IA-32 architecture
CPUID

Before trying to rely upon CPUID, a program must properly detect and sometimes enable the instruction. In particular, the program must detect the presence of a 32bit IA-32 processor, which supports the EFLAGS register. Next, if it is a Cyrix or a NexGen processor, the CPUID instruction may have to be enabled. Then the program must try to toggle the ID bit in the EFLAGS register, to determine whether the instruction is supported or not. Note that the program may face one of the early Intel P5 processors: they do neither return a vendor ID string nor the maximum supported standard level, when level 0000_0000h is queried. Finally, notice that some chips support a partially programmable CPUID instruction -- thanks to those idiot programmers who hard-coded "GenuineIntel" all over the place...

---

standard level 0000_0000h			
input	EAX=0000_0000h	get maximum supported standard level and vendor ID string	
output	EAX=xxxx_xxxxh	maximum supported standard level [#1]	
	EBX-EDX-ECX	vendor ID string [#2]	
		`GenuineIntel`	Intel processor
		`UMC UMC UMC`	UMC processor
		`AuthenticAMD`	AMD processor
		`CyrixInstead`	Cyrix processor
		`NexGenDriven`	NexGen processor
		`CentaurHauls`	Centaur processor
		`RiseRiseRise`	Rise Technology processor
		`SiS SiS SiS`	SiS processor
		`GenuineTMx86`	Transmeta processor
		`Geode by NSC`	National Semiconductor processor
notes	description		
#1	According to [1] and [2] the pre-B0 step Intel P5 processors return EAX=0000_05xxh.		
#2	According to [1] and [2] the pre-B0 step Intel P5 processors don't return a vendor ID string.		

standard level 0000_0001h

input	EAX=0000_0001h	get processor type/family/model/stepping and feature flags		
output	EAX=xxxx_xxxxh	processor type/family/model/stepping		
		extended family	The extended processor family is encoded in bits 27..20.	
			00h	Intel P4 AMD K8
			01h	Intel Itanium 2 (IA-64)
		extended model	The extended processor model is encoded in bits 19..16.	
		type	The processor type is encoded in bit 13 and bit 12.	
			11b	reserved
			10b	secondary processor (for MP)
			01b	Overdrive processor
			00b	primary processor
		family	The family is encoded in bits 11..8.	
			4	most 80486s AMD 5x86 Cyrix 5x86
			5	Intel P5, P54C, P55C, P24T NexGen Nx586 Cyrix M1 Cyrix MediaGX NS Geode AMD K5, K6 Centaur C6, C2, C3 Rise mP6 SiS 55x Transmeta Crusoe TM3x00 and TM5x00
			6	Intel P6, P2, P3, PM AMD K7 Cyrix M2 VIA C3
			7	Intel Itanium (IA-64)
			F	refer to extended family (here: add)
		model	The model is encoded in bits 7..4.	

all	F	refer to extended model (here: concat)
Intel 80486	0	i80486DX-25/33
	1	i80486DX-50
	2	i80486SX
	3	i80486DX2
	4	i80486SL
	5	i80486SX2
	7	i80486DX2WB
	8	i80486DX4
	9	i80486DX4WB
UMC 80486	1	U5D
	2	U5S
AMD 80486	3	80486DX2
	7	80486DX2WB
	8	80486DX4
	9	80486DX4WB
	E	5x86
	F	5x86WB
Cyrix 5x86	9	5x86
Cyrix MediaGX	4	GX, GXm
Intel P5-core	0	P5 A-step
	1	P5
	2	P54C
	3	P24T Overdrive
	4	P55C
	7	P54C
	8	P55C (0.25µm)
NexGen Nx586	0	Nx586 or Nx586FPU (only later ones)
Cyrix M1	2	6x86
Cyrix M2	0	6x86MX
NS Geode	4	GX1, GXLV, GXm
	5	GX2
AMD K5	0	SSA5 (PR75, PR90, PR100)
	1	5k86 (PR120, PR133)
	2	5k86 (PR166)
	3	5k86 (PR200)

			AMD K6	6	K6 (0.30 µm)
				7	K6 (0.25 µm)
				8	K6-2
				9	K6-III
				D	K6-2+ or K6-III+ (0.18 µm)
			Centaur	4	C6
				8	C2
				9	C3
			VIA C3	5	Cyrix M2 core
				6	WinChip C5A core
				7	WinChip C5B core (if stepping = 0..7)
				7	WinChip C5C core (if stepping = 8..F)
				8	WinChip C5N core (if stepping = 0..7)
				9	WinChip C5XL core (if stepping = 0..7)
				9	WinChip C5P core (if stepping = 8..F)
				10	WinChip C5J core
			Rise	0	mP6 (0.25 µm)
				2	mP6 (0.18 µm)
			SiS	0	55x
			Transmeta	4	Crusoe TM3x00 and TM5x00
			Intel P6-core	0	P6 A-step
				1	P6
				3	P2 (0.28 µm)
				5	P2 (0.25 µm)
				6	P2 with on-die L2 cache
				7	P3 (0.25 µm)
				8	P3 (0.18 µm) with 256 KB on-die L2 cache
				9	PM (0.13 µm) with 512 KB or 1 MB on-die L2 cache
				A	P3 (0.18 µm) with 1 or 2 MB on-die

