

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-02/msg10063.html>

- *From:* "Jesper Juhl" <jesper.juhl@xxxxxxxxx>
 - *Date:* Tue, 28 Feb 2006 20:59:55 +0100
-

Hi everyone,

This is a continuation of the issue I initially reported in the "make -j with j <= 4 seems to only load a single CPU core" thread (<http://lkml.org/lkml/2006/2/21/231>).

I've now got some more data, so hopefully someone can help me get a handle on this thing.

In a nutshell the problem is that I need to run "make -j N" where N is

= 5 in order to put maximum load on both cores of my Athlon 64 X2

4400+ when building kernels and with N < 5 the build appears to be pretty much done serially and not at all parallel. This baffles me to be honest.

The expected behaviour is that running with "make -j 2" on an otherwise idle machine would schedule a CC to run on each core most of the time, and with "make -j 3" & "make -j 4" there should definately be something executing on both cores all the time.

What I see is that unless I bump it up to -j 5 or greater only one core seems to be put to work and the other one mostly just sits around spinning its wheels doing nothing.

To try and gather some data on this I ran the following in a 2.6.16-rc5-mm1 source tree :

```
$ for i in `seq 1 6`; do make distclean ; make allnoconfig ; sleep 1 ;  
vmstat 5 2>&1 > vmstat${i}.log & sleep 1 ; time nice make -j ${i} 2>&1 |  
tee build${i}.log ; sleep 1 ; killall vmstat ; done
```

The timing information clearly show a distinct difference between "-j 4" and "-j 5" :

```
"make -j 1" run :  
real 1m20.454s  
user 0m56.836s  
sys 0m12.023s
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
"make -j 2" run :  
real 1m13.786s  
user 0m56.723s  
sys 0m12.164s
```

```
"make -j 3" run :  
real 1m14.208s  
user 0m56.960s  
sys 0m11.912s
```

```
"make -j 4" run :  
real 1m14.689s  
user 0m56.897s  
sys 0m12.062s
```

```
"make -j 5" run :  
real 0m47.613s  
user 0m57.118s  
sys 0m12.503s
```

```
"make -j 6" run :  
real 0m47.750s  
user 0m57.096s  
sys 0m12.516s
```

The "make -j 1" run is slightly slower than the 2, 3 & 4 ones. The 2, 3 & 4 runs are more or less identical. But when we get to the 5 & 6 runs we finally see both cores getting some useful work done and the build time improves significantly (note, I did the "-j 6" run just to show that going beyond 5 doesn't seem to change things).

Dick Johnson suggested in the previous thread that my problem might be due to an I/O bottleneck (but I don't think that's the case), so that's why I decided to have vmstat running in the background during the builds to get a little data on that (it also shows quite nicely that approximately half the CPU resources are not getting utilized until we get to "make -j 5").

Here's the vmstat data :

```
$ cat vmstat1.log  
procs -----memory-----swap-- -----io----- --system-- -----cpu-----  
r b swpd free buff cache si so bi bo in cs us sy id wa  
0 0 0 1601296 48296 175492 0 0 286 59 531 206 6 3 82 10  
1 0 0 1584208 49292 179188 0 0 326 464 1069 424 23 10 50 18  
2 0 0 1584672 50364 180428 0 0 110 550 1045 360 43 7 39 11  
1 0 0 1578096 51268 181292 0 0 45 474 1042 439 50 10 33 8  
1 0 0 1571188 52296 182780 0 0 160 492 1046 386 45 7 38 9  
2 0 0 1576412 53188 183656 0 0 78 452 1036 333 48 7 41 4
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
1 0 0 1570780 54044 184432 0 0 58 490 1050 383 46 8 37 9
1 0 0 1570180 54976 185404 0 0 65 521 1043 307 44 6 38 12
1 0 0 1572244 55784 186228 0 0 86 394 1033 323 47 7 42 5
1 0 0 1559940 56704 187144 0 0 81 447 1059 428 51 5 37 7
1 0 0 1558040 57508 187768 0 0 62 332 1051 495 59 6 33 3
1 0 0 1565836 58336 188640 0 0 54 351 1052 551 56 8 33 3
1 0 0 1559440 59424 189796 0 0 62 586 1055 449 50 7 36 7
1 0 0 1551172 60376 190680 0 0 61 512 1050 405 48 8 33 12
1 0 0 1553376 61196 191696 0 0 77 355 1023 353 50 7 39 5
0 1 0 1555624 62304 192900 0 0 36 621 1036 475 52 10 31 7
0 2 0 1547804 63476 198324 0 0 71 1027 1038 520 52 11 30 7
```

