CPUID Specification

Publication #	**25481** Revision: **2.34**
Issue Date:	**September 2010**

Advanced Micro Devices 

Trademarks

Table of Contents

Revision History

Date	Rev	Description
September 2010	2.34	• 1.2 [Conventions]: Text needs to be exposed related to "_YYY". • CPUID Fn0000_0006_ECX[EffFreq]: Added. • CPUID Fn0000_0007_EBX_x0[BMI]: Added. • CPUID Fn0000_0001_ECX[AVX, OSXSAVE, XSAVE, AES, SSE42, PCLMULQDQ]: Added. • CPUID Fn0000_0001_ECX[F16C]: Added. • CPUID Fn0000_0001_ECX[FMA]: Added. • CPUID Fn0000_000D_EAX_x0-CPUID Fn0000_000D_EDX_x3E: Added. • CPUID Fn8000_0001_ECX[FMA4, LWP, XOP]: Added. • CPUID Fn8000_0001_ECX[SSE5]: Dropped and replaced by XOP. • CPUID Fn8000_0001_ECX[TopologyExtensions]: Added. • CPUID Fn8000_0001_ECX[TBM]: Added. • CPUID Fn8000_0007_EDX[CPB]: Added. • CPUID Fn8000_0007_EDX[EffFreqRO]: Added. • CPUID Fn8000_000A_EDX[DecodeAssists, FlushByAsid, VmcbClean, TscRateMsr]: Added. • CPUID Fn8000_000A_EDX[PauseFilterThreshold]: Added. • CPUID Fn8000_001B_EAX[RipInvalidChk, OpCntExt]: Added. • CPUID Fn8000_001C_EAX [Lightweight Profiling Capabilities 0]-CPUID Fn8000_001C_EDX: Added. • CPUID Fn8000_001D_EAX_x[N:0]-CPUID Fn8000_001E_EDX: Added.
November 2009	2.32	• CPUID Fn8000_000A_EDX[SseIsa10Compat]: Dropped. • CPUID Fn8000_000A_EDX[PauseFilter]: Added. • CPUID Fn8000_0001_ECX[NodeId]: Added. • CPUID Fn8000_001B_EAX[BrnTrgt, OpCnt, RdWrOpCnt, OpSam, FetchSam, IBSFFV]: Added.
October 2008	2.30	• CPUID Fn0000_0001_ECX[31]: Updated. • CPUID Fn4000_00[FF:00]: Added.
April 2008	2.28	• 3.1 [Legacy Method] on page 36: Clarified. • CPUID Fn0000_0001_ECX[SSE41]: Added. • CPUID Fn0000_0001_ECX[SSSE3]: Added. • CPUID Fn8000_0001_ECX[SSE5]: Added. • CPUID Fn8000_0001_ECX[IBS]: Added. • CPUID Fn8000_0008_EAX[GuestPhysAddrSize]: Added. • CPUID Fn8000_0008_EAX[PhysAddrSize]: Updated. • CPUID Fn8000_000A_EDX[SseIsa10Compat]: Added.

Date	Rev	Description
July 2007	2.26	• CPUID Fn0000_0001_ECX[Monitor, POPCNT]: Added. • CPUID Fn0000_000[4:2]: Added as reserved. • CPUID Fn0000_0005_EAX-EDX: Added. • CPUID Fn0000_0005_ECX[IBE, EMX]: Added. • CPUID Fn8000_0001_EBX[PkgType[3:0]]: Added. • CPUID Fn8000_0001_ECX[WDT, SKINIT, OSVW, 3DNowPrefetch, MisAlignSse, SSE4A, ABM, ExtApicSpace]: Added. • CPUID Fn8000_0001_EDX[Page1GB]: Added. • CPUID Fn8000_0006_EDX[L3Size, L3Assoc, L3LinesPerTag, L3LineSize]: Added. • CPUID Fn8000_0006_EAX: Table 4: Added additional associativity definitions. • CPUID Fn8000_0007_EDX[HwPState, 100MhzSteps]: Added. • CPUID Fn8000_000A_EDX[NRIPS, SVML, LbrVirt, NP]: Added. • CPUID Fn8000_0019_EAX [TLB 1GB Page Identifiers]-EBX: Added. • CPUID Fn8000_001A_EAX [Performance Optimization Identifiers]: Added. • CPUID Fn8000_001B_EAX [Instruction Based Sampling Identifiers]: Added.
January 2006	2.18	• CPUID Fn8000_0007_EDX[8]: Renamed from TscPStateInvariant to TscInvariant. • CPUID Fn8000_0008_ECX[ApicIdCoreIdSize[3:0]]: Added.
September 2005	2.16	• Reformatted document for clarity. • Moved the chapter titled “Programming The Processor Name String” to the processor revision guide. • Added definition for HTT, CmpLegacy, and LogicalProcessorCount for multi-threading. See 3.1 [Legacy Method], CPUID Fn8000_0001_ECX[CmpLegacy], CPUID Fn0000_0001_EBX[LogicalProcessorCount], CPUID Fn0000_0001_EDX[HTT]. • CPUID Fn8000_0001_EBX[BrandId[15:12]]: Added. • CPUID Fn8000_0001_ECX[SVM, CMPXCHG16B, AltMovCr8, LahfSahf]: Added. • CPUID Fn8000_0001_EDX[RDTSCP]: Added.
See revision 2.15 for the change history prior to rev 2.16.		

1 Overview

This document specifies the CPUID instruction functions and return values in the EAX, EBX, ECX, and EDX registers, for all AMD processors of family 0Fh or greater. The architectural definition of the CPUID instruction is also documented in the section titled “CPUID” in the AMD64 Architectural Programmer's Manual Volume 3: General-Purpose and System Instructions, #24594.

1.1 Reference Documents

The following documents provide background information:

- *AMD64 Architecture Programmer’s Manual Volume 1: Application Programming*, #24592. (APM1)
- *AMD64 Architecture Programmer’s Manual Volume 2: System Programming*, #24593. (APM2)
- *AMD64 Architecture Programmer’s Manual Volume 3: General Purpose and System Instructions*, #24594. (APM3)
- *AMD64 Architecture Programmer’s Manual Volume 4: 128-Bit Media Instructions*, #26568. (APM4)
- *AMD64 Architecture Programmer’s Manual Volume 5: 64-Bit Media and x87 Floating-Point Instructions*, #26569. (APM5)
- *AMD64 Architecture Programmer’s Manual Volume 6: 128-Bit and 256-Bit, XOP, and FMA4 Instructions*, #43479. (APM6)
- *AMD64 Architecture Programmer’s Manual Documentation Updates*, #45988. (APMU)
- *BIOS and Kernel Developer’s Guide (BKDG)* for the specific result value for each of the registers affected by the CPUID instruction for each function. The order number varies by processor family and sometimes by processor model.
- *AMD Processor Recognition Application Note*, #20734, for the definition of CPUID for processors belonging to family 0Fh or less.
- The appropriate revision guide for your target processor describes the process of programming the processor name string.
- *AMD64 Technology Lightweight Profiling Specification*, #43724. (LWP Spec)
- *Software Optimization Guide for AMD Family 10h Processors*, #40546. (SWOG10)

1.2 Conventions

The following conventions are used in this document:

- The syntax for CPUID functions is: CPUID FnXXXX_XXXX_RRR_xYYY[FieldName].
  - XXXX_XXXX is a hex constant equal to the EAX input.
  - RRR is one of {EDX, ECX, EBX, EAX}.
  - YYY (optional) is a hex constant equal to the ECX input.
  - E.g. CPUID Fn8000_0001_EDX[SVM], CPUID Fn0000_000D_EAX_x0[XFeatureEnabledSizeMax].
- Unless otherwise specified, the 1-bit feature fields are encoded as 1 = Feature is supported by the processor; 0 = Feature is not supported by the processor.
- References to the *AMD64 Architecture Programmer’s Manual* are abbreviated as APM*n*, where *n* specifies the volume, from 1 to 6.
- The 8-bit family of a processor (Family[7:0]) is determined by CPUID Fn0000_0001_EAX[Extended-

Family,BaseFamily].