					L2 cache
				B	P3 (0.13 µm) with 256 or 512 KB on-die L2 cache
				D	PM (0.09 µm) with 512 KB or 2 MB on-die L2 cache
			AMD K7	1	Athlon (0.25 µm)
				2	Athlon (0.18 µm)
				3	Duron (SF core)
				4	Athlon (TB core)
				6	Athlon (PM core)
				7	Duron (MG core)
				8	Athlon (TH/AP core)
				A	Athlon (BT core)
			AMD K8	5	Athlon 64 FX or Opteron (0.13 µm)
				4	Athlon 64 (0.13 µm 754)
				7	Athlon 64 (0.13 µm 939)
				8	Athlon 64 (0.13 µm 754)
				B	Athlon 64 (0.13 µm 939)
				C	Athlon 64 (0.13 µm 754)
				F	Athlon 64 (0.13 µm 939)
				E	due to erratum #108: interpret like C
			Intel P4-core	0	P4 (0.18 µm)
				1	P4 (0.18 µm)
				2	P4 (0.13 µm)
				3	P4 (0.09 µm)
				4	P4 (0.09 µm)
			Intel Itanium	0	Merced (0.18 µm)
			Intel Itanium 2	0	McKinley (0.18 µm)
				1	Madison or Deerfield (0.13 µm)
				2	Madison 9M (0.13 µm)

		stepping	The stepping is encoded in bits 3..0.	
			The stepping values are processor-specific.	
	EBX=aall_ccbbh	brand ID	The brand ID is encoded in bits 7..0.	
			00h	not supported
			01h	0.18 µm Intel Celeron
			02h	0.18 µm Intel Pentium III
			03h	0.18 µm Intel Pentium III Xeon
			03h	0.13 µm Intel Celeron
			04h	0.13 µm Intel Pentium III
			07h	0.13 µm Intel Celeron mobile
			06h	0.13 µm Intel Pentium III mobile
			0Ah	0.18 µm Intel Celeron 4
			08h	0.18 µm Intel Pentium 4
			09h	0.13 µm Intel Pentium 4
			0Eh	0.18 µm Intel Pentium 4 Xeon
			0Bh	0.18 µm Intel Pentium 4 Xeon MP
			0Bh	0.13 µm Intel Pentium 4 Xeon
			0Ch	0.13 µm Intel Pentium 4 Xeon MP
			08h	0.13 µm Intel Celeron 4 mobile (0F24h)
			0Fh	0.13 µm Intel Celeron 4 mobile (0F27h)
			0Eh	0.13 µm Intel Pentium 4 mobile (production)
			0Fh	0.13 µm Intel Pentium 4 mobile (samples)
			11h	mobile Intel ??? processor
			12h	0.13 µm Intel Celeron M
			12h	0.09 µm Intel Celeron M
			13h	mobile Intel Celeron ? processor
			14h	Intel Celeron ? processor
			15h	mobile Intel ??? processor
			16h	0.13 µm Intel Pentium M

			16h	0.09 µm Intel Pentium M
			17h	mobile Intel Celeron ? processr
			000xxxxxb	engineering sample NN (NN=xxxxxb)
			001xxxxxb	AMD Athlon 64 XX00+ (XX=22+xxxxxb)
			010xxxxxb	AMD Athlon 64 XX00+ (XX=22+xxxxxb) mobile
			011xxxxxb	AMD Opteron UP 1YY (YY=38+2*xxxxxb)
			100xxxxxb	AMD Opteron DP 2YY (YY=38+2*xxxxxb)
			101xxxxxb	AMD Opteron MP 8YY (YY=38+2*xxxxxb)
			n/a [#6]	AMD Athlon 64 FX-ZZ (ZZ=24+xxxxxb)
		CLFLUSH	The CLFLUSH (8-byte) chunk count is encoded in bits 15..8.	
		CPU count	The logical processor count is encoded in bits 23..16.	
		APIC ID	The (fixed) default APIC ID is encoded in bits 31..24.	
	ECX=xxxx_xxxxh	feature flags	description	
		bits 31...15	reserved	
		bit 14 (ETPRD)	MISC_ENABLE.ETPRD	
		bit 13 (CX16)	CMPXCHG16B	
		bit 12	reserved	
		bit 11	reserved	
		bit 10 (CID)	context ID: the L1 data cache can be set to adaptive or shared mode MISC_ENABLE.L1DCCM	
		bit 9	reserved	
		bit 8 (TM2)	MISC_ENABLE.TM2E THERM_INTERRUPT and THERM_STATUS MSRs xAPIC thermal LVT entry THERM2_CONTROL MSR	
		bit 7 (EST)	Enhanced SpeedStep Technology	
		bit 6	reserved	
		bit 5	reserved	

