2H, UMask=01H	Counts resource-related stall cycles. Reasons for stalls can be as follows:a. *any* u-arch structure got full (LB, SB, RS, ROB, BOB, LM, Physical Register Reclaim Table (PRRT), or Physical History Table (PHT) slots).b. *any* u-arch structure got empty (like INT/SIMD FreeLists).c. FPU control word (FPCW), MXCSR.and others. This counts cycles that the pipeline back-end blocked uop delivery from the front-end.
RESOURCE_STALLS.SB	
EventSel=A2H, UMask=08H	Counts allocation stall cycles caused by the store buffer (SB) being full. This counts cycles that the pipeline back-end blocked uop delivery from the front-end.
CYCLE_ACTIVITY.CYCLES_L2_MISS	
EventSel=A3H, UMask=01H, CMask=1	Cycles while L2 cache miss demand load is outstanding.
CYCLE_ACTIVITY.CYCLES_L3_MISS	
EventSel=A3H, UMask=02H, CMask=2	Cycles while L3 cache miss demand load is outstanding.
CYCLE_ACTIVITY.STALLS_TOTAL	
EventSel=A3H, UMask=04H, CMask=4	Total execution stalls.
CYCLE_ACTIVITY.STALLS_L2_MISS	
EventSel=A3H, UMask=05H, CMask=5	Execution stalls while L2 cache miss demand load is outstanding.
CYCLE_ACTIVITY.STALLS_L3_MISS	
EventSel=A3H, UMask=06H, CMask=6	Execution stalls while L3 cache miss demand load is outstanding.
CYCLE_ACTIVITY.CYCLES_L1D_MISS	
EventSel=A3H, UMask=08H, CMask=8	Cycles while L1 cache miss demand load is outstanding.
CYCLE_ACTIVITY.STALLS_L1D_MISS	
EventSel=A3H, UMask=0CH, CMask=12	Execution stalls while L1 cache miss demand load is outstanding.
CYCLE_ACTIVITY.CYCLES_MEM_ANY	
EventSel=A3H, UMask=10H, CMask=16	Cycles while memory subsystem has an outstanding load.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
CYCLE_ACTIVITY.STALLS_MEM_ANY	
EventSel=A3H, UMask=14H, CMask=20	Execution stalls while memory subsystem has an outstanding load.
EXE_ACTIVITY.EXE_BOUND_0_PORTS	
EventSel=A6H, UMask=01H	Counts cycles during which no uops were executed on all ports and Reservation Station (RS) was not empty.
EXE_ACTIVITY.1_PORTS_UTIL	
EventSel=A6H, UMask=02H	Counts cycles during which a total of 1 uop was executed on all ports and Reservation Station (RS) was not empty.
EXE_ACTIVITY.2_PORTS_UTIL	
EventSel=A6H, UMask=04H	Counts cycles during which a total of 2 uops were executed on all ports and Reservation Station (RS) was not empty.
EXE_ACTIVITY.3_PORTS_UTIL	
EventSel=A6H, UMask=08H	Cycles total of 3 uops are executed on all ports and Reservation Station (RS) was not empty.
EXE_ACTIVITY.4_PORTS_UTIL	
EventSel=A6H, UMask=10H	Cycles total of 4 uops are executed on all ports and Reservation Station (RS) was not empty.
EXE_ACTIVITY.BOUND_ON_STORES	
EventSel=A6H, UMask=40H	Cycles where the Store Buffer was full and no outstanding load.
LSD.UOPS	
EventSel=A8H, UMask=01H	Number of uops delivered to the back-end by the LSD(Loop Stream Detector).
LSD.CYCLES_ACTIVE	
EventSel=A8H, UMask=01H, CMask=1	Counts the cycles when at least one uop is delivered by the LSD (Loop-stream detector).
LSD.CYCLES_4_UOPS	
EventSel=A8H, UMask=01H, CMask=4	Counts the cycles when 4 uops are delivered by the LSD (Loop-stream detector).

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
DSB2MITE_SWITCHES.PENALTY_CYCLES	
EventSel=ABH, UMask=02H	Counts Decode Stream Buffer (DSB)-to-MITE switch true penalty cycles. These cycles do not include uops routed through because of the switch itself, for example, when Instruction Decode Queue (IDQ) pre-allocation is unavailable, or Instruction Decode Queue (IDQ) is full. SBD-to-MITE switch true penalty cycles happen after the merge mux (MM) receives Decode Stream Buffer (DSB) Sync-indication until receiving the first MITE uop. MM is placed before Instruction Decode Queue (IDQ) to merge uops being fed from the MITE and Decode Stream Buffer (DSB) paths. Decode Stream Buffer (DSB) inserts the Sync-indication whenever a Decode Stream Buffer (DSB)-to-MITE switch occurs.Penalty: A Decode Stream Buffer (DSB) hit followed by a Decode Stream Buffer (DSB) miss can cost up to six cycles in which no uops are delivered to the IDQ. Most often, such switches from the Decode Stream Buffer (DSB) to the legacy pipeline cost 0–2 cycles.
ITLB.ITLB_FLUSH	
EventSel=AEH, UMask=01H	Counts the number of flushes of the big or small ITLB pages. Counting include both TLB Flush (covering all sets) and TLB Set Clear (set-specific).
OFFCORE_REQUESTS.DEMAND_DATA_RD	
EventSel=B0H, UMask=01H	Counts the Demand Data Read requests sent to uncore. Use it in conjunction with OFFCORE_REQUESTS_OUTSTANDING to determine average latency in the uncore.
OFFCORE_REQUESTS.DEMAND_CODE_RD	
EventSel=B0H, UMask=02H	Counts both cacheable and non-cacheable code read requests.
OFFCORE_REQUESTS.DEMAND_RFO	
EventSel=B0H, UMask=04H	Counts the demand RFO (read for ownership) requests including regular RFOs, locks, ItoM.
OFFCORE_REQUESTS.ALL_DATA_RD	
EventSel=B0H, UMask=08H	Counts the demand and prefetch data reads. All Core Data Reads include cacheable 'Demands' and L2 prefetchers (not L3 prefetchers). Counting also covers reads due to page walks resulted from any request type.
OFFCORE_REQUESTS.L3_MISS_DEMAND_DATA_RD	
EventSel=B0H, UMask=10H	Demand Data Read requests who miss L3 cache.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
OFFCORE_REQUESTS.ALL_REQUESTS	
EventSel=B0H, UMask=80H	Counts memory transactions reached the super queue including requests initiated by the core, all L3 prefetches, page walks, etc..
UOPS_EXECUTED.THREAD	
EventSel=B1H, UMask=01H	Number of uops to be executed per-thread each cycle.
UOPS_EXECUTED.STALL_CYCLES	
EventSel=B1H, UMask=01H, Invert=1, CMask=1	Counts cycles during which no uops were dispatched from the Reservation Station (RS) per thread.
UOPS_EXECUTED.CYCLES_GE_1_UOP_EXEC	
EventSel=B1H, UMask=01H, CMask=1	Cycles where at least 1 uop was executed per-thread.
UOPS_EXECUTED.CYCLES_GE_2_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=2	Cycles where at least 2 uops were executed per-thread.
UOPS_EXECUTED.CYCLES_GE_3_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=3	Cycles where at least 3 uops were executed per-thread.
UOPS_EXECUTED.CYCLES_GE_4_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=4	Cycles where at least 4 uops were executed per-thread.
UOPS_EXECUTED.CORE	
EventSel=B1H, UMask=02H	Number of uops executed from any thread.
UOPS_EXECUTED.CORE_CYCLES_GE_1	
EventSel=B1H, UMask=02H, CMask=1	Cycles at least 1 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_2	
EventSel=B1H, UMask=02H, CMask=2	Cycles at least 2 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_3	
EventSel=B1H, UMask=02H, CMask=3	Cycles at least 3 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_4	
EventSel=B1H, UMask=02H, CMask=4	Cycles at least 4 micro-op is executed from any thread on physical core.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
UOPS_EXECUTED.CORE_CYCLES_NONE	
EventSel=B1H, UMask=02H, Invert=1, CMask=1	Cycles with no micro-ops executed from any thread on physical core.
UOPS_EXECUTED.X87	
EventSel=B1H, UMask=10H	Counts the number of x87 uops executed.
OFFCORE_REQUESTS_BUFFER.SQ_FULL	
EventSel=B2H, UMask=01H	Counts the number of cases when the offcore requests buffer cannot take more entries for the core. This can happen when the superqueue does not contain eligible entries, or when L1D writeback pending FIFO requests is full.Note: Writeback pending FIFO has six entries.
TLB_FLUSH.DTLB_THREAD	
EventSel=BDH, UMask=01H	Counts the number of DTLB flush attempts of the thread-specific entries.
TLB_FLUSH.STLB_ANY	
EventSel=BDH, UMask=20H	Counts the number of any STLB flush attempts (such as entire, VPID, PCID, InvPage, CR3 write, etc.).
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural	Counts the number of instructions (EOMs) retired. Counting covers macro-fused instructions individually (that is, increments by two).
INST_RETIRED.PREC_DIST	
EventSel=C0H, UMask=01H, Precise	A version of INST_RETIRED that allows for a more unbiased distribution of samples across instructions retired. It utilizes the Precise Distribution of Instructions Retired (PDIR) feature to mitigate some bias in how retired instructions get sampled.
OTHER_ASSISTS.ANY	
EventSel=C1H, UMask=3FH	Number of times a microcode assist is invoked by HW other than FP-assist. Examples include AD (page Access Dirty) and AVX* related assists.
UOPS_RETIRED.RETIRE_SLOTS	
EventSel=C2H, UMask=02H	Counts the retirement slots used.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
UOPS_RETIRED.STALL_CYCLES	
EventSel=C2H, UMask=02H, Invert=1, CMask=1	This event counts cycles without actually retired uops.
UOPS_RETIRED.TOTAL_CYCLES	
EventSel=C2H, UMask=02H, Invert=1, CMask=10	Number of cycles using always true condition (uops_ret < 16) applied to non PEBS uops retired event.
MACHINE_CLEARS.COUNT	
EventSel=C3H, UMask=01H, EdgeDetect=1, CMask=1	Number of machine clears (nukes) of any type.
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	Counts the number of memory ordering Machine Clears detected. Memory Ordering Machine Clears can result from one of the following:a. memory disambiguation,b. external snoop, orc. cross SMT-HW-thread snoop (stores) hitting load buffer.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=04H	Counts self-modifying code (SMC) detected, which causes a machine clear.
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	Counts all (macro) branch instructions retired.
BR_INST_RETIRED.CONDITIONAL	
EventSel=C4H, UMask=01H, Precise	This event counts conditional branch instructions retired.
BR_INST_RETIRED.NEAR_CALL	
EventSel=C4H, UMask=02H, Precise	This event counts both direct and indirect near call instructions retired.
BR_INST_RETIRED.NEAR_RETURN	
EventSel=C4H, UMask=08H, Precise	This event counts return instructions retired.
BR_INST_RETIRED.NOT_TAKEN	
EventSel=C4H, UMask=10H	This event counts not taken branch instructions retired.
BR_INST_RETIRED.NEAR_TAKEN	
EventSel=C4H, UMask=20H, Precise	This event counts taken branch instructions retired.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=40H, Precise	This event counts far branch instructions retired.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	Counts all the retired branch instructions that were mispredicted by the processor. A branch misprediction occurs when the processor incorrectly predicts the destination of the branch. When the misprediction is discovered at execution, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path.
BR_MISP_RETIRED.CONDITIONAL	
EventSel=C5H, UMask=01H, Precise	This event counts mispredicted conditional branch instructions retired.
BR_MISP_RETIRED.NEAR_CALL	
EventSel=C5H, UMask=02H, Precise	Counts both taken and not taken retired mispredicted direct and indirect near calls, including both register and memory indirect.
BR_MISP_RETIRED.NEAR_TAKEN	
EventSel=C5H, UMask=20H, Precise	Number of near branch instructions retired that were mispredicted and taken.
FRONTEND_RETIRED.DSB_MISS	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x11 , Precise	Counts retired Instructions that experienced DSB (Decode stream buffer i.e. the decoded instruction-cache) miss. .
FRONTEND_RETIRED.L1I_MISS	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x12 , Precise	Retired Instructions who experienced Instruction L1 Cache true miss.
FRONTEND_RETIRED.L2_MISS	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x13 , Precise	Retired Instructions who experienced Instruction L2 Cache true miss.
FRONTEND_RETIRED.ITLB_MISS	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x14 , Precise	Counts retired Instructions that experienced iTLB (Instruction TLB) true miss.
FRONTEND_RETIRED.STLB_MISS	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x15 , Precise	Counts retired Instructions that experienced STLB (2nd level TLB) true miss. .

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
FRONTEND_RETIRED.LATENCY_GE_2	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x400206 , Precise	Retired instructions that are fetched after an interval where the front-end delivered no uops for a period of 2 cycles which was not interrupted by a back-end stall.
FRONTEND_RETIRED.LATENCY_GE_2_BUBBLES_GE_2	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x200206 , Precise	Retired instructions that are fetched after an interval where the front-end had at least 2 bubble-slots for a period of 2 cycles which was not interrupted by a back-end stall.
FRONTEND_RETIRED.LATENCY_GE_4	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x400406 , Precise	Retired instructions that are fetched after an interval where the front-end delivered no uops for a period of 4 cycles which was not interrupted by a back-end stall.
FRONTEND_RETIRED.LATENCY_GE_8	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x400806 , Precise	Counts retired instructions that are delivered to the back-end after a front-end stall of at least 8 cycles. During this period the front-end delivered no uops.
FRONTEND_RETIRED.LATENCY_GE_16	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x401006 , Precise	Counts retired instructions that are delivered to the back-end after a front-end stall of at least 16 cycles. During this period the front-end delivered no uops.
FRONTEND_RETIRED.LATENCY_GE_32	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x402006 , Precise	Counts retired instructions that are delivered to the back-end after a front-end stall of at least 32 cycles. During this period the front-end delivered no uops.
FRONTEND_RETIRED.LATENCY_GE_64	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x404006 , Precise	Retired instructions that are fetched after an interval where the front-end delivered no uops for a period of 64 cycles which was not interrupted by a back-end stall.
FRONTEND_RETIRED.LATENCY_GE_128	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x408006 , Precise	Retired instructions that are fetched after an interval where the front-end delivered no uops for a period of 128 cycles which was not interrupted by a back-end stall.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
FRONTEND_RETIRED.LATENCY_GE_256	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x410006 , Precise	Retired instructions that are fetched after an interval where the front-end delivered no uops for a period of 256 cycles which was not interrupted by a back-end stall.
FRONTEND_RETIRED.LATENCY_GE_512	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x420006 , Precise	Retired instructions that are fetched after an interval where the front-end delivered no uops for a period of 512 cycles which was not interrupted by a back-end stall.
FRONTEND_RETIRED.LATENCY_GE_2_BUBBLES_GE_1	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x100206 , Precise	Counts retired instructions that are delivered to the back-end after the front-end had at least 1 bubble-slot for a period of 2 cycles. A bubble-slot is an empty issue-pipeline slot while there was no RAT stall.
FRONTEND_RETIRED.LATENCY_GE_2_BUBBLES_GE_3	
EventSel=C6H, UMask=01H, MSR_PEBS_FRONTEND=0x300206 , Precise	Retired instructions that are fetched after an interval where the front-end had at least 3 bubble-slots for a period of 2 cycles which was not interrupted by a back-end stall.
FP_ARITH_INST_RETIRED.SCALAR_DOUBLE	
EventSel=C7H, UMask=01H	Number of SSE/AVX computational scalar double precision floating-point instructions retired. Each count represents 1 computation. Applies to SSE* and AVX* scalar double precision floating-point instructions: ADD SUB MUL DIV MIN MAX SQRT FM(N)ADD/SUB. FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.SCALAR_SINGLE	
EventSel=C7H, UMask=02H	Number of SSE/AVX computational scalar single precision floating-point instructions retired. Each count represents 1 computation. Applies to SSE* and AVX* scalar single precision floating-point instructions: ADD SUB MUL DIV MIN MAX RCP RSQRT SQRT FM(N)ADD/SUB. FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.128B_PACKED_DOUBLE	
EventSel=C7H, UMask=04H	Number of SSE/AVX computational 128-bit packed double precision floating-point instructions retired. Each count represents 2 computations. Applies to SSE* and AVX* packed double precision floating-point instructions: ADD SUB MUL DIV MIN MAX SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
FP_ARITH_INST_RETIRED.128B_PACKED_SINGLE	
EventSel=C7H, UMask=08H	Number of SSE/AVX computational 128-bit packed single precision floating-point instructions retired. Each count represents 4 computations. Applies to SSE* and AVX* packed single precision floating-point instructions: ADD SUB MUL DIV MIN MAX RCP RSQRT SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.256B_PACKED_DOUBLE	
EventSel=C7H, UMask=10H	Number of SSE/AVX computational 256-bit packed double precision floating-point instructions retired. Each count represents 4 computations. Applies to SSE* and AVX* packed double precision floating-point instructions: ADD SUB MUL DIV MIN MAX SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.256B_PACKED_SINGLE	
EventSel=C7H, UMask=20H	Number of SSE/AVX computational 256-bit packed single precision floating-point instructions retired. Each count represents 8 computations. Applies to SSE* and AVX* packed single precision floating-point instructions: ADD SUB MUL DIV MIN MAX RCP RSQRT SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
HLE_RETIRED.START	
EventSel=C8H, UMask=01H	Number of times we entered an HLE region. Does not count nested transactions.
HLE_RETIRED.COMMIT	
EventSel=C8H, UMask=02H	Number of times HLE commit succeeded.
HLE_RETIRED.ABORTED	
EventSel=C8H, UMask=04H, Precise	Number of times HLE abort was triggered.
HLE_RETIRED.ABORTED_MEM	
EventSel=C8H, UMask=08H	Number of times an HLE execution aborted due to various memory events (e.g., read/write capacity and conflicts).
HLE_RETIRED.ABORTED_TIMER	
EventSel=C8H, UMask=10H	Number of times an HLE execution aborted due to hardware timer expiration.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
HLE_RETIRED.ABORTED_UNFRIENDLY	
EventSel=C8H, UMask=20H	Number of times an HLE execution aborted due to HLE-unfriendly instructions and certain unfriendly events (such as AD assists etc.).
HLE_RETIRED.ABORTED_MEMTYPE	
EventSel=C8H, UMask=40H	Number of times an HLE execution aborted due to incompatible memory type.
HLE_RETIRED.ABORTED_EVENTS	
EventSel=C8H, UMask=80H	Number of times an HLE execution aborted due to unfriendly events (such as interrupts).
RTM_RETIRED.START	
EventSel=C9H, UMask=01H	Number of times we entered an RTM region. Does not count nested transactions.
RTM_RETIRED.COMMIT	
EventSel=C9H, UMask=02H	Number of times RTM commit succeeded.
RTM_RETIRED.ABORTED	
EventSel=C9H, UMask=04H, Precise	Number of times RTM abort was triggered.
RTM_RETIRED.ABORTED_MEM	
EventSel=C9H, UMask=08H	Number of times an RTM execution aborted due to various memory events (e.g. read/write capacity and conflicts).
RTM_RETIRED.ABORTED_TIMER	
EventSel=C9H, UMask=10H	Number of times an RTM execution aborted due to uncommon conditions.
RTM_RETIRED.ABORTED_UNFRIENDLY	
EventSel=C9H, UMask=20H	Number of times an RTM execution aborted due to HLE-unfriendly instructions.
RTM_RETIRED.ABORTED_MEMTYPE	
EventSel=C9H, UMask=40H	Number of times an RTM execution aborted due to incompatible memory type.
RTM_RETIRED.ABORTED_EVENTS	
EventSel=C9H, UMask=80H	Number of times an RTM execution aborted due to none of the previous 4 categories (e.g. interrupt).

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
FP_ASSIST.ANY	
EventSel=CAH, UMask=1EH, CMask=1	Counts cycles with any input and output SSE or x87 FP assist. If an input and output assist are detected on the same cycle the event increments by 1.
HW_INTERRUPTS.RECEIVED	
EventSel=CBH, UMask=01H	Counts the number of hardware interruptions received by the processor.
ROB_MISC_EVENTS.LBR_INSERTS	
EventSel=CCH, UMask=20H	Increments when an entry is added to the Last Branch Record (LBR) array (or removed from the array in case of RETURNs in call stack mode). The event requires LBR enable via IA32_DEBUGCTL MSR and branch type selection via MSR_LBR_SELECT.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_4	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x4 , Precise	Counts loads when the latency from first dispatch to completion is greater than 4 cycles. Reported latency may be longer than just the memory latency.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_8	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x8 , Precise	Counts loads when the latency from first dispatch to completion is greater than 8 cycles. Reported latency may be longer than just the memory latency.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_16	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x10 , Precise	Counts loads when the latency from first dispatch to completion is greater than 16 cycles. Reported latency may be longer than just the memory latency.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_32	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x20 , Precise	Counts loads when the latency from first dispatch to completion is greater than 32 cycles. Reported latency may be longer than just the memory latency.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_64	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x40 , Precise	Counts loads when the latency from first dispatch to completion is greater than 64 cycles. Reported latency may be longer than just the memory latency.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_128	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x80 , Precise	Counts loads when the latency from first dispatch to completion is greater than 128 cycles. Reported latency may be longer than just the memory latency.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_256	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x100 , Precise	Counts loads when the latency from first dispatch to completion is greater than 256 cycles. Reported latency may be longer than just the memory latency.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_512	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x200 , Precise	Counts loads when the latency from first dispatch to completion is greater than 512 cycles. Reported latency may be longer than just the memory latency.
MEM_INST_RETIRED.STLB_MISS_LOADS	
EventSel=D0H, UMask=11H, Precise	Retired load instructions that miss the STLB.
MEM_INST_RETIRED.STLB_MISS_STORES	
EventSel=D0H, UMask=12H, Precise	Retired store instructions that miss the STLB.
MEM_INST_RETIRED.LOCK_LOADS	
EventSel=D0H, UMask=21H, Precise	Retired load instructions with locked access.
MEM_INST_RETIRED.SPLIT_LOADS	
EventSel=D0H, UMask=41H, Precise	Counts retired load instructions that split across a cacheline boundary.
MEM_INST_RETIRED.SPLIT_STORES	
EventSel=D0H, UMask=42H, Precise	Counts retired store instructions that split across a cacheline boundary.
MEM_INST_RETIRED.ALL_LOADS	
EventSel=D0H, UMask=81H, Precise	All retired load instructions.
MEM_INST_RETIRED.ALL_STORES	
EventSel=D0H, UMask=82H, Precise	All retired store instructions.
MEM_LOAD_RETIRED.L1_HIT	
EventSel=D1H, UMask=01H, Precise	Counts retired load instructions with at least one uop that hit in the L1 data cache. This event includes all SW prefetches and lock instructions regardless of the data source.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
MEM_LOAD_RETIRED.L2_HIT	
EventSel=D1H, UMask=02H, Precise	Retired load instructions with L2 cache hits as data sources.
MEM_LOAD_RETIRED.L3_HIT	
EventSel=D1H, UMask=04H, Precise	Counts retired load instructions with at least one uop that hit in the L3 cache. .
MEM_LOAD_RETIRED.L1_MISS	
EventSel=D1H, UMask=08H, Precise	Counts retired load instructions with at least one uop that missed in the L1 cache.
MEM_LOAD_RETIRED.L2_MISS	
EventSel=D1H, UMask=10H, Precise	Retired load instructions missed L2 cache as data sources.
MEM_LOAD_RETIRED.L3_MISS	
EventSel=D1H, UMask=20H, Precise	Counts retired load instructions with at least one uop that missed in the L3 cache. .
MEM_LOAD_RETIRED.FB_HIT	
EventSel=D1H, UMask=40H, Precise	Counts retired load instructions with at least one uop was load missed in L1 but hit FB (Fill Buffers) due to preceding miss to the same cache line with data not ready. .
MEM_LOAD_L3_HIT_RETIRED.XSNP_MISS	
EventSel=D2H, UMask=01H, Precise	Retired load instructions which data sources were L3 hit and cross-core snoop missed in on-pkg core cache.
MEM_LOAD_L3_HIT_RETIRED.XSNP_HIT	
EventSel=D2H, UMask=02H, Precise	Retired load instructions which data sources were L3 and cross-core snoop hits in on-pkg core cache.
MEM_LOAD_L3_HIT_RETIRED.XSNP_HITM	
EventSel=D2H, UMask=04H, Precise	Retired load instructions which data sources were HitM responses from shared L3.
MEM_LOAD_L3_HIT_RETIRED.XSNP_NONE	
EventSel=D2H, UMask=08H, Precise	Retired load instructions which data sources were hits in L3 without snoops required.
MEM_LOAD_MISC_RETIRED.UC	
EventSel=D4H, UMask=04H, Precise	Retired instructions with at least 1 uncacheable load or lock.

Table 2: Performance Events of the Processor Core Supported by Skylake Microarchitecture (06_4EH, 06_5EH) and Kaby Lake Microarchitecture (06_8EH, 06_9EH)

Event Name	
Configuration	Description
BACLEARS.ANY	
EventSel=E6H, UMask=01H	Counts the number of times the front-end is resteered when it finds a branch instruction in a fetch line. This occurs for the first time a branch instruction is fetched or when the branch is not tracked by the BPU (Branch Prediction Unit) anymore.
L2_TRANS.L2_WB	
EventSel=F0H, UMask=40H	Counts L2 writebacks that access L2 cache.
L2_LINES_IN.ALL	
EventSel=F1H, UMask=1FH	Counts the number of L2 cache lines filling the L2. Counting does not cover rejects.
L2_LINES_OUT.SILENT	
EventSel=F2H, UMask=01H	Counts the number of lines that are silently dropped by L2 cache when triggered by an L2 cache fill. These lines are typically in Shared or Exclusive state. A non-threaded event.
L2_LINES_OUT.NON_SILENT	
EventSel=F2H, UMask=02H	Counts the number of lines that are evicted by L2 cache when triggered by an L2 cache fill. Those lines are in Modified state. Modified lines are written back to L3.
*L2_LINES_OUT.USELESS_PREF DEPRECATED	
EventSel=F2H, UMask=04H	Counts the number of lines that have been hardware prefetched but not used and now evicted by L2 cache. **Note:This event is deprecated.Use other event L2_LINES_OUT.USELESS_HWPF*
L2_LINES_OUT.USELESS_HWPF	
EventSel=F2H, UMask=04H	Counts the number of lines that have been hardware prefetched but not used and now evicted by L2 cache.*Counts the number of lines that have been hardware prefetched but not used and now evicted by L2 cache*
SQ_MISC.SPLIT_LOCK	
EventSel=F4H, UMask=10H	Counts the number of cache line split locks sent to the uncore.

Performance Monitoring Events based on Broadwell Microarchitecture - Intel® Core™ M and 5th Generation Intel® Core™ Processors

The Intel® Core™ M processors, the 5th generation Intel® Core™ processors and the Intel Xeon processor E3 1200 v4 product family are based on the Broadwell Microarchitecture. performance-monitoring events in the processor core are listed in the table below.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed	This event counts the number of instructions retired from execution. For instructions that consist of multiple micro-ops, this event counts the retirement of the last micro-op of the instruction. Counting continues during hardware interrupts, traps, and inside interrupt handlers. Notes: INST_RETIRED.ANY is counted by a designated fixed counter, leaving the four (eight when Hyperthreading is disabled) programmable counters available for other events. INST_RETIRED.ANY_P is counted by a programmable counter and it is an architectural performance event. Counting: Faulting executions of GETSEC/VM entry/VM Exit/MWait will not count as retired instructions.
CPU_CLK_UNHALTED.THREAD	
Architectural, Fixed	This event counts the number of core cycles while the thread is not in a halt state. The thread enters the halt state when it is running the HLT instruction. This event is a component in many key event ratios. The core frequency may change from time to time due to transitions associated with Enhanced Intel SpeedStep Technology or TM2. For this reason this event may have a changing ratio with regards to time. When the core frequency is constant, this event can approximate elapsed time while the core was not in the halt state. It is counted on a dedicated fixed counter, leaving the four (eight when Hyperthreading is disabled) programmable counters available for other events.
CPU_CLK_UNHALTED.THREAD_ANY	
AnyThread=1, Architectural, Fixed	Core cycles when at least one thread on the physical core is not in halt state.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
CPU_CLK_UNHALTED.REF_TSC	
Architectural, Fixed	This event counts the number of reference cycles when the core is not in a halt state. The core enters the halt state when it is running the HLT instruction or the MWAIT instruction. This event is not affected by core frequency changes (for example, P states, TM2 transitions) but has the same incrementing frequency as the time stamp counter. This event can approximate elapsed time while the core was not in a halt state. This event has a constant ratio with the CPU_CLK_UNHALTED.REF_XCLK event. It is counted on a dedicated fixed counter, leaving the four (eight when Hyperthreading is disabled) programmable counters available for other events. Note: On all current platforms this event stops counting during 'throttling (TM)' states duty off periods the processor is 'halted'. This event is clocked by base clock (100 Mhz) on Sandy Bridge. The counter update is done at a lower clock rate then the core clock the overflow status bit for this counter may appear 'sticky'. After the counter has overflowed and software clears the overflow status bit and resets the counter to less than MAX. The reset value to the counter is not clocked immediately so the overflow status bit will flip 'high (1)' and generate another PMI (if enabled) after which the reset value gets clocked into the counter. Therefore, software will get the interrupt, read the overflow status bit '1 for bit 34 while the counter value is less than MAX. Software should ignore this case.
LD_BLOCKS.STORE_FORWARD	
EventSel=03H, UMask=02H	This event counts how many times the load operation got the true Block-on-Store blocking code preventing store forwarding. This includes cases when: - preceding store conflicts with the load (incomplete overlap); - store forwarding is impossible due to u-arch limitations; - preceding lock RMW operations are not forwarded; - store has the no-forward bit set (uncacheable/page-split/masked stores); - all-blocking stores are used (mostly, fences and port I/O); and others. The most common case is a load blocked due to its address range overlapping with a preceding smaller uncompleted store. Note: This event does not take into account cases of out-of-SW-control (for example, SbTailHit), unknown physical STA, and cases of blocking loads on store due to being non-WB memory type or a lock. These cases are covered by other events. See the table of not supported store forwards in the Optimization Guide.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
LD_BLOCKS.NO_SR	
EventSel=03H, UMask=08H	This event counts the number of times that split load operations are temporarily blocked because all resources for handling the split accesses are in use.
MISALIGN_MEM_REF.LOADS	
EventSel=05H, UMask=01H	This event counts speculative cache-line split load uops dispatched to the L1 cache.
MISALIGN_MEM_REF.STORES	
EventSel=05H, UMask=02H	This event counts speculative cache line split store-address (STA) uops dispatched to the L1 cache.
LD_BLOCKS_PARTIAL.ADDRESS_ALIAS	
EventSel=07H, UMask=01H	This event counts false dependencies in MOB when the partial comparison upon loose net check and dependency was resolved by the Enhanced Loose net mechanism. This may not result in high performance penalties. Loose net checks can fail when loads and stores are 4k aliased.
DTLB_LOAD_MISSES.MISS_CAUSES_A_WALK	
EventSel=08H, UMask=01H	This event counts load misses in all DTLB levels that cause page walks of any page size (4K/2M/4M/1G).
DTLB_LOAD_MISSES.WALK_COMPLETED_4K	
EventSel=08H, UMask=02H	This event counts load misses in all DTLB levels that cause a completed page walk (4K page size). The page walk can end with or without a fault.
DTLB_LOAD_MISSES.WALK_COMPLETED_2M_4M	
EventSel=08H, UMask=04H	This event counts load misses in all DTLB levels that cause a completed page walk (2M and 4M page sizes). The page walk can end with or without a fault.
DTLB_LOAD_MISSES.WALK_COMPLETED_1G	
EventSel=08H, UMask=08H	This event counts load misses in all DTLB levels that cause a completed page walk (1G page size). The page walk can end with or without a fault.
DTLB_LOAD_MISSES.WALK_COMPLETED	
EventSel=08H, UMask=0EH	Demand load Miss in all translation lookaside buffer (TLB) levels causes a page walk that completes of any page size.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
DTLB_LOAD_MISSES.WALK_DURATION	
EventSel=08H, UMask=10H	This event counts the number of cycles while PMH is busy with the page walk.
DTLB_LOAD_MISSES.STLB_HIT_4K	
EventSel=08H, UMask=20H	Load misses that miss the DTLB and hit the STLB (4K).
DTLB_LOAD_MISSES.STLB_HIT_2M	
EventSel=08H, UMask=40H	Load misses that miss the DTLB and hit the STLB (2M).
DTLB_LOAD_MISSES.STLB_HIT	
EventSel=08H, UMask=60H	Load operations that miss the first DTLB level but hit the second and do not cause page walks.
INT_MISC.RECOVERY_CYCLES	
EventSel=0DH, UMask=03H, CMask=1	Cycles checkpoints in Resource Allocation Table (RAT) are recovering from JEClear or machine clear.
INT_MISC.RECOVERY_CYCLES_ANY	
EventSel=0DH, UMask=03H, AnyThread=1, CMask=1	Core cycles the allocator was stalled due to recovery from earlier clear event for any thread running on the physical core (e.g. misprediction or memory nuke).
INT_MISC.RAT_STALL_CYCLES	
EventSel=0DH, UMask=08H	This event counts the number of cycles during which Resource Allocation Table (RAT) external stall is sent to Instruction Decode Queue (IDQ) for the current thread. This also includes the cycles during which the Allocator is serving another thread.
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=01H	This event counts the number of Uops issued by the Resource Allocation Table (RAT) to the reservation station (RS).
UOPS_ISSUED.STALL_CYCLES	
EventSel=0EH, UMask=01H, Invert=1, CMask=1	This event counts cycles during which the Resource Allocation Table (RAT) does not issue any Uops to the reservation station (RS) for the current thread.
UOPS_ISSUED.FLAGS_MERGE	
EventSel=0EH, UMask=10H	Number of flags-merge uops being allocated. Such uops considered perf sensitive added by GSR u-arch.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
UOPS_ISSUED.SLOW_LEA	
EventSel=0EH, UMask=20H	Number of slow LEA uops being allocated. A uop is generally considered SlowLea if it has 3 sources (e.g. 2 sources + immediate) regardless if as a result of LEA instruction or not.
UOPS_ISSUED.SINGLE_MUL	
EventSel=0EH, UMask=40H	Number of Multiply packed/scalar single precision uops allocated.
ARITH.FPU_DIV_ACTIVE	
EventSel=14H, UMask=01H	This event counts the number of the divide operations executed. Uses edge-detect and a cmask value of 1 on ARITH.FPU_DIV_ACTIVE to get the number of the divide operations executed.
L2_RQSTS.DEMAND_DATA_RD_MISS	
EventSel=24H, UMask=21H	This event counts the number of demand Data Read requests that miss L2 cache. Only not rejected loads are counted.
L2_RQSTS.RFO_MISS	
EventSel=24H, UMask=22H	RFO requests that miss L2 cache.
L2_RQSTS.CODE_RD_MISS	
EventSel=24H, UMask=24H	L2 cache misses when fetching instructions.
L2_RQSTS.ALL_DEMAND_MISS	
EventSel=24H, UMask=27H	Demand requests that miss L2 cache.
L2_RQSTS.L2_PF_MISS	
EventSel=24H, UMask=30H	This event counts the number of requests from the L2 hardware prefetchers that miss L2 cache.
L2_RQSTS.MISS	
EventSel=24H, UMask=3FH	All requests that miss L2 cache.
L2_RQSTS.DEMAND_DATA_RD_HIT	
EventSel=24H, UMask=41H	This event counts the number of demand Data Read requests that hit L2 cache. Only not rejected loads are counted.
L2_RQSTS.RFO_HIT	
EventSel=24H, UMask=42H	RFO requests that hit L2 cache.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
L2_RQSTS.CODE_RD_HIT	
EventSel=24H, UMask=44H	L2 cache hits when fetching instructions, code reads.
L2_RQSTS.L2_PF_HIT	
EventSel=24H, UMask=50H	This event counts the number of requests from the L2 hardware prefetchers that hit L2 cache. L3 prefetch new types.
L2_RQSTS.ALL_DEMAND_DATA_RD	
EventSel=24H, UMask=E1H	This event counts the number of demand Data Read requests (including requests from L1D hardware prefetchers). These loads may hit or miss L2 cache. Only non rejected loads are counted.
L2_RQSTS.ALL_RFO	
EventSel=24H, UMask=E2H	This event counts the total number of RFO (read for ownership) requests to L2 cache. L2 RFO requests include both L1D demand RFO misses as well as L1D RFO prefetches.
L2_RQSTS.ALL_CODE_RD	
EventSel=24H, UMask=E4H	This event counts the total number of L2 code requests.
L2_RQSTS.ALL_DEMAND_REFERENCES	
EventSel=24H, UMask=E7H	Demand requests to L2 cache.
L2_RQSTS.ALL_PF	
EventSel=24H, UMask=F8H	This event counts the total number of requests from the L2 hardware prefetchers.
L2_RQSTS.REFERENCES	
EventSel=24H, UMask=FFH	All L2 requests.
L2_DEMAND_RQSTS.WB_HIT	
EventSel=27H, UMask=50H	This event counts the number of WB requests that hit L2 cache.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	This event counts core-originated cacheable demand requests that miss the last level cache (LLC). Demand requests include loads, RFOs, and hardware prefetches from L1D, and instruction fetches from IFU.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	This event counts core-originated cacheable demand requests that refer to the last level cache (LLC). Demand requests include loads, RFOs, and hardware prefetches from L1D, and instruction fetches from IFU.
CPU_CLK_UNHALTED.THREAD_P	
EventSel=3CH, UMask=00H, Architectural	This is an architectural event that counts the number of thread cycles while the thread is not in a halt state. The thread enters the halt state when it is running the HLT instruction. The core frequency may change from time to time due to power or thermal throttling. For this reason, this event may have a changing ratio with regards to wall clock time.
CPU_CLK_UNHALTED.THREAD_P_ANY	
EventSel=3CH, UMask=00H, AnyThread=1, Architectural	Core cycles when at least one thread on the physical core is not in halt state.
CPU_CLK_THREAD_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	This is a fixed-frequency event programmed to general counters. It counts when the core is unhalted at 100 Mhz.
CPU_CLK_THREAD_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Reference cycles when the at least one thread on the physical core is unhalted (counts at 100 MHz rate).
CPU_CLK_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	Reference cycles when the thread is unhalted (counts at 100 MHz rate).
CPU_CLK_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Reference cycles when the at least one thread on the physical core is unhalted (counts at 100 MHz rate).
CPU_CLK_THREAD_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Count XClk pulses when this thread is unhalted and the other thread is halted.
CPU_CLK_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Count XClk pulses when this thread is unhalted and the other thread is halted.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
L1D_PEND_MISS.PENDING	
EventSel=48H, UMask=01H	This event counts duration of L1D miss outstanding, that is each cycle number of Fill Buffers (FB) outstanding required by Demand Reads. FB either is held by demand loads, or it is held by non-demand loads and gets hit at least once by demand. The valid outstanding interval is defined until the FB deallocation by one of the following ways: from FB allocation, if FB is allocated by demand; from the demand Hit FB, if it is allocated by hardware or software prefetch. Note: In the L1D, a Demand Read contains cacheable or noncacheable demand loads, including ones causing cache-line splits and reads due to page walks resulted from any request type.
L1D_PEND_MISS.PENDING_CYCLES	
EventSel=48H, UMask=01H, CMask=1	This event counts duration of L1D miss outstanding in cycles.
L1D_PEND_MISS.PENDING_CYCLES_ANY	
EventSel=48H, UMask=01H, AnyThread=1, CMask=1	Cycles with L1D load Misses outstanding from any thread on physical core.
L1D_PEND_MISS.FB_FULL	
EventSel=48H, UMask=02H, CMask=1	Cycles a demand request was blocked due to Fill Buffers inavailability.
DTLB_STORE_MISSES.MISS_CAUSES_A_WALK	
EventSel=49H, UMask=01H	This event counts store misses in all DTLB levels that cause page walks of any page size (4K/2M/4M/1G).
DTLB_STORE_MISSES.WALK_COMPLETED_4K	
EventSel=49H, UMask=02H	This event counts store misses in all DTLB levels that cause a completed page walk (4K page size). The page walk can end with or without a fault.
DTLB_STORE_MISSES.WALK_COMPLETED_2M_4M	
EventSel=49H, UMask=04H	This event counts store misses in all DTLB levels that cause a completed page walk (2M and 4M page sizes). The page walk can end with or without a fault.
DTLB_STORE_MISSES.WALK_COMPLETED_1G	
EventSel=49H, UMask=08H	This event counts store misses in all DTLB levels that cause a completed page walk (1G page size). The page walk can end with or without a fault.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
DTLB_STORE_MISSES.WALK_COMPLETED	
EventSel=49H, UMask=0EH	Store misses in all DTLB levels that cause completed page walks.
DTLB_STORE_MISSES.WALK_DURATION	
EventSel=49H, UMask=10H	This event counts the number of cycles while PMH is busy with the page walk.
DTLB_STORE_MISSES.STLB_HIT_4K	
EventSel=49H, UMask=20H	Store misses that miss the DTLB and hit the STLB (4K).
DTLB_STORE_MISSES.STLB_HIT_2M	
EventSel=49H, UMask=40H	Store misses that miss the DTLB and hit the STLB (2M).
DTLB_STORE_MISSES.STLB_HIT	
EventSel=49H, UMask=60H	Store operations that miss the first TLB level but hit the second and do not cause page walks.
LOAD_HIT_PRE.SW_PF	
EventSel=4CH, UMask=01H	This event counts all not software-prefetch load dispatches that hit the fill buffer (FB) allocated for the software prefetch. It can also be incremented by some lock instructions. So it should only be used with profiling so that the locks can be excluded by asm inspection of the nearby instructions.
LOAD_HIT_PRE.HW_PF	
EventSel=4CH, UMask=02H	This event counts all not software-prefetch load dispatches that hit the fill buffer (FB) allocated for the hardware prefetch.
EPT.WALK_CYCLES	
EventSel=4FH, UMask=10H	This event counts cycles for an extended page table walk. The Extended Page directory cache differs from standard TLB caches by the operating system that use it. Virtual machine operating systems use the extended page directory cache, while guest operating systems use the standard TLB caches.
L1D.REPLACEMENT	
EventSel=51H, UMask=01H	This event counts L1D data line replacements including opportunistic replacements, and replacements that require stall-for-replace or block-for-replace.
TX_MEM.ABORT_CONFLICT	
EventSel=54H, UMask=01H	Number of times a TSX line had a cache conflict.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
TX_MEM.ABORT_CAPACITY_WRITE	
EventSel=54H, UMask=02H	Number of times a TSX Abort was triggered due to an evicted line caused by a transaction overflow.
TX_MEM.ABORT_HLE_STORE_TO_ELIDED_LOCK	
EventSel=54H, UMask=04H	Number of times a TSX Abort was triggered due to a non-release/commit store to lock.
TX_MEM.ABORT_HLE_ELISION_BUFFER_NOT_EMPTY	
EventSel=54H, UMask=08H	Number of times a TSX Abort was triggered due to commit but Lock Buffer not empty.
TX_MEM.ABORT_HLE_ELISION_BUFFER_MISMATCH	
EventSel=54H, UMask=10H	Number of times a TSX Abort was triggered due to release/commit but data and address mismatch.
TX_MEM.ABORT_HLE_ELISION_BUFFER_UNSUPPORTED_ALIGNMENT	
EventSel=54H, UMask=20H	Number of times a TSX Abort was triggered due to attempting an unsupported alignment from Lock Buffer.
TX_MEM.HLE_ELISION_BUFFER_FULL	
EventSel=54H, UMask=40H	Number of times we could not allocate Lock Buffer.
MOVE_ELIMINATION.INT_ELIMINATED	
EventSel=58H, UMask=01H	Number of integer Move Elimination candidate uops that were eliminated.
MOVE_ELIMINATION.SIMD_ELIMINATED	
EventSel=58H, UMask=02H	Number of SIMD Move Elimination candidate uops that were eliminated.
MOVE_ELIMINATION.INT_NOT_ELIMINATED	
EventSel=58H, UMask=04H	Number of integer Move Elimination candidate uops that were not eliminated.
MOVE_ELIMINATION.SIMD_NOT_ELIMINATED	
EventSel=58H, UMask=08H	Number of SIMD Move Elimination candidate uops that were not eliminated.
CPL_CYCLES.RING0	
EventSel=5CH, UMask=01H	This event counts the unhalted core cycles during which the thread is in the ring 0 privileged mode.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
CPL_CYCLES.RING0_TRANS	
EventSel=5CH, UMask=01H, EdgeDetect=1, CMask=1	This event counts when there is a transition from ring 1,2 or 3 to ring0.
CPL_CYCLES.RING123	
EventSel=5CH, UMask=02H	This event counts unhalted core cycles during which the thread is in rings 1, 2, or 3.
TX_EXEC.MISC1	
EventSel=5DH, UMask=01H	Counts the number of times a class of instructions that may cause a transactional abort was executed. Since this is the count of execution, it may not always cause a transactional abort.
TX_EXEC.MISC2	
EventSel=5DH, UMask=02H	Unfriendly TSX abort triggered by a vzeroupper instruction.
TX_EXEC.MISC3	
EventSel=5DH, UMask=04H	Unfriendly TSX abort triggered by a nest count that is too deep.
TX_EXEC.MISC4	
EventSel=5DH, UMask=08H	RTM region detected inside HLE.
TX_EXEC.MISC5	
EventSel=5DH, UMask=10H	Counts the number of times an HLE XACQUIRE instruction was executed inside an RTM transactional region.
RS_EVENTS.EMPTY_CYCLES	
EventSel=5EH, UMask=01H	This event counts cycles during which the reservation station (RS) is empty for the thread. Note: In ST-mode, not active thread should drive 0. This is usually caused by severely costly branch mispredictions, or allocator/FE issues.
RS_EVENTS.EMPTY_END	
EventSel=5EH, UMask=01H, EdgeDetect=1, Invert=1, CMask=1	Counts end of periods where the Reservation Station (RS) was empty. Could be useful to precisely locate Frontend Latency Bound issues.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD	
EventSel=60H, UMask=01H	This event counts the number of offcore outstanding Demand Data Read transactions in the super queue (SQ) every cycle. A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor. See the corresponding Umask under OFFCORE_REQUESTS. Note: A prefetch promoted to Demand is counted from the promotion point.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_DATA_RD	
EventSel=60H, UMask=01H, CMask=1	This event counts cycles when offcore outstanding Demand Data Read transactions are present in the super queue (SQ). A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor (SQ de-allocation).
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD_GE_6	
EventSel=60H, UMask=01H, CMask=6	Cycles with at least 6 offcore outstanding Demand Data Read transactions in uncore queue.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_CODE_RD	
EventSel=60H, UMask=02H	This event counts the number of offcore outstanding Code Reads transactions in the super queue every cycle. The "Offcore outstanding" state of the transaction lasts from the L2 miss until the sending transaction completion to requestor (SQ deallocation). See the corresponding Umask under OFFCORE_REQUESTS.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_RFO	
EventSel=60H, UMask=04H	This event counts the number of offcore outstanding RFO (store) transactions in the super queue (SQ) every cycle. A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor (SQ de-allocation). See corresponding Umask under OFFCORE_REQUESTS.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_RFO	
EventSel=60H, UMask=04H, CMask=1	This event counts the number of offcore outstanding demand rfo Reads transactions in the super queue every cycle. The "Offcore outstanding" state of the transaction lasts from the L2 miss until the sending transaction completion to requestor (SQ deallocation). See the corresponding Umask under OFFCORE_REQUESTS.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
OFFCORE_REQUESTS_OUTSTANDING.ALL_DATA_RD	
EventSel=60H, UMask=08H	This event counts the number of offcore outstanding cacheable Core Data Read transactions in the super queue every cycle. A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor (SQ de-allocation). See corresponding Umask under OFFCORE_REQUESTS.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DATA_RD	
EventSel=60H, UMask=08H, CMask=1	This event counts cycles when offcore outstanding cacheable Core Data Read transactions are present in the super queue. A transaction is considered to be in the Offcore outstanding state between L2 miss and transaction completion sent to requestor (SQ de-allocation). See corresponding Umask under OFFCORE_REQUESTS.
LOCK_CYCLES.SPLIT_LOCK_UC_LOCK_DURATION	
EventSel=63H, UMask=01H	This event counts cycles in which the L1 and L2 are locked due to a UC lock or split lock. A lock is asserted in case of locked memory access, due to noncacheable memory, locked operation that spans two cache lines, or a page walk from the noncacheable page table. L1D and L2 locks have a very high performance penalty and it is highly recommended to avoid such access.
LOCK_CYCLES.CACHE_LOCK_DURATION	
EventSel=63H, UMask=02H	This event counts the number of cycles when the L1D is locked. It is a superset of the 0x1 mask (BUS_LOCK_CLOCKS.BUS_LOCK_DURATION).
IDQ.EMPTY	
EventSel=79H, UMask=02H	This counts the number of cycles that the instruction decoder queue is empty and can indicate that the application may be bound in the front end. It does not determine whether there are uops being delivered to the Alloc stage since uops can be delivered by bypass skipping the Instruction Decode Queue (IDQ) when it is empty.
IDQ.MITE_UOPS	
EventSel=79H, UMask=04H	This event counts the number of uops delivered to Instruction Decode Queue (IDQ) from the MITE path. Counting includes uops that may "bypass" the IDQ. This also means that uops are not being delivered from the Decode Stream Buffer (DSB).

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
IDQ.MITE_CYCLES	
EventSel=79H, UMask=04H, CMask=1	This event counts cycles during which uops are being delivered to Instruction Decode Queue (IDQ) from the MITE path. Counting includes uops that may "bypass" the IDQ.
IDQ.DSB_UOPS	
EventSel=79H, UMask=08H	This event counts the number of uops delivered to Instruction Decode Queue (IDQ) from the Decode Stream Buffer (DSB) path. Counting includes uops that may "bypass" the IDQ.
IDQ.DSB_CYCLES	
EventSel=79H, UMask=08H, CMask=1	This event counts cycles during which uops are being delivered to Instruction Decode Queue (IDQ) from the Decode Stream Buffer (DSB) path. Counting includes uops that may "bypass" the IDQ.
IDQ.MS_DSB_UOPS	
EventSel=79H, UMask=10H	This event counts the number of uops initiated by Decode Stream Buffer (DSB) that are being delivered to Instruction Decode Queue (IDQ) while the Microcode Sequencer (MS) is busy. Counting includes uops that may "bypass" the IDQ.
IDQ.MS_DSB_CYCLES	
EventSel=79H, UMask=10H, CMask=1	This event counts cycles during which uops initiated by Decode Stream Buffer (DSB) are being delivered to Instruction Decode Queue (IDQ) while the Microcode Sequencer (MS) is busy. Counting includes uops that may "bypass" the IDQ.
IDQ.MS_DSB_OCCUR	
EventSel=79H, UMask=10H, EdgeDetect=1, CMask=1	This event counts the number of deliveries to Instruction Decode Queue (IDQ) initiated by Decode Stream Buffer (DSB) while the Microcode Sequencer (MS) is busy. Counting includes uops that may "bypass" the IDQ.
IDQ.ALL_DSB_CYCLES_4_UOPS	
EventSel=79H, UMask=18H, CMask=4	This event counts the number of cycles 4 uops were delivered to Instruction Decode Queue (IDQ) from the Decode Stream Buffer (DSB) path. Counting includes uops that may "bypass" the IDQ.
IDQ.ALL_DSB_CYCLES_ANY_UOPS	
EventSel=79H, UMask=18H, CMask=1	This event counts the number of cycles uops were delivered to Instruction Decode Queue (IDQ) from the Decode Stream Buffer (DSB) path. Counting includes uops that may "bypass" the IDQ.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
IDQ.MS_MITE_UOPS	
EventSel=79H, UMask=20H	This event counts the number of uops initiated by MITE and delivered to Instruction Decode Queue (IDQ) while the Microcode Sequenser (MS) is busy. Counting includes uops that may "bypass" the IDQ.
IDQ.ALL_MITE_CYCLES_4_UOPS	
EventSel=79H, UMask=24H, CMask=4	This event counts the number of cycles 4 uops were delivered to Instruction Decode Queue (IDQ) from the MITE path. Counting includes uops that may "bypass" the IDQ. This also means that uops are not being delivered from the Decode Stream Buffer (DSB).
IDQ.ALL_MITE_CYCLES_ANY_UOPS	
EventSel=79H, UMask=24H, CMask=1	This event counts the number of cycles uops were delivered to Instruction Decode Queue (IDQ) from the MITE path. Counting includes uops that may "bypass" the IDQ. This also means that uops are not being delivered from the Decode Stream Buffer (DSB).
IDQ.MS_UOPS	
EventSel=79H, UMask=30H	This event counts the total number of uops delivered to Instruction Decode Queue (IDQ) while the Microcode Sequenser (MS) is busy. Counting includes uops that may "bypass" the IDQ. Uops maybe initiated by Decode Stream Buffer (DSB) or MITE.
IDQ.MS_CYCLES	
EventSel=79H, UMask=30H, CMask=1	This event counts cycles during which uops are being delivered to Instruction Decode Queue (IDQ) while the Microcode Sequenser (MS) is busy. Counting includes uops that may "bypass" the IDQ. Uops maybe initiated by Decode Stream Buffer (DSB) or MITE.
IDQ.MS_SWITCHES	
EventSel=79H, UMask=30H, EdgeDetect=1, CMask=1	Number of switches from DSB (Decode Stream Buffer) or MITE (legacy decode pipeline) to the Microcode Sequencer.
IDQ.MITE_ALL_UOPS	
EventSel=79H, UMask=3CH	This event counts the number of uops delivered to Instruction Decode Queue (IDQ) from the MITE path. Counting includes uops that may "bypass" the IDQ. This also means that uops are not being delivered from the Decode Stream Buffer (DSB).

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
ICACHE.HIT	
EventSel=80H, UMask=01H	This event counts the number of both cacheable and noncacheable Instruction Cache, Streaming Buffer and Victim Cache Reads including UC fetches.
ICACHE.MISSES	
EventSel=80H, UMask=02H	This event counts the number of instruction cache, streaming buffer and victim cache misses. Counting includes UC accesses.
ICACHE.IFDATA_STALL	
EventSel=80H, UMask=04H	This event counts cycles during which the demand fetch waits for data (wfdM104H) from L2 or iSB (opportunistic hit).
ITLB_MISSES.MISS_CAUSES_A_WALK	
EventSel=85H, UMask=01H	This event counts store misses in all DTLB levels that cause page walks of any page size (4K/2M/4M/1G).
ITLB_MISSES.WALK_COMPLETED_4K	
EventSel=85H, UMask=02H	This event counts store misses in all DTLB levels that cause a completed page walk (4K page size). The page walk can end with or without a fault.
ITLB_MISSES.WALK_COMPLETED_2M_4M	
EventSel=85H, UMask=04H	This event counts store misses in all DTLB levels that cause a completed page walk (2M and 4M page sizes). The page walk can end with or without a fault.
ITLB_MISSES.WALK_COMPLETED_1G	
EventSel=85H, UMask=08H	This event counts store misses in all DTLB levels that cause a completed page walk (1G page size). The page walk can end with or without a fault.
ITLB_MISSES.WALK_COMPLETED	
EventSel=85H, UMask=0EH	Misses in all ITLB levels that cause completed page walks.
ITLB_MISSES.WALK_DURATION	
EventSel=85H, UMask=10H	This event counts the number of cycles while PMH is busy with the page walk.
ITLB_MISSES.STLB_HIT_4K	
EventSel=85H, UMask=20H	Core misses that miss the DTLB and hit the STLB (4K).