1.2.1 Numbering

- **Binary numbers**. Binary numbers are indicated by appending a “b” at the end; e.g., 0110b.
- **Decimal numbers**. Unless specified otherwise, all numbers are decimal. This rule does not apply to the register mnemonics; register mnemonics all utilize hexadecimal numbering.
- **Hexadecimal numbers**. Hexadecimal numbers are indicated by appending an “h” to the end; e.g., 45f8h.
- **Underscores in numbers**. Underscores are used to break up numbers to make them more readable. They do not imply any operation; e.g., 0110_1100b.

1.2.2 Arithmetic and Logical Operators

In this document, formulas follow some Verilog conventions for logic equations.

Table 1: Arithmetic and Logical Operators

Operator	Definition
{}	Curly brackets are used to indicate a group of bits that are concatenated together. Each set of bits is separated by a comma. E.g., {Addr[3:2], Xlate[3:0]} represents a 6-bit value; the two MSBs are Addr[3:2] and the four LSB’s are Xlate[3:0].
|	Bitwise OR operator. E.g. (01b | 10b == 11b).
||	Logical OR operator. E.g. (01b || 10b == 1b); logical treats multibit operand as 1 if >=1 and produces a 1-bit result.
&	Bitwise AND operator. E.g. (01b & 10b == 00b).
&&	Logical AND operator. E.g. (01b && 10b == 1b); logical treats multibit operand as 1 if >=1 and produces a 1-bit result.
^	Bitwise exclusive-OR operator; sometimes used as “raised to the power of” as well, as indicated by the context in which it is used. E.g. (01b ^ 10b == 11b). E.g. (2^2 == 4).
~	Bitwise NOT operator (also known as one’s complement). E.g. (~10b == 01b).
!	Logical NOT operator. E.g. (!10b == 0b); logical treats multibit operand as 1 if >=1 and produces a 1-bit result.
==	Logical “is equal to” operator.
!=	Logical “is not equal to” operator.
<=	Less than or equal operator.
>=	Greater than or equal operator.
*****	Arithmetic multiplication operator.
/	Arithmetic division operator.

1.3 Definitions

The following definitions are used in this document:

- **APM*n***. Abbreviation for *AMD64 Architecture Programmer’s Manual: Volume* n.
- **APMU**. Abbreviation for *AMD64 Architecture Programmer’s Manual: Documentation Updates*.

- **BKDG**. BIOS and Kernel Developer's Guide
- **CMP**. Chip multi-processing. Refers to processors that include multiple cores.
- **Core**. Executes x86 instructions and contains a set of MSRs and APIC registers.
- **DW or Doubleword**. Double word. A 32-bit value.
- **Family**. An 8-bit value that identifies one or more processors as belonging to a group that possess some common definition for software or hardware purposes. See CPUID Fn0000_0001_EAX.
- **GB or Gbyte**. Gigabyte; 1,024 Mbytes.
- **HTC**. Hardware thermal control.
- **KB or Kbyte**. Kilobyte; 1024 bytes.
- **LWP**. Lightweight Profiling.
- **LWP Spec**. Abbreviation for *AMD64 Technology Lightweight Profiling Specification*, #43724.
- **MB or Mbyte**. Megabyte; 1024 Kbytes.
- **Model**. Model specifies one instance of a processor family. See CPUID Fn0000_0001_EAX.
- **MSR**. Model specific register. The core includes several MSRs for general configuration and control.
- **NB**. Northbridge. The transaction routing block of the processor.
- **Processor**. A single package that contains one or more cores.
- **QW or Quadword**. Quad word. A 64-bit value.
- **OW or Octword**. Eight word. A 128-bit value.
- **RAZ**. Read as zero. Writes are ignored.
- **Reserved**. Field is reserved for future use. Software may not depend on the state of reserved fields.
- **STC**. Software thermal control.
- **SVM**. Secure virtual machine.
- **SWOG10**. Abbreviation for *Software Optimization Guide for AMD Family 10h Processors*, #40546.
- **Thread**. One architectural context for instruction execution.

1.4 Standard, Extended, and Undefined Functions

The CPUID instruction supports two sets or ranges of functions, standard and extended.

- The smallest function number of the standard function range is Fn0000_0000. The largest function number of the standard function range, for a particular implementation, is returned in CPUID Fn0000_0000_EAX.
- The smallest function number of the extended function range is Fn8000_0000. The largest function number of the extended function range, for a particular implementation, is returned in CPUID Fn8000_0000_EAX.

Functions that are neither standard nor extended are undefined and should not be relied upon.

2 CPUID Function Specification

This chapter defines each of the supported CPUID functions, both standard and extended.

CPUID Fn0000_0000_EAX Largest Standard Function Number

Bits	Description
31:0	**LFuncStd: largest standard function**. The largest CPUID standard function input value supported by the processor implementation. See “Standard, Extended, and Undefined Functions” on page 9.

CPUID Fn0000_0000_E[D,C,B]X Processor Vendor

CPUID Fn0000_0000_E[D,C,B]X and CPUID Fn8000_0000_E[D,C,B]X return the same value.

Table 2: CPUID Fn0000_0000_E[D,C,B]X value

Register	Value	Description
CPUID Fn0000_0000_EBX	6874_7541h	The ASCII characters “h t u A”.
CPUID Fn0000_0000_ECX	444D_4163h	The ASCII characters “D M A c”.
CPUID Fn0000_0000_EDX	6974_6E65h	The ASCII characters “i t n e”.

Bits	Description
31:0	**Vendor**. The 12 8-bit ASCII character codes to create the string “AuthenticAMD”.

CPUID Fn0000_0001_EAX Family, Model, Stepping Identifiers

The value returned in EAX is the processor identification signature and is identical for CPUID Fn0000_0001 and CPUID Fn8000_0001. This function is an identical copy of CPUID Fn8000_0001_EAX. Reserved fields should be masked before using the value of EAX for processor identification purposes. Three values are used by software to identify a processor: Family, Model, and Stepping.

The processor *Family* identifies one or more processors as belonging to a group that possesses some common definition for software or hardware purposes. The *Model* specifies one instance of a processor family. The *Stepping* identifies a particular version of a specific model. Therefore, Family, Model and Stepping, when taken together, form a unique identification or signature for a processor.

The **Family** is an 8-bit value and is defined as: **Family[7:0] =** ({0000b,BaseFamily[3:0]} + ExtendedFamily[7:0]). For example, if BaseFamily[3:0] = Fh and ExtendedFamily[7:0] = 01h, then Family[7:0] = 10h. If BaseFamily[3:0] is less than Fh then ExtendedFamily[7:0] is reserved and Family is equal to BaseFamily[3:0].

Model is an 8-bit value and is defined as: **Model[7:0]** = {ExtendedModel[3:0],BaseModel[3:0]}. For example, if ExtendedModel[3:0] = Eh and BaseModel[3:0] = 8h, then Model[7:0] = E8h. If BaseFamily[3:0] is less than 0Fh then ExtendedModel[3:0] is reserved and Model is equal to BaseModel[3:0].

Stepping is analogous to a revision number.

Bits	Description
31:28	Reserved.
27:20	**ExtFamily: processor extended family**. See above for definition of Family[7:0].
19:16	**ExtModel: processor extended model**. See above for definition of Model[7:0].
15:12	Reserved.
11:8	**BaseFamily: base processor family**. See above for definition of Family[7:0].
7:4	**BaseModel: base processor model**. See above for definition of Model[7:0].
3:0	**Stepping: processor stepping**. Processor stepping (revision) for a specific model.

CPUID Fn0000_0001_EBX LocalApicId, LogicalProcessorCount, CLFlush

This function returns miscellaneous information regarding the processor brand, the number of logical threads per processor socket, the CLFLUSH instruction and APIC.

Bits	Description
31:24	**LocalApicId: Initial local APIC physical ID**. The 8-bit value assigned to the local APIC physical ID register at power-up. Some of the bits of LocalApicId represent the core within a processor and other bits represent the processor ID. See the APIC20 “APIC ID” register in the processor BKDG for details.
23:16	**LogicalProcessorCount: logical processor count**. If CPUID Fn0000_0001_EDX[HTT] = 1 then LogicalProcessorCount is the number of cores per processor. If CPUID Fn0000_0001_EDX[HTT] = 0 then LogicalProcessorCount is reserved. See 3.1 [Legacy Method].
15:8	**CLFlush: CLFLUSH size**. Specifies the size of a cache line in quadwords flushed by the CLFLUSH instruction. See “CLFLUSH” in APM3.
7:0	**8BitBrandId: 8-bit brand ID**. This field, in conjunction with CPUID Fn8000_0001_EBX[BrandId], is used by the BIOS to generate the processor name string. See the appropriate processor revision guide for how to program the processor name string.