\$ cat vmstat2.log

```
procs -----memory-----swap-----io-----system-----cpu-----
r b swpd free buff cache si so bi bo in cs us sy id wa
0 0 0 1568828 64716 184844 0 0 235 109 530 208 13 4 73 10
1 0 0 1564056 64728 186260 0 0 695 1067 430 34 11 39 15
1 0 0 1555020 64736 186728 0 0 485 1035 381 50 8 37 4
1 0 0 1549500 64748 187532 0 0 455 1028 477 56 11 30 2
0 1 0 1560680 64760 187996 0 0 1 514 1026 313 49 7 40 4
1 0 0 1555364 64768 188396 0 0 461 1026 384 51 8 38 3
2 0 0 1550752 64776 189136 0 0 289 1029 366 52 8 37 3
1 0 0 1549824 64784 189536 0 0 258 1031 303 48 7 38 7
1 0 0 1551600 64792 189936 0 0 470 1046 268 43 5 41 11
1 0 0 1549540 64800 190336 0 0 494 1042 302 52 6 41 1
1 0 0 1549728 64808 190804 0 0 388 1030 354 51 7 39 3
1 0 0 1549532 64820 191608 0 0 474 1041 384 51 8 38 3
1 0 0 1548908 64836 192136 0 0 531 1043 361 48 8 37 6
4 0 0 1553396 64844 192944 0 0 359 1023 361 53 7 40 0
1 0 0 1551280 64852 193820 0 0 658 1052 462 50 10 31 9
0 0 0 1544148 64884 202152 0 0 496 1033 528 51 11 35 3
```

