strange CPU speedups with SMP on Athlon 64 X2

Source: http://linux.derkeiler.com/Mailing−Lists/Kernel/2005−08/7684.html

---

From: Nathan Becker (*nbecker_at_physics.ucsb.edu*)
Date: 08/30/05

```
Date:   Tue, 30 Aug 2005 12:16:04 −0700 (PDT)
To: linux−kernel@vger.kernel.org
```

Hi,

I'm having a strange problem when I benchmark some of my physics simulation code on my new Athlon 64 X2 4800 machine. It occurs on all current kernels that I have tested including 2.6.12.5 and 2.6.13.

If I run my benchmark single threaded, so that one of the two CPU cores is just idling then the calculation goes pretty fast. But if I load both CPU cores simultaneously but with INDEPENDENT calculations, then each calculation runs about 12−15% faster than when running alone. I have found this to be always reproducible. There is no disk access involved in the calculation and RAM usage is fairly minimal so this is not caused by caching. Also, if I compile the kernel to disable SMP then the machine runs a single calculation at the same speed as when running alone when SMP is enabled.

I am aware of the timing issues on these machines (especially since I reported the bug http://bugzilla.kernel.org/show_bug.cgi?id=5105). However, I double−checked my benchmark with a stop−watch, so this is independent of something strange happening in the timer.

I also checked the cpufreq governor and according to the logs, my CPU is holding steady at the maximum setting of 2.4GHz. I set the governor to "performance" mode which should prevent unintended downclocking.

I would be happy to post my exact C source that I use to do the benchmark, but I wanted to get some feedback first in case I'm just doing something stupid. Also, since I'm not subscribed to this list, please cc me directly regarding this topic.

Thanks very much,

Nathan
−
To unsubscribe from this list: send the line "unsubscribe linux−kernel" in the body of a message to majordomo@vger.kernel.org
More majordomo info at http://vger.kernel.org/majordomo−info.html

Please read the FAQ at http://www.tux.org/lkml/