CPUID Fn0000_0001_ECX Feature Identifiers

This function contains the following miscellaneous feature identifiers.

Bits	Description
31	RAZ. Reserved for use by hypervisor to indicate guest status.
30	Reserved.
29	**F16C: half-precision convert instruction support**.
28	**AVX: AVX instruction support**. See APM6.
27	**OSXSAVE: XSAVE (and related) instructions are enabled**.

Bits	Description
26	**XSAVE: XSAVE (and related) instructions are supported by hardware**. .
25	**AES: AES instruction support**.
24	Reserved.
23	**POPCNT: POPCNT instruction**. See “POPCNT” in APM3.
22:21	Reserved.
20	**SSE42: SSE4.2 instruction support**.
19	**SSE41: SSE4.1 instruction support**.
18:14	Reserved.
13	**CMPXCHG16B: CMPXCHG16B instruction**. See “CMPXCHG16B” in APM3.
12	**FMA: FMA instruction support**.
11:10	Reserved.
9	**SSSE3: supplemental SSE3 instruction support**.
8:4	Reserved.
3	**MONITOR: MONITOR/MWAIT instructions**. See “MONITOR” and “MWAIT” in APM3.
2	Reserved.
1	**PCLMULQDQ**: **PCLMULQDQ instruction support**. See “PCLMUL Instructions” in APMU.
0	**SSE3: SSE3 instruction support**. See Appendix D “Instruction Subsets and CPUID Feature Sets” in APM3 for the list of instructions covered by the SSE3 feature bit. See APM4 for the definition of the SSE3 instructions.

CPUID Fn0000_0001_EDX Feature Identifiers

This function contains the following miscellaneous feature identifiers.

Bits	Description
31:29	Reserved.
28	**HTT: hyper-threading technology**. Indicates either that there is more than one thread per core or more than one core per processor. See “Legacy Method” on page 36.
27	Reserved.
26	**SSE2: SSE2 instruction support**. See Appendix D “CPUID Feature Sets” in APM3.
25	**SSE: SSE instruction support**. See Appendix D “CPUID Feature Sets” in APM3 appendix and “64-Bit Media Programming” in APM1.
24	**FXSR: FXSAVE and FXRSTOR instructions**. See “FXSAVE” and “FXRSTOR” in APM4.
23	**MMX: MMX™ instructions**. See Appendix D “CPUID Feature Sets” in APM3 and “128-Bit Media and Scientific Programming” in APM1.
22:20	Reserved.
19	**CLFSH: CLFLUSH instruction support**. See “CLFLUSH” in APM3.
18	Reserved.
17	**PSE36: page-size extensions**. The PDE[20:13] supplies physical address [39:32]. See “Page Translation and Protection” in APM2.
16	**PAT: page attribute table**. See “Page-Attribute Table Mechanism” in APM2.
15	**CMOV: conditional move instructions**. See “CMOV”, “FCMOV” in APM3.
14	**MCA: machine check architecture**. See “Machine Check Mechanism” in APM2.
13	**PGE: page global extension**. See “Page Translation and Protection” in APM2.
12	**MTRR: memory-type range registers**. See “Page Translation and Protection” in APM2.
11	**SysEnterSysExit: SYSENTER and SYSEXIT instructions**. See “SYSENTER”, “SYSEXIT“ in APM3.
10	Reserved.
9	**APIC: advanced programmable interrupt controller**. Indicates APIC exists and is enabled. See “Exceptions and Interrupts” in APM2.
8	**CMPXCHG8B: CMPXCHG8B instruction**. See “CMPXCHG8B” in APM3.
7	**MCE: Machine check exception**. See “Machine Check Mechanism” in APM2.
6	**PAE: physical-address extensions**. Indicates support for physical addresses ≥ 32b. Number of physical address bits above 32b is implementation specific. See “Page Translation and Protection” in APM2.
5	**MSR: AMD model-specific registers**. Indicates support for AMD model-specific registers (MSRs), with RDMSR and WRMSR instructions. See “Model Specific Registers” in APM2.
4	**TSC: time stamp counter**. RDTSC and RDTSCP instruction support. See “Debug and Performance Resources” in APM2.
3	**PSE: page-size extensions**. See “Page Translation and Protection” in APM2.

Bits	Description
2	**DE: debugging extensions**. See “Debug and Performance Resources” in APM2.
1	**VME: virtual-mode enhancements**. CR4.VME, CR4.PVI, software interrupt indirection, expansion of the TSS with the software, indirection bitmap, EFLAGS.VIF, EFLAGS.VIP. See “System Resources” in APM2.
0	**FPU: x87 floating point unit on-chip**. See “x87 Floating Point Programming” in APM1.

CPUID Fn0000_000[4:2] Reserved

Bits	Description
31:0	Reserved.

CPUID Fn0000_0005_EAX Monitor/MWait

This function contains the feature identifiers for the MONITOR and MWAIT instructions. See “MONITOR” and “MWAIT” in APM3.

Bits	Description
31:16	Reserved.
15:0	**MonLineSizeMin: smallest monitor-line size in bytes**.

CPUID Fn0000_0005_EBX Monitor/MWait

Bits	Description
31:16	Reserved.
15:0	**MonLineSizeMax: largest monitor-line size in bytes**.

CPUID Fn0000_0005_ECX Monitor/MWait

Bits	Description
31:2	Reserved.
1	**IBE: interrupt break-event**. Indicates MWAIT can use ECX bit 0 to allow interrupts to cause an exit from the monitor event pending state, even if EFLAGS.IF=0.
0	**EMX: enumerate MONITOR/MWAIT extensions**: Indicates enumeration MONITOR/MWAIT extensions are supported.

CPUID Fn0000_0005_EDX Monitor/MWait

Bits	Description
31:0	Reserved.

CPUID Fn0000_0006_EAX Thermal and Power Management

This function contains the feature identifiers for the digital thermal sensor and power management parameters.

Bits	Description
31:0	Reserved.

CPUID Fn0000_0006_EBX Thermal and Power Management

See CPUID Fn0000_0006_EAX.

Bits	Description
31:0	Reserved.

CPUID Fn0000_0006_ECX Thermal and Power Management

See CPUID Fn0000_0006_EAX.

Bits	Description
31:1	Reserved.
0	**EffFreq: effective frequency interface**. Indicates presence of MSR0000_00E7 (MPERF) and MSR0000_00E8 (APERF). See “Effective Frequency” in APMU.

CPUID Fn0000_0006_EDX Thermal and Power Management

See CPUID Fn0000_0006_EAX.

Bits	Description
31:0	Reserved.

CPUID Fn0000_0007_EAX_x0 Structured Extended Feature Identifiers (ECX=0)

Bits	Description
31:0	Reserved.

CPUID Fn0000_0007_EBX_x0 Structured Extended Feature Identifiers (ECX=0)

Bits	Description
31:4	Reserved.
3	**BMI: bit manipulation instruction support**.
2:0	Reserved.

CPUID Fn0000_0007_ECX_x0 Structured Extended Feature Identifiers (ECX=0)

Bits	Description
31:0	Reserved.

CPUID Fn0000_0007_EDX_x0 Structured Extended Feature Identifiers (ECX=0)

Bits	Description
31:0	Reserved.

CPUID Fn0000_000[C:8] Reserved

Bits	Description
31:0	Reserved.

CPUID Fn0000_000D_EAX_x0 Processor Extended State Enumeration (ECX=0)

Bits	Description
31:0	**XFeatureSupportedMask[31:0]**. Reports the valid bit positions for the lower 32 bits of the XFeatureEnabledMask register. 1=Valid. 0=Reserved.