\$ cat vmstat3.log

```
procs -----memory-----swap-----io-----system-----cpu-----
r b swpd free buff cache si so bi bo in cs us sy id wa
0 3 0 1567996 64904 184928 0 0 198 142 529 207 18 5 67 10
1 0 0 1554976 64916 186548 0 0 239 1099 464 40 12 31 18
3 0 0 1562264 64924 186880 0 0 759 1056 345 42 8 37 13
1 0 0 1552476 64932 187688 0 0 416 1033 462 54 11 32 4
2 0 0 1561064 64940 188156 0 0 661 1058 294 45 6 41 7
1 0 0 1557116 64948 188488 0 0 451 1033 316 48 7 40 5
1 0 0 1553528 64956 189228 0 0 453 1036 374 52 8 37 3
2 0 0 1555884 64964 189628 0 0 379 1022 348 53 8 37 3
1 0 0 1552480 64976 190160 0 0 1 488 1037 271 49 6 41 5
1 0 0 1549816 64984 190492 0 0 427 1036 284 52 6 39 3
1 0 0 1550748 64992 190960 0 0 398 1033 342 50 8 38 4
1 0 0 1550820 65004 191832 0 0 474 1032 387 54 8 36 2
1 0 0 1546484 65012 192368 0 0 534 1045 388 51 8 37 4
2 0 0 1548228 65020 193176 0 0 419 1029 351 51 7 39 3
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
1 0 0 1545136 65028 193916 0 0 0 608 1047 432 51 9 34 7
```

```
$ cat vmstat4.log
```

```
procs -----memory-----swap-----io-----system-----cpu-----
r b swpd free buff cache si so bi bo in cs us sy id wa
1 0 0 1567896 65088 185084 0 0 170 168 529 205 22 5 62 10
1 0 0 1553288 65096 186504 0 0 0 574 1040 437 35 11 45 9
1 0 0 1558744 65104 187040 0 0 0 527 1045 373 49 8 37 6
2 0 0 1558024 65112 187848 0 0 0 559 1040 444 52 11 32 6
1 0 0 1552320 65120 188248 0 0 0 486 1036 309 46 6 41 6
1 0 0 1555052 65128 188648 0 0 0 337 1023 353 51 7 40 3
1 0 0 1558096 65136 189388 0 0 0 525 1028 393 54 9 37 0
0 1 0 1550844 65144 189788 0 0 0 406 1029 341 51 7 38 4
1 1 0 1549176 65152 190324 0 0 0 330 1027 275 49 5 43 2
1 0 0 1552092 65160 190588 0 0 0 298 1049 251 40 5 43 12
2 0 0 1551660 65168 190988 0 0 0 371 1030 323 48 6 41 4
1 0 0 1548632 65180 191928 0 0 1 318 1019 395 54 9 37 0
1 0 0 1545100 65180 192404 0 0 0 587 1047 397 52 9 36 4
1 0 0 1548332 65188 193076 0 0 0 481 1041 300 47 6 42 5
1 0 0 1545900 65196 194020 0 0 0 490 1039 444 50 10 32 8
1 0 0 1540988 65216 199032 0 0 0 529 1040 547 55 12 29 4
```

```
$ cat vmstat5.log
```

```
procs -----memory-----swap-----io-----system-----cpu-----
r b swpd free buff cache si so bi bo in cs us sy id wa
1 0 0 1567272 65252 185260 0 0 149 188 528 204 24 6 59 10
5 0 0 1557680 65260 186748 0 0 0 669 1059 609 46 13 30 12
4 0 0 1542556 65268 187556 0 0 0 450 1056 724 85 14 0 1
4 1 0 1557872 65268 188372 0 0 0 549 1050 577 82 13 0 5
0 6 0 1528352 65276 189180 0 0 0 618 1029 492 85 12 0 3
3 1 0 1540072 65292 189912 0 0 0 168 1040 454 79 13 0 8
3 0 0 1532288 65312 190980 0 0 0 388 1028 478 88 12 0 0
4 0 0 1539840 65320 191720 0 0 0 706 1054 513 74 13 3 11
3 0 0 1538432 65328 192596 0 0 0 477 1037 518 85 14 0 1
4 0 0 1536152 65340 193876 0 0 0 634 1053 514 81 12 3 5
```

```
$ cat vmstat6.log
```

```
procs -----memory-----swap-----io-----system-----cpu-----
r b swpd free buff cache si so bi bo in cs us sy id wa
1 0 0 1566872 65384 185468 0 0 137 201 528 209 28 6 56 10
0 3 0 1552864 65400 187016 0 0 1 750 1047 505 48 14 30 8
1 3 0 1549040 65408 187756 0 0 0 609 1054 452 81 11 2 5
1 3 0 1540672 65416 188496 0 0 0 574 1032 473 83 14 0 3
4 1 0 1541544 65424 189304 0 0 0 176 1033 503 77 14 0 9
5 0 0 1540624 65432 190316 0 0 0 418 1035 481 80 12 0 8
3 0 0 1539580 65444 191120 0 0 0 673 1053 430 77 10 0 12
4 0 0 1535676 65452 191996 0 0 0 648 1058 506 76 15 1 9
4 0 0 1534440 65460 192736 0 0 0 508 1038 477 84 13 1 2
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
3 0 0 1536492 65468 194156 0 0 0 418 1020 549 86 14 0 0
```

To show that the builds with `-j 1, 2, 3 & 4` all seem to be quite serial but that the `-j 5 & 6` builds seem to be parallel here are the top 40 lines from the build logs.