		bit 4 (DSCPL)	CPL-qualified Debug Store
		bit 3 (MON)	MONITOR/MWAIT, MISC_ENABLE.MONE, MISC_ENABLE.LCMV MONITOR_FILTER_LINE_SIZE MSR also see standard level 0000_0005h setting MISC_ENABLE.MONE=0 causes MON=0
		bit 2	reserved
		bit 1	reserved
		bit 0 (SSE3)	SSE3, MXCSR, CR4.OSXMMEXCPT, #XF, if FPU=1 then also FISTTP
	EDX=xxxx_xxxxh	feature flags	description
		bit 31 (PBE)	Pending Break Event, STPCLK, FERR#, MISC_ENABLE.PBE
		bit 30 (IA-64)	IA-64, JMPE Jv, JMPE Ev
		bit 29 (TM1)	MISC_ENABLE.TM1E THERM_INTERRUPT and THERM_STATUS MSRs xAPIC thermal LVT entry
		bit 28 (HTT)	Hyper-Threading Technology, PAUSE
		bit 27 (SS)	selfsnoop
		bit 26 (SSE2)	SSE2, MXCSR, CR4.OSXMMEXCPT, #XF
		bit 25 (SSE)	SSE, MXCSR, CR4.OSXMMEXCPT, #XF
		bit 24 (FXSR)	FXSAVE/FXRSTOR, CR4.OSFXSR
		bit 23 (MMX)	MMX
		bit 22 (ACPI)	THERM_CONTROL MSR
		bit 21 (DTES)	Debug Trace and EMON Store MSRs
		bit 20	reserved
		bit 19 (CLFL)	CLFLUSH
		bit 18 (PSN)	PSN (see standard level 0000_0003h), MISC_CTL.PSND #1
		bit 17 (PSE36)	4 MB PDE bits 16..13, CR4.PSE
		bit 16 (PAT)	PAT MSR, PDE/PTE.PAT
		bit 15 (CMOV)	CMOVcc, if FPU=1 then also FCMOVcc/F (U)COMI(P)
		bit 14 (MCA)	MCG_*/MCn_* MSRs, CR4.MCE, #MC
		bit 13 (PGE)	PDE/PTE.G, CR4.PGE

		bit 12 (MTRR)	MTRR* MSRs
		bit 11 (SEP)	SYSENTER/SYSEXIT, SEP_* MSRs #2
		bit 10	reserved
		bit 9 (APIC)	APIC #3, #4
		bit 8 (CX8)	CMPXCHG8B #5
		bit 7 (MCE)	MCAR/MCTR MSRs, CR4.MCE, #MC
		bit 6 (PAE)	64bit PDPTE/PDE/PTEs, CR4.PAE
		bit 5 (MSR)	MSRs, RDMSR/WRMSR
		bit 4 (TSC)	TSC, RDTSC, CR4.TSD (doesn't imply MSR=1)
		bit 3 (PSE)	PDE.PS, PDE/PTE.res, CR4.PSE, #PF (1xxxb)
		bit 2 (DE)	CR4.DE, DR7.RW=10b, #UD on MOV from/to DR4/5
		bit 1 (VME)	CR4.VME/PVI, EFLAGS.VIP/VIF, TSS32.IRB
		bit 0 (FPU)	FPU

notes	description
#1	If the PSN has been disabled, then the PSN feature flag will read as 0. In addition the value for the maximum supported standard level (reported by standard level 0000_0000h, register EAX) will be lower.
#2	The Intel P6 processor does not support SEP, but inadvertently reports it.
#3	If the APIC has been disabled, then the APIC feature flag will read as 0.
#4	Early AMD K5 processors (SSA5) inadvertently used this bit to report PGE support.
#5	Some processors do support CMPXCHG8B, but don't report it by default. This is due to a Windows NT bug.
#6	The upper 3 bits aren't sufficient to encode a value of 9=1001b. Thus the Athlon 64 FX requires the 12-bit brand ID.

standard level 0000_0002h

input	EAX=0000_0002h	get processor configuration descriptors	
output	AL	number of times this level must be queried to obtain all configuration descriptors #1	
	EAX.15..8 EAX.23..16 EAX.31..24 EBX.0..7 EBX.15..8	configuration descriptors #2	
		value	description
		00h	null descriptor (=unused descriptor)