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
ITLB_MISSES.STLB_HIT_2M	
EventSel=85H, UMask=40H	Code misses that miss the DTLB and hit the STLB (2M).
ITLB_MISSES.STLB_HIT	
EventSel=85H, UMask=60H	Operations that miss the first ITLB level but hit the second and do not cause any page walks.
ILD_STALL.LCP	
EventSel=87H, UMask=01H	This event counts stalls occured due to changing prefix length (66, 67 or REX.W when they change the length of the decoded instruction). Occurrences counting is proportional to the number of prefixes in a 16B-line. This may result in the following penalties: three-cycle penalty for each LCP in a 16-byte chunk.
BR_INST_EXEC.NONTAKEN_CONDITIONAL	
EventSel=88H, UMask=41H	This event counts not taken macro-conditional branch instructions.
BR_INST_EXEC.TAKEN_CONDITIONAL	
EventSel=88H, UMask=81H	This event counts taken speculative and retired macro-conditional branch instructions.
BR_INST_EXEC.TAKEN_DIRECT_JUMP	
EventSel=88H, UMask=82H	This event counts taken speculative and retired macro-conditional branch instructions excluding calls and indirect branches.
BR_INST_EXEC.TAKEN_INDIRECT_JUMP_NON_CALL_RET	
EventSel=88H, UMask=84H	This event counts taken speculative and retired indirect branches excluding calls and return branches.
BR_INST_EXEC.TAKEN_INDIRECT_NEAR_RETURN	
EventSel=88H, UMask=88H	This event counts taken speculative and retired indirect branches that have a return mnemonic.
BR_INST_EXEC.TAKEN_DIRECT_NEAR_CALL	
EventSel=88H, UMask=90H	This event counts taken speculative and retired direct near calls.
BR_INST_EXEC.TAKEN_INDIRECT_NEAR_CALL	
EventSel=88H, UMask=A0H	This event counts taken speculative and retired indirect calls including both register and memory indirect.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
BR_INST_EXEC.ALL_CONDITIONAL	
EventSel=88H, UMask=C1H	This event counts both taken and not taken speculative and retired macro-conditional branch instructions.
BR_INST_EXEC.ALL_DIRECT_JMP	
EventSel=88H, UMask=C2H	This event counts both taken and not taken speculative and retired macro-unconditional branch instructions, excluding calls and indirects.
BR_INST_EXEC.ALL_INDIRECT_JUMP_NON_CALL_RET	
EventSel=88H, UMask=C4H	This event counts both taken and not taken speculative and retired indirect branches excluding calls and return branches.
BR_INST_EXEC.ALL_INDIRECT_NEAR_RETURN	
EventSel=88H, UMask=C8H	This event counts both taken and not taken speculative and retired indirect branches that have a return mnemonic.
BR_INST_EXEC.ALL_DIRECT_NEAR_CALL	
EventSel=88H, UMask=D0H	This event counts both taken and not taken speculative and retired direct near calls.
BR_INST_EXEC.ALL_BRANCHES	
EventSel=88H, UMask=FFH	This event counts both taken and not taken speculative and retired branch instructions.
BR_MISP_EXEC.NONTAKEN_CONDITIONAL	
EventSel=89H, UMask=41H	This event counts not taken speculative and retired mispredicted macro conditional branch instructions.
BR_MISP_EXEC.TAKEN_CONDITIONAL	
EventSel=89H, UMask=81H	This event counts taken speculative and retired mispredicted macro conditional branch instructions.
BR_MISP_EXEC.TAKEN_INDIRECT_JUMP_NON_CALL_RET	
EventSel=89H, UMask=84H	This event counts taken speculative and retired mispredicted indirect branches excluding calls and returns.
BR_MISP_EXEC.TAKEN_RETURN_NEAR	
EventSel=89H, UMask=88H	This event counts taken speculative and retired mispredicted indirect branches that have a return mnemonic.
BR_MISP_EXEC.TAKEN_INDIRECT_NEAR_CALL	
EventSel=89H, UMask=A0H	Taken speculative and retired mispredicted indirect calls.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
BR_MISP_EXEC.ALL_CONDITIONAL	
EventSel=89H, UMask=C1H	This event counts both taken and not taken speculative and retired mispredicted macro conditional branch instructions.
BR_MISP_EXEC.ALL_INDIRECT_JUMP_NON_CALL_RET	
EventSel=89H, UMask=C4H	This event counts both taken and not taken mispredicted indirect branches excluding calls and returns.
BR_MISP_EXEC.ALL_BRANCHES	
EventSel=89H, UMask=FFH	This event counts both taken and not taken speculative and retired mispredicted branch instructions.
IDQ_UOPS_NOT_DELIVERED.CORE	
EventSel=9CH, UMask=01H	This event counts the number of uops not delivered to Resource Allocation Table (RAT) per thread adding "4 – x" when Resource Allocation Table (RAT) is not stalled and Instruction Decode Queue (IDQ) delivers x uops to Resource Allocation Table (RAT) (where x belongs to {0,1,2,3}). Counting does not cover cases when: a. IDQ-Resource Allocation Table (RAT) pipe serves the other thread; b. Resource Allocation Table (RAT) is stalled for the thread (including uop drops and clear BE conditions); c. Instruction Decode Queue (IDQ) delivers four uops.
IDQ_UOPS_NOT_DELIVERED.CYCLES_0_UOPS_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=4	This event counts, on the per-thread basis, cycles when no uops are delivered to Resource Allocation Table (RAT). IDQ_Uops_Not_Delivered.core =4.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_1_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=3	This event counts, on the per-thread basis, cycles when less than 1 uop is delivered to Resource Allocation Table (RAT). IDQ_Uops_Not_Delivered.core >=3.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_2_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=2	Cycles with less than 2 uops delivered by the front end.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_3_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=1	Cycles with less than 3 uops delivered by the front end.
IDQ_UOPS_NOT_DELIVERED.CYCLES_FE_WAS_OK	
EventSel=9CH, UMask=01H, Invert=1, CMask=1	Counts cycles FE delivered 4 uops or Resource Allocation Table (RAT) was stalling FE.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
UOP_DISPATCHES_CANCELLED.SIMD_PRF	
EventSel=A0H, UMask=03H	This event counts the number of micro-operations cancelled after they were dispatched from the scheduler to the execution units when the total number of physical register read ports across all dispatch ports exceeds the read bandwidth of the physical register file. The SIMD_PRF subevent applies to the following instructions: VDPPS, DPPS, VPCMPESTRI, PCMPESTRI, VPCMPESTRM, PCMPESTRM, VFMADD*, VFMADDSUB*, VFMSUB*, VMSUBADD*, VFNMADD*, VFNMSUB*. See the Broadwell Optimization Guide for more information.
UOPS_DISPATCHED_PORT.PORT_0	
EventSel=A1H, UMask=01H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 0.
UOPS_EXECUTED_PORT.PORT_0_CORE	
EventSel=A1H, UMask=01H, AnyThread=1	Cycles per core when uops are exectuted in port 0.
UOPS_EXECUTED_PORT.PORT_0	
EventSel=A1H, UMask=01H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 0.
UOPS_DISPATCHED_PORT.PORT_1	
EventSel=A1H, UMask=02H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 1.
UOPS_EXECUTED_PORT.PORT_1_CORE	
EventSel=A1H, UMask=02H, AnyThread=1	Cycles per core when uops are exectuted in port 1.
UOPS_EXECUTED_PORT.PORT_1	
EventSel=A1H, UMask=02H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 1.
UOPS_DISPATCHED_PORT.PORT_2	
EventSel=A1H, UMask=04H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 2.
UOPS_EXECUTED_PORT.PORT_2_CORE	
EventSel=A1H, UMask=04H, AnyThread=1	Cycles per core when uops are dispatched to port 2.
UOPS_EXECUTED_PORT.PORT_2	
EventSel=A1H, UMask=04H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 2.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
UOPS_DISPATCHED_PORT.PORT_3	
EventSel=A1H, UMask=08H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 3.
UOPS_EXECUTED_PORT.PORT_3_CORE	
EventSel=A1H, UMask=08H, AnyThread=1	Cycles per core when uops are dispatched to port 3.
UOPS_EXECUTED_PORT.PORT_3	
EventSel=A1H, UMask=08H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 3.
UOPS_DISPATCHED_PORT.PORT_4	
EventSel=A1H, UMask=10H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 4.
UOPS_EXECUTED_PORT.PORT_4_CORE	
EventSel=A1H, UMask=10H, AnyThread=1	Cycles per core when uops are exectuted in port 4.
UOPS_EXECUTED_PORT.PORT_4	
EventSel=A1H, UMask=10H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 4.
UOPS_DISPATCHED_PORT.PORT_5	
EventSel=A1H, UMask=20H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 5.
UOPS_EXECUTED_PORT.PORT_5_CORE	
EventSel=A1H, UMask=20H, AnyThread=1	Cycles per core when uops are exectuted in port 5.
UOPS_EXECUTED_PORT.PORT_5	
EventSel=A1H, UMask=20H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 5.
UOPS_DISPATCHED_PORT.PORT_6	
EventSel=A1H, UMask=40H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 6.
UOPS_EXECUTED_PORT.PORT_6_CORE	
EventSel=A1H, UMask=40H, AnyThread=1	Cycles per core when uops are exectuted in port 6.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
UOPS_EXECUTED_PORT.PORT_6	
EventSel=A1H, UMask=40H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 6.
UOPS_DISPATCHED_PORT.PORT_7	
EventSel=A1H, UMask=80H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 7.
UOPS_EXECUTED_PORT.PORT_7_CORE	
EventSel=A1H, UMask=80H, AnyThread=1	Cycles per core when uops are dispatched to port 7.
UOPS_EXECUTED_PORT.PORT_7	
EventSel=A1H, UMask=80H	This event counts, on the per-thread basis, cycles during which uops are dispatched from the Reservation Station (RS) to port 7.
RESOURCE_STALLS.ANY	
EventSel=A2H, UMask=01H	This event counts resource-related stall cycles. Reasons for stalls can be as follows: - *any* u-arch structure got full (LB, SB, RS, ROB, BOB, LM, Physical Register Reclaim Table (PRRT), or Physical History Table (PHT) slots) - *any* u-arch structure got empty (like INT/SIMD FreeLists) - FPU control word (FPCW), MXCSR and others. This counts cycles that the pipeline backend blocked uop delivery from the front end.
RESOURCE_STALLS.RS	
EventSel=A2H, UMask=04H	This event counts stall cycles caused by absence of eligible entries in the reservation station (RS). This may result from RS overflow, or from RS deallocation because of the RS array Write Port allocation scheme (each RS entry has two write ports instead of four. As a result, empty entries could not be used, although RS is not really full). This counts cycles that the pipeline backend blocked uop delivery from the front end.
RESOURCE_STALLS.SB	
EventSel=A2H, UMask=08H	This event counts stall cycles caused by the store buffer (SB) overflow (excluding draining from synch). This counts cycles that the pipeline backend blocked uop delivery from the front end.
RESOURCE_STALLS.ROB	
EventSel=A2H, UMask=10H	This event counts ROB full stall cycles. This counts cycles that the pipeline backend blocked uop delivery from the front end.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
CYCLE_ACTIVITY.CYCLES_L2_PENDING	
EventSel=A3H, UMask=01H, CMask=1	Counts number of cycles the CPU has at least one pending demand* load request missing the L2 cache.
CYCLE_ACTIVITY.CYCLES_L2_MISS	
EventSel=A3H, UMask=01H, CMask=1	Cycles while L2 cache miss demand load is outstanding.
CYCLE_ACTIVITY.CYCLES_LDM_PENDING	
EventSel=A3H, UMask=02H, CMask=2	Counts number of cycles the CPU has at least one pending demand load request (that is cycles with non-completed load waiting for its data from memory subsystem).
CYCLE_ACTIVITY.CYCLES_MEM_ANY	
EventSel=A3H, UMask=02H, CMask=2	Cycles while memory subsystem has an outstanding load.
CYCLE_ACTIVITY.CYCLES_NO_EXECUTE	
EventSel=A3H, UMask=04H, CMask=4	Counts number of cycles nothing is executed on any execution port.
CYCLE_ACTIVITY.STALLS_TOTAL	
EventSel=A3H, UMask=04H, CMask=4	Total execution stalls.
CYCLE_ACTIVITY.STALLS_L2_PENDING	
EventSel=A3H, UMask=05H, CMask=5	Counts number of cycles nothing is executed on any execution port, while there was at least one pending demand* load request missing the L2 cache.(as a footprint) * includes also L1 HW prefetch requests that may or may not be required by demands.
CYCLE_ACTIVITY.STALLS_L2_MISS	
EventSel=A3H, UMask=05H, CMask=5	Execution stalls while L2 cache miss demand load is outstanding.
CYCLE_ACTIVITY.STALLS_LDM_PENDING	
EventSel=A3H, UMask=06H, CMask=6	Counts number of cycles nothing is executed on any execution port, while there was at least one pending demand load request.
CYCLE_ACTIVITY.STALLS_MEM_ANY	
EventSel=A3H, UMask=06H, CMask=6	Execution stalls while memory subsystem has an outstanding load.
CYCLE_ACTIVITY.CYCLES_L1D_PENDING	
EventSel=A3H, UMask=08H, CMask=8	Counts number of cycles the CPU has at least one pending demand load request missing the L1 data cache.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
CYCLE_ACTIVITY.CYCLES_L1D_MISS	
EventSel=A3H, UMask=08H, CMask=8	Cycles while L1 cache miss demand load is outstanding.
CYCLE_ACTIVITY.STALLS_L1D_PENDING	
EventSel=A3H, UMask=0CH, CMask=12	Counts number of cycles nothing is executed on any execution port, while there was at least one pending demand load request missing the L1 data cache.
CYCLE_ACTIVITY.STALLS_L1D_MISS	
EventSel=A3H, UMask=0CH, CMask=12	Execution stalls while L1 cache miss demand load is outstanding.
LSD.UOPS	
EventSel=A8H, UMask=01H	Number of Uops delivered by the LSD. .
LSD.CYCLES_4_UOPS	
EventSel=A8H, UMask=01H, CMask=4	Cycles 4 Uops delivered by the LSD, but didn't come from the decoder.
LSD.CYCLES_ACTIVE	
EventSel=A8H, UMask=01H, CMask=1	Cycles Uops delivered by the LSD, but didn't come from the decoder.
DSB2MITE_SWITCHES.PENALTY_CYCLES	
EventSel=ABH, UMask=02H	This event counts Decode Stream Buffer (DSB)-to-MITE switch true penalty cycles. These cycles do not include uops routed through because of the switch itself, for example, when Instruction Decode Queue (IDQ) pre-allocation is unavailable, or Instruction Decode Queue (IDQ) is full. SBD-to-MITE switch true penalty cycles happen after the merge mux (MM) receives Decode Stream Buffer (DSB) Sync-indication until receiving the first MITE uop. MM is placed before Instruction Decode Queue (IDQ) to merge uops being fed from the MITE and Decode Stream Buffer (DSB) paths. Decode Stream Buffer (DSB) inserts the Sync-indication whenever a Decode Stream Buffer (DSB)-to-MITE switch occurs. Penalty: A Decode Stream Buffer (DSB) hit followed by a Decode Stream Buffer (DSB) miss can cost up to six cycles in which no uops are delivered to the IDQ. Most often, such switches from the Decode Stream Buffer (DSB) to the legacy pipeline cost 0-2 cycles.
ITLB.ITLB_FLUSH	
EventSel=AEH, UMask=01H	This event counts the number of flushes of the big or small ITLB pages. Counting include both TLB Flush (covering all sets) and TLB Set Clear (set-specific).

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
OFFCORE_REQUESTS.DEMAND_DATA_RD	
EventSel=B0H, UMask=01H	This event counts the Demand Data Read requests sent to uncore. Use it in conjunction with OFFCORE_REQUESTS_OUTSTANDING to determine average latency in the uncore.
OFFCORE_REQUESTS.DEMAND_CODE_RD	
EventSel=B0H, UMask=02H	This event counts both cacheable and noncachaeble code read requests.
OFFCORE_REQUESTS.DEMAND_RFO	
EventSel=B0H, UMask=04H	This event counts the demand RFO (read for ownership) requests including regular RFOs, locks, ItoM.
OFFCORE_REQUESTS.ALL_DATA_RD	
EventSel=B0H, UMask=08H	This event counts the demand and prefetch data reads. All Core Data Reads include cacheable "Demands" and L2 prefetchers (not L3 prefetchers). Counting also covers reads due to page walks resulted from any request type.
UOPS_EXECUTED.THREAD	
EventSel=B1H, UMask=01H	Number of uops to be executed per-thread each cycle.
UOPS_EXECUTED.STALL_CYCLES	
EventSel=B1H, UMask=01H, Invert=1, CMask=1	This event counts cycles during which no uops were dispatched from the Reservation Station (RS) per thread.
UOPS_EXECUTED.CYCLES_GE_1_UOP_EXEC	
EventSel=B1H, UMask=01H, CMask=1	Cycles where at least 1 uop was executed per-thread.
UOPS_EXECUTED.CYCLES_GE_2_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=2	Cycles where at least 2 uops were executed per-thread.
UOPS_EXECUTED.CYCLES_GE_3_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=3	Cycles where at least 3 uops were executed per-thread.
UOPS_EXECUTED.CYCLES_GE_4_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=4	Cycles where at least 4 uops were executed per-thread.
UOPS_EXECUTED.CORE	
EventSel=B1H, UMask=02H	Number of uops executed from any thread.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
UOPS_EXECUTED.CORE_CYCLES_GE_1	
EventSel=B1H, UMask=02H, CMask=1	Cycles at least 1 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_2	
EventSel=B1H, UMask=02H, CMask=2	Cycles at least 2 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_3	
EventSel=B1H, UMask=02H, CMask=3	Cycles at least 3 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_4	
EventSel=B1H, UMask=02H, CMask=4	Cycles at least 4 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_NONE	
EventSel=B1H, UMask=02H, Invert=1	Cycles with no micro-ops executed from any thread on physical core.
OFFCORE_REQUESTS_BUFFER.SQ_FULL	
EventSel=B2H, UMask=01H	This event counts the number of cases when the offcore requests buffer cannot take more entries for the core. This can happen when the superqueue does not contain eligible entries, or when L1D writeback pending FIFO requests is full. Note: Writeback pending FIFO has six entries.
PAGE_WALKER_LOADS.DTLB_L1	
EventSel=BCH, UMask=11H	Number of DTLB page walker hits in the L1+FB.
PAGE_WALKER_LOADS.DTLB_L2	
EventSel=BCH, UMask=12H	Number of DTLB page walker hits in the L2.
PAGE_WALKER_LOADS.DTLB_L3	
EventSel=BCH, UMask=14H	Number of DTLB page walker hits in the L3 + XSNP.
PAGE_WALKER_LOADS.DTLB_MEMORY	
EventSel=BCH, UMask=18H	Number of DTLB page walker hits in Memory.
PAGE_WALKER_LOADS.ITLB_L1	
EventSel=BCH, UMask=21H	Number of ITLB page walker hits in the L1+FB.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
PAGE_WALKER_LOADS.ITLB_L2	
EventSel=BCH, UMask=22H	Number of ITLB page walker hits in the L2.
PAGE_WALKER_LOADS.ITLB_L3	
EventSel=BCH, UMask=24H	Number of ITLB page walker hits in the L3 + XSNP.
TLB_FLUSH.DTLB_THREAD	
EventSel=BDH, UMask=01H	This event counts the number of DTLB flush attempts of the thread-specific entries.
TLB_FLUSH.STLB_ANY	
EventSel=BDH, UMask=20H	This event counts the number of any STLB flush attempts (such as entire, VPID, PCID, InvPage, CR3 write, and so on).
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural	This event counts the number of instructions (EOMs) retired. Counting covers macro-fused instructions individually (that is, increments by two).
INST_RETIRED.PREC_DIST	
EventSel=C0H, UMask=01H, Precise	This is a precise version (that is, uses PEBS) of the event that counts instructions retired.
INST_RETIRED.X87	
EventSel=C0H, UMask=02H	This event counts FP operations retired. For X87 FP operations that have no exceptions counting also includes flows that have several X87, or flows that use X87 uops in the exception handling.
OTHER_ASSISTS.AVX_TO_SSE	
EventSel=C1H, UMask=08H	This event counts the number of transitions from AVX-256 to legacy SSE when penalty is applicable.
OTHER_ASSISTS.SSE_TO_AVX	
EventSel=C1H, UMask=10H	This event counts the number of transitions from legacy SSE to AVX-256 when penalty is applicable.
OTHER_ASSISTS.ANY_WB_ASSIST	
EventSel=C1H, UMask=40H	Number of times any microcode assist is invoked by HW upon uop writeback.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
UOPS_RETIRED.ALL	
EventSel=C2H, UMask=01H, Precise	This event counts all actually retired uops. Counting increments by two for micro-fused uops, and by one for macro-fused and other uops. Maximal increment value for one cycle is eight.
UOPS_RETIRED.STALL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=1	This event counts cycles without actually retired uops.
UOPS_RETIRED.TOTAL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=10	Number of cycles using always true condition (uops_ret < 16) applied to non PEBS uops retired event.
UOPS_RETIRED.RETIRE_SLOTS	
EventSel=C2H, UMask=02H, Precise	This event counts the number of retirement slots used.
MACHINE_CLEARS.CYCLES	
EventSel=C3H, UMask=01H	This event counts both thread-specific (TS) and all-thread (AT) nukes.
MACHINE_CLEARS.COUNT	
EventSel=C3H, UMask=01H, EdgeDetect=1, CMask=1	Number of machine clears (nukes) of any type.
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	This event counts the number of memory ordering Machine Clears detected. Memory Ordering Machine Clears can result from one of the following: 1. memory disambiguation, 2. external snoop, or 3. cross SMT-HW-thread snoop (stores) hitting load buffer.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=04H	This event counts self-modifying code (SMC) detected, which causes a machine clear.
MACHINE_CLEARS.MASKMOV	
EventSel=C3H, UMask=20H	Maskmov false fault - counts number of time ucode passes through Maskmov flow due to instruction's mask being 0 while the flow was completed without raising a fault.
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	This event counts all (macro) branch instructions retired.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
BR_INST_RETIRED.CONDITIONAL	
EventSel=C4H, UMask=01H, Precise	This event counts conditional branch instructions retired.
BR_INST_RETIRED.NEAR_CALL	
EventSel=C4H, UMask=02H, Precise	This event counts both direct and indirect near call instructions retired.
BR_INST_RETIRED.NEAR_CALL_R3	
EventSel=C4H, UMask=02H, USR=1,OS=0, Precise	This event counts both direct and indirect macro near call instructions retired (captured in ring 3).
BR_INST_RETIRED.NEAR_RETURN	
EventSel=C4H, UMask=08H, Precise	This event counts return instructions retired.
BR_INST_RETIRED.NOT_TAKEN	
EventSel=C4H, UMask=10H	This event counts not taken branch instructions retired.
BR_INST_RETIRED.NEAR_TAKEN	
EventSel=C4H, UMask=20H, Precise	This event counts taken branch instructions retired.
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=40H	This event counts far branch instructions retired.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	This event counts all mispredicted macro branch instructions retired.
BR_MISP_RETIRED.CONDITIONAL	
EventSel=C5H, UMask=01H, Precise	This event counts mispredicted conditional branch instructions retired.
BR_MISP_RETIRED.RET	
EventSel=C5H, UMask=08H, Precise	This event counts mispredicted return instructions retired.
BR_MISP_RETIRED.NEAR_TAKEN	
EventSel=C5H, UMask=20H, Precise	Number of near branch instructions retired that were mispredicted and taken.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
FP_ARITH_INST_RETIRED.SCALAR_DOUBLE	
EventSel=C7H, UMask=01H	Number of SSE/AVX computational scalar double precision floating-point instructions retired. Each count represents 1 computation. Applies to SSE* and AVX* scalar double precision floating-point instructions: ADD SUB MUL DIV MIN MAX SQRT FM(N)ADD/SUB. FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.SCALAR_SINGLE	
EventSel=C7H, UMask=02H	Number of SSE/AVX computational scalar single precision floating-point instructions retired. Each count represents 1 computation. Applies to SSE* and AVX* scalar single precision floating-point instructions: ADD SUB MUL DIV MIN MAX RCP RSQRT SQRT FM(N)ADD/SUB. FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.SCALAR	
EventSel=C7H, UMask=03H	Number of SSE/AVX computational scalar floating-point instructions retired. Applies to SSE* and AVX* scalar, double and single precision floating-point: ADD SUB MUL DIV MIN MAX RSQRT RCP SQRT FM(N)ADD/SUB. FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.128B_PACKED_DOUBLE	
EventSel=C7H, UMask=04H	Number of SSE/AVX computational 128-bit packed double precision floating-point instructions retired. Each count represents 2 computations. Applies to SSE* and AVX* packed double precision floating-point instructions: ADD SUB MUL DIV MIN MAX SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.128B_PACKED_SINGLE	
EventSel=C7H, UMask=08H	Number of SSE/AVX computational 128-bit packed single precision floating-point instructions retired. Each count represents 4 computations. Applies to SSE* and AVX* packed single precision floating-point instructions: ADD SUB MUL DIV MIN MAX RCP RSQRT SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
FP_ARITH_INST_RETIRED.256B_PACKED_DOUBLE	
EventSel=C7H, UMask=10H	Number of SSE/AVX computational 256-bit packed double precision floating-point instructions retired. Each count represents 4 computations. Applies to SSE* and AVX* packed double precision floating-point instructions: ADD SUB MUL DIV MIN MAX SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.DOUBLE	
EventSel=C7H, UMask=15H	Number of SSE/AVX computational double precision floating-point instructions retired. Applies to SSE* and AVX*scalar, double and single precision floating-point: ADD SUB MUL DIV MIN MAX SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element. ?.
FP_ARITH_INST_RETIRED.256B_PACKED_SINGLE	
EventSel=C7H, UMask=20H	Number of SSE/AVX computational 256-bit packed single precision floating-point instructions retired. Each count represents 8 computations. Applies to SSE* and AVX* packed single precision floating-point instructions: ADD SUB MUL DIV MIN MAX RCP RSQRT SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
FP_ARITH_INST_RETIRED.SINGLE	
EventSel=C7H, UMask=2AH	Number of SSE/AVX computational single precision floating-point instructions retired. Applies to SSE* and AVX*scalar, double and single precision floating-point: ADD SUB MUL DIV MIN MAX RCP RSQRT SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element. ?.
FP_ARITH_INST_RETIRED.PACKED	
EventSel=C7H, UMask=3CH	Number of SSE/AVX computational packed floating-point instructions retired. Applies to SSE* and AVX*, packed, double and single precision floating-point: ADD SUB MUL DIV MIN MAX RSQRT RCP SQRT DPP FM(N)ADD/SUB. DPP and FM(N)ADD/SUB instructions count twice as they perform multiple calculations per element.
HLE_RETIRED.START	
EventSel=C8H, UMask=01H	Number of times we entered an HLE region does not count nested transactions.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
HLE_RETIRED.COMMIT	
EventSel=C8H, UMask=02H	Number of times HLE commit succeeded.
HLE_RETIRED.ABORTED	
EventSel=C8H, UMask=04H, Precise	Number of times HLE abort was triggered.
HLE_RETIRED.ABORTED_MISC1	
EventSel=C8H, UMask=08H	Number of times an HLE abort was attributed to a Memory condition (See TSX_Memory event for additional details).
HLE_RETIRED.ABORTED_MISC2	
EventSel=C8H, UMask=10H	Number of times the TSX watchdog signaled an HLE abort.
HLE_RETIRED.ABORTED_MISC3	
EventSel=C8H, UMask=20H	Number of times a disallowed operation caused an HLE abort.
HLE_RETIRED.ABORTED_MISC4	
EventSel=C8H, UMask=40H	Number of times HLE caused a fault.
HLE_RETIRED.ABORTED_MISC5	
EventSel=C8H, UMask=80H	Number of times HLE aborted and was not due to the abort conditions in subevents 3-6.
RTM_RETIRED.START	
EventSel=C9H, UMask=01H	Number of times we entered an RTM region does not count nested transactions.
RTM_RETIRED.COMMIT	
EventSel=C9H, UMask=02H	Number of times RTM commit succeeded.
RTM_RETIRED.ABORTED	
EventSel=C9H, UMask=04H, Precise	Number of times RTM abort was triggered .
RTM_RETIRED.ABORTED_MISC1	
EventSel=C9H, UMask=08H	Number of times an RTM abort was attributed to a Memory condition (See TSX_Memory event for additional details).
RTM_RETIRED.ABORTED_MISC2	
EventSel=C9H, UMask=10H	Number of times the TSX watchdog signaled an RTM abort.
RTM_RETIRED.ABORTED_MISC3	
EventSel=C9H, UMask=20H	Number of times a disallowed operation caused an RTM abort.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
RTM_RETIRED.ABORTED_MISC4	
EventSel=C9H, UMask=40H	Number of times a RTM caused a fault.
RTM_RETIRED.ABORTED_MISC5	
EventSel=C9H, UMask=80H	Number of times RTM aborted and was not due to the abort conditions in subevents 3-6.
FP_ASSIST.X87_OUTPUT	
EventSel=CAH, UMask=02H	This event counts the number of x87 floating point (FP) micro-code assist (numeric overflow/underflow, inexact result) when the output value (destination register) is invalid.
FP_ASSIST.X87_INPUT	
EventSel=CAH, UMask=04H	This event counts x87 floating point (FP) micro-code assist (invalid operation, denormal operand, SNaN operand) when the input value (one of the source operands to an FP instruction) is invalid.
FP_ASSIST.SIMD_OUTPUT	
EventSel=CAH, UMask=08H	This event counts the number of SSE* floating point (FP) micro-code assist (numeric overflow/underflow) when the output value (destination register) is invalid. Counting covers only cases involving penalties that require micro-code assist intervention.
FP_ASSIST.SIMD_INPUT	
EventSel=CAH, UMask=10H	This event counts any input SSE* FP assist - invalid operation, denormal operand, dividing by zero, SNaN operand. Counting includes only cases involving penalties that required micro-code assist intervention.
FP_ASSIST.ANY	
EventSel=CAH, UMask=1EH, CMask=1	This event counts cycles with any input and output SSE or x87 FP assist. If an input and output assist are detected on the same cycle the event increments by 1.
ROB_MISC_EVENTS.LBR_INSERTS	
EventSel=CCH, UMask=20H	This event counts cases of saving new LBR records by hardware. This assumes proper enabling of LBRs and takes into account LBR filtering done by the LBR_SELECT register.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_4	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x4 , Precise	This event counts loads with latency value being above four.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_8	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x8 , Precise	This event counts loads with latency value being above eight.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_16	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x10 , Precise	This event counts loads with latency value being above 16.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_32	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x20 , Precise	This event counts loads with latency value being above 32.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_64	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x40 , Precise	This event counts loads with latency value being above 64.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_128	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x80 , Precise	This event counts loads with latency value being above 128.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_256	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x100 , Precise	This event counts loads with latency value being above 256.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_512	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x200 , Precise	This event counts loads with latency value being above 512.
MEM_UOPS_RETIRED.STLB_MISS_LOADS	
EventSel=D0H, UMask=11H, Precise	This event counts load uops with true STLB miss retired to the architected path. True STLB miss is an uop triggering page walk that gets completed without blocks, and later gets retired. This page walk can end up with or without a fault.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
MEM_UOPS_RETIRED.STLB_MISS_STORES	
EventSel=D0H, UMask=12H, Precise	This event counts store uops with true STLB miss retired to the architected path. True STLB miss is an uop triggering page walk that gets completed without blocks, and later gets retired. This page walk can end up with or without a fault.
MEM_UOPS_RETIRED.LOCK_LOADS	
EventSel=D0H, UMask=21H, Precise	This event counts load uops with locked access retired to the architected path.
MEM_UOPS_RETIRED.SPLIT_LOADS	
EventSel=D0H, UMask=41H, Precise	This event counts line-splitted load uops retired to the architected path. A line split is across 64B cache-line which includes a page split (4K).
MEM_UOPS_RETIRED.SPLIT_STORES	
EventSel=D0H, UMask=42H, Precise	This event counts line-splitted store uops retired to the architected path. A line split is across 64B cache-line which includes a page split (4K).
MEM_UOPS_RETIRED.ALL_LOADS	
EventSel=D0H, UMask=81H, Precise	This event counts load uops retired to the architected path with a filter on bits 0 and 1 applied. Note: This event counts AVX-256bit load/store double-pump memory uops as a single uop at retirement. This event also counts SW prefetches.
MEM_UOPS_RETIRED.ALL_STORES	
EventSel=D0H, UMask=82H, Precise	This event counts store uops retired to the architected path with a filter on bits 0 and 1 applied. Note: This event counts AVX-256bit load/store double-pump memory uops as a single uop at retirement.
MEM_LOAD_UOPS_RETIRED.L1_HIT	
EventSel=D1H, UMask=01H, Precise	This event counts retired load uops which data sources were hits in the nearest-level (L1) cache. Note: Only two data-sources of L1/FB are applicable for AVX-256bit even though the corresponding AVX load could be serviced by a deeper level in the memory hierarchy. Data source is reported for the Low-half load. This event also counts SW prefetches independent of the actual data source.
MEM_LOAD_UOPS_RETIRED.L2_HIT	
EventSel=D1H, UMask=02H, Precise	This event counts retired load uops which data sources were hits in the mid-level (L2) cache.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
MEM_LOAD_UOPS_RETIRED.L3_HIT	
EventSel=D1H, UMask=04H, Precise	This event counts retired load uops which data sources were data hits in the last-level (L3) cache without snoops required.
MEM_LOAD_UOPS_RETIRED.L1_MISS	
EventSel=D1H, UMask=08H, Precise	This event counts retired load uops which data sources were misses in the nearest-level (L1) cache. Counting excludes unknown and UC data source.
MEM_LOAD_UOPS_RETIRED.L2_MISS	
EventSel=D1H, UMask=10H, Precise	This event counts retired load uops which data sources were misses in the mid-level (L2) cache. Counting excludes unknown and UC data source.
MEM_LOAD_UOPS_RETIRED.L3_MISS	
EventSel=D1H, UMask=20H, Precise	Miss in last-level (L3) cache. Excludes Unknown data-source.
MEM_LOAD_UOPS_RETIRED.HIT_LFB	
EventSel=D1H, UMask=40H, Precise	This event counts retired load uops which data sources were load uops missed L1 but hit a fill buffer due to a preceding miss to the same cache line with the data not ready. Note: Only two data-sources of L1/FB are applicable for AVX-256bit even though the corresponding AVX load could be serviced by a deeper level in the memory hierarchy. Data source is reported for the Low-half load.
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_MISS	
EventSel=D2H, UMask=01H, Precise	This event counts retired load uops which data sources were L3 Hit and a cross-core snoop missed in the on-pkg core cache.
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_HIT	
EventSel=D2H, UMask=02H, Precise	This event counts retired load uops which data sources were L3 hit and a cross-core snoop hit in the on-pkg core cache.
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_HITM	
EventSel=D2H, UMask=04H, Precise	This event counts retired load uops which data sources were HitM responses from a core on same socket (shared L3).
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_NONE	
EventSel=D2H, UMask=08H, Precise	This event counts retired load uops which data sources were hits in the last-level (L3) cache without snoops required.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
MEM_LOAD_UOPS_L3_MISS_RETIRED.LOCAL_DRAM	
EventSel=D3H, UMask=01H, Precise	Retired load uop whose Data Source was: local DRAM either Snoop not needed or Snoop Miss (Rspl).
BACLEARS.ANY	
EventSel=E6H, UMask=1FH	Counts the total number when the front end is resteered, mainly when the BPU cannot provide a correct prediction and this is corrected by other branch handling mechanisms at the front end.
L2_TRANS.DEMAND_DATA_RD	
EventSel=F0H, UMask=01H	This event counts Demand Data Read requests that access L2 cache, including rejects.
L2_TRANS.RFO	
EventSel=F0H, UMask=02H	This event counts Read for Ownership (RFO) requests that access L2 cache.
L2_TRANS.CODE_RD	
EventSel=F0H, UMask=04H	This event counts the number of L2 cache accesses when fetching instructions.
L2_TRANS.ALL_PF	
EventSel=F0H, UMask=08H	This event counts L2 or L3 HW prefetches that access L2 cache including rejects.
L2_TRANS.L1D_WB	
EventSel=F0H, UMask=10H	This event counts L1D writebacks that access L2 cache.
L2_TRANS.L2_FILL	
EventSel=F0H, UMask=20H	This event counts L2 fill requests that access L2 cache.
L2_TRANS.L2_WB	
EventSel=F0H, UMask=40H	This event counts L2 writebacks that access L2 cache.
L2_TRANS.ALL_REQUESTS	
EventSel=F0H, UMask=80H	This event counts transactions that access the L2 pipe including snoops, pagewalks, and so on.
L2_LINES_IN.I	
EventSel=F1H, UMask=01H	This event counts the number of L2 cache lines in the Invalidate state filling the L2. Counting does not cover rejects.

Table 3: Performance Events of the Processor Core Supported by Broadwell Microarchitecture (06_3DH, 06_47H)

Event Name	
Configuration	Description
L2_LINES_IN.S	
EventSel=F1H, UMask=02H	This event counts the number of L2 cache lines in the Shared state filling the L2. Counting does not cover rejects.
L2_LINES_IN.E	
EventSel=F1H, UMask=04H	This event counts the number of L2 cache lines in the Exclusive state filling the L2. Counting does not cover rejects.
L2_LINES_IN.ALL	
EventSel=F1H, UMask=07H	This event counts the number of L2 cache lines filling the L2. Counting does not cover rejects.
L2_LINES_OUT.DEMAND_CLEAN	
EventSel=F2H, UMask=05H	Clean L2 cache lines evicted by demand.
SQ_MISC.SPLIT_LOCK	
EventSel=F4H, UMask=10H	This event counts the number of split locks in the super queue.

Performance Monitoring Events based on Haswell Microarchitecture - Intel Xeon® Processor E5 v3 Family

Performance monitoring events in the processor core of the Intel Xeon® processor E5 v3 family based on the Haswell Microarchitecture are listed in the table below.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed	This event counts the number of instructions retired from execution. For instructions that consist of multiple micro-ops, this event counts the retirement of the last micro-op of the instruction. Counting continues during hardware interrupts, traps, and inside interrupt handlers. INST_RETIRED.ANY is counted by a designated fixed counter, leaving the programmable counters available for other events. Faulting executions of GETSEC/VM entry/VM Exit/MWait will not count as retired instructions.
CPU_CLK_UNHALTED.THREAD	
Architectural, Fixed	This event counts the number of thread cycles while the thread is not in a halt state. The thread enters the halt state when it is running the HLT instruction. The core frequency may change from time to time due to power or thermal throttling.
CPU_CLK_UNHALTED.THREAD_ANY	
AnyThread=1, Architectural, Fixed	Core cycles when at least one thread on the physical core is not in halt state.
CPU_CLK_UNHALTED.REF_TSC	
Architectural, Fixed	This event counts the number of reference cycles when the core is not in a halt state. The core enters the halt state when it is running the HLT instruction or the MWAIT instruction. This event is not affected by core frequency changes (for example, P states, TM2 transitions) but has the same incrementing frequency as the time stamp counter. This event can approximate elapsed time while the core was not in a halt state.
LD_BLOCKS.STORE_FORWARD	
EventSel=03H, UMask=02H	This event counts loads that followed a store to the same address, where the data could not be forwarded inside the pipeline from the store to the load. The most common reason why store forwarding would be blocked is when a load's address range overlaps with a preceding smaller uncompleted store. The penalty for blocked store forwarding is that the load must wait for the store to write its value to the cache before it can be issued.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
LD_BLOCKS.NO_SR	
EventSel=03H, UMask=08H	The number of times that split load operations are temporarily blocked because all resources for handling the split accesses are in use.
MISALIGN_MEM_REF.LOADS	
EventSel=05H, UMask=01H	Speculative cache-line split load uops dispatched to L1D.
MISALIGN_MEM_REF.STORES	
EventSel=05H, UMask=02H	Speculative cache-line split store-address uops dispatched to L1D.
LD_BLOCKS_PARTIAL.ADDRESS_ALIAS	
EventSel=07H, UMask=01H	Aliasing occurs when a load is issued after a store and their memory addresses are offset by 4K. This event counts the number of loads that aliased with a preceding store, resulting in an extended address check in the pipeline which can have a performance impact.
DTLB_LOAD_MISSES.MISS_CAUSES_A_WALK	
EventSel=08H, UMask=01H	Misses in all TLB levels that cause a page walk of any page size.
DTLB_LOAD_MISSES.WALK_COMPLETED_4K	
EventSel=08H, UMask=02H	Completed page walks due to demand load misses that caused 4K page walks in any TLB levels.
DTLB_LOAD_MISSES.WALK_COMPLETED_2M_4M	
EventSel=08H, UMask=04H	Completed page walks due to demand load misses that caused 2M/4M page walks in any TLB levels.
DTLB_LOAD_MISSES.WALK_COMPLETED_1G	
EventSel=08H, UMask=08H	Load miss in all TLB levels causes a page walk that completes. (1G).
DTLB_LOAD_MISSES.WALK_COMPLETED	
EventSel=08H, UMask=0EH	Completed page walks in any TLB of any page size due to demand load misses.
DTLB_LOAD_MISSES.WALK_DURATION	
EventSel=08H, UMask=10H	This event counts cycles when the page miss handler (PMH) is servicing page walks caused by DTLB load misses.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
DTLB_LOAD_MISSES.STLB_HIT_4K	
EventSel=08H, UMask=20H	This event counts load operations from a 4K page that miss the first DTLB level but hit the second and do not cause page walks.
DTLB_LOAD_MISSES.STLB_HIT_2M	
EventSel=08H, UMask=40H	This event counts load operations from a 2M page that miss the first DTLB level but hit the second and do not cause page walks.
DTLB_LOAD_MISSES.STLB_HIT	
EventSel=08H, UMask=60H	Number of cache load STLB hits. No page walk.
DTLB_LOAD_MISSES.PDE_CACHE_MISS	
EventSel=08H, UMask=80H	DTLB demand load misses with low part of linear-to-physical address translation missed.
INT_MISC.RECOVERY_CYCLES	
EventSel=0DH, UMask=03H, CMask=1	This event counts the number of cycles spent waiting for a recovery after an event such as a processor nuke, JEClear, assist, hle/rtm abort etc.
INT_MISC.RECOVERY_CYCLES_ANY	
EventSel=0DH, UMask=03H, AnyThread=1, CMask=1	Core cycles the allocator was stalled due to recovery from earlier clear event for any thread running on the physical core (e.g. misprediction or memory nuke).
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=01H	This event counts the number of uops issued by the Front-end of the pipeline to the Back-end. This event is counted at the allocation stage and will count both retired and non-retired uops.
UOPS_ISSUED.STALL_CYCLES	
EventSel=0EH, UMask=01H, Invert=1, CMask=1	Cycles when Resource Allocation Table (RAT) does not issue Uops to Reservation Station (RS) for the thread.
UOPS_ISSUED.CORE_STALL_CYCLES	
EventSel=0EH, UMask=01H, AnyThread=1, Invert=1, CMask=1	Cycles when Resource Allocation Table (RAT) does not issue Uops to Reservation Station (RS) for all threads.
UOPS_ISSUED.FLAGS_MERGE	
EventSel=0EH, UMask=10H	Number of flags-merge uops allocated. Such uops add delay.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
UOPS_ISSUED.SLOW_LEA	
EventSel=0EH, UMask=20H	Number of slow LEA or similar uops allocated. Such uop has 3 sources (for example, 2 sources + immediate) regardless of whether it is a result of LEA instruction or not.
UOPS_ISSUED.SINGLE_MUL	
EventSel=0EH, UMask=40H	Number of multiply packed/scalar single precision uops allocated.
ARITH.DIVIDER_UOPS	
EventSel=14H, UMask=02H	Any uop executed by the Divider. (This includes all divide uops, sqrt, ...).
L2_RQSTS.DEMAND_DATA_RD_MISS	
EventSel=24H, UMask=21H	Demand data read requests that missed L2, no rejects.
L2_RQSTS.RFO_MISS	
EventSel=24H, UMask=22H	Counts the number of store RFO requests that miss the L2 cache.
L2_RQSTS.CODE_RD_MISS	
EventSel=24H, UMask=24H	Number of instruction fetches that missed the L2 cache.
L2_RQSTS.ALL_DEMAND_MISS	
EventSel=24H, UMask=27H	Demand requests that miss L2 cache.
L2_RQSTS.L2_PF_MISS	
EventSel=24H, UMask=30H	Counts all L2 HW prefetcher requests that missed L2.
L2_RQSTS.MISS	
EventSel=24H, UMask=3FH	All requests that missed L2.
L2_RQSTS.DEMAND_DATA_RD_HIT	
EventSel=24H, UMask=41H	Demand data read requests that hit L2 cache.
L2_RQSTS.RFO_HIT	
EventSel=24H, UMask=42H	Counts the number of store RFO requests that hit the L2 cache.
L2_RQSTS.CODE_RD_HIT	
EventSel=24H, UMask=44H	Number of instruction fetches that hit the L2 cache.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
L2_RQSTS.L2_PF_HIT	
EventSel=24H, UMask=50H	Counts all L2 HW prefetcher requests that hit L2.
L2_RQSTS.ALL_DEMAND_DATA_RD	
EventSel=24H, UMask=E1H	Counts any demand and L1 HW prefetch data load requests to L2.
L2_RQSTS.ALL_RFO	
EventSel=24H, UMask=E2H	Counts all L2 store RFO requests.
L2_RQSTS.ALL_CODE_RD	
EventSel=24H, UMask=E4H	Counts all L2 code requests.
L2_RQSTS.ALL_DEMAND_REFERENCES	
EventSel=24H, UMask=E7H	Demand requests to L2 cache.
L2_RQSTS.ALL_PF	
EventSel=24H, UMask=F8H	Counts all L2 HW prefetcher requests.
L2_RQSTS.REFERENCES	
EventSel=24H, UMask=FFH	All requests to L2 cache.
L2_DEMAND_RQSTS.WB_HIT	
EventSel=27H, UMask=50H	Not rejected writebacks that hit L2 cache.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	This event counts each cache miss condition for references to the last level cache.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	This event counts requests originating from the core that reference a cache line in the last level cache.
CPU_CLK_UNHALTED.THREAD_P	
EventSel=3CH, UMask=00H, Architectural	Counts the number of thread cycles while the thread is not in a halt state. The thread enters the halt state when it is running the HLT instruction. The core frequency may change from time to time due to power or thermal throttling.
CPU_CLK_UNHALTED.THREAD_P_ANY	
EventSel=3CH, UMask=00H, AnyThread=1, Architectural	Core cycles when at least one thread on the physical core is not in halt state.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
CPU_CLK_THREAD_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	Increments at the frequency of XCLK (100 MHz) when not halted.
CPU_CLK_THREAD_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Reference cycles when the at least one thread on the physical core is unhalted (counts at 100 MHz rate).
CPU_CLK_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	Reference cycles when the thread is unhalted. (counts at 100 MHz rate).
CPU_CLK_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Reference cycles when the at least one thread on the physical core is unhalted (counts at 100 MHz rate).
CPU_CLK_THREAD_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Count XClk pulses when this thread is unhalted and the other thread is halted.
CPU_CLK_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Count XClk pulses when this thread is unhalted and the other thread is halted.
L1D_PEND_MISS.PENDING	
EventSel=48H, UMask=01H	Increments the number of outstanding L1D misses every cycle. Set Cmask = 1 and Edge =1 to count occurrences.
L1D_PEND_MISS.PENDING_CYCLES	
EventSel=48H, UMask=01H, CMask=1	Cycles with L1D load Misses outstanding.
L1D_PEND_MISS.PENDING_CYCLES_ANY	
EventSel=48H, UMask=01H, AnyThread=1, CMask=1	Cycles with L1D load Misses outstanding from any thread on physical core.
L1D_PEND_MISS.REQUEST_FB_FULL	
EventSel=48H, UMask=02H	Number of times a request needed a FB entry but there was no entry available for it. That is the FB unavailability was dominant reason for blocking the request. A request includes cacheable/uncacheable demands that is load, store or SW prefetch. HWP are e.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
L1D_PEND_MISS.FB_FULL	
EventSel=48H, UMask=02H, CMask=1	Cycles a demand request was blocked due to Fill Buffers inavailability.
DTLB_STORE_MISSES.MISS_CAUSES_A_WALK	
EventSel=49H, UMask=01H	Miss in all TLB levels causes a page walk of any page size (4K/2M/4M/1G).
DTLB_STORE_MISSES.WALK_COMPLETED_4K	
EventSel=49H, UMask=02H	Completed page walks due to store misses in one or more TLB levels of 4K page structure.
DTLB_STORE_MISSES.WALK_COMPLETED_2M_4M	
EventSel=49H, UMask=04H	Completed page walks due to store misses in one or more TLB levels of 2M/4M page structure.
DTLB_STORE_MISSES.WALK_COMPLETED_1G	
EventSel=49H, UMask=08H	Store misses in all DTLB levels that cause completed page walks. (1G).
DTLB_STORE_MISSES.WALK_COMPLETED	
EventSel=49H, UMask=0EH	Completed page walks due to store miss in any TLB levels of any page size (4K/2M/4M/1G).
DTLB_STORE_MISSES.WALK_DURATION	
EventSel=49H, UMask=10H	This event counts cycles when the page miss handler (PMH) is servicing page walks caused by DTLB store misses.
DTLB_STORE_MISSES.STLB_HIT_4K	
EventSel=49H, UMask=20H	This event counts store operations from a 4K page that miss the first DTLB level but hit the second and do not cause page walks.
DTLB_STORE_MISSES.STLB_HIT_2M	
EventSel=49H, UMask=40H	This event counts store operations from a 2M page that miss the first DTLB level but hit the second and do not cause page walks.
DTLB_STORE_MISSES.STLB_HIT	
EventSel=49H, UMask=60H	Store operations that miss the first TLB level but hit the second and do not cause page walks.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
DTLB_STORE_MISSES.PDE_CACHE_MISS	
EventSel=49H, UMask=80H	DTLB store misses with low part of linear-to-physical address translation missed.
LOAD_HIT_PRE.SW_PF	
EventSel=4CH, UMask=01H	Non-SW-prefetch load dispatches that hit fill buffer allocated for S/W prefetch.
LOAD_HIT_PRE.HW_PF	
EventSel=4CH, UMask=02H	Non-SW-prefetch load dispatches that hit fill buffer allocated for H/W prefetch.
EPT.WALK_CYCLES	
EventSel=4FH, UMask=10H	Cycle count for an Extended Page table walk.
L1D.REPLACEMENT	
EventSel=51H, UMask=01H	This event counts when new data lines are brought into the L1 Data cache, which cause other lines to be evicted from the cache.
TX_MEM.ABORT_CONFLICT	
EventSel=54H, UMask=01H	Number of times a transactional abort was signaled due to a data conflict on a transactionally accessed address.
TX_MEM.ABORT_CAPACITY_WRITE	
EventSel=54H, UMask=02H	Number of times a transactional abort was signaled due to a data capacity limitation for transactional writes.
TX_MEM.ABORT_HLE_STORE_TO_ELIDED_LOCK	
EventSel=54H, UMask=04H	Number of times a HLE transactional region aborted due to a non XRELEASE prefixed instruction writing to an elided lock in the elision buffer.
TX_MEM.ABORT_HLE_ELISION_BUFFER_NOT_EMPTY	
EventSel=54H, UMask=08H	Number of times an HLE transactional execution aborted due to NoAllocatedElisionBuffer being non-zero.
TX_MEM.ABORT_HLE_ELISION_BUFFER_MISMATCH	
EventSel=54H, UMask=10H	Number of times an HLE transactional execution aborted due to XRELEASE lock not satisfying the address and value requirements in the elision buffer.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
TX_MEM.ABORT_HLE_ELISION_BUFFER_UNSUPPORTED_ALIGNMENT	
EventSel=54H, UMask=20H	Number of times an HLE transactional execution aborted due to an unsupported read alignment from the elision buffer.
TX_MEM.HLE_ELISION_BUFFER_FULL	
EventSel=54H, UMask=40H	Number of times HLE lock could not be elided due to ElisionBufferAvailable being zero.
MOVE_ELIMINATION.INT_ELIMINATED	
EventSel=58H, UMask=01H	Number of integer move elimination candidate uops that were eliminated.
MOVE_ELIMINATION.SIMD_ELIMINATED	
EventSel=58H, UMask=02H	Number of SIMD move elimination candidate uops that were eliminated.
MOVE_ELIMINATION.INT_NOT_ELIMINATED	
EventSel=58H, UMask=04H	Number of integer move elimination candidate uops that were not eliminated.
MOVE_ELIMINATION.SIMD_NOT_ELIMINATED	
EventSel=58H, UMask=08H	Number of SIMD move elimination candidate uops that were not eliminated.
CPL_CYCLES.RING0	
EventSel=5CH, UMask=01H	Unhalted core cycles when the thread is in ring 0.
CPL_CYCLES.RING0_TRANS	
EventSel=5CH, UMask=01H, EdgeDetect=1, CMask=1	Number of intervals between processor halts while thread is in ring 0.
CPL_CYCLES.RING123	
EventSel=5CH, UMask=02H	Unhalted core cycles when the thread is not in ring 0.
TX_EXEC.MISC1	
EventSel=5DH, UMask=01H	Counts the number of times a class of instructions that may cause a transactional abort was executed. Since this is the count of execution, it may not always cause a transactional abort.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
TX_EXEC.MISC2	
EventSel=5DH, UMask=02H	Counts the number of times a class of instructions (e.g., vzeroupper) that may cause a transactional abort was executed inside a transactional region.
TX_EXEC.MISC3	
EventSel=5DH, UMask=04H	Counts the number of times an instruction execution caused the transactional nest count supported to be exceeded.
TX_EXEC.MISC4	
EventSel=5DH, UMask=08H	Counts the number of times a XBEGIN instruction was executed inside an HLE transactional region.
TX_EXEC.MISC5	
EventSel=5DH, UMask=10H	Counts the number of times an HLE XACQUIRE instruction was executed inside an RTM transactional region.
RS_EVENTS.EMPTY_CYCLES	
EventSel=5EH, UMask=01H	This event counts cycles when the Reservation Station (RS) is empty for the thread. The RS is a structure that buffers allocated micro-ops from the Front-end. If there are many cycles when the RS is empty, it may represent an underflow of instructions delivered from the Front-end.
RS_EVENTS.EMPTY_END	
EventSel=5EH, UMask=01H, EdgeDetect=1, Invert=1, CMask=1	Counts end of periods where the Reservation Station (RS) was empty. Could be useful to precisely locate Frontend Latency Bound issues.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD	
EventSel=60H, UMask=01H	Offcore outstanding demand data read transactions in SQ to uncore. Set Cmask=1 to count cycles.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_DATA_RD	
EventSel=60H, UMask=01H, CMask=1	Cycles when offcore outstanding Demand Data Read transactions are present in SuperQueue (SQ), queue to uncore.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD_GE_6	
EventSel=60H, UMask=01H, CMask=6	Cycles with at least 6 offcore outstanding Demand Data Read transactions in uncore queue.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_CODE_RD	
EventSel=60H, UMask=02H	Offcore outstanding Demand code Read transactions in SQ to uncore. Set Cmask=1 to count cycles.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_RFO	
EventSel=60H, UMask=04H	Offcore outstanding RFO store transactions in SQ to uncore. Set Cmask=1 to count cycles.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_RFO	
EventSel=60H, UMask=04H, CMask=1	Offcore outstanding demand rfo reads transactions in SuperQueue (SQ), queue to uncore, every cycle.
OFFCORE_REQUESTS_OUTSTANDING.ALL_DATA_RD	
EventSel=60H, UMask=08H	Offcore outstanding cacheable data read transactions in SQ to uncore. Set Cmask=1 to count cycles.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DATA_RD	
EventSel=60H, UMask=08H, CMask=1	Cycles when offcore outstanding cacheable Core Data Read transactions are present in SuperQueue (SQ), queue to uncore.
LOCK_CYCLES.SPLIT_LOCK_UC_LOCK_DURATION	
EventSel=63H, UMask=01H	Cycles in which the L1D and L2 are locked, due to a UC lock or split lock.
LOCK_CYCLES.CACHE_LOCK_DURATION	
EventSel=63H, UMask=02H	Cycles in which the L1D is locked.
IDQ.EMPTY	
EventSel=79H, UMask=02H	Counts cycles the IDQ is empty.
IDQ.MITE_UOPS	
EventSel=79H, UMask=04H	Increment each cycle # of uops delivered to IDQ from MITE path. Set Cmask = 1 to count cycles.
IDQ.MITE_CYCLES	
EventSel=79H, UMask=04H, CMask=1	Cycles when uops are being delivered to Instruction Decode Queue (IDQ) from MITE path.
IDQ.DSB_UOPS	
EventSel=79H, UMask=08H	Increment each cycle. # of uops delivered to IDQ from DSB path. Set Cmask = 1 to count cycles.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
IDQ.DSB_CYCLES	
EventSel=79H, UMask=08H, CMask=1	Cycles when uops are being delivered to Instruction Decode Queue (IDQ) from Decode Stream Buffer (DSB) path.
IDQ.MS_DSB_UOPS	
EventSel=79H, UMask=10H	Increment each cycle # of uops delivered to IDQ when MS_busy by DSB. Set Cmask = 1 to count cycles. Add Edge=1 to count # of delivery.
IDQ.MS_DSB_CYCLES	
EventSel=79H, UMask=10H, CMask=1	Cycles when uops initiated by Decode Stream Buffer (DSB) are being delivered to Instruction Decode Queue (IDQ) while Microcode Sequenser (MS) is busy.
IDQ.MS_DSB_OCCUR	
EventSel=79H, UMask=10H, EdgeDetect=1, CMask=1	Deliveries to Instruction Decode Queue (IDQ) initiated by Decode Stream Buffer (DSB) while Microcode Sequenser (MS) is busy.
IDQ.ALL_DSB_CYCLES_4_UOPS	
EventSel=79H, UMask=18H, CMask=4	Counts cycles DSB is delivered four uops. Set Cmask = 4.
IDQ.ALL_DSB_CYCLES_ANY_UOPS	
EventSel=79H, UMask=18H, CMask=1	Counts cycles DSB is delivered at least one uops. Set Cmask = 1.
IDQ.MS_MITE_UOPS	
EventSel=79H, UMask=20H	Increment each cycle # of uops delivered to IDQ when MS_busy by MITE. Set Cmask = 1 to count cycles.
IDQ.ALL_MITE_CYCLES_4_UOPS	
EventSel=79H, UMask=24H, CMask=4	Counts cycles MITE is delivered four uops. Set Cmask = 4.
IDQ.ALL_MITE_CYCLES_ANY_UOPS	
EventSel=79H, UMask=24H, CMask=1	Counts cycles MITE is delivered at least one uop. Set Cmask = 1.
IDQ.MS_UOPS	
EventSel=79H, UMask=30H	This event counts uops delivered by the Front-end with the assistance of the microcode sequencer. Microcode assists are used for complex instructions or scenarios that can't be handled by the standard decoder. Using other instructions, if possible, will usually improve performance.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
IDQ.MS_CYCLES	
EventSel=79H, UMask=30H, CMask=1	This event counts cycles during which the microcode sequencer assisted the Front-end in delivering uops. Microcode assists are used for complex instructions or scenarios that can't be handled by the standard decoder. Using other instructions, if possible, will usually improve performance.
IDQ.MS_SWITCHES	
EventSel=79H, UMask=30H, EdgeDetect=1, CMask=1	Number of switches from DSB (Decode Stream Buffer) or MITE (legacy decode pipeline) to the Microcode Sequencer.
IDQ.MITE_ALL_UOPS	
EventSel=79H, UMask=3CH	Number of uops delivered to IDQ from any path.
ICACHE.HIT	
EventSel=80H, UMask=01H	Number of Instruction Cache, Streaming Buffer and Victim Cache Reads. both cacheable and noncacheable, including UC fetches.
ICACHE.MISSES	
EventSel=80H, UMask=02H	This event counts Instruction Cache (ICACHE) misses.
ICACHE.IFETCH_STALL	
EventSel=80H, UMask=04H	Cycles where a code fetch is stalled due to L1 instruction-cache miss.
ICACHE.IFDATA_STALL	
EventSel=80H, UMask=04H	Cycles where a code fetch is stalled due to L1 instruction-cache miss.
ITLB_MISSES.MISS_CAUSES_A_WALK	
EventSel=85H, UMask=01H	Misses in ITLB that causes a page walk of any page size.
ITLB_MISSES.WALK_COMPLETED_4K	
EventSel=85H, UMask=02H	Completed page walks due to misses in ITLB 4K page entries.
ITLB_MISSES.WALK_COMPLETED_2M_4M	
EventSel=85H, UMask=04H	Completed page walks due to misses in ITLB 2M/4M page entries.
ITLB_MISSES.WALK_COMPLETED_1G	
EventSel=85H, UMask=08H	Store miss in all TLB levels causes a page walk that completes. (1G).

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
ITLB_MISSES.WALK_COMPLETED	
EventSel=85H, UMask=0EH	Completed page walks in ITLB of any page size.
ITLB_MISSES.WALK_DURATION	
EventSel=85H, UMask=10H	This event counts cycles when the page miss handler (PMH) is servicing page walks caused by ITLB misses.
ITLB_MISSES.STLB_HIT_4K	
EventSel=85H, UMask=20H	ITLB misses that hit STLB (4K).
ITLB_MISSES.STLB_HIT_2M	
EventSel=85H, UMask=40H	ITLB misses that hit STLB (2M).
ITLB_MISSES.STLB_HIT	
EventSel=85H, UMask=60H	ITLB misses that hit STLB. No page walk.
ILD_STALL.LCP	
EventSel=87H, UMask=01H	This event counts cycles where the decoder is stalled on an instruction with a length changing prefix (LCP).
ILD_STALL.IQ_FULL	
EventSel=87H, UMask=04H	Stall cycles due to IQ is full.
BR_INST_EXEC.NONTAKEN_CONDITIONAL	
EventSel=88H, UMask=41H	Not taken macro-conditional branches.
BR_INST_EXEC.TAKEN_CONDITIONAL	
EventSel=88H, UMask=81H	Taken speculative and retired macro-conditional branches.
BR_INST_EXEC.TAKEN_DIRECT_JUMP	
EventSel=88H, UMask=82H	Taken speculative and retired macro-conditional branch instructions excluding calls and indirects.
BR_INST_EXEC.TAKEN_INDIRECT_JUMP_NON_CALL_RET	
EventSel=88H, UMask=84H	Taken speculative and retired indirect branches excluding calls and returns.
BR_INST_EXEC.TAKEN_INDIRECT_NEAR_RETURN	
EventSel=88H, UMask=88H	Taken speculative and retired indirect branches with return mnemonic.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
BR_INST_EXEC.TAKEN_DIRECT_NEAR_CALL	
EventSel=88H, UMask=90H	Taken speculative and retired direct near calls.
BR_INST_EXEC.TAKEN_INDIRECT_NEAR_CALL	
EventSel=88H, UMask=A0H	Taken speculative and retired indirect calls.
BR_INST_EXEC.ALL_CONDITIONAL	
EventSel=88H, UMask=C1H	Speculative and retired macro-conditional branches.
BR_INST_EXEC.ALL_DIRECT_JMP	
EventSel=88H, UMask=C2H	Speculative and retired macro-unconditional branches excluding calls and indirects.
BR_INST_EXEC.ALL_INDIRECT_JUMP_NON_CALL_RET	
EventSel=88H, UMask=C4H	Speculative and retired indirect branches excluding calls and returns.
BR_INST_EXEC.ALL_INDIRECT_NEAR_RETURN	
EventSel=88H, UMask=C8H	Speculative and retired indirect return branches.
BR_INST_EXEC.ALL_DIRECT_NEAR_CALL	
EventSel=88H, UMask=D0H	Speculative and retired direct near calls.
BR_INST_EXEC.ALL_BRANCHES	
EventSel=88H, UMask=FFH	Counts all near executed branches (not necessarily retired).
BR_MISP_EXEC.NONTAKEN_CONDITIONAL	
EventSel=89H, UMask=41H	Not taken speculative and retired mispredicted macro conditional branches.
BR_MISP_EXEC.TAKEN_CONDITIONAL	
EventSel=89H, UMask=81H	Taken speculative and retired mispredicted macro conditional branches.
BR_MISP_EXEC.TAKEN_INDIRECT_JUMP_NON_CALL_RET	
EventSel=89H, UMask=84H	Taken speculative and retired mispredicted indirect branches excluding calls and returns.
BR_MISP_EXEC.TAKEN_RETURN_NEAR	
EventSel=89H, UMask=88H	Taken speculative and retired mispredicted indirect branches with return mnemonic.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
BR_MISP_EXEC.TAKEN_INDIRECT_NEAR_CALL	
EventSel=89H, UMask=A0H	Taken speculative and retired mispredicted indirect calls.
BR_MISP_EXEC.ALL_CONDITIONAL	
EventSel=89H, UMask=C1H	Speculative and retired mispredicted macro conditional branches.
BR_MISP_EXEC.ALL_INDIRECT_JUMP_NON_CALL_RET	
EventSel=89H, UMask=C4H	Mispredicted indirect branches excluding calls and returns.
BR_MISP_EXEC.ALL_BRANCHES	
EventSel=89H, UMask=FFH	Counts all near executed branches (not necessarily retired).
IDQ_UOPS_NOT_DELIVERED.CORE	
EventSel=9CH, UMask=01H	This event count the number of undelivered (unallocated) uops from the Front-end to the Resource Allocation Table (RAT) while the Back-end of the processor is not stalled. The Front-end can allocate up to 4 uops per cycle so this event can increment 0-4 times per cycle depending on the number of unallocated uops. This event is counted on a per-core basis.
IDQ_UOPS_NOT_DELIVERED.CYCLES_0_UOPS_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=4	This event counts the number cycles during which the Front-end allocated exactly zero uops to the Resource Allocation Table (RAT) while the Back-end of the processor is not stalled. This event is counted on a per-core basis.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_1_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=3	Cycles per thread when 3 or more uops are not delivered to Resource Allocation Table (RAT) when backend of the machine is not stalled.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_2_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=2	Cycles with less than 2 uops delivered by the front end.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_3_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=1	Cycles with less than 3 uops delivered by the front end.
IDQ_UOPS_NOT_DELIVERED.CYCLES_FE_WAS_OK	
EventSel=9CH, UMask=01H, Invert=1, CMask=1	Counts cycles FE delivered 4 uops or Resource Allocation Table (RAT) was stalling FE.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
UOPS_EXECUTED_PORT.PORT_0	
EventSel=A1H, UMask=01H	Cycles which a uop is dispatched on port 0 in this thread.
UOPS_EXECUTED_PORT.PORT_0_CORE	
EventSel=A1H, UMask=01H, AnyThread=1	Cycles per core when uops are exectuted in port 0.
UOPS_DISPATCHED_PORT.PORT_0	
EventSel=A1H, UMask=01H	Cycles per thread when uops are executed in port 0.
UOPS_EXECUTED_PORT.PORT_1	
EventSel=A1H, UMask=02H	Cycles which a uop is dispatched on port 1 in this thread.
UOPS_EXECUTED_PORT.PORT_1_CORE	
EventSel=A1H, UMask=02H, AnyThread=1	Cycles per core when uops are exectuted in port 1.
UOPS_DISPATCHED_PORT.PORT_1	
EventSel=A1H, UMask=02H	Cycles per thread when uops are executed in port 1.
UOPS_EXECUTED_PORT.PORT_2	
EventSel=A1H, UMask=04H	Cycles which a uop is dispatched on port 2 in this thread.
UOPS_EXECUTED_PORT.PORT_2_CORE	
EventSel=A1H, UMask=04H, AnyThread=1	Cycles per core when uops are dispatched to port 2.
UOPS_DISPATCHED_PORT.PORT_2	
EventSel=A1H, UMask=04H	Cycles per thread when uops are executed in port 2.
UOPS_EXECUTED_PORT.PORT_3	
EventSel=A1H, UMask=08H	Cycles which a uop is dispatched on port 3 in this thread.
UOPS_EXECUTED_PORT.PORT_3_CORE	
EventSel=A1H, UMask=08H, AnyThread=1	Cycles per core when uops are dispatched to port 3.
UOPS_DISPATCHED_PORT.PORT_3	
EventSel=A1H, UMask=08H	Cycles per thread when uops are executed in port 3.
UOPS_EXECUTED_PORT.PORT_4	
EventSel=A1H, UMask=10H	Cycles which a uop is dispatched on port 4 in this thread.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
UOPS_EXECUTED_PORT.PORT_4_CORE	
EventSel=A1H, UMask=10H, AnyThread=1	Cycles per core when uops are exectuted in port 4.
UOPS_DISPATCHED_PORT.PORT_4	
EventSel=A1H, UMask=10H	Cycles per thread when uops are executed in port 4.
UOPS_EXECUTED_PORT.PORT_5	
EventSel=A1H, UMask=20H	Cycles which a uop is dispatched on port 5 in this thread.
UOPS_EXECUTED_PORT.PORT_5_CORE	
EventSel=A1H, UMask=20H, AnyThread=1	Cycles per core when uops are exectuted in port 5.
UOPS_DISPATCHED_PORT.PORT_5	
EventSel=A1H, UMask=20H	Cycles per thread when uops are executed in port 5.
UOPS_EXECUTED_PORT.PORT_6	
EventSel=A1H, UMask=40H	Cycles which a uop is dispatched on port 6 in this thread.
UOPS_EXECUTED_PORT.PORT_6_CORE	
EventSel=A1H, UMask=40H, AnyThread=1	Cycles per core when uops are exectuted in port 6.
UOPS_DISPATCHED_PORT.PORT_6	
EventSel=A1H, UMask=40H	Cycles per thread when uops are executed in port 6.
UOPS_EXECUTED_PORT.PORT_7	
EventSel=A1H, UMask=80H	Cycles which a uop is dispatched on port 7 in this thread.
UOPS_EXECUTED_PORT.PORT_7_CORE	
EventSel=A1H, UMask=80H, AnyThread=1	Cycles per core when uops are dispatched to port 7.
UOPS_DISPATCHED_PORT.PORT_7	
EventSel=A1H, UMask=80H	Cycles per thread when uops are executed in port 7.
RESOURCE_STALLS.ANY	
EventSel=A2H, UMask=01H	Cycles allocation is stalled due to resource related reason.
RESOURCE_STALLS.RS	
EventSel=A2H, UMask=04H	Cycles stalled due to no eligible RS entry available.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
RESOURCE_STALLS.SB	
EventSel=A2H, UMask=08H	This event counts cycles during which no instructions were allocated because no Store Buffers (SB) were available.
RESOURCE_STALLS.ROB	
EventSel=A2H, UMask=10H	Cycles stalled due to re-order buffer full.
CYCLE_ACTIVITY.CYCLES_L2_PENDING	
EventSel=A3H, UMask=01H, CMask=1	Cycles with pending L2 miss loads. Set Cmask=2 to count cycle.
CYCLE_ACTIVITY.CYCLES_LDM_PENDING	
EventSel=A3H, UMask=02H, CMask=2	Cycles with pending memory loads. Set Cmask=2 to count cycle.
CYCLE_ACTIVITY.CYCLES_NO_EXECUTE	
EventSel=A3H, UMask=04H, CMask=4	This event counts cycles during which no instructions were executed in the execution stage of the pipeline.
CYCLE_ACTIVITY.STALLS_L2_PENDING	
EventSel=A3H, UMask=05H, CMask=5	Number of loads missed L2.
CYCLE_ACTIVITY.STALLS_LDM_PENDING	
EventSel=A3H, UMask=06H, CMask=6	This event counts cycles during which no instructions were executed in the execution stage of the pipeline and there were memory instructions pending (waiting for data).
CYCLE_ACTIVITY.CYCLES_L1D_PENDING	
EventSel=A3H, UMask=08H, CMask=8	Cycles with pending L1 data cache miss loads. Set Cmask=8 to count cycle.
CYCLE_ACTIVITY.STALLS_L1D_PENDING	
EventSel=A3H, UMask=0CH, CMask=12	Execution stalls due to L1 data cache miss loads. Set Cmask=0CH.
LSD.UOPS	
EventSel=A8H, UMask=01H	Number of uops delivered by the LSD.
LSD.CYCLES_ACTIVE	
EventSel=A8H, UMask=01H, CMask=1	Cycles Uops delivered by the LSD, but didn't come from the decoder.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
LSD.CYCLES_4_UOPS	
EventSel=A8H, UMask=01H, CMask=4	Cycles 4 Uops delivered by the LSD, but didn't come from the decoder.
DSB2MITE_SWITCHES.PENALTY_CYCLES	
EventSel=ABH, UMask=02H	Decode Stream Buffer (DSB)-to-MITE switch true penalty cycles.
ITLB.ITLB_FLUSH	
EventSel=AEH, UMask=01H	Counts the number of ITLB flushes, includes 4k/2M/4M pages.
OFFCORE_REQUESTS.DEMAND_DATA_RD	
EventSel=B0H, UMask=01H	Demand data read requests sent to uncore.
OFFCORE_REQUESTS.DEMAND_CODE_RD	
EventSel=B0H, UMask=02H	Demand code read requests sent to uncore.
OFFCORE_REQUESTS.DEMAND_RFO	
EventSel=B0H, UMask=04H	Demand RFO read requests sent to uncore, including regular RFOs, locks, ItoM.
OFFCORE_REQUESTS.ALL_DATA_RD	
EventSel=B0H, UMask=08H	Data read requests sent to uncore (demand and prefetch).
UOPS_EXECUTED.STALL_CYCLES	
EventSel=B1H, UMask=01H, Invert=1, CMask=1	Counts number of cycles no uops were dispatched to be executed on this thread.
UOPS_EXECUTED.CYCLES_GE_1_UOP_EXEC	
EventSel=B1H, UMask=01H, CMask=1	This events counts the cycles where at least one uop was executed. It is counted per thread.
UOPS_EXECUTED.CYCLES_GE_2_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=2	This events counts the cycles where at least two uop were executed. It is counted per thread.
UOPS_EXECUTED.CYCLES_GE_3_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=3	This events counts the cycles where at least three uop were executed. It is counted per thread.
UOPS_EXECUTED.CYCLES_GE_4_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=4	Cycles where at least 4 uops were executed per-thread.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
UOPS_EXECUTED.CORE	
EventSel=B1H, UMask=02H	Counts total number of uops to be executed per-core each cycle.
UOPS_EXECUTED.CORE_CYCLES_GE_1	
EventSel=B1H, UMask=02H, CMask=1	Cycles at least 1 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_2	
EventSel=B1H, UMask=02H, CMask=2	Cycles at least 2 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_3	
EventSel=B1H, UMask=02H, CMask=3	Cycles at least 3 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_4	
EventSel=B1H, UMask=02H, CMask=4	Cycles at least 4 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_NONE	
EventSel=B1H, UMask=02H, Invert=1	Cycles with no micro-ops executed from any thread on physical core.
OFFCORE_REQUESTS_BUFFER.SQ_FULL	
EventSel=B2H, UMask=01H	Offcore requests buffer cannot take more entries for this thread core.
PAGE_WALKER_LOADS.DTLB_L1	
EventSel=BCH, UMask=11H	Number of DTLB page walker loads that hit in the L1+FB.
PAGE_WALKER_LOADS.DTLB_L2	
EventSel=BCH, UMask=12H	Number of DTLB page walker loads that hit in the L2.
PAGE_WALKER_LOADS.DTLB_L3	
EventSel=BCH, UMask=14H	Number of DTLB page walker loads that hit in the L3.
PAGE_WALKER_LOADS.DTLB_MEMORY	
EventSel=BCH, UMask=18H	Number of DTLB page walker loads from memory.
PAGE_WALKER_LOADS.ITLB_L1	
EventSel=BCH, UMask=21H	Number of ITLB page walker loads that hit in the L1+FB.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
PAGE_WALKER_LOADS.ITLB_L2	
EventSel=BCH, UMask=22H	Number of ITLB page walker loads that hit in the L2.
PAGE_WALKER_LOADS.ITLB_L3	
EventSel=BCH, UMask=24H	Number of ITLB page walker loads that hit in the L3.
PAGE_WALKER_LOADS.ITLB_MEMORY	
EventSel=BCH, UMask=28H	Number of ITLB page walker loads from memory.
PAGE_WALKER_LOADS.EPT_DTLB_L1	
EventSel=BCH, UMask=41H	Counts the number of Extended Page Table walks from the DTLB that hit in the L1 and FB.
PAGE_WALKER_LOADS.EPT_DTLB_L2	
EventSel=BCH, UMask=42H	Counts the number of Extended Page Table walks from the DTLB that hit in the L2.
PAGE_WALKER_LOADS.EPT_DTLB_L3	
EventSel=BCH, UMask=44H	Counts the number of Extended Page Table walks from the DTLB that hit in the L3.
PAGE_WALKER_LOADS.EPT_DTLB_MEMORY	
EventSel=BCH, UMask=48H	Counts the number of Extended Page Table walks from the DTLB that hit in memory.
PAGE_WALKER_LOADS.EPT_ITLB_L1	
EventSel=BCH, UMask=81H	Counts the number of Extended Page Table walks from the ITLB that hit in the L1 and FB.
PAGE_WALKER_LOADS.EPT_ITLB_L2	
EventSel=BCH, UMask=82H	Counts the number of Extended Page Table walks from the ITLB that hit in the L2.
PAGE_WALKER_LOADS.EPT_ITLB_L3	
EventSel=BCH, UMask=84H	Counts the number of Extended Page Table walks from the ITLB that hit in the L2.
PAGE_WALKER_LOADS.EPT_ITLB_MEMORY	
EventSel=BCH, UMask=88H	Counts the number of Extended Page Table walks from the ITLB that hit in memory.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
TLB_FLUSH.DTLB_THREAD	
EventSel=BDH, UMask=01H	DTLB flush attempts of the thread-specific entries.
TLB_FLUSH.STLB_ANY	
EventSel=BDH, UMask=20H	Count number of STLB flush attempts.
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural	Number of instructions at retirement.
INST_RETIRED.PREC_DIST	
EventSel=C0H, UMask=01H, Precise	Precise instruction retired event with HW to reduce effect of PEBS shadow in IP distribution.
INST_RETIRED.X87	
EventSel=C0H, UMask=02H	This is a non-precise version (that is, does not use PEBS) of the event that counts FP operations retired. For X87 FP operations that have no exceptions counting also includes flows that have several X87, or flows that use X87 uops in the exception handling.
OTHER_ASSISTS.AVX_TO_SSE	
EventSel=C1H, UMask=08H	Number of transitions from AVX-256 to legacy SSE when penalty applicable.
OTHER_ASSISTS.SSE_TO_AVX	
EventSel=C1H, UMask=10H	Number of transitions from SSE to AVX-256 when penalty applicable.
OTHER_ASSISTS.ANY_WB_ASSIST	
EventSel=C1H, UMask=40H	Number of microcode assists invoked by HW upon uop writeback.
UOPS_RETIRED.ALL	
EventSel=C2H, UMask=01H, Precise	Counts the number of micro-ops retired. Use Cmask=1 and invert to count active cycles or stalled cycles.
UOPS_RETIRED.STALL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=1	Cycles without actually retired uops.
UOPS_RETIRED.TOTAL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=10	Cycles with less than 10 actually retired uops.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
UOPS_RETIRED.CORE_STALL_CYCLES	
EventSel=C2H, UMask=01H, AnyThread=1, Invert=1, CMask=1	Cycles without actually retired uops.
UOPS_RETIRED.RETIRE_SLOTS	
EventSel=C2H, UMask=02H, Precise	This event counts the number of retirement slots used each cycle. There are potentially 4 slots that can be used each cycle - meaning, 4 uops or 4 instructions could retire each cycle.
MACHINE_CLEARS.CYCLES	
EventSel=C3H, UMask=01H	Cycles there was a Nuke. Account for both thread-specific and All Thread Nukes.
MACHINE_CLEARS.COUNT	
EventSel=C3H, UMask=01H, EdgeDetect=1, CMask=1	Number of machine clears (nukes) of any type.
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	This event counts the number of memory ordering machine clears detected. Memory ordering machine clears can result from memory address aliasing or snoops from another hardware thread or core to data inflight in the pipeline. Machine clears can have a significant performance impact if they are happening frequently.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=04H	This event is incremented when self-modifying code (SMC) is detected, which causes a machine clear. Machine clears can have a significant performance impact if they are happening frequently.
MACHINE_CLEARS.MASKMOV	
EventSel=C3H, UMask=20H	This event counts the number of executed Intel AVX masked load operations that refer to an illegal address range with the mask bits set to 0.
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	Branch instructions at retirement.
BR_INST_RETIRED.CONDITIONAL	
EventSel=C4H, UMask=01H, Precise	Counts the number of conditional branch instructions retired.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
BR_INST_RETIRED.NEAR_CALL	
EventSel=C4H, UMask=02H, Precise	Direct and indirect near call instructions retired.
BR_INST_RETIRED.NEAR_CALL_R3	
EventSel=C4H, UMask=02H, USR=1,OS=0, Precise	Direct and indirect macro near call instructions retired (captured in ring 3).
BR_INST_RETIRED.NEAR_RETURN	
EventSel=C4H, UMask=08H, Precise	Counts the number of near return instructions retired.
BR_INST_RETIRED.NOT_TAKEN	
EventSel=C4H, UMask=10H	Counts the number of not taken branch instructions retired.
BR_INST_RETIRED.NEAR_TAKEN	
EventSel=C4H, UMask=20H, Precise	Number of near taken branches retired.
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=40H	Number of far branches retired.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	Mispredicted branch instructions at retirement.
BR_MISP_RETIRED.CONDITIONAL	
EventSel=C5H, UMask=01H, Precise	Mispredicted conditional branch instructions retired.
BR_MISP_RETIRED.NEAR_TAKEN	
EventSel=C5H, UMask=20H, Precise	Number of near branch instructions retired that were taken but mispredicted.
AVX_INSTS.ALL	
EventSel=C6H, UMask=07H	Note that a whole rep string only counts AVX_INST.ALL once.
HLE_RETIRED.START	
EventSel=C8H, UMask=01H	Number of times an HLE execution started.
HLE_RETIRED.COMMIT	
EventSel=C8H, UMask=02H	Number of times an HLE execution successfully committed.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
HLE_RETIRED.ABORTED	
EventSel=C8H, UMask=04H, Precise	Number of times an HLE execution aborted due to any reasons (multiple categories may count as one).
HLE_RETIRED.ABORTED_MISC1	
EventSel=C8H, UMask=08H	Number of times an HLE execution aborted due to various memory events (e.g., read/write capacity and conflicts).
HLE_RETIRED.ABORTED_MISC2	
EventSel=C8H, UMask=10H	Number of times an HLE execution aborted due to uncommon conditions.
HLE_RETIRED.ABORTED_MISC3	
EventSel=C8H, UMask=20H	Number of times an HLE execution aborted due to HLE-unfriendly instructions.
HLE_RETIRED.ABORTED_MISC4	
EventSel=C8H, UMask=40H	Number of times an HLE execution aborted due to incompatible memory type.
HLE_RETIRED.ABORTED_MISC5	
EventSel=C8H, UMask=80H	Number of times an HLE execution aborted due to none of the previous 4 categories (e.g. interrupts).
RTM_RETIRED.START	
EventSel=C9H, UMask=01H	Number of times an RTM execution started.
RTM_RETIRED.COMMIT	
EventSel=C9H, UMask=02H	Number of times an RTM execution successfully committed.
RTM_RETIRED.ABORTED	
EventSel=C9H, UMask=04H, Precise	Number of times an RTM execution aborted due to any reasons (multiple categories may count as one).
RTM_RETIRED.ABORTED_MISC1	
EventSel=C9H, UMask=08H	Number of times an RTM execution aborted due to various memory events (e.g. read/write capacity and conflicts).
RTM_RETIRED.ABORTED_MISC2	
EventSel=C9H, UMask=10H	Number of times an RTM execution aborted due to various memory events (e.g., read/write capacity and conflicts).

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
RTM_RETIRED.ABORTED_MISC3	
EventSel=C9H, UMask=20H	Number of times an RTM execution aborted due to HLE-unfriendly instructions.
RTM_RETIRED.ABORTED_MISC4	
EventSel=C9H, UMask=40H	Number of times an RTM execution aborted due to incompatible memory type.
RTM_RETIRED.ABORTED_MISC5	
EventSel=C9H, UMask=80H	Number of times an RTM execution aborted due to none of the previous 4 categories (e.g. interrupt).
FP_ASSIST.X87_OUTPUT	
EventSel=CAH, UMask=02H	Number of X87 FP assists due to output values.
FP_ASSIST.X87_INPUT	
EventSel=CAH, UMask=04H	Number of X87 FP assists due to input values.
FP_ASSIST.SIMD_OUTPUT	
EventSel=CAH, UMask=08H	Number of SIMD FP assists due to output values.
FP_ASSIST.SIMD_INPUT	
EventSel=CAH, UMask=10H	Number of SIMD FP assists due to input values.
FP_ASSIST.ANY	
EventSel=CAH, UMask=1EH, CMask=1	Cycles with any input/output SSE* or FP assists.
ROB_MISC_EVENTS.LBR_INSERTS	
EventSel=CCH, UMask=20H	Count cases of saving new LBR records by hardware.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_4	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x4 , Precise	Loads with latency value being above 4.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_8	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x8 , Precise	Loads with latency value being above 8.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_16	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x10 , Precise	Loads with latency value being above 16.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_32	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x20 , Precise	Loads with latency value being above 32.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_64	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x40 , Precise	Loads with latency value being above 64.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_128	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x80 , Precise	Loads with latency value being above 128.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_256	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x100 , Precise	Loads with latency value being above 256.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_512	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x200 , Precise	Loads with latency value being above 512.
MEM_UOPS_RETIRED.STLB_MISS_LOADS	
EventSel=D0H, UMask=11H, Precise	Retired load uops that miss the STLB.
MEM_UOPS_RETIRED.STLB_MISS_STORES	
EventSel=D0H, UMask=12H, Precise	Retired store uops that miss the STLB.
MEM_UOPS_RETIRED.LOCK_LOADS	
EventSel=D0H, UMask=21H, Precise	Retired load uops with locked access.
MEM_UOPS_RETIRED.SPLIT_LOADS	
EventSel=D0H, UMask=41H, Precise	Retired load uops that split across a cacheline boundary.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
MEM_UOPS_RETIRED.SPLIT_STORES	
EventSel=D0H, UMask=42H, Precise	Retired store uops that split across a cacheline boundary.
MEM_UOPS_RETIRED.ALL_LOADS	
EventSel=D0H, UMask=81H, Precise	All retired load uops.
MEM_UOPS_RETIRED.ALL_STORES	
EventSel=D0H, UMask=82H, Precise	All retired store uops.
MEM_LOAD_UOPS_RETIRED.L1_HIT	
EventSel=D1H, UMask=01H, Precise	Retired load uops with L1 cache hits as data sources.
MEM_LOAD_UOPS_RETIRED.L2_HIT	
EventSel=D1H, UMask=02H, Precise	Retired load uops with L2 cache hits as data sources.
MEM_LOAD_UOPS_RETIRED.L3_HIT	
EventSel=D1H, UMask=04H, Precise	Retired load uops with L3 cache hits as data sources.
MEM_LOAD_UOPS_RETIRED.L1_MISS	
EventSel=D1H, UMask=08H, Precise	Retired load uops missed L1 cache as data sources.
MEM_LOAD_UOPS_RETIRED.L2_MISS	
EventSel=D1H, UMask=10H, Precise	Retired load uops missed L2. Unknown data source excluded.
MEM_LOAD_UOPS_RETIRED.L3_MISS	
EventSel=D1H, UMask=20H, Precise	Retired load uops missed L3. Excludes unknown data source .
MEM_LOAD_UOPS_RETIRED.HIT_LFB	
EventSel=D1H, UMask=40H, Precise	Retired load uops which data sources were load uops missed L1 but hit FB due to preceding miss to the same cache line with data not ready.
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_MISS	
EventSel=D2H, UMask=01H, Precise	Retired load uops which data sources were L3 hit and cross-core snoop missed in on-pkg core cache.
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_HIT	
EventSel=D2H, UMask=02H, Precise	Retired load uops which data sources were L3 and cross-core snoop hits in on-pkg core cache.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_HITM	
EventSel=D2H, UMask=04H, Precise	Retired load uops which data sources were HitM responses from shared L3.
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_NONE	
EventSel=D2H, UMask=08H, Precise	Retired load uops which data sources were hits in L3 without snoops required.
MEM_LOAD_UOPS_L3_MISS_RETIRED.LOCAL_DRAM	
EventSel=D3H, UMask=01H, Precise	This event counts retired load uops where the data came from local DRAM. This does not include hardware prefetches.
BACLEARS.ANY	
EventSel=E6H, UMask=1FH	Number of front end re-steers due to BPU misprediction.
L2_TRANS.DEMAND_DATA_RD	
EventSel=F0H, UMask=01H	Demand data read requests that access L2 cache.
L2_TRANS.RFO	
EventSel=F0H, UMask=02H	RFO requests that access L2 cache.
L2_TRANS.CODE_RD	
EventSel=F0H, UMask=04H	L2 cache accesses when fetching instructions.
L2_TRANS.ALL_PF	
EventSel=F0H, UMask=08H	Any MLC or L3 HW prefetch accessing L2, including rejects.
L2_TRANS.L1D_WB	
EventSel=F0H, UMask=10H	L1D writebacks that access L2 cache.
L2_TRANS.L2_FILL	
EventSel=F0H, UMask=20H	L2 fill requests that access L2 cache.
L2_TRANS.L2_WB	
EventSel=F0H, UMask=40H	L2 writebacks that access L2 cache.
L2_TRANS.ALL_REQUESTS	
EventSel=F0H, UMask=80H	Transactions accessing L2 pipe.
L2_LINES_IN.I	
EventSel=F1H, UMask=01H	L2 cache lines in I state filling L2.

Table 4: Performance Events in the Processor Core Based on the Haswell Microarchitecture Intel® Xeon® Processor E5 v3 Family (06_3CH, 06_45H and 06_46H)

Event Name	
Configuration	Description
L2_LINES_IN.S	
EventSel=F1H, UMask=02H	L2 cache lines in S state filling L2.
L2_LINES_IN.E	
EventSel=F1H, UMask=04H	L2 cache lines in E state filling L2.
L2_LINES_IN.ALL	
EventSel=F1H, UMask=07H	This event counts the number of L2 cache lines brought into the L2 cache. Lines are filled into the L2 cache when there was an L2 miss.
L2_LINES_OUT.DEMAND_CLEAN	
EventSel=F2H, UMask=05H	Clean L2 cache lines evicted by demand.
L2_LINES_OUT.DEMAND_DIRTY	
EventSel=F2H, UMask=06H	Dirty L2 cache lines evicted by demand.
SQ_MISC.SPLIT_LOCK	
EventSel=F4H, UMask=10H	Split locks in SQ.

Performance Monitoring Events based on Haswell-E Microarchitecture- Intel Xeon Processor E5 v3 Family

Performance monitoring events in the processor core of the Intel Xeon processor E5 v3 family based on the Haswell-E Microarchitecture are listed in the table below.

Table 5: Performance Events in the Processor Core of Intel® Xeon® Processor E5 v3 Family (06_3FH)

Event Name	
Configuration	Description
MEM_LOAD_UOPS_L3_MISS_RETIRED.REMOTE_DRAM	
EventSel=D3H, UMask=04H	Retired load uop whose Data Source was: remote DRAM either Snoop not needed or Snoop Miss (Rspl).
MEM_LOAD_UOPS_L3_MISS_RETIRED.REMOTE_HITM	
EventSel=D3H, UMask=10H	Retired load uop whose Data Source was: Remote cache HITM.
MEM_LOAD_UOPS_L3_MISS_RETIRED.REMOTE_FWD	
EventSel=D3H, UMask=20H	Retired load uop whose Data Source was: forwarded from remote cache.

Performance Monitoring Events based on Ivy Bridge Microarchitecture - 3rd Generation Intel® Core™ Processors

3rd generation Intel® Core™ processors and Intel Xeon processor E3-1200 v2 product family are based on Intel Microarchitecture code name Ivy Bridge. Performance-monitoring events in the processor core are listed in the table below.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed	Instructions retired from execution.
CPU_CLK_UNHALTED.THREAD	
Architectural, Fixed	Core cycles when the thread is not in halt state.
CPU_CLK_UNHALTED.THREAD_ANY	
AnyThread=1, Architectural, Fixed	Core cycles when at least one thread on the physical core is not in halt state.
CPU_CLK_UNHALTED.REF_TSC	
Architectural, Fixed	Reference cycles when the core is not in halt state.
LD_BLOCKS.STORE_FORWARD	
EventSel=03H, UMask=02H	Loads blocked by overlapping with store buffer that cannot be forwarded.
LD_BLOCKS.NO_SR	
EventSel=03H, UMask=08H	The number of times that split load operations are temporarily blocked because all resources for handling the split accesses are in use.
MISALIGN_MEM_REF.LOADS	
EventSel=05H, UMask=01H	Speculative cache-line split load uops dispatched to L1D.
MISALIGN_MEM_REF.STORES	
EventSel=05H, UMask=02H	Speculative cache-line split Store-address uops dispatched to L1D.
LD_BLOCKS_PARTIAL.ADDRESS_ALIAS	
EventSel=07H, UMask=01H	False dependencies in MOB due to partial compare on address.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
DTLB_LOAD_MISSES.MISS_CAUSES_A_WALK	
EventSel=08H, UMask=81H	Misses in all TLB levels that cause a page walk of any page size from demand loads.
DTLB_LOAD_MISSES.WALK_COMPLETED	
EventSel=08H, UMask=82H	Misses in all TLB levels that caused page walk completed of any size by demand loads.
DTLB_LOAD_MISSES.WALK_DURATION	
EventSel=08H, UMask=84H	Cycle PMH is busy with a walk due to demand loads.
DTLB_LOAD_MISSES.LARGE_PAGE_WALK_COMPLETED	
EventSel=08H, UMask=88H	Page walk for a large page completed for Demand load.
INT_MISC.RECOVERY_CYCLES	
EventSel=0DH, UMask=03H, CMask=1	Number of cycles waiting for the checkpoints in Resource Allocation Table (RAT) to be recovered after Nuke due to all other cases except JEClear (e.g. whenever a ucode assist is needed like SSE exception, memory disambiguation, etc.).
INT_MISC.RECOVERY_STALLS_COUNT	
EventSel=0DH, UMask=03H, EdgeDetect=1, CMask=1	Number of occurences waiting for the checkpoints in Resource Allocation Table (RAT) to be recovered after Nuke due to all other cases except JEClear (e.g. whenever a ucode assist is needed like SSE exception, memory disambiguation, etc.).
INT_MISC.RECOVERY_CYCLES_ANY	
EventSel=0DH, UMask=03H, AnyThread=1, CMask=1	Core cycles the allocator was stalled due to recovery from earlier clear event for any thread running on the physical core (e.g. misprediction or memory nuke).
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=01H	Increments each cycle the # of Uops issued by the RAT to RS. Set Cmask = 1, Inv = 1, Any= 1to count stalled cycles of this core.
UOPS_ISSUED.STALL_CYCLES	
EventSel=0EH, UMask=01H, Invert=1, CMask=1	Cycles when Resource Allocation Table (RAT) does not issue Uops to Reservation Station (RS) for the thread.
UOPS_ISSUED.CORE_STALL_CYCLES	
EventSel=0EH, UMask=01H, AnyThread=1, Invert=1, CMask=1	Cycles when Resource Allocation Table (RAT) does not issue Uops to Reservation Station (RS) for all threads.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
UOPS_ISSUED.FLAGS_MERGE	
EventSel=0EH, UMask=10H	Number of flags-merge uops allocated. Such uops adds delay.
UOPS_ISSUED.SLOW_LEA	
EventSel=0EH, UMask=20H	Number of slow LEA or similar uops allocated. Such uop has 3 sources (e.g. 2 sources + immediate) regardless if as a result of LEA instruction or not.
UOPS_ISSUED.SINGLE_MUL	
EventSel=0EH, UMask=40H	Number of multiply packed/scalar single precision uops allocated.
FP_COMP_OPS_EXE.X87	
EventSel=10H, UMask=01H	Counts number of X87 uops executed.
FP_COMP_OPS_EXE.SSE_PACKED_DOUBLE	
EventSel=10H, UMask=10H	Number of SSE* or AVX-128 FP Computational packed double-precision uops issued this cycle.
FP_COMP_OPS_EXE.SSE_SCALAR_SINGLE	
EventSel=10H, UMask=20H	Number of SSE* or AVX-128 FP Computational scalar single-precision uops issued this cycle.
FP_COMP_OPS_EXE.SSE_PACKED_SINGLE	
EventSel=10H, UMask=40H	Number of SSE* or AVX-128 FP Computational packed single-precision uops issued this cycle.
FP_COMP_OPS_EXE.SSE_SCALAR_DOUBLE	
EventSel=10H, UMask=80H	Counts number of SSE* or AVX-128 double precision FP scalar uops executed.
SIMD_FP_256.PACKED_SINGLE	
EventSel=11H, UMask=01H	Counts 256-bit packed single-precision floating-point instructions.
SIMD_FP_256.PACKED_DOUBLE	
EventSel=11H, UMask=02H	Counts 256-bit packed double-precision floating-point instructions.
ARITH.FPU_DIV_ACTIVE	
EventSel=14H, UMask=01H	Cycles that the divider is active, includes INT and FP. Set 'edge =1, cmask=1' to count the number of divides.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
ARITH.FPU_DIV	
EventSel=14H, UMask=04H, EdgeDetect=1, CMask=1	Divide operations executed.
L2_RQSTS.DEMAND_DATA_RD_HIT	
EventSel=24H, UMask=01H	Demand Data Read requests that hit L2 cache.
L2_RQSTS.ALL_DEMAND_DATA_RD	
EventSel=24H, UMask=03H	Counts any demand and L1 HW prefetch data load requests to L2.
L2_RQSTS.RFO_HIT	
EventSel=24H, UMask=04H	RFO requests that hit L2 cache.
L2_RQSTS.RFO_MISS	
EventSel=24H, UMask=08H	Counts the number of store RFO requests that miss the L2 cache.
L2_RQSTS.ALL_RFO	
EventSel=24H, UMask=0CH	Counts all L2 store RFO requests.
L2_RQSTS.CODE_RD_HIT	
EventSel=24H, UMask=10H	Number of instruction fetches that hit the L2 cache.
L2_RQSTS.CODE_RD_MISS	
EventSel=24H, UMask=20H	Number of instruction fetches that missed the L2 cache.
L2_RQSTS.ALL_CODE_RD	
EventSel=24H, UMask=30H	Counts all L2 code requests.
L2_RQSTS.PF_HIT	
EventSel=24H, UMask=40H	Counts all L2 HW prefetcher requests that hit L2.
L2_RQSTS.PF_MISS	
EventSel=24H, UMask=80H	Counts all L2 HW prefetcher requests that missed L2.
L2_RQSTS.ALL_PF	
EventSel=24H, UMask=C0H	Counts all L2 HW prefetcher requests.
L2_STORE_LOCK_RQSTS.MISS	
EventSel=27H, UMask=01H	RFOs that miss cache lines.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
L2_STORE_LOCK_RQSTS.HIT_M	
EventSel=27H, UMask=08H	RFOs that hit cache lines in M state.
L2_STORE_LOCK_RQSTS.ALL	
EventSel=27H, UMask=0FH	RFOs that access cache lines in any state.
L2_L1D_WB_RQSTS.MISS	
EventSel=28H, UMask=01H	Not rejected writebacks that missed LLC.
L2_L1D_WB_RQSTS.HIT_E	
EventSel=28H, UMask=04H	Not rejected writebacks from L1D to L2 cache lines in E state.
L2_L1D_WB_RQSTS.HIT_M	
EventSel=28H, UMask=08H	Not rejected writebacks from L1D to L2 cache lines in M state.
L2_L1D_WB_RQSTS.ALL	
EventSel=28H, UMask=0FH	Not rejected writebacks from L1D to L2 cache lines in any state.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	This event counts each cache miss condition for references to the last level cache.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	This event counts requests originating from the core that reference a cache line in the last level cache.
CPU_CLK_UNHALTED.THREAD_P	
EventSel=3CH, UMask=00H, Architectural	Counts the number of thread cycles while the thread is not in a halt state. The thread enters the halt state when it is running the HLT instruction. The core frequency may change from time to time due to power or thermal throttling.
CPU_CLK_UNHALTED.THREAD_P_ANY	
EventSel=3CH, UMask=00H, AnyThread=1, Architectural	Core cycles when at least one thread on the physical core is not in halt state.
CPU_CLK_THREAD_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	Increments at the frequency of XCLK (100 MHz) when not halted.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
CPU_CLK_THREAD_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Reference cycles when the at least one thread on the physical core is unhalted. (counts at 100 MHz rate).
CPU_CLK_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	Reference cycles when the thread is unhalted. (counts at 100 MHz rate).
CPU_CLK_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Reference cycles when the at least one thread on the physical core is unhalted. (counts at 100 MHz rate).
CPU_CLK_THREAD_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Count XClk pulses when this thread is unhalted and the other is halted.
CPU_CLK_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Count XClk pulses when this thread is unhalted and the other thread is halted.
L1D_PEND_MISS.PENDING	
EventSel=48H, UMask=01H	Increments the number of outstanding L1D misses every cycle. Set Cmask = 1 and Edge =1 to count occurrences.
L1D_PEND_MISS.PENDING_CYCLES	
EventSel=48H, UMask=01H, CMask=1	Cycles with L1D load Misses outstanding.
L1D_PEND_MISS.PENDING_CYCLES_ANY	
EventSel=48H, UMask=01H, AnyThread=1, CMask=1	Cycles with L1D load Misses outstanding from any thread on physical core.
L1D_PEND_MISS.FB_FULL	
EventSel=48H, UMask=02H, CMask=1	Cycles a demand request was blocked due to Fill Buffers inavailability.
DTLB_STORE_MISSES.MISS_CAUSES_A_WALK	
EventSel=49H, UMask=01H	Miss in all TLB levels causes a page walk of any page size (4K/2M/4M/1G).
DTLB_STORE_MISSES.WALK_COMPLETED	
EventSel=49H, UMask=02H	Miss in all TLB levels causes a page walk that completes of any page size (4K/2M/4M/1G).

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
DTLB_STORE_MISSES.WALK_DURATION	
EventSel=49H, UMask=04H	Cycles PMH is busy with this walk.
DTLB_STORE_MISSES.STLB_HIT	
EventSel=49H, UMask=10H	Store operations that miss the first TLB level but hit the second and do not cause page walks.
LOAD_HIT_PRE.SW_PF	
EventSel=4CH, UMask=01H	Non-SW-prefetch load dispatches that hit fill buffer allocated for S/W prefetch.
LOAD_HIT_PRE.HW_PF	
EventSel=4CH, UMask=02H	Non-SW-prefetch load dispatches that hit fill buffer allocated for H/W prefetch.
EPT.WALK_CYCLES	
EventSel=4FH, UMask=10H	Cycle count for an Extended Page table walk. The Extended Page Directory cache is used by Virtual Machine operating systems while the guest operating systems use the standard TLB caches.
L1D.REPLACEMENT	
EventSel=51H, UMask=01H	Counts the number of lines brought into the L1 data cache.
MOVE_ELIMINATION.INT_ELIMINATED	
EventSel=58H, UMask=01H	Number of integer Move Elimination candidate uops that were eliminated.
MOVE_ELIMINATION.SIMD_ELIMINATED	
EventSel=58H, UMask=02H	Number of SIMD Move Elimination candidate uops that were eliminated.
MOVE_ELIMINATION.INT_NOT_ELIMINATED	
EventSel=58H, UMask=04H	Number of integer Move Elimination candidate uops that were not eliminated.
MOVE_ELIMINATION.SIMD_NOT_ELIMINATED	
EventSel=58H, UMask=08H	Number of SIMD Move Elimination candidate uops that were not eliminated.
CPL_CYCLES.RING0	
EventSel=5CH, UMask=01H	Unhalted core cycles when the thread is in ring 0.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
CPL_CYCLES.RING0_TRANS	
EventSel=5CH, UMask=01H, EdgeDetect=1, CMask=1	Number of intervals between processor halts while thread is in ring 0.
CPL_CYCLES.RING123	
EventSel=5CH, UMask=02H	Unhalted core cycles when the thread is not in ring 0.
RS_EVENTS.EMPTY_CYCLES	
EventSel=5EH, UMask=01H	Cycles the RS is empty for the thread.
RS_EVENTS.EMPTY_END	
EventSel=5EH, UMask=01H, EdgeDetect=1, Invert=1, CMask=1	Counts end of periods where the Reservation Station (RS) was empty. Could be useful to precisely locate Frontend Latency Bound issues.
DTLB_LOAD_MISSES.STLB_HIT	
EventSel=5FH, UMask=04H	Counts load operations that missed 1st level DTLB but hit the 2nd level.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD	
EventSel=60H, UMask=01H	Offcore outstanding Demand Data Read transactions in SQ to uncore. Set Cmask=1 to count cycles.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_DATA_RD	
EventSel=60H, UMask=01H, CMask=1	Cycles when offcore outstanding Demand Data Read transactions are present in SuperQueue (SQ), queue to uncore.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD_GE_6	
EventSel=60H, UMask=01H, CMask=6	Cycles with at least 6 offcore outstanding Demand Data Read transactions in uncore queue.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_CODE_RD	
EventSel=60H, UMask=02H	Offcore outstanding Demand Code Read transactions in SQ to uncore. Set Cmask=1 to count cycles.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_CODE_RD	
EventSel=60H, UMask=02H, CMask=1	Offcore outstanding code reads transactions in SuperQueue (SQ), queue to uncore, every cycle.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_RFO	
EventSel=60H, UMask=04H	Offcore outstanding RFO store transactions in SQ to uncore. Set Cmask=1 to count cycles.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_RFO	
EventSel=60H, UMask=04H, CMask=1	Offcore outstanding demand rfo reads transactions in SuperQueue (SQ), queue to uncore, every cycle.
OFFCORE_REQUESTS_OUTSTANDING.ALL_DATA_RD	
EventSel=60H, UMask=08H	Offcore outstanding cacheable data read transactions in SQ to uncore. Set Cmask=1 to count cycles.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DATA_RD	
EventSel=60H, UMask=08H, CMask=1	Cycles when offcore outstanding cacheable Core Data Read transactions are present in SuperQueue (SQ), queue to uncore.
LOCK_CYCLES.SPLIT_LOCK_UC_LOCK_DURATION	
EventSel=63H, UMask=01H	Cycles in which the L1D and L2 are locked, due to a UC lock or split lock.
LOCK_CYCLES.CACHE_LOCK_DURATION	
EventSel=63H, UMask=02H	Cycles in which the L1D is locked.
IDQ.EMPTY	
EventSel=79H, UMask=02H	Counts cycles the IDQ is empty.
IDQ.MITE_UOPS	
EventSel=79H, UMask=04H	Increment each cycle # of uops delivered to IDQ from MITE path. Set Cmask = 1 to count cycles.
IDQ.MITE_CYCLES	
EventSel=79H, UMask=04H, CMask=1	Cycles when uops are being delivered to Instruction Decode Queue (IDQ) from MITE path.
IDQ.DSB_UOPS	
EventSel=79H, UMask=08H	Increment each cycle. # of uops delivered to IDQ from DSB path. Set Cmask = 1 to count cycles.
IDQ.DSB_CYCLES	
EventSel=79H, UMask=08H, CMask=1	Cycles when uops are being delivered to Instruction Decode Queue (IDQ) from Decode Stream Buffer (DSB) path.
IDQ.MS_DSB_UOPS	
EventSel=79H, UMask=10H	Increment each cycle # of uops delivered to IDQ when MS_busy by DSB. Set Cmask = 1 to count cycles. Add Edge=1 to count # of delivery.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
IDQ.MS_DSB_CYCLES	
EventSel=79H, UMask=10H, CMask=1	Cycles when uops initiated by Decode Stream Buffer (DSB) are being delivered to Instruction Decode Queue (IDQ) while Microcode Sequenser (MS) is busy.
IDQ.MS_DSB_OCCUR	
EventSel=79H, UMask=10H, EdgeDetect=1, CMask=1	Deliveries to Instruction Decode Queue (IDQ) initiated by Decode Stream Buffer (DSB) while Microcode Sequenser (MS) is busy.
IDQ.ALL_DSB_CYCLES_4_UOPS	
EventSel=79H, UMask=18H, CMask=4	Counts cycles DSB is delivered four uops. Set Cmask = 4.
IDQ.ALL_DSB_CYCLES_ANY_UOPS	
EventSel=79H, UMask=18H, CMask=1	Counts cycles DSB is delivered at least one uops. Set Cmask = 1.
IDQ.MS_MITE_UOPS	
EventSel=79H, UMask=20H	Increment each cycle # of uops delivered to IDQ when MS_busy by MITE. Set Cmask = 1 to count cycles.
IDQ.ALL_MITE_CYCLES_4_UOPS	
EventSel=79H, UMask=24H, CMask=4	Counts cycles MITE is delivered four uops. Set Cmask = 4.
IDQ.ALL_MITE_CYCLES_ANY_UOPS	
EventSel=79H, UMask=24H, CMask=1	Counts cycles MITE is delivered at least one uops. Set Cmask = 1.
IDQ.MS_UOPS	
EventSel=79H, UMask=30H	Increment each cycle # of uops delivered to IDQ from MS by either DSB or MITE. Set Cmask = 1 to count cycles.
IDQ.MS_CYCLES	
EventSel=79H, UMask=30H, CMask=1	Cycles when uops are being delivered to Instruction Decode Queue (IDQ) while Microcode Sequenser (MS) is busy.
IDQ.MS_SWITCHES	
EventSel=79H, UMask=30H, EdgeDetect=1, CMask=1	Number of switches from DSB (Decode Stream Buffer) or MITE (legacy decode pipeline) to the Microcode Sequencer.
IDQ.MITE_ALL_UOPS	
EventSel=79H, UMask=3CH	Number of uops delivered to IDQ from any path.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
ICACHE.HIT	
EventSel=80H, UMask=01H	Number of Instruction Cache, Streaming Buffer and Victim Cache Reads. both cacheable and noncacheable, including UC fetches.
ICACHE.MISSES	
EventSel=80H, UMask=02H	Number of Instruction Cache, Streaming Buffer and Victim Cache Misses. Includes UC accesses.
ICACHE.IFETCH_STALL	
EventSel=80H, UMask=04H	Cycles where a code-fetch stalled due to L1 instruction-cache miss or an iTLB miss.
ITLB_MISSES.MISS_CAUSES_A_WALK	
EventSel=85H, UMask=01H	Misses in all ITLB levels that cause page walks.
ITLB_MISSES.WALK_COMPLETED	
EventSel=85H, UMask=02H	Misses in all ITLB levels that cause completed page walks.
ITLB_MISSES.WALK_DURATION	
EventSel=85H, UMask=04H	Cycle PMH is busy with a walk.
ITLB_MISSES.STLB_HIT	
EventSel=85H, UMask=10H	Number of cache load STLB hits. No page walk.
ITLB_MISSES.LARGE_PAGE_WALK_COMPLETED	
EventSel=85H, UMask=80H	Completed page walks in ITLB due to STLB load misses for large pages.
ILD_STALL.LCP	
EventSel=87H, UMask=01H	Stalls caused by changing prefix length of the instruction.
ILD_STALL.IQ_FULL	
EventSel=87H, UMask=04H	Stall cycles due to IQ is full.
BR_INST_EXEC.NONTAKEN_CONDITIONAL	
EventSel=88H, UMask=41H	Not taken macro-conditional branches.
BR_INST_EXEC.TAKEN_CONDITIONAL	
EventSel=88H, UMask=81H	Taken speculative and retired macro-conditional branches.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
BR_INST_EXEC.TAKEN_DIRECT_JUMP	
EventSel=88H, UMask=82H	Taken speculative and retired macro-conditional branch instructions excluding calls and indirects.
BR_INST_EXEC.TAKEN_INDIRECT_JUMP_NON_CALL_RET	
EventSel=88H, UMask=84H	Taken speculative and retired indirect branches excluding calls and returns.
BR_INST_EXEC.TAKEN_INDIRECT_NEAR_RETURN	
EventSel=88H, UMask=88H	Taken speculative and retired indirect branches with return mnemonic.
BR_INST_EXEC.TAKEN_DIRECT_NEAR_CALL	
EventSel=88H, UMask=90H	Taken speculative and retired direct near calls.
BR_INST_EXEC.TAKEN_INDIRECT_NEAR_CALL	
EventSel=88H, UMask=A0H	Taken speculative and retired indirect calls.
BR_INST_EXEC.ALL_CONDITIONAL	
EventSel=88H, UMask=C1H	Speculative and retired macro-conditional branches.
BR_INST_EXEC.ALL_DIRECT_JMP	
EventSel=88H, UMask=C2H	Speculative and retired macro-unconditional branches excluding calls and indirects.
BR_INST_EXEC.ALL_INDIRECT_JUMP_NON_CALL_RET	
EventSel=88H, UMask=C4H	Speculative and retired indirect branches excluding calls and returns.
BR_INST_EXEC.ALL_INDIRECT_NEAR_RETURN	
EventSel=88H, UMask=C8H	Speculative and retired indirect return branches.
BR_INST_EXEC.ALL_DIRECT_NEAR_CALL	
EventSel=88H, UMask=D0H	Speculative and retired direct near calls.
BR_INST_EXEC.ALL_BRANCHES	
EventSel=88H, UMask=FFH	Counts all near executed branches (not necessarily retired).
BR_MISP_EXEC.NONTAKEN_CONDITIONAL	
EventSel=89H, UMask=41H	Not taken speculative and retired mispredicted macro conditional branches.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
BR_MISP_EXEC.TAKEN_CONDITIONAL	
EventSel=89H, UMask=81H	Taken speculative and retired mispredicted macro conditional branches.
BR_MISP_EXEC.TAKEN_INDIRECT_JUMP_NON_CALL_RET	
EventSel=89H, UMask=84H	Taken speculative and retired mispredicted indirect branches excluding calls and returns.
BR_MISP_EXEC.TAKEN_RETURN_NEAR	
EventSel=89H, UMask=88H	Taken speculative and retired mispredicted indirect branches with return mnemonic.
BR_MISP_EXEC.TAKEN_INDIRECT_NEAR_CALL	
EventSel=89H, UMask=A0H	Taken speculative and retired mispredicted indirect calls.
BR_MISP_EXEC.ALL_CONDITIONAL	
EventSel=89H, UMask=C1H	Speculative and retired mispredicted macro conditional branches.
BR_MISP_EXEC.ALL_INDIRECT_JUMP_NON_CALL_RET	
EventSel=89H, UMask=C4H	Mispredicted indirect branches excluding calls and returns.
BR_MISP_EXEC.ALL_BRANCHES	
EventSel=89H, UMask=FFH	Counts all near executed branches (not necessarily retired).
IDQ_UOPS_NOT_DELIVERED.CORE	
EventSel=9CH, UMask=01H	Count issue pipeline slots where no uop was delivered from the front end to the back end when there is no back-end stall.
IDQ_UOPS_NOT_DELIVERED.CYCLES_0_UOPS_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=4	Cycles per thread when 4 or more uops are not delivered to Resource Allocation Table (RAT) when backend of the machine is not stalled.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_1_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=3	Cycles per thread when 3 or more uops are not delivered to Resource Allocation Table (RAT) when backend of the machine is not stalled.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_2_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=2	Cycles with less than 2 uops delivered by the front end.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_3_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=1	Cycles with less than 3 uops delivered by the front end.
IDQ_UOPS_NOT_DELIVERED.CYCLES_FE_WAS_OK	
EventSel=9CH, UMask=01H, Invert=1, CMask=1	Counts cycles FE delivered 4 uops or Resource Allocation Table (RAT) was stalling FE.
UOPS_DISPATCHED_PORT.PORT_0	
EventSel=A1H, UMask=01H	Cycles which a Uop is dispatched on port 0.
UOPS_DISPATCHED_PORT.PORT_0_CORE	
EventSel=A1H, UMask=01H, AnyThread=1	Cycles per core when uops are dispatched to port 0.
UOPS_DISPATCHED_PORT.PORT_1	
EventSel=A1H, UMask=02H	Cycles which a Uop is dispatched on port 1.
UOPS_DISPATCHED_PORT.PORT_1_CORE	
EventSel=A1H, UMask=02H, AnyThread=1	Cycles per core when uops are dispatched to port 1.
UOPS_DISPATCHED_PORT.PORT_2	
EventSel=A1H, UMask=0CH	Cycles which a Uop is dispatched on port 2.
UOPS_DISPATCHED_PORT.PORT_2_CORE	
EventSel=A1H, UMask=0CH, AnyThread=1	Uops dispatched to port 2, loads and stores per core (speculative and retired).
UOPS_DISPATCHED_PORT.PORT_3	
EventSel=A1H, UMask=30H	Cycles which a Uop is dispatched on port 3.
UOPS_DISPATCHED_PORT.PORT_3_CORE	
EventSel=A1H, UMask=30H, AnyThread=1	Cycles per core when load or STA uops are dispatched to port 3.
UOPS_DISPATCHED_PORT.PORT_4	
EventSel=A1H, UMask=40H	Cycles which a Uop is dispatched on port 4.
UOPS_DISPATCHED_PORT.PORT_4_CORE	
EventSel=A1H, UMask=40H, AnyThread=1	Cycles per core when uops are dispatched to port 4.
UOPS_DISPATCHED_PORT.PORT_5	
EventSel=A1H, UMask=80H	Cycles which a Uop is dispatched on port 5.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
UOPS_DISPATCHED_PORT.PORT_5_CORE	
EventSel=A1H, UMask=80H, AnyThread=1	Cycles per core when uops are dispatched to port 5.
RESOURCE_STALLS.ANY	
EventSel=A2H, UMask=01H	Cycles Allocation is stalled due to Resource Related reason.
RESOURCE_STALLS.RS	
EventSel=A2H, UMask=04H	Cycles stalled due to no eligible RS entry available.
RESOURCE_STALLS.SB	
EventSel=A2H, UMask=08H	Cycles stalled due to no store buffers available (not including draining form sync).
RESOURCE_STALLS.ROB	
EventSel=A2H, UMask=10H	Cycles stalled due to re-order buffer full.
CYCLE_ACTIVITY.CYCLES_L2_PENDING	
EventSel=A3H, UMask=01H, CMask=1	Cycles with pending L2 miss loads. Set AnyThread to count per core.
CYCLE_ACTIVITY.CYCLES_L2_MISS	
EventSel=A3H, UMask=01H, CMask=1	Cycles while L2 cache miss load* is outstanding.
CYCLE_ACTIVITY.CYCLES_LDM_PENDING	
EventSel=A3H, UMask=02H, CMask=2	Cycles with pending memory loads. Set AnyThread to count per core.
CYCLE_ACTIVITY.CYCLES_MEM_ANY	
EventSel=A3H, UMask=02H, CMask=2	Cycles while memory subsystem has an outstanding load.
CYCLE_ACTIVITY.CYCLES_NO_EXECUTE	
EventSel=A3H, UMask=04H, CMask=4	Total execution stalls.
CYCLE_ACTIVITY.STALLS_TOTAL	
EventSel=A3H, UMask=04H, CMask=4	Total execution stalls.
CYCLE_ACTIVITY.STALLS_L2_PENDING	
EventSel=A3H, UMask=05H, CMask=5	Number of loads missed L2.
CYCLE_ACTIVITY.STALLS_L2_MISS	
EventSel=A3H, UMask=05H, CMask=5	Execution stalls while L2 cache miss load* is outstanding.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
CYCLE_ACTIVITY.STALLS_LDM_PENDING	
EventSel=A3H, UMask=06H, CMask=6	Execution stalls due to memory subsystem.
CYCLE_ACTIVITY.STALLS_MEM_ANY	
EventSel=A3H, UMask=06H, CMask=6	Execution stalls while memory subsystem has an outstanding load.
CYCLE_ACTIVITY.CYCLES_L1D_PENDING	
EventSel=A3H, UMask=08H, CMask=8	Cycles with pending L1 cache miss loads. Set AnyThread to count per core.
CYCLE_ACTIVITY.CYCLES_L1D_MISS	
EventSel=A3H, UMask=08H, CMask=8	Cycles while L1 cache miss demand load is outstanding.
CYCLE_ACTIVITY.STALLS_L1D_PENDING	
EventSel=A3H, UMask=0CH, CMask=12	Execution stalls due to L1 data cache miss loads. Set Cmask=0CH.
CYCLE_ACTIVITY.STALLS_L1D_MISS	
EventSel=A3H, UMask=0CH, CMask=12	Execution stalls while L1 cache miss demand load is outstanding.
LSD.UOPS	
EventSel=A8H, UMask=01H	Number of Uops delivered by the LSD.
LSD.CYCLES_ACTIVE	
EventSel=A8H, UMask=01H, CMask=1	Cycles Uops delivered by the LSD, but didn't come from the decoder.
LSD.CYCLES_4_UOPS	
EventSel=A8H, UMask=01H, CMask=4	Cycles 4 Uops delivered by the LSD, but didn't come from the decoder.
DSB2MITE_SWITCHES.COUNT	
EventSel=ABH, UMask=01H	Number of DSB to MITE switches.
DSB2MITE_SWITCHES.PENALTY_CYCLES	
EventSel=ABH, UMask=02H	Cycles DSB to MITE switches caused delay.
DSB_FILL.EXCEED_DSB_LINES	
EventSel=ACH, UMask=08H	DSB Fill encountered > 3 DSB lines.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
ITLB.ITLB_FLUSH	
EventSel=AEH, UMask=01H	Counts the number of ITLB flushes, includes 4k/2M/4M pages.
OFFCORE_REQUESTS.DEMAND_DATA_RD	
EventSel=B0H, UMask=01H	Demand data read requests sent to uncore.
OFFCORE_REQUESTS.DEMAND_CODE_RD	
EventSel=B0H, UMask=02H	Demand code read requests sent to uncore.
OFFCORE_REQUESTS.DEMAND_RFO	
EventSel=B0H, UMask=04H	Demand RFO read requests sent to uncore, including regular RFOs, locks, ItoM.
OFFCORE_REQUESTS.ALL_DATA_RD	
EventSel=B0H, UMask=08H	Data read requests sent to uncore (demand and prefetch).
UOPS_EXECUTED.THREAD	
EventSel=B1H, UMask=01H	Counts total number of uops to be executed per-thread each cycle. Set Cmask = 1, INV =1 to count stall cycles.
UOPS_EXECUTED.STALL_CYCLES	
EventSel=B1H, UMask=01H, Invert=1, CMask=1	Counts number of cycles no uops were dispatched to be executed on this thread.
UOPS_EXECUTED.CYCLES_GE_1_UOP_EXEC	
EventSel=B1H, UMask=01H, CMask=1	Cycles where at least 1 uop was executed per-thread.
UOPS_EXECUTED.CYCLES_GE_2_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=2	Cycles where at least 2 uops were executed per-thread.
UOPS_EXECUTED.CYCLES_GE_3_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=3	Cycles where at least 3 uops were executed per-thread.
UOPS_EXECUTED.CYCLES_GE_4_UOPS_EXEC	
EventSel=B1H, UMask=01H, CMask=4	Cycles where at least 4 uops were executed per-thread.
UOPS_EXECUTED.CORE	
EventSel=B1H, UMask=02H	Counts total number of uops to be executed per-core each cycle.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
UOPS_EXECUTED.CORE_CYCLES_GE_1	
EventSel=B1H, UMask=02H, CMask=1	Cycles at least 1 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_2	
EventSel=B1H, UMask=02H, CMask=2	Cycles at least 2 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_3	
EventSel=B1H, UMask=02H, CMask=3	Cycles at least 3 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_4	
EventSel=B1H, UMask=02H, CMask=4	Cycles at least 4 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_NONE	
EventSel=B1H, UMask=02H, Invert=1	Cycles with no micro-ops executed from any thread on physical core.
OFFCORE_REQUESTS_BUFFER.SQ_FULL	
EventSel=B2H, UMask=01H	Cases when offcore requests buffer cannot take more entries for core.
TLB_FLUSH.DTLB_THREAD	
EventSel=BDH, UMask=01H	DTLB flush attempts of the thread-specific entries.
TLB_FLUSH.STLB_ANY	
EventSel=BDH, UMask=20H	Count number of STLB flush attempts.
PAGE_WALKS.LLC_MISS	
EventSel=BEH, UMask=01H	Number of any page walk that had a miss in LLC.
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural	Number of instructions at retirement.
INST_RETIRED.PREC_DIST	
EventSel=C0H, UMask=01H, Precise	Precise instruction retired event with HW to reduce effect of PEBS shadow in IP distribution.
OTHER_ASSISTS.AVX_STORE	
EventSel=C1H, UMask=08H	Number of assists associated with 256-bit AVX store operations.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
OTHER_ASSISTS.AVX_TO_SSE	
EventSel=C1H, UMask=10H	Number of transitions from AVX-256 to legacy SSE when penalty applicable.
OTHER_ASSISTS.SSE_TO_AVX	
EventSel=C1H, UMask=20H	Number of transitions from SSE to AVX-256 when penalty applicable.
OTHER_ASSISTS.ANY_WB_ASSIST	
EventSel=C1H, UMask=80H	Number of times any microcode assist is invoked by HW upon uop writeback.
UOPS_RETIRED.ALL	
EventSel=C2H, UMask=01H, Precise	Counts the number of micro-ops retired, Use cmask=1 and invert to count active cycles or stalled cycles.
UOPS_RETIRED.STALL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=1	Cycles without actually retired uops.
UOPS_RETIRED.TOTAL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=10	Cycles with less than 10 actually retired uops.
UOPS_RETIRED.CORE_STALL_CYCLES	
EventSel=C2H, UMask=01H, AnyThread=1, Invert=1, CMask=1	Cycles without actually retired uops.
UOPS_RETIRED.RETIRE_SLOTS	
EventSel=C2H, UMask=02H, Precise	Counts the number of retirement slots used each cycle.
MACHINE_CLEARS.COUNT	
EventSel=C3H, UMask=01H, EdgeDetect=1, CMask=1	Number of machine clears (nukes) of any type.
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	Counts the number of machine clears due to memory order conflicts.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=04H	Number of self-modifying-code machine clears detected.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
MACHINE_CLEARS.MASKMOV	
EventSel=C3H, UMask=20H	Counts the number of executed AVX masked load operations that refer to an illegal address range with the mask bits set to 0.
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	Branch instructions at retirement.
BR_INST_RETIRED.CONDITIONAL	
EventSel=C4H, UMask=01H, Precise	Counts the number of conditional branch instructions retired.
BR_INST_RETIRED.NEAR_CALL	
EventSel=C4H, UMask=02H, Precise	Direct and indirect near call instructions retired.
BR_INST_RETIRED.NEAR_CALL_R3	
EventSel=C4H, UMask=02H, USR=1,OS=0, Precise	Direct and indirect macro near call instructions retired (captured in ring 3).
BR_INST_RETIRED.NEAR_RETURN	
EventSel=C4H, UMask=08H, Precise	Counts the number of near return instructions retired.
BR_INST_RETIRED.NOT_TAKEN	
EventSel=C4H, UMask=10H	Counts the number of not taken branch instructions retired.
BR_INST_RETIRED.NEAR_TAKEN	
EventSel=C4H, UMask=20H, Precise	Number of near taken branches retired.
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=40H	Number of far branches retired.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	Mispredicted branch instructions at retirement.
BR_MISP_RETIRED.CONDITIONAL	
EventSel=C5H, UMask=01H, Precise	Mispredicted conditional branch instructions retired.
BR_MISP_RETIRED.NEAR_TAKEN	
EventSel=C5H, UMask=20H, Precise	Mispredicted taken branch instructions retired.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
FP_ASSIST.X87_OUTPUT	
EventSel=CAH, UMask=02H	Number of X87 FP assists due to output values.
FP_ASSIST.X87_INPUT	
EventSel=CAH, UMask=04H	Number of X87 FP assists due to input values.
FP_ASSIST.SIMD_OUTPUT	
EventSel=CAH, UMask=08H	Number of SIMD FP assists due to output values.
FP_ASSIST.SIMD_INPUT	
EventSel=CAH, UMask=10H	Number of SIMD FP assists due to input values.
FP_ASSIST.ANY	
EventSel=CAH, UMask=1EH, CMask=1	Cycles with any input/output SSE* or FP assists.
ROB_MISC_EVENTS.LBR_INSERTS	
EventSel=CCH, UMask=20H	Count cases of saving new LBR records by hardware.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_4	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x4 , Precise	Loads with latency value being above 4.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_8	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x8 , Precise	Loads with latency value being above 8.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_16	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x10 , Precise	Loads with latency value being above 16.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_32	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x20 , Precise	Loads with latency value being above 32.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_64	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x40 , Precise	Loads with latency value being above 64.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_128	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x80 , Precise	Loads with latency value being above 128.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_256	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x100 , Precise	Loads with latency value being above 256.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_512	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x200 , Precise	Loads with latency value being above 512.
MEM_TRANS_RETIRED.PRECISE_STORE	
EventSel=CDH, UMask=02H, Precise	Sample stores and collect precise store operation via PEBS record. PMC3 only.
MEM_UOPS_RETIRED.STLB_MISS_LOADS	
EventSel=D0H, UMask=11H, Precise	Retired load uops that miss the STLB.
MEM_UOPS_RETIRED.STLB_MISS_STORES	
EventSel=D0H, UMask=12H, Precise	Retired store uops that miss the STLB.
MEM_UOPS_RETIRED.LOCK_LOADS	
EventSel=D0H, UMask=21H, Precise	Retired load uops with locked access.
MEM_UOPS_RETIRED.SPLIT_LOADS	
EventSel=D0H, UMask=41H, Precise	Retired load uops that split across a cacheline boundary.
MEM_UOPS_RETIRED.SPLIT_STORES	
EventSel=D0H, UMask=42H, Precise	Retired store uops that split across a cacheline boundary.
MEM_UOPS_RETIRED.ALL_LOADS	
EventSel=D0H, UMask=81H, Precise	All retired load uops.
MEM_UOPS_RETIRED.ALL_STORES	
EventSel=D0H, UMask=82H, Precise	All retired store uops.
MEM_LOAD_UOPS_RETIRED.L1_HIT	
EventSel=D1H, UMask=01H, Precise	Retired load uops with L1 cache hits as data sources.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
MEM_LOAD_UOPS_RETIRED.L2_HIT	
EventSel=D1H, UMask=02H, Precise	Retired load uops with L2 cache hits as data sources.
MEM_LOAD_UOPS_RETIRED.LLC_HIT	
EventSel=D1H, UMask=04H, Precise	Retired load uops whose data source was LLC hit with no snoop required.
MEM_LOAD_UOPS_RETIRED.L1_MISS	
EventSel=D1H, UMask=08H, Precise	Retired load uops whose data source followed an L1 miss.
MEM_LOAD_UOPS_RETIRED.L2_MISS	
EventSel=D1H, UMask=10H, Precise	Retired load uops that missed L2, excluding unknown sources.
MEM_LOAD_UOPS_RETIRED.LLC_MISS	
EventSel=D1H, UMask=20H, Precise	Retired load uops whose data source is LLC miss.
MEM_LOAD_UOPS_RETIRED.HIT_LFB	
EventSel=D1H, UMask=40H, Precise	Retired load uops which data sources were load uops missed L1 but hit FB due to preceding miss to the same cache line with data not ready.
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_MISS	
EventSel=D2H, UMask=01H, Precise	Retired load uops whose data source was an on-package core cache LLC hit and cross-core snoop missed.
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_HIT	
EventSel=D2H, UMask=02H, Precise	Retired load uops whose data source was an on-package LLC hit and cross-core snoop hits.
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_HITM	
EventSel=D2H, UMask=04H, Precise	Retired load uops whose data source was an on-package core cache with HitM responses.
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_NONE	
EventSel=D2H, UMask=08H, Precise	Retired load uops whose data source was LLC hit with no snoop required.
MEM_LOAD_UOPS_LLC_MISS_RETIRED.LOCAL_DRAM	
EventSel=D3H, UMask=01H	Retired load uops whose data source was local memory (cross-socket snoop not needed or missed).

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
BACLEARS.ANY	
EventSel=E6H, UMask=1FH	Number of front end re-steers due to BPU misprediction.
L2_TRANS.DEMAND_DATA_RD	
EventSel=F0H, UMask=01H	Demand Data Read requests that access L2 cache.
L2_TRANS.RFO	
EventSel=F0H, UMask=02H	RFO requests that access L2 cache.
L2_TRANS.CODE_RD	
EventSel=F0H, UMask=04H	L2 cache accesses when fetching instructions.
L2_TRANS.ALL_PF	
EventSel=F0H, UMask=08H	Any MLC or LLC HW prefetch accessing L2, including rejects.
L2_TRANS.L1D_WB	
EventSel=F0H, UMask=10H	L1D writebacks that access L2 cache.
L2_TRANS.L2_FILL	
EventSel=F0H, UMask=20H	L2 fill requests that access L2 cache.
L2_TRANS.L2_WB	
EventSel=F0H, UMask=40H	L2 writebacks that access L2 cache.
L2_TRANS.ALL_REQUESTS	
EventSel=F0H, UMask=80H	Transactions accessing L2 pipe.
L2_LINES_IN.I	
EventSel=F1H, UMask=01H	L2 cache lines in I state filling L2.
L2_LINES_IN.S	
EventSel=F1H, UMask=02H	L2 cache lines in S state filling L2.
L2_LINES_IN.E	
EventSel=F1H, UMask=04H	L2 cache lines in E state filling L2.
L2_LINES_IN.ALL	
EventSel=F1H, UMask=07H	L2 cache lines filling L2.

Table 6: Performance Events In the Processor Core Based on the Ivy Bridge Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3AH)

Event Name	
Configuration	Description
L2_LINES_OUT.DEMAND_CLEAN	
EventSel=F2H, UMask=01H	Clean L2 cache lines evicted by demand.
L2_LINES_OUT.DEMAND_DIRTY	
EventSel=F2H, UMask=02H	Dirty L2 cache lines evicted by demand.
L2_LINES_OUT.PF_CLEAN	
EventSel=F2H, UMask=04H	Clean L2 cache lines evicted by the MLC prefetcher.
L2_LINES_OUT.PF_DIRTY	
EventSel=F2H, UMask=08H	Dirty L2 cache lines evicted by the MLC prefetcher.
L2_LINES_OUT.DIRTY_ALL	
EventSel=F2H, UMask=0AH	Dirty L2 cache lines filling the L2.
SQ_MISC.SPLIT_LOCK	
EventSel=F4H, UMask=10H	Split locks in SQ.

Additional information on event specifics (e.g. derivative events using specific IA32_PERFEVTSELx modifiers, limitations, special notes and recommendations) can be found at https://software.intel.com/en-us/forums/software-tuning-performance-optimization-platform-monitoring

Performance Monitoring Events based on Ivy Bridge-E Microarchitecture - 3rd Generation Intel® Core™ Processors

3rd generation Intel® Core™ processors Intel Xeon processor E5 v2 family and Intel Xeon processor E7 v2 family are based on Intel Microarchitecture code name Ivy Bridge-E. Performance-monitoring events in the processor core are listed in the table below.

Table 7: Performance Events In the Processor Core Based on the Ivy Bridge-E Microarchitecture 3rd Generation Intel® Core™ i7, i5, i3 Processors (06_3EH)

Event Name	
Configuration	Description
DTLB_LOAD_MISSES.DEMAND_LD_WALK_COMPLETED	
EventSel=08H, UMask=82H	Demand load Miss in all translation lookaside buffer (TLB) levels causes a page walk that completes of any page size.
DTLB_LOAD_MISSES.DEMAND_LD_WALK_DURATION	
EventSel=08H, UMask=84H	Demand load cycles page miss handler (PMH) is busy with this walk.
MEM_LOAD_UOPS_LLC_MISS_RETIRED.LOCAL_DRAM	
EventSel=D3H, UMask=03H	Retired load uops whose data source was local DRAM (Snoop not needed, Snoop Miss, or Snoop Hit data not forwarded).
MEM_LOAD_UOPS_LLC_MISS_RETIRED.REMOTE_DRAM	
EventSel=D3H, UMask=0CH	Retired load uops whose data source was remote DRAM (Snoop not needed, Snoop Miss, or Snoop Hit data not forwarded).
MEM_LOAD_UOPS_LLC_MISS_RETIRED.REMOTE_HITM	
EventSel=D3H, UMask=10H	Remote cache HITM.
MEM_LOAD_UOPS_LLC_MISS_RETIRED.REMOTE_FWD	
EventSel=D3H, UMask=20H	Data forwarded from remote cache.

Additional information on event specifics (e.g. derivative events using specific IA32_PERFEVTSELx modifiers, limitations, special notes and recommendations) can be found at https://software.intel.com/en-us/forums/software-tuning-performance-optimization-platform-monitoring

Performance Monitoring Events based on Sandy Bridge Microarchitecture - 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series

2nd generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx processor series, and Intel Xeon processor E3-1200 product family are based on the Intel Microarchitecture code name Sandy Bridge. performance-monitoring events in the processor core are listed in the following tables

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed	This event counts the number of instructions retired from execution. For instructions that consist of multiple micro-ops, this event counts the retirement of the last micro-op of the instruction. Counting continues during hardware interrupts, traps, and inside interrupt handlers. .
CPU_CLK_UNHALTED.THREAD	
Architectural, Fixed	This event counts the number of core cycles while the thread is not in a halt state. The thread enters the halt state when it is running the HLT instruction. This event is a component in many key event ratios. The core frequency may change from time to time due to transitions associated with Enhanced Intel SpeedStep Technology or TM2. For this reason this event may have a changing ratio with regards to time. When the core frequency is constant, this event can approximate elapsed time while the core was not in the halt state. It is counted on a dedicated fixed counter, leaving the four (eight when Hyperthreading is disabled) programmable counters available for other events. .
CPU_CLK_UNHALTED.THREAD_ANY	
AnyThread=1, Architectural, Fixed	Core cycles when at least one thread on the physical core is not in halt state.
LD_BLOCKS.DATA_UNKNOWN	
EventSel=03H, UMask=01H	Loads delayed due to SB blocks, preceding store operations with known addresses but unknown data.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
LD_BLOCKS.STORE_FORWARD	
EventSel=03H, UMask=02H	This event counts loads that followed a store to the same address, where the data could not be forwarded inside the pipeline from the store to the load. The most common reason why store forwarding would be blocked is when a load's address range overlaps with a preceeding smaller uncompleted store. See the table of not supported store forwards in the Intel® 64 and IA-32 Architectures Optimization Reference Manual. The penalty for blocked store forwarding is that the load must wait for the store to complete before it can be issued.
LD_BLOCKS.NO_SR	
EventSel=03H, UMask=08H	This event counts the number of times that split load operations are temporarily blocked because all resources for handling the split accesses are in use.
LD_BLOCKS.ALL_BLOCK	
EventSel=03H, UMask=10H	Number of cases where any load ends up with a valid block-code written to the load buffer (including blocks due to Memory Order Buffer (MOB), Data Cache Unit (DCU), TLB, but load has no DCU miss).
MISALIGN_MEM_REF.LOADS	
EventSel=05H, UMask=01H	Speculative cache line split load uops dispatched to L1 cache.
MISALIGN_MEM_REF.STORES	
EventSel=05H, UMask=02H	Speculative cache line split STA uops dispatched to L1 cache.
LD_BLOCKS_PARTIAL.ADDRESS_ALIAS	
EventSel=07H, UMask=01H	Aliasing occurs when a load is issued after a store and their memory addresses are offset by 4K. This event counts the number of loads that aliased with a preceding store, resulting in an extended address check in the pipeline. The enhanced address check typically has a performance penalty of 5 cycles.
LD_BLOCKS_PARTIAL.ALL_STA_BLOCK	
EventSel=07H, UMask=08H	This event counts the number of times that load operations are temporarily blocked because of older stores, with addresses that are not yet known. A load operation may incur more than one block of this type.
DTLB_LOAD_MISSES.MISS_CAUSES_A_WALK	
EventSel=08H, UMask=01H	Load misses in all DTLB levels that cause page walks.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
DTLB_LOAD_MISSES.WALK_COMPLETED	
EventSel=08H, UMask=02H	Load misses at all DTLB levels that cause completed page walks.
DTLB_LOAD_MISSES.WALK_DURATION	
EventSel=08H, UMask=04H	This event counts cycles when the page miss handler (PMH) is servicing page walks caused by DTLB load misses.
DTLB_LOAD_MISSES.STLB_HIT	
EventSel=08H, UMask=10H	This event counts load operations that miss the first DTLB level but hit the second and do not cause any page walks. The penalty in this case is approximately 7 cycles.
INT_MISC.RECOVERY_CYCLES	
EventSel=0DH, UMask=03H, CMask=1	Number of cycles waiting for the checkpoints in Resource Allocation Table (RAT) to be recovered after Nuke due to all other cases except JEClear (e.g. whenever a ucode assist is needed like SSE exception, memory disambiguation, etc...).
INT_MISC.RECOVERY_STALLS_COUNT	
EventSel=0DH, UMask=03H, EdgeDetect=1, CMask=1	Number of occurences waiting for the checkpoints in Resource Allocation Table (RAT) to be recovered after Nuke due to all other cases except JEClear (e.g. whenever a ucode assist is needed like SSE exception, memory disambiguation, etc...).
INT_MISC.RECOVERY_CYCLES_ANY	
EventSel=0DH, UMask=03H, AnyThread=1, CMask=1	Core cycles the allocator was stalled due to recovery from earlier clear event for any thread running on the physical core (e.g. misprediction or memory nuke).
INT_MISC.RAT_STALL_CYCLES	
EventSel=0DH, UMask=40H	Cycles when Resource Allocation Table (RAT) external stall is sent to Instruction Decode Queue (IDQ) for the thread.
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=01H	This event counts the number of Uops issued by the front-end of the pipeilne to the back-end.
UOPS_ISSUED.STALL_CYCLES	
EventSel=0EH, UMask=01H, Invert=1, CMask=1	Cycles when Resource Allocation Table (RAT) does not issue Uops to Reservation Station (RS) for the thread.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
UOPS_ISSUED.CORE_STALL_CYCLES	
EventSel=0EH, UMask=01H, AnyThread=1, Invert=1, CMask=1	Cycles when Resource Allocation Table (RAT) does not issue Uops to Reservation Station (RS) for all threads.
FP_COMP_OPS_EXE.X87	
EventSel=10H, UMask=01H	Number of FP Computational Uops Executed this cycle. The number of FADD, FSUB, FCOM, FMULs, integer MULsand IMULs, FDIVs, FPREMs, FSQRTS, integer DIVs, and IDIVs. This event does not distinguish an FADD used in the middle of a transcendental flow from a s.
FP_COMP_OPS_EXE.SSE_PACKED_DOUBLE	
EventSel=10H, UMask=10H	Number of SSE* or AVX-128 FP Computational packed double-precision uops issued this cycle.
FP_COMP_OPS_EXE.SSE_SCALAR_SINGLE	
EventSel=10H, UMask=20H	Number of SSE* or AVX-128 FP Computational scalar single-precision uops issued this cycle.
FP_COMP_OPS_EXE.SSE_PACKED_SINGLE	
EventSel=10H, UMask=40H	Number of SSE* or AVX-128 FP Computational packed single-precision uops issued this cycle.
FP_COMP_OPS_EXE.SSE_SCALAR_DOUBLE	
EventSel=10H, UMask=80H	Number of SSE* or AVX-128 FP Computational scalar double-precision uops issued this cycle.
SIMD_FP_256.PACKED_SINGLE	
EventSel=11H, UMask=01H	Number of GSSE-256 Computational FP single precision uops issued this cycle.
SIMD_FP_256.PACKED_DOUBLE	
EventSel=11H, UMask=02H	Number of AVX-256 Computational FP double precision uops issued this cycle.
ARITH.FPU_DIV_ACTIVE	
EventSel=14H, UMask=01H	Cycles when divider is busy executing divide operations.
ARITH.FPU_DIV	
EventSel=14H, UMask=01H, EdgeDetect=1, CMask=1	This event counts the number of the divide operations executed.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
INSTS_WRITTEN_TO_IQ.INSTS	
EventSel=17H, UMask=01H	Valid instructions written to IQ per cycle.
L2_RQSTS.DEMAND_DATA_RD_HIT	
EventSel=24H, UMask=01H	Demand Data Read requests that hit L2 cache.
L2_RQSTS.ALL_DEMAND_DATA_RD	
EventSel=24H, UMask=03H	Demand Data Read requests.
L2_RQSTS.RFO_HIT	
EventSel=24H, UMask=04H	RFO requests that hit L2 cache.
L2_RQSTS.RFO_MISS	
EventSel=24H, UMask=08H	RFO requests that miss L2 cache.
L2_RQSTS.ALL_RFO	
EventSel=24H, UMask=0CH	RFO requests to L2 cache.
L2_RQSTS.CODE_RD_HIT	
EventSel=24H, UMask=10H	L2 cache hits when fetching instructions, code reads.
L2_RQSTS.CODE_RD_MISS	
EventSel=24H, UMask=20H	L2 cache misses when fetching instructions.
L2_RQSTS.ALL_CODE_RD	
EventSel=24H, UMask=30H	L2 code requests.
L2_RQSTS.PF_HIT	
EventSel=24H, UMask=40H	Requests from the L2 hardware prefetchers that hit L2 cache.
L2_RQSTS.PF_MISS	
EventSel=24H, UMask=80H	Requests from the L2 hardware prefetchers that miss L2 cache.
L2_RQSTS.ALL_PF	
EventSel=24H, UMask=C0H	Requests from L2 hardware prefetchers.
L2_STORE_LOCK_RQSTS.MISS	
EventSel=27H, UMask=01H	RFOs that miss cache lines.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
L2_STORE_LOCK_RQSTS.HIT_E	
EventSel=27H, UMask=04H	RFOs that hit cache lines in E state.
L2_STORE_LOCK_RQSTS.HIT_M	
EventSel=27H, UMask=08H	RFOs that hit cache lines in M state.
L2_STORE_LOCK_RQSTS.ALL	
EventSel=27H, UMask=0FH	RFOs that access cache lines in any state.
L2_L1D_WB_RQSTS.MISS	
EventSel=28H, UMask=01H	Count the number of modified Lines evicted from L1 and missed L2. (Non-rejected WBs from the DCU.).
L2_L1D_WB_RQSTS.HIT_S	
EventSel=28H, UMask=02H	Not rejected writebacks from L1D to L2 cache lines in S state.
L2_L1D_WB_RQSTS.HIT_E	
EventSel=28H, UMask=04H	Not rejected writebacks from L1D to L2 cache lines in E state.
L2_L1D_WB_RQSTS.HIT_M	
EventSel=28H, UMask=08H	Not rejected writebacks from L1D to L2 cache lines in M state.
L2_L1D_WB_RQSTS.ALL	
EventSel=28H, UMask=0FH	Not rejected writebacks from L1D to L2 cache lines in any state.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	Core-originated cacheable demand requests missed LLC.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	Core-originated cacheable demand requests that refer to LLC.
CPU_CLK_UNHALTED.THREAD_P	
EventSel=3CH, UMask=00H, Architectural	Thread cycles when thread is not in halt state.
CPU_CLK_UNHALTED.THREAD_P_ANY	
EventSel=3CH, UMask=00H, AnyThread=1, Architectural	Core cycles when at least one thread on the physical core is not in halt state.
CPU_CLK_THREAD_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	Reference cycles when the thread is unhalted (counts at 100 MHz rate).

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
CPU_CLK_THREAD_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Reference cycles when the at least one thread on the physical core is unhalted (counts at 100 MHz rate).
CPU_CLK_UNHALTED.REF_XCLK	
EventSel=3CH, UMask=01H, Architectural	Reference cycles when the thread is unhalted (counts at 100 MHz rate).
CPU_CLK_UNHALTED.REF_XCLK_ANY	
EventSel=3CH, UMask=01H, AnyThread=1, Architectural	Reference cycles when the at least one thread on the physical core is unhalted (counts at 100 MHz rate).
CPU_CLK_THREAD_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Count XClk pulses when this thread is unhalted and the other is halted.
CPU_CLK_UNHALTED.ONE_THREAD_ACTIVE	
EventSel=3CH, UMask=02H	Count XClk pulses when this thread is unhalted and the other thread is halted.
L1D_PEND_MISS.PENDING	
EventSel=48H, UMask=01H	L1D miss oustandings duration in cycles.
L1D_PEND_MISS.PENDING_CYCLES	
EventSel=48H, UMask=01H, CMask=1	Cycles with L1D load Misses outstanding.
L1D_PEND_MISS.PENDING_CYCLES_ANY	
EventSel=48H, UMask=01H, AnyThread=1, CMask=1	Cycles with L1D load Misses outstanding from any thread on physical core.
L1D_PEND_MISS.FB_FULL	
EventSel=48H, UMask=02H, CMask=1	Cycles a demand request was blocked due to Fill Buffers inavailability.
DTLB_STORE_MISSES.MISS_CAUSES_A_WALK	
EventSel=49H, UMask=01H	Store misses in all DTLB levels that cause page walks.
DTLB_STORE_MISSES.WALK_COMPLETED	
EventSel=49H, UMask=02H	Store misses in all DTLB levels that cause completed page walks.
DTLB_STORE_MISSES.WALK_DURATION	
EventSel=49H, UMask=04H	Cycles when PMH is busy with page walks.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
DTLB_STORE_MISSES.STLB_HIT	
EventSel=49H, UMask=10H	Store operations that miss the first TLB level but hit the second and do not cause page walks.
LOAD_HIT_PRE.SW_PF	
EventSel=4CH, UMask=01H	Not software-prefetch load dispatches that hit FB allocated for software prefetch.
LOAD_HIT_PRE.HW_PF	
EventSel=4CH, UMask=02H	Not software-prefetch load dispatches that hit FB allocated for hardware prefetch.
HW_PRE_REQ.DL1_MISS	
EventSel=4EH, UMask=02H	Hardware Prefetch requests that miss the L1D cache. This accounts for both L1 streamer and IP-based (IPP) HW prefetchers. A request is being counted each time it access the cache & miss it, including if a block is applicable or if hit the Fill Buffer for .
EPT.WALK_CYCLES	
EventSel=4FH, UMask=10H	Cycle count for an Extended Page table walk. The Extended Page Directory cache is used by Virtual Machine operating systems while the guest operating systems use the standard TLB caches.
L1D.REPLACEMENT	
EventSel=51H, UMask=01H	This event counts L1D data line replacements. Replacements occur when a new line is brought into the cache, causing eviction of a line loaded earlier. .
L1D.ALLOCATED_IN_M	
EventSel=51H, UMask=02H	Allocated L1D data cache lines in M state.
L1D.EVICTION	
EventSel=51H, UMask=04H	L1D data cache lines in M state evicted due to replacement.
L1D.ALL_M_REPLACEMENT	
EventSel=51H, UMask=08H	Cache lines in M state evicted out of L1D due to Snoop HitM or dirty line replacement.
PARTIAL_RAT_STALLS.FLAGS_MERGE_UOP	
EventSel=59H, UMask=20H	Increments the number of flags-merge uops in flight each cycle.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
PARTIAL_RAT_STALLS.FLAGS_MERGE_UOP_CYCLES	
EventSel=59H, UMask=20H, CMask=1	This event counts the number of cycles spent executing performance-sensitive flags-merging uops. For example, shift CL (merge_arith_flags). For more details, See the Intel® 64 and IA-32 Architectures Optimization Reference Manual.
PARTIAL_RAT_STALLS.SLOW_LEA_WINDOW	
EventSel=59H, UMask=40H	This event counts the number of cycles with at least one slow LEA uop being allocated. A uop is generally considered as slow LEA if it has three sources (for example, two sources and immediate) regardless of whether it is a result of LEA instruction or not. Examples of the slow LEA uop are or uops with base, index, and offset source operands using base and index reqisters, where base is EBR/RBP/R13, using RIP relative or 16-bit addressing modes. See the Intel® 64 and IA-32 Architectures Optimization Reference Manual for more details about slow LEA instructions.
PARTIAL_RAT_STALLS.MUL_SINGLE_UOP	
EventSel=59H, UMask=80H	Multiply packed/scalar single precision uops allocated.
RESOURCE_STALLS2.ALL_FL_EMPTY	
EventSel=5BH, UMask=0CH	Cycles with either free list is empty.
RESOURCE_STALLS2.ALL_PRF_CONTROL	
EventSel=5BH, UMask=0FH	Resource stalls2 control structures full for physical registers.
RESOURCE_STALLS2.BOB_FULL	
EventSel=5BH, UMask=40H	Cycles when Allocator is stalled if BOB is full and new branch needs it.
RESOURCE_STALLS2.OOO_RSRC	
EventSel=5BH, UMask=4FH	Resource stalls out of order resources full.
CPL_CYCLES.RING0	
EventSel=5CH, UMask=01H	Unhalted core cycles when the thread is in ring 0.
CPL_CYCLES.RING0_TRANS	
EventSel=5CH, UMask=01H, EdgeDetect=1, CMask=1	Number of intervals between processor halts while thread is in ring 0.
CPL_CYCLES.RING123	
EventSel=5CH, UMask=02H	Unhalted core cycles when thread is in rings 1, 2, or 3.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
RS_EVENTS.EMPTY_CYCLES	
EventSel=5EH, UMask=01H	Cycles when Reservation Station (RS) is empty for the thread.
RS_EVENTS.EMPTY_END	
EventSel=5EH, UMask=01H, EdgeDetect=1, Invert=1, CMask=1	Counts end of periods where the Reservation Station (RS) was empty. Could be useful to precisely locate Frontend Latency Bound issues.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD	
EventSel=60H, UMask=01H	Offcore outstanding Demand Data Read transactions in uncore queue.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_DATA_RD	
EventSel=60H, UMask=01H, CMask=1	Cycles when offcore outstanding Demand Data Read transactions are present in SuperQueue (SQ), queue to uncore.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_DATA_RD_C6	
EventSel=60H, UMask=01H, CMask=6	Cycles with at least 6 offcore outstanding Demand Data Read transactions in uncore queue.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND_RFO	
EventSel=60H, UMask=04H	Offcore outstanding RFO store transactions in SuperQueue (SQ), queue to uncore.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DEMAND_RFO	
EventSel=60H, UMask=04H, CMask=1	Offcore outstanding demand rfo reads transactions in SuperQueue (SQ), queue to uncore, every cycle.
OFFCORE_REQUESTS_OUTSTANDING.ALL_DATA_RD	
EventSel=60H, UMask=08H	Offcore outstanding cacheable Core Data Read transactions in SuperQueue (SQ), queue to uncore.
OFFCORE_REQUESTS_OUTSTANDING.CYCLES_WITH_DATA_RD	
EventSel=60H, UMask=08H, CMask=1	Cycles when offcore outstanding cacheable Core Data Read transactions are present in SuperQueue (SQ), queue to uncore.
LOCK_CYCLES.SPLIT_LOCK_UC_LOCK_DURATION	
EventSel=63H, UMask=01H	Cycles when L1 and L2 are locked due to UC or split lock.
LOCK_CYCLES.CACHE_LOCK_DURATION	
EventSel=63H, UMask=02H	Cycles when L1D is locked.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
IDQ.EMPTY	
EventSel=79H, UMask=02H	Instruction Decode Queue (IDQ) empty cycles.
IDQ.MITE_UOPS	
EventSel=79H, UMask=04H	Uops delivered to Instruction Decode Queue (IDQ) from MITE path.
IDQ.MITE_CYCLES	
EventSel=79H, UMask=04H, CMask=1	Cycles when uops are being delivered to Instruction Decode Queue (IDQ) from MITE path.
IDQ.DSB_UOPS	
EventSel=79H, UMask=08H	Uops delivered to Instruction Decode Queue (IDQ) from the Decode Stream Buffer (DSB) path.
IDQ.DSB_CYCLES	
EventSel=79H, UMask=08H, CMask=1	Cycles when uops are being delivered to Instruction Decode Queue (IDQ) from Decode Stream Buffer (DSB) path.
IDQ.MS_DSB_UOPS	
EventSel=79H, UMask=10H	Uops initiated by Decode Stream Buffer (DSB) that are being delivered to Instruction Decode Queue (IDQ) while Microcode Sequenser (MS) is busy.
IDQ.MS_DSB_CYCLES	
EventSel=79H, UMask=10H, CMask=1	Cycles when uops initiated by Decode Stream Buffer (DSB) are being delivered to Instruction Decode Queue (IDQ) while Microcode Sequenser (MS) is busy.
IDQ.MS_DSB_OCCUR	
EventSel=79H, UMask=10H, EdgeDetect=1, CMask=1	Deliveries to Instruction Decode Queue (IDQ) initiated by Decode Stream Buffer (DSB) while Microcode Sequenser (MS) is busy.
IDQ.ALL_DSB_CYCLES_4_UOPS	
EventSel=79H, UMask=18H, CMask=4	Cycles Decode Stream Buffer (DSB) is delivering 4 Uops.
IDQ.ALL_DSB_CYCLES_ANY_UOPS	
EventSel=79H, UMask=18H, CMask=1	Cycles Decode Stream Buffer (DSB) is delivering any Uop.
IDQ.MS_MITE_UOPS	
EventSel=79H, UMask=20H	Uops initiated by MITE and delivered to Instruction Decode Queue (IDQ) while Microcode Sequenser (MS) is busy.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
IDQ.ALL_MITE_CYCLES_4_UOPS	
EventSel=79H, UMask=24H, CMask=4	Cycles MITE is delivering 4 Uops.
IDQ.ALL_MITE_CYCLES_ANY_UOPS	
EventSel=79H, UMask=24H, CMask=1	Cycles MITE is delivering any Uop.
IDQ.MS_UOPS	
EventSel=79H, UMask=30H	Uops delivered to Instruction Decode Queue (IDQ) while Microcode Sequenser (MS) is busy.
IDQ.MS_CYCLES	
EventSel=79H, UMask=30H, CMask=1	This event counts cycles during which the microcode sequencer assisted the front-end in delivering uops. Microcode assists are used for complex instructions or scenarios that can't be handled by the standard decoder. Using other instructions, if possible, will usually improve performance. See the Intel® 64 and IA-32 Architectures Optimization Reference Manual for more information.
IDQ.MS_SWITCHES	
EventSel=79H, UMask=30H, EdgeDetect=1, CMask=1	Number of switches from DSB (Decode Stream Buffer) or MITE (legacy decode pipeline) to the Microcode Sequencer.
IDQ.MITE_ALL_UOPS	
EventSel=79H, UMask=3CH	Uops delivered to Instruction Decode Queue (IDQ) from MITE path.
ICACHE.HIT	
EventSel=80H, UMask=01H	Number of Instruction Cache, Streaming Buffer and Victim Cache Reads. both cacheable and noncacheable, including UC fetches.
ICACHE.MISSES	
EventSel=80H, UMask=02H	This event counts the number of instruction cache, streaming buffer and victim cache misses. Counting includes unchacheable accesses.
ITLB_MISSES.MISS_CAUSES_A_WALK	
EventSel=85H, UMask=01H	Misses at all ITLB levels that cause page walks.
ITLB_MISSES.WALK_COMPLETED	
EventSel=85H, UMask=02H	Misses in all ITLB levels that cause completed page walks.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
ITLB_MISSES.WALK_DURATION	
EventSel=85H, UMask=04H	This event count cycles when Page Miss Handler (PMH) is servicing page walks caused by ITLB misses.
ITLB_MISSES.STLB_HIT	
EventSel=85H, UMask=10H	Operations that miss the first ITLB level but hit the second and do not cause any page walks.
ILD_STALL.LCP	
EventSel=87H, UMask=01H	Stalls caused by changing prefix length of the instruction.
ILD_STALL.IQ_FULL	
EventSel=87H, UMask=04H	Stall cycles because IQ is full.
BR_INST_EXEC.NONTAKEN_CONDITIONAL	
EventSel=88H, UMask=41H	Not taken macro-conditional branches.
BR_INST_EXEC.TAKEN_CONDITIONAL	
EventSel=88H, UMask=81H	Taken speculative and retired macro-conditional branches.
BR_INST_EXEC.TAKEN_DIRECT_JUMP	
EventSel=88H, UMask=82H	Taken speculative and retired macro-conditional branch instructions excluding calls and indirects.
BR_INST_EXEC.TAKEN_INDIRECT_JUMP_NON_CALL_RET	
EventSel=88H, UMask=84H	Taken speculative and retired indirect branches excluding calls and returns.
BR_INST_EXEC.TAKEN_INDIRECT_NEAR_RETURN	
EventSel=88H, UMask=88H	Taken speculative and retired indirect branches with return mnemonic.
BR_INST_EXEC.TAKEN_DIRECT_NEAR_CALL	
EventSel=88H, UMask=90H	Taken speculative and retired direct near calls.
BR_INST_EXEC.TAKEN_INDIRECT_NEAR_CALL	
EventSel=88H, UMask=A0H	Taken speculative and retired indirect calls.
BR_INST_EXEC.ALL_CONDITIONAL	
EventSel=88H, UMask=C1H	Speculative and retired macro-conditional branches.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
BR_INST_EXEC.ALL_DIRECT_JMP	
EventSel=88H, UMask=C2H	Speculative and retired macro-unconditional branches excluding calls and indirects.
BR_INST_EXEC.ALL_INDIRECT_JUMP_NON_CALL_RET	
EventSel=88H, UMask=C4H	Speculative and retired indirect branches excluding calls and returns.
BR_INST_EXEC.ALL_INDIRECT_NEAR_RETURN	
EventSel=88H, UMask=C8H	Speculative and retired indirect return branches.
BR_INST_EXEC.ALL_DIRECT_NEAR_CALL	
EventSel=88H, UMask=D0H	Speculative and retired direct near calls.
BR_INST_EXEC.ALL_BRANCHES	
EventSel=88H, UMask=FFH	Speculative and retired branches.
BR_MISP_EXEC.NONTAKEN_CONDITIONAL	
EventSel=89H, UMask=41H	Not taken speculative and retired mispredicted macro conditional branches.
BR_MISP_EXEC.TAKEN_CONDITIONAL	
EventSel=89H, UMask=81H	Taken speculative and retired mispredicted macro conditional branches.
BR_MISP_EXEC.TAKEN_INDIRECT_JUMP_NON_CALL_RET	
EventSel=89H, UMask=84H	Taken speculative and retired mispredicted indirect branches excluding calls and returns.
BR_MISP_EXEC.TAKEN_RETURN_NEAR	
EventSel=89H, UMask=88H	Taken speculative and retired mispredicted indirect branches with return mnemonic.
BR_MISP_EXEC.TAKEN_DIRECT_NEAR_CALL	
EventSel=89H, UMask=90H	Taken speculative and retired mispredicted direct near calls.
BR_MISP_EXEC.TAKEN_INDIRECT_NEAR_CALL	
EventSel=89H, UMask=A0H	Taken speculative and retired mispredicted indirect calls.
BR_MISP_EXEC.ALL_CONDITIONAL	
EventSel=89H, UMask=C1H	Speculative and retired mispredicted macro conditional branches.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
BR_MISP_EXEC.ALL_INDIRECT_JUMP_NON_CALL_RET	
EventSel=89H, UMask=C4H	Mispredicted indirect branches excluding calls and returns.
BR_MISP_EXEC.ALL_DIRECT_NEAR_CALL	
EventSel=89H, UMask=D0H	Speculative and retired mispredicted direct near calls.
BR_MISP_EXEC.ALL_BRANCHES	
EventSel=89H, UMask=FFH	Speculative and retired mispredicted macro conditional branches.
IDQ_UOPS_NOT_DELIVERED.CORE	
EventSel=9CH, UMask=01H	This event counts the number of uops not delivered to the back-end per cycle, per thread, when the back-end was not stalled. In the ideal case 4 uops can be delivered each cycle. The event counts the undelivered uops - so if 3 were delivered in one cycle, the counter would be incremented by 1 for that cycle (4 - 3). If the back-end is stalled, the count for this event is not incremented even when uops were not delivered, because the back-end would not have been able to accept them. This event is used in determining the front-end bound category of the top-down pipeline slots characterization.
IDQ_UOPS_NOT_DELIVERED.CYCLES_0_UOPS_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=4	Cycles per thread when 4 or more uops are not delivered to Resource Allocation Table (RAT) when backend of the machine is not stalled.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_1_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=3	Cycles per thread when 3 or more uops are not delivered to Resource Allocation Table (RAT) when backend of the machine is not stalled.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_2_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=2	Cycles with less than 2 uops delivered by the front end.
IDQ_UOPS_NOT_DELIVERED.CYCLES_LE_3_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, CMask=1	Cycles with less than 3 uops delivered by the front end.
IDQ_UOPS_NOT_DELIVERED.CYCLES_GE_1_UOP_DELIV.CORE	
EventSel=9CH, UMask=01H, Invert=1, CMask=4	Cycles when 1 or more uops were delivered to the by the front end.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
IDQ_UOPS_NOT_DELIVERED.CYCLES_FE_WAS_OK	
EventSel=9CH, UMask=01H, Invert=1, CMask=1	Counts cycles FE delivered 4 uops or Resource Allocation Table (RAT) was stalling FE.
UOPS_DISPATCHED_PORT.PORT_0	
EventSel=A1H, UMask=01H	Cycles per thread when uops are dispatched to port 0.
UOPS_DISPATCHED_PORT.PORT_0_CORE	
EventSel=A1H, UMask=01H, AnyThread=1	Cycles per core when uops are dispatched to port 0.
UOPS_DISPATCHED_PORT.PORT_1	
EventSel=A1H, UMask=02H	Cycles per thread when uops are dispatched to port 1.
UOPS_DISPATCHED_PORT.PORT_1_CORE	
EventSel=A1H, UMask=02H, AnyThread=1	Cycles per core when uops are dispatched to port 1.
UOPS_DISPATCHED_PORT.PORT_2	
EventSel=A1H, UMask=0CH	Cycles per thread when load or STA uops are dispatched to port 2.
UOPS_DISPATCHED_PORT.PORT_2_CORE	
EventSel=A1H, UMask=0CH, AnyThread=1	Cycles per core when load or STA uops are dispatched to port 2.
UOPS_DISPATCHED_PORT.PORT_3	
EventSel=A1H, UMask=30H	Cycles per thread when load or STA uops are dispatched to port 3.
UOPS_DISPATCHED_PORT.PORT_3_CORE	
EventSel=A1H, UMask=30H, AnyThread=1	Cycles per core when load or STA uops are dispatched to port 3.
UOPS_DISPATCHED_PORT.PORT_4	
EventSel=A1H, UMask=40H	Cycles per thread when uops are dispatched to port 4.
UOPS_DISPATCHED_PORT.PORT_4_CORE	
EventSel=A1H, UMask=40H, AnyThread=1	Cycles per core when uops are dispatched to port 4.
UOPS_DISPATCHED_PORT.PORT_5	
EventSel=A1H, UMask=80H	Cycles per thread when uops are dispatched to port 5.
UOPS_DISPATCHED_PORT.PORT_5_CORE	
EventSel=A1H, UMask=80H, AnyThread=1	Cycles per core when uops are dispatched to port 5.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
RESOURCE_STALLS.ANY	
EventSel=A2H, UMask=01H	Resource-related stall cycles.
RESOURCE_STALLS.LB	
EventSel=A2H, UMask=02H	Counts the cycles of stall due to lack of load buffers.
RESOURCE_STALLS.RS	
EventSel=A2H, UMask=04H	Cycles stalled due to no eligible RS entry available.
RESOURCE_STALLS.SB	
EventSel=A2H, UMask=08H	Cycles stalled due to no store buffers available. (not including draining form sync).
RESOURCE_STALLS.LB_SB	
EventSel=A2H, UMask=0AH	Resource stalls due to load or store buffers all being in use.
RESOURCE_STALLS.MEM_RS	
EventSel=A2H, UMask=0EH	Resource stalls due to memory buffers or Reservation Station (RS) being fully utilized.
RESOURCE_STALLS.ROB	
EventSel=A2H, UMask=10H	Cycles stalled due to re-order buffer full.
RESOURCE_STALLS.OOO_RSRC	
EventSel=A2H, UMask=F0H	Resource stalls due to Rob being full, FCSW, MXCSR and OTHER.
CYCLE_ACTIVITY.CYCLES_L2_PENDING	
EventSel=A3H, UMask=01H, CMask=1	Each cycle there was a MLC-miss pending demand load this thread (i.e. Non-completed valid SQ entry allocated for demand load and waiting for Uncore), increment by 1. Note this is in MLC and connected to Umask 0.
CYCLE_ACTIVITY.CYCLES_L1D_PENDING	
EventSel=A3H, UMask=02H, CMask=2	Each cycle there was a miss-pending demand load this thread, increment by 1. Note this is in DCU and connected to Umask 1. Miss Pending demand load should be deduced by OR-ing increment bits of DCACHE_MISS_PEND.PENDING.
CYCLE_ACTIVITY.CYCLES_NO_DISPATCH	
EventSel=A3H, UMask=04H, CMask=4	Each cycle there was no dispatch for this thread, increment by 1. Note this is connect to Umask 2. No dispatch can be deduced from the UOPS_EXECUTED event.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
CYCLE_ACTIVITY.STALLS_L2_PENDING	
EventSel=A3H, UMask=05H, CMask=5	Each cycle there was a MLC-miss pending demand load and no uops dispatched on this thread (i.e. Non-completed valid SQ entry allocated for demand load and waiting for Uncore), increment by 1. Note this is in MLC and connected to Umask 0 and 2.
CYCLE_ACTIVITY.STALLS_L1D_PENDING	
EventSel=A3H, UMask=06H, CMask=6	Each cycle there was a miss-pending demand load this thread and no uops dispatched, increment by 1. Note this is in DCU and connected to Umask 1 and 2. Miss Pending demand load should be deduced by OR-ing increment bits of DCACHE_MISS_PEND.PENDING.
LSD.UOPS	
EventSel=A8H, UMask=01H	Number of Uops delivered by the LSD.
LSD.CYCLES_ACTIVE	
EventSel=A8H, UMask=01H, CMask=1	Cycles Uops delivered by the LSD, but didn't come from the decoder.
LSD.CYCLES_4_UOPS	
EventSel=A8H, UMask=01H, CMask=4	Cycles 4 Uops delivered by the LSD, but didn't come from the decoder.
DSB2MITE_SWITCHES.COUNT	
EventSel=ABH, UMask=01H	Decode Stream Buffer (DSB)-to-MITE switches.
DSB2MITE_SWITCHES.PENALTY_CYCLES	
EventSel=ABH, UMask=02H	This event counts the cycles attributed to a switch from the Decoded Stream Buffer (DSB), which holds decoded instructions, to the legacy decode pipeline. It excludes cycles when the back-end cannot accept new micro-ops. The penalty for these switches is potentially several cycles of instruction starvation, where no micro-ops are delivered to the back-end.
DSB_FILL.OTHER_CANCEL	
EventSel=ACH, UMask=02H	Cases of cancelling valid DSB fill not because of exceeding way limit.
DSB_FILL.EXCEED_DSB_LINES	
EventSel=ACH, UMask=08H	Cycles when Decode Stream Buffer (DSB) fill encounter more than 3 Decode Stream Buffer (DSB) lines.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
DSB_FILL.ALL_CANCEL	
EventSel=ACH, UMask=0AH	Cases of cancelling valid Decode Stream Buffer (DSB) fill not because of exceeding way limit.
ITLB.ITLB_FLUSH	
EventSel=AEH, UMask=01H	Flushing of the Instruction TLB (ITLB) pages, includes 4k/2M/4M pages.
OFFCORE_REQUESTS.DEMAND_DATA_RD	
EventSel=B0H, UMask=01H	Demand Data Read requests sent to uncore.
OFFCORE_REQUESTS.DEMAND_CODE_RD	
EventSel=B0H, UMask=02H	Cacheable and noncachaeble code read requests.
OFFCORE_REQUESTS.DEMAND_RFO	
EventSel=B0H, UMask=04H	Demand RFO requests including regular RFOs, locks, ItoM.
OFFCORE_REQUESTS.ALL_DATA_RD	
EventSel=B0H, UMask=08H	Demand and prefetch data reads.
UOPS_DISPATCHED.THREAD	
EventSel=B1H, UMask=01H	Uops dispatched per thread.
UOPS_DISPATCHED.STALL_CYCLES	
EventSel=B1H, UMask=01H, Invert=1, CMask=1	Cases of no uops dispatched per thread.
UOPS_DISPATCHED.CORE	
EventSel=B1H, UMask=02H	Uops dispatched from any thread.
UOPS_EXECUTED.CORE_CYCLES_GE_1	
EventSel=B1H, UMask=02H, CMask=1	Cycles at least 1 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_2	
EventSel=B1H, UMask=02H, CMask=2	Cycles at least 2 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_GE_3	
EventSel=B1H, UMask=02H, CMask=3	Cycles at least 3 micro-op is executed from any thread on physical core.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
UOPS_EXECUTED.CORE_CYCLES_GE_4	
EventSel=B1H, UMask=02H, CMask=4	Cycles at least 4 micro-op is executed from any thread on physical core.
UOPS_EXECUTED.CORE_CYCLES_NONE	
EventSel=B1H, UMask=02H, Invert=1	Cycles with no micro-ops executed from any thread on physical core.
OFFCORE_REQUESTS_BUFFER.SQ_FULL	
EventSel=B2H, UMask=01H	Cases when offcore requests buffer cannot take more entries for core.
AGU_BYPASS_CANCEL.COUNT	
EventSel=B6H, UMask=01H	This event counts executed load operations with all the following traits: 1. addressing of the format [base + offset], 2. the offset is between 1 and 2047, 3. the address specified in the base register is in one page and the address [base+offset] is in an.
TLB_FLUSH.DTLB_THREAD	
EventSel=BDH, UMask=01H	DTLB flush attempts of the thread-specific entries.
TLB_FLUSH.STLB_ANY	
EventSel=BDH, UMask=20H	STLB flush attempts.
PAGE_WALKS.LLC_MISS	
EventSel=BEH, UMask=01H	Number of any page walk that had a miss in LLC. Does not necessary cause a SUSPEND.
L1D_BLOCKS.BANK_CONFLICT_CYCLES	
EventSel=BFH, UMask=05H, CMask=1	Cycles when dispatched loads are cancelled due to L1D bank conflicts with other load ports.
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural	Number of instructions retired. General Counter - architectural event.
INST_RETIRED.PREC_DIST	
EventSel=C0H, UMask=01H, Precise	Instructions retired. (Precise Event - PEBS).
OTHER_ASSISTS.ITLB_MISS_RETIRED	
EventSel=C1H, UMask=02H	Retired instructions experiencing ITLB misses.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
OTHER_ASSISTS.AVX_STORE	
EventSel=C1H, UMask=08H	Number of GSSE memory assist for stores. GSSE microcode assist is being invoked whenever the hardware is unable to properly handle GSSE-256b operations.
OTHER_ASSISTS.AVX_TO_SSE	
EventSel=C1H, UMask=10H	Number of transitions from AVX-256 to legacy SSE when penalty applicable.
OTHER_ASSISTS.SSE_TO_AVX	
EventSel=C1H, UMask=20H	Number of transitions from SSE to AVX-256 when penalty applicable.
UOPS_RETIRED.ALL	
EventSel=C2H, UMask=01H, Precise	This event counts the number of micro-ops retired.
UOPS_RETIRED.STALL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=1	Cycles without actually retired uops.
UOPS_RETIRED.TOTAL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=10	Cycles with less than 10 actually retired uops.
UOPS_RETIRED.CORE_STALL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=1	Cycles without actually retired uops.
UOPS_RETIRED.RETIRE_SLOTS	
EventSel=C2H, UMask=02H, Precise	This event counts the number of retirement slots used each cycle. There are potentially 4 slots that can be used each cycle - meaning, 4 micro-ops or 4 instructions could retire each cycle. This event is used in determining the 'Retiring' category of the Top-Down pipeline slots characterization.
MACHINE_CLEARS.COUNT	
EventSel=C3H, UMask=01H, EdgeDetect=1, CMask=1	Number of machine clears (nukes) of any type.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	This event counts the number of memory ordering Machine Clears detected. Memory Ordering Machine Clears can result from memory disambiguation, external snoops, or cross SMT-HW-thread snoop (stores) hitting load buffers. Machine clears can have a significant performance impact if they are happening frequently.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=04H	This event is incremented when self-modifying code (SMC) is detected, which causes a machine clear. Machine clears can have a significant performance impact if they are happening frequently.
MACHINE_CLEARS.MASKMOV	
EventSel=C3H, UMask=20H	Maskmov false fault - counts number of time ucode passes through Maskmov flow due to instruction's mask being 0 while the flow was completed without raising a fault.
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	All (macro) branch instructions retired.
BR_INST_RETIRED.CONDITIONAL	
EventSel=C4H, UMask=01H, Precise	Conditional branch instructions retired.
BR_INST_RETIRED.NEAR_CALL	
EventSel=C4H, UMask=02H, Precise	Direct and indirect near call instructions retired.
BR_INST_RETIRED.NEAR_CALL_R3	
EventSel=C4H, UMask=02H, USR=1,OS=0, Precise	Direct and indirect macro near call instructions retired (captured in ring 3).
BR_INST_RETIRED.NEAR_RETURN	
EventSel=C4H, UMask=08H, Precise	Return instructions retired.
BR_INST_RETIRED.NOT_TAKEN	
EventSel=C4H, UMask=10H	Not taken branch instructions retired.
BR_INST_RETIRED.NEAR_TAKEN	
EventSel=C4H, UMask=20H, Precise	Taken branch instructions retired.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=40H	Far branch instructions retired.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	All mispredicted macro branch instructions retired.
BR_MISP_RETIRED.CONDITIONAL	
EventSel=C5H, UMask=01H, Precise	Mispredicted conditional branch instructions retired.
BR_MISP_RETIRED.NEAR_CALL	
EventSel=C5H, UMask=02H, Precise	Direct and indirect mispredicted near call instructions retired.
BR_MISP_RETIRED.NOT_TAKEN	
EventSel=C5H, UMask=10H, Precise	Mispredicted not taken branch instructions retired.
BR_MISP_RETIRED.TAKEN	
EventSel=C5H, UMask=20H, Precise	Mispredicted taken branch instructions retired.
FP_ASSIST.X87_OUTPUT	
EventSel=CAH, UMask=02H	Number of X87 assists due to output value.
FP_ASSIST.X87_INPUT	
EventSel=CAH, UMask=04H	Number of X87 assists due to input value.
FP_ASSIST.SIMD_OUTPUT	
EventSel=CAH, UMask=08H	Number of SIMD FP assists due to Output values.
FP_ASSIST.SIMD_INPUT	
EventSel=CAH, UMask=10H	Number of SIMD FP assists due to input values.
FP_ASSIST.ANY	
EventSel=CAH, UMask=1EH, CMask=1	Cycles with any input/output SSE or FP assist.
ROB_MISC_EVENTS.LBR_INSERTS	
EventSel=CCH, UMask=20H	Count cases of saving new LBR.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_4	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x4 , Precise	Loads with latency value being above 4 .

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_8	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x8 , Precise	Loads with latency value being above 8.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_16	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x10 , Precise	Loads with latency value being above 16.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_32	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x20 , Precise	Loads with latency value being above 32.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_64	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x40 , Precise	Loads with latency value being above 64.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_128	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x80 , Precise	Loads with latency value being above 128.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_256	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x100 , Precise	Loads with latency value being above 256.
MEM_TRANS_RETIRED.LOAD_LATENCY_GT_512	
EventSel=CDH, UMask=01H, MSR_PEBS_LD_LAT_THRESHOLD=0x200 , Precise	Loads with latency value being above 512.
MEM_TRANS_RETIRED.PRECISE_STORE	
EventSel=CDH, UMask=02H, Precise	Sample stores and collect precise store operation via PEBS record. PMC3 only. (Precise Event - PEBS).
MEM_UOPS_RETIRED.STLB_MISS_LOADS	
EventSel=D0H, UMask=11H, Precise	Retired load uops that miss the STLB.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
MEM_UOPS_RETIRED.STLB_MISS_STORES	
EventSel=D0H, UMask=12H, Precise	Retired store uops that miss the STLB.
MEM_UOPS_RETIRED.LOCK_LOADS	
EventSel=D0H, UMask=21H, Precise	Retired load uops with locked access.
MEM_UOPS_RETIRED.SPLIT_LOADS	
EventSel=D0H, UMask=41H, Precise	This event counts line-splitted load uops retired to the architected path. A line split is across 64B cache-line which includes a page split (4K).
MEM_UOPS_RETIRED.SPLIT_STORES	
EventSel=D0H, UMask=42H, Precise	This event counts line-splitted store uops retired to the architected path. A line split is across 64B cache-line which includes a page split (4K).
MEM_UOPS_RETIRED.ALL_LOADS	
EventSel=D0H, UMask=81H, Precise	This event counts the number of load uops retired.
MEM_UOPS_RETIRED.ALL_STORES	
EventSel=D0H, UMask=82H, Precise	This event counts the number of store uops retired.
MEM_LOAD_UOPS_RETIRED.L1_HIT	
EventSel=D1H, UMask=01H, Precise	Retired load uops with L1 cache hits as data sources.
MEM_LOAD_UOPS_RETIRED.L2_HIT	
EventSel=D1H, UMask=02H, Precise	Retired load uops with L2 cache hits as data sources.
MEM_LOAD_UOPS_RETIRED.LLC_HIT	
EventSel=D1H, UMask=04H, Precise	This event counts retired load uops that hit in the last-level (L3) cache without snoops required.
MEM_LOAD_UOPS_RETIRED.HIT_LFB	
EventSel=D1H, UMask=40H, Precise	Retired load uops which data sources were load uops missed L1 but hit FB due to preceding miss to the same cache line with data not ready.
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_MISS	
EventSel=D2H, UMask=01H, Precise	Retired load uops which data sources were LLC hit and cross-core snoop missed in on-pkg core cache.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_HIT	
EventSel=D2H, UMask=02H, Precise	This event counts retired load uops that hit in the last-level cache (L3) and were found in a non-modified state in a neighboring core's private cache (same package). Since the last level cache is inclusive, hits to the L3 may require snooping the private L2 caches of any cores on the same socket that have the line. In this case, a snoop was required, and another L2 had the line in a non-modified state.
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_HITM	
EventSel=D2H, UMask=04H, Precise	This event counts retired load uops that hit in the last-level cache (L3) and were found in a non-modified state in a neighboring core's private cache (same package). Since the last level cache is inclusive, hits to the L3 may require snooping the private L2 caches of any cores on the same socket that have the line. In this case, a snoop was required, and another L2 had the line in a modified state, so the line had to be invalidated in that L2 cache and transferred to the requesting L2.
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_NONE	
EventSel=D2H, UMask=08H, Precise	Retired load uops which data sources were hits in LLC without snoops required.
MEM_LOAD_UOPS_MISC_RETIRED.LLC_MISS	
EventSel=D4H, UMask=02H, Precise	This event counts retired demand loads that missed the last-level (L3) cache. This means that the load is usually satisfied from memory in a client system or possibly from the remote socket in a server. Demand loads are non speculative load uops.
BACLEARS.ANY	
EventSel=E6H, UMask=1FH	Counts the total number when the front end is resteered, mainly when the BPU cannot provide a correct prediction and this is corrected by other branch handling mechanisms at the front end.
L2_TRANS.DEMAND_DATA_RD	
EventSel=F0H, UMask=01H	Demand Data Read requests that access L2 cache.
L2_TRANS.RFO	
EventSel=F0H, UMask=02H	RFO requests that access L2 cache.
L2_TRANS.CODE_RD	
EventSel=F0H, UMask=04H	L2 cache accesses when fetching instructions.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
L2_TRANS.ALL_PF	
EventSel=F0H, UMask=08H	L2 or LLC HW prefetches that access L2 cache.
L2_TRANS.L1D_WB	
EventSel=F0H, UMask=10H	L1D writebacks that access L2 cache.
L2_TRANS.L2_FILL	
EventSel=F0H, UMask=20H	L2 fill requests that access L2 cache.
L2_TRANS.L2_WB	
EventSel=F0H, UMask=40H	L2 writebacks that access L2 cache.
L2_TRANS.ALL_REQUESTS	
EventSel=F0H, UMask=80H	Transactions accessing L2 pipe.
L2_LINES_IN.I	
EventSel=F1H, UMask=01H	L2 cache lines in I state filling L2.
L2_LINES_IN.S	
EventSel=F1H, UMask=02H	L2 cache lines in S state filling L2.
L2_LINES_IN.E	
EventSel=F1H, UMask=04H	L2 cache lines in E state filling L2.
L2_LINES_IN.ALL	
EventSel=F1H, UMask=07H	This event counts the number of L2 cache lines brought into the L2 cache. Lines are filled into the L2 cache when there was an L2 miss.
L2_LINES_OUT.DEMAND_CLEAN	
EventSel=F2H, UMask=01H	Clean L2 cache lines evicted by demand.
L2_LINES_OUT.DEMAND_DIRTY	
EventSel=F2H, UMask=02H	Dirty L2 cache lines evicted by demand.
L2_LINES_OUT.PF_CLEAN	
EventSel=F2H, UMask=04H	Clean L2 cache lines evicted by L2 prefetch.
L2_LINES_OUT.PF_DIRTY	
EventSel=F2H, UMask=08H	Dirty L2 cache lines evicted by L2 prefetch.

Table 8: Performance Events in the Processor Core Common to 2nd Generation Intel® Core™ i7-2xxx, Intel® Core™ i5-2xxx, Intel® Core™ i3-2xxx Processor Series and Intel® Xeon® Processors E3 and E5 Family (06_2AH, 06_2DH)

Event Name	
Configuration	Description
L2_LINES_OUT.DIRTY_ALL	
EventSel=F2H, UMask=0AH	Dirty L2 cache lines filling the L2.
SQ_MISC.SPLIT_LOCK	
EventSel=F4H, UMask=10H	Split locks in SQ.

Additional information on event specifics (e.g. derivative events using specific IA32_PERFEVTSELx modifiers, limitations, special notes and recommendations) can be found at https://software.intel.com/en-us/forums/software-tuning-performance-optimization-platform-monitoring

Performance Monitoring Events based on Westmere-EP-SP Microarchitecture

Intel 64 processors based on Intel® Microarchitecture code name Westmere support the performance-monitoring events listed in the table below.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
CPU_CLK_UNHALTED.REF	
Architectural, Fixed	Reference cycles when thread is not halted (fixed counter).
CPU_CLK_UNHALTED.THREAD	
Architectural, Fixed	Cycles when thread is not halted (fixed counter).
INST_RETIRED.ANY	
Architectural, Fixed	Instructions retired (fixed counter).
LOAD_BLOCK.OVERLAP_STORE	
EventSel=03H, UMask=02H	Loads that partially overlap an earlier store.
SB_DRAIN.ANY	
EventSel=04H, UMask=07H	All Store buffer stall cycles.
STORE_BLOCKS.AT_RET	
EventSel=06H, UMask=04H	Loads delayed with at-Retirement block code.
STORE_BLOCKS.L1D_BLOCK	
EventSel=06H, UMask=08H	Cacheable loads delayed with L1D block code.
PARTIAL_ADDRESS_ALIAS	
EventSel=07H, UMask=01H	False dependencies due to partial address aliasing.
DTLB_LOAD_MISSES.ANY	
EventSel=08H, UMask=01H	DTLB load misses.
DTLB_LOAD_MISSES.WALK_COMPLETED	
EventSel=08H, UMask=02H	DTLB load miss page walks complete.
DTLB_LOAD_MISSES.WALK_CYCLES	
EventSel=08H, UMask=04H	DTLB load miss page walk cycles.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
DTLB_LOAD_MISSES.STLB_HIT	
EventSel=08H, UMask=10H	DTLB second level hit.
DTLB_LOAD_MISSES.PDE_MISS	
EventSel=08H, UMask=20H	DTLB load miss caused by low part of address.
MEM_INST_RETIRED.LOADS	
EventSel=0BH, UMask=01H, Precise	Instructions retired which contains a load (Precise Event).
MEM_INST_RETIRED.STORES	
EventSel=0BH, UMask=02H, Precise	Instructions retired which contains a store (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_0	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x0 , Precise	Memory instructions retired above 0 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_1024	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x400 , Precise	Memory instructions retired above 1024 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_128	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x80 , Precise	Memory instructions retired above 128 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_16	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x10 , Precise	Memory instructions retired above 16 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_16384	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x4000 , Precise	Memory instructions retired above 16384 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_2048	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x800 , Precise	Memory instructions retired above 2048 clocks (Precise Event).

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_256	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x100 , Precise	Memory instructions retired above 256 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_32	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x20 , Precise	Memory instructions retired above 32 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_32768	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x8000 , Precise	Memory instructions retired above 32768 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_4	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x4 , Precise	Memory instructions retired above 4 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_4096	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x1000 , Precise	Memory instructions retired above 4096 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_512	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x200 , Precise	Memory instructions retired above 512 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_64	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x40 , Precise	Memory instructions retired above 64 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_8	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x8 , Precise	Memory instructions retired above 8 clocks (Precise Event).

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_8192	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x2000 , Precise	Memory instructions retired above 8192 clocks (Precise Event).
MEM_STORE_RETIRED.DTLB_MISS	
EventSel=0CH, UMask=01H, Precise	Retired stores that miss the DTLB (Precise Event).
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=01H	Uops issued.
UOPS_ISSUED.CORE_STALL_CYCLES	
EventSel=0EH, UMask=01H, AnyThread=1, Invert=1, CMask=1	Cycles no Uops were issued on any thread.
UOPS_ISSUED.CYCLES_ALL_THREADS	
EventSel=0EH, UMask=01H, AnyThread=1, CMask=1	Cycles Uops were issued on either thread.
UOPS_ISSUED.STALL_CYCLES	
EventSel=0EH, UMask=01H, Invert=1, CMask=1	Cycles no Uops were issued.
UOPS_ISSUED.FUSED	
EventSel=0EH, UMask=02H	Fused Uops issued.
MEM_UNCORE_RETIRED.OTHER_CORE_L2_HITM	
EventSel=0FH, UMask=02H, Precise	Load instructions retired that HIT modified data in sibling core (Precise Event).
MEM_UNCORE_RETIRED.REMOTE_CACHE_LOCAL_HOME_HIT	
EventSel=0FH, UMask=08H, Precise	Load instructions retired remote cache HIT data source (Precise Event).
MEM_UNCORE_RETIRED.LOCAL_DRAM	
EventSel=0FH, UMask=10H, Precise	Load instructions retired with a data source of local DRAM or locally homed remote hitm (Precise Event).
MEM_UNCORE_RETIRED.REMOTE_DRAM	
EventSel=0FH, UMask=20H, Precise	Load instructions retired remote DRAM and remote home-remote cache HITM (Precise Event).

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
MEM_UNCORE_RETIRED.UNCACHEABLE	
EventSel=0FH, UMask=80H, Precise	Load instructions retired IO (Precise Event).
FP_COMP_OPS_EXE.X87	
EventSel=10H, UMask=01H	Computational floating-point operations executed.
FP_COMP_OPS_EXE.MMX	
EventSel=10H, UMask=02H	MMX Uops.
FP_COMP_OPS_EXE.SSE_FP	
EventSel=10H, UMask=04H	SSE and SSE2 FP Uops.
FP_COMP_OPS_EXE.SSE2_INTEGER	
EventSel=10H, UMask=08H	SSE2 integer Uops.
FP_COMP_OPS_EXE.SSE_FP_PACKED	
EventSel=10H, UMask=10H	SSE FP packed Uops.
FP_COMP_OPS_EXE.SSE_FP_SCALAR	
EventSel=10H, UMask=20H	SSE FP scalar Uops.
FP_COMP_OPS_EXE.SSE_SINGLE_PRECISION	
EventSel=10H, UMask=40H	SSE* FP single precision Uops.
FP_COMP_OPS_EXE.SSE_DOUBLE_PRECISION	
EventSel=10H, UMask=80H	SSE* FP double precision Uops.
SIMD_INT_128.PACKED_MPY	
EventSel=12H, UMask=01H	128 bit SIMD integer multiply operations.
SIMD_INT_128.PACKED_SHIFT	
EventSel=12H, UMask=02H	128 bit SIMD integer shift operations.
SIMD_INT_128.PACK	
EventSel=12H, UMask=04H	128 bit SIMD integer pack operations.
SIMD_INT_128.UNPACK	
EventSel=12H, UMask=08H	128 bit SIMD integer unpack operations.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
SIMD_INT_128.PACKED_LOGICAL	
EventSel=12H, UMask=10H	128 bit SIMD integer logical operations.
SIMD_INT_128.PACKED_ARITH	
EventSel=12H, UMask=20H	128 bit SIMD integer arithmetic operations.
SIMD_INT_128.SHUFFLE_MOVE	
EventSel=12H, UMask=40H	128 bit SIMD integer shuffle/move operations.
LOAD_DISPATCH.RS	
EventSel=13H, UMask=01H	Loads dispatched that bypass the MOB.
LOAD_DISPATCH.RS_DELAYED	
EventSel=13H, UMask=02H	Loads dispatched from stage 305.
LOAD_DISPATCH.MOB	
EventSel=13H, UMask=04H	Loads dispatched from the MOB.
LOAD_DISPATCH.ANY	
EventSel=13H, UMask=07H	All loads dispatched.
ARITH.CYCLES_DIV_BUSY	
EventSel=14H, UMask=01H	Cycles the divider is busy.
ARITH.DIV	
EventSel=14H, UMask=01H, EdgeDetect=1, Invert=1, CMask=1	Divide Operations executed.
ARITH.MUL	
EventSel=14H, UMask=02H	Multiply operations executed.
INST_QUEUE_WRITES	
EventSel=17H, UMask=01H	Instructions written to instruction queue.
INST_DECODED.DEC0	
EventSel=18H, UMask=01H	Instructions that must be decoded by decoder 0.
TWO_UOP_INSTS_DECODED	
EventSel=19H, UMask=01H	Two Uop instructions decoded.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
INST_QUEUE_WRITE_CYCLES	
EventSel=1EH, UMask=01H	Cycles instructions are written to the instruction queue.
LSD_OVERFLOW	
EventSel=20H, UMask=01H	Loops that can't stream from the instruction queue.
L2_RQSTS.LD_HIT	
EventSel=24H, UMask=01H	L2 load hits.
L2_RQSTS.LD_MISS	
EventSel=24H, UMask=02H	L2 load misses.
L2_RQSTS.LOADS	
EventSel=24H, UMask=03H	L2 requests.
L2_RQSTS.RFO_HIT	
EventSel=24H, UMask=04H	L2 RFO hits.
L2_RQSTS.RFO_MISS	
EventSel=24H, UMask=08H	L2 RFO misses.
L2_RQSTS.RFOS	
EventSel=24H, UMask=0CH	L2 RFO requests.
L2_RQSTS.IFETCH_HIT	
EventSel=24H, UMask=10H	L2 instruction fetch hits.
L2_RQSTS.IFETCH_MISS	
EventSel=24H, UMask=20H	L2 instruction fetch misses.
L2_RQSTS.IFETCHES	
EventSel=24H, UMask=30H	L2 instruction fetches.
L2_RQSTS.PREFETCH_HIT	
EventSel=24H, UMask=40H	L2 prefetch hits.
L2_RQSTS.PREFETCH_MISS	
EventSel=24H, UMask=80H	L2 prefetch misses.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
L2_RQSTS.MISS	
EventSel=24H, UMask=AAH	All L2 misses.
L2_RQSTS.PREFETCHES	
EventSel=24H, UMask=C0H	All L2 prefetches.
L2_RQSTS.REFERENCES	
EventSel=24H, UMask=FFH	All L2 requests.
L2_DATA_RQSTS.DEMAND.I_STATE	
EventSel=26H, UMask=01H	L2 data demand loads in I state (misses).
L2_DATA_RQSTS.DEMAND.S_STATE	
EventSel=26H, UMask=02H	L2 data demand loads in S state.
L2_DATA_RQSTS.DEMAND.E_STATE	
EventSel=26H, UMask=04H	L2 data demand loads in E state.
L2_DATA_RQSTS.DEMAND.M_STATE	
EventSel=26H, UMask=08H	L2 data demand loads in M state.
L2_DATA_RQSTS.DEMAND.MESI	
EventSel=26H, UMask=0FH	L2 data demand requests.
L2_DATA_RQSTS.PREFETCH.I_STATE	
EventSel=26H, UMask=10H	L2 data prefetches in the I state (misses).
L2_DATA_RQSTS.PREFETCH.S_STATE	
EventSel=26H, UMask=20H	L2 data prefetches in the S state.
L2_DATA_RQSTS.PREFETCH.E_STATE	
EventSel=26H, UMask=40H	L2 data prefetches in E state.
L2_DATA_RQSTS.PREFETCH.M_STATE	
EventSel=26H, UMask=80H	L2 data prefetches in M state.
L2_DATA_RQSTS.PREFETCH.MESI	
EventSel=26H, UMask=F0H	All L2 data prefetches.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
L2_DATA_RQSTS.ANY	
EventSel=26H, UMask=FFH	All L2 data requests.
L2_WRITE.RFO.I_STATE	
EventSel=27H, UMask=01H	L2 demand store RFOs in I state (misses).
L2_WRITE.RFO.S_STATE	
EventSel=27H, UMask=02H	L2 demand store RFOs in S state.
L2_WRITE.RFO.M_STATE	
EventSel=27H, UMask=08H	L2 demand store RFOs in M state.
L2_WRITE.RFO.HIT	
EventSel=27H, UMask=0EH	All L2 demand store RFOs that hit the cache.
L2_WRITE.RFO.MESI	
EventSel=27H, UMask=0FH	All L2 demand store RFOs.
L2_WRITE.LOCK.I_STATE	
EventSel=27H, UMask=10H	L2 demand lock RFOs in I state (misses).
L2_WRITE.LOCK.S_STATE	
EventSel=27H, UMask=20H	L2 demand lock RFOs in S state.
L2_WRITE.LOCK.E_STATE	
EventSel=27H, UMask=40H	L2 demand lock RFOs in E state.
L2_WRITE.LOCK.M_STATE	
EventSel=27H, UMask=80H	L2 demand lock RFOs in M state.
L2_WRITE.LOCK.HIT	
EventSel=27H, UMask=E0H	All demand L2 lock RFOs that hit the cache.
L2_WRITE.LOCK.MESI	
EventSel=27H, UMask=F0H	All demand L2 lock RFOs.
L1D_WB_L2.I_STATE	
EventSel=28H, UMask=01H	L1 writebacks to L2 in I state (misses).

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
L1D_WB_L2.S_STATE	
EventSel=28H, UMask=02H	L1 writebacks to L2 in S state.
L1D_WB_L2.E_STATE	
EventSel=28H, UMask=04H	L1 writebacks to L2 in E state.
L1D_WB_L2.M_STATE	
EventSel=28H, UMask=08H	L1 writebacks to L2 in M state.
L1D_WB_L2.MESI	
EventSel=28H, UMask=0FH	All L1 writebacks to L2.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	Longest latency cache miss.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	Longest latency cache reference.
CPU_CLK_UNHALTED.THREAD_P	
EventSel=3CH, UMask=00H, Architectural	Cycles when thread is not halted (programmable counter).
CPU_CLK_UNHALTED.TOTAL_CYCLES	
EventSel=3CH, UMask=00H, Invert=1, CMask=2, Architectural	Total CPU cycles.
CPU_CLK_UNHALTED.REF_P	
EventSel=3CH, UMask=01H, Architectural	Reference base clock (133 Mhz) cycles when thread is not halted (programmable counter).
DTLB_MISSES.ANY	
EventSel=49H, UMask=01H	DTLB misses.
DTLB_MISSES.WALK_COMPLETED	
EventSel=49H, UMask=02H	DTLB miss page walks.
DTLB_MISSES.WALK_CYCLES	
EventSel=49H, UMask=04H	DTLB miss page walk cycles.
DTLB_MISSES.STLB_HIT	
EventSel=49H, UMask=10H	DTLB first level misses but second level hit.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
DTLB_MISSES.LARGE_WALK_COMPLETED	
EventSel=49H, UMask=80H	DTLB miss large page walks.
LOAD_HIT_PRE	
EventSel=4CH, UMask=01H	Load operations conflicting with software prefetches.
L1D_PREFETCH.REQUESTS	
EventSel=4EH, UMask=01H	L1D hardware prefetch requests.
L1D_PREFETCH.MISS	
EventSel=4EH, UMask=02H	L1D hardware prefetch misses.
L1D_PREFETCH.TRIGGERS	
EventSel=4EH, UMask=04H	L1D hardware prefetch requests triggered.
EPT.WALK_CYCLES	
EventSel=4FH, UMask=10H	Extended Page Table walk cycles.
L1D.REPL	
EventSel=51H, UMask=01H	L1 data cache lines allocated.
L1D.M_REPL	
EventSel=51H, UMask=02H	L1D cache lines allocated in the M state.
L1D.M_EVICT	
EventSel=51H, UMask=04H	L1D cache lines replaced in M state.
L1D.M_SNOOP_EVICT	
EventSel=51H, UMask=08H	L1D snoop eviction of cache lines in M state.
L1D_CACHE_PREFETCH_LOCK_FB_HIT	
EventSel=52H, UMask=01H	L1D prefetch load lock accepted in fill buffer.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.READ_DATA	
EventSel=60H, UMask=01H	Outstanding offcore demand data reads.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.READ_DATA_NOT_EMPTY	
EventSel=60H, UMask=01H, CMask=1	Cycles offcore demand data read busy.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.READ_CODE	
EventSel=60H, UMask=02H	Outstanding offcore demand code reads.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.READ_CODE_NOT_EMPTY	
EventSel=60H, UMask=02H, CMask=1	Cycles offcore demand code read busy.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.RFO	
EventSel=60H, UMask=04H	Outstanding offcore demand RFOs.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.RFO_NOT_EMPTY	
EventSel=60H, UMask=04H, CMask=1	Cycles offcore demand RFOs busy.
OFFCORE_REQUESTS_OUTSTANDING.ANY.READ	
EventSel=60H, UMask=08H	Outstanding offcore reads.
OFFCORE_REQUESTS_OUTSTANDING.ANY.READ_NOT_EMPTY	
EventSel=60H, UMask=08H, CMask=1	Cycles offcore reads busy.
CACHE_LOCK_CYCLES.L1D_L2	
EventSel=63H, UMask=01H	Cycles L1D and L2 locked.
CACHE_LOCK_CYCLES.L1D	
EventSel=63H, UMask=02H	Cycles L1D locked.
IO_TRANSACTIONS	
EventSel=6CH, UMask=01H	I/O transactions.
L1I.HITS	
EventSel=80H, UMask=01H	L1I instruction fetch hits.
L1I.MISSES	
EventSel=80H, UMask=02H	L1I instruction fetch misses.
L1I.READS	
EventSel=80H, UMask=03H	L1I Instruction fetches.
L1I.CYCLES_STALLED	
EventSel=80H, UMask=04H	L1I instruction fetch stall cycles.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
LARGE_ITLB.HIT	
EventSel=82H, UMask=01H	Large ITLB hit.
ITLB_MISSES.ANY	
EventSel=85H, UMask=01H	ITLB miss.
ITLB_MISSES.WALK_COMPLETED	
EventSel=85H, UMask=02H	ITLB miss page walks.
ITLB_MISSES.WALK_CYCLES	
EventSel=85H, UMask=04H	ITLB miss page walk cycles.
ILD_STALL.LCP	
EventSel=87H, UMask=01H	Length Change Prefix stall cycles.
ILD_STALL.MRU	
EventSel=87H, UMask=02H	Stall cycles due to BPU MRU bypass.
ILD_STALL.IQ_FULL	
EventSel=87H, UMask=04H	Instruction Queue full stall cycles.
ILD_STALL.REGEN	
EventSel=87H, UMask=08H	Regen stall cycles.
ILD_STALL.ANY	
EventSel=87H, UMask=0FH	Any Instruction Length Decoder stall cycles.
BR_INST_EXEC.COND	
EventSel=88H, UMask=01H	Conditional branch instructions executed.
BR_INST_EXEC.DIRECT	
EventSel=88H, UMask=02H	Unconditional branches executed.
BR_INST_EXEC.INDIRECT_NON_CALL	
EventSel=88H, UMask=04H	Indirect non call branches executed.
BR_INST_EXEC.NON_CALLS	
EventSel=88H, UMask=07H	All non call branches executed.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
BR_INST_EXEC.RETURN_NEAR	
EventSel=88H, UMask=08H	Indirect return branches executed.
BR_INST_EXEC.DIRECT_NEAR_CALL	
EventSel=88H, UMask=10H	Unconditional call branches executed.
BR_INST_EXEC.INDIRECT_NEAR_CALL	
EventSel=88H, UMask=20H	Indirect call branches executed.
BR_INST_EXEC.NEAR_CALLS	
EventSel=88H, UMask=30H	Call branches executed.
BR_INST_EXEC.TAKEN	
EventSel=88H, UMask=40H	Taken branches executed.
BR_INST_EXEC.ANY	
EventSel=88H, UMask=7FH	Branch instructions executed.
BR_MISP_EXEC.COND	
EventSel=89H, UMask=01H	Mispredicted conditional branches executed.
BR_MISP_EXEC.DIRECT	
EventSel=89H, UMask=02H	Mispredicted unconditional branches executed.
BR_MISP_EXEC.INDIRECT_NON_CALL	
EventSel=89H, UMask=04H	Mispredicted indirect non call branches executed.
BR_MISP_EXEC.NON_CALLS	
EventSel=89H, UMask=07H	Mispredicted non call branches executed.
BR_MISP_EXEC.RETURN_NEAR	
EventSel=89H, UMask=08H	Mispredicted return branches executed.
BR_MISP_EXEC.DIRECT_NEAR_CALL	
EventSel=89H, UMask=10H	Mispredicted non call branches executed.
BR_MISP_EXEC.INDIRECT_NEAR_CALL	
EventSel=89H, UMask=20H	Mispredicted indirect call branches executed.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
BR_MISP_EXEC.NEAR_CALLS	
EventSel=89H, UMask=30H	Mispredicted call branches executed.
BR_MISP_EXEC.TAKEN	
EventSel=89H, UMask=40H	Mispredicted taken branches executed.
BR_MISP_EXEC.ANY	
EventSel=89H, UMask=7FH	Mispredicted branches executed.
RESOURCE_STALLS.ANY	
EventSel=A2H, UMask=01H	Resource related stall cycles.
RESOURCE_STALLS.LOAD	
EventSel=A2H, UMask=02H	Load buffer stall cycles.
RESOURCE_STALLS.RS_FULL	
EventSel=A2H, UMask=04H	Reservation Station full stall cycles.
RESOURCE_STALLS.STORE	
EventSel=A2H, UMask=08H	Store buffer stall cycles.
RESOURCE_STALLS.ROB_FULL	
EventSel=A2H, UMask=10H	ROB full stall cycles.
RESOURCE_STALLS.FPCW	
EventSel=A2H, UMask=20H	FPU control word write stall cycles.
RESOURCE_STALLS.MXCSR	
EventSel=A2H, UMask=40H	MXCSR rename stall cycles.
RESOURCE_STALLS.OTHER	
EventSel=A2H, UMask=80H	Other Resource related stall cycles.
MACRO_INSTS.FUSIONS_DECODED	
EventSel=A6H, UMask=01H	Macro-fused instructions decoded.
BACLEAR_FORCE_IQ	
EventSel=A7H, UMask=01H	Instruction queue forced BACLEAR.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
LSD.ACTIVE	
EventSel=A8H, UMask=01H, CMask=1	Cycles when uops were delivered by the LSD.
LSD.INACTIVE	
EventSel=A8H, UMask=01H, Invert=1, CMask=1	Cycles no uops were delivered by the LSD.
ITLB_FLUSH	
EventSel=AEH, UMask=01H	ITLB flushes.
OFFCORE_REQUESTS.DEMAND.READ_DATA	
EventSel=B0H, UMask=01H	Offcore demand data read requests.
OFFCORE_REQUESTS.DEMAND.READ_CODE	
EventSel=B0H, UMask=02H	Offcore demand code read requests.
OFFCORE_REQUESTS.DEMAND.RFO	
EventSel=B0H, UMask=04H	Offcore demand RFO requests.
OFFCORE_REQUESTS.ANY.READ	
EventSel=B0H, UMask=08H	Offcore read requests.
OFFCORE_REQUESTS.ANY.RFO	
EventSel=B0H, UMask=10H	Offcore RFO requests.
OFFCORE_REQUESTS.UNCACHED_MEM	
EventSel=B0H, UMask=20H	Offcore uncached memory accesses.
OFFCORE_REQUESTS.L1D_WRITEBACK	
EventSel=B0H, UMask=40H	Offcore L1 data cache writebacks.
OFFCORE_REQUESTS.ANY	
EventSel=B0H, UMask=80H	All offcore requests.
UOPS_EXECUTED.PORT0	
EventSel=B1H, UMask=01H	Uops executed on port 0.
UOPS_EXECUTED.PORT1	
EventSel=B1H, UMask=02H	Uops executed on port 1.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
UOPS_EXECUTED.PORT2_CORE	
EventSel=B1H, UMask=04H, AnyThread=1	Uops executed on port 2 (core count).
UOPS_EXECUTED.PORT3_CORE	
EventSel=B1H, UMask=08H, AnyThread=1	Uops executed on port 3 (core count).
UOPS_EXECUTED.PORT4_CORE	
EventSel=B1H, UMask=10H, AnyThread=1	Uops executed on port 4 (core count).
UOPS_EXECUTED.CORE_ACTIVE_CYCLES_NO_PORT5	
EventSel=B1H, UMask=1FH, AnyThread=1, CMask=1	Cycles Uops executed on ports 0-4 (core count).
UOPS_EXECUTED.CORE_STALL_COUNT_NO_PORT5	
EventSel=B1H, UMask=1FH, EdgeDetect=1, AnyThread=1, Invert=1, CMask=1	Uops executed on ports 0-4 (core count).
UOPS_EXECUTED.CORE_STALL_CYCLES_NO_PORT5	
EventSel=B1H, UMask=1FH, AnyThread=1, Invert=1, CMask=1	Cycles no Uops issued on ports 0-4 (core count).
UOPS_EXECUTED.PORT5	
EventSel=B1H, UMask=20H	Uops executed on port 5.
UOPS_EXECUTED.CORE_ACTIVE_CYCLES	
EventSel=B1H, UMask=3FH, AnyThread=1, CMask=1	Cycles Uops executed on any port (core count).
UOPS_EXECUTED.CORE_STALL_COUNT	
EventSel=B1H, UMask=3FH, EdgeDetect=1, AnyThread=1, Invert=1, CMask=1	Uops executed on any port (core count).
UOPS_EXECUTED.CORE_STALL_CYCLES	
EventSel=B1H, UMask=3FH, AnyThread=1, Invert=1, CMask=1	Cycles no Uops issued on any port (core count).
UOPS_EXECUTED.PORT015	
EventSel=B1H, UMask=40H	Uops issued on ports 0, 1 or 5.
UOPS_EXECUTED.PORT015_STALL_CYCLES	
EventSel=B1H, UMask=40H, Invert=1, CMask=1	Cycles no Uops issued on ports 0, 1 or 5.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
UOPS_EXECUTED.PORT234_CORE	
EventSel=B1H, UMask=80H, AnyThread=1	Uops issued on ports 2, 3 or 4.
OFFCORE_REQUESTS_SQ_FULL	
EventSel=B2H, UMask=01H	Offcore requests blocked due to Super Queue full.
SNOOPQ_REQUESTS_OUTSTANDING.DATA	
EventSel=B3H, UMask=01H	Outstanding snoop data requests.
SNOOPQ_REQUESTS_OUTSTANDING.DATA_NOT_EMPTY	
EventSel=B3H, UMask=01H, CMask=1	Cycles snoop data requests queued.
SNOOPQ_REQUESTS_OUTSTANDING.INVALIDATE	
EventSel=B3H, UMask=02H	Outstanding snoop invalidate requests.
SNOOPQ_REQUESTS_OUTSTANDING.INVALIDATE_NOT_EMPTY	
EventSel=B3H, UMask=02H, CMask=1	Cycles snoop invalidate requests queued.
SNOOPQ_REQUESTS_OUTSTANDING.CODE	
EventSel=B3H, UMask=04H	Outstanding snoop code requests.
SNOOPQ_REQUESTS_OUTSTANDING.CODE_NOT_EMPTY	
EventSel=B3H, UMask=04H, CMask=1	Cycles snoop code requests queued.
SNOOPQ_REQUESTS.DATA	
EventSel=B4H, UMask=01H	Snoop data requests.
SNOOPQ_REQUESTS.INVALIDATE	
EventSel=B4H, UMask=02H	Snoop invalidate requests.
SNOOPQ_REQUESTS.CODE	
EventSel=B4H, UMask=04H	Snoop code requests.
SNOOP_RESPONSE.HIT	
EventSel=B8H, UMask=01H	Thread responded HIT to snoop.
SNOOP_RESPONSE.HITE	
EventSel=B8H, UMask=02H	Thread responded HITE to snoop.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
SNOOP_RESPONSE.HITM	
EventSel=B8H, UMask=04H	Thread responded HITM to snoop.
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=01H, Precise	Instructions retired (Programmable counter and Precise Event).
INST_RETIRED.TOTAL_CYCLES	
EventSel=C0H, UMask=01H, Invert=1, CMask=16, Precise	Total cycles (Precise Event).
INST_RETIRED.X87	
EventSel=C0H, UMask=02H, Precise	Retired floating-point operations (Precise Event).
INST_RETIRED.MMX	
EventSel=C0H, UMask=04H, Precise	Retired MMX instructions (Precise Event).
UOPS_RETIRED.ACTIVE_CYCLES	
EventSel=C2H, UMask=01H, CMask=1, Precise	Cycles Uops are being retired.
UOPS_RETIRED.ANY	
EventSel=C2H, UMask=01H, Precise	Uops retired (Precise Event).
UOPS_RETIRED.STALL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=1, Precise	Cycles Uops are not retiring (Precise Event).
UOPS_RETIRED.TOTAL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=16, Precise	Total cycles using precise uop retired event (Precise Event).
UOPS_RETIRED.RETIRE_SLOTS	
EventSel=C2H, UMask=02H, Precise	Retirement slots used (Precise Event).
UOPS_RETIRED.MACRO_FUSED	
EventSel=C2H, UMask=04H, Precise	Macro-fused Uops retired (Precise Event).
MACHINE_CLEARS.CYCLES	
EventSel=C3H, UMask=01H	Cycles machine clear asserted.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
MACHINE_CLEARS.MEM_ORDER	
EventSel=C3H, UMask=02H	Execution pipeline restart due to Memory ordering conflicts.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=04H	Self-Modifying Code detected.
BR_INST_RETIRED.CONDITIONAL	
EventSel=C4H, UMask=01H, Precise	Retired conditional branch instructions (Precise Event).
BR_INST_RETIRED.NEAR_CALL	
EventSel=C4H, UMask=02H, Precise	Retired near call instructions (Precise Event).
BR_INST_RETIRED.NEAR_CALL_R3	
EventSel=C4H, UMask=02H, USR=1,OS=0, Precise	Retired near call instructions Ring 3 only(Precise Event).
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=04H, Precise	Retired branch instructions (Precise Event).
BR_MISP_RETIRED.CONDITIONAL	
EventSel=C5H, UMask=01H, Precise	Mispredicted conditional retired branches (Precise Event).
BR_MISP_RETIRED.NEAR_CALL	
EventSel=C5H, UMask=02H, Precise	Mispredicted near retired calls (Precise Event).
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=04H, Precise	Mispredicted retired branch instructions (Precise Event).
SSEX_UOPS_RETIRED.PACKED_SINGLE	
EventSel=C7H, UMask=01H, Precise	SIMD Packed-Single Uops retired (Precise Event).
SSEX_UOPS_RETIRED.SCALAR_SINGLE	
EventSel=C7H, UMask=02H, Precise	SIMD Scalar-Single Uops retired (Precise Event).
SSEX_UOPS_RETIRED.PACKED_DOUBLE	
EventSel=C7H, UMask=04H, Precise	SIMD Packed-Double Uops retired (Precise Event).
SSEX_UOPS_RETIRED.SCALAR_DOUBLE	
EventSel=C7H, UMask=08H, Precise	SIMD Scalar-Double Uops retired (Precise Event).

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
SSEX_UOPS_RETIRED.VECTOR_INTEGER	
EventSel=C7H, UMask=10H, Precise	SIMD Vector Integer Uops retired (Precise Event).
ITLB_MISS_RETIRED	
EventSel=C8H, UMask=20H, Precise	Retired instructions that missed the ITLB (Precise Event).
MEM_LOAD_RETIRED.L1D_HIT	
EventSel=CBH, UMask=01H, Precise	Retired loads that hit the L1 data cache (Precise Event).
MEM_LOAD_RETIRED.L2_HIT	
EventSel=CBH, UMask=02H, Precise	Retired loads that hit the L2 cache (Precise Event).
MEM_LOAD_RETIRED.LLC_UNSHARED_HIT	
EventSel=CBH, UMask=04H, Precise	Retired loads that hit valid versions in the LLC cache (Precise Event).
MEM_LOAD_RETIRED.OTHER_CORE_L2_HIT_HITM	
EventSel=CBH, UMask=08H, Precise	Retired loads that hit sibling core's L2 in modified or unmodified states (Precise Event).
MEM_LOAD_RETIRED.LLC_MISS	
EventSel=CBH, UMask=10H, Precise	Retired loads that miss the LLC cache (Precise Event).
MEM_LOAD_RETIRED.HIT_LFB	
EventSel=CBH, UMask=40H, Precise	Retired loads that miss L1D and hit an previously allocated LFB (Precise Event).
MEM_LOAD_RETIRED.DTLB_MISS	
EventSel=CBH, UMask=80H, Precise	Retired loads that miss the DTLB (Precise Event).
FP_MMX_TRANS.TO_FP	
EventSel=CCH, UMask=01H	Transitions from MMX to Floating Point instructions.
FP_MMX_TRANS.TO_MMX	
EventSel=CCH, UMask=02H	Transitions from Floating Point to MMX instructions.
FP_MMX_TRANS.ANY	
EventSel=CCH, UMask=03H	All Floating Point to and from MMX transitions.
MACRO_INSTS.DECODED	
EventSel=D0H, UMask=01H	Instructions decoded.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
UOPS_DECODED.STALL_CYCLES	
EventSel=D1H, UMask=01H, Invert=1, CMask=1	Cycles no Uops are decoded.
UOPS_DECODED.MS_CYCLES_ACTIVE	
EventSel=D1H, UMask=02H, CMask=1	Uops decoded by Microcode Sequencer.
UOPS_DECODED.ESP_FOLDING	
EventSel=D1H, UMask=04H	Stack pointer instructions decoded.
UOPS_DECODED.ESP_SYNC	
EventSel=D1H, UMask=08H	Stack pointer sync operations.
RAT_STALLS.FLAGS	
EventSel=D2H, UMask=01H	Flag stall cycles.
RAT_STALLS.REGISTERS	
EventSel=D2H, UMask=02H	Partial register stall cycles.
RAT_STALLS.ROB_READ_PORT	
EventSel=D2H, UMask=04H	ROB read port stalls cycles.
RAT_STALLS.SCOREBOARD	
EventSel=D2H, UMask=08H	Scoreboard stall cycles.
RAT_STALLS.ANY	
EventSel=D2H, UMask=0FH	All RAT stall cycles.
SEG_RENAME_STALLS	
EventSel=D4H, UMask=01H	Segment rename stall cycles.
ES_REG_RENAMES	
EventSel=D5H, UMask=01H	ES segment renames.
UOP_UNFUSION	
EventSel=DBH, UMask=01H	Uop unfusions due to FP exceptions.
BR_INST_DECODED	
EventSel=E0H, UMask=01H	Branch instructions decoded.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
BPU_MISSED_CALL_RET	
EventSel=E5H, UMask=01H	Branch prediction unit missed call or return.
BACLEAR.CLEAR	
EventSel=E6H, UMask=01H	BACLEAR asserted, regardless of cause .
BACLEAR.BAD_TARGET	
EventSel=E6H, UMask=02H	BACLEAR asserted with bad target address.
BPU_CLEARS.EARLY	
EventSel=E8H, UMask=01H	Early Branch Prediciton Unit clears.
BPU_CLEARS.LATE	
EventSel=E8H, UMask=02H	Late Branch Prediction Unit clears.
L2_TRANSACTIONS.LOAD	
EventSel=F0H, UMask=01H	L2 Load transactions.
L2_TRANSACTIONS.RFO	
EventSel=F0H, UMask=02H	L2 RFO transactions.
L2_TRANSACTIONS.IFETCH	
EventSel=F0H, UMask=04H	L2 instruction fetch transactions.
L2_TRANSACTIONS.PREFETCH	
EventSel=F0H, UMask=08H	L2 prefetch transactions.
L2_TRANSACTIONS.L1D_WB	
EventSel=F0H, UMask=10H	L1D writeback to L2 transactions.
L2_TRANSACTIONS.FILL	
EventSel=F0H, UMask=20H	L2 fill transactions.
L2_TRANSACTIONS.WB	
EventSel=F0H, UMask=40H	L2 writeback to LLC transactions.
L2_TRANSACTIONS.ANY	
EventSel=F0H, UMask=80H	All L2 transactions.

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
L2_LINES_IN.S_STATE	
EventSel=F1H, UMask=02H	L2 lines allocated in the S state.
L2_LINES_IN.E_STATE	
EventSel=F1H, UMask=04H	L2 lines allocated in the E state.
L2_LINES_IN.ANY	
EventSel=F1H, UMask=07H	L2 lines alloacated.
L2_LINES_OUT.DEMAND_CLEAN	
EventSel=F2H, UMask=01H	L2 lines evicted by a demand request.
L2_LINES_OUT.DEMAND_DIRTY	
EventSel=F2H, UMask=02H	L2 modified lines evicted by a demand request.
L2_LINES_OUT.PREFETCH_CLEAN	
EventSel=F2H, UMask=04H	L2 lines evicted by a prefetch request.
L2_LINES_OUT.PREFETCH_DIRTY	
EventSel=F2H, UMask=08H	L2 modified lines evicted by a prefetch request.
L2_LINES_OUT.ANY	
EventSel=F2H, UMask=0FH	L2 lines evicted.
SQ_MISC.LRU_HINTS	
EventSel=F4H, UMask=04H	Super Queue LRU hints sent to LLC.
SQ_MISC.SPLIT_LOCK	
EventSel=F4H, UMask=10H	Super Queue lock splits across a cache line.
SQ_FULL_STALL_CYCLES	
EventSel=F6H, UMask=01H	Super Queue full stall cycles.
FP_ASSIST.ALL	
EventSel=F7H, UMask=01H, Precise	X87 Floating point assists (Precise Event).
FP_ASSIST.OUTPUT	
EventSel=F7H, UMask=02H, Precise	X87 Floating point assists for invalid output value (Precise Event).

Table 9: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture

Event Name	
Configuration	Description
FP_ASSIST.INPUT	
EventSel=F7H, UMask=04H, Precise	X87 Floating poiint assists for invalid input value (Precise Event).
SIMD_INT_64.PACKED_MPY	
EventSel=FDH, UMask=01H	SIMD integer 64 bit packed multiply operations.
SIMD_INT_64.PACKED_SHIFT	
EventSel=FDH, UMask=02H	SIMD integer 64 bit shift operations.
SIMD_INT_64.PACK	
EventSel=FDH, UMask=04H	SIMD integer 64 bit pack operations.
SIMD_INT_64.UNPACK	
EventSel=FDH, UMask=08H	SIMD integer 64 bit unpack operations.
SIMD_INT_64.PACKED_LOGICAL	
EventSel=FDH, UMask=10H	SIMD integer 64 bit logical operations.
SIMD_INT_64.PACKED_ARITH	
EventSel=FDH, UMask=20H	SIMD integer 64 bit arithmetic operations.
SIMD_INT_64.SHUFFLE_MOVE	
EventSel=FDH, UMask=40H	SIMD integer 64 bit shuffle/move operations.

Performance Monitoring Events based on Westmere-EP-DP Microarchitecture

Intel 64 processors based on Intel® Microarchitecture code name Westmere support the performance-monitoring events listed in the table below.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
CPU_CLK_UNHALTED.REF	
Architectural, Fixed	Reference cycles when thread is not halted (fixed counter).
CPU_CLK_UNHALTED.THREAD	
Architectural, Fixed	Cycles when thread is not halted (fixed counter).
INST_RETIRED.ANY	
Architectural, Fixed	Instructions retired (fixed counter).
LOAD_BLOCK.OVERLAP_STORE	
EventSel=03H, UMask=02H	Loads that partially overlap an earlier store.
SB_DRAIN.ANY	
EventSel=04H, UMask=07H	All Store buffer stall cycles.
MISALIGN_MEM_REF.STORE	
EventSel=05H, UMask=02H	Misaligned store references.
STORE_BLOCKS.AT_RET	
EventSel=06H, UMask=04H	Loads delayed with at-Retirement block code.
STORE_BLOCKS.L1D_BLOCK	
EventSel=06H, UMask=08H	Cacheable loads delayed with L1D block code.
PARTIAL_ADDRESS_ALIAS	
EventSel=07H, UMask=01H	False dependencies due to partial address aliasing.
DTLB_LOAD_MISSES.ANY	
EventSel=08H, UMask=01H	DTLB load misses.
DTLB_LOAD_MISSES.WALK_COMPLETED	
EventSel=08H, UMask=02H	DTLB load miss page walks complete.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
DTLB_LOAD_MISSES.WALK_CYCLES	
EventSel=08H, UMask=04H	DTLB load miss page walk cycles.
DTLB_LOAD_MISSES.STLB_HIT	
EventSel=08H, UMask=10H	DTLB second level hit.
DTLB_LOAD_MISSES.PDE_MISS	
EventSel=08H, UMask=20H	DTLB load miss caused by low part of address.
DTLB_LOAD_MISSES.LARGE_WALK_COMPLETED	
EventSel=08H, UMask=80H	DTLB load miss large page walks.
MEM_INST_RETIRED.LOADS	
EventSel=0BH, UMask=01H, Precise	Instructions retired which contains a load (Precise Event).
MEM_INST_RETIRED.STORES	
EventSel=0BH, UMask=02H, Precise	Instructions retired which contains a store (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_0	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x0 , Precise	Memory instructions retired above 0 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_1024	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x400 , Precise	Memory instructions retired above 1024 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_128	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x80 , Precise	Memory instructions retired above 128 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_16	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x10 , Precise	Memory instructions retired above 16 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_16384	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x4000 , Precise	Memory instructions retired above 16384 clocks (Precise Event).

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_2048	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x800 , Precise	Memory instructions retired above 2048 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_256	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x100 , Precise	Memory instructions retired above 256 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_32	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x20 , Precise	Memory instructions retired above 32 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_32768	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x8000 , Precise	Memory instructions retired above 32768 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_4	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x4 , Precise	Memory instructions retired above 4 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_4096	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x1000 , Precise	Memory instructions retired above 4096 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_512	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x200 , Precise	Memory instructions retired above 512 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_64	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x40 , Precise	Memory instructions retired above 64 clocks (Precise Event).

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_8	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x8 , Precise	Memory instructions retired above 8 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_8192	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x2000 , Precise	Memory instructions retired above 8192 clocks (Precise Event).
MEM_STORE_RETIRED.DTLB_MISS	
EventSel=0CH, UMask=01H, Precise	Retired stores that miss the DTLB (Precise Event).
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=01H	Uops issued.
UOPS_ISSUED.CORE_STALL_CYCLES	
EventSel=0EH, UMask=01H, AnyThread=1, Invert=1, CMask=1	Cycles no Uops were issued on any thread.
UOPS_ISSUED.CYCLES_ALL_THREADS	
EventSel=0EH, UMask=01H, AnyThread=1, CMask=1	Cycles Uops were issued on either thread.
UOPS_ISSUED.STALL_CYCLES	
EventSel=0EH, UMask=01H, Invert=1, CMask=1	Cycles no Uops were issued.
UOPS_ISSUED.FUSED	
EventSel=0EH, UMask=02H	Fused Uops issued.
FP_COMP_OPS_EXE.X87	
EventSel=10H, UMask=01H	Computational floating-point operations executed.
FP_COMP_OPS_EXE.MMX	
EventSel=10H, UMask=02H	MMX Uops.
FP_COMP_OPS_EXE.SSE_FP	
EventSel=10H, UMask=04H	SSE and SSE2 FP Uops.
FP_COMP_OPS_EXE.SSE2_INTEGER	
EventSel=10H, UMask=08H	SSE2 integer Uops.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
FP_COMP_OPS_EXE.SSE_FP_PACKED	
EventSel=10H, UMask=10H	SSE FP packed Uops.
FP_COMP_OPS_EXE.SSE_FP_SCALAR	
EventSel=10H, UMask=20H	SSE FP scalar Uops.
FP_COMP_OPS_EXE.SSE_SINGLE_PRECISION	
EventSel=10H, UMask=40H	SSE* FP single precision Uops.
FP_COMP_OPS_EXE.SSE_DOUBLE_PRECISION	
EventSel=10H, UMask=80H	SSE* FP double precision Uops.
SIMD_INT_128.PACKED_MPY	
EventSel=12H, UMask=01H	128 bit SIMD integer multiply operations.
SIMD_INT_128.PACKED_SHIFT	
EventSel=12H, UMask=02H	128 bit SIMD integer shift operations.
SIMD_INT_128.PACK	
EventSel=12H, UMask=04H	128 bit SIMD integer pack operations.
SIMD_INT_128.UNPACK	
EventSel=12H, UMask=08H	128 bit SIMD integer unpack operations.
SIMD_INT_128.PACKED_LOGICAL	
EventSel=12H, UMask=10H	128 bit SIMD integer logical operations.
SIMD_INT_128.PACKED_ARITH	
EventSel=12H, UMask=20H	128 bit SIMD integer arithmetic operations.
SIMD_INT_128.SHUFFLE_MOVE	
EventSel=12H, UMask=40H	128 bit SIMD integer shuffle/move operations.
LOAD_DISPATCH.RS	
EventSel=13H, UMask=01H	Loads dispatched that bypass the MOB.
LOAD_DISPATCH.RS_DELAYED	
EventSel=13H, UMask=02H	Loads dispatched from stage 305.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
LOAD_DISPATCH.MOB	
EventSel=13H, UMask=04H	Loads dispatched from the MOB.
LOAD_DISPATCH.ANY	
EventSel=13H, UMask=07H	All loads dispatched.
ARITH.CYCLES_DIV_BUSY	
EventSel=14H, UMask=01H	Cycles the divider is busy.
ARITH.DIV	
EventSel=14H, UMask=01H, EdgeDetect=1, Invert=1, CMask=1	Divide Operations executed.
ARITH.MUL	
EventSel=14H, UMask=02H	Multiply operations executed.
INST_QUEUE_WRITES	
EventSel=17H, UMask=01H	Instructions written to instruction queue.
INST_DECODED.DEC0	
EventSel=18H, UMask=01H	Instructions that must be decoded by decoder 0.
TWO_UOP_INSTS_DECODED	
EventSel=19H, UMask=01H	Two Uop instructions decoded.
INST_QUEUE_WRITE_CYCLES	
EventSel=1EH, UMask=01H	Cycles instructions are written to the instruction queue.
LSD_OVERFLOW	
EventSel=20H, UMask=01H	Loops that can't stream from the instruction queue.
L2_RQSTS.LD_HIT	
EventSel=24H, UMask=01H	L2 load hits.
L2_RQSTS.LD_MISS	
EventSel=24H, UMask=02H	L2 load misses.
L2_RQSTS.LOADS	
EventSel=24H, UMask=03H	L2 requests.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
L2_RQSTS.RFO_HIT	
EventSel=24H, UMask=04H	L2 RFO hits.
L2_RQSTS.RFO_MISS	
EventSel=24H, UMask=08H	L2 RFO misses.
L2_RQSTS.RFOS	
EventSel=24H, UMask=0CH	L2 RFO requests.
L2_RQSTS.IFETCH_HIT	
EventSel=24H, UMask=10H	L2 instruction fetch hits.
L2_RQSTS.IFETCH_MISS	
EventSel=24H, UMask=20H	L2 instruction fetch misses.
L2_RQSTS.IFETCHES	
EventSel=24H, UMask=30H	L2 instruction fetches.
L2_RQSTS.PREFETCH_HIT	
EventSel=24H, UMask=40H	L2 prefetch hits.
L2_RQSTS.PREFETCH_MISS	
EventSel=24H, UMask=80H	L2 prefetch misses.
L2_RQSTS.MISS	
EventSel=24H, UMask=AAH	All L2 misses.
L2_RQSTS.PREFETCHES	
EventSel=24H, UMask=C0H	All L2 prefetches.
L2_RQSTS.REFERENCES	
EventSel=24H, UMask=FFH	All L2 requests.
L2_DATA_RQSTS.DEMAND.I_STATE	
EventSel=26H, UMask=01H	L2 data demand loads in I state (misses).
L2_DATA_RQSTS.DEMAND.S_STATE	
EventSel=26H, UMask=02H	L2 data demand loads in S state.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
L2_DATA_RQSTS.DEMAND.E_STATE	
EventSel=26H, UMask=04H	L2 data demand loads in E state.
L2_DATA_RQSTS.DEMAND.M_STATE	
EventSel=26H, UMask=08H	L2 data demand loads in M state.
L2_DATA_RQSTS.DEMAND.MESI	
EventSel=26H, UMask=0FH	L2 data demand requests.
L2_DATA_RQSTS.PREFETCH.I_STATE	
EventSel=26H, UMask=10H	L2 data prefetches in the I state (misses).
L2_DATA_RQSTS.PREFETCH.S_STATE	
EventSel=26H, UMask=20H	L2 data prefetches in the S state.
L2_DATA_RQSTS.PREFETCH.E_STATE	
EventSel=26H, UMask=40H	L2 data prefetches in E state.
L2_DATA_RQSTS.PREFETCH.M_STATE	
EventSel=26H, UMask=80H	L2 data prefetches in M state.
L2_DATA_RQSTS.PREFETCH.MESI	
EventSel=26H, UMask=F0H	All L2 data prefetches.
L2_DATA_RQSTS.ANY	
EventSel=26H, UMask=FFH	All L2 data requests.
L2_WRITE.RFO.I_STATE	
EventSel=27H, UMask=01H	L2 demand store RFOs in I state (misses).
L2_WRITE.RFO.S_STATE	
EventSel=27H, UMask=02H	L2 demand store RFOs in S state.
L2_WRITE.RFO.M_STATE	
EventSel=27H, UMask=08H	L2 demand store RFOs in M state.
L2_WRITE.RFO.HIT	
EventSel=27H, UMask=0EH	All L2 demand store RFOs that hit the cache.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
L2_WRITE.RFO.MESI	
EventSel=27H, UMask=0FH	All L2 demand store RFOs.
L2_WRITE.LOCK.I_STATE	
EventSel=27H, UMask=10H	L2 demand lock RFOs in I state (misses).
L2_WRITE.LOCK.S_STATE	
EventSel=27H, UMask=20H	L2 demand lock RFOs in S state.
L2_WRITE.LOCK.E_STATE	
EventSel=27H, UMask=40H	L2 demand lock RFOs in E state.
L2_WRITE.LOCK.M_STATE	
EventSel=27H, UMask=80H	L2 demand lock RFOs in M state.
L2_WRITE.LOCK.HIT	
EventSel=27H, UMask=E0H	All demand L2 lock RFOs that hit the cache.
L2_WRITE.LOCK.MESI	
EventSel=27H, UMask=F0H	All demand L2 lock RFOs.
L1D_WB_L2.I_STATE	
EventSel=28H, UMask=01H	L1 writebacks to L2 in I state (misses).
L1D_WB_L2.S_STATE	
EventSel=28H, UMask=02H	L1 writebacks to L2 in S state.
L1D_WB_L2.E_STATE	
EventSel=28H, UMask=04H	L1 writebacks to L2 in E state.
L1D_WB_L2.M_STATE	
EventSel=28H, UMask=08H	L1 writebacks to L2 in M state.
L1D_WB_L2.MESI	
EventSel=28H, UMask=0FH	All L1 writebacks to L2.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	Longest latency cache miss.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	Longest latency cache reference.
CPU_CLK_UNHALTED.THREAD_P	
EventSel=3CH, UMask=00H, Architectural	Cycles when thread is not halted (programmable counter).
CPU_CLK_UNHALTED.TOTAL_CYCLES	
EventSel=3CH, UMask=00H, Invert=1, CMask=2, Architectural	Total CPU cycles.
CPU_CLK_UNHALTED.REF_P	
EventSel=3CH, UMask=01H, Architectural	Reference base clock (133 Mhz) cycles when thread is not halted (programmable counter).
DTLB_MISSES.ANY	
EventSel=49H, UMask=01H	DTLB misses.
DTLB_MISSES.WALK_COMPLETED	
EventSel=49H, UMask=02H	DTLB miss page walks.
DTLB_MISSES.WALK_CYCLES	
EventSel=49H, UMask=04H	DTLB miss page walk cycles.
DTLB_MISSES.STLB_HIT	
EventSel=49H, UMask=10H	DTLB first level misses but second level hit.
DTLB_MISSES.PDE_MISS	
EventSel=49H, UMask=20H	DTLB misses casued by low part of address.
DTLB_MISSES.LARGE_WALK_COMPLETED	
EventSel=49H, UMask=80H	DTLB miss large page walks.
LOAD_HIT_PRE	
EventSel=4CH, UMask=01H	Load operations conflicting with software prefetches.
L1D_PREFETCH.REQUESTS	
EventSel=4EH, UMask=01H	L1D hardware prefetch requests.
L1D_PREFETCH.MISS	
EventSel=4EH, UMask=02H	L1D hardware prefetch misses.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
L1D_PREFETCH.TRIGGERS	
EventSel=4EH, UMask=04H	L1D hardware prefetch requests triggered.
EPT.WALK_CYCLES	
EventSel=4FH, UMask=10H	Extended Page Table walk cycles.
L1D.REPL	
EventSel=51H, UMask=01H	L1 data cache lines allocated.
L1D.M_REPL	
EventSel=51H, UMask=02H	L1D cache lines allocated in the M state.
L1D.M_EVICT	
EventSel=51H, UMask=04H	L1D cache lines replaced in M state.
L1D.M_SNOOP_EVICT	
EventSel=51H, UMask=08H	L1D snoop eviction of cache lines in M state.
L1D_CACHE_PREFETCH_LOCK_FB_HIT	
EventSel=52H, UMask=01H	L1D prefetch load lock accepted in fill buffer.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.READ_DATA	
EventSel=60H, UMask=01H	Outstanding offcore demand data reads.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.READ_DATA_NOT_EMPTY	
EventSel=60H, UMask=01H, CMask=1	Cycles offcore demand data read busy.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.READ_CODE	
EventSel=60H, UMask=02H	Outstanding offcore demand code reads.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.READ_CODE_NOT_EMPTY	
EventSel=60H, UMask=02H, CMask=1	Cycles offcore demand code read busy.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.RFO	
EventSel=60H, UMask=04H	Outstanding offcore demand RFOs.
OFFCORE_REQUESTS_OUTSTANDING.DEMAND.RFO_NOT_EMPTY	
EventSel=60H, UMask=04H, CMask=1	Cycles offcore demand RFOs busy.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
OFFCORE_REQUESTS_OUTSTANDING.ANY.READ	
EventSel=60H, UMask=08H	Outstanding offcore reads.
OFFCORE_REQUESTS_OUTSTANDING.ANY.READ_NOT_EMPTY	
EventSel=60H, UMask=08H, CMask=1	Cycles offcore reads busy.
CACHE_LOCK_CYCLES.L1D_L2	
EventSel=63H, UMask=01H	Cycles L1D and L2 locked.
CACHE_LOCK_CYCLES.L1D	
EventSel=63H, UMask=02H	Cycles L1D locked.
IO_TRANSACTIONS	
EventSel=6CH, UMask=01H	I/O transactions.
L1I.HITS	
EventSel=80H, UMask=01H	L1I instruction fetch hits.
L1I.MISSES	
EventSel=80H, UMask=02H	L1I instruction fetch misses.
L1I.READS	
EventSel=80H, UMask=03H	L1I Instruction fetches.
L1I.CYCLES_STALLED	
EventSel=80H, UMask=04H	L1I instruction fetch stall cycles.
LARGE_ITLB.HIT	
EventSel=82H, UMask=01H	Large ITLB hit.
ITLB_MISSES.ANY	
EventSel=85H, UMask=01H	ITLB miss.
ITLB_MISSES.WALK_COMPLETED	
EventSel=85H, UMask=02H	ITLB miss page walks.
ITLB_MISSES.WALK_CYCLES	
EventSel=85H, UMask=04H	ITLB miss page walk cycles.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
ITLB_MISSES.LARGE_WALK_COMPLETED	
EventSel=85H, UMask=80H	ITLB miss large page walks.
ILD_STALL.LCP	
EventSel=87H, UMask=01H	Length Change Prefix stall cycles.
ILD_STALL.MRU	
EventSel=87H, UMask=02H	Stall cycles due to BPU MRU bypass.
ILD_STALL.IQ_FULL	
EventSel=87H, UMask=04H	Instruction Queue full stall cycles.
ILD_STALL.REGEN	
EventSel=87H, UMask=08H	Regen stall cycles.
ILD_STALL.ANY	
EventSel=87H, UMask=0FH	Any Instruction Length Decoder stall cycles.
BR_INST_EXEC.COND	
EventSel=88H, UMask=01H	Conditional branch instructions executed.
BR_INST_EXEC.DIRECT	
EventSel=88H, UMask=02H	Unconditional branches executed.
BR_INST_EXEC.INDIRECT_NON_CALL	
EventSel=88H, UMask=04H	Indirect non call branches executed.
BR_INST_EXEC.NON_CALLS	
EventSel=88H, UMask=07H	All non call branches executed.
BR_INST_EXEC.RETURN_NEAR	
EventSel=88H, UMask=08H	Indirect return branches executed.
BR_INST_EXEC.DIRECT_NEAR_CALL	
EventSel=88H, UMask=10H	Unconditional call branches executed.
BR_INST_EXEC.INDIRECT_NEAR_CALL	
EventSel=88H, UMask=20H	Indirect call branches executed.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
BR_INST_EXEC.NEAR_CALLS	
EventSel=88H, UMask=30H	Call branches executed.
BR_INST_EXEC.TAKEN	
EventSel=88H, UMask=40H	Taken branches executed.
BR_INST_EXEC.ANY	
EventSel=88H, UMask=7FH	Branch instructions executed.
BR_MISP_EXEC.COND	
EventSel=89H, UMask=01H	Mispredicted conditional branches executed.
BR_MISP_EXEC.DIRECT	
EventSel=89H, UMask=02H	Mispredicted unconditional branches executed.
BR_MISP_EXEC.INDIRECT_NON_CALL	
EventSel=89H, UMask=04H	Mispredicted indirect non call branches executed.
BR_MISP_EXEC.NON_CALLS	
EventSel=89H, UMask=07H	Mispredicted non call branches executed.
BR_MISP_EXEC.RETURN_NEAR	
EventSel=89H, UMask=08H	Mispredicted return branches executed.
BR_MISP_EXEC.DIRECT_NEAR_CALL	
EventSel=89H, UMask=10H	Mispredicted non call branches executed.
BR_MISP_EXEC.INDIRECT_NEAR_CALL	
EventSel=89H, UMask=20H	Mispredicted indirect call branches executed.
BR_MISP_EXEC.NEAR_CALLS	
EventSel=89H, UMask=30H	Mispredicted call branches executed.
BR_MISP_EXEC.TAKEN	
EventSel=89H, UMask=40H	Mispredicted taken branches executed.
BR_MISP_EXEC.ANY	
EventSel=89H, UMask=7FH	Mispredicted branches executed.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
RESOURCE_STALLS.ANY	
EventSel=A2H, UMask=01H	Resource related stall cycles.
RESOURCE_STALLS.LOAD	
EventSel=A2H, UMask=02H	Load buffer stall cycles.
RESOURCE_STALLS.RS_FULL	
EventSel=A2H, UMask=04H	Reservation Station full stall cycles.
RESOURCE_STALLS.STORE	
EventSel=A2H, UMask=08H	Store buffer stall cycles.
RESOURCE_STALLS.ROB_FULL	
EventSel=A2H, UMask=10H	ROB full stall cycles.
RESOURCE_STALLS.FPCW	
EventSel=A2H, UMask=20H	FPU control word write stall cycles.
RESOURCE_STALLS.MXCSR	
EventSel=A2H, UMask=40H	MXCSR rename stall cycles.
RESOURCE_STALLS.OTHER	
EventSel=A2H, UMask=80H	Other Resource related stall cycles.
MACRO_INSTS.FUSIONS_DECODED	
EventSel=A6H, UMask=01H	Macro-fused instructions decoded.
BACLEAR_FORCE_IQ	
EventSel=A7H, UMask=01H	Instruction queue forced BACLEAR.
LSD.ACTIVE	
EventSel=A8H, UMask=01H, CMask=1	Cycles when uops were delivered by the LSD.
LSD.INACTIVE	
EventSel=A8H, UMask=01H, Invert=1, CMask=1	Cycles no uops were delivered by the LSD.
ITLB_FLUSH	
EventSel=AEH, UMask=01H	ITLB flushes.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
OFFCORE_REQUESTS.DEMAND.READ_DATA	
EventSel=B0H, UMask=01H	Offcore demand data read requests.
OFFCORE_REQUESTS.DEMAND.READ_CODE	
EventSel=B0H, UMask=02H	Offcore demand code read requests.
OFFCORE_REQUESTS.DEMAND.RFO	
EventSel=B0H, UMask=04H	Offcore demand RFO requests.
OFFCORE_REQUESTS.ANY.READ	
EventSel=B0H, UMask=08H	Offcore read requests.
OFFCORE_REQUESTS.ANY.RFO	
EventSel=B0H, UMask=10H	Offcore RFO requests.
OFFCORE_REQUESTS.L1D_WRITEBACK	
EventSel=B0H, UMask=40H	Offcore L1 data cache writebacks.
OFFCORE_REQUESTS.ANY	
EventSel=B0H, UMask=80H	All offcore requests.
UOPS_EXECUTED.PORT0	
EventSel=B1H, UMask=01H	Uops executed on port 0.
UOPS_EXECUTED.PORT1	
EventSel=B1H, UMask=02H	Uops executed on port 1.
UOPS_EXECUTED.PORT2_CORE	
EventSel=B1H, UMask=04H, AnyThread=1	Uops executed on port 2 (core count).
UOPS_EXECUTED.PORT3_CORE	
EventSel=B1H, UMask=08H, AnyThread=1	Uops executed on port 3 (core count).
UOPS_EXECUTED.PORT4_CORE	
EventSel=B1H, UMask=10H, AnyThread=1	Uops executed on port 4 (core count).
UOPS_EXECUTED.CORE_ACTIVE_CYCLES_NO_PORT5	
EventSel=B1H, UMask=1FH, AnyThread=1, CMask=1	Cycles Uops executed on ports 0-4 (core count).

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
UOPS_EXECUTED.CORE_STALL_COUNT_NO_PORT5	
EventSel=B1H, UMask=1FH, EdgeDetect=1, AnyThread=1, Invert=1, CMask=1	Uops executed on ports 0-4 (core count).
UOPS_EXECUTED.CORE_STALL_CYCLES_NO_PORT5	
EventSel=B1H, UMask=1FH, AnyThread=1, Invert=1, CMask=1	Cycles no Uops issued on ports 0-4 (core count).
UOPS_EXECUTED.PORT5	
EventSel=B1H, UMask=20H	Uops executed on port 5.
UOPS_EXECUTED.CORE_ACTIVE_CYCLES	
EventSel=B1H, UMask=3FH, AnyThread=1, CMask=1	Cycles Uops executed on any port (core count).
UOPS_EXECUTED.CORE_STALL_COUNT	
EventSel=B1H, UMask=3FH, EdgeDetect=1, AnyThread=1, Invert=1, CMask=1	Uops executed on any port (core count).
UOPS_EXECUTED.CORE_STALL_CYCLES	
EventSel=B1H, UMask=3FH, AnyThread=1, Invert=1, CMask=1	Cycles no Uops issued on any port (core count).
UOPS_EXECUTED.PORT015	
EventSel=B1H, UMask=40H	Uops issued on ports 0, 1 or 5.
UOPS_EXECUTED.PORT015_STALL_CYCLES	
EventSel=B1H, UMask=40H, Invert=1, CMask=1	Cycles no Uops issued on ports 0, 1 or 5.
UOPS_EXECUTED.PORT234_CORE	
EventSel=B1H, UMask=80H, AnyThread=1	Uops issued on ports 2, 3 or 4.
OFFCORE_REQUESTS_SQ_FULL	
EventSel=B2H, UMask=01H	Offcore requests blocked due to Super Queue full.
SNOOPQ_REQUESTS_OUTSTANDING.DATA	
EventSel=B3H, UMask=01H	Outstanding snoop data requests.
SNOOPQ_REQUESTS_OUTSTANDING.DATA_NOT_EMPTY	
EventSel=B3H, UMask=01H, CMask=1	Cycles snoop data requests queued.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
SNOOPQ_REQUESTS_OUTSTANDING.INVALIDATE	
EventSel=B3H, UMask=02H	Outstanding snoop invalidate requests.
SNOOPQ_REQUESTS_OUTSTANDING.INVALIDATE_NOT_EMPTY	
EventSel=B3H, UMask=02H, CMask=1	Cycles snoop invalidate requests queued.
SNOOPQ_REQUESTS_OUTSTANDING.CODE	
EventSel=B3H, UMask=04H	Outstanding snoop code requests.
SNOOPQ_REQUESTS_OUTSTANDING.CODE_NOT_EMPTY	
EventSel=B3H, UMask=04H, CMask=1	Cycles snoop code requests queued.
SNOOPQ_REQUESTS.DATA	
EventSel=B4H, UMask=01H	Snoop data requests.
SNOOPQ_REQUESTS.INVALIDATE	
EventSel=B4H, UMask=02H	Snoop invalidate requests.
SNOOPQ_REQUESTS.CODE	
EventSel=B4H, UMask=04H	Snoop code requests.
SNOOP_RESPONSE.HIT	
EventSel=B8H, UMask=01H	Thread responded HIT to snoop.
SNOOP_RESPONSE.HITE	
EventSel=B8H, UMask=02H	Thread responded HITE to snoop.
SNOOP_RESPONSE.HITM	
EventSel=B8H, UMask=04H	Thread responded HITM to snoop.
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=01H, Precise	Instructions retired (Programmable counter and Precise Event).
INST_RETIRED.TOTAL_CYCLES	
EventSel=C0H, UMask=01H, Invert=1, CMask=16, Precise	Total cycles (Precise Event).
INST_RETIRED.X87	
EventSel=C0H, UMask=02H, Precise	Retired floating-point operations (Precise Event).

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
INST_RETIRED.MMX	
EventSel=C0H, UMask=04H, Precise	Retired MMX instructions (Precise Event).
UOPS_RETIRED.ACTIVE_CYCLES	
EventSel=C2H, UMask=01H, CMask=1, Precise	Cycles Uops are being retired.
UOPS_RETIRED.ANY	
EventSel=C2H, UMask=01H, Precise	Uops retired (Precise Event).
UOPS_RETIRED.STALL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=1, Precise	Cycles Uops are not retiring (Precise Event).
UOPS_RETIRED.TOTAL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=16, Precise	Total cycles using precise uop retired event (Precise Event).
UOPS_RETIRED.RETIRE_SLOTS	
EventSel=C2H, UMask=02H, Precise	Retirement slots used (Precise Event).
UOPS_RETIRED.MACRO_FUSED	
EventSel=C2H, UMask=04H, Precise	Macro-fused Uops retired (Precise Event).
MACHINE_CLEARS.CYCLES	
EventSel=C3H, UMask=01H	Cycles machine clear asserted.
MACHINE_CLEARS.MEM_ORDER	
EventSel=C3H, UMask=02H	Execution pipeline restart due to Memory ordering conflicts.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=04H	Self-Modifying Code detected.
BR_INST_RETIRED.CONDITIONAL	
EventSel=C4H, UMask=01H, Precise	Retired conditional branch instructions (Precise Event).
BR_INST_RETIRED.NEAR_CALL	
EventSel=C4H, UMask=02H, Precise	Retired near call instructions (Precise Event).

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
BR_INST_RETIRED.NEAR_CALL_R3	
EventSel=C4H, UMask=02H, USR=1,OS=0, Precise	Retired near call instructions Ring 3 only(Precise Event).
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=04H, Precise	Retired branch instructions (Precise Event).
BR_MISP_RETIRED.CONDITIONAL	
EventSel=C5H, UMask=01H, Precise	Mispredicted conditional retired branches (Precise Event).
BR_MISP_RETIRED.NEAR_CALL	
EventSel=C5H, UMask=02H, Precise	Mispredicted near retired calls (Precise Event).
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=04H, Precise	Mispredicted retired branch instructions (Precise Event).
SSEX_UOPS_RETIRED.PACKED_SINGLE	
EventSel=C7H, UMask=01H, Precise	SIMD Packed-Single Uops retired (Precise Event).
SSEX_UOPS_RETIRED.SCALAR_SINGLE	
EventSel=C7H, UMask=02H, Precise	SIMD Scalar-Single Uops retired (Precise Event).
SSEX_UOPS_RETIRED.PACKED_DOUBLE	
EventSel=C7H, UMask=04H, Precise	SIMD Packed-Double Uops retired (Precise Event).
SSEX_UOPS_RETIRED.SCALAR_DOUBLE	
EventSel=C7H, UMask=08H, Precise	SIMD Scalar-Double Uops retired (Precise Event).
SSEX_UOPS_RETIRED.VECTOR_INTEGER	
EventSel=C7H, UMask=10H, Precise	SIMD Vector Integer Uops retired (Precise Event).
ITLB_MISS_RETIRED	
EventSel=C8H, UMask=20H, Precise	Retired instructions that missed the ITLB (Precise Event).
MEM_LOAD_RETIRED.L1D_HIT	
EventSel=CBH, UMask=01H, Precise	Retired loads that hit the L1 data cache (Precise Event).
MEM_LOAD_RETIRED.L2_HIT	
EventSel=CBH, UMask=02H, Precise	Retired loads that hit the L2 cache (Precise Event).

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
MEM_LOAD_RETIRED.LLC_UNSHARED_HIT	
EventSel=CBH, UMask=04H, Precise	Retired loads that hit valid versions in the LLC cache (Precise Event).
MEM_LOAD_RETIRED.OTHER_CORE_L2_HIT_HITM	
EventSel=CBH, UMask=08H, Precise	Retired loads that hit sibling core's L2 in modified or unmodified states (Precise Event).
MEM_LOAD_RETIRED.LLC_MISS	
EventSel=CBH, UMask=10H, Precise	Retired loads that miss the LLC cache (Precise Event).
MEM_LOAD_RETIRED.HIT_LFB	
EventSel=CBH, UMask=40H, Precise	Retired loads that miss L1D and hit an previously allocated LFB (Precise Event).
MEM_LOAD_RETIRED.DTLB_MISS	
EventSel=CBH, UMask=80H, Precise	Retired loads that miss the DTLB (Precise Event).
FP_MMX_TRANS.TO_FP	
EventSel=CCH, UMask=01H	Transitions from MMX to Floating Point instructions.
FP_MMX_TRANS.TO_MMX	
EventSel=CCH, UMask=02H	Transitions from Floating Point to MMX instructions.
FP_MMX_TRANS.ANY	
EventSel=CCH, UMask=03H	All Floating Point to and from MMX transitions.
MACRO_INSTS.DECODED	
EventSel=D0H, UMask=01H	Instructions decoded.
UOPS_DECODED.STALL_CYCLES	
EventSel=D1H, UMask=01H, Invert=1, CMask=1	Cycles no Uops are decoded.
UOPS_DECODED.MS_CYCLES_ACTIVE	
EventSel=D1H, UMask=02H, CMask=1	Uops decoded by Microcode Sequencer.
UOPS_DECODED.ESP_FOLDING	
EventSel=D1H, UMask=04H	Stack pointer instructions decoded.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
UOPS_DECODED.ESP_SYNC	
EventSel=D1H, UMask=08H	Stack pointer sync operations.
RAT_STALLS.FLAGS	
EventSel=D2H, UMask=01H	Flag stall cycles.
RAT_STALLS.REGISTERS	
EventSel=D2H, UMask=02H	Partial register stall cycles.
RAT_STALLS.ROB_READ_PORT	
EventSel=D2H, UMask=04H	ROB read port stalls cycles.
RAT_STALLS.SCOREBOARD	
EventSel=D2H, UMask=08H	Scoreboard stall cycles.
RAT_STALLS.ANY	
EventSel=D2H, UMask=0FH	All RAT stall cycles.
SEG_RENAME_STALLS	
EventSel=D4H, UMask=01H	Segment rename stall cycles.
ES_REG_RENAMES	
EventSel=D5H, UMask=01H	ES segment renames.
UOP_UNFUSION	
EventSel=DBH, UMask=01H	Uop unfusions due to FP exceptions.
BR_INST_DECODED	
EventSel=E0H, UMask=01H	Branch instructions decoded.
BPU_MISSED_CALL_RET	
EventSel=E5H, UMask=01H	Branch prediction unit missed call or return.
BACLEAR.CLEAR	
EventSel=E6H, UMask=01H	BACLEAR asserted, regardless of cause .
BACLEAR.BAD_TARGET	
EventSel=E6H, UMask=02H	BACLEAR asserted with bad target address.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
BPU_CLEARS.EARLY	
EventSel=E8H, UMask=01H	Early Branch Prediciton Unit clears.
BPU_CLEARS.LATE	
EventSel=E8H, UMask=02H	Late Branch Prediction Unit clears.
L2_TRANSACTIONS.LOAD	
EventSel=F0H, UMask=01H	L2 Load transactions.
L2_TRANSACTIONS.RFO	
EventSel=F0H, UMask=02H	L2 RFO transactions.
L2_TRANSACTIONS.IFETCH	
EventSel=F0H, UMask=04H	L2 instruction fetch transactions.
L2_TRANSACTIONS.PREFETCH	
EventSel=F0H, UMask=08H	L2 prefetch transactions.
L2_TRANSACTIONS.L1D_WB	
EventSel=F0H, UMask=10H	L1D writeback to L2 transactions.
L2_TRANSACTIONS.FILL	
EventSel=F0H, UMask=20H	L2 fill transactions.
L2_TRANSACTIONS.WB	
EventSel=F0H, UMask=40H	L2 writeback to LLC transactions.
L2_TRANSACTIONS.ANY	
EventSel=F0H, UMask=80H	All L2 transactions.
L2_LINES_IN.S_STATE	
EventSel=F1H, UMask=02H	L2 lines allocated in the S state.
L2_LINES_IN.E_STATE	
EventSel=F1H, UMask=04H	L2 lines allocated in the E state.
L2_LINES_IN.ANY	
EventSel=F1H, UMask=07H	L2 lines alloacated.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
L2_LINES_OUT.DEMAND_CLEAN	
EventSel=F2H, UMask=01H	L2 lines evicted by a demand request.
L2_LINES_OUT.DEMAND_DIRTY	
EventSel=F2H, UMask=02H	L2 modified lines evicted by a demand request.
L2_LINES_OUT.PREFETCH_CLEAN	
EventSel=F2H, UMask=04H	L2 lines evicted by a prefetch request.
L2_LINES_OUT.PREFETCH_DIRTY	
EventSel=F2H, UMask=08H	L2 modified lines evicted by a prefetch request.
L2_LINES_OUT.ANY	
EventSel=F2H, UMask=0FH	L2 lines evicted.
SQ_MISC.LRU_HINTS	
EventSel=F4H, UMask=04H	Super Queue LRU hints sent to LLC.
SQ_MISC.SPLIT_LOCK	
EventSel=F4H, UMask=10H	Super Queue lock splits across a cache line.
SQ_FULL_STALL_CYCLES	
EventSel=F6H, UMask=01H	Super Queue full stall cycles.
FP_ASSIST.ALL	
EventSel=F7H, UMask=01H, Precise	X87 Floating point assists (Precise Event).
FP_ASSIST.OUTPUT	
EventSel=F7H, UMask=02H, Precise	X87 Floating point assists for invalid output value (Precise Event).
FP_ASSIST.INPUT	
EventSel=F7H, UMask=04H, Precise	X87 Floating poiint assists for invalid input value (Precise Event).
SIMD_INT_64.PACKED_MPY	
EventSel=FDH, UMask=01H	SIMD integer 64 bit packed multiply operations.
SIMD_INT_64.PACKED_SHIFT	
EventSel=FDH, UMask=02H	SIMD integer 64 bit shift operations.

Table 10: Performance Events In the Processor Core for Processors Based on code name Westmere Intel® Microarchitecture Code Name Westmere (06_25H, 06_2CH)

Event Name	
Configuration	Description
SIMD_INT_64.PACK	
EventSel=FDH, UMask=04H	SIMD integer 64 bit pack operations.
SIMD_INT_64.UNPACK	
EventSel=FDH, UMask=08H	SIMD integer 64 bit unpack operations.
SIMD_INT_64.PACKED_LOGICAL	
EventSel=FDH, UMask=10H	SIMD integer 64 bit logical operations.
SIMD_INT_64.PACKED_ARITH	
EventSel=FDH, UMask=20H	SIMD integer 64 bit arithmetic operations.
SIMD_INT_64.SHUFFLE_MOVE	
EventSel=FDH, UMask=40H	SIMD integer 64 bit shuffle/move operations.

Performance Monitoring Events based on Nehalem Microarchitecture - Intel® Core™ i7 Processor Family and Intel® Xeon®® Processor Family

Processors based on the Intel Microarchitecture code name Nehalem support the performance-monitoring events listed in the table below. Intel Xeon® processors with CPUID signature of DisplayFamily_DisplayModel 06_2EH have a small number of events that are not supported in processors with CPUID signature 06_1AH, 06_1EH, and 06_1FH. These events are noted in the comment column

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
CPU_CLK_UNHALTED.REF	
Architectural, Fixed	Reference cycles when thread is not halted (fixed counter).
CPU_CLK_UNHALTED.THREAD	
Architectural, Fixed	Cycles when thread is not halted (fixed counter).
INST_RETIRED.ANY	
Architectural, Fixed	Instructions retired (fixed counter).
SB_DRAIN.ANY	
EventSel=04H, UMask=07H	All Store buffer stall cycles.
STORE_BLOCKS.AT_RET	
EventSel=06H, UMask=04H	Loads delayed with at-Retirement block code.
STORE_BLOCKS.L1D_BLOCK	
EventSel=06H, UMask=08H	Cacheable loads delayed with L1D block code.
PARTIAL_ADDRESS_ALIAS	
EventSel=07H, UMask=01H	False dependencies due to partial address aliasing.
DTLB_LOAD_MISSES.ANY	
EventSel=08H, UMask=01H	DTLB load misses.
DTLB_LOAD_MISSES.WALK_COMPLETED	
EventSel=08H, UMask=02H	DTLB load miss page walks complete.
DTLB_LOAD_MISSES.STLB_HIT	
EventSel=08H, UMask=10H	DTLB second level hit.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
DTLB_LOAD_MISSES.PDE_MISS	
EventSel=08H, UMask=20H	DTLB load miss caused by low part of address.
MEM_INST_RETIRED.LOADS	
EventSel=0BH, UMask=01H, Precise	Instructions retired which contains a load (Precise Event).
MEM_INST_RETIRED.STORES	
EventSel=0BH, UMask=02H, Precise	Instructions retired which contains a store (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_0	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x0 , Precise	Memory instructions retired above 0 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_1024	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x400 , Precise	Memory instructions retired above 1024 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_128	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x80 , Precise	Memory instructions retired above 128 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_16	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x10 , Precise	Memory instructions retired above 16 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_16384	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x4000 , Precise	Memory instructions retired above 16384 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_2048	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x800 , Precise	Memory instructions retired above 2048 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_256	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x100 , Precise	Memory instructions retired above 256 clocks (Precise Event).

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_32	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x20 , Precise	Memory instructions retired above 32 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_32768	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x8000 , Precise	Memory instructions retired above 32768 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_4	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x4 , Precise	Memory instructions retired above 4 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_4096	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x1000 , Precise	Memory instructions retired above 4096 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_512	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x200 , Precise	Memory instructions retired above 512 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_64	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x40 , Precise	Memory instructions retired above 64 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_8	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x8 , Precise	Memory instructions retired above 8 clocks (Precise Event).
MEM_INST_RETIRED.LATENCY_ABOVE_THRESHOLD_8192	
EventSel=0BH, UMask=10H, MSR_PEBS_LD_LAT_THRESHOLD=0x2000 , Precise	Memory instructions retired above 8192 clocks (Precise Event).
MEM_STORE_RETIRED.DTLB_MISS	
EventSel=0CH, UMask=01H, Precise	Retired stores that miss the DTLB (Precise Event).

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=01H	Uops issued.
UOPS_ISSUED.CORE_STALL_CYCLES	
EventSel=0EH, UMask=01H, AnyThread=1, Invert=1, CMask=1	Cycles no Uops were issued on any thread.
UOPS_ISSUED.CYCLES_ALL_THREADS	
EventSel=0EH, UMask=01H, AnyThread=1, CMask=1	Cycles Uops were issued on either thread.
UOPS_ISSUED.STALL_CYCLES	
EventSel=0EH, UMask=01H, Invert=1, CMask=1	Cycles no Uops were issued.
UOPS_ISSUED.FUSED	
EventSel=0EH, UMask=02H	Fused Uops issued.
MEM_UNCORE_RETIRED.OTHER_CORE_L2_HITM	
EventSel=0FH, UMask=02H, Precise	Load instructions retired that HIT modified data in sibling core (Precise Event).
MEM_UNCORE_RETIRED.REMOTE_CACHE_LOCAL_HOME_HIT	
EventSel=0FH, UMask=08H, Precise	Load instructions retired remote cache HIT data source (Precise Event).
MEM_UNCORE_RETIRED.REMOTE_DRAM	
EventSel=0FH, UMask=10H, Precise	Load instructions retired remote DRAM and remote home-remote cache HITM (Precise Event).
MEM_UNCORE_RETIRED.LOCAL_DRAM	
EventSel=0FH, UMask=20H, Precise	Load instructions retired with a data source of local DRAM or locally homed remote hitm (Precise Event).
MEM_UNCORE_RETIRED.UNCACHEABLE	
EventSel=0FH, UMask=80H, Precise	Load instructions retired IO (Precise Event).
FP_COMP_OPS_EXE.X87	
EventSel=10H, UMask=01H	Computational floating-point operations executed.
FP_COMP_OPS_EXE.MMX	
EventSel=10H, UMask=02H	MMX Uops.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
FP_COMP_OPS_EXE.SSE_FP	
EventSel=10H, UMask=04H	SSE and SSE2 FP Uops.
FP_COMP_OPS_EXE.SSE2_INTEGER	
EventSel=10H, UMask=08H	SSE2 integer Uops.
FP_COMP_OPS_EXE.SSE_FP_PACKED	
EventSel=10H, UMask=10H	SSE FP packed Uops.
FP_COMP_OPS_EXE.SSE_FP_SCALAR	
EventSel=10H, UMask=20H	SSE FP scalar Uops.
FP_COMP_OPS_EXE.SSE_SINGLE_PRECISION	
EventSel=10H, UMask=40H	SSE* FP single precision Uops.
FP_COMP_OPS_EXE.SSE_DOUBLE_PRECISION	
EventSel=10H, UMask=80H	SSE* FP double precision Uops.
SIMD_INT_128.PACKED_MPY	
EventSel=12H, UMask=01H	128 bit SIMD integer multiply operations.
SIMD_INT_128.PACKED_SHIFT	
EventSel=12H, UMask=02H	128 bit SIMD integer shift operations.
SIMD_INT_128.PACK	
EventSel=12H, UMask=04H	128 bit SIMD integer pack operations.
SIMD_INT_128.UNPACK	
EventSel=12H, UMask=08H	128 bit SIMD integer unpack operations.
SIMD_INT_128.PACKED_LOGICAL	
EventSel=12H, UMask=10H	128 bit SIMD integer logical operations.
SIMD_INT_128.PACKED_ARITH	
EventSel=12H, UMask=20H	128 bit SIMD integer arithmetic operations.
SIMD_INT_128.SHUFFLE_MOVE	
EventSel=12H, UMask=40H	128 bit SIMD integer shuffle/move operations.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
LOAD_DISPATCH.RS	
EventSel=13H, UMask=01H	Loads dispatched that bypass the MOB.
LOAD_DISPATCH.RS_DELAYED	
EventSel=13H, UMask=02H	Loads dispatched from stage 305.
LOAD_DISPATCH.MOB	
EventSel=13H, UMask=04H	Loads dispatched from the MOB.
LOAD_DISPATCH.ANY	
EventSel=13H, UMask=07H	All loads dispatched.
ARITH.CYCLES_DIV_BUSY	
EventSel=14H, UMask=01H	Cycles the divider is busy.
ARITH.DIV	
EventSel=14H, UMask=01H, EdgeDetect=1, Invert=1, CMask=1	Divide Operations executed.
ARITH.MUL	
EventSel=14H, UMask=02H	Multiply operations executed.
INST_QUEUE_WRITES	
EventSel=17H, UMask=01H	Instructions written to instruction queue.
INST_DECODED.DEC0	
EventSel=18H, UMask=01H	Instructions that must be decoded by decoder 0.
TWO_UOP_INSTS_DECODED	
EventSel=19H, UMask=01H	Two Uop instructions decoded.
INST_QUEUE_WRITE_CYCLES	
EventSel=1EH, UMask=01H	Cycles instructions are written to the instruction queue.
LSD_OVERFLOW	
EventSel=20H, UMask=01H	Loops that can't stream from the instruction queue.
L2_RQSTS.LD_HIT	
EventSel=24H, UMask=01H	L2 load hits.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
L2_RQSTS.LD_MISS	
EventSel=24H, UMask=02H	L2 load misses.
L2_RQSTS.LOADS	
EventSel=24H, UMask=03H	L2 requests.
L2_RQSTS.RFO_HIT	
EventSel=24H, UMask=04H	L2 RFO hits.
L2_RQSTS.RFO_MISS	
EventSel=24H, UMask=08H	L2 RFO misses.
L2_RQSTS.RFOS	
EventSel=24H, UMask=0CH	L2 RFO requests.
L2_RQSTS.IFETCH_HIT	
EventSel=24H, UMask=10H	L2 instruction fetch hits.
L2_RQSTS.IFETCH_MISS	
EventSel=24H, UMask=20H	L2 instruction fetch misses.
L2_RQSTS.IFETCHES	
EventSel=24H, UMask=30H	L2 instruction fetches.
L2_RQSTS.PREFETCH_HIT	
EventSel=24H, UMask=40H	L2 prefetch hits.
L2_RQSTS.PREFETCH_MISS	
EventSel=24H, UMask=80H	L2 prefetch misses.
L2_RQSTS.MISS	
EventSel=24H, UMask=AAH	All L2 misses.
L2_RQSTS.PREFETCHES	
EventSel=24H, UMask=C0H	All L2 prefetches.
L2_RQSTS.REFERENCES	
EventSel=24H, UMask=FFH	All L2 requests.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
L2_DATA_RQSTS.DEMAND.I_STATE	
EventSel=26H, UMask=01H	L2 data demand loads in I state (misses).
L2_DATA_RQSTS.DEMAND.S_STATE	
EventSel=26H, UMask=02H	L2 data demand loads in S state.
L2_DATA_RQSTS.DEMAND.E_STATE	
EventSel=26H, UMask=04H	L2 data demand loads in E state.
L2_DATA_RQSTS.DEMAND.M_STATE	
EventSel=26H, UMask=08H	L2 data demand loads in M state.
L2_DATA_RQSTS.DEMAND.MESI	
EventSel=26H, UMask=0FH	L2 data demand requests.
L2_DATA_RQSTS.PREFETCH.I_STATE	
EventSel=26H, UMask=10H	L2 data prefetches in the I state (misses).
L2_DATA_RQSTS.PREFETCH.S_STATE	
EventSel=26H, UMask=20H	L2 data prefetches in the S state.
L2_DATA_RQSTS.PREFETCH.E_STATE	
EventSel=26H, UMask=40H	L2 data prefetches in E state.
L2_DATA_RQSTS.PREFETCH.M_STATE	
EventSel=26H, UMask=80H	L2 data prefetches in M state.
L2_DATA_RQSTS.PREFETCH.MESI	
EventSel=26H, UMask=F0H	All L2 data prefetches.
L2_DATA_RQSTS.ANY	
EventSel=26H, UMask=FFH	All L2 data requests.
L2_WRITE.RFO.I_STATE	
EventSel=27H, UMask=01H	L2 demand store RFOs in I state (misses).
L2_WRITE.RFO.S_STATE	
EventSel=27H, UMask=02H	L2 demand store RFOs in S state.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
L2_WRITE.RFO.M_STATE	
EventSel=27H, UMask=08H	L2 demand store RFOs in M state.
L2_WRITE.RFO.HIT	
EventSel=27H, UMask=0EH	All L2 demand store RFOs that hit the cache.
L2_WRITE.RFO.MESI	
EventSel=27H, UMask=0FH	All L2 demand store RFOs.
L2_WRITE.LOCK.I_STATE	
EventSel=27H, UMask=10H	L2 demand lock RFOs in I state (misses).
L2_WRITE.LOCK.S_STATE	
EventSel=27H, UMask=20H	L2 demand lock RFOs in S state.
L2_WRITE.LOCK.E_STATE	
EventSel=27H, UMask=40H	L2 demand lock RFOs in E state.
L2_WRITE.LOCK.M_STATE	
EventSel=27H, UMask=80H	L2 demand lock RFOs in M state.
L2_WRITE.LOCK.HIT	
EventSel=27H, UMask=E0H	All demand L2 lock RFOs that hit the cache.
L2_WRITE.LOCK.MESI	
EventSel=27H, UMask=F0H	All demand L2 lock RFOs.
L1D_WB_L2.I_STATE	
EventSel=28H, UMask=01H	L1 writebacks to L2 in I state (misses).
L1D_WB_L2.S_STATE	
EventSel=28H, UMask=02H	L1 writebacks to L2 in S state.
L1D_WB_L2.E_STATE	
EventSel=28H, UMask=04H	L1 writebacks to L2 in E state.
L1D_WB_L2.M_STATE	
EventSel=28H, UMask=08H	L1 writebacks to L2 in M state.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
L1D_WB_L2.MESI	
EventSel=28H, UMask=0FH	All L1 writebacks to L2.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	Longest latency cache miss.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	Longest latency cache reference.
CPU_CLK_UNHALTED.THREAD_P	
EventSel=3CH, UMask=00H, Architectural	Cycles when thread is not halted (programmable counter).
CPU_CLK_UNHALTED.TOTAL_CYCLES	
EventSel=3CH, UMask=00H, Invert=1, CMask=2, Architectural	Total CPU cycles.
CPU_CLK_UNHALTED.REF_P	
EventSel=3CH, UMask=01H, Architectural	Reference base clock (133 Mhz) cycles when thread is not halted (programmable counter).
L1D_CACHE_LD.I_STATE	
EventSel=40H, UMask=01H	L1 data cache read in I state (misses).
L1D_CACHE_LD.S_STATE	
EventSel=40H, UMask=02H	L1 data cache read in S state.
L1D_CACHE_LD.E_STATE	
EventSel=40H, UMask=04H	L1 data cache read in E state.
L1D_CACHE_LD.M_STATE	
EventSel=40H, UMask=08H	L1 data cache read in M state.
L1D_CACHE_LD.MESI	
EventSel=40H, UMask=0FH	L1 data cache reads.
L1D_CACHE_ST.S_STATE	
EventSel=41H, UMask=02H	L1 data cache stores in S state.
L1D_CACHE_ST.E_STATE	
EventSel=41H, UMask=04H	L1 data cache stores in E state.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
L1D_CACHE_ST.M_STATE	
EventSel=41H, UMask=08H	L1 data cache stores in M state.
L1D_CACHE_LOCK.HIT	
EventSel=42H, UMask=01H	L1 data cache load lock hits.
L1D_CACHE_LOCK.S_STATE	
EventSel=42H, UMask=02H	L1 data cache load locks in S state.
L1D_CACHE_LOCK.E_STATE	
EventSel=42H, UMask=04H	L1 data cache load locks in E state.
L1D_CACHE_LOCK.M_STATE	
EventSel=42H, UMask=08H	L1 data cache load locks in M state.
L1D_ALL_REF.ANY	
EventSel=43H, UMask=01H	All references to the L1 data cache.
L1D_ALL_REF.CACHEABLE	
EventSel=43H, UMask=02H	L1 data cacheable reads and writes.
DTLB_MISSES.ANY	
EventSel=49H, UMask=01H	DTLB misses.
DTLB_MISSES.WALK_COMPLETED	
EventSel=49H, UMask=02H	DTLB miss page walks.
DTLB_MISSES.STLB_HIT	
EventSel=49H, UMask=10H	DTLB first level misses but second level hit.
LOAD_HIT_PRE	
EventSel=4CH, UMask=01H	Load operations conflicting with software prefetches.
L1D_PREFETCH.REQUESTS	
EventSel=4EH, UMask=01H	L1D hardware prefetch requests.
L1D_PREFETCH.MISS	
EventSel=4EH, UMask=02H	L1D hardware prefetch misses.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
L1D_PREFETCH.TRIGGERS	
EventSel=4EH, UMask=04H	L1D hardware prefetch requests triggered.
L1D.REPL	
EventSel=51H, UMask=01H	L1 data cache lines allocated.
L1D.M_REPL	
EventSel=51H, UMask=02H	L1D cache lines allocated in the M state.
L1D.M_EVICT	
EventSel=51H, UMask=04H	L1D cache lines replaced in M state.
L1D.M_SNOOP_EVICT	
EventSel=51H, UMask=08H	L1D snoop eviction of cache lines in M state.
L1D_CACHE_PREFETCH_LOCK_FB_HIT	
EventSel=52H, UMask=01H	L1D prefetch load lock accepted in fill buffer.
L1D_CACHE_LOCK_FB_HIT	
EventSel=53H, UMask=01H	L1D load lock accepted in fill buffer.
CACHE_LOCK_CYCLES.L1D_L2	
EventSel=63H, UMask=01H	Cycles L1D and L2 locked.
CACHE_LOCK_CYCLES.L1D	
EventSel=63H, UMask=02H	Cycles L1D locked.
IO_TRANSACTIONS	
EventSel=6CH, UMask=01H	I/O transactions.
L1I.HITS	
EventSel=80H, UMask=01H	L1I instruction fetch hits.
L1I.MISSES	
EventSel=80H, UMask=02H	L1I instruction fetch misses.
L1I.READS	
EventSel=80H, UMask=03H	L1I Instruction fetches.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
L1I.CYCLES_STALLED	
EventSel=80H, UMask=04H	L1I instruction fetch stall cycles.
LARGE_ITLB.HIT	
EventSel=82H, UMask=01H	Large ITLB hit.
ITLB_MISSES.ANY	
EventSel=85H, UMask=01H	ITLB miss.
ITLB_MISSES.WALK_COMPLETED	
EventSel=85H, UMask=02H	ITLB miss page walks.
ILD_STALL.LCP	
EventSel=87H, UMask=01H	Length Change Prefix stall cycles.
ILD_STALL.MRU	
EventSel=87H, UMask=02H	Stall cycles due to BPU MRU bypass.
ILD_STALL.IQ_FULL	
EventSel=87H, UMask=04H	Instruction Queue full stall cycles.
ILD_STALL.REGEN	
EventSel=87H, UMask=08H	Regen stall cycles.
ILD_STALL.ANY	
EventSel=87H, UMask=0FH	Any Instruction Length Decoder stall cycles.
BR_INST_EXEC.COND	
EventSel=88H, UMask=01H	Conditional branch instructions executed.
BR_INST_EXEC.DIRECT	
EventSel=88H, UMask=02H	Unconditional branches executed.
BR_INST_EXEC.INDIRECT_NON_CALL	
EventSel=88H, UMask=04H	Indirect non call branches executed.
BR_INST_EXEC.NON_CALLS	
EventSel=88H, UMask=07H	All non call branches executed.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
BR_INST_EXEC.RETURN_NEAR	
EventSel=88H, UMask=08H	Indirect return branches executed.
BR_INST_EXEC.DIRECT_NEAR_CALL	
EventSel=88H, UMask=10H	Unconditional call branches executed.
BR_INST_EXEC.INDIRECT_NEAR_CALL	
EventSel=88H, UMask=20H	Indirect call branches executed.
BR_INST_EXEC.NEAR_CALLS	
EventSel=88H, UMask=30H	Call branches executed.
BR_INST_EXEC.TAKEN	
EventSel=88H, UMask=40H	Taken branches executed.
BR_INST_EXEC.ANY	
EventSel=88H, UMask=7FH	Branch instructions executed.
BR_MISP_EXEC.COND	
EventSel=89H, UMask=01H	Mispredicted conditional branches executed.
BR_MISP_EXEC.DIRECT	
EventSel=89H, UMask=02H	Mispredicted unconditional branches executed.
BR_MISP_EXEC.INDIRECT_NON_CALL	
EventSel=89H, UMask=04H	Mispredicted indirect non call branches executed.
BR_MISP_EXEC.NON_CALLS	
EventSel=89H, UMask=07H	Mispredicted non call branches executed.
BR_MISP_EXEC.RETURN_NEAR	
EventSel=89H, UMask=08H	Mispredicted return branches executed.
BR_MISP_EXEC.DIRECT_NEAR_CALL	
EventSel=89H, UMask=10H	Mispredicted non call branches executed.
BR_MISP_EXEC.INDIRECT_NEAR_CALL	
EventSel=89H, UMask=20H	Mispredicted indirect call branches executed.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
BR_MISP_EXEC.NEAR_CALLS	
EventSel=89H, UMask=30H	Mispredicted call branches executed.
BR_MISP_EXEC.TAKEN	
EventSel=89H, UMask=40H	Mispredicted taken branches executed.
BR_MISP_EXEC.ANY	
EventSel=89H, UMask=7FH	Mispredicted branches executed.
RESOURCE_STALLS.ANY	
EventSel=A2H, UMask=01H	Resource related stall cycles.
RESOURCE_STALLS.LOAD	
EventSel=A2H, UMask=02H	Load buffer stall cycles.
RESOURCE_STALLS.RS_FULL	
EventSel=A2H, UMask=04H	Reservation Station full stall cycles.
RESOURCE_STALLS.STORE	
EventSel=A2H, UMask=08H	Store buffer stall cycles.
RESOURCE_STALLS.ROB_FULL	
EventSel=A2H, UMask=10H	ROB full stall cycles.
RESOURCE_STALLS.FPCW	
EventSel=A2H, UMask=20H	FPU control word write stall cycles.
RESOURCE_STALLS.MXCSR	
EventSel=A2H, UMask=40H	MXCSR rename stall cycles.
RESOURCE_STALLS.OTHER	
EventSel=A2H, UMask=80H	Other Resource related stall cycles.
MACRO_INSTS.FUSIONS_DECODED	
EventSel=A6H, UMask=01H	Macro-fused instructions decoded.
BACLEAR_FORCE_IQ	
EventSel=A7H, UMask=01H	Instruction queue forced BACLEAR.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
LSD.ACTIVE	
EventSel=A8H, UMask=01H, CMask=1	Cycles when uops were delivered by the LSD.
LSD.INACTIVE	
EventSel=A8H, UMask=01H, Invert=1, CMask=1	Cycles no uops were delivered by the LSD.
ITLB_FLUSH	
EventSel=AEH, UMask=01H	ITLB flushes.
OFFCORE_REQUESTS.L1D_WRITEBACK	
EventSel=B0H, UMask=40H	Offcore L1 data cache writebacks.
UOPS_EXECUTED.PORT0	
EventSel=B1H, UMask=01H	Uops executed on port 0.
UOPS_EXECUTED.PORT1	
EventSel=B1H, UMask=02H	Uops executed on port 1.
UOPS_EXECUTED.PORT2_CORE	
EventSel=B1H, UMask=04H, AnyThread=1	Uops executed on port 2 (core count).
UOPS_EXECUTED.PORT3_CORE	
EventSel=B1H, UMask=08H, AnyThread=1	Uops executed on port 3 (core count).
UOPS_EXECUTED.PORT4_CORE	
EventSel=B1H, UMask=10H, AnyThread=1	Uops executed on port 4 (core count).
UOPS_EXECUTED.CORE_ACTIVE_CYCLES_NO_PORT5	
EventSel=B1H, UMask=1FH, AnyThread=1, CMask=1	Cycles Uops executed on ports 0-4 (core count).
UOPS_EXECUTED.CORE_STALL_COUNT_NO_PORT5	
EventSel=B1H, UMask=1FH, EdgeDetect=1, AnyThread=1, Invert=1, CMask=1	Uops executed on ports 0-4 (core count).
UOPS_EXECUTED.CORE_STALL_CYCLES_NO_PORT5	
EventSel=B1H, UMask=1FH, AnyThread=1, Invert=1, CMask=1	Cycles no Uops issued on ports 0-4 (core count).

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
UOPS_EXECUTED.PORT5	
EventSel=B1H, UMask=20H	Uops executed on port 5.
UOPS_EXECUTED.CORE_ACTIVE_CYCLES	
EventSel=B1H, UMask=3FH, AnyThread=1, CMask=1	Cycles Uops executed on any port (core count).
UOPS_EXECUTED.CORE_STALL_COUNT	
EventSel=B1H, UMask=3FH, EdgeDetect=1, AnyThread=1, Invert=1, CMask=1	Uops executed on any port (core count).
UOPS_EXECUTED.CORE_STALL_CYCLES	
EventSel=B1H, UMask=3FH, AnyThread=1, Invert=1, CMask=1	Cycles no Uops issued on any port (core count).
UOPS_EXECUTED.PORT015	
EventSel=B1H, UMask=40H	Uops issued on ports 0, 1 or 5.
UOPS_EXECUTED.PORT015_STALL_CYCLES	
EventSel=B1H, UMask=40H, Invert=1, CMask=1	Cycles no Uops issued on ports 0, 1 or 5.
UOPS_EXECUTED.PORT234_CORE	
EventSel=B1H, UMask=80H, AnyThread=1	Uops issued on ports 2, 3 or 4.
OFFCORE_REQUESTS_SQ_FULL	
EventSel=B2H, UMask=01H	Offcore requests blocked due to Super Queue full.
SNOOP_RESPONSE.HIT	
EventSel=B8H, UMask=01H	Thread responded HIT to snoop.
SNOOP_RESPONSE.HITE	
EventSel=B8H, UMask=02H	Thread responded HITE to snoop.
SNOOP_RESPONSE.HITM	
EventSel=B8H, UMask=04H	Thread responded HITM to snoop.
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=01H, Precise	Instructions retired (Programmable counter and Precise Event).

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
INST_RETIRED.TOTAL_CYCLES	
EventSel=C0H, UMask=01H, Invert=1, CMask=16, Precise	Total cycles (Precise Event).
INST_RETIRED.X87	
EventSel=C0H, UMask=02H, Precise	Retired floating-point operations (Precise Event).
INST_RETIRED.MMX	
EventSel=C0H, UMask=04H, Precise	Retired MMX instructions (Precise Event).
UOPS_RETIRED.ACTIVE_CYCLES	
EventSel=C2H, UMask=01H, CMask=1, Precise	Cycles Uops are being retired.
UOPS_RETIRED.ANY	
EventSel=C2H, UMask=01H, Precise	Uops retired (Precise Event).
UOPS_RETIRED.STALL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=1, Precise	Cycles Uops are not retiring (Precise Event).
UOPS_RETIRED.TOTAL_CYCLES	
EventSel=C2H, UMask=01H, Invert=1, CMask=16, Precise	Total cycles using precise uop retired event (Precise Event).
UOPS_RETIRED.RETIRE_SLOTS	
EventSel=C2H, UMask=02H, Precise	Retirement slots used (Precise Event).
UOPS_RETIRED.MACRO_FUSED	
EventSel=C2H, UMask=04H, Precise	Macro-fused Uops retired (Precise Event).
MACHINE_CLEARS.CYCLES	
EventSel=C3H, UMask=01H	Cycles machine clear asserted.
MACHINE_CLEARS.MEM_ORDER	
EventSel=C3H, UMask=02H	Execution pipeline restart due to Memory ordering conflicts.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=04H	Self-Modifying Code detected.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
BR_INST_RETIRED.CONDITIONAL	
EventSel=C4H, UMask=01H, Precise	Retired conditional branch instructions (Precise Event).
BR_INST_RETIRED.NEAR_CALL	
EventSel=C4H, UMask=02H, Precise	Retired near call instructions (Precise Event).
BR_INST_RETIRED.NEAR_CALL_R3	
EventSel=C4H, UMask=02H, USR=1,OS=0, Precise	Retired near call instructions Ring 3 only(Precise Event).
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=04H, Precise	Retired branch instructions (Precise Event).
BR_MISP_RETIRED.NEAR_CALL	
EventSel=C5H, UMask=02H, Precise	Mispredicted near retired calls (Precise Event).
SSEX_UOPS_RETIRED.PACKED_SINGLE	
EventSel=C7H, UMask=01H, Precise	SIMD Packed-Single Uops retired (Precise Event).
SSEX_UOPS_RETIRED.SCALAR_SINGLE	
EventSel=C7H, UMask=02H, Precise	SIMD Scalar-Single Uops retired (Precise Event).
SSEX_UOPS_RETIRED.PACKED_DOUBLE	
EventSel=C7H, UMask=04H, Precise	SIMD Packed-Double Uops retired (Precise Event).
SSEX_UOPS_RETIRED.SCALAR_DOUBLE	
EventSel=C7H, UMask=08H, Precise	SIMD Scalar-Double Uops retired (Precise Event).
SSEX_UOPS_RETIRED.VECTOR_INTEGER	
EventSel=C7H, UMask=10H, Precise	SIMD Vector Integer Uops retired (Precise Event).
ITLB_MISS_RETIRED	
EventSel=C8H, UMask=20H, Precise	Retired instructions that missed the ITLB (Precise Event).
MEM_LOAD_RETIRED.L1D_HIT	
EventSel=CBH, UMask=01H, Precise	Retired loads that hit the L1 data cache (Precise Event).
MEM_LOAD_RETIRED.L2_HIT	
EventSel=CBH, UMask=02H, Precise	Retired loads that hit the L2 cache (Precise Event).

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
MEM_LOAD_RETIRED.LLC_UNSHARED_HIT	
EventSel=CBH, UMask=04H, Precise	Retired loads that hit valid versions in the LLC cache (Precise Event).
MEM_LOAD_RETIRED.OTHER_CORE_L2_HIT_HITM	
EventSel=CBH, UMask=08H, Precise	Retired loads that hit sibling core's L2 in modified or unmodified states (Precise Event).
MEM_LOAD_RETIRED.LLC_MISS	
EventSel=CBH, UMask=10H, Precise	Retired loads that miss the LLC cache (Precise Event).
MEM_LOAD_RETIRED.HIT_LFB	
EventSel=CBH, UMask=40H, Precise	Retired loads that miss L1D and hit an previously allocated LFB (Precise Event).
MEM_LOAD_RETIRED.DTLB_MISS	
EventSel=CBH, UMask=80H, Precise	Retired loads that miss the DTLB (Precise Event).
FP_MMX_TRANS.TO_FP	
EventSel=CCH, UMask=01H	Transitions from MMX to Floating Point instructions.
FP_MMX_TRANS.TO_MMX	
EventSel=CCH, UMask=02H	Transitions from Floating Point to MMX instructions.
FP_MMX_TRANS.ANY	
EventSel=CCH, UMask=03H	All Floating Point to and from MMX transitions.
MACRO_INSTS.DECODED	
EventSel=D0H, UMask=01H	Instructions decoded.
UOPS_DECODED.STALL_CYCLES	
EventSel=D1H, UMask=01H, Invert=1, CMask=1	Cycles no Uops are decoded.
UOPS_DECODED.MS_CYCLES_ACTIVE	
EventSel=D1H, UMask=02H, CMask=1	Uops decoded by Microcode Sequencer.
UOPS_DECODED.ESP_FOLDING	
EventSel=D1H, UMask=04H	Stack pointer instructions decoded.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
UOPS_DECODED.ESP_SYNC	
EventSel=D1H, UMask=08H	Stack pointer sync operations.
RAT_STALLS.FLAGS	
EventSel=D2H, UMask=01H	Flag stall cycles.
RAT_STALLS.REGISTERS	
EventSel=D2H, UMask=02H	Partial register stall cycles.
RAT_STALLS.ROB_READ_PORT	
EventSel=D2H, UMask=04H	ROB read port stalls cycles.
RAT_STALLS.SCOREBOARD	
EventSel=D2H, UMask=08H	Scoreboard stall cycles.
RAT_STALLS.ANY	
EventSel=D2H, UMask=0FH	All RAT stall cycles.
SEG_RENAME_STALLS	
EventSel=D4H, UMask=01H	Segment rename stall cycles.
ES_REG_RENAMES	
EventSel=D5H, UMask=01H	ES segment renames.
UOP_UNFUSION	
EventSel=DBH, UMask=01H	Uop unfusions due to FP exceptions.
BR_INST_DECODED	
EventSel=E0H, UMask=01H	Branch instructions decoded.
BPU_MISSED_CALL_RET	
EventSel=E5H, UMask=01H	Branch prediction unit missed call or return.
BACLEAR.CLEAR	
EventSel=E6H, UMask=01H	BACLEAR asserted, regardless of cause .
BACLEAR.BAD_TARGET	
EventSel=E6H, UMask=02H	BACLEAR asserted with bad target address.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
BPU_CLEARS.EARLY	
EventSel=E8H, UMask=01H	Early Branch Prediciton Unit clears.
BPU_CLEARS.LATE	
EventSel=E8H, UMask=02H	Late Branch Prediction Unit clears.
L2_TRANSACTIONS.LOAD	
EventSel=F0H, UMask=01H	L2 Load transactions.
L2_TRANSACTIONS.RFO	
EventSel=F0H, UMask=02H	L2 RFO transactions.
L2_TRANSACTIONS.IFETCH	
EventSel=F0H, UMask=04H	L2 instruction fetch transactions.
L2_TRANSACTIONS.PREFETCH	
EventSel=F0H, UMask=08H	L2 prefetch transactions.
L2_TRANSACTIONS.L1D_WB	
EventSel=F0H, UMask=10H	L1D writeback to L2 transactions.
L2_TRANSACTIONS.FILL	
EventSel=F0H, UMask=20H	L2 fill transactions.
L2_TRANSACTIONS.WB	
EventSel=F0H, UMask=40H	L2 writeback to LLC transactions.
L2_TRANSACTIONS.ANY	
EventSel=F0H, UMask=80H	All L2 transactions.
L2_LINES_IN.S_STATE	
EventSel=F1H, UMask=02H	L2 lines allocated in the S state.
L2_LINES_IN.E_STATE	
EventSel=F1H, UMask=04H	L2 lines allocated in the E state.
L2_LINES_IN.ANY	
EventSel=F1H, UMask=07H	L2 lines alloacated.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
L2_LINES_OUT.DEMAND_CLEAN	
EventSel=F2H, UMask=01H	L2 lines evicted by a demand request.
L2_LINES_OUT.DEMAND_DIRTY	
EventSel=F2H, UMask=02H	L2 modified lines evicted by a demand request.
L2_LINES_OUT.PREFETCH_CLEAN	
EventSel=F2H, UMask=04H	L2 lines evicted by a prefetch request.
L2_LINES_OUT.PREFETCH_DIRTY	
EventSel=F2H, UMask=08H	L2 modified lines evicted by a prefetch request.
L2_LINES_OUT.ANY	
EventSel=F2H, UMask=0FH	L2 lines evicted.
SQ_MISC.SPLIT_LOCK	
EventSel=F4H, UMask=10H	Super Queue lock splits across a cache line.
SQ_FULL_STALL_CYCLES	
EventSel=F6H, UMask=01H	Super Queue full stall cycles.
FP_ASSIST.ALL	
EventSel=F7H, UMask=01H, Precise	X87 Floating point assists (Precise Event).
FP_ASSIST.OUTPUT	
EventSel=F7H, UMask=02H, Precise	X87 Floating point assists for invalid output value (Precise Event).
FP_ASSIST.INPUT	
EventSel=F7H, UMask=04H, Precise	X87 Floating poiint assists for invalid input value (Precise Event).
SIMD_INT_64.PACKED_MPY	
EventSel=FDH, UMask=01H	SIMD integer 64 bit packed multiply operations.
SIMD_INT_64.PACKED_SHIFT	
EventSel=FDH, UMask=02H	SIMD integer 64 bit shift operations.
SIMD_INT_64.PACK	
EventSel=FDH, UMask=04H	SIMD integer 64 bit pack operations.

Table 11: Performance Events In the Processor Core for Nehalem Microarchitecture - Intel® Core™ i7 Processor and Intel® Xeon®® Processor 5500 Series (06_1AH, 06_1EH, 06_1FH, 06_2EH)

Event Name	
Configuration	Description
SIMD_INT_64.UNPACK	
EventSel=FDH, UMask=08H	SIMD integer 64 bit unpack operations.
SIMD_INT_64.PACKED_LOGICAL	
EventSel=FDH, UMask=10H	SIMD integer 64 bit logical operations.
SIMD_INT_64.PACKED_ARITH	
EventSel=FDH, UMask=20H	SIMD integer 64 bit arithmetic operations.
SIMD_INT_64.SHUFFLE_MOVE	
EventSel=FDH, UMask=40H	SIMD integer 64 bit shuffle/move operations.

Performance monitoring Intel® Xeon® Phi™ Processors

Performance Monitoring Events based on Knights Landing Microarchitecture - Intel® Xeon® Phi™ Processor 3200, 5200, 7200 Series

Intel® Xeon® Phi™ processors 3200/5200/7200 series are based on the Knights Landing Microarchitecture.Performance-monitoring events in the processor core are listed in the table below.

Table 12: Performance Events of the Processor Core Supported by Knights Landing Microarchitecture (06_57H)

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed	This event counts the number of instructions that retire. For instructions that consist of multiple micro-ops, this event counts exactly once, as the last micro-op of the instruction retires. The event continues counting while instructions retire, including during interrupt service routines caused by hardware interrupts, faults or traps.
CPU_CLK_UNHALTED.THREAD	
Architectural, Fixed	This event counts the number of core cycles while the thread is not in a halt state. The thread enters the halt state when it is running the HLT instruction. This event is a component in many key event ratios. The core frequency may change from time to time due to transitions associated with Enhanced Intel SpeedStep Technology or TM2. For this reason this event may have a changing ratio with regards to time. When the core frequency is constant, this event can approximate elapsed time while the core was not in the halt state. It is counted on a dedicated fixed counter .
CPU_CLK_UNHALTED.REF_TSC	
Architectural, Fixed	Fixed Counter: Counts the number of unhalted reference clock cycles.
RECYCLEQ.LD_BLOCK_ST_FORWARD	
EventSel=03H, UMask=01H, Precise	Counts the number of occurrences a retired load gets blocked because its address partially overlaps with a store.
RECYCLEQ.LD_BLOCK_STD_NOTREADY	
EventSel=03H, UMask=02H	Counts the number of occurrences a retired load gets blocked because its address overlaps with a store whose data is not ready.

Table 12: Performance Events of the Processor Core Supported by Knights Landing Microarchitecture (06_57H)

Event Name	
Configuration	Description
RECYCLEQ.ST_SPLITS	
EventSel=03H, UMask=04H	This event counts the number of retired store that experienced a cache line boundary split(Precise Event). Note that each spilt should be counted only once.
RECYCLEQ.LD_SPLITS	
EventSel=03H, UMask=08H, Precise	Counts the number of occurrences a retired load that is a cache line split. Each split should be counted only once.
RECYCLEQ.LOCK	
EventSel=03H, UMask=10H	Counts all the retired locked loads. It does not include stores because we would double count if we count stores.
RECYCLEQ.STA_FULL	
EventSel=03H, UMask=20H	Counts the store micro-ops retired that were pushed in the rehad queue because the store address buffer is full.
RECYCLEQ.ANY_LD	
EventSel=03H, UMask=40H	Counts any retired load that was pushed into the recycle queue for any reason.
RECYCLEQ.ANY_ST	
EventSel=03H, UMask=80H	Counts any retired store that was pushed into the recycle queue for any reason.
MEM_UOPS_RETIRED.L1_MISS_LOADS	
EventSel=04H, UMask=01H	This event counts the number of load micro-ops retired that miss in L1 Data cache. Note that prefetch misses will not be counted. .
MEM_UOPS_RETIRED.L2_HIT_LOADS	
EventSel=04H, UMask=02H, Precise	Counts the number of load micro-ops retired that hit in the L2.
MEM_UOPS_RETIRED.L2_MISS_LOADS	
EventSel=04H, UMask=04H, Precise	Counts the number of load micro-ops retired that miss in the L2.
MEM_UOPS_RETIRED.DTLB_MISS_LOADS	
EventSel=04H, UMask=08H, Precise	Counts the number of load micro-ops retired that cause a DTLB miss.
MEM_UOPS_RETIRED.UTLB_MISS_LOADS	
EventSel=04H, UMask=10H	Counts the number of load micro-ops retired that caused micro TLB miss.

Table 12: Performance Events of the Processor Core Supported by Knights Landing Microarchitecture (06_57H)

Event Name	
Configuration	Description
MEM_UOPS_RETIRED.HITM	
EventSel=04H, UMask=20H, Precise	Counts the loads retired that get the data from the other core in the same tile in M state.
MEM_UOPS_RETIRED.ALL_LOADS	
EventSel=04H, UMask=40H	This event counts the number of load micro-ops retired.
MEM_UOPS_RETIRED.ALL_STORES	
EventSel=04H, UMask=80H	This event counts the number of store micro-ops retired.
PAGE_WALKS.D_SIDE_WALKS	
EventSel=05H, UMask=01H, EdgeDetect=1	Counts the total D-side page walks that are completed or started. The page walks started in the speculative path will also be counted.
PAGE_WALKS.D_SIDE_CYCLES	
EventSel=05H, UMask=01H	Counts the total number of core cycles for all the D-side page walks. The cycles for page walks started in speculative path will also be included.
PAGE_WALKS.I_SIDE_WALKS	
EventSel=05H, UMask=02H, EdgeDetect=1	Counts the total I-side page walks that are completed.
PAGE_WALKS.I_SIDE_CYCLES	
EventSel=05H, UMask=02H	This event counts every cycle when an I-side (walks due to an instruction fetch) page walk is in progress. .
PAGE_WALKS.WALKS	
EventSel=05H, UMask=03H, EdgeDetect=1	Counts the total page walks that are completed (I-side and D-side).
PAGE_WALKS.CYCLES	
EventSel=05H, UMask=03H	This event counts every cycle when a data (D) page walk or instruction (I) page walk is in progress.
L2_REQUESTS.MISS	
EventSel=2EH, UMask=41H, Architectural	Counts the number of L2 cache misses.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	Counts the number of L2 cache misses.

Table 12: Performance Events of the Processor Core Supported by Knights Landing Microarchitecture (06_57H)

Event Name	
Configuration	Description
L2_REQUESTS.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	Counts the total number of L2 cache references.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	Counts the total number of L2 cache references.
L2_REQUESTS_REJECT.ALL	
EventSel=30H, UMask=00H	Counts the number of MEC requests from the L2Q that reference a cache line (cacheable requests) excluding SW prefetches filling only to L2 cache and L1 evictions (automatically excludes L2HWP, UC, WC) that were rejected - Multiple repeated rejects should be counted multiple times.
CORE_REJECT_L2Q.ALL	
EventSel=31H, UMask=00H	Counts the number of MEC requests that were not accepted into the L2Q because of any L2 queue reject condition. There is no concept of at-ret here. It might include requests due to instructions in the speculative path.
CPU_CLK_UNHALTED.THREAD_P	
EventSel=3CH, UMask=00H, Architectural	Counts the number of unhalted core clock cycles.
CPU_CLK_UNHALTED.REF	
EventSel=3CH, UMask=01H, Architectural	Counts the number of unhalted reference clock cycles.
L2_PREFETCHER.ALLOC_XQ	
EventSel=3EH, UMask=04H	Counts the number of L2HWP allocated into XQ GP.
ICACHE.HIT	
EventSel=80H, UMask=01H	Counts all instruction fetches that hit the instruction cache.
ICACHE.MISSES	
EventSel=80H, UMask=02H	Counts all instruction fetches that miss the instruction cache or produce memory requests. An instruction fetch miss is counted only once and not once for every cycle it is outstanding.
ICACHE.ACCESSES	
EventSel=80H, UMask=03H	Counts all instruction fetches, including uncacheable fetches.
FETCH_STALL.ICACHE_FILL_PENDING_CYCLES	
EventSel=86H, UMask=04H	This event counts the number of core cycles the fetch stalls because of an icache miss. This is a cumulative count of cycles the NIP stalled for all icache misses. .

Table 12: Performance Events of the Processor Core Supported by Knights Landing Microarchitecture (06_57H)

Event Name	
Configuration	Description
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural	Counts the total number of instructions retired.
UOPS_RETIRED.MS	
EventSel=C2H, UMask=01H	This event counts the number of micro-ops retired that were supplied from MSROM.
UOPS_RETIRED.ALL	
EventSel=C2H, UMask=10H	This event counts the number of micro-ops (uops) retired. The processor decodes complex macro instructions into a sequence of simpler uops. Most instructions are composed of one or two uops. Some instructions are decoded into longer sequences such as repeat instructions, floating point transcendental instructions, and assists. .
UOPS_RETIRED.SCALAR_SIMD	
EventSel=C2H, UMask=20H	This event is defined at the micro-op level and not instruction level. Most instructions are implemented with one micro-op but not all.
UOPS_RETIRED.PACKED_SIMD	
EventSel=C2H, UMask=40H	The length of the packed operation (128bits, 256bits or 512bits) is not taken into account when updating the counter; all count the same (+1). Mask (k) registers are ignored. For example: a micro-op operating with a mask that only enables one element or even zero elements will still trigger this counter (+1) This event is defined at the micro-op level and not instruction level. Most instructions are implemented with one micro-op but not all.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=01H	Counts the number of times that the machine clears due to program modifying data within 1K of a recently fetched code page.
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	Counts the number of times the machine clears due to memory ordering hazards.
MACHINE_CLEARS.FP_ASSIST	
EventSel=C3H, UMask=04H	This event counts the number of times that the pipeline stalled due to FP operations needing assists.

Table 12: Performance Events of the Processor Core Supported by Knights Landing Microarchitecture (06_57H)

Event Name	
Configuration	Description
MACHINE_CLEARS.ALL	
EventSel=C3H, UMask=08H	Counts all machine clears.
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	Counts the number of branch instructions retired.
BR_INST_RETIRED.JCC	
EventSel=C4H, UMask=7EH, Precise	Counts the number of branch instructions retired that were conditional jumps.
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=BFH, Precise	Counts the number of far branch instructions retired.
BR_INST_RETIRED.NON_RETURN_IND	
EventSel=C4H, UMask=EBH, Precise	Counts the number of branch instructions retired that were near indirect CALL or near indirect JMP.
BR_INST_RETIRED.RETURN	
EventSel=C4H, UMask=F7H, Precise	Counts the number of near RET branch instructions retired.
BR_INST_RETIRED.CALL	
EventSel=C4H, UMask=F9H, Precise	Counts the number of near CALL branch instructions retired.
BR_INST_RETIRED.IND_CALL	
EventSel=C4H, UMask=FBH, Precise	Counts the number of near indirect CALL branch instructions retired.
BR_INST_RETIRED.REL_CALL	
EventSel=C4H, UMask=FDH, Precise	Counts the number of near relative CALL branch instructions retired.
BR_INST_RETIRED.TAKEN_JCC	
EventSel=C4H, UMask=FEH, Precise	Counts the number of branch instructions retired that were taken conditional jumps.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	Counts the number of mispredicted branch instructions retired.

Table 12: Performance Events of the Processor Core Supported by Knights Landing Microarchitecture (06_57H)

Event Name	
Configuration	Description
BR_MISP_RETIRED.JCC	
EventSel=C5H, UMask=7EH, Precise	Counts the number of mispredicted branch instructions retired that were conditional jumps.
BR_MISP_RETIRED.FAR_BRANCH	
EventSel=C5H, UMask=BFH, Precise	Counts the number of mispredicted far branch instructions retired.
BR_MISP_RETIRED.NON_RETURN_IND	
EventSel=C5H, UMask=EBH, Precise	Counts the number of mispredicted branch instructions retired that were near indirect CALL or near indirect JMP.
BR_MISP_RETIRED.RETURN	
EventSel=C5H, UMask=F7H, Precise	Counts the number of mispredicted near RET branch instructions retired.
BR_MISP_RETIRED.CALL	
EventSel=C5H, UMask=F9H, Precise	Counts the number of mispredicted near CALL branch instructions retired.
BR_MISP_RETIRED.IND_CALL	
EventSel=C5H, UMask=FBH, Precise	Counts the number of mispredicted near indirect CALL branch instructions retired.
BR_MISP_RETIRED.REL_CALL	
EventSel=C5H, UMask=FDH, Precise	Counts the number of mispredicted near relative CALL branch instructions retired.
BR_MISP_RETIRED.TAKEN_JCC	
EventSel=C5H, UMask=FEH, Precise	Counts the number of mispredicted branch instructions retired that were taken conditional jumps.
NO_ALLOC_CYCLES.ROB_FULL	
EventSel=CAH, UMask=01H	Counts the number of core cycles when no micro-ops are allocated and the ROB is full.
NO_ALLOC_CYCLES.MISPREDICTS	
EventSel=CAH, UMask=04H	This event counts the number of core cycles when no uops are allocated and the alloc pipe is stalled waiting for a mispredicted branch to retire.

Table 12: Performance Events of the Processor Core Supported by Knights Landing Microarchitecture (06_57H)

Event Name	
Configuration	Description
NO_ALLOC_CYCLES.RAT_STALL	
EventSel=CAH, UMask=20H	Counts the number of core cycles when no micro-ops are allocated and a RATstall (caused by reservation station full) is asserted. .
NO_ALLOC_CYCLES.ALL	
EventSel=CAH, UMask=7FH	Counts the total number of core cycles when no micro-ops are allocated for any reason.
NO_ALLOC_CYCLES.NOT_DELIVERED	
EventSel=CAH, UMask=90H	This event counts the number of core cycles when no uops are allocated, the instruction queue is empty and the alloc pipe is stalled waiting for instructions to be fetched.
RS_FULL_STALL.MEC	
EventSel=CBH, UMask=01H	Counts the number of core cycles when allocation pipeline is stalled and is waiting for a free MEC reservation station entry.
RS_FULL_STALL.ALL	
EventSel=CBH, UMask=1FH	Counts the total number of core cycles allocation pipeline is stalled when any one of the reservation stations is full.
CYCLES_DIV_BUSY.ALL	
EventSel=CDH, UMask=01H	This event counts cycles when the divider is busy. More specifically cycles when the divide unit is unable to accept a new divide uop because it is busy processing a previously dispatched uop. The cycles will be counted irrespective of whether or not another divide uop is waiting to enter the divide unit (from the RS). This event counts integer divides, x87 divides, divss, divsd, sqrtss, sqrtsd event and does not count vector divides.
BACLEARS.ALL	
EventSel=E6H, UMask=01H	Counts the number of times the front end resteers for any branch as a result of another branch handling mechanism in the front end.
BACLEARS.RETURN	
EventSel=E6H, UMask=08H	Counts the number of times the front end resteers for RET branches as a result of another branch handling mechanism in the front end.

Table 12: Performance Events of the Processor Core Supported by Knights Landing Microarchitecture (06_57H)

Event Name	
Configuration	Description
BACLEARS.COND	
EventSel=E6H, UMask=10H	Counts the number of times the front end resteers for conditional branches as a result of another branch handling mechanism in the front end.
MS_DECODED.MS_ENTRY	
EventSel=E7H, UMask=01H	Counts the number of times the MSROM starts a flow of uops.

Performance Monitoring Events based on Knights Corner Microarchitecture

Intel® Microarchitecture code named Knights Corner are based on the Knights Corner Microarchitecture.Performance-monitoring events in the processor core are listed in the table below.

Table 13: Performance Events of the Processor Core Supported by Knights Corner Microarchitecture (06_57H)

Event Name	
Configuration	Description
DATA_READ	
EventSel=00H, UMask=00H, AnyThread=1	Number of memory data reads which hit the internal data cache (L1). Cache accesses resulting from prefetch instructions are included.
VPU_DATA_READ	
EventSel=00H, UMask=20H, AnyThread=1	Number of read transactions that were issued. In general each read transaction will read 1 64B cacheline. If there are alignment issues, then reads against multiple cache lines will each be counted individually.
DATA_WRITE	
EventSel=01H, UMask=00H, AnyThread=1	Number of memory data writes which hit the internal data cache (L1).
VPU_DATA_WRITE	
EventSel=01H, UMask=20H, AnyThread=1	Number of write transactions that were issued. In general each write transaction will write 1 64B cacheline. If there are alignment issues, then write against multiple cache lines will each be counted individually.
DATA_PAGE_WALK	
EventSel=02H, UMask=00H, AnyThread=1	Counts misses in the L1 TLB, at the hardware thread level. TLB Misses could have been caused by either demand data loads and stores or data prefetches.
DATA_READ_MISS	
EventSel=03H, UMask=00H, AnyThread=1	Number of memory read accesses that miss the internal data cache whether or not the access is cacheable or noncacheable. Cache accesses resulting from prefetch instructions are included.
VPU_DATA_READ_MISS	
EventSel=03H, UMask=20H, AnyThread=1	VPU L1 data cache readmiss. Counts the number of occurrences.
DATA_WRITE_MISS	
EventSel=04H, UMask=00H, AnyThread=1	Number of memory write accesses that miss the internal data cache whether or not the access is cacheable or noncacheable.

Table 13: Performance Events of the Processor Core Supported by Knights Corner Microarchitecture (06_57H)

Event Name	
Configuration	Description
VPU_DATA_WRITE_MISS	
EventSel=04H, UMask=20H, AnyThread=1	VPU L1 data cache write miss. Counts the number of occurrences.
VPU_STALL_REG	
EventSel=05H, UMask=20H, AnyThread=1	VPU stall on Register Dependency. Counts the number of occurrences. Dependencies will include RAW, WAW, WAR.
DATA_CACHE_LINES_WRITTEN_BACK	
EventSel=06H, UMask=00H, AnyThread=1	Number of dirty lines (all) that are written back, regardless of the cause.
MEMORY_ACCESSES_IN_BOTH_PIPES	
EventSel=09H, UMask=00H, AnyThread=1	Number of data memory reads or writes that are paired in both pipes of the pipeline.
BANK_CONFLICTS	
EventSel=0AH, UMask=00H, AnyThread=1	Number of actual bank conflicts.
CODE_READ	
EventSel=0CH, UMask=00H, AnyThread=1	Number of instruction reads; whether the read is cacheable or noncacheable.
L1_DATA_PF1	
EventSel=11H, UMask=00H, AnyThread=1	Counts software prefetches that are intended for the local L1 cache. May include both L1 and L2 prefetches. This event counts at the hardware thread level.
BRANCHES	
EventSel=12H, UMask=00H, AnyThread=1	Number of taken and not taken branches, including: conditional branches, jumps, calls, returns, software interrupts, and interrupt returns.
PIPELINE_FLUSHES	
EventSel=15H, UMask=00H, AnyThread=1	Number of pipeline flushes that occur.
INSTRUCTIONS_EXECUTED	
EventSel=16H, UMask=00H, AnyThread=1	Counts the number of instructions executed by a hardware thread. This event includes INSTRUCTIONS_EXECUTED_V_PIPE and VPU_INSTRUCTIONS_EXECUTED.

Table 13: Performance Events of the Processor Core Supported by Knights Corner Microarchitecture (06_57H)

Event Name	
Configuration	Description
VPU_INSTRUCTIONS_EXECUTED	
EventSel=16H, UMask=20H, AnyThread=1	Counts the number of VPU instructions executed by a hardware thread. This event is a subset of INSTRUCTIONS_EXECUTED.
INSTRUCTIONS_EXECUTED_V_PIPE	
EventSel=17H, UMask=00H, AnyThread=1	Counts the number of instructions executed on the alternate pipeline, called the V-pipe. Two instructions can be executed every clock cycle, one on the U-pipe, and one on the V-pipe. The V-pipe cannot execute all instruction types, and will execute instructions only when pairing rules are met. This event can be used to see the extent of instruction pairing on a workload. It is included in INSTRUCTIONS_EXECUTED. It counts at the hardware thread level.
VPU_INSTRUCTIONS_EXECUTED_V_PIPE	
EventSel=17H, UMask=20H, AnyThread=1	Counts the number of VPU instructions that paired and executed in the v-pipe.
VPU_ELEMENTS_ACTIVE	
EventSel=18H, UMask=20H, AnyThread=1	Increments by 1 for every element to which an executed VPU instruction applies. For example, if a VPU instruction executes with a mask register containing 1, it applies to only one element and so this event increments by 1. If a VPU instruction executes with a mask register containing 0xFF, this event is incremented by 8. Counts at the hardware thread level.
L1_DATA_PF1_MISS	
EventSel=1CH, UMask=00H, AnyThread=1	Counts software prefetches that missed the local L1 cache. May include both L1 and L2 prefetches. This event counts at the hardware thread level.
PIPELINE_AGI_STALLS	
EventSel=1FH, UMask=00H, AnyThread=1	Number of address generation interlock (AGI) stalls. An AGI occurring in both the U- and V- pipelines in the same clock signals this event twice.
L1_DATA_HIT_INFLIGHT_PF1	
EventSel=20H, UMask=00H, AnyThread=1	Counts demand data loads and stores that missed the L1 cache, but did hit a prefetch buffer. This means the cacheline was already in the process of being prefetched into L1. This is a second type of miss and is not included in DATA_READ_MISS_OR_WRITE_MISS. It is counted at the hardware thread level. This event does not count data cache misses due to hardware or software prefetches.

Table 13: Performance Events of the Processor Core Supported by Knights Corner Microarchitecture (06_57H)

Event Name	
Configuration	Description
PIPELINE_SG_AGI_STALLS	
EventSel=21H, UMask=00H, AnyThread=1	Number of address generation interlock (AGI) stalls due to vscatter* and vgather* instructions.
HARDWARE_INTERRUPTS	
EventSel=27H, UMask=00H, AnyThread=1	Number of taken INTR and NMI interrupts.
DATA_READ_OR_WRITE	
EventSel=28H, UMask=00H, AnyThread=1	Counts demand data loads and stores, at the hardware thread level. This event could also be referred to as L1 data cache accesses. This event does not count data cache accesses due to hardware or software prefetches. It does include VPU loads generated by instructions like vgather/vloadunpack/etc. VPU_DATA_READ and VPU_DATA_WRITE are subsets of this event.
DATA_READ_MISS_OR_WRITE_MISS	
EventSel=29H, UMask=00H, AnyThread=1	Counts demand data loads and stores that missed the L1 cache, at the hardware thread level. This event does not include misses for cachelines that were in the process of being prefetched into L1. This event does not count data cache misses due to hardware or software prefetches.
CPU_CLK_UNHALTED	
EventSel=2AH, UMask=00H, AnyThread=1	The number of cycles (commonly known as clockticks) where any thread on a core is active. A core is active if any thread on that core is not halted. This event is counted at the core level – at any given time, all the hardware threads running on the same core will have the same value.
BRANCHES_MISPREDICTED	
EventSel=2BH, UMask=00H, AnyThread=1	Number of branch mispredictions that occurred on BTB hits. BTB misses are not considered branch mispredicts because no prediction exists for them yet.
MICROCODE_CYCLES	
EventSel=2CH, UMask=00H, AnyThread=1	The number of cycles microcode is executing. While microcode is executing, all other threads are stalled.
FE_STALLED	
EventSel=2DH, UMask=00H, AnyThread=1	Number of cycles where the front-end could not advance. Any multi-cycle instructions which delay pipeline advance and apply backpressure to the front-end will be included, e.g. read-modify-write instructions. Includes cycles when the front-end did not hav.