As can be seen from the logs, the first four builds progress in a nice and orderly serial fashion while the fifth and sixth build mix the files up a lot and are clearly parallel :

```
$ head -n 40 build1.log
CHK include/linux/version.h
UPD include/linux/version.h
SYMLINK include/asm -> include/asm-i386
SPLIT include/linux/autoconf.h -> include/config/*
CC arch/i386/kernel/asm-offsets.s
GEN include/asm-i386/asm-offsets.h
HOSTCC scripts/kallsyms
HOSTCC scripts/conmakehash
CC init/main.o
CHK include/linux/compile.h
UPD include/linux/compile.h
CC init/version.o
CC init/do_mounts.o
LD init/mounts.o
CC init/initramfs.o
CC init/calibrate.o
LD init/built-in.o
HOSTCC usr/gen_init_cpio
CHK usr/initramfs_list
UPD usr/initramfs_list
CPIO usr/initramfs_data.cpio
GZIP usr/initramfs_data.cpio.gz
AS usr/initramfs_data.o
LD usr/built-in.o
CC arch/i386/kernel/process.o
CC arch/i386/kernel/semaphore.o
CC arch/i386/kernel/signal.o
AS arch/i386/kernel/entry.o
CC arch/i386/kernel/traps.o
CC arch/i386/kernel/irq.o
CC arch/i386/kernel/ptrace.o
CC arch/i386/kernel/time.o
CC arch/i386/kernel/ioport.o
CC arch/i386/kernel/ldt.o
CC arch/i386/kernel/setup.o
CC arch/i386/kernel/i8259.o
CC arch/i386/kernel/sys_i386.o
CC arch/i386/kernel/pci-dma.o
CC arch/i386/kernel/i386_ksyms.o
CC arch/i386/kernel/i387.o
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
$ head -n 40 build2.log
CHK include/linux/version.h
UPD include/linux/version.h
SYMLINK include/asm -> include/asm-i386
SPLIT include/linux/autoconf.h -> include/config/*
HOSTCC scripts/kallsyms
HOSTCC scripts/conmakehash
CC arch/i386/kernel/asm-offsets.s
GEN include/asm-i386/asm-offsets.h
CC init/main.o
CHK include/linux/compile.h
UPD include/linux/compile.h
CC init/do_mounts.o
CC init/initramfs.o
CC init/calibrate.o
CC init/version.o
LD init/mounts.o
LD init/built-in.o
HOSTCC usr/gen_init_cpio
CHK usr/initramfs_list
UPD usr/initramfs_list
CPIO usr/initramfs_data.cpio
GZIP usr/initramfs_data.cpio.gz
AS usr/initramfs_data.o
LD usr/built-in.o
CC arch/i386/kernel/process.o
CC arch/i386/kernel/semaphore.o
CC arch/i386/kernel/signal.o
AS arch/i386/kernel/entry.o
CC arch/i386/kernel/traps.o
CC arch/i386/kernel/irq.o
CC arch/i386/kernel/ptrace.o
CC arch/i386/kernel/time.o
CC arch/i386/kernel/ioport.o
CC arch/i386/kernel/ldt.o
CC arch/i386/kernel/setup.o
CC arch/i386/kernel/i8259.o
CC arch/i386/kernel/sys_i386.o
CC arch/i386/kernel/pci-dma.o
CC arch/i386/kernel/i386_ksyms.o
CC arch/i386/kernel/i387.o
```

```
$ head -n 40 build3.log
CHK include/linux/version.h
UPD include/linux/version.h
SYMLINK include/asm -> include/asm-i386
SPLIT include/linux/autoconf.h -> include/config/*
HOSTCC scripts/kallsyms
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
HOSTCC scripts/conmakehash
CC arch/i386/kernel/asm-offsets.s
GEN include/asm-i386/asm-offsets.h
CC init/main.o
CHK include/linux/compile.h
UPD include/linux/compile.h
CC init/do_mounts.o
CC init/initramfs.o
CC init/calibrate.o
CC init/version.o
LD init/mounts.o
LD init/built-in.o
HOSTCC usr/gen_init_cpio
CHK usr/initramfs_list
UPD usr/initramfs_list