CPUID Fn0000_000D_EBX_x0 Processor Extended State Enumeration (ECX=0)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	**XFeatureEnabledSizeMax**. Size in bytes of XSAVE/XRSTOR area for the currently enabled features in XCR0. Value: 512 + 64 + (IF (XCR0[AVX] | XCR0[LWP]) THEN 256 ELSE 0 ENDIF) + (IF XCR0[LWP]) THEN 128 ELSE 0 ENDIF). The components of this sum are described as follows: • 512: FPU/SSE save area (needed even if XCR0[SSE]=0) • 64: Header size (always needed). • Size of YMM area if YMM enabled OR if LWP enabled. • Size of LWP area if LWP enabled. See LWP Spec.

CPUID Fn0000_000D_ECX_x0 Processor Extended State Enumeration (ECX=0)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	**XFeatureSupportedSizeMax**. Size in bytes of XSAVE/XRSTOR area for all features that the core supports. See XFeatureEnabledSizeMax.

CPUID Fn0000_000D_EDX_x0 Processor Extended State Enumeration (ECX=0)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	**XFeatureSupportedMask[63:32]**. Reports the valid bit positions for the upper 32 bits of the XFeatureEnabledMask register. 1=Valid. 0=Reserved.

CPUID Fn0000_000D_EAX_x2 Processor Extended State Enumeration (ECX=2)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	**YmmSaveStateSize: YMM save state byte size**. The save state area size in bytes for The YMM registers.

CPUID Fn0000_000D_EBX_x2 Processor Extended State Enumeration (ECX=2)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	**YmmSaveStateOffset: YMM save state byte offset**. The save state area offset in bytes for The YMM registers.

CPUID Fn0000_000D_ECX_x2 Processor Extended State Enumeration (ECX=2)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	Reserved.

CPUID Fn0000_000D_EDX_x2 Processor Extended State Enumeration (ECX=2)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	Reserved.

For CPUID Fn0000_000D, if ECX>2 and ECX<62 then EAX/EBX/ECX/EDX will return 0.

CPUID Fn0000_000D_EAX_x3E Processor Extended State Enumeration (ECX=62)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	**LwpSaveStateSize: LWP save state byte size**. The save state area size in bytes for LWP. See LWP Spec.

CPUID Fn0000_000D_EBX_x3E Processor Extended State Enumeration (ECX=62)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	**LwpSaveStateOffset: LWP save state byte offset**. The save state area offset in bytes for LWP. See LWP Spec.

CPUID Fn0000_000D_ECX_x3E Processor Extended State Enumeration (ECX=62)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	Reserved.

CPUID Fn0000_000D_EDX_x3E Processor Extended State Enumeration (ECX=62)

See CPUID Fn0000_000D_EAX_x0.

Bits	Description
31:0	Reserved.

For CPUID Fn0000_000D, if ECX>62 then EAX/EBX/ECX/EDX will return 0.

CPUID Fn4000_00[FF:00] Reserved

Reserved for use by hypervisor.

Bits	Description
31:0	Reserved.

CPUID Fn8000_0000_EAX Largest Extended Function Number

Bits	Description
31:0	**LFuncExt: largest extended function**. The largest CPUID extended function input value supported by the processor implementation.

CPUID Fn8000_0000_E[D,C,B]X Processor Vendor

CPUID Fn0000_0000_E[D,C,B]X and CPUID Fn8000_0000_E[D,C,B]X return the same value.

Table 3: CPUID Fn8000_0000_E[D,C,B]X value

Register	Value	Description
CPUID Fn8000_0000_EBX	6874_7541h	The ASCII characters “h t u A”.
CPUID Fn8000_0000_ECX	444D_4163h	The ASCII characters “D M A c”.
CPUID Fn8000_0000_EDX	6974_6E65h	The ASCII characters “i t n e”.

Bits	Description
31:0	**Vendor**. The 12 8-bit ASCII character codes to create the string “AuthenticAMD”.

CPUID Fn8000_0001_EAX AMD Family, Model, Stepping

Bits	Description
31:0	See: CPUID Fn0000_0001_EAX.

CPUID Fn8000_0001_EBX BrandId Identifier

This function returns the extended brand ID field.

Bits	Description
31:28	**PkgType: package type**. If (Family[7:0] >= 10h) then the definition of PkgType is contained in the processor BKDG. If (Family[7:0]<10h) then the definition of PkgType is reserved.
27:16	Reserved.
15:0	**BrandId: brand ID**. This field, in conjunction with CPUID Fn0000_0001_EBX[8BitBrandId], is used by BIOS to generate the processor name string. See your processor revision guide for how to program the processor name string.

CPUID Fn8000_0001_ECX Feature Identifiers

This function contains the following miscellaneous feature identifiers.

Bits	Description
31:23	Reserved.
22	**TopologyExtensions: topology extensions support**. Indicates support for CPUID Fn8000_001D_EAX_x[N:0]-CPUID Fn8000_001E_EDX.
21	**TBM: trailing bit manipulation instruction support**.
20	Reserved.
19	**NodeId**. Indicates support for MSRC001_100C[NodeId, NodesPerProcessor].
18	Reserved.
17	Reserved.
16	**FMA4: 4-operand FMA instruction support**. See APM6.
15	**LWP: lightweight profiling support**. See LWP Spec.
14	Reserved.
13	**WDT: watchdog timer support**. See APM2 and APM3.
12	**SKINIT: SKINIT and STGI are supported, independent of the value of MSRC000_0080[SVME]**. See APM2 and APM3.
11	**XOP: extended operation support**. See APM6.
10	**IBS: instruction based sampling**. See “Guide to Instruction Based Sampling on AMD Family 10h Processors” in SWOG10.
9	**OSVW: OS visible workaround**. Indicates OS-visible workaround support. See “OS Visible Workaround (OSVW) Information” in APM2.
8	**3DNowPrefetch: PREFETCH and PREFETCHW instruction support**. See “PREFETCH” and “PREFETCHW” in APM3.
7	**MisAlignSse: misaligned SSE mode**. See “Misaligned Access Support Added for SSE Instructions” in APM1.
6	**SSE4A: EXTRQ, INSERTQ, MOVNTSS, and MOVNTSD instruction support**. See “EXTRQ”, “INSERTQ”, “MOVNTSS”, and “MOVNTSD” in APM4.
5	**ABM: advanced bit manipulation**. LZCNT instruction support. See “LZCNT” in APM3.
4	**AltMovCr8: LOCK MOV CR0 means MOV CR8**. See “MOV(CRn)” in APM3.
3	**ExtApicSpace: extended APIC space**. This bit indicates the presence of extended APIC register space starting at offset 400h from the “APIC Base Address Register,” as specified in the BKDG.
2	**SVM: secure virtual machine**. See “Secure Virtual Machine” in APM2.
1	**CmpLegacy: core multi-processing legacy mode**. See “Legacy Method” on page 36.
0	**LahfSahf: LAHF and SAHF instruction support in 64-bit mode**. See “LAHF” and “SAHF” in APM3.

CPUID Fn8000_0001_EDX Feature Identifiers

This function contains the following miscellaneous feature identifiers.

Bits	Description
31	**3DNow: 3DNow!™ instructions**. See Appendix D “Instruction Subsets and CPUID Feature Sets” in APM3.
30	**3DNowExt: AMD extensions to 3DNow! instructions**. See Appendix D “Instruction Subsets and CPUID Feature Sets” in APM3.
29	**LM: long mode**. See “Processor Initialization and Long-Mode Activation” in APM2.
28	Reserved.
27	**RDTSCP: RDTSCP instruction**. See “RDTSCP” in APM3.
26	**Page1GB: 1-GB large page support**. See “1-GB Paging Support” in APM2.
25	**FFXSR: FXSAVE and FXRSTOR instruction optimizations**. See “FXSAVE” and “FXRSTOR” in APM4.
24	**FXSR: FXSAVE and FXRSTOR instructions**. Same as CPUID Fn0000_0001_EDX[FXSR].
23	**MMX: MMX™ instructions**. Same as CPUID Fn0000_0001_EDX[MMX].
22	**MmxExt: AMD extensions to MMX instructions**. See Appendix D “Instruction Subsets and CPUID Feature Sets” in APM3 and “128-Bit Media and Scientific Programming” in APM1.
21	Reserved.
20	**NX: no-execute page protection**. See “Page Translation and Protection” in APM2.
19:18	Reserved.
17	**PSE36: page-size extensions**. Same as CPUID Fn0000_0001_EDX[PSE36].
16	**PAT: page attribute table**. Same as CPUID Fn0000_0001_EDX[PAT].
15	**CMOV: conditional move instructions**. Same as CPUID Fn0000_0001_EDX[CMOV].
14	**MCA: machine check architecture**. Same as CPUID Fn0000_0001_EDX[MCA].
13	**PGE: page global extension**. Same as CPUID Fn0000_0001_EDX[PGE].
12	**MTRR: memory-type range registers**. Same as CPUID Fn0000_0001_EDX[MTRR].
11	**SysCallSysRet: SYSCALL and SYSRET instructions**. See “SYSCALL” and “SYSRET” in APM3.
10	Reserved.
9	**APIC: advanced programmable interrupt controller**. Same as CPUID Fn0000_0001_EDX[APIC].
8	**CMPXCHG8B: CMPXCHG8B instruction**. Same as CPUID Fn0000_0001_EDX[CMPXCHG8B].
7	**MCE: machine check exception**. Same as CPUID Fn0000_0001_EDX[MCE].
6	**PAE: physical-address extensions**. Same as CPUID Fn0000_0001_EDX[PAE].
5	**MSR: AMD model-specific registers**. Same as CPUID Fn0000_0001_EDX[MSR].
4	**TSC: time stamp counter**. Same as CPUID Fn0000_0001_EDX[TSC].