		EBX.23..16 EBX.31..24 ECX.0..7 ECX.15..8 ECX.23..16 ECX.31..24 EDX.0..7 EDX.15..8 EDX.23..16 EDX.31..24	01h	code TLB, 4K pages, 4 ways, 32 entries
			02h	code TLB, 4M pages, fully, 2 entries
			03h	data TLB, 4K pages, 4 ways, 64 entries
			04h	data TLB, 4M pages, 4 ways, 8 entries
			06h	code L1 cache, 8 KB, 4 ways, 32 byte lines
			08h	code L1 cache, 16 KB, 4 ways, 32 byte lines
			0Ah	data L1 cache, 8 KB, 2 ways, 32 byte lines
			0Ch	data L1 cache, 16 KB, 4 ways, 32 byte lines
			10h	data L1 cache, 16 KB, 4 ways, 32 byte lines (IA-64)
			15h	code L1 cache, 16 KB, 4 ways, 32 byte lines (IA-64)
			1Ah	code and data L2 cache, 96 KB, 6 ways, 64 byte lines (IA-64)
			22h	code and data L3 cache, 512 KB, 4 ways (!), 64 byte lines, dual-sectored
			23h	code and data L3 cache, 1024 KB, 8 ways, 64 byte lines, dual-sectored
			25h	code and data L3 cache, 2048 KB, 8 ways, 64 byte lines, dual-sectored
			29h	code and data L3 cache, 4096 KB, 8 ways, 64 byte lines, dual-sectored
			2Ch	data L1 cache, 32 KB, 8 ways, 64 byte lines
			30h	code L1 cache, 32 KB, 8 ways, 64 byte lines
			39h	code and data L2 cache, 128 KB, 4 ways, 64 byte lines, sectored
			3Bh	code and data L2 cache, 128 KB, 2 ways, 64 byte lines, sectored
				code and data L2 cache, 256

3Ch	KB, 4 ways, 64 byte lines, sectored
40h	no integrated L2 cache (P6 core) or L3 cache (P4 core)
41h	code and data L2 cache, 128 KB, 4 ways, 32 byte lines
42h	code and data L2 cache, 256 KB, 4 ways, 32 byte lines
43h	code and data L2 cache, 512 KB, 4 ways, 32 byte lines
44h	code and data L2 cache, 1024 KB, 4 ways, 32 byte lines
45h	code and data L2 cache, 2048 KB, 4 ways, 32 byte lines
50h	code TLB, 4K/4M/2M pages, fully, 64 entries
51h	code TLB, 4K/4M/2M pages, fully, 128 entries
52h	code TLB, 4K/4M/2M pages, fully, 256 entries
5Bh	data TLB, 4K/4M pages, fully, 64 entries
5Ch	data TLB, 4K/4M pages, fully, 128 entries
5Dh	data TLB, 4K/4M pages, fully, 256 entries
60h	data L1 cache, 16 KB, 8 ways, 64 byte lines, sectored
66h	data L1 cache, 8 KB, 4 ways, 64 byte lines, sectored
67h	data L1 cache, 16 KB, 4 ways, 64 byte lines, sectored
68h	data L1 cache, 32 KB, 4 ways, 64 byte lines, sectored
70h	trace L1 cache, 12 KμOPs, 8 ways
71h	trace L1 cache, 16 KμOPs, 8 ways
72h	trace L1 cache, 32 KμOPs, 8 ways
77h	code L1 cache, 16 KB, 4 ways, 64 byte lines, sectored (IA-64)
78h	code and data L2 cache, 1024 KB, 4 ways, 64 byte lines

			79h	code and data L2 cache, 128 KB, 8 ways, 64 byte lines, dual-sectored
			7Ah	code and data L2 cache, 256 KB, 8 ways, 64 byte lines, dual-sectored
			7Bh	code and data L2 cache, 512 KB, 8 ways, 64 byte lines, dual-sectored
			7Ch	code and data L2 cache, 1024 KB, 8 ways, 64 byte lines, dual-sectored
			7Dh	code and data L2 cache, 2048 KB, 8 ways, 64 byte lines
			7Eh	code and data L2 cache, 256 KB, 8 ways, 128 byte lines, sect. (IA-64)
			7Fh	code and data L2 cache, 512 KB, 2 ways, 64 byte lines
			81h	code and data L2 cache, 128 KB, 8 ways, 32 byte lines
			82h	code and data L2 cache, 256 KB, 8 ways, 32 byte lines
			83h	code and data L2 cache, 512 KB, 8 ways, 32 byte lines
			84h	code and data L2 cache, 1024 KB, 8 ways, 32 byte lines
			85h	code and data L2 cache, 2048 KB, 8 ways, 32 byte lines
			86h	code and data L2 cache, 512 KB, 4 ways, 64 byte lines
			87h	code and data L2 cache, 1024 KB, 8 ways, 64 byte lines
			88h	code and data L3 cache, 2048 KB, 4 ways, 64 byte lines (IA-64)
			89h	code and data L3 cache, 4096 KB, 4 ways, 64 byte lines (IA-64)
			8Ah	code and data L3 cache, 8192 KB, 4 ways, 64 byte lines (IA-64)
			8Dh	code and data L3 cache, 3096 KB, 12 ways, 128 byte lines (IA-64)

		90h	code TLB, 4K...256M pages, fully, 64 entries (IA-64)
		96h	data L1 TLB, 4K...256M pages, fully, 32 entries (IA-64)
		9Bh	data L2 TLB, 4K...256M pages, fully, 96 entries (IA-64)
		B0h	code TLB, 4K pages, 4 ways, 128 entries
		B3h	data TLB, 4K pages, 4 ways, 128 entries
		F0h	64 byte prefetching
		F1h	128 byte prefetching
		value	description
		70h	Cyrix specific: code and data TLB, 4K pages, 4 ways, 32 entries
		74h	Cyrix specific: ???
		77h	Cyrix specific: ???
		80h	Cyrix specific: code and data L1 cache, 16 KB, 4 ways, 16 byte lines
		82h	Cyrix specific: ???
		84h	Cyrix specific: ???
		value	description
		others	reserved
example (here: P6)		EAX=0302_0101h EBX=0000_0000h ECX=0000_0000h EDX=0604_0A43h	Because AL is 01h, one invocation of the level is enough to obtain all the configuration descriptors. All of them are valid because their highest bits are 0. This P6 processor includes a 4K/M code/data TLB, an 8+8 KB code/data L1 cache and an integrated 512 KB code and data L2 cache.

notes	description
#1	In a MP system special precautions must be taken when executing standard level 0000_0002h more than once. In particular it must be ensured that the same CPU is used during that entire process.
#2	Programs must no expect any particular order for the reported configuration descriptors.

standard level 0000_0003h

input	EAX=0000_0003h	get processor serial number [#1]
output	EBX=xxxx_xxxxh	processor serial number (Transmeta Crusoe processors only)
	ECX=xxxx_xxxxh	processor serial number
	EDX=xxxx_xxxxh	processor serial number

note	description
#1	This level is only supported and enabled if the PSN feature flag is set. The reported processor serial number should be combined with the vendor ID string and the processor type/family/model/stepping value, to distinguish cases in which two processors from different vendors happen to have the same serial number. Finally, it should be noted that most vendors can not guarantee that their serial numbers are truely unique.

standard level 0000_0004h

input	EAX=0000_0004h	get cache configuration descriptors [#1, #2]
	ECX=xxxx_xxxxh	cache level to query (0=L1D, 1=L2, all other values result in zero output)

output		bits	description
	EAX	31..26	cores per die - 1
		25..14	threads per cache - 1
		13..10	reserved
		9	fully associative?
		8	self-initializing?
		7..5	cache level (starts at 1)
		4..0	cache type (0=null, 1=data, 2=code, 3=unified, 4..31=reserved)
	EBX	bits	description
		31..22	ways of associativity - 1
		21..12	physical line partitions - 1
		11..0	system coherency line size - 1
	ECX	bits	description
		31..0	sets - 1
	EDX	bits	description
		31..0	reserved

notes	description
#1	This level is only enabled if MISC_ENABLE.LCMV is set to 0. This is due to a Windows NT bug.

#2	In a MP system special precautions must be taken when executing standard level 0000_0004h more than once. In particular it must be ensured that the same CPU is used during that entire process.

standard level 0000_0005h

input	EAX=0000_0005h	get MON information #1	
output	EAX	bits	description
		31..16	reserved
		15..0	smallest monitor line size in bytes
	EBX	bits	description
		31..16	reserved
		15..0	largest monitor line size in bytes
	ECX	bits	description
		31..0	reserved
	EDX	bits	description
		31..0	reserved

notes	description
#1	This level is only enabled if MISC_ENABLE.LCMV is set to 0. This is due to a Windows NT bug.

---

extended level 8000_0000h

input	EAX=8000_0000h	get maximum supported extended level and vendor ID string	
output	EAX=xxxx_xxxxh	maximum supported extended level	
	EBX-EDX-ECX	vendor ID string	
		`AuthenticAMD`	AMD processor
		reserved	Cyrix processor
		reserved	Centaur processor
		reserved	Intel processor
		`TransmetaCPU`	Transmeta processor
		reserved	National Semiconductor processor (GX1, GXLV, GXm)
		`Geode by NSC`	National Semiconductor processor (GX2)

extended level 8000_0001h

input	EAX=8000_0001h	get processor family/model/stepping and features flags #0		
output	EAX=0000_0xxxh	processor family/model/stepping		
		extended family	The extended processor family is encoded in bits 27..20.	
			00h	AMD K8
		extended model	The extended processor model is encoded in bits 19..16.	
		family	The family is encoded in bits 11..8.	
			5	AMD K5 NS Geode Centaur C2 and C3 Transmeta Crusoe TM3x00 and TM5x00
			6	AMD K6 VIA C3
			7	AMD K7
			F	refer to extended family (here: add)
		model	The model is encoded in bits 7..4.	
			all	F	refer to extended model (here: concat)
			AMD K5	1	5k86 (PR120 or PR133)
				2	5k86 (PR166)
				3	5k86 (PR200)
			AMD K6	6	K6 (0.30 µm)
				7	K6 (0.25 µm)
				8	K6-2
				9	K6-III
				D	K6-2+ or K6-III+ (0.18 µm)
			AMD K7	1	Athlon (0.25 µm)
				2	Athlon (0.18 µm)
				3	Duron (SF core)
				4	Athlon (TB core)
				6	Athlon (PM core)
				7	Duron (MG core)

				8	Athlon (TH/AP core)
				A	Athlon (BT core)
			AMD K8	5	Athlon 64 FX or Opteron (0.13 µm)
				4	Athlon 64 (0.13 µm 754)
				7	Athlon 64 (0.13 µm 939)
				8	Athlon 64 (0.13 µm 754)
				B	Athlon 64 (0.13 µm 939)
				C	Athlon 64 (0.13 µm 754)
				F	Athlon 64 (0.13 µm 939)
			NS Geode	4	GX1, GXLV, GXm
				5	GX2
			Centaur	8	C2
				9	C3
			VIA C3	5	Cyrix M2 core
				6	WinChip C5A core
				7	WinChip C5B core (if stepping = 0..7)
				7	WinChip C5C core (if stepping = 8..F)
				8	WinChip C5N core (if stepping = 0..7)
				9	WinChip C5XL core (if stepping = 0..7)
				9	WinChip C5P core (if stepping = 8..F)
				10	WinChip C5J core
			Transmeta	4	Crusoe TM3x00 and TM5x00
		stepping	The stepping is encoded in bits 3..0.		
			The stepping values are processor-specific.		
	EBX=0000_0xxxh	brand ID	The brand ID is encoded in bits 11..0		
			000h	not supported	
			0xxh	engineering sample NN (NN=xxh)	

			1xxh	AMD Athlon 64 XX00+ (XX=22+xxh)
			2xxh	AMD Athlon 64 XX00+ (XX=22+xxh) mobile
			3xxh	AMD Opteron UP 1YY (YY=38+2*xxh)
			4xxh	AMD Opteron DP 2YY (YY=38+2*xxh)
			5xxh	AMD Opteron MP 8YY (YY=38+2*xxh)
			9xxh	AMD Athlon 64 FX-ZZ (ZZ=24+xxh)
	ECX=xxxx_xxxxh	feature flags	description of indicated feature	
		bits 31...5	reserved	
		bit 4 (CR8D)	MOV from/to CR8D by means of LOCK-prefixed MOV from/to CR0	
		bit 3	reserved	
		bit 2	reserved	
		bit 1 (CMP)	HTT=1 indicates HTT (0) or CMP (1)	
		bit 0 (AHF64)	LAHF and SAHF in PM64	
	EDX=xxxx_xxxxh	feature flags	description of indicated feature	
		bit 31 (3DNow!)	3DNow!	
		bit 30 (3DNow!+)	extended 3DNow!	
		bit 29 (LM)	AA-64, Long Mode	
		bit 28	reserved	
		bit 27 (TSCP)	TSC, TSC_AUX, RDTSCP, CR4.TSD	
		bit 26	reserved	
		bit 25 (FFXSR)	EFER.FFXSR	
		bit 24 (MMX+) bit 24 (FXSR)	Cyrix specific: extended MMX AMD K7: FXSAVE/FXRSTOR, CR4.OSFXSR	
		bit 23 (MMX)	MMX	
		bit 22 (MMX+)	AMD specific: MMX-SSE and SSE-MEM	
		bit 21	reserved	
		bit 20 (NX)	EFER.NXE, P?E.NX, #PF(1xxxx)	
		bit 19 (MP)	MP-capable [#3]	

		bit 18	reserved
		bit 17 (PSE36)	4 MB PDE bits 16..13, CR4.PSE
		bit 16 (FCMOV) bit 16 (PAT)	FCMOVcc/F(U)COMI(P) (implies FPU=1) AMD K7: PAT MSR, PDE/PTE.PAT
		bit 15 (CMOV)	CMOVcc
		bit 14 (MCA)	MCG_*/MCn_* MSRs, CR4.MCE, #MC
		bit 13 (PGE)	PDE/PTE.G, CR4.PGE
		bit 12 (MTRR)	MTRR* MSRs
		bit 11 (SEP)	SYSCALL/SYSRET, EFER/STAR MSRs #1
		bit 10	reserved #1
		bit 9 (APIC)	APIC #2
		bit 8 (CX8)	CMPXCHG8B
		bit 7 (MCE)	MCAR/MCTR MSRs, CR4.MCE, #MC
		bit 6 (PAE)	64bit PDPTE/PDE/PTEs, CR4.PAE
		bit 5 (MSR)	MSRs, RDMSR/WRMSR
		bit 4 (TSC)	TSC, RDTSC, CR4.TSD (doesn't imply MSR=1)
		bit 3 (PSE)	PDE.PS, PDE/PTE.res, CR4.PSE, #PF (1xxxb)
		bit 2 (DE)	CR4.DE, DR7.RW=10b, #UD on MOV from/to DR4/5
		bit 1 (VME)	CR4.VME/PVI, EFLAGS.VIP/VIF, TSS32.IRB
		bit 0 (FPU)	FPU

note	description
#0	Intel processors don't support this function; they return zero in EAX, EBX, ECX, and EDX.
#1	The AMD K6 processor, model 6, uses bit 10 to indicate SEP. Beginning with model 7, bit 11 is used instead.
#2	If the APIC has been disabled, then the APIC feature flag will read as 0.
#3	AMD K7 processors prior to CPUID=0662h may report 0 even if they are MP-capable.

extended levels 8000_0002h, 8000_0003h, and 8000_0004h

input	EAX=8000_0002h	get processor name string (part 1)	
	EAX=8000_0003h	get processor name string (part 2)	
	EAX=8000_0004h	get processor name string (part 3)	
output	EAX EBX ECX EDX	processor name string [#1]	
		AMD K5	`AMD-K5(tm) Processor`
		AMD K6	`AMD-K6tm w/ multimedia extensions`
		AMD K6-2	`AMD-K6(tm) 3D processor` `AMD-K6(tm)-2 Processor`
		AMD K6-III	`AMD-K6(tm) 3D+ Processor` `AMD-K6(tm)-III Processor`
		AMD K6-2+	`AMD-K6(tm)-III Processor` (?)
		AMD K6-III+	`AMD-K6(tm)-III Processor` (?)
		AMD K7	`AMD-K7(tm) Processor` (model 1) `AMD Athlon(tm) Processor` (model 2) newer models: programmable
		AMD K8	programmable via MSRs C001_0030h..C001_0035h, default is 48x 00h
		NS Geode GX2	`Geode(TM) Integrated Processor by National Semi` programmable via MSRs 0000_300Ah..0000_300Fh
		Centaur C2 [#2]	`IDT WinChip 2` `IDT WinChip 2-3D`
		Centaur C3	`IDT WinChip 3`
		VIA C3	`CYRIX III(tm)` (?) `VIA Samuel` (?) `VIA Ezra` (?) `VIA C3 Nehemiah` (?)
		Intel PM [#3]	`Intel(R) Pentium(R) M processor xxxxMHz`
		Intel P4 [#3]	`Intel(R) Pentium(R) 4 CPU xxxxMHz`
		Transmeta	`Transmeta(tm) Crusoe(tm) Processor TMxxxx`

notes	description
#1	Unused characters at the end of the string are filled with 00h.
#2	The string depends on whether 3DNow! is disabled or enabled.
#3	The string is right-justified, with leading whitespaces.

extended level 8000_0005h

input	EAX=8000_0005h	get L1 cache and TLB configuration descriptors [#1]	
output	EAX	4/2 MB L1 TLB configuration descriptor	
		bits	description
		31..24	data TLB associativity (FFh=full)
		23..16	data TLB entries
		15..8	code TLB associativity (FFh=full)
		7..0	code TLB entries
	EBX	4 KB L1 TLB configuration descriptor [#2]	
		bits	description
		31..24	data TLB associativity (FFh=full)
		23..16	data TLB entries
		15..8	code TLB associativity (FFh=full)
		7..0	code TLB entries
	ECX	data L1 cache configuration descriptor	
		bits	description
		31..24	data L1 cache size in KBs
		23..16	data L1 cache associativity (FFh=full)
		15..8	data L1 cache lines per tag
		7..0	data L1 cache line size in bytes
	EDX	code L1 cache configuration descriptor	
		bits	description
		31..24	code L1 cache size in KBs
		23..16	code L1 cache associativity (FFh=full)
		15..8	code L1 cache lines per tag
		7..0	code L1 cache line size in bytes

notes	description
#1	Cyrix processors return CPUID level 0000_0002h-like descriptors instead. (Though the NS Geode GX2 does not.)
#2	While Transmeta Crusoe processors have 256 entries, the CPUID definition constrains them to reporting only 255. For compatibility reasons they report their unified TLB twice: once for the code TLB, and once for the data TLB.

extended level 8000_0006h

input	EAX=8000_0006h	get L2 cache configuration descriptors

output	EAX	4/2 MB L2 TLB configuration descriptor [#1]	
		bits	description
		31..28	data TLB associativity [#2]
		27..16	data TLB entries
		15..12	code TLB associativity [#2]
		11..0	code TLB entries
	EBX	4 KB L2 TLB configuration descriptor [#1]	
		bits	description
		31..28	data TLB associativity [#2]
		27..16	data TLB entries
		15..12	code TLB associativity [#2]
		11..0	code TLB entries
	ECX	unified L2 cache configuration descriptor [#3]	
		bits	description
		31..16 [#5]	unified L2 cache size in KBs [#4]
		15..12 [#5]	unified L2 cache associativity [#2, #6]
		11..8 [#5]	unified L2 cache lines per tag
		7..0	unified L2 cache line size in bytes

note	description
#1	A unified L2 TLB is indicated by a value of 0000h in the upper 16 bits.
#2	0000b=L2 off, 0001b=direct mapped, 0010b=2-way, 0100b=4-way, 0110b=8-way, 1000b=16-way, 1111b=full
#3	The AMD K7 processor's L2 cache must be configured prior to relying upon this information, if the model is 1 or 2.
#4	AMD K7 processors with CPUID=0630h (Duron) inadvertently report 1 KB instead of 64 KB.
#5	VIA C3 processors with CPUID=0670..068Fh (C5B/C5C) inadvertently use bits 31..24, 23..16, and 15..8 instead.
#6	VIA C3 processors with CPUID=069x (C5XL) and stepping 1 inadvertently report 0 ways instead of 16 ways.

extended level 8000_0007h

input	EAX=8000_0007h	get enhanced power management (EPM) information	
output	EDX	EPM flags	
		bits	description
		31..6	reserved