Table 13: Performance Events of the Processor Core Supported by Knights Corner Microarchitecture (06_57H)

Event Name	
Configuration	Description
EXEC_STAGE_CYCLES	
EventSel=2EH, UMask=00H, AnyThread=1	Counts the number of cycles where an instruction was in execution stage, except in the FP or VPU execution units. Counts at the hardware thread level.
L1_DATA_PF2	
EventSel=37H, UMask=00H, AnyThread=1	Number of data vprefetch0, vprefetch1 and vprefetch2 requests seen by the L1. This is not necessarily the same number as seen by the L2 because this count includes requests that are dropped by the core.
LONG_DATA_PAGE_WALK	
EventSel=3AH, UMask=00H, AnyThread=1	Counts misses in the L2 TLB, at the hardware thread level. TLB Misses could have been caused by either demand data loads and stores or data prefetches.
HWP_L2MISS	
EventSel=C4H, UMask=10H, AnyThread=1	Counts hardware prefetches that missed the L2 data cache. This event counts at the hardware thread level.
L2_READ_HIT_E	
EventSel=C8H, UMask=10H, AnyThread=1	Counts data loads that hit a cacheline in Exclusive state in the local L2 cache. This event counts at the hardware thread level. It includes L2 prefetches and so is not useful for determining standard metrics like L2 Hit/Miss rate that are normally based on demand accesses.
L2_READ_HIT_M	
EventSel=C9H, UMask=10H, AnyThread=1	Counts data loads that hit a cacheline in Modified state in the local L2 cache. This event counts at the hardware thread level. It includes L2 prefetches and so is not useful for determining standard metrics like L2 Hit/Miss rate that are normally based on demand accesses.
L2_READ_HIT_S	
EventSel=CAH, UMask=10H, AnyThread=1	Counts data loads that hit a cacheline in Shared state in the local L2 cache. This event counts at the hardware thread level. It includes L2 prefetches and so is not useful for determining standard metrics like L2 Hit/Miss rate that are normally based on demand accesses.

Table 13: Performance Events of the Processor Core Supported by Knights Corner Microarchitecture (06_57H)

Event Name	
Configuration	Description
L2_READ_MISS	
EventSel=CBH, UMask=10H, AnyThread=1	Counts data loads that missed the local L2 cache, at the hardware thread level. It includes L2 prefetches that missed the local L2 cache and so is not useful for determining standard metrics like L2 Hit/Miss rate that are normally based on demand misses.
L2_WRITE_HIT	
EventSel=CCH, UMask=10H, AnyThread=1	L2 Write HIT.
L2_STRONGLY_ORDERED_STREAMING_VSTORES_MISS	
EventSel=CEH, UMask=10H	Number of strongly ordered streaming vector stores that missed the L2 and were sent to the ring.
L2_WEAKLY_ORDERED_STREAMING_VSTORE_MISS	
EventSel=CFH, UMask=10H	Number of weakly ordered streaming vector stores that missed the L2 and were sent to the ring.
L2_VICTIM_REQ_WITH_DATA	
EventSel=D7H, UMask=10H, AnyThread=1	Counts the number of modified cachelines evicted from the L2 Data cache. These result in a memory write operation, also known as an explicit L2 write-back. This event counts at the hardware core level; at any given time, every executing hardware thread on the core has the same value for this counter.
SNP_HIT_L2	
EventSel=E6H, UMask=10H, AnyThread=1	Snoop HIT in L2.
SNP_HITM_L2	
EventSel=E7H, UMask=10H, AnyThread=1	Counts incoming snoops that hit a modified cacheline in a hardware thread's local L2. These result in a cache-to-cache transfer: the line will be evicted from the local L2, written back to memory (also called an implicit write-back), and the line will be loaded exclusively into the requesting core's cache. This event counts at the hardware core level; at any given time, every executing hardware thread on the core has the same value for this counter.
L2_DATA_READ_MISS_CACHE_FILL	
EventSel=F1H, UMask=10H, AnyThread=1	Counts data loads that missed the local L2 cache, but were serviced by a remote L2 cache on the same Intel Xeon Phi coprocessor. This event counts at the hardware thread level. It includes L2 prefetches that missed the local L2 cache and so is not useful for determining demand cache fills.

Table 13: Performance Events of the Processor Core Supported by Knights Corner Microarchitecture (06_57H)

Event Name	
Configuration	Description
L2_DATA_WRITE_MISS_CACHE_FILL	
EventSel=F2H, UMask=10H, AnyThread=1	Counts data Reads for Ownership (due to a store operation) that missed the local L2 cache, but were serviced by a remote L2 cache on the same Intel Xeon Phi coprocessor. This event counts at the hardware thread level.
L2_DATA_READ_MISS_MEM_FILL	
EventSel=F6H, UMask=10H, AnyThread=1	Counts data loads that missed the local L2 cache, and were serviced from memory (on the same Intel Xeon Phi coprocessor). This event counts at the hardware thread level. It includes L2 prefetches that missed the local L2 cache and so is not useful for determining demand cache fills or standard metrics like L2 Hit/Miss Rate.
L2_DATA_WRITE_MISS_MEM_FILL	
EventSel=F7H, UMask=10H, AnyThread=1	Counts data Reads for Ownership (due to a store operation) that missed the local L2 cache, and were serviced from memory (on the same Intel Xeon Phi coprocessor). This event counts at the hardware thread level.
L2_DATA_PF2	
EventSel=FCH, UMask=10H, AnyThread=1	Counts software prefetches that are intended for the local L2 cache. May include both L1 and L2 prefetches. This event counts at the hardware thread level.
L2_DATA_PF2_MISS	
EventSel=FDH, UMask=10H, AnyThread=1	Counts software prefetches that missed the local L2 cache. May include both L1 and L2 prefetches. This event counts at the hardware thread level.

Performance Monitoring Intel® Atom™ Processors

Performance Monitoring Events based on Goldmont Plus Microarchitecture

Next Generation Intel Atom processors based on the Goldmont Plus Microarchitecture support the performance-monitoring events listed in the table below.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed, Precise	Counts the number of instructions that retire execution. For instructions that consist of multiple uops, this event counts the retirement of the last uop of the instruction. The counter continues counting during hardware interrupts, traps, and inside interrupt handlers. This event uses fixed counter 0. You cannot collect a PEBs record for this event.
CPU_CLK_UNHALTED.CORE	
Architectural, Fixed	Counts the number of core cycles while the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. In mobile systems the core frequency may change from time to time. For this reason this event may have a changing ratio with regards to time. This event uses fixed counter 1. You cannot collect a PEBs record for this event.
CPU_CLK_UNHALTED.REF_TSC	
Architectural, Fixed	Counts the number of reference cycles that the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. In mobile systems the core frequency may change from time. This event is not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time. This event uses fixed counter 2. You cannot collect a PEBs record for this event.
LD_BLOCKS.DATA_UNKNOWN	
EventSel=03H, UMask=01H, Precise	Counts a load blocked from using a store forward, but did not occur because the store data was not available at the right time. The forward might occur subsequently when the data is available.
LD_BLOCKS.STORE_FORWARD	
EventSel=03H, UMask=02H, Precise	Counts a load blocked from using a store forward because of an address/size mismatch, only one of the loads blocked from each store will be counted.
LD_BLOCKS.4K_ALIAS	
EventSel=03H, UMask=04H, Precise	Counts loads that block because their address modulo 4K matches a pending store.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
LD_BLOCKS.UTLB_MISS	
EventSel=03H, UMask=08H, Precise	Counts loads blocked because they are unable to find their physical address in the micro TLB (UTLB).
LD_BLOCKS.ALL_BLOCK	
EventSel=03H, UMask=10H, Precise	Counts anytime a load that retires is blocked for any reason.
DTLB_LOAD_MISSES.WALK_COMPLETED_4K	
EventSel=08H, UMask=02H	Counts page walks completed due to demand data loads (including SW prefetches) whose address translations missed in all TLB levels and were mapped to 4K pages. The page walks can end with or without a page fault.
DTLB_LOAD_MISSES.WALK_COMPLETED_2M_4M	
EventSel=08H, UMask=04H	Counts page walks completed due to demand data loads (including SW prefetches) whose address translations missed in all TLB levels and were mapped to 2M or 4M pages. The page walks can end with or without a page fault.
DTLB_LOAD_MISSES.WALK_COMPLETED_1GB	
EventSel=08H, UMask=08H	Counts page walks completed due to demand data loads (including SW prefetches) whose address translations missed in all TLB levels and were mapped to 1GB pages. The page walks can end with or without a page fault.
DTLB_LOAD_MISSES.WALK_PENDING	
EventSel=08H, UMask=10H	Counts once per cycle for each page walk occurring due to a load (demand data loads or SW prefetches). Includes cycles spent traversing the Extended Page Table (EPT). Average cycles per walk can be calculated by dividing by the number of walks.
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=00H	Counts uops issued by the front end and allocated into the back end of the machine. This event counts uops that retire as well as uops that were speculatively executed but didn't retire. The sort of speculative uops that might be counted includes, but is not limited to those uops issued in the shadow of a miss-predicted branch, those uops that are inserted during an assist (such as for a denormal floating point result), and (previously allocated) uops that might be canceled during a machine clear.
MISALIGN_MEM_REF.LOAD_PAGE_SPLIT	
EventSel=13H, UMask=02H, Precise	Counts when a memory load of a uop spans a page boundary (a split) is retired.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
MISALIGN_MEM_REF.STORE_PAGE_SPLIT	
EventSel=13H, UMask=04H, Precise	Counts when a memory store of a uop spans a page boundary (a split) is retired.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	Counts memory requests originating from the core that miss in the L2 cache.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	Counts memory requests originating from the core that reference a cache line in the L2 cache.
L2_REJECT_XQ.ALL	
EventSel=30H, UMask=00H	Counts the number of demand and prefetch transactions that the L2 XQ rejects due to a full or near full condition which likely indicates back pressure from the intra-die interconnect (IDI) fabric. The XQ may reject transactions from the L2Q (non-cacheable requests), L2 misses and L2 write-back victims.
CORE_REJECT_L2Q.ALL	
EventSel=31H, UMask=00H	Counts the number of demand and L1 prefetcher requests rejected by the L2Q due to a full or nearly full condition which likely indicates back pressure from L2Q. It also counts requests that would have gone directly to the XQ, but are rejected due to a full or nearly full condition, indicating back pressure from the IDI link. The L2Q may also reject transactions from a core to insure fairness between cores, or to delay a core's dirty eviction when the address conflicts with incoming external snoops.
CPU_CLK_UNHALTED.CORE_P	
EventSel=3CH, UMask=00H, Architectural	Core cycles when core is not halted. This event uses a (_P)rogrammable general purpose performance counter.
CPU_CLK_UNHALTED.REF	
EventSel=3CH, UMask=01H, Architectural	Reference cycles when core is not halted. This event uses a (_P)rogrammable general purpose performance counter.
DTLB_STORE_MISSES.WALK_COMPLETED_4K	
EventSel=49H, UMask=02H	Counts page walks completed due to demand data stores whose address translations missed in the TLB and were mapped to 4K pages. The page walks can end with or without a page fault.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
DTLB_STORE_MISSES.WALK_COMPLETED_2M_4M	
EventSel=49H, UMask=04H	Counts page walks completed due to demand data stores whose address translations missed in the TLB and were mapped to 2M or 4M pages. The page walks can end with or without a page fault.
DTLB_STORE_MISSES.WALK_COMPLETED_1GB	
EventSel=49H, UMask=08H	Counts page walks completed due to demand data stores whose address translations missed in the TLB and were mapped to 1GB pages. The page walks can end with or without a page fault.
DTLB_STORE_MISSES.WALK_PENDING	
EventSel=49H, UMask=10H	Counts once per cycle for each page walk occurring due to a demand data store. Includes cycles spent traversing the Extended Page Table (EPT). Average cycles per walk can be calculated by dividing by the number of walks.
EPT.WALK_PENDING	
EventSel=4FH, UMask=10H	Counts once per cycle for each page walk only while traversing the Extended Page Table (EPT), and does not count during the rest of the translation. The EPT is used for translating Guest-Physical Addresses to Physical Addresses for Virtual Machine Monitors (VMMs). Average cycles per walk can be calculated by dividing the count by number of walks. .
DL1.REPLACEMENT	
EventSel=51H, UMask=01H	Counts when a modified (dirty) cache line is evicted from the data L1 cache and needs to be written back to memory. No count will occur if the evicted line is clean, and hence does not require a writeback.
ICACHE.HIT	
EventSel=80H, UMask=01H	Counts requests to the Instruction Cache (ICache) for one or more bytes in an ICache Line and that cache line is in the ICache (hit). The event strives to count on a cache line basis, so that multiple accesses which hit in a single cache line count as one ICACHE.HIT. Specifically, the event counts when straight line code crosses the cache line boundary, or when a branch target is to a new line, and that cache line is in the ICache. This event counts differently than Intel processors based on Silvermont microarchitecture.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
ICACHE.MISSES	
EventSel=80H, UMask=02H	Counts requests to the Instruction Cache (ICache) for one or more bytes in an ICache Line and that cache line is not in the ICache (miss). The event strives to count on a cache line basis, so that multiple accesses which miss in a single cache line count as one ICACHE.MISS. Specifically, the event counts when straight line code crosses the cache line boundary, or when a branch target is to a new line, and that cache line is not in the ICache. This event counts differently than Intel processors based on Silvermont microarchitecture.
ICACHE.ACCESSES	
EventSel=80H, UMask=03H	Counts requests to the Instruction Cache (ICache) for one or more bytes in an ICache Line. The event strives to count on a cache line basis, so that multiple fetches to a single cache line count as one ICACHE.ACCESS. Specifically, the event counts when accesses from straight line code crosses the cache line boundary, or when a branch target is to a new line. This event counts differently than Intel processors based on Silvermont microarchitecture.
ITLB.MISS	
EventSel=81H, UMask=04H	Counts the number of times the machine was unable to find a translation in the Instruction Translation Lookaside Buffer (ITLB) for a linear address of an instruction fetch. It counts when new translation are filled into the ITLB. The event is speculative in nature, but will not count translations (page walks) that are begun and not finished, or translations that are finished but not filled into the ITLB.
ITLB_MISSES.WALK_COMPLETED_4K	
EventSel=85H, UMask=02H	Counts page walks completed due to instruction fetches whose address translations missed in the TLB and were mapped to 4K pages. The page walks can end with or without a page fault.
ITLB_MISSES.WALK_COMPLETED_2M_4M	
EventSel=85H, UMask=04H	Counts page walks completed due to instruction fetches whose address translations missed in the TLB and were mapped to 2M or 4M pages. The page walks can end with or without a page fault.
ITLB_MISSES.WALK_COMPLETED_1GB	
EventSel=85H, UMask=08H	Counts page walks completed due to instruction fetches whose address translations missed in the TLB and were mapped to 1GB pages. The page walks can end with or without a page fault.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
ITLB_MISSES.WALK_PENDING	
EventSel=85H, UMask=10H	Counts once per cycle for each page walk occurring due to an instruction fetch. Includes cycles spent traversing the Extended Page Table (EPT). Average cycles per walk can be calculated by dividing by the number of walks.
FETCH_STALL.ALL	
EventSel=86H, UMask=00H	Counts cycles that fetch is stalled due to any reason. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes. This will include cycles due to an ITLB miss, ICache miss and other events.
FETCH_STALL.ITLB_FILL_PENDING_CYCLES	
EventSel=86H, UMask=01H	Counts cycles that fetch is stalled due to an outstanding ITLB miss. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes due to an ITLB miss. Note: this event is not the same as page walk cycles to retrieve an instruction translation.
FETCH_STALL.ICACHE_FILL_PENDING_CYCLES	
EventSel=86H, UMask=02H	Counts cycles that fetch is stalled due to an outstanding ICache miss. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes due to an ICache miss. Note: this event is not the same as the total number of cycles spent retrieving instruction cache lines from the memory hierarchy.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
UOPS_NOT_DELIVERED.ANY	
EventSel=9CH, UMask=00H	This event used to measure front-end inefficiencies. I.e. when front-end of the machine is not delivering uops to the back-end and the back-end has is not stalled. This event can be used to identify if the machine is truly front-end bound. When this event occurs, it is an indication that the front-end of the machine is operating at less than its theoretical peak performance. Background: We can think of the processor pipeline as being divided into 2 broader parts: Front-end and Back-end. Front-end is responsible for fetching the instruction, decoding into uops in machine understandable format and putting them into a uop queue to be consumed by back end. The back-end then takes these uops, allocates the required resources. When all resources are ready, uops are executed. If the back-end is not ready to accept uops from the front-end, then we do not want to count these as front-end bottlenecks. However, whenever we have bottlenecks in the back-end, we will have allocation unit stalls and eventually forcing the front-end to wait until the back-end is ready to receive more uops. This event counts only when back-end is requesting more uops and front-end is not able to provide them. When 3 uops are requested and no uops are delivered, the event counts 3. When 3 are requested, and only 1 is delivered, the event counts 2. When only 2 are delivered, the event counts 1. Alternatively stated, the event will not count if 3 uops are delivered, or if the back end is stalled and not requesting any uops at all. Counts indicate missed opportunities for the front-end to deliver a uop to the back end. Some examples of conditions that cause front-end efficiencies are: ICache misses, ITLB misses, and decoder restrictions that limit the front-end bandwidth. Known Issues: Some uops require multiple allocation slots. These uops will not be charged as a front end 'not delivered' opportunity, and will be regarded as a back end problem. For example, the INC instruction has one uop that requires 2 issue slots. A stream of INC instructions will not count as UOPS_NOT_DELIVERED, even though only one instruction can be issued per clock. The low uop issue rate for a stream of INC instructions is considered to be a back end issue.
TLB_FLUSHES.STLB_ANY	
EventSel=BDH, UMask=20H	Counts STLB flushes. The TLBs are flushed on instructions like INVLPG and MOV to CR3.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural, Precise	Counts the number of instructions that retire execution. For instructions that consist of multiple uops, this event counts the retirement of the last uop of the instruction. The event continues counting during hardware interrupts, traps, and inside interrupt handlers. This is an architectural performance event. This event uses a (_P)rogrammable general purpose performance counter. *This event is Precise Event capable: The EventingRIP field in the PEBS record is precise to the address of the instruction which caused the event. Note: Because PEBS records can be collected only on IA32_PMC0, only one event can use the PEBS facility at a time.
INST_RETIRED.PREC_DIST	
EventSel=C0H, UMask=00H, Precise	Counts INST_RETIRED.ANY using the Reduced Skid PEBS feature that reduces the shadow in which events aren't counted allowing for a more unbiased distribution of samples across instructions retired.
UOPS_RETIRED.ANY	
EventSel=C2H, UMask=00H, Precise	Counts uops which retired.
UOPS_RETIRED.MS	
EventSel=C2H, UMask=01H, Precise	Counts uops retired that are from the complex flows issued by the micro-sequencer (MS). Counts both the uops from a micro-coded instruction, and the uops that might be generated from a micro-coded assist.
UOPS_RETIRED.FPDIV	
EventSel=C2H, UMask=08H, Precise	Counts the number of floating point divide uops retired.
UOPS_RETIRED.IDIV	
EventSel=C2H, UMask=10H, Precise	Counts the number of integer divide uops retired.
MACHINE_CLEARS.ALL	
EventSel=C3H, UMask=00H	Counts machine clears for any reason.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=01H	Counts the number of times that the processor detects that a program is writing to a code section and has to perform a machine clear because of that modification. Self-modifying code (SMC) causes a severe penalty in all Intel® architecture processors.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	Counts machine clears due to memory ordering issues. This occurs when a snoop request happens and the machine is uncertain if memory ordering will be preserved - as another core is in the process of modifying the data.
MACHINE_CLEARS.FP_ASSIST	
EventSel=C3H, UMask=04H	Counts machine clears due to floating point (FP) operations needing assists. For instance, if the result was a floating point denormal, the hardware clears the pipeline and reissues uops to produce the correct IEEE compliant denormal result.
MACHINE_CLEARS.DISAMBIGUATION	
EventSel=C3H, UMask=08H	Counts machine clears due to memory disambiguation. Memory disambiguation happens when a load which has been issued conflicts with a previous unretired store in the pipeline whose address was not known at issue time, but is later resolved to be the same as the load address.
MACHINE_CLEARS.PAGE_FAULT	
EventSel=C3H, UMask=20H	Counts the number of times that the machines clears due to a page fault. Covers both I-side and D-side(Loads/Stores) page faults. A page fault occurs when either page is not present, or an access violation.
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	Counts branch instructions retired for all branch types. This is an architectural performance event.
BR_INST_RETIRED.JCC	
EventSel=C4H, UMask=7EH, Precise	Counts retired Jcc (Jump on Conditional Code/Jump if Condition is Met) branch instructions retired, including both when the branch was taken and when it was not taken.
BR_INST_RETIRED.ALL_TAKEN_BRANCHES	
EventSel=C4H, UMask=80H, Precise	Counts the number of taken branch instructions retired.
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=BFH, Precise	Counts far branch instructions retired. This includes far jump, far call and return, and Interrupt call and return.
BR_INST_RETIRED.NON_RETURN_IND	
EventSel=C4H, UMask=EBH, Precise	Counts near indirect call or near indirect jmp branch instructions retired.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
BR_INST_RETIRED.RETURN	
EventSel=C4H, UMask=F7H, Precise	Counts near return branch instructions retired.
BR_INST_RETIRED.CALL	
EventSel=C4H, UMask=F9H, Precise	Counts near CALL branch instructions retired.
BR_INST_RETIRED.IND_CALL	
EventSel=C4H, UMask=FBH, Precise	Counts near indirect CALL branch instructions retired.
BR_INST_RETIRED.REL_CALL	
EventSel=C4H, UMask=FDH, Precise	Counts near relative CALL branch instructions retired.
BR_INST_RETIRED.TAKEN_JCC	
EventSel=C4H, UMask=FEH, Precise	Counts Jcc (Jump on Conditional Code/Jump if Condition is Met) branch instructions retired that were taken and does not count when the Jcc branch instruction were not taken.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	Counts mispredicted branch instructions retired including all branch types.
BR_MISP_RETIRED.JCC	
EventSel=C5H, UMask=7EH, Precise	Counts mispredicted retired Jcc (Jump on Conditional Code/Jump if Condition is Met) branch instructions retired, including both when the branch was supposed to be taken and when it was not supposed to be taken (but the processor predicted the opposite condition).
BR_MISP_RETIRED.NON_RETURN_IND	
EventSel=C5H, UMask=EBH, Precise	Counts mispredicted branch instructions retired that were near indirect call or near indirect jmp, where the target address taken was not what the processor predicted.
BR_MISP_RETIRED.RETURN	
EventSel=C5H, UMask=F7H, Precise	Counts mispredicted near RET branch instructions retired, where the return address taken was not what the processor predicted.
BR_MISP_RETIRED.IND_CALL	
EventSel=C5H, UMask=FBH, Precise	Counts mispredicted near indirect CALL branch instructions retired, where the target address taken was not what the processor predicted.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
BR_MISP_RETIRED.TAKEN_JCC	
EventSel=C5H, UMask=FEH, Precise	Counts mispredicted retired Jcc (Jump on Conditional Code/Jump if Condition is Met) branch instructions retired that were supposed to be taken but the processor predicted that it would not be taken.
ISSUE_SLOTS_NOT_CONSUMED.ANY	
EventSel=CAH, UMask=00H	Counts the number of issue slots per core cycle that were not consumed by the backend due to either a full resource in the backend (RESOURCE_FULL) or due to the processor recovering from some event (RECOVERY).
ISSUE_SLOTS_NOT_CONSUMED.RESOURCE_FULL	
EventSel=CAH, UMask=01H	Counts the number of issue slots per core cycle that were not consumed because of a full resource in the backend. Including but not limited to resources such as the Re-order Buffer (ROB), reservation stations (RS), load/store buffers, physical registers, or any other needed machine resource that is currently unavailable. Note that uops must be available for consumption in order for this event to fire. If a uop is not available (Instruction Queue is empty), this event will not count.
ISSUE_SLOTS_NOT_CONSUMED.RECOVERY	
EventSel=CAH, UMask=02H	Counts the number of issue slots per core cycle that were not consumed by the backend because allocation is stalled waiting for a mispredicted jump to retire or other branch-like conditions (e.g. the event is relevant during certain microcode flows). Counts all issue slots blocked while within this window including slots where uops were not available in the Instruction Queue.
HW_INTERRUPTS.RECEIVED	
EventSel=CBH, UMask=01H	Counts hardware interrupts received by the processor.
HW_INTERRUPTS.MASKED	
EventSel=CBH, UMask=02H	Counts the number of core cycles during which interrupts are masked (disabled). Increments by 1 each core cycle that EFLAGS.IF is 0, regardless of whether interrupts are pending or not.
HW_INTERRUPTS.PENDING_AND_MASKED	
EventSel=CBH, UMask=04H	Counts core cycles during which there are pending interrupts, but interrupts are masked (EFLAGS.IF = 0).
CYCLES_DIV_BUSY.ALL	
EventSel=CDH, UMask=00H	Counts core cycles if either divide unit is busy.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
CYCLES_DIV_BUSY.IDIV	
EventSel=CDH, UMask=01H	Counts core cycles the integer divide unit is busy.
CYCLES_DIV_BUSY.FPDIV	
EventSel=CDH, UMask=02H	Counts core cycles the floating point divide unit is busy.
MEM_UOPS_RETIRED.DTLB_MISS_LOADS	
EventSel=D0H, UMask=11H, Precise	Counts load uops retired that caused a DTLB miss.
MEM_UOPS_RETIRED.DTLB_MISS_STORES	
EventSel=D0H, UMask=12H, Precise	Counts store uops retired that caused a DTLB miss.
MEM_UOPS_RETIRED.DTLB_MISS	
EventSel=D0H, UMask=13H, Precise	Counts uops retired that had a DTLB miss on load, store or either. Note that when two distinct memory operations to the same page miss the DTLB, only one of them will be recorded as a DTLB miss.
MEM_UOPS_RETIRED.LOCK_LOADS	
EventSel=D0H, UMask=21H, Precise	Counts locked memory uops retired. This includes "regular" locks and bus locks. (To specifically count bus locks only, see the Offcore response event.) A locked access is one with a lock prefix, or an exchange to memory. See the SDM for a complete description of which memory load accesses are locks.
MEM_UOPS_RETIRED.SPLIT_LOADS	
EventSel=D0H, UMask=41H, Precise	Counts load uops retired where the data requested spans a 64 byte cache line boundary.
MEM_UOPS_RETIRED.SPLIT_STORES	
EventSel=D0H, UMask=42H, Precise	Counts store uops retired where the data requested spans a 64 byte cache line boundary.
MEM_UOPS_RETIRED.SPLIT	
EventSel=D0H, UMask=43H, Precise	Counts memory uops retired where the data requested spans a 64 byte cache line boundary.
MEM_UOPS_RETIRED.ALL_LOADS	
EventSel=D0H, UMask=81H, Precise	Counts the number of load uops retired.
MEM_UOPS_RETIRED.ALL_STORES	
EventSel=D0H, UMask=82H, Precise	Counts the number of store uops retired.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
MEM_UOPS_RETIRED.ALL	
EventSel=D0H, UMask=83H, Precise	Counts the number of memory uops retired that is either a loads or a store or both.
MEM_LOAD_UOPS_RETIRED.L1_HIT	
EventSel=D1H, UMask=01H, Precise	Counts load uops retired that hit the L1 data cache.
MEM_LOAD_UOPS_RETIRED.L2_HIT	
EventSel=D1H, UMask=02H, Precise	Counts load uops retired that hit in the L2 cache.
MEM_LOAD_UOPS_RETIRED.L1_MISS	
EventSel=D1H, UMask=08H, Precise	Counts load uops retired that miss the L1 data cache.
MEM_LOAD_UOPS_RETIRED.L2_MISS	
EventSel=D1H, UMask=10H, Precise	Counts load uops retired that miss in the L2 cache.
MEM_LOAD_UOPS_RETIRED.HITM	
EventSel=D1H, UMask=20H, Precise	Counts load uops retired where the cache line containing the data was in the modified state of another core or modules cache (HITM). More specifically, this means that when the load address was checked by other caching agents (typically another processor) in the system, one of those caching agents indicated that they had a dirty copy of the data. Loads that obtain a HITM response incur greater latency than most is typical for a load. In addition, since HITM indicates that some other processor had this data in its cache, it implies that the data was shared between processors, or potentially was a lock or semaphore value. This event is useful for locating sharing, false sharing, and contended locks.
MEM_LOAD_UOPS_RETIRED.WCB_HIT	
EventSel=D1H, UMask=40H, Precise	Counts memory load uops retired where the data is retrieved from the WCB (or fill buffer), indicating that the load found its data while that data was in the process of being brought into the L1 cache. Typically a load will receive this indication when some other load or prefetch missed the L1 cache and was in the process of retrieving the cache line containing the data, but that process had not yet finished (and written the data back to the cache). For example, consider load X and Y, both referencing the same cache line that is not in the L1 cache. If load X misses cache first, it obtains and WCB (or fill buffer) and begins the process of requesting the data. When load Y requests the data, it will either hit the WCB, or the L1 cache, depending on exactly what time the request to Y occurs.

Table 14: Performance Events of the Processor Core Supported by Goldmont Plus Microarchitecture

Event Name	
Configuration	Description
MEM_LOAD_UOPS_RETIRED.DRAM_HIT	
EventSel=D1H, UMask=80H, Precise	Counts memory load uops retired where the data is retrieved from DRAM. Event is counted at retirement, so the speculative loads are ignored. A memory load can hit (or miss) the L1 cache, hit (or miss) the L2 cache, hit DRAM, hit in the WCB or receive a HITM response.
BACLEARS.ALL	
EventSel=E6H, UMask=01H	Counts the number of times a BACLEAR is signaled for any reason, including, but not limited to indirect branch/call, Jcc (Jump on Conditional Code/Jump if Condition is Met) branch, unconditional branch/call, and returns.
BACLEARS.RETURN	
EventSel=E6H, UMask=08H	Counts BACLEARS on return instructions.
BACLEARS.COND	
EventSel=E6H, UMask=10H	Counts BACLEARS on Jcc (Jump on Conditional Code/Jump if Condition is Met) branches.
MS_DECODED.MS_ENTRY	
EventSel=E7H, UMask=01H	Counts the number of times the Microcode Sequencer (MS) starts a flow of uops from the MSROM. It does not count every time a uop is read from the MSROM. The most common case that this counts is when a micro-coded instruction is encountered by the front end of the machine. Other cases include when an instruction encounters a fault, trap, or microcode assist of any sort that initiates a flow of uops. The event will count MS startups for uops that are speculative, and subsequently cleared by branch mispredict or a machine clear.
DECODE_RESTRICTION.PREDECODE_WRONG	
EventSel=E9H, UMask=01H	Counts the number of times the prediction (from the predecode cache) for instruction length is incorrect.

Performance Monitoring Events based on Goldmont Microarchitecture

Next Generation Intel Atom processors based on the Goldmont Microarchitecture support the performance-monitoring events listed in the table below.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed	Counts the number of instructions that retire execution. For instructions that consist of multiple uops, this event counts the retirement of the last uop of the instruction. The counter continues counting during hardware interrupts, traps, and inside interrupt handlers. This event uses fixed counter 0. You cannot collect a PEBs record for this event.
CPU_CLK_UNHALTED.CORE	
Architectural, Fixed	Counts the number of core cycles while the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. In mobile systems the core frequency may change from time to time. For this reason this event may have a changing ratio with regards to time. This event uses fixed counter 1. You cannot collect a PEBs record for this event.
CPU_CLK_UNHALTED.REF_TSC	
Architectural, Fixed	Counts the number of reference cycles that the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. In mobile systems the core frequency may change from time. This event is not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time. This event uses fixed counter 2. You cannot collect a PEBs record for this event.
LD_BLOCKS.DATA_UNKNOWN	
EventSel=03H, UMask=01H, Precise	Counts a load blocked from using a store forward, but did not occur because the store data was not available at the right time. The forward might occur subsequently when the data is available.
LD_BLOCKS.STORE_FORWARD	
EventSel=03H, UMask=02H, Precise	Counts a load blocked from using a store forward because of an address/size mismatch, only one of the loads blocked from each store will be counted.
LD_BLOCKS.4K_ALIAS	
EventSel=03H, UMask=04H, Precise	Counts loads that block because their address modulo 4K matches a pending store.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
LD_BLOCKS.UTLB_MISS	
EventSel=03H, UMask=08H, Precise	Counts loads blocked because they are unable to find their physical address in the micro TLB (UTLB).
LD_BLOCKS.ALL_BLOCK	
EventSel=03H, UMask=10H, Precise	Counts anytime a load that retires is blocked for any reason.
PAGE_WALKS.D_SIDE_CYCLES	
EventSel=05H, UMask=01H	Counts every core cycle when a Data-side (walks due to a data operation) page walk is in progress.
PAGE_WALKS.I_SIDE_CYCLES	
EventSel=05H, UMask=02H	Counts every core cycle when a Instruction-side (walks due to an instruction fetch) page walk is in progress.
PAGE_WALKS.CYCLES	
EventSel=05H, UMask=03H	Counts every core cycle a page-walk is in progress due to either a data memory operation or an instruction fetch.
UOPS_ISSUED.ANY	
EventSel=0EH, UMask=00H	Counts uops issued by the front end and allocated into the back end of the machine. This event counts uops that retire as well as uops that were speculatively executed but didn't retire. The sort of speculative uops that might be counted includes, but is not limited to those uops issued in the shadow of a miss-predicted branch, those uops that are inserted during an assist (such as for a denormal floating point result), and (previously allocated) uops that might be canceled during a machine clear.
MISALIGN_MEM_REF.LOAD_PAGE_SPLIT	
EventSel=13H, UMask=02H, Precise	Counts when a memory load of a uop spans a page boundary (a split) is retired.
MISALIGN_MEM_REF.STORE_PAGE_SPLIT	
EventSel=13H, UMask=04H, Precise	Counts when a memory store of a uop spans a page boundary (a split) is retired.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	Counts memory requests originating from the core that miss in the L2 cache.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	Counts memory requests originating from the core that reference a cache line in the L2 cache.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
L2_REJECT_XQ.ALL	
EventSel=30H, UMask=00H	Counts the number of demand and prefetch transactions that the L2 XQ rejects due to a full or near full condition which likely indicates back pressure from the intra-die interconnect (IDI) fabric. The XQ may reject transactions from the L2Q (non-cacheable requests), L2 misses and L2 write-back victims.
CORE_REJECT_L2Q.ALL	
EventSel=31H, UMask=00H	Counts the number of demand and L1 prefetcher requests rejected by the L2Q due to a full or nearly full condition which likely indicates back pressure from L2Q. It also counts requests that would have gone directly to the XQ, but are rejected due to a full or nearly full condition, indicating back pressure from the IDI link. The L2Q may also reject transactions from a core to ensure fairness between cores, or to delay a core's dirty eviction when the address conflicts with incoming external snoops.
CPU_CLK_UNHALTED.CORE_P	
EventSel=3CH, UMask=00H, Architectural	Core cycles when core is not halted. This event uses a (_P)rogrammable general purpose performance counter.
CPU_CLK_UNHALTED.REF	
EventSel=3CH, UMask=01H, Architectural	Reference cycles when core is not halted. This event uses a programmable general purpose performance counter.
DL1.DIRTY_EVICTION	
EventSel=51H, UMask=01H	Counts when a modified (dirty) cache line is evicted from the data L1 cache and needs to be written back to memory. No count will occur if the evicted line is clean, and hence does not require a writeback.
ICACHE.HIT	
EventSel=80H, UMask=01H	Counts requests to the Instruction Cache (ICache) for one or more bytes in an ICache Line and that cache line is in the ICache (hit). The event strives to count on a cache line basis, so that multiple accesses which hit in a single cache line count as one ICACHE.HIT. Specifically, the event counts when straight line code crosses the cache line boundary, or when a branch target is to a new line, and that cache line is in the ICache. This event counts differently than Intel processors based on Silvermont microarchitecture.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
ICACHE.MISSES	
EventSel=80H, UMask=02H	Counts requests to the Instruction Cache (ICache) for one or more bytes in an ICache Line and that cache line is not in the ICache (miss). The event strives to count on a cache line basis, so that multiple accesses which miss in a single cache line count as one ICACHE.MISS. Specifically, the event counts when straight line code crosses the cache line boundary, or when a branch target is to a new line, and that cache line is not in the ICache. This event counts differently than Intel processors based on Silvermont microarchitecture.
ICACHE.ACCESSES	
EventSel=80H, UMask=03H	Counts requests to the Instruction Cache (ICache) for one or more bytes in an ICache Line. The event strives to count on a cache line basis, so that multiple fetches to a single cache line count as one ICACHE.ACCESS. Specifically, the event counts when accesses from straight line code crosses the cache line boundary, or when a branch target is to a new line. This event counts differently than Intel processors based on Silvermont microarchitecture.
ITLB.MISS	
EventSel=81H, UMask=04H	Counts the number of times the machine was unable to find a translation in the Instruction Translation Lookaside Buffer (ITLB) for a linear address of an instruction fetch. It counts when new translation are filled into the ITLB. The event is speculative in nature, but will not count translations (page walks) that are begun and not finished, or translations that are finished but not filled into the ITLB.
FETCH_STALL.ALL	
EventSel=86H, UMask=00H	Counts cycles that fetch is stalled due to any reason. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes. This will include cycles due to an ITLB miss, ICache miss and other events. .
FETCH_STALL.ITLB_FILL_PENDING_CYCLES	
EventSel=86H, UMask=01H	Counts cycles that fetch is stalled due to an outstanding ITLB miss. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes due to an ITLB miss. Note: this event is not the same as page walk cycles to retrieve an instruction translation.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
FETCH_STALL.ICACHE_FILL_PENDING_CYCLES	
EventSel=86H, UMask=02H	Counts cycles that fetch is stalled due to an outstanding ICache miss. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes due to an ICache miss. Note: this event is not the same as the total number of cycles spent retrieving instruction cache lines from the memory hierarchy.
UOPS_NOT_DELIVERED.ANY	
EventSel=9CH, UMask=00H	This event used to measure front-end inefficiencies. I.e. when front-end of the machine is not delivering uops to the back-end and the back-end has is not stalled. This event can be used to identify if the machine is truly front-end bound. When this event occurs, it is an indication that the front-end of the machine is operating at less than its theoretical peak performance. Background: We can think of the processor pipeline as being divided into 2 broader parts: Front-end and Back-end. Front-end is responsible for fetching the instruction, decoding into uops in machine understandable format and putting them into a uop queue to be consumed by back end. The back-end then takes these uops, allocates the required resources. When all resources are ready, uops are executed. If the back-end is not ready to accept uops from the front-end, then we do not want to count these as front-end bottlenecks. However, whenever we have bottlenecks in the back-end, we will have allocation unit stalls and eventually forcing the front-end to wait until the back-end is ready to receive more uops. This event counts only when back-end is requesting more uops and front-end is not able to provide them. When 3 uops are requested and no uops are delivered, the event counts 3. When 3 are requested, and only 1 is delivered, the event counts 2. When only 2 are delivered, the event counts 1. Alternatively stated, the event will not count if 3 uops are delivered, or if the back end is stalled and not requesting any uops at all. Counts indicate missed opportunities for the front-end to deliver a uop to the back end. Some examples of conditions that cause front-end efficiencies are: ICache misses, ITLB misses, and decoder restrictions that limit the front-end bandwidth. Known Issues: Some uops require multiple allocation slots. These uops will not be charged as a front end 'not delivered' opportunity, and will be regarded as a back end problem. For example, the INC instruction has one uop that requires 2 issue slots. A stream of INC instructions will not count as UOPS_NOT_DELIVERED, even though only one instruction can be issued per clock. The low uop issue rate for a stream of INC instructions is considered to be a back end issue.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural, Precise	Counts the number of instructions that retire execution. For instructions that consist of multiple uops, this event counts the retirement of the last uop of the instruction. The event continues counting during hardware interrupts, traps, and inside interrupt handlers. This is an architectural performance event. This event uses a (_P)rogrammable general purpose performance counter. *This event is Precise Event capable: The EventingRIP field in the PEBS record is precise to the address of the instruction which caused the event. Note: Because PEBS records can be collected only on IA32_PMC0, only one event can use the PEBS facility at a time.
UOPS_RETIRED.ANY	
EventSel=C2H, UMask=00H, Precise	Counts uops which retired.
UOPS_RETIRED.MS	
EventSel=C2H, UMask=01H, Precise	Counts uops retired that are from the complex flows issued by the micro-sequencer (MS). Counts both the uops from a micro-coded instruction, and the uops that might be generated from a micro-coded assist.
UOPS_RETIRED.FPDIV	
EventSel=C2H, UMask=08H, Precise	Counts the number of floating point divide uops retired.
UOPS_RETIRED.IDIV	
EventSel=C2H, UMask=10H, Precise	Counts the number of integer divide uops retired.
MACHINE_CLEARS.ALL	
EventSel=C3H, UMask=00H	Counts machine clears for any reason.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=01H	Counts the number of times that the processor detects that a program is writing to a code section and has to perform a machine clear because of that modification. Self-modifying code (SMC) causes a severe penalty in all Intel® architecture processors.
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	Counts machine clears due to memory ordering issues. This occurs when a snoop request happens and the machine is uncertain if memory ordering will be preserved as another core is in the process of modifying the data.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
MACHINE_CLEARS.FP_ASSIST	
EventSel=C3H, UMask=04H	Counts machine clears due to floating point (FP) operations needing assists. For instance, if the result was a floating point denormal, the hardware clears the pipeline and reissues uops to produce the correct IEEE compliant denormal result.
MACHINE_CLEARS.DISAMBIGUATION	
EventSel=C3H, UMask=08H	Counts machine clears due to memory disambiguation. Memory disambiguation happens when a load which has been issued conflicts with a previous unretired store in the pipeline whose address was not known at issue time, but is later resolved to be the same as the load address.
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	Counts branch instructions retired for all branch types. This is an architectural performance event.
BR_INST_RETIRED.JCC	
EventSel=C4H, UMask=7EH, Precise	Counts retired Jcc (Jump on Conditional Code/Jump if Condition is Met) branch instructions retired, including both when the branch was taken and when it was not taken.
BR_INST_RETIRED.ALL_TAKEN_BRANCHES	
EventSel=C4H, UMask=80H, Precise	Counts the number of taken branch instructions retired.
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=BFH, Precise	Counts far branch instructions retired. This includes far jump, far call and return, and Interrupt call and return.
BR_INST_RETIRED.NON_RETURN_IND	
EventSel=C4H, UMask=EBH, Precise	Counts near indirect call or near indirect jmp branch instructions retired.
BR_INST_RETIRED.RETURN	
EventSel=C4H, UMask=F7H, Precise	Counts near return branch instructions retired.
BR_INST_RETIRED.CALL	
EventSel=C4H, UMask=F9H, Precise	Counts near CALL branch instructions retired.
BR_INST_RETIRED.IND_CALL	
EventSel=C4H, UMask=FBH, Precise	Counts near indirect CALL branch instructions retired.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
BR_INST_RETIRED.REL_CALL	
EventSel=C4H, UMask=FDH, Precise	Counts near relative CALL branch instructions retired.
BR_INST_RETIRED.TAKEN_JCC	
EventSel=C4H, UMask=FEH, Precise	Counts Jcc (Jump on Conditional Code/Jump if Condition is Met) branch instructions retired that were taken and does not count when the Jcc branch instruction were not taken.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	Counts mispredicted branch instructions retired including all branch types.
BR_MISP_RETIRED.JCC	
EventSel=C5H, UMask=7EH, Precise	Counts mispredicted retired Jcc (Jump on Conditional Code/Jump if Condition is Met) branch instructions retired, including both when the branch was supposed to be taken and when it was not supposed to be taken (but the processor predicted the opposite condition).
BR_MISP_RETIRED.NON_RETURN_IND	
EventSel=C5H, UMask=EBH, Precise	Counts mispredicted branch instructions retired that were near indirect call or near indirect jmp, where the target address taken was not what the processor predicted.
BR_MISP_RETIRED.RETURN	
EventSel=C5H, UMask=F7H, Precise	Counts mispredicted near RET branch instructions retired, where the return address taken was not what the processor predicted.
BR_MISP_RETIRED.IND_CALL	
EventSel=C5H, UMask=FBH, Precise	Counts mispredicted near indirect CALL branch instructions retired, where the target address taken was not what the processor predicted.
BR_MISP_RETIRED.TAKEN_JCC	
EventSel=C5H, UMask=FEH, Precise	Counts mispredicted retired Jcc (Jump on Conditional Code/Jump if Condition is Met) branch instructions retired that were supposed to be taken but the processor predicted that it would not be taken.
ISSUE_SLOTS_NOT_CONSUMED.ANY	
EventSel=CAH, UMask=00H	Counts the number of issue slots per core cycle that were not consumed by the backend due to either a full resource in the backend (RESOURCE_FULL) or due to the processor recovering from some event (RECOVERY).

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
ISSUE_SLOTS_NOT_CONSUMED.RESOURCE_FULL	
EventSel=CAH, UMask=01H	Counts the number of issue slots per core cycle that were not consumed because of a full resource in the backend. Including but not limited to resources such as the Re-order Buffer (ROB), reservation stations (RS), load/store buffers, physical registers, or any other needed machine resource that is currently unavailable. Note that uops must be available for consumption in order for this event to fire. If a uop is not available (Instruction Queue is empty), this event will not count.
ISSUE_SLOTS_NOT_CONSUMED.RECOVERY	
EventSel=CAH, UMask=02H	Counts the number of issue slots per core cycle that were not consumed by the backend because allocation is stalled waiting for a mispredicted jump to retire or other branch-like conditions (e.g. the event is relevant during certain microcode flows). Counts all issue slots blocked while within this window including slots where uops were not available in the Instruction Queue.
HW_INTERRUPTS.RECEIVED	
EventSel=CBH, UMask=01H	Counts hardware interrupts received by the processor.
HW_INTERRUPTS.MASKED	
EventSel=CBH, UMask=02H	Counts the number of core cycles during which interrupts are masked (disabled). Increments by 1 each core cycle that EFLAGS.IF is 0, regardless of whether interrupts are pending or not.
HW_INTERRUPTS.PENDING_AND_MASKED	
EventSel=CBH, UMask=04H	Counts core cycles during which there are pending interrupts, but interrupts are masked (EFLAGS.IF = 0).
CYCLES_DIV_BUSY.ALL	
EventSel=CDH, UMask=00H	Counts core cycles if either divide unit is busy.
CYCLES_DIV_BUSY.IDIV	
EventSel=CDH, UMask=01H	Counts core cycles the integer divide unit is busy.
CYCLES_DIV_BUSY.FPDIV	
EventSel=CDH, UMask=02H	Counts core cycles the floating point divide unit is busy.
MEM_UOPS_RETIRED.DTLB_MISS_LOADS	
EventSel=D0H, UMask=11H, Precise	Counts load uops retired that caused a DTLB miss.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
MEM_UOPS_RETIRED.DTLB_MISS_STORES	
EventSel=D0H, UMask=12H, Precise	Counts store uops retired that caused a DTLB miss.
MEM_UOPS_RETIRED.DTLB_MISS	
EventSel=D0H, UMask=13H, Precise	Counts uops retired that had a DTLB miss on load, store or either. Note that when two distinct memory operations to the same page miss the DTLB, only one of them will be recorded as a DTLB miss.
MEM_UOPS_RETIRED.LOCK_LOADS	
EventSel=D0H, UMask=21H, Precise	Counts locked memory uops retired. This includes "regular" locks and bus locks. (To specifically count bus locks only, see the Offcore response event.) A locked access is one with a lock prefix, or an exchange to memory. See the SDM for a complete description of which memory load accesses are locks.
MEM_UOPS_RETIRED.SPLIT_LOADS	
EventSel=D0H, UMask=41H, Precise	Counts load uops retired where the data requested spans a 64 byte cache line boundary.
MEM_UOPS_RETIRED.SPLIT_STORES	
EventSel=D0H, UMask=42H, Precise	Counts store uops retired where the data requested spans a 64 byte cache line boundary.
MEM_UOPS_RETIRED.SPLIT	
EventSel=D0H, UMask=43H, Precise	Counts memory uops retired where the data requested spans a 64 byte cache line boundary.
MEM_UOPS_RETIRED.ALL_LOADS	
EventSel=D0H, UMask=81H, Precise	Counts the number of load uops retired.
MEM_UOPS_RETIRED.ALL_STORES	
EventSel=D0H, UMask=82H, Precise	Counts the number of store uops retired.
MEM_UOPS_RETIRED.ALL	
EventSel=D0H, UMask=83H, Precise	Counts the number of memory uops retired that is either a loads or a store or both.
MEM_LOAD_UOPS_RETIRED.L1_HIT	
EventSel=D1H, UMask=01H, Precise	Counts load uops retired that hit the L1 data cache.
MEM_LOAD_UOPS_RETIRED.L2_HIT	
EventSel=D1H, UMask=02H, Precise	Counts load uops retired that hit in the L2 cache.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
MEM_LOAD_UOPS_RETIRED.L1_MISS	
EventSel=D1H, UMask=08H, Precise	Counts load uops retired that miss the L1 data cache.
MEM_LOAD_UOPS_RETIRED.L2_MISS	
EventSel=D1H, UMask=10H, Precise	Counts load uops retired that miss in the L2 cache.
MEM_LOAD_UOPS_RETIRED.HITM	
EventSel=D1H, UMask=20H, Precise	Counts load uops retired where the cache line containing the data was in the modified state of another core or modules cache (HITM). More specifically, this means that when the load address was checked by other caching agents (typically another processor) in the system, one of those caching agents indicated that they had a dirty copy of the data. Loads that obtain a HITM response incur greater latency than most is typical for a load. In addition, since HITM indicates that some other processor had this data in its cache, it implies that the data was shared between processors, or potentially was a lock or semaphore value. This event is useful for locating sharing, false sharing, and contended locks.
MEM_LOAD_UOPS_RETIRED.WCB_HIT	
EventSel=D1H, UMask=40H, Precise	Counts memory load uops retired where the data is retrieved from the WCB (or fill buffer), indicating that the load found its data while that data was in the process of being brought into the L1 cache. Typically a load will receive this indication when some other load or prefetch missed the L1 cache and was in the process of retrieving the cache line containing the data, but that process had not yet finished (and written the data back to the cache). For example, consider load X and Y, both referencing the same cache line that is not in the L1 cache. If load X misses cache first, it obtains and WCB (or fill buffer) and begins the process of requesting the data. When load Y requests the data, it will either hit the WCB, or the L1 cache, depending on exactly what time the request to Y occurs.
MEM_LOAD_UOPS_RETIRED.DRAM_HIT	
EventSel=D1H, UMask=80H, Precise	Counts memory load uops retired where the data is retrieved from DRAM. Event is counted at retirement, so the speculative loads are ignored. A memory load can hit (or miss) the L1 cache, hit (or miss) the L2 cache, hit DRAM, hit in the WCB or receive a HITM response.

Table 15: Performance Events of the Processor Core Supported by Goldmont Microarchitecture

Event Name	
Configuration	Description
BACLEARS.ALL	
EventSel=E6H, UMask=01H	Counts the number of times a BACLEAR is signaled for any reason, including, but not limited to indirect branch/call, Jcc (Jump on Conditional Code/Jump if Condition is Met) branch, unconditional branch/call, and returns.
BACLEARS.RETURN	
EventSel=E6H, UMask=08H	Counts BACLEARS on return instructions.
BACLEARS.COND	
EventSel=E6H, UMask=10H	Counts BACLEARS on Jcc (Jump on Conditional Code/Jump if Condition is Met) branches.
MS_DECODED.MS_ENTRY	
EventSel=E7H, UMask=01H	Counts the number of times the Microcode Sequencer (MS) starts a flow of uops from the MSROM. It does not count every time a uop is read from the MSROM. The most common case that this counts is when a micro-coded instruction is encountered by the front end of the machine. Other cases include when an instruction encounters a fault, trap, or microcode assist of any sort that initiates a flow of uops. The event will count MS startups for uops that are speculative, and subsequently cleared by branch mispredict or a machine clear.
DECODE_RESTRICTION.PREDECODE_WRONG	
EventSel=E9H, UMask=01H	Counts the number of times the prediction (from the predecode cache) for instruction length is incorrect.

Performance Monitoring Events based on Airmont Microarchitecture

Next Generation Intel Atom processors based on the Airmont Microarchitecture support the performance-monitoring events listed in the table below.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed	This event counts the number of instructions that retire. For instructions that consist of multiple micro-ops, this event counts exactly once, as the last micro-op of the instruction retires. The event continues counting while instructions retire, including during interrupt service routines caused by hardware interrupts, faults or traps. Background: Modern microprocessors employ extensive pipelining and speculative techniques. Since sometimes an instruction is started but never completed, the notion of 'retirement' is introduced. A retired instruction is one that commits its states. Or stated differently, an instruction might be abandoned at some point. No instruction is truly finished until it retires. This counter measures the number of completed instructions. The fixed event is INST_RETIRED.ANY and the programmable event is INST_RETIRED.ANY_P.
CPU_CLK_UNHALTED.CORE	
Architectural, Fixed	Counts the number of core cycles while the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. This event is a component in many key event ratios. The core frequency may change from time to time. For this reason this event may have a changing ratio with regards to time. In systems with a constant core frequency, this event can give you a measurement of the elapsed time while the core was not in halt state by dividing the event count by the core frequency. This event is architecturally defined and is a designated fixed counter. CPU_CLK_UNHALTED.CORE and CPU_CLK_UNHALTED.CORE_P use the core frequency which may change from time to time. CPU_CLK_UNHALTE.REF_TSC and CPU_CLK_UNHALTED.REF are not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time. The fixed events are CPU_CLK_UNHALTED.CORE and CPU_CLK_UNHALTED.REF_TSC and the programmable events are CPU_CLK_UNHALTED.CORE_P and CPU_CLK_UNHALTED.REF.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
CPU_CLK_UNHALTED.REF_TSC	
Architectural, Fixed	Counts the number of reference cycles while the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. This event is a component in many key event ratios. The core frequency may change from time. This event is not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time. Divide this event count by core frequency to determine the elapsed time while the core was not in halt state. Divide this event count by core frequency to determine the elapsed time while the core was not in halt state. This event is architecturally defined and is a designated fixed counter. CPU_CLK_UNHALTED.CORE and CPU_CLK_UNHALTED.CORE_P use the core frequency which may change from time to time. CPU_CLK_UNHALTE.REF_TSC and CPU_CLK_UNHALTED.REF are not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time. The fixed events are CPU_CLK_UNHALTED.CORE and CPU_CLK_UNHALTED.REF_TSC and the programmable events are CPU_CLK_UNHALTED.CORE_P and CPU_CLK_UNHALTED.REF.
REHABQ.LD_BLOCK_ST_FORWARD	
EventSel=03H, UMask=01H, Precise	This event counts the number of retired loads that were prohibited from receiving forwarded data from the store because of address mismatch.
REHABQ.LD_BLOCK_STD_NOTREADY	
EventSel=03H, UMask=02H	This event counts the cases where a forward was technically possible, but did not occur because the store data was not available at the right time .
REHABQ.ST_SPLITS	
EventSel=03H, UMask=04H	This event counts the number of retire stores that experienced cache line boundary splits.
REHABQ.LD_SPLITS	
EventSel=03H, UMask=08H, Precise	This event counts the number of retire loads that experienced cache line boundary splits.
REHABQ.LOCK	
EventSel=03H, UMask=10H	This event counts the number of retired memory operations with lock semantics. These are either implicit locked instructions such as the XCHG instruction or instructions with an explicit LOCK prefix (0xF0).

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
REHABQ.STA_FULL	
EventSel=03H, UMask=20H	This event counts the number of retired stores that are delayed because there is not a store address buffer available.
REHABQ.ANY_LD	
EventSel=03H, UMask=40H	This event counts the number of load uops reissued from Rehabq.
REHABQ.ANY_ST	
EventSel=03H, UMask=80H	This event counts the number of store uops reissued from Rehabq.
MEM_UOPS_RETIRED.L1_MISS_LOADS	
EventSel=04H, UMask=01H	This event counts the number of load ops retired that miss in L1 Data cache. Note that prefetch misses will not be counted.
MEM_UOPS_RETIRED.L2_HIT_LOADS	
EventSel=04H, UMask=02H, Precise	This event counts the number of load ops retired that hit in the L2.
MEM_UOPS_RETIRED.L2_MISS_LOADS	
EventSel=04H, UMask=04H, Precise	This event counts the number of load ops retired that miss in the L2.
MEM_UOPS_RETIRED.DTLB_MISS_LOADS	
EventSel=04H, UMask=08H, Precise	This event counts the number of load ops retired that had DTLB miss.
MEM_UOPS_RETIRED.UTLB_MISS	
EventSel=04H, UMask=10H	This event counts the number of load ops retired that had UTLB miss.
MEM_UOPS_RETIRED.HITM	
EventSel=04H, UMask=20H, Precise	This event counts the number of load ops retired that got data from the other core or from the other module.
MEM_UOPS_RETIRED.ALL_LOADS	
EventSel=04H, UMask=40H	This event counts the number of load ops retired.
MEM_UOPS_RETIRED.ALL_STORES	
EventSel=04H, UMask=80H	This event counts the number of store ops retired.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
PAGE_WALKS.D_SIDE_WALKS	
EventSel=05H, UMask=01H, EdgeDetect=1	This event counts when a data (D) page walk is completed or started. Since a page walk implies a TLB miss, the number of TLB misses can be counted by counting the number of pagewalks.
PAGE_WALKS.D_SIDE_CYCLES	
EventSel=05H, UMask=01H	This event counts every cycle when a D-side (walks due to a load) page walk is in progress. Page walk duration divided by number of page walks is the average duration of page-walks.
PAGE_WALKS.I_SIDE_WALKS	
EventSel=05H, UMask=02H, EdgeDetect=1	This event counts when an instruction (I) page walk is completed or started. Since a page walk implies a TLB miss, the number of TLB misses can be counted by counting the number of pagewalks.
PAGE_WALKS.I_SIDE_CYCLES	
EventSel=05H, UMask=02H	This event counts every cycle when a I-side (walks due to an instruction fetch) page walk is in progress. Page walk duration divided by number of page walks is the average duration of page-walks.
PAGE_WALKS.WALKS	
EventSel=05H, UMask=03H, EdgeDetect=1	This event counts when a data (D) page walk or an instruction (I) page walk is completed or started. Since a page walk implies a TLB miss, the number of TLB misses can be counted by counting the number of pagewalks.
PAGE_WALKS.CYCLES	
EventSel=05H, UMask=03H	This event counts every cycle when a data (D) page walk or instruction (I) page walk is in progress. Since a pagewalk implies a TLB miss, the approximate cost of a TLB miss can be determined from this event.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	This event counts the total number of L2 cache references and the number of L2 cache misses respectively.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	This event counts requests originating from the core that references a cache line in the L2 cache.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
L2_REJECT_XQ.ALL	
EventSel=30H, UMask=00H	This event counts the number of demand and prefetch transactions that the L2 XQ rejects due to a full or near full condition which likely indicates back pressure from the IDI link. The XQ may reject transactions from the L2Q (non-cacheable requests), BBS (L2 misses) and WOB (L2 write-back victims) .
CORE_REJECT_L2Q.ALL	
EventSel=31H, UMask=00H	Counts the number of (demand and L1 prefetchers) core requests rejected by the L2Q due to a full or nearly full w condition which likely indicates back pressure from L2Q. It also counts requests that would have gone directly to the XQ, but are rejected due to a full or nearly full condition, indicating back pressure from the IDI link. The L2Q may also reject transactions from a core to insure fairness between cores, or to delay a core's dirty eviction when the address conflicts incoming external snoops. (Note that L2 prefetcher requests that are dropped are not counted by this event.).
CPU_CLK_UNHALTED.CORE_P	
EventSel=3CH, UMask=00H, Architectural	This event counts the number of core cycles while the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. In mobile systems the core frequency may change from time to time. For this reason this event may have a changing ratio with regards to time.
CPU_CLK_UNHALTED.REF	
EventSel=3CH, UMask=01H, Architectural	This event counts the number of bus cycles that the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. In mobile systems the core frequency may change from time. This event is not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time.
ICACHE.HIT	
EventSel=80H, UMask=01H	This event counts all instruction fetches from the instruction cache.
ICACHE.MISSES	
EventSel=80H, UMask=02H	This event counts all instruction fetches that miss the Instruction cache or produce memory requests. This includes uncacheable fetches. An instruction fetch miss is counted only once and not once for every cycle it is outstanding.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
ICACHE.ACCESSES	
EventSel=80H, UMask=03H	This event counts all instruction fetches, not including most uncacheable fetches.
FETCH_STALL.ITLB_FILL_PENDING_CYCLES	
EventSel=86H, UMask=02H	Counts cycles that fetch is stalled due to an outstanding ITLB miss. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes due to an ITLB miss. Note: this event is not the same as page walk cycles to retrieve an instruction translation.
FETCH_STALL.ICACHE_FILL_PENDING_CYCLES	
EventSel=86H, UMask=04H	Counts cycles that fetch is stalled due to an outstanding ICache miss. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes due to an ICache miss. Note: this event is not the same as the total number of cycles spent retrieving instruction cache lines from the memory hierarchy.
FETCH_STALL.ALL	
EventSel=86H, UMask=3FH	Counts cycles that fetch is stalled due to any reason. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes. This will include cycles due to an ITLB miss, ICache miss and other events. .
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural	This event counts the number of instructions that retire execution. For instructions that consist of multiple micro-ops, this event counts the retirement of the last micro-op of the instruction. The counter continues counting during hardware interrupts, traps, and inside interrupt handlers. .
UOPS_RETIRED.MS	
EventSel=C2H, UMask=01H	This event counts the number of micro-ops retired that were supplied from MSROM.
UOPS_RETIRED.ALL	
EventSel=C2H, UMask=10H	This event counts the number of micro-ops retired. The processor decodes complex macro instructions into a sequence of simpler micro-ops. Most instructions are composed of one or two micro-ops. Some instructions are decoded into longer sequences such as repeat instructions, floating point transcendental instructions, and assists. In some cases micro-op sequences are fused or whole instructions are fused into one micro-op. See other UOPS_RETIRED events for differentiating retired fused and non-fused micro-ops. .