Bits	Description
3	**PSE: page-size extensions**. Same as CPUID Fn0000_0001_EDX[PSE].
2	**DE: debugging extensions**. Same as CPUID Fn0000_0001_EDX[DE].
1	**VME: virtual-mode enhancements**. Same as CPUID Fn0000_0001_EDX[VME].
0	**FPU: x87 floating-point unit on-chip**. Same as CPUID Fn0000_0001_EDX[FPU].

CPUID Fn8000_000[4:2]_E[D,C,B,A]X Processor Name String Identifier

The three extended functions from Fn8000_0002 to Fn8000_0004 are initialized to and return a null terminated ASCII string up to 48 characters in length corresponding to the processor name. (The 48 character maximum includes the null character.) The 48 character sequence is ordered first to last as follows:

Fn8000_0002[EAX[7:0],..., EAX[31:24], EBX[7:0],..., EBX[31:24], ECX[7:0],..., ECX[31:24],EDX[7:0],..., EDX[31:24]],
Fn8000_0003[EAX[7:0],..., EAX[31:24], EBX[7:0],..., EBX[31:24], ECX[7:0],..., ECX[31:24], EDX[7:0],..., EDX[31:24]],
Fn8000_0004[EAX[7:0],..., EAX[31:24], EBX[7:0],..., EBX[31:24], ECX[7:0],..., ECX[31:24], EDX[7:0],..., EDX[31:24]].

The processor name string must be programmed by the BIOS during system initialization. See your processor revision guide for information about how to program and display the processor name string.

Bits	Description
31:0	**ProcName: processor name**.

CPUID Fn8000_0005_EAX L1 Cache and TLB Identifiers

This function contains the processor's first level cache and TLB characteristics for each core.

The *associativity* fields are encoded as follows:

00h: Reserved.
01h: Direct mapped.
02h-FEh: Associativity. (e.g., 04h = 4-way associative.)
FFh: Fully associative.

Bits	Description
31:24	**L1DTlb2and4MAssoc: data TLB associativity for 2 MB and 4 MB pages**. Data TLB associativity for 2-MB and 4-MB pages. See: CPUID Fn8000_0005_EDX[L1IcAssoc].
23:16	**L1DTlb2and4MSize: data TLB number of entries for 2 MB and 4 MB pages**. Data TLB number of entries for 2-MB and 4-MB pages. The value returned is for the number of entries available for the 2-MB page size; 4-MB pages require two 2-MB entries, so the number of entries available for the 4-MB page size is one-half the returned value.

15:8	**L1ITlb2and4MAssoc: instruction TLB associativity for 2 MB and 4 MB pages**. Instruction TLB associativity for 2-MB and 4-MB pages. See: CPUID Fn8000_0005_EDX[L1IcAssoc].
7:0	**L1ITlb2and4MSize: instruction TLB number of entries for 2 MB and 4 MB pages**. Instruction TLB number of entries for 2-MB and 4-MB pages. The value returned is for the number of entries available for the 2-MB page size; 4-MB pages require two 2-MB entries, so the number of entries available for the 4-MB page size is one-half the returned value.

CPUID Fn8000_0005_EBX L1 Cache and TLB Identifiers

This provides the processor's first level cache and TLB characteristics for each core.

Bits	Description
31:24	**L1DTlb4KAssoc: data TLB associativity for 4 KB pages**. Data TLB associativity for 4 KB pages. See: CPUID Fn8000_0005_EDX[L1IcAssoc].
23:16	**L1DTlb4KSize: data TLB number of entries for 4 KB pages**. Data TLB number of entries for 4 KB pages.
15:8	**L1ITlb4KAssoc: instruction TLB associativity for 4 KB pages**. Instruction TLB associativity for 4 KB pages.See: CPUID Fn8000_0005_EDX[L1IcAssoc].
7:0	**L1ITlb4KSize: instruction TLB number of entries for 4 KB pages**. Instruction TLB number of entries for 4 KB pages.

CPUID Fn8000_0005_ECX L1 Cache and TLB Identifiers

This provides the processor's first level cache and TLB characteristics for each core.

Bits	Description
31:24	**L1DcSize: L1 data cache size in KB**. L1 data cache size in KB.
23:16	**L1DcAssoc: L1 data cache associativity**. L1 data cache associativity. See: CPUID Fn8000_0005_EDX[L1IcAssoc].
15:8	**L1DcLinesPerTag: L1 data cache lines per tag**. L1 data cache lines per tag.
7:0	**L1DcLineSize: L1 data cache line size in bytes**. L1 data cache line size in bytes.

CPUID Fn8000_0005_EDX L1 Cache and TLB Identifiers

This provides the processor's first level cache and TLB characteristics for each core.

Bits	Description
31:24	**L1IcSize: L1 instruction cache size KB**. L1 instruction cache size KB.
23:16	**L1IcAssoc: L1 instruction cache associativity**. L1 instruction cache associativity. Bits — Description 00h — Reserved 01h — 1 way (direct mapped) FEh-02h — Specifies the associativity; e.g., 04h would indicate a 4-way associativity. FFh — Fully associative

15:8	**L1IcLinesPerTag: L1 instruction cache lines per tag**. L1 instruction cache lines per tag.
7:0	**L1IcLineSize: L1 instruction cache line size in bytes**. L1 instruction cache line size in bytes.

CPUID Fn8000_0006_EAX L2 TLB Identifiers

This function contains the processor's second level cache and TLB characteristics for each core. The EDX register contains the processor's third level cache characteristics that are shared by all cores of the processor.

The *associativity* fields are encoded as follows:

Table 4: L2/L3 Cache and TLB Associativity Field Definition

Associativity [3:0]	Definition
0h	L2/L3 cache or TLB is disabled.
1h	Direct mapped.
2h	2-way associative.
4h	4-way associative.
6h	8-way associative.
8h	16-way associative.
Ah	32-way associative.
Bh	48-way associative.
Ch	64-way associative.
Dh	96-way associative.
Eh	128-way associative.
Fh	Fully associative.
All other encodings are reserved.	

Bits	Description
31:28	**L2DTlb2and4MAssoc: L2 data TLB associativity for 2 MB and 4 MB pages**. L2 data TLB associativity for 2-MB and 4-MB pages. See Table 4.
27:16	**L2DTlb2and4MSize: L2 data TLB number of entries for 2 MB and 4 MB pages**. L2 data TLB number of entries for 2-MB and 4-MB pages. The value returned is for the number of entries available for the 2 MB page size; 4 MB pages require two 2 MB entries, so the number of entries available for the 4 MB page size is one-half the returned value.
15:12	**L2ITlb2and4MAssoc: L2 instruction TLB associativity for 2 MB and 4 MB pages**. L2 instruction TLB associativity for 2-MB and 4-MB pages. See Table 4.
11:0	**L2ITlb2and4MSize: L2 instruction TLB number of entries for 2 MB and 4 MB pages**. L2 instruction TLB number of entries for 2-MB and 4-MB pages. The value returned is for the number of entries available for the 2 MB page size; 4 MB pages require two 2 MB entries, so the number of entries available for the 4 MB page size is one-half the returned value.

CPUID Fn8000_0006_EBX L2 TLB Identifiers

See CPUID Fn8000_0006_EAX.

Bits	Description
31:28	**L2DTlb4KAssoc: L2 data TLB associativity for 4 KB pages**. L2 data TLB associativity for 4-KB pages. See Table 4.
27:16	**L2DTlb4KSize: L2 data TLB number of entries for 4 KB pages**. L2 data TLB number of entries for 4-KB pages.
15:12	**L2ITlb4KAssoc: L2 instruction TLB associativity for 4 KB pages**. L2 instruction TLB associativity for 4-KB pages. See Table 4.
11:0	**L2ITlb4KSize: L2 instruction TLB number of entries for 4 KB pages**. L2 instruction TLB number of entries for 4-KB pages.

CPUID Fn8000_0006_ECX L2 Cache Identifiers

See CPUID Fn8000_0006_EAX.

<table>
<tr><th>Bits</th><th>Description</th></tr>
<tr><td>31:16</td><td>L2Size: L2 cache size in KB.</td></tr>
<tr><td>15:12</td><td>L2Assoc: L2 cache associativity.
<table>
<tr><th>Bits</th><th>Description</th><th>Bits</th><th>Description</th></tr>
<tr><td>0000b</td><td>Disabled.</td><td>1000b</td><td>16 ways</td></tr>
<tr><td>0001b</td><td>1 way (direct mapped)</td><td>1001b</td><td>Reserved.</td></tr>
<tr><td>0010b</td><td>2 ways</td><td>1010b</td><td>32 ways</td></tr>
<tr><td>0011b</td><td>Reserved.</td><td>1011b</td><td>48 ways</td></tr>
<tr><td>0100b</td><td>4 ways</td><td>1100b</td><td>64 ways</td></tr>
<tr><td>0101b</td><td>Reserved.</td><td>1101b</td><td>96 ways</td></tr>
<tr><td>0110b</td><td>8 ways</td><td>1110b</td><td>128 ways</td></tr>
<tr><td>0111b</td><td>Reserved</td><td>1111b</td><td>Fully associative</td></tr>
</table></td></tr>
<tr><td>11:8</td><td>L2LinesPerTag: L2 cache lines per tag.</td></tr>
<tr><td>7:0</td><td>L2LineSize: L2 cache line size in bytes.</td></tr>
</table>

CPUID Fn8000_0006_EDX L3 Cache Identifiers

This provides the processor's third level cache characteristics shared by all cores.

Bits	Description
31:18	**L3Size: L3 cache size**. Specifies the L3 cache size is within the following range: (L3Size[31:18] * 512KB) ≤ L3 cache size < ((L3Size[31:18]+1) * 512KB).
17:16	Reserved.
15:12	**L3Assoc: L3 cache associativity**. L3 cache associativity. See Table 4.

11:8	**L3LinesPerTag: L3 cache lines per tag**.
7:0	**L3LineSize: L3 cache line size in bytes**.

CPUID Fn8000_0007_E[C,B,A]X Advanced Power Management Information

Bits	Description
31:0	Reserved.

CPUID Fn8000_0007_EDX Advanced Power Management Information

This function provides advanced power management feature identifiers. Refer to the processor BKDG for a detailed description of the definition of each power management feature and whether that feature is supported.

Bits	Description
31:11	Reserved.
10	**EffFreqRO: read-only effective frequency interface**. 1=Indicates presence of MSRC000_00E7 [Read-Only Max Performance Frequency Clock Count (MPerfReadOnly)] and MSRC000_00E8 [Read-Only Actual Performance Frequency Clock Count (APerfReadOnly)].
9	**CPB: core performance boost**.
8	**TscInvariant: TSC invariant**. The TSC rate is ensured to be invariant across all P-States, C-States, and stop grant transitions (such as STPCLK Throttling); therefore the TSC is suitable for use as a source of time. 0 = No such guarantee is made and software should avoid attempting to use the TSC as a source of time.
7	**HwPstate: hardware P-state control**. MSRC001_0061 [P-state Current Limit], MSRC001_0062 [P-state Control] and MSRC001_0063 [P-state Status] exist.
6	**100MHzSteps: 100 MHz multiplier Control**.
5	Reserved.
4	**TM: hardware thermal control (HTC)**.
3	**TTP: THERMTRIP**.
2	**VID: Voltage ID control**. Function replaced by HwPstate.
1	**FID: Frequency ID control**. Function replaced by HwPstate.
0	**TS: Temperature sensor**.

CPUID Fn8000_0008_EAX Long Mode Address Size Identifiers

This function returns information about the maximum physical and linear address width (in bits) supported by the processor. The width reported is the maximum supported in any mode. For long mode capable processors, the size reported is independent of whether long mode is enabled. See “Processor Initialization and Long-Mode Activation” in APM2.

Bits	Description
31:24	Reserved.
23:16	**GuestPhysAddrSize: maximum guest physical byte address size in bits**. This number applies only to guests using nested paging. When this field is zero, refer to the PhysAddrSize field for the maximum guest physical address size. See “Secure Virtual Machine” in APM2.
15:8	**LinAddrSize: Maximum linear byte address size in bits**.
7:0	**PhysAddrSize: Maximum physical byte address size in bits**. When GuestPhysAddrSize is zero, this field also indicates the maximum guest physical address size.

CPUID Fn8000_0008_EBX Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_0008_ECX APIC ID Size and Core Count

This provides information about the number of cores supported by the processor.

Bits	Description
31:16	Reserved.

15:12	**ApicIdCoreIdSize: APIC ID size**. The number of bits in the initial APIC20[ApicId] value that indicate core ID within a processor. A zero value indicates that legacy methods must be used to derive the maximum number of cores. The size of this field determines the maximum number of cores (MNC) that the processor could theoretically support, not the actual number of cores that are actually implemented or enabled on the processor, as indicated by CPUID Fn8000_0008_ECX[NC].
	
	if (ApicIdCoreIdSize[3:0] == 0){
	// Used by legacy dual-core/single-core processors
	MNC = CPUID Fn8000_0008_ECX[NC] + 1;
	} else {
	// use ApicIdCoreIdSize[3:0] field
	MNC = (2 ^ ApicIdCoreIdSize[3:0]);
	}

11:8	Reserved.
7:0	**NC: number of physical cores - 1**. The number of cores in the processor is NC+1 (e.g., if NC=0, then there is one core). See “Legacy Method” on page 36.

CPUID Fn8000_0008_EDX Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_0009 Reserved

This function is reserved.

Bits	Description
31:0	Reserved.

CPUID Fn8000_000A_EAX SVM Revision

This provides SVM revision. If CPUID Fn8000_0001_ECX[SVM]=0 then CPUID Fn8000_000A_EAX is reserved.

Bits	Description
31:8	Reserved.
7:0	**SvmRev: SVM revision**.

CPUID Fn8000_000A_EBX SVM Revision and Feature Identification

This provides SVM revision and feature information. If CPUID Fn8000_0001_ECX[SVM]=0 then CPUID Fn8000_000A_EBX is reserved.

Bits	Description
31:0	**NASID: number of address space identifiers (ASID)**.

CPUID Fn8000_000A_ECX Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_000A_EDX SVM Feature Identification

This provides SVM feature information. See APM2 and “Changes to the VMCB Control Area” in APMU. If CPUID Fn8000_0001_ECX[SVM]=0 then CPUID Fn8000_000A_EDX is reserved.

Bits	Description
31:13	Reserved.
12	**PauseFilterThreshold: PAUSE filter threshold**. Indicates support for the PAUSE filter cycle count threshold.
11	Reserved.
10	**PauseFilter: pause intercept filter**. Indicates support for the pause intercept filter. See “Instruction Intercepts” in APM2.
9	Reserved.
8	Reserved.

Bits	Description
7	**DecodeAssists: decode assists**. Indicates support for the decode assists.
6	**FlushByAsid: flush by ASID**. Indicates that TLB flush events, including CR3 writes and CR4.PGE toggles, flush only the current ASID's TLB entries. Also indicates support for the extended VMCB TLB_Control. See “Flush by ASID” in APMU TBD.
5	**VmcbClean: VMCB clean bits**. Indicates support for VMCB clean bits. See “VMCB Clean Bits” in APMU.
4	**TscRateMsr: MSR based TSC rate control**. Indicates support for MSR TSC ratio MSRC000_0104. See “TSC Ratio” in APMU.
3	**NRIPS: NRIP save**. Indicates support for NRIP save on #VMEXIT. See “Save Next Sequential Instruction Pointer on #VMEXIT” in APM2.
2	**SVML: SVM lock**. Indicates support for SVM-Lock. See “Locking the SVM enable bit” in APM2.
1	**LbrVirt: LBR virtualization**. Support is provided for VMRUN to save and for VMEXIT to restore the following five MSRs: MSR0000_01D9 Debug Control Register (DBG_CTL_MSR), MSR0000_01DB Last Branch From IP Register (BR_FROM), MSR0000_01DC Last Branch To IP Register (BR_TO), MSR0000_01DD Last Exception From IP Register, MSR0000_01DE Last Exception To IP Register. See “Secure Virtual Machine” in APM2.
0	**NP: nested paging**. See “Secure Virtual Machine” in APM2.

CPUID Fn8000_00[18:0B] Reserved

These functions are reserved.

Bits	Description
31:0	Reserved.

CPUID Fn8000_0019_EAX TLB 1GB Page Identifiers

This function provides 1 GB paging information.

Bits	Description
31:28	**L1DTlb1GAssoc: L1 data TLB associativity for 1 GB pages**. See Table 4.
27:16	**L1DTlb1GSize: L1 data TLB number of entries for 1 GB pages**.
15:12	**L1ITlb1GAssoc: L1 instruction TLB associativity for 1 GB pages**. See Table 4.
11:0	**L1ITlb1GSize: L1 instruction TLB number of entries for 1 GB pages**.

CPUID Fn8000_0019_EBX TLB 1GB Page Identifiers

This provides 1 GB paging information..

Bits	Description
31:28	**L2DTlb1GAssoc: L2 data TLB associativity for 1 GB pages**. See Table 4.
27:16	**L2DTlb1GSize: L2 data TLB number of entries for 1 GB pages**.
15:12	**L2ITlb1GAssoc: L2 instruction TLB associativity for 1 GB pages**. See Table 4.
11:0	**L2ITlb1GSize: L2 instruction TLB number of entries for 1 GB pages**.

CPUID Fn8000_0019_E[D,C]X Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_001A_EAX Performance Optimization Identifiers

This function returns performance related information. For more details on how to use these bits to optimize software, see the optimization guide for your processor implementation.

Bits	Description
31:2	Reserved.
1	**MOVU**: MOVU SSE (multimedia) instructions are more efficient and should be preferred to SSE (multimedia) MOVL/MOVH. MOVUPS is more efficient than MOVLPS/MOVHPS. MOVUPD is more efficient than MOVLPD/MOVHPD.
0	**FP128**: 128-bit SSE (multimedia) instructions are executed with full-width internal operations and pipelines rather than decomposing them into internal 64-bit suboperations. This may impact how software performs instruction selection and scheduling.

CPUID Fn8000_001A_E[D,C,B]X Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_001B_EAX Instruction Based Sampling Identifiers

This function returns IBS feature information. If CPUID Fn8000_0001_ECX[IBS] = 0 then CPUID Fn8000_001B_EAX is reserved. See Appendix G, “Guide to Instruction-Based Sampling on AMD Family 10h Processors” in SWOG10.

Bits	Description
31:8	Reserved.
7	**RipInvalidChk: invalid RIP indication supported**.
6	**OpCntExt: IbsOpCurCnt and IbsOpMaxCnt extend by 7 bits**.

Bits	Description
5	**BrnTrgt: branch target address reporting supported**.
4	**OpCnt: op counting mode supported**.
3	**RdWrOpCnt: read write of op counter supported**.
2	**OpSam: IBS execution sampling supported**.
1	**FetchSam: IBS fetch sampling supported**.
0	**IBSFFV: IBS feature flags valid**.

CPUID Fn8000_001B_E[D,C,B]X Reserved

Bits	Description
31:0	Reserved.

CPUID Fn8000_001C_EAX Lightweight Profiling Capabilities 0

This function returns LWP feature information; see "Detecting LWP" in the LWP Spec. If CPUID Fn8000_0001_ECX[LWP] = 0 then CPUID Fn8000_001C_E[D,C,B,A]X is reserved.

Bits	Description
31	**LwpInt: interrupt on threshold overflow available**. 1=Interrupt on threshold overflow is available.
30:7	Reserved.
6	**LwpRNH: core reference clocks not halted event available**. 1=Core reference clocks not halted event is available.
5	**LwpCNH: core clocks not halted event available**. 1=Core clocks not halted event is available.
4	**LwpDME: DC miss event available**. 1=DC miss event is available.
3	**LwpBRE: branch retired event available**. 1=Branch retired event is available.
2	**LwpIRE: instructions retired event available**. 1=Instructions retired event is available.
1	**LwpVAL: LWPVAL instruction available**. 1=LWPVAL instruction is available.
0	**LwpAvail: LWP available**. 1=LWP is available.

CPUID Fn8000_001C_EBX Lightweight Profiling Capabilities 0

See CPUID Fn8000_001C_EAX. See LWP Spec.

Bits	Description
31:24	**LwpEventOffset: offset to the EventInterval1 field**. Offset from the start of the LWPCB to the EventInterval1 field.
23:16	**LwpMaxEvents: maximum EventId**. Maximum EventId value that is supported.
15:8	**LwpEventSize: event record size**. Size in bytes of an event record in the LWP event ring buffer.
7:0	**LwpCbSize: control block size**. Size in bytes of the LWPCB.

CPUID Fn8000_001C_ECX Lightweight Profiling Capabilities 0

See CPUID Fn8000_001C_EAX. See LWP Spec.

Bits	Description
31	**LwpCacheLatency: cache latency filtering supported**. 1=Cache-related events can be filtered by latency.
30	**LwpCacheLevels: cache level filtering supported**. 1=Cache-related events can be filtered by the cache level that returned the data.
29	**LwpIpFiltering: IP filtering supported**. 1=IP filtering is supported.
28	**LwpBranchPrediction: branch prediction filtering supported**. 1=Branches Retired events can be filtered based on whether the branch was predicted properly.
27:24	Reserved.
23:16	**LwpMinBufferSize: event ring buffer size**. Minimum size of the LWP event ring buffer, in units of 32 event records.
15:9	**LwpVersion: version**. Version of LWP implementation.
8:6	**LwpLatencyRnd: amount cache latency is rounded**. The amount by which cache latency is rounded.
5	**LwpDataAddress: data cache miss address valid**. 1=Address is valid for cache miss event records.
4:0	**LwpLatencyMax: latency counter bit size**. Size in bits of the cache latency counters.

CPUID Fn8000_001C_EDX Lightweight Profiling Capabilities 0

See CPUID Fn8000_001C_EAX. See LWP Spec.

Bits	Description
31	**LwpInt: interrupt on threshold overflow supported**. 1=Interrupt on threshold overflow is supported.
30:7	Reserved.
6	**LwpRNH: core reference clocks not halted event supported**. 1=Core reference clocks not halted event is supported.
5	**LwpCNH: core clocks not halted event supported**. 1=Core clocks not halted event is supported.
4	**LwpDME: DC miss event supported**. 1=DC miss event is supported.
3	**LwpBRE: branch retired event supported**. 1=Branch retired event is supported.
2	**LwpIRE: instructions retired event supported**. 1=Instructions retired event is supported.
1	**LwpVAL: LWPVAL instruction supported**. 1=LWPVAL instruction is supported.
0	**LwpAvail: lightweight profiling supported**. 1=Lightweight profiling is supported.

CPUID Fn8000_001D_EAX_x[N:0] Cache Properties

CPUID Fn8000_001D_E[D,C,B,A]X reports cache topology information for the cache indicated by

Fn8000_001D_EAX_xN[CacheType, CacheLevel]. If (CPUID Fn8000_0001_ECX[TopologyExtensions]==0) then CPUID Fn8000_001D_E[D,C,B,A]X is reserved. The value of N for ECX in "_x[N:0]" is the last defined cache level such that CPUID Fn8000_001D_EAX_xN[CacheType]==00h, which is Null.