		5 (STC)	software thermal control
		4 (TM)	thermal monitoring
		3 (TTP)	thermal trip
		2 (VID)	voltage ID control
		1 (FID)	frequency ID control
		0 (TS)	temperature sensor

extended level 8000_0008h

input	EAX=8000_0008h	get miscellaneous information	
output	EAX	address size information	
		bits	description
		31..16	reserved
		15..8	virtual address bits
		7..0	physical address bits
	ECX	processor count information	
		bits	description
		31..8	reserved
		7..0	cores per die - 1

Transmeta level 8086_0000h

input	EAX=8086_0000h	get maximum supported level and vendor ID string	
output	EAX=xxxx_xxxxh	maximum supported level	
	EBX-EDX-ECX	vendor ID string	
		`TransmetaCPU`	Transmeta processor

Transmeta level 8086_0001h

input	EAX=8086_0001h	get processor information		
output	EAX=0000_0xxxh	processor family/model/stepping		
		family	The family is encoded in bits 11..8.	
			5	Transmeta Crusoe TM3x00 and TM5x00
		model	The model is encoded in bits 7..4.	

			Transmeta	4	Crusoe TM3x00 and TM5x00
		stepping	The stepping is encoded in bits 3..0.		
			The stepping values are processor-specific.		
	EBX=aabb_ccddh	hardware revision (a.b-c.d), if 2000_0000h: see level 8086_0002h register EAX instead			
	ECX=xxxx_xxxxh	nominal core clock frequency (MHz)			
	EDX=xxxx_xxxxh	feature flags	description of indicated feature		
		bits 31..4	reserved		
		bit 3 (LRTI)	LongRun Table Interface		
		bit 2 (???)	unknown		
		bit 1 (LR)	LongRun		
		bit 0 (BAD)	recovery CMS active (due to a failed upgrade)		

Transmeta level 8086_0002h

input	EAX=8086_0002h	get processor information	
output	EAX	xxxx_xxxxh	reserved or hardware revision (xxxxxxxxh) see level 8086_0001h register EBX for details
	EBX	aabb_ccddh	software revision, part 1/2 (a.b.c-d-x)
	ECX	xxxx_xxxxh	software revision, part 2/2 (a.b.c-d-x)

Transmeta levels 8086_0003h, 8086_0004h, 8086_0005h, and 8086_0006h

input	EAX=8086_0003h	get information string (part 1)	
	EAX=8086_0004h	get information string (part 2)	
	EAX=8086_0005h	get information string (part 3)	
	EAX=8086_0006h	get information string (part 4)	
output	EAX-EBX-ECX-EDX	information string #1	
		Transmeta	`20000805 23:30 official release 4.1.4#2` (example)
notes	description		
#1	Unused characters at the end of the string are filled with 00h.		

Transmeta level 8086_0007h

input	EAX=8086_0007h	get processor information	
output	EAX	xxxx_xxxxh	current core clock frequency (MHz)
	EBX	xxxx_xxxxh	current core clock voltage (mV)
	ECX	xxxx_xxxxh	current (LongRun) performance level (0..100%)
	EDX	xxxx_xxxxh	current gate delay (fs)

Centaur level C000_0000h

input	EAX=C000_0000h	get maximum supported level
output	EAX=xxxx_xxxxh	maximum supported level

Centaur level C000_0001h

input	EAX=C000_0001h	get processor information	
output	EDX=xxxx_xxxxh	feature flags	description of indicated feature
		bits 31..10	reserved
		bit 9 (MM/HE-E)	Montgomery Multiplier and Hash Engine enabled
		bit 8 (MM/HE)	Montgomery Multiplier and Hash Engine
		bit 7 (ACE-E)	Advanced Cryptography Engine enabled
		bit 6 (ACE)	Advanced Cryptography Engine
		bit 5 (FEMMS)	FEMMS
		bit 4 (LH)	LongHaul MSR 0000_110Ah
		bit 3 (RNG-E)	Random Number Generator enabled
		bit 2 (RNG)	Random Number Generator
		bit 1 (AIS-E)	Alternate Instruction Set enabled
		bit 0 (AIS)	Alternate Instruction Set

mystery level 8FFF_FFFEh

input	EAX=8FFF_FFFEh	unknown #1	
output	EAX	0049_4544h	DEI
	EBX	0000_0000h	reserved
	ECX	0000_0000h	reserved
	EDX	0000_0000h	reserved

note	description
#1	This level is only supported by the AMD K6 processor family.

mystery level 8FFF_FFFFh

input	EAX=8FFF_FFFFh	unknown #1	
output	EAX EBX ECX EDX	string	NexGenerationAMD (K6) IT'S HAMMER TIME (K8)

note	description
#1	This level is only supported by the AMD K6 and K8 processor families.

---

all other levels

input	EAX=xxxx_xxxxh	desired CPUID level
output	EAX=xxxx_xxxxh EBX=xxxx_xxxxh ECX=xxxx_xxxxh EDX=xxxx_xxxxh	undefined

---