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=01H	This event counts the number of times that a program writes to a code section. Self-modifying code causes a severe penalty in all Intel® architecture processors.
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	This event counts the number of times that pipeline was cleared due to memory ordering issues.
MACHINE_CLEARS.FP_ASSIST	
EventSel=C3H, UMask=04H	This event counts the number of times that pipeline stalled due to FP operations needing assists.
MACHINE_CLEARS.ALL	
EventSel=C3H, UMask=08H	Machine clears happen when something happens in the machine that causes the hardware to need to take special care to get the right answer. When such a condition is signaled on an instruction, the front end of the machine is notified that it must restart, so no more instructions will be decoded from the current path. All instructions 'older' than this one will be allowed to finish. This instruction and all 'younger' instructions must be cleared, since they must not be allowed to complete. Essentially, the hardware waits until the problematic instruction is the oldest instruction in the machine. This means all older instructions are retired, and all pending stores (from older instructions) are completed. Then the new path of instructions from the front end are allowed to start into the machine. There are many conditions that might cause a machine clear (including the receipt of an interrupt, or a trap or a fault). All those conditions (including but not limited to MACHINE_CLEARS.MEMORY_ORDERING, MACHINE_CLEARS.SMC, and MACHINE_CLEARS.FP_ASSIST) are captured in the ANY event. In addition, some conditions can be specifically counted (i.e. SMC, MEMORY_ORDERING, FP_ASSIST). However, the sum of SMC, MEMORY_ORDERING, and FP_ASSIST machine clears will not necessarily equal the number of ANY.
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	ALL_BRANCHES counts the number of any branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
BR_INST_RETIRED.JCC	
EventSel=C4H, UMask=7EH, Precise	JCC counts the number of conditional branch (JCC) instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.ALL_TAKEN_BRANCHES	
EventSel=C4H, UMask=80H, Precise	ALL_TAKEN_BRANCHES counts the number of all taken branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=BFH, Precise	FAR counts the number of far branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
BR_INST_RETIRED.NON_RETURN_IND	
EventSel=C4H, UMask=EBH, Precise	NON_RETURN_IND counts the number of near indirect JMP and near indirect CALL branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.RETURN	
EventSel=C4H, UMask=F7H, Precise	RETURN counts the number of near RET branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.CALL	
EventSel=C4H, UMask=F9H, Precise	CALL counts the number of near CALL branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
BR_INST_RETIRED.IND_CALL	
EventSel=C4H, UMask=FBH, Precise	IND_CALL counts the number of near indirect CALL branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.REL_CALL	
EventSel=C4H, UMask=FDH, Precise	REL_CALL counts the number of near relative CALL branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.TAKEN_JCC	
EventSel=C4H, UMask=FEH, Precise	TAKEN_JCC counts the number of taken conditional branch (JCC) instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	ALL_BRANCHES counts the number of any mispredicted branch instructions retired. This umask is an architecturally defined event. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path. .

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
BR_MISP_RETIRED.JCC	
EventSel=C5H, UMask=7EH, Precise	JCC counts the number of mispredicted conditional branches (JCC) instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path. .
BR_MISP_RETIRED.NON_RETURN_IND	
EventSel=C5H, UMask=EBH, Precise	NON_RETURN_IND counts the number of mispredicted near indirect JMP and near indirect CALL branch instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path. .
BR_MISP_RETIRED.RETURN	
EventSel=C5H, UMask=F7H, Precise	RETURN counts the number of mispredicted near RET branch instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path. .
BR_MISP_RETIRED.IND_CALL	
EventSel=C5H, UMask=FBH, Precise	IND_CALL counts the number of mispredicted near indirect CALL branch instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path. .

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
BR_MISP_RETIRED.TAKEN_JCC	
EventSel=C5H, UMask=FEH, Precise	TAKEN_JCC counts the number of mispredicted taken conditional branch (JCC) instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path. .
NO_ALLOC_CYCLES.ROB_FULL	
EventSel=CAH, UMask=01H	Counts the number of cycles when no uops are allocated and the ROB is full (less than 2 entries available).
NO_ALLOC_CYCLES.MISPREDICTS	
EventSel=CAH, UMask=04H	Counts the number of cycles when no uops are allocated and the alloc pipe is stalled waiting for a mispredicted jump to retire. After the misprediction is detected, the front end will start immediately but the allocate pipe stalls until the mispredicted .
NO_ALLOC_CYCLES.RAT_STALL	
EventSel=CAH, UMask=20H	Counts the number of cycles when no uops are allocated and a RATstall is asserted.
NO_ALLOC_CYCLES.ALL	
EventSel=CAH, UMask=3FH	The NO_ALLOC_CYCLES.ALL event counts the number of cycles when the front-end does not provide any instructions to be allocated for any reason. This event indicates the cycles where an allocation stalls occurs, and no UOPS are allocated in that cycle.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
NO_ALLOC_CYCLES.NOT_DELIVERED	
EventSel=CAH, UMask=50H	The NO_ALLOC_CYCLES.NOT_DELIVERED event is used to measure front-end inefficiencies, i.e. when front-end of the machine is not delivering micro-ops to the back-end and the back-end is not stalled. This event can be used to identify if the machine is truly front-end bound. When this event occurs, it is an indication that the front-end of the machine is operating at less than its theoretical peak performance. Background: We can think of the processor pipeline as being divided into 2 broader parts: Front-end and Back-end. Front-end is responsible for fetching the instruction, decoding into micro-ops (uops) in machine understandable format and putting them into a micro-op queue to be consumed by back end. The back-end then takes these micro-ops, allocates the required resources. When all resources are ready, micro-ops are executed. If the back-end is not ready to accept micro-ops from the front-end, then we do not want to count these as front-end bottlenecks. However, whenever we have bottlenecks in the back-end, we will have allocation unit stalls and eventually forcing the front-end to wait until the back-end is ready to receive more UOPS. This event counts the cycles only when back-end is requesting more uops and front-end is not able to provide them. Some examples of conditions that cause front-end efficiencies are: Icache misses, ITLB misses, and decoder restrictions that limit the the front-end bandwidth.
RS_FULL_STALL.MEC	
EventSel=CBH, UMask=01H	Counts the number of cycles and allocation pipeline is stalled and is waiting for a free MEC reservation station entry. The cycles should be appropriately counted in case of the cracked ops e.g. In case of a cracked load-op, the load portion is sent to M.
RS_FULL_STALL.ALL	
EventSel=CBH, UMask=1FH	Counts the number of cycles the Alloc pipeline is stalled when any one of the RSs (IEC, FPC and MEC) is full. This event is a superset of all the individual RS stall event counts.
CYCLES_DIV_BUSY.ALL	
EventSel=CDH, UMask=01H	Cycles the divider is busy.This event counts the cycles when the divide unit is unable to accept a new divide UOP because it is busy processing a previously dispatched UOP. The cycles will be counted irrespective of whether or not another divide UOP is waiting to enter the divide unit (from the RS). This event might count cycles while a divide is in progress even if the RS is empty. The divide instruction is one of the longest latency instructions in the machine. Hence, it has a special event associated with it to help determine if divides are delaying the retirement of instructions.

Table 16: Performance Events of the Processor Core Supported by Airmont Microarchitecture

Event Name	
Configuration	Description
BACLEARS.ALL	
EventSel=E6H, UMask=01H	The BACLEARS event counts the number of times the front end is resteered, mainly when the Branch Prediction Unit cannot provide a correct prediction and this is corrected by the Branch Address Calculator at the front end. The BACLEARS.ANY event counts the number of baclears for any type of branch.
BACLEARS.RETURN	
EventSel=E6H, UMask=08H	The BACLEARS event counts the number of times the front end is resteered, mainly when the Branch Prediction Unit cannot provide a correct prediction and this is corrected by the Branch Address Calculator at the front end. The BACLEARS.RETURN event counts the number of RETURN baclears.
BACLEARS.COND	
EventSel=E6H, UMask=10H	The BACLEARS event counts the number of times the front end is resteered, mainly when the Branch Prediction Unit cannot provide a correct prediction and this is corrected by the Branch Address Calculator at the front end. The BACLEARS.COND event counts the number of JCC (Jump on Condtional Code) baclears.
MS_DECODED.MS_ENTRY	
EventSel=E7H, UMask=01H	Counts the number of times the MSROM starts a flow of UOPS. It does not count every time a UOP is read from the microcode ROM. The most common case that this counts is when a micro-coded instruction is encountered by the front end of the machine. Other cases include when an instruction encounters a fault, trap, or microcode assist of any sort. The event will count MSROM startups for UOPS that are speculative, and subsequently cleared by branch mispredict or machine clear. Background: UOPS are produced by two mechanisms. Either they are generated by hardware that decodes instructions into UOPS, or they are delivered by a ROM (called the MSROM) that holds UOPS associated with a specific instruction. MSROM UOPS might also be delivered in response to some condition such as a fault or other exceptional condition. This event is an excellent mechanism for detecting instructions that require the use of MSROM instructions.
DECODE_RESTRICTION.PREDECODE_WRONG	
EventSel=E9H, UMask=01H	Counts the number of times a decode restriction reduced the decode throughput due to wrong instruction length prediction.

Performance Monitoring Events based on Silvermont Microarchitecture

Next Generation Intel Atom processors based on the Silvermont Microarchitecture support the performance-monitoring events listed in the table below.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
INST_RETIRED.ANY	
Architectural, Fixed	This event counts the number of instructions that retire. For instructions that consist of multiple micro-ops, this event counts exactly once, as the last micro-op of the instruction retires. The event continues counting while instructions retire, including during interrupt service routines caused by hardware interrupts, faults or traps. Background: Modern microprocessors employ extensive pipelining and speculative techniques. Since sometimes an instruction is started but never completed, the notion of "retirement" is introduced. A retired instruction is one that commits its states. Or stated differently, an instruction might be abandoned at some point. No instruction is truly finished until it retires. This counter measures the number of completed instructions. The fixed event is INST_RETIRED.ANY and the programmable event is INST_RETIRED.ANY_P.
CPU_CLK_UNHALTED.CORE	
Architectural, Fixed	Counts the number of core cycles while the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. This event is a component in many key event ratios. The core frequency may change from time to time. For this reason this event may have a changing ratio with regards to time. In systems with a constant core frequency, this event can give you a measurement of the elapsed time while the core was not in halt state by dividing the event count by the core frequency. This event is architecturally defined and is a designated fixed counter. CPU_CLK_UNHALTED.CORE and CPU_CLK_UNHALTED.CORE_P use the core frequency which may change from time to time. CPU_CLK_UNHALTE.REF_TSC and CPU_CLK_UNHALTED.REF are not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time. The fixed events are CPU_CLK_UNHALTED.CORE and CPU_CLK_UNHALTED.REF_TSC and the programmable events are CPU_CLK_UNHALTED.CORE_P and CPU_CLK_UNHALTED.REF.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
CPU_CLK_UNHALTED.REF_TSC	
Architectural, Fixed	Counts the number of reference cycles while the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. This event is a component in many key event ratios. The core frequency may change from time. This event is not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time. Divide this event count by core frequency to determine the elapsed time while the core was not in halt state. Divide this event count by core frequency to determine the elapsed time while the core was not in halt state. This event is architecturally defined and is a designated fixed counter. CPU_CLK_UNHALTED.CORE and CPU_CLK_UNHALTED.CORE_P use the core frequency which may change from time to time. CPU_CLK_UNHALTE.REF_TSC and CPU_CLK_UNHALTED.REF are not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time. The fixed events are CPU_CLK_UNHALTED.CORE and CPU_CLK_UNHALTED.REF_TSC and the programmable events are CPU_CLK_UNHALTED.CORE_P and CPU_CLK_UNHALTED.REF.
REHABQ.LD_BLOCK_ST_FORWARD	
EventSel=03H, UMask=01H, Precise	This event counts the number of retired loads that were prohibited from receiving forwarded data from the store because of address mismatch.
REHABQ.LD_BLOCK_STD_NOTREADY	
EventSel=03H, UMask=02H	This event counts the cases where a forward was technically possible, but did not occur because the store data was not available at the right time.
REHABQ.ST_SPLITS	
EventSel=03H, UMask=04H	This event counts the number of retire stores that experienced cache line boundary splits.
REHABQ.LD_SPLITS	
EventSel=03H, UMask=08H, Precise	This event counts the number of retire loads that experienced cache line boundary splits.
REHABQ.LOCK	
EventSel=03H, UMask=10H	This event counts the number of retired memory operations with lock semantics. These are either implicit locked instructions such as the XCHG instruction or instructions with an explicit LOCK prefix (0xF0).

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
REHABQ.STA_FULL	
EventSel=03H, UMask=20H	This event counts the number of retired stores that are delayed because there is not a store address buffer available.
REHABQ.ANY_LD	
EventSel=03H, UMask=40H	This event counts the number of load uops reissued from Rehabq.
REHABQ.ANY_ST	
EventSel=03H, UMask=80H	This event counts the number of store uops reissued from Rehabq.
MEM_UOPS_RETIRED.L1_MISS_LOADS	
EventSel=04H, UMask=01H	This event counts the number of load ops retired that miss in L1 Data cache. Note that prefetch misses will not be counted.
MEM_UOPS_RETIRED.L2_HIT_LOADS	
EventSel=04H, UMask=02H, Precise	This event counts the number of load ops retired that hit in the L2.
MEM_UOPS_RETIRED.L2_MISS_LOADS	
EventSel=04H, UMask=04H, Precise	This event counts the number of load ops retired that miss in the L2.
MEM_UOPS_RETIRED.DTLB_MISS_LOADS	
EventSel=04H, UMask=08H, Precise	This event counts the number of load ops retired that had DTLB miss.
MEM_UOPS_RETIRED.UTLB_MISS	
EventSel=04H, UMask=10H	This event counts the number of load ops retired that had UTLB miss.
MEM_UOPS_RETIRED.HITM	
EventSel=04H, UMask=20H, Precise	This event counts the number of load ops retired that got data from the other core or from the other module.
MEM_UOPS_RETIRED.ALL_LOADS	
EventSel=04H, UMask=40H	This event counts the number of load ops retired.
MEM_UOPS_RETIRED.ALL_STORES	
EventSel=04H, UMask=80H	This event counts the number of store ops retired.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
PAGE_WALKS.D_SIDE_WALKS	
EventSel=05H, UMask=01H, EdgeDetect=1	This event counts when a data (D) page walk is completed or started. Since a page walk implies a TLB miss, the number of TLB misses can be counted by counting the number of pagewalks.
PAGE_WALKS.D_SIDE_CYCLES	
EventSel=05H, UMask=01H	This event counts every cycle when a D-side (walks due to a load) page walk is in progress. Page walk duration divided by number of page walks is the average duration of page-walks.
PAGE_WALKS.I_SIDE_WALKS	
EventSel=05H, UMask=02H, EdgeDetect=1	This event counts when an instruction (I) page walk is completed or started. Since a page walk implies a TLB miss, the number of TLB misses can be counted by counting the number of pagewalks.
PAGE_WALKS.I_SIDE_CYCLES	
EventSel=05H, UMask=02H	This event counts every cycle when a I-side (walks due to an instruction fetch) page walk is in progress. Page walk duration divided by number of page walks is the average duration of page-walks.
PAGE_WALKS.WALKS	
EventSel=05H, UMask=03H, EdgeDetect=1	This event counts when a data (D) page walk or an instruction (I) page walk is completed or started. Since a page walk implies a TLB miss, the number of TLB misses can be counted by counting the number of pagewalks.
PAGE_WALKS.CYCLES	
EventSel=05H, UMask=03H	This event counts every cycle when a data (D) page walk or instruction (I) page walk is in progress. Since a pagewalk implies a TLB miss, the approximate cost of a TLB miss can be determined from this event.
LONGEST_LAT_CACHE.MISS	
EventSel=2EH, UMask=41H, Architectural	This event counts the total number of L2 cache references and the number of L2 cache misses respectively.
LONGEST_LAT_CACHE.REFERENCE	
EventSel=2EH, UMask=4FH, Architectural	This event counts requests originating from the core that references a cache line in the L2 cache.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
L2_REJECT_XQ.ALL	
EventSel=30H, UMask=00H	This event counts the number of demand and prefetch transactions that the L2 XQ rejects due to a full or near full condition which likely indicates back pressure from the IDI link. The XQ may reject transactions from the L2Q (non-cacheable requests), BBS (L2 misses) and WOB (L2 write-back victims).
CORE_REJECT_L2Q.ALL	
EventSel=31H, UMask=00H	Counts the number of (demand and L1 prefetchers) core requests rejected by the L2Q due to a full or nearly full w condition which likely indicates back pressure from L2Q. It also counts requests that would have gone directly to the XQ, but are rejected due to a full or nearly full condition, indicating back pressure from the IDI link. The L2Q may also reject transactions from a core to insure fairness between cores, or to delay a core's dirty eviction when the address conflicts incoming external snoops. (Note that L2 prefetcher requests that are dropped are not counted by this event.).
CPU_CLK_UNHALTED.CORE_P	
EventSel=3CH, UMask=00H, Architectural	This event counts the number of core cycles while the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. In mobile systems the core frequency may change from time to time. For this reason this event may have a changing ratio with regards to time.
CPU_CLK_UNHALTED.REF	
EventSel=3CH, UMask=01H, Architectural	This event counts the number of bus cycles that the core is not in a halt state. The core enters the halt state when it is running the HLT instruction. In mobile systems the core frequency may change from time. This event is not affected by core frequency changes but counts as if the core is running at the maximum frequency all the time.
ICACHE.HIT	
EventSel=80H, UMask=01H	This event counts all instruction fetches from the instruction cache.
ICACHE.MISSES	
EventSel=80H, UMask=02H	This event counts all instruction fetches that miss the Instruction cache or produce memory requests. This includes uncacheable fetches. An instruction fetch miss is counted only once and not once for every cycle it is outstanding.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
ICACHE.ACCESSES	
EventSel=80H, UMask=03H	This event counts all instruction fetches, not including most uncacheable fetches.
FETCH_STALL.ITLB_FILL_PENDING_CYCLES	
EventSel=86H, UMask=02H	Counts cycles that fetch is stalled due to an outstanding ITLB miss. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes due to an ITLB miss. Note: this event is not the same as page walk cycles to retrieve an instruction translation.
FETCH_STALL.ICACHE_FILL_PENDING_CYCLES	
EventSel=86H, UMask=04H	Counts cycles that fetch is stalled due to an outstanding ICache miss. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes due to an ICache miss. Note: this event is not the same as the total number of cycles spent retrieving instruction cache lines from the memory hierarchy. Counts cycles that fetch is stalled due to any reason. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes. This will include cycles due to an ITLB miss, ICache miss and other events. .
FETCH_STALL.ALL	
EventSel=86H, UMask=3FH	Counts cycles that fetch is stalled due to any reason. That is, the decoder queue is able to accept bytes, but the fetch unit is unable to provide bytes. This will include cycles due to an ITLB miss, ICache miss and other events. .
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Architectural	This event counts the number of instructions that retire execution. For instructions that consist of multiple micro-ops, this event counts the retirement of the last micro-op of the instruction. The counter continues counting during hardware interrupts, traps, and inside interrupt handlers.
UOPS_RETIRED.MS	
EventSel=C2H, UMask=01H	This event counts the number of micro-ops retired that were supplied from MSROM.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
UOPS_RETIRED.ALL	
EventSel=C2H, UMask=10H	This event counts the number of micro-ops retired. The processor decodes complex macro instructions into a sequence of simpler micro-ops. Most instructions are composed of one or two micro-ops. Some instructions are decoded into longer sequences such as repeat instructions, floating point transcendental instructions, and assists. In some cases micro-op sequences are fused or whole instructions are fused into one micro-op. See other UOPS_RETIRED events for differentiating retired fused and non-fused micro-ops.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=01H	This event counts the number of times that a program writes to a code section. Self-modifying code causes a severe penalty in all Intel® architecture processors.
MACHINE_CLEARS.MEMORY_ORDERING	
EventSel=C3H, UMask=02H	This event counts the number of times that pipeline was cleared due to memory ordering issues.
MACHINE_CLEARS.FP_ASSIST	
EventSel=C3H, UMask=04H	This event counts the number of times that pipeline stalled due to FP operations needing assists.
MACHINE_CLEARS.ALL	
EventSel=C3H, UMask=08H	Machine clears happen when something happens in the machine that causes the hardware to need to take special care to get the right answer. When such a condition is signaled on an instruction, the front end of the machine is notified that it must restart, so no more instructions will be decoded from the current path. All instructions "older" than this one will be allowed to finish. This instruction and all "younger" instructions must be cleared, since they must not be allowed to complete. Essentially, the hardware waits until the problematic instruction is the oldest instruction in the machine. This means all older instructions are retired, and all pending stores (from older instructions) are completed. Then the new path of instructions from the front end are allowed to start into the machine. There are many conditions that might cause a machine clear (including the receipt of an interrupt, or a trap or a fault). All those conditions (including but not limited to MACHINE_CLEARS.MEMORY_ORDERING, MACHINE_CLEARS.SMC, and MACHINE_CLEARS.FP_ASSIST) are captured in the ANY event. In addition, some conditions can be specifically counted (i.e. SMC, MEMORY_ORDERING, FP_ASSIST). However, the sum of SMC, MEMORY_ORDERING, and FP_ASSIST machine clears will not necessarily equal the number of ANY.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
BR_INST_RETIRED.ALL_BRANCHES	
EventSel=C4H, UMask=00H, Architectural, Precise	ALL_BRANCHES counts the number of any branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.JCC	
EventSel=C4H, UMask=7EH, Precise	JCC counts the number of conditional branch (JCC) instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.ALL_TAKEN_BRANCHES	
EventSel=C4H, UMask=80H, Precise	ALL_TAKEN_BRANCHES counts the number of all taken branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.FAR_BRANCH	
EventSel=C4H, UMask=BFH, Precise	FAR counts the number of far branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
BR_INST_RETIRED.NON_RETURN_IND	
EventSel=C4H, UMask=EBH, Precise	NON_RETURN_IND counts the number of near indirect JMP and near indirect CALL branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.RETURN	
EventSel=C4H, UMask=F7H, Precise	RETURN counts the number of near RET branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.CALL	
EventSel=C4H, UMask=F9H, Precise	CALL counts the number of near CALL branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
BR_INST_RETIRED.IND_CALL	
EventSel=C4H, UMask=FBH, Precise	IND_CALL counts the number of near indirect CALL branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.REL_CALL	
EventSel=C4H, UMask=FDH, Precise	REL_CALL counts the number of near relative CALL branch instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_INST_RETIRED.TAKEN_JCC	
EventSel=C4H, UMask=FEH, Precise	TAKEN_JCC counts the number of taken conditional branch (JCC) instructions retired. Branch prediction predicts the branch target and enables the processor to begin executing instructions long before the branch true execution path is known. All branches utilize the branch prediction unit (BPU) for prediction. This unit predicts the target address not only based on the EIP of the branch but also based on the execution path through which execution reached this EIP. The BPU can efficiently predict the following branch types: conditional branches, direct calls and jumps, indirect calls and jumps, returns.
BR_MISP_RETIRED.ALL_BRANCHES	
EventSel=C5H, UMask=00H, Architectural, Precise	ALL_BRANCHES counts the number of any mispredicted branch instructions retired. This umask is an architecturally defined event. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
BR_MISP_RETIRED.JCC	
EventSel=C5H, UMask=7EH, Precise	JCC counts the number of mispredicted conditional branches (JCC) instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path.
BR_MISP_RETIRED.NON_RETURN_IND	
EventSel=C5H, UMask=EBH, Precise	NON_RETURN_IND counts the number of mispredicted near indirect JMP and near indirect CALL branch instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path.
BR_MISP_RETIRED.RETURN	
EventSel=C5H, UMask=F7H, Precise	RETURN counts the number of mispredicted near RET branch instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path.
BR_MISP_RETIRED.IND_CALL	
EventSel=C5H, UMask=FBH, Precise	IND_CALL counts the number of mispredicted near indirect CALL branch instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
BR_MISP_RETIRED.TAKEN_JCC	
EventSel=C5H, UMask=FEH, Precise	TAKEN_JCC counts the number of mispredicted taken conditional branch (JCC) instructions retired. This event counts the number of retired branch instructions that were mispredicted by the processor, categorized by type. A branch misprediction occurs when the processor predicts that the branch would be taken, but it is not, or vice-versa. When the misprediction is discovered, all the instructions executed in the wrong (speculative) path must be discarded, and the processor must start fetching from the correct path.
NO_ALLOC_CYCLES.ROB_FULL	
EventSel=CAH, UMask=01H	Counts the number of cycles when no uops are allocated and the ROB is full (less than 2 entries available).
NO_ALLOC_CYCLES.MISPREDICTS	
EventSel=CAH, UMask=04H	Counts the number of cycles when no uops are allocated and the alloc pipe is stalled waiting for a mispredicted jump to retire. After the misprediction is detected, the front end will start immediately but the allocate pipe stalls until the mispredicted.
NO_ALLOC_CYCLES.RAT_STALL	
EventSel=CAH, UMask=20H	Counts the number of cycles when no uops are allocated and a RATstall is asserted.
NO_ALLOC_CYCLES.ALL	
EventSel=CAH, UMask=3FH	The NO_ALLOC_CYCLES.ALL event counts the number of cycles when the front-end does not provide any instructions to be allocated for any reason. This event indicates the cycles where an allocation stalls occurs, and no UOPS are allocated in that cycle.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
NO_ALLOC_CYCLES.NOT_DELIVERED	
EventSel=CAH, UMask=50H	The NO_ALLOC_CYCLES.NOT_DELIVERED event is used to measure front-end inefficiencies, i.e. when front-end of the machine is not delivering micro-ops to the back-end and the back-end is not stalled. This event can be used to identify if the machine is truly front-end bound. When this event occurs, it is an indication that the front-end of the machine is operating at less than its theoretical peak performance. Background: We can think of the processor pipeline as being divided into 2 broader parts: Front-end and Back-end. Front-end is responsible for fetching the instruction, decoding into micro-ops (uops) in machine understandable format and putting them into a micro-op queue to be consumed by back end. The back-end then takes these micro-ops, allocates the required resources. When all resources are ready, micro-ops are executed. If the back-end is not ready to accept micro-ops from the front-end, then we do not want to count these as front-end bottlenecks. However, whenever we have bottlenecks in the back-end, we will have allocation unit stalls and eventually forcing the front-end to wait until the back-end is ready to receive more UOPS. This event counts the cycles only when back-end is requesting more uops and front-end is not able to provide them. Some examples of conditions that cause front-end efficiencies are: Icache misses, ITLB misses, and decoder restrictions that limit the the front-end bandwidth.
RS_FULL_STALL.MEC	
EventSel=CBH, UMask=01H	Counts the number of cycles and allocation pipeline is stalled and is waiting for a free MEC reservation station entry. The cycles should be appropriately counted in case of the cracked ops e.g. In case of a cracked load-op, the load portion is sent to M.
RS_FULL_STALL.ALL	
EventSel=CBH, UMask=1FH	Counts the number of cycles the Alloc pipeline is stalled when any one of the RSs (IEC, FPC and MEC) is full. This event is a superset of all the individual RS stall event counts.
CYCLES_DIV_BUSY.ALL	
EventSel=CDH, UMask=01H	Cycles the divider is busy.This event counts the cycles when the divide unit is unable to accept a new divide UOP because it is busy processing a previously dispatched UOP. The cycles will be counted irrespective of whether or not another divide UOP is waiting to enter the divide unit (from the RS). This event might count cycles while a divide is in progress even if the RS is empty. The divide instruction is one of the longest latency instructions in the machine. Hence, it has a special event associated with it to help determine if divides are delaying the retirement of instructions.

Table 17: Performance Events of the Processor Core Supported by Silvermont Microarchitecture

Event Name	
Configuration	Description
BACLEARS.ALL	
EventSel=E6H, UMask=01H	The BACLEARS event counts the number of times the front end is resteered, mainly when the Branch Prediction Unit cannot provide a correct prediction and this is corrected by the Branch Address Calculator at the front end. The BACLEARS.ANY event counts the number of baclears for any type of branch.
BACLEARS.RETURN	
EventSel=E6H, UMask=08H	The BACLEARS event counts the number of times the front end is resteered, mainly when the Branch Prediction Unit cannot provide a correct prediction and this is corrected by the Branch Address Calculator at the front end. The BACLEARS.RETURN event counts the number of RETURN baclears.
BACLEARS.COND	
EventSel=E6H, UMask=10H	The BACLEARS event counts the number of times the front end is resteered, mainly when the Branch Prediction Unit cannot provide a correct prediction and this is corrected by the Branch Address Calculator at the front end. The BACLEARS.COND event counts the number of JCC (Jump on Condtional Code) baclears.
MS_DECODED.MS_ENTRY	
EventSel=E7H, UMask=01H	Counts the number of times the MSROM starts a flow of UOPS. It does not count every time a UOP is read from the microcode ROM. The most common case that this counts is when a micro-coded instruction is encountered by the front end of the machine. Other cases include when an instruction encounters a fault, trap, or microcode assist of any sort. The event will count MSROM startups for UOPS that are speculative, and subsequently cleared by branch mispredict or machine clear. Background: UOPS are produced by two mechanisms. Either they are generated by hardware that decodes instructions into UOPS, or they are delivered by a ROM (called the MSROM) that holds UOPS associated with a specific instruction. MSROM UOPS might also be delivered in response to some condition such as a fault or other exceptional condition. This event is an excellent mechanism for detecting instructions that require the use of MSROM instructions.
DECODE_RESTRICTION.PREDECODE_WRONG	
EventSel=E9H, UMask=01H	Counts the number of times a decode restriction reduced the decode throughput due to wrong instruction length prediction.

Performance Monitoring Events based on Bonnell Microarchitecture

Next Generation Intel Atom processors based on the Bonnell Microarchitecture support the performance-monitoring events listed in the table below.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
STORE_FORWARDS.GOOD	
EventSel=02H, UMask=81H	Good store forwards.
REISSUE.OVERLAP_STORE	
EventSel=03H, UMask=01H	Micro-op reissues on a store-load collision.
REISSUE.ANY	
EventSel=03H, UMask=7FH	Micro-op reissues for any cause.
REISSUE.OVERLAP_STORE.AR	
EventSel=03H, UMask=81H	Micro-op reissues on a store-load collision (At Retirement).
REISSUE.ANY.AR	
EventSel=03H, UMask=FFH	Micro-op reissues for any cause (At Retirement).
MISALIGN_MEM_REF.LD_SPLIT	
EventSel=05H, UMask=09H	Load splits.
MISALIGN_MEM_REF.ST_SPLIT	
EventSel=05H, UMask=0AH	Store splits.
MISALIGN_MEM_REF.SPLIT	
EventSel=05H, UMask=0FH	Memory references that cross an 8-byte boundary.
MISALIGN_MEM_REF.LD_SPLIT.AR	
EventSel=05H, UMask=89H	Load splits (At Retirement).
MISALIGN_MEM_REF.ST_SPLIT.AR	
EventSel=05H, UMask=8AH	Store splits (Ar Retirement).
MISALIGN_MEM_REF.RMW_SPLIT	
EventSel=05H, UMask=8CH	ld-op-st splits.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
MISALIGN_MEM_REF.SPLIT.AR	
EventSel=05H, UMask=8FH	Memory references that cross an 8-byte boundary (At Retirement).
MISALIGN_MEM_REF.LD_BUBBLE	
EventSel=05H, UMask=91H	Nonzero segbase load 1 bubble.
MISALIGN_MEM_REF.ST_BUBBLE	
EventSel=05H, UMask=92H	Nonzero segbase store 1 bubble.
MISALIGN_MEM_REF.RMW_BUBBLE	
EventSel=05H, UMask=94H	Nonzero segbase ld-op-st 1 bubble.
MISALIGN_MEM_REF.BUBBLE	
EventSel=05H, UMask=97H	Nonzero segbase 1 bubble.
SEGMENT_REG_LOADS.ANY	
EventSel=06H, UMask=80H	Number of segment register loads.
PREFETCH.SOFTWARE_PREFETCH	
EventSel=07H, UMask=0FH	Any Software prefetch.
PREFETCH.HW_PREFETCH	
EventSel=07H, UMask=10H	L1 hardware prefetch request.
PREFETCH.PREFETCHT0	
EventSel=07H, UMask=81H	Streaming SIMD Extensions (SSE) PrefetchT0 instructions executed.
PREFETCH.PREFETCHT1	
EventSel=07H, UMask=82H	Streaming SIMD Extensions (SSE) PrefetchT1 instructions executed.
PREFETCH.PREFETCHT2	
EventSel=07H, UMask=84H	Streaming SIMD Extensions (SSE) PrefetchT2 instructions executed.
PREFETCH.SW_L2	
EventSel=07H, UMask=86H	Streaming SIMD Extensions (SSE) PrefetchT1 and PrefetchT2 instructions executed.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
PREFETCH.PREFETCHNTA	
EventSel=07H, UMask=88H	Streaming SIMD Extensions (SSE) Prefetch NTA instructions executed.
PREFETCH.SOFTWARE_PREFETCH.AR	
EventSel=07H, UMask=8FH	Any Software prefetch.
DATA_TLB_MISSES.DTLB_MISS_LD	
EventSel=08H, UMask=05H	DTLB misses due to load operations.
DATA_TLB_MISSES.DTLB_MISS_ST	
EventSel=08H, UMask=06H	DTLB misses due to store operations.
DATA_TLB_MISSES.DTLB_MISS	
EventSel=08H, UMask=07H	Memory accesses that missed the DTLB.
DATA_TLB_MISSES.L0_DTLB_MISS_LD	
EventSel=08H, UMask=09H	L0 DTLB misses due to load operations.
DATA_TLB_MISSES.L0_DTLB_MISS_ST	
EventSel=08H, UMask=0AH	L0 DTLB misses due to store operations.
DISPATCH_BLOCKED.ANY	
EventSel=09H, UMask=20H	Memory cluster signals to block micro-op dispatch for any reason.
CPU_CLK_UNHALTED.CORE	
Architectural, Fixed	Core cycles when core is not halted.
CPU_CLK_UNHALTED.REF	
Architectural, Fixed	Reference cycles when core is not halted.
INST_RETIRED.ANY	
Architectural, Fixed	Instructions retired.
PAGE_WALKS.D_SIDE_WALKS	
EventSel=0CH, UMask=01H	Number of D-side only page walks.
PAGE_WALKS.D_SIDE_CYCLES	
EventSel=0CH, UMask=01H	Duration of D-side only page walks.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
PAGE_WALKS.I_SIDE_WALKS	
EventSel=0CH, UMask=02H	Number of I-Side page walks.
PAGE_WALKS.I_SIDE_CYCLES	
EventSel=0CH, UMask=02H	Duration of I-Side page walks.
PAGE_WALKS.WALKS	
EventSel=0CH, UMask=03H	Number of page-walks executed.
PAGE_WALKS.CYCLES	
EventSel=0CH, UMask=03H	Duration of page-walks in core cycles.
X87_COMP_OPS_EXE.ANY.S	
EventSel=10H, UMask=01H	Floating point computational micro-ops executed.
X87_COMP_OPS_EXE.FXCH.S	
EventSel=10H, UMask=02H	FXCH uops executed.
X87_COMP_OPS_EXE.ANY.AR	
EventSel=10H, UMask=81H, Precise	Floating point computational micro-ops retired.
X87_COMP_OPS_EXE.FXCH.AR	
EventSel=10H, UMask=82H, Precise	FXCH uops retired.
FP_ASSIST.S	
EventSel=11H, UMask=01H	Floating point assists.
FP_ASSIST.AR	
EventSel=11H, UMask=81H	Floating point assists for retired operations.
MUL.S	
EventSel=12H, UMask=01H	Multiply operations executed.
MUL.AR	
EventSel=12H, UMask=81H	Multiply operations retired.
DIV.S	
EventSel=13H, UMask=01H	Divide operations executed.
DIV.AR	
EventSel=13H, UMask=81H	Divide operations retired.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
CYCLES_DIV_BUSY	
EventSel=14H, UMask=01H	Cycles the divider is busy.
L2_ADS.SELF	
EventSel=21H, UMask=40H	Cycles L2 address bus is in use.
L2_DBUS_BUSY.SELF	
EventSel=22H, UMask=40H	Cycles the L2 cache data bus is busy.
L2_DBUS_BUSY_RD.SELF	
EventSel=23H, UMask=40H	Cycles the L2 transfers data to the core.
L2_LINES_IN.SELF.DEMAND	
EventSel=24H, UMask=40H	L2 cache misses.
L2_LINES_IN.SELF.PREFETCH	
EventSel=24H, UMask=50H	L2 cache misses.
L2_LINES_IN.SELF.ANY	
EventSel=24H, UMask=70H	L2 cache misses.
L2_M_LINES_IN.SELF	
EventSel=25H, UMask=40H	L2 cache line modifications.
L2_LINES_OUT.SELF.DEMAND	
EventSel=26H, UMask=40H	L2 cache lines evicted.
L2_LINES_OUT.SELF.PREFETCH	
EventSel=26H, UMask=50H	L2 cache lines evicted.
L2_LINES_OUT.SELF.ANY	
EventSel=26H, UMask=70H	L2 cache lines evicted.
L2_M_LINES_OUT.SELF.DEMAND	
EventSel=27H, UMask=40H	Modified lines evicted from the L2 cache.
L2_M_LINES_OUT.SELF.PREFETCH	
EventSel=27H, UMask=50H	Modified lines evicted from the L2 cache.
L2_M_LINES_OUT.SELF.ANY	
EventSel=27H, UMask=70H	Modified lines evicted from the L2 cache.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
L2_IFETCH.SELF.I_STATE	
EventSel=28H, UMask=41H	L2 cacheable instruction fetch requests.
L2_IFETCH.SELF.S_STATE	
EventSel=28H, UMask=42H	L2 cacheable instruction fetch requests.
L2_IFETCH.SELF.E_STATE	
EventSel=28H, UMask=44H	L2 cacheable instruction fetch requests.
L2_IFETCH.SELF.M_STATE	
EventSel=28H, UMask=48H	L2 cacheable instruction fetch requests.
L2_IFETCH.SELF.MESI	
EventSel=28H, UMask=4FH	L2 cacheable instruction fetch requests.
L2_LD.SELF.DEMAND.I_STATE	
EventSel=29H, UMask=41H	L2 cache reads.
L2_LD.SELF.DEMAND.S_STATE	
EventSel=29H, UMask=42H	L2 cache reads.
L2_LD.SELF.DEMAND.E_STATE	
EventSel=29H, UMask=44H	L2 cache reads.
L2_LD.SELF.DEMAND.M_STATE	
EventSel=29H, UMask=48H	L2 cache reads.
L2_LD.SELF.DEMAND.MESI	
EventSel=29H, UMask=4FH	L2 cache reads.
L2_LD.SELF.PREFETCH.I_STATE	
EventSel=29H, UMask=51H	L2 cache reads.
L2_LD.SELF.PREFETCH.S_STATE	
EventSel=29H, UMask=52H	L2 cache reads.
L2_LD.SELF.PREFETCH.E_STATE	
EventSel=29H, UMask=54H	L2 cache reads.
L2_LD.SELF.PREFETCH.M_STATE	
EventSel=29H, UMask=58H	L2 cache reads.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
L2_LD.SELF.PREFETCH.MESI	
EventSel=29H, UMask=5FH	L2 cache reads.
L2_LD.SELF.ANY.I_STATE	
EventSel=29H, UMask=71H	L2 cache reads.
L2_LD.SELF.ANY.S_STATE	
EventSel=29H, UMask=72H	L2 cache reads.
L2_LD.SELF.ANY.E_STATE	
EventSel=29H, UMask=74H	L2 cache reads.
L2_LD.SELF.ANY.M_STATE	
EventSel=29H, UMask=78H	L2 cache reads.
L2_LD.SELF.ANY.MESI	
EventSel=29H, UMask=7FH	L2 cache reads.
L2_ST.SELF.I_STATE	
EventSel=2AH, UMask=41H	L2 store requests.
L2_ST.SELF.S_STATE	
EventSel=2AH, UMask=42H	L2 store requests.
L2_ST.SELF.E_STATE	
EventSel=2AH, UMask=44H	L2 store requests.
L2_ST.SELF.M_STATE	
EventSel=2AH, UMask=48H	L2 store requests.
L2_ST.SELF.MESI	
EventSel=2AH, UMask=4FH	L2 store requests.
L2_LOCK.SELF.I_STATE	
EventSel=2BH, UMask=41H	L2 locked accesses.
L2_LOCK.SELF.S_STATE	
EventSel=2BH, UMask=42H	L2 locked accesses.
L2_LOCK.SELF.E_STATE	
EventSel=2BH, UMask=44H	L2 locked accesses.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
L2_LOCK.SELF.M_STATE	
EventSel=2BH, UMask=48H	L2 locked accesses.
L2_LOCK.SELF.MESI	
EventSel=2BH, UMask=4FH	L2 locked accesses.
L2_DATA_RQSTS.SELF.I_STATE	
EventSel=2CH, UMask=41H	All data requests from the L1 data cache.
L2_DATA_RQSTS.SELF.S_STATE	
EventSel=2CH, UMask=42H	All data requests from the L1 data cache.
L2_DATA_RQSTS.SELF.E_STATE	
EventSel=2CH, UMask=44H	All data requests from the L1 data cache.
L2_DATA_RQSTS.SELF.M_STATE	
EventSel=2CH, UMask=48H	All data requests from the L1 data cache.
L2_DATA_RQSTS.SELF.MESI	
EventSel=2CH, UMask=4FH	All data requests from the L1 data cache.
L2_LD_IFETCH.SELF.I_STATE	
EventSel=2DH, UMask=41H	All read requests from L1 instruction and data caches.
L2_LD_IFETCH.SELF.S_STATE	
EventSel=2DH, UMask=42H	All read requests from L1 instruction and data caches.
L2_LD_IFETCH.SELF.E_STATE	
EventSel=2DH, UMask=44H	All read requests from L1 instruction and data caches.
L2_LD_IFETCH.SELF.M_STATE	
EventSel=2DH, UMask=48H	All read requests from L1 instruction and data caches.
L2_LD_IFETCH.SELF.MESI	
EventSel=2DH, UMask=4FH	All read requests from L1 instruction and data caches.
L2_RQSTS.SELF.DEMAND.I_STATE	
EventSel=2EH, UMask=41H, Architectural	L2 cache demand requests from this core that missed the L2.
L2_RQSTS.SELF.DEMAND.S_STATE	
EventSel=2EH, UMask=42H	L2 cache requests.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
L2_RQSTS.SELF.DEMAND.E_STATE	
EventSel=2EH, UMask=44H	L2 cache requests.
L2_RQSTS.SELF.DEMAND.M_STATE	
EventSel=2EH, UMask=48H	L2 cache requests.
L2_RQSTS.SELF.DEMAND.MESI	
EventSel=2EH, UMask=4FH, Architectural	L2 cache demand requests from this core.
L2_RQSTS.SELF.PREFETCH.I_STATE	
EventSel=2EH, UMask=51H	L2 cache requests.
L2_RQSTS.SELF.PREFETCH.S_STATE	
EventSel=2EH, UMask=52H	L2 cache requests.
L2_RQSTS.SELF.PREFETCH.E_STATE	
EventSel=2EH, UMask=54H	L2 cache requests.
L2_RQSTS.SELF.PREFETCH.M_STATE	
EventSel=2EH, UMask=58H	L2 cache requests.
L2_RQSTS.SELF.PREFETCH.MESI	
EventSel=2EH, UMask=5FH	L2 cache requests.
L2_RQSTS.SELF.ANY.I_STATE	
EventSel=2EH, UMask=71H	L2 cache requests.
L2_RQSTS.SELF.ANY.S_STATE	
EventSel=2EH, UMask=72H	L2 cache requests.
L2_RQSTS.SELF.ANY.E_STATE	
EventSel=2EH, UMask=74H	L2 cache requests.
L2_RQSTS.SELF.ANY.M_STATE	
EventSel=2EH, UMask=78H	L2 cache requests.
L2_RQSTS.SELF.ANY.MESI	
EventSel=2EH, UMask=7FH	L2 cache requests.
L2_REJECT_BUSQ.SELF.DEMAND.I_STATE	
EventSel=30H, UMask=41H	Rejected L2 cache requests.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
L2_REJECT_BUSQ.SELF.DEMAND.S_STATE	
EventSel=30H, UMask=42H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.DEMAND.E_STATE	
EventSel=30H, UMask=44H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.DEMAND.M_STATE	
EventSel=30H, UMask=48H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.DEMAND.MESI	
EventSel=30H, UMask=4FH	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.PREFETCH.I_STATE	
EventSel=30H, UMask=51H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.PREFETCH.S_STATE	
EventSel=30H, UMask=52H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.PREFETCH.E_STATE	
EventSel=30H, UMask=54H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.PREFETCH.M_STATE	
EventSel=30H, UMask=58H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.PREFETCH.MESI	
EventSel=30H, UMask=5FH	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.ANY.I_STATE	
EventSel=30H, UMask=71H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.ANY.S_STATE	
EventSel=30H, UMask=72H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.ANY.E_STATE	
EventSel=30H, UMask=74H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.ANY.M_STATE	
EventSel=30H, UMask=78H	Rejected L2 cache requests.
L2_REJECT_BUSQ.SELF.ANY.MESI	
EventSel=30H, UMask=7FH	Rejected L2 cache requests.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
L2_NO_REQ.SELF	
EventSel=32H, UMask=40H	Cycles no L2 cache requests are pending.
EIST_TRANS	
EventSel=3AH, UMask=00H	Number of Enhanced Intel SpeedStep(R) Technology (EIST) transitions.
THERMAL_TRIP	
EventSel=3BH, UMask=C0H	Number of thermal trips.
CPU_CLK_UNHALTED.CORE_P	
EventSel=3CH, UMask=00H, Architectural	Core cycles when core is not halted.
CPU_CLK_UNHALTED.BUS	
EventSel=3CH, UMask=01H, Architectural	Bus cycles when core is not halted.
L1D_CACHE.REPL	
EventSel=40H, UMask=08H	L1 Data line replacements.
L1D_CACHE.EVICT	
EventSel=40H, UMask=10H	Modified cache lines evicted from the L1 data cache.
L1D_CACHE.REPLM	
EventSel=40H, UMask=48H	Modified cache lines allocated in the L1 data cache.
L1D_CACHE.ALL_REF	
EventSel=40H, UMask=83H	L1 Data reads and writes.
L1D_CACHE.LD	
EventSel=40H, UMask=A1H	L1 Cacheable Data Reads.
L1D_CACHE.ST	
EventSel=40H, UMask=A2H	L1 Cacheable Data Writes.
L1D_CACHE.ALL_CACHE_REF	
EventSel=40H, UMask=A3H	L1 Data Cacheable reads and writes.
BUS_REQUEST_OUTSTANDING.SELF	
EventSel=60H, UMask=40H	Outstanding cacheable data read bus requests duration.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
BUS_REQUEST_OUTSTANDING.ALL_AGENTS	
EventSel=60H, UMask=E0H	Outstanding cacheable data read bus requests duration.
BUS_BNR_DRV.THIS_AGENT	
EventSel=61H, UMask=00H	Number of Bus Not Ready signals asserted.
BUS_BNR_DRV.ALL_AGENTS	
EventSel=61H, UMask=20H	Number of Bus Not Ready signals asserted.
BUS_DRDY_CLOCKS.THIS_AGENT	
EventSel=62H, UMask=00H	Bus cycles when data is sent on the bus.
BUS_DRDY_CLOCKS.ALL_AGENTS	
EventSel=62H, UMask=20H	Bus cycles when data is sent on the bus.
BUS_LOCK_CLOCKS.SELF	
EventSel=63H, UMask=40H	Bus cycles when a LOCK signal is asserted.
BUS_LOCK_CLOCKS.ALL_AGENTS	
EventSel=63H, UMask=E0H	Bus cycles when a LOCK signal is asserted.
BUS_DATA_RCV.SELF	
EventSel=64H, UMask=40H	Bus cycles while processor receives data.
BUS_TRANS_BRD.SELF	
EventSel=65H, UMask=40H	Burst read bus transactions.
BUS_TRANS_BRD.ALL_AGENTS	
EventSel=65H, UMask=E0H	Burst read bus transactions.
BUS_TRANS_RFO.SELF	
EventSel=66H, UMask=40H	RFO bus transactions.
BUS_TRANS_RFO.ALL_AGENTS	
EventSel=66H, UMask=E0H	RFO bus transactions.
BUS_TRANS_WB.SELF	
EventSel=67H, UMask=40H	Explicit writeback bus transactions.
BUS_TRANS_WB.ALL_AGENTS	
EventSel=67H, UMask=E0H	Explicit writeback bus transactions.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
BUS_TRANS_IFETCH.SELF	
EventSel=68H, UMask=40H	Instruction-fetch bus transactions.
BUS_TRANS_IFETCH.ALL_AGENTS	
EventSel=68H, UMask=E0H	Instruction-fetch bus transactions.
BUS_TRANS_INVAL.SELF	
EventSel=69H, UMask=40H	Invalidate bus transactions.
BUS_TRANS_INVAL.ALL_AGENTS	
EventSel=69H, UMask=E0H	Invalidate bus transactions.
BUS_TRANS_PWR.SELF	
EventSel=6AH, UMask=40H	Partial write bus transaction.
BUS_TRANS_PWR.ALL_AGENTS	
EventSel=6AH, UMask=E0H	Partial write bus transaction.
BUS_TRANS_P.SELF	
EventSel=6BH, UMask=40H	Partial bus transactions.
BUS_TRANS_P.ALL_AGENTS	
EventSel=6BH, UMask=E0H	Partial bus transactions.
BUS_TRANS_IO.SELF	
EventSel=6CH, UMask=40H	IO bus transactions.
BUS_TRANS_IO.ALL_AGENTS	
EventSel=6CH, UMask=E0H	IO bus transactions.
BUS_TRANS_DEF.SELF	
EventSel=6DH, UMask=40H	Deferred bus transactions.
BUS_TRANS_DEF.ALL_AGENTS	
EventSel=6DH, UMask=E0H	Deferred bus transactions.
BUS_TRANS_BURST.SELF	
EventSel=6EH, UMask=40H	Burst (full cache-line) bus transactions.
BUS_TRANS_BURST.ALL_AGENTS	
EventSel=6EH, UMask=E0H	Burst (full cache-line) bus transactions.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
BUS_TRANS_MEM.SELF	
EventSel=6FH, UMask=40H	Memory bus transactions.
BUS_TRANS_MEM.ALL_AGENTS	
EventSel=6FH, UMask=E0H	Memory bus transactions.
BUS_TRANS_ANY.SELF	
EventSel=70H, UMask=40H	All bus transactions.
BUS_TRANS_ANY.ALL_AGENTS	
EventSel=70H, UMask=E0H	All bus transactions.
EXT_SNOOP.THIS_AGENT.CLEAN	
EventSel=77H, UMask=01H	External snoops.
EXT_SNOOP.THIS_AGENT.HIT	
EventSel=77H, UMask=02H	External snoops.
EXT_SNOOP.THIS_AGENT.HITM	
EventSel=77H, UMask=08H	External snoops.
EXT_SNOOP.THIS_AGENT.ANY	
EventSel=77H, UMask=0BH	External snoops.
EXT_SNOOP.ALL_AGENTS.CLEAN	
EventSel=77H, UMask=21H	External snoops.
EXT_SNOOP.ALL_AGENTS.HIT	
EventSel=77H, UMask=22H	External snoops.
EXT_SNOOP.ALL_AGENTS.HITM	
EventSel=77H, UMask=28H	External snoops.
EXT_SNOOP.ALL_AGENTS.ANY	
EventSel=77H, UMask=2BH	External snoops.
BUS_HIT_DRV.THIS_AGENT	
EventSel=7AH, UMask=00H	HIT signal asserted.
BUS_HIT_DRV.ALL_AGENTS	
EventSel=7AH, UMask=20H	HIT signal asserted.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
BUS_HITM_DRV.THIS_AGENT	
EventSel=7BH, UMask=00H	HITM signal asserted.
BUS_HITM_DRV.ALL_AGENTS	
EventSel=7BH, UMask=20H	HITM signal asserted.
BUSQ_EMPTY.SELF	
EventSel=7DH, UMask=40H	Bus queue is empty.
SNOOP_STALL_DRV.SELF	
EventSel=7EH, UMask=40H	Bus stalled for snoops.
SNOOP_STALL_DRV.ALL_AGENTS	
EventSel=7EH, UMask=E0H	Bus stalled for snoops.
BUS_IO_WAIT.SELF	
EventSel=7FH, UMask=40H	IO requests waiting in the bus queue.
ICACHE.HIT	
EventSel=80H, UMask=01H	Icache hit.
ICACHE.MISSES	
EventSel=80H, UMask=02H	Icache miss.
ICACHE.ACCESSES	
EventSel=80H, UMask=03H	Instruction fetches.
ITLB.HIT	
EventSel=82H, UMask=01H	ITLB hits.
ITLB.MISSES	
EventSel=82H, UMask=02H, Precise	ITLB misses.
ITLB.FLUSH	
EventSel=82H, UMask=04H	ITLB flushes.
CYCLES_ICACHE_MEM_STALLED.ICACHE_MEM_STALLED	
EventSel=86H, UMask=01H	Cycles during which instruction fetches are stalled.
DECODE_STALL.PFB_EMPTY	
EventSel=87H, UMask=01H	Decode stall due to PFB empty.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
DECODE_STALL.IQ_FULL	
EventSel=87H, UMask=02H	Decode stall due to IQ full.
BR_INST_TYPE_RETIRED.COND	
EventSel=88H, UMask=01H	All macro conditional branch instructions.
BR_INST_TYPE_RETIRED.UNCOND	
EventSel=88H, UMask=02H	All macro unconditional branch instructions, excluding calls and indirects.
BR_INST_TYPE_RETIRED.IND	
EventSel=88H, UMask=04H	All indirect branches that are not calls.
BR_INST_TYPE_RETIRED.RET	
EventSel=88H, UMask=08H	All indirect branches that have a return mnemonic.
BR_INST_TYPE_RETIRED.DIR_CALL	
EventSel=88H, UMask=10H	All non-indirect calls.
BR_INST_TYPE_RETIRED.IND_CALL	
EventSel=88H, UMask=20H	All indirect calls, including both register and memory indirect.
BR_INST_TYPE_RETIRED.COND_TAKEN	
EventSel=88H, UMask=41H	Only taken macro conditional branch instructions.
BR_MISSP_TYPE_RETIRED.COND	
EventSel=89H, UMask=01H	Mispredicted cond branch instructions retired.
BR_MISSP_TYPE_RETIRED.IND	
EventSel=89H, UMask=02H	Mispredicted ind branches that are not calls.
BR_MISSP_TYPE_RETIRED.RETURN	
EventSel=89H, UMask=04H	Mispredicted return branches.
BR_MISSP_TYPE_RETIRED.IND_CALL	
EventSel=89H, UMask=08H	Mispredicted indirect calls, including both register and memory indirect. .
BR_MISSP_TYPE_RETIRED.COND_TAKEN	
EventSel=89H, UMask=11H	Mispredicted and taken cond branch instructions retired.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
UOPS.MS_CYCLES	
EventSel=A9H, UMask=01H, CMask=1	This event counts the cycles where 1 or more uops are issued by the micro-sequencer (MS), including microcode assists and inserted flows, and written to the IQ. .
MACRO_INSTS.NON_CISC_DECODED	
EventSel=AAH, UMask=01H	Non-CISC nacro instructions decoded.
MACRO_INSTS.CISC_DECODED	
EventSel=AAH, UMask=02H	CISC macro instructions decoded.
MACRO_INSTS.ALL_DECODED	
EventSel=AAH, UMask=03H	All Instructions decoded.
SIMD_UOPS_EXEC.S	
EventSel=B0H, UMask=00H	SIMD micro-ops executed (excluding stores).
SIMD_UOPS_EXEC.AR	
EventSel=B0H, UMask=80H, Precise	SIMD micro-ops retired (excluding stores).
SIMD_SAT_UOP_EXEC.S	
EventSel=B1H, UMask=00H	SIMD saturated arithmetic micro-ops executed.
SIMD_SAT_UOP_EXEC.AR	
EventSel=B1H, UMask=80H	SIMD saturated arithmetic micro-ops retired.
SIMD_UOP_TYPE_EXEC.MUL.S	
EventSel=B3H, UMask=01H	SIMD packed multiply micro-ops executed.
SIMD_UOP_TYPE_EXEC.SHIFT.S	
EventSel=B3H, UMask=02H	SIMD packed shift micro-ops executed.
SIMD_UOP_TYPE_EXEC.PACK.S	
EventSel=B3H, UMask=04H	SIMD packed micro-ops executed.
SIMD_UOP_TYPE_EXEC.UNPACK.S	
EventSel=B3H, UMask=08H	SIMD unpacked micro-ops executed.
SIMD_UOP_TYPE_EXEC.LOGICAL.S	
EventSel=B3H, UMask=10H	SIMD packed logical micro-ops executed.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
SIMD_UOP_TYPE_EXEC.ARITHMETIC.S	
EventSel=B3H, UMask=20H	SIMD packed arithmetic micro-ops executed.
SIMD_UOP_TYPE_EXEC.MUL.AR	
EventSel=B3H, UMask=81H	SIMD packed multiply micro-ops retired.
SIMD_UOP_TYPE_EXEC.SHIFT.AR	
EventSel=B3H, UMask=82H	SIMD packed shift micro-ops retired.
SIMD_UOP_TYPE_EXEC.PACK.AR	
EventSel=B3H, UMask=84H	SIMD packed micro-ops retired.
SIMD_UOP_TYPE_EXEC.UNPACK.AR	
EventSel=B3H, UMask=88H	SIMD unpacked micro-ops retired.
SIMD_UOP_TYPE_EXEC.LOGICAL.AR	
EventSel=B3H, UMask=90H	SIMD packed logical micro-ops retired.
SIMD_UOP_TYPE_EXEC.ARITHMETIC.AR	
EventSel=B3H, UMask=A0H	SIMD packed arithmetic micro-ops retired.
INST_RETIRED.ANY_P	
EventSel=C0H, UMask=00H, Precise	Instructions retired (precise event).
UOPS_RETIRED.ANY	
EventSel=C2H, UMask=10H	Micro-ops retired.
UOPS_RETIRED.STALLED_CYCLES	
EventSel=C2H, UMask=10H	Cycles no micro-ops retired.
UOPS_RETIRED.STALLS	
EventSel=C2H, UMask=10H	Periods no micro-ops retired.
MACHINE_CLEARS.SMC	
EventSel=C3H, UMask=01H	Self-Modifying Code detected.
BR_INST_RETIRED.ANY	
EventSel=C4H, UMask=00H, Architectural	Retired branch instructions.
BR_INST_RETIRED.PRED_NOT_TAKEN	
EventSel=C4H, UMask=01H	Retired branch instructions that were predicted not-taken.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
BR_INST_RETIRED.MISPRED_NOT_TAKEN	
EventSel=C4H, UMask=02H	Retired branch instructions that were mispredicted not-taken.
BR_INST_RETIRED.PRED_TAKEN	
EventSel=C4H, UMask=04H	Retired branch instructions that were predicted taken.
BR_INST_RETIRED.MISPRED_TAKEN	
EventSel=C4H, UMask=08H	Retired branch instructions that were mispredicted taken.
BR_INST_RETIRED.TAKEN	
EventSel=C4H, UMask=0CH	Retired taken branch instructions.
BR_INST_RETIRED.ANY1	
EventSel=C4H, UMask=0FH	Retired branch instructions.
BR_INST_RETIRED.MISPRED.PS	
EventSel=C5H, UMask=00H, Precise	Retired mispredicted branch instructions.
BR_INST_RETIRED.MISPRED	
EventSel=C5H, UMask=00H, Architectural	Retired mispredicted branch instructions (precise event).
CYCLES_INT_MASKED.CYCLES_INT_MASKED	
EventSel=C6H, UMask=01H	Cycles during which interrupts are disabled.
CYCLES_INT_MASKED.CYCLES_INT_PENDING_AND_MASKED	
EventSel=C6H, UMask=02H	Cycles during which interrupts are pending and disabled.
SIMD_INST_RETIRED.PACKED_SINGLE	
EventSel=C7H, UMask=01H	Retired Streaming SIMD Extensions (SSE) packed-single instructions.
SIMD_INST_RETIRED.SCALAR_SINGLE	
EventSel=C7H, UMask=02H	Retired Streaming SIMD Extensions (SSE) scalar-single instructions.
SIMD_INST_RETIRED.SCALAR_DOUBLE	
EventSel=C7H, UMask=08H	Retired Streaming SIMD Extensions 2 (SSE2) scalar-double instructions.
SIMD_INST_RETIRED.VECTOR	
EventSel=C7H, UMask=10H	Retired Streaming SIMD Extensions 2 (SSE2) vector instructions.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
HW_INT_RCV	
EventSel=C8H, UMask=00H	Hardware interrupts received.
SIMD_COMP_INST_RETIRED.PACKED_SINGLE	
EventSel=CAH, UMask=01H	Retired computational Streaming SIMD Extensions (SSE) packed-single instructions.
SIMD_COMP_INST_RETIRED.SCALAR_SINGLE	
EventSel=CAH, UMask=02H	Retired computational Streaming SIMD Extensions (SSE) scalar-single instructions.
SIMD_COMP_INST_RETIRED.SCALAR_DOUBLE	
EventSel=CAH, UMask=08H	Retired computational Streaming SIMD Extensions 2 (SSE2) scalar-double instructions.
MEM_LOAD_RETIRED.L2_HIT	
EventSel=CBH, UMask=01H	Retired loads that hit the L2 cache (precise event).
MEM_LOAD_RETIRED.L2_MISS	
EventSel=CBH, UMask=02H	Retired loads that miss the L2 cache.
MEM_LOAD_RETIRED.DTLB_MISS	
EventSel=CBH, UMask=04H	Retired loads that miss the DTLB (precise event).
MEM_LOAD_RETIRED.DTLB_MISS.PS	
EventSel=CBH, UMask=04H, Precise	Retired loads that miss the DTLB (precise event).
MEM_LOAD_RETIRED.L2_HIT.PS	
EventSel=CBH, UMask=81H, Precise	Retired loads that hit the L2 cache (precise event).
MEM_LOAD_RETIRED.L2_MISS.PS	
EventSel=CBH, UMask=82H, Precise	Retired loads that miss the L2 cache (precise event).
SIMD_ASSIST	
EventSel=CDH, UMask=00H	SIMD assists invoked.
SIMD_INSTR_RETIRED	
EventSel=CEH, UMask=00H	SIMD Instructions retired.
SIMD_SAT_INSTR_RETIRED	
EventSel=CFH, UMask=00H	Saturated arithmetic instructions retired.

Table 18: Performance Events of the Processor Core Supported by Bonnell Microarchitecture

Event Name	
Configuration	Description
RESOURCE_STALLS.DIV_BUSY	
EventSel=DCH, UMask=02H	Cycles issue is stalled due to div busy.
BR_INST_DECODED	
EventSel=E0H, UMask=01H	Branch instructions decoded.
BOGUS_BR	
EventSel=E4H, UMask=01H	Bogus branches.
BACLEARS.ANY	
EventSel=E6H, UMask=01H	BACLEARS asserted.