Bits	Description
31:26	Reserved.
25:14	**NumSharingCache: number of cores sharing cache**. The number of cores sharing this cache is NumSharingCache+1.
13:10	Reserved.
9	**FullyAssociative: fully associative cache**. 1=Cache is fully associative.
8	**SelfInitialization: cache is self-initializing**. 1=Cache is self initializing; cache does not need software initialization.
7:5	**CacheLevel: cache level.** Identifies the cache level. Bits — Description 000b — Reserved. 001b — Level 1 010b — Level 2 011b — Level 3 111b-100b — Reserved.
4:0	**CacheType: cache type**. Identifies the type of cache. Bits — Description 00h — Null; no more caches. 01h — Data cache 02h — Instruction cache 03h — Unified cache 1Fh-04h — Reserved.

CPUID Fn8000_001D_EBX_x[N:0] Cache Properties

See CPUID Fn8000_001D_EAX_x[N:0].

Bits	Description
31:22	**CacheNumWays: cache number of ways**. Cache number of ways is CacheNumWays+1.
21:12	**CachePhysPartitions: cache physical line partitions**. Cache partitions is CachePhysPartitions+1.
11:0	**CacheLineSize: cache line size in bytes**. Value: 03Fh. Cache line size in bytes is CacheLineSize+1.

CPUID Fn8000_001D_ECX_x[N:0] Cache Properties

See CPUID Fn8000_001D_EAX_x[N:0].

Bits	Description
31:0	**CacheNumSets: cache number of sets**. Value: 0000_003Fh. Cache number of sets is CacheNumSets+1.

CPUID Fn8000_001D_EDX_x[N:0] Cache Properties

See CPUID Fn8000_001D_EAX_x[N:0].

Bits	Description
31:2	Reserved.
1	**CacheInclusive: cache inclusive**. 0=Cache is not inclusive of lower cache levels, as indicated by starthere1. 1=Cache is inclusive of lower cache levels.
0	**WBINVD: Write-Back Invalidate/Invalidate**. 0=WBINVD/INVD invalidates all lower level caches of non-originating cores sharing this cache. 1=WBINVD/INVD not guaranteed to invalidate all lower level caches of non-originating cores sharing this cache.

CPUID Fn8000_001E_EAX Extended APIC ID

If CPUID Fn8000_0001_ECX[TopologyExtensions]==0 then CPUID Fn8000_001E_E[D,C,B,A]X is reserved.

Bits	Description
31:0	**ExtendedApicId: extended APIC ID**.

CPUID Fn8000_001E_EBX Compute Unit Identifiers

See CPUID Fn8000_001E_EAX.

Bits	Description
31:10	Reserved.
9:8	**CoresPerComputeUnit: cores per compute unit**. The number of cores per compute unit is CoresPerComputeUnit+1.
7:0	**ComputeUnitId: compute unit ID**. Identifies the processor compute unit ID.

CPUID Fn8000_001E_ECX Node Identifiers

See CPUID Fn8000_001E_EAX.

Bits	Description
31:11	Reserved.
10:8	**NodesPerProcessor**. Specifies the number of nodes per processor. Bits — Description 000b — 1 node per processor 001b — 2 nodes per processor 111b-010b — Reserved
7:0	**NodeId**. Specifies the node ID.

CPUID Fn8000_001E_EDX Reserved

See CPUID Fn8000_001E_EAX.

Bits	Description
31:0	Reserved.

3 Multiple Core Calculation

Operating systems use one of two possible methods to calculate the number of cores per processor (NC), and the maximum number of cores per processor (MNC). The extended method is recommended, but a legacy method is also available for existing operating systems.

3.1 Legacy Method

The CPUID identification of total number of cores per processor (c) is derived from information returned by the following fields:

- CPUID Fn0000_0001_EBX[LogicalProcessorCount]
- CPUID Fn0000_0001_EDX[HTT] (Hyper-Threading Technology)
- CPUID Fn8000_0001_ECX[CmpLegacy]
- CPUID Fn8000_0008_ECX[NC] (number of cores - 1)

Table 5 defines LogicalProcessorCount, HTT, CmpLegacy, and NC as a function of the number of cores per processor (c).

When HTT = 0, LogicalProcessorCount is reserved and the processor contains one core.

When HTT = 1 and CmpLegacy = 1, LogicalProcessorCount represents the number of cores per processor (c).

Table 5: LogicalProcessorCount, CmpLegacy, HTT, and NC

Cores per Processor (c)	CmpLegacy	HTT	LogicalProcessorCount	NC
1	0	0	Reserved	0
2 or more	1	1	c	c-1

The use of CmpLegacy and LogicalProcessorCount for the determination of the number of cores is deprecated. Instead, use NC to determine the number of cores.

3.2 Extended Method (Recommended)

The CPUID identification of total number of cores per processor is derived from information returned by the CPUID Fn8000_0008_ECX[ApicIdCoreIdSize[3:0]]. This field indicates the number of least significant bits in the CPUID Fn0000_0001_EBX[LocalApicId] that indicates core ID within the processor. The size of this field determines the maximum number of cores (MNC) that the processor could theoretically support, not the actual number of cores that are actually implemented or enabled on the processor, as indicated by CPUID Fn8000_0008_ECX[NC].

A value of zero for ApicIdCoreIdSize[3:0] indicates that the legacy method (section 2.1) should be used to derive the maximum number of cores:

MNC = CPUID Fn8000_0008_ECX[NC] + 1.

For non-zero values of ApicIdCoreIdSize[3:0],

MNC = (2 ^ ApicIdCoreIdSize[3:0])

3.2.1 APIC Enumeration Requirements

System hardware and BIOS must ensure that the maximum number of cores per processor (MNC) exposed to the operating system across all cores and processors in the system is identical.

Local ApicId MNC rule: The ApicId of core j on processor node i must be enumerated/assigned as:

LocalApicId[proc=i, core=j] = (OFFSET_IDX + i) * MNC + j

Where "OFFSET_IDX" is an integer offset (0 to N) used to shift up the core LocalApicId values to allow room for IOAPIC devices. This assignment allows software to use a simple bitmask in addressing all the cores of a single processor. (The assignment also has the effect of reserving some IDs from use to ensure alignment of the ID of core 0 on each processor.)

For example, consider a 3-processor system where:

processor 0 has 4 cores
processor 1 has 1 core
processor 2 has 2 cores
there are 8 IOAPIC devices
cpuid.core_id_bits =2 for all cases, so MNC=4

The LocalApicId and IOAPIC ID spaces cannot be disjointed and must be enumerated in the same ID space in order to support legacy operating systems. Each core can support an 8-bit ApicId. But if each IOAPIC device supports only a 4-bit IOAPIC ID, then the problem can be solved by shifting the LocalApicId space to start at some integer multiple of MNC, such as offset 8 (MNC = 4; OFFSET_IDX=2):

LocalApicId[proc=0,core=0] = (2+0)*4 + 0 = 0x08
LocalApicId[proc=0,core=1] = (2+0)*4 + 1 = 0x09
LocalApicId[proc=0,core=2] = (2+0)*4 + 2 = 0x0A
LocalApicId[proc=0,core=3] = (2+0)*4 + 3 = 0x0B

LocalApicId[proc=1,core=0] = (2+1)*4 + 0 = 0x0C
LocalApicId 0xD to 0xF are reserved

LocalApicId[proc=2,core=0] = (2+2)*4 + 0 = 0x10
LocalApicId[proc=2,core=1] = (2+2)*4 + 1 = 0x11
LocalApicId 0x12 and 0x13 are reserved

It is recommended that BIOS use the following LocalApicId assignments for the broadest operating system support. Given N = (Number_Of_Processors * MNC) and M = Number_Of_IOAPICs:

- If (N+M) <16, assign the LocalApicIds for the cores first from 0 to N-1, and the IOAPIC IDs from N to N+(M-1).
- If (N+M) > =16, assign the IOAPIC IDs first from 0 to M-1, and the LocalApicIds for the cores from K to K+(N-1), where K is an integer multiple of MNC greater than M-1.