PyCPU 1.0 Manual

PyCPU is a small, efficient CPU detection library for Python. It is written mainly in C based on source code from the MPlayer project.

It is designed for use on Intel/AMD processors, both 32- and 64-bit. It will also run on Cyrix and Centaur CPUs.

What PyCPU Can't Do (Right Now)

PyCPU cannot detect the number of cores in a processor nor the total number of processors on an SMP system. This may be added on later.

Using PyCPU

To use PyCPU, copy pycpu.py and pycpu.dll (on Windows) to your project's folder.

In your Python script at the top, type:

```
import pycpu
```

It is recommended that you do not use the alternate “from pycpu import *” form as this may cause hard-to-locate errors.

To gather CPU information, call the getCPUInfo() function:

```
pycpu.getCPUInfo()
```

After calling this function, the ProcInfo structure will be filled with information about the processor.

An example script (example.py) has also been included that shows PyCPU in action.

The ProcInfo Structure

After calling pycpu.getCPUInfo(), the ProcInfo structure will be populated with information on the processor.

It contains the following members:

Member	Type	Purpose
vendor_id	String	CPU vendor ID, i.e., “GenuineIntel”
max_cpuid	Integer	Number of CPUID function codes available
amd_flags	Integer	Flags specific to AMD processors
amd_flags2	Integer	Extra AMD-specific flags
model_name	String	Name of the processor model. May be the same as processor_name.
processor_name	String	Full name of the processor.
family	Integer	The family number
model	Integer	Model number
stepping	Integer	The version number of the CPU
clock_mhz	Float	Clock speed of the CPU in MHz.
clock_ghz	Float	Clock speed of the CPU in GHz, if applicable. Under 1GHz this value is 0.00.

CPU Capabilities

PyCPU will query a processor for its capabilities (caps) and stores them in a list in the library. Caps are features that are enabled on the processor. There are currently 82 capabilities PyCPU checks for.

Note: Not all processors will have all capabilities available. There are some that are only specific

Caps can be retrieved by calling pycpu.getCap(), supplying the index of the cap in the list as the argument.

Calling pycpu.getCap() will return a string mnemonic. The meaning of each mnemonic is listed below in alphabetical order.

Mnemonic	Meaning
de	Debugging Extension
3dnow	3DNow!
3dnowext	3DNow! Extensions
3dnowprefetch	3DNow! Prefetch/PrefetchW
abm	Advanced Bit Manipulation
acpi	Thermal Monitor and Clock Ctrl (ACPI)
aes	AES Instruction
apic	On-chip APIC Hardware Enabled
avx	Advanced Vector Extension
centaur_mcr	Centaur Memory Control Registers
cid	L1 Context ID
clflush	CFLUSH instruction
cmov	Conditional Move/Compare Instruction
cmp_legacy	Chip Multi-Core
cr8_legacy	CR8 Available in Legacy Mode
cx16	CMPXCHG16B Available
cx8	CMPXCHG8B Instruction Supported
cyrix_arr	Cyrix Address Range Registers
dca	Direct Cache Access
ds_cpl	CPL Qualified Debug Store
dtes64	64-bit Debug Store
dts	Debug Store
est	Enhanced Intel SpeedStep Technology
extapic	Extended APIC Space
fma	Fused Multiply Add

Mnemonic	Meaning
fpu	Floating-point unit on-chip
fxsr	FXSAVE/FXRSTOR
fxsr_opt	Fast FXSAVE/FXRSTOR
ht	Hyper-Threading
ia64	IA-64 Processor (Itanium)
ibs	Instruction Based Sampling
k6_mtrr	AMD K6-2/K6-III Memory Type Range Registers
lahf_lm	LAHF/SAHF Supported in 64-bit Mode
lm	Long Mode Capable
mca	Machine Check Architecture
mce	Machine Check Exception
misalignsse	Misaligned SSE Mode
mmx	MMX Technology
mmxext	MMX Technology (AMD Extensions)
monitor	MONITOR/MWAIT
movbe	MOVBE Instruction
mp	Multiprocessor Capable
msr	Pentium Processor MSR
mtrr	Memory Type Range Registers
nx	No-Execute Page Protection
osvw	OS Visible Workaround
osxsave	XSAVE/XRSTOR Enabled in the OS
pae	Physical Address Extension
pat	Page Attribute Table
pbe	Pending Break Encoding
pclmulqdq	Carryless Multiplication
pdcm	Perf/Debug Capability MSR
pdpe1gb	PDP Entry for 1GiB Page

Mnemonic	Meaning
pge	PTE Global Bit
pn	Processor Serial Number
pni	SSE3 Extensions
popcnt	Pop Count Instruction
pse	Page Size Extension
pse36	Page Size Extension 36-bit
rdtscp	RDTSCP Instruction
sep	SYSENTER and SYSEXIT
skinit	SKINIT, STGI, and DEV Support
smx	Safer Mode Extensions
ss	Self Snoop
sse	SSE Extensions
sse2	SSE2 Extensions
sse4_1	SSE4.1 Extensions
sse4_2	SSE4.2 Extensions
sse4a	SSE4A Extensions
sse5	SSE5 Extensions
ssse3	Supplemental SSE3
svm	Secure Virtual Machine
syscall	SYSCALL and SYSRET
tm	Thermal Monitor
tm2	Thermal Monitor 2
tsc	Time Stamp Counter
vme	Virtual Mode Enhancements
vmx	Virtual Machine Extensions
wdt	Watchdog Timer Support
x2apic	x2APIC Feature
xsave	XSAVE/XRSTOR Extensions
xtpr	xTPR Disable