CPIO usr/initramfs_data.cpio
GZIP usr/initramfs_data.cpio.gz
AS usr/initramfs_data.o
LD usr/built-in.o
CC arch/i386/kernel/process.o
CC arch/i386/kernel/semaphore.o
CC arch/i386/kernel/signal.o
AS arch/i386/kernel/entry.o
CC arch/i386/kernel/traps.o
CC arch/i386/kernel/irq.o
CC arch/i386/kernel/ptrace.o
CC arch/i386/kernel/time.o
CC arch/i386/kernel/ioport.o
CC arch/i386/kernel/ldt.o
CC arch/i386/kernel/setup.o
CC arch/i386/kernel/i8259.o
CC arch/i386/kernel/sys_i386.o
CC arch/i386/kernel/pci-dma.o
CC arch/i386/kernel/i386_ksyms.o
CC arch/i386/kernel/i387.o
```

```
$ head -n 40 build4.log
CHK include/linux/version.h
UPD include/linux/version.h
SYMLINK include/asm -> include/asm-i386
SPLIT include/linux/autoconf.h -> include/config/*
HOSTCC scripts/kallsyms
HOSTCC scripts/conmakehash
CC arch/i386/kernel/asm-offsets.s
GEN include/asm-i386/asm-offsets.h
CC init/main.o
CHK include/linux/compile.h
UPD include/linux/compile.h
CC init/do_mounts.o
CC init/initramfs.o
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
CC init/calibrate.o
CC init/version.o
LD init/mounts.o
LD init/built-in.o
HOSTCC usr/gen_init_cpio
CHK usr/initramfs_list
UPD usr/initramfs_list
CPIO usr/initramfs_data.cpio
GZIP usr/initramfs_data.cpio.gz
AS usr/initramfs_data.o
LD usr/built-in.o
CC arch/i386/kernel/process.o
CC arch/i386/kernel/semaphore.o
CC arch/i386/kernel/signal.o
AS arch/i386/kernel/entry.o
CC arch/i386/kernel/traps.o
CC arch/i386/kernel/irq.o
CC arch/i386/kernel/ptrace.o
CC arch/i386/kernel/time.o
CC arch/i386/kernel/ioport.o
CC arch/i386/kernel/ldt.o
CC arch/i386/kernel/setup.o
CC arch/i386/kernel/i8259.o
CC arch/i386/kernel/sys_i386.o
CC arch/i386/kernel/pci-dma.o
CC arch/i386/kernel/i386_ksyms.o
CC arch/i386/kernel/i387.o
```

```
$ head -n 40 build5.log
CHK include/linux/version.h
UPD include/linux/version.h
SYMLINK include/asm -> include/asm-i386
SPLIT include/linux/autoconf.h -> include/config/*
HOSTCC scripts/kallsyms
HOSTCC scripts/conmakehash
CC arch/i386/kernel/asm-offsets.s
GEN include/asm-i386/asm-offsets.h
CC init/main.o
CHK include/linux/compile.h
UPD include/linux/compile.h
CC init/do_mounts.o
HOSTCC usr/gen_init_cpio
CHK usr/initramfs_list
UPD usr/initramfs_list
CPIO usr/initramfs_data.cpio
GZIP usr/initramfs_data.cpio.gz
AS usr/initramfs_data.o
LD usr/built-in.o
CC arch/i386/kernel/process.o
CC arch/i386/kernel/semaphore.o
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
CC arch/i386/kernel/signal.o
CC init/initramfs.o
AS arch/i386/kernel/entry.o
CC arch/i386/kernel/traps.o
CC arch/i386/mm/init.o
CC init/calibrate.o
CC arch/i386/kernel/irq.o
CC arch/i386/mm/pgtable.o
CC init/version.o
CC arch/i386/mm/fault.o
CC arch/i386/kernel/ptrace.o
LD init/mounts.o
CC arch/i386/mm/ioremap.o
LD init/built-in.o
CC arch/i386/kernel/time.o
CC arch/i386/mm/extable.o
CC arch/i386/mach-default/setup.o
CC arch/i386/mm/pageattr.o
CC arch/i386/kernel/ioport.o
```

```
$ head -n 40 build6.log
CHK include/linux/version.h
UPD include/linux/version.h
SYMLINK include/asm -> include/asm-i386
SPLIT include/linux/autoconf.h -> include/config/*
HOSTCC scripts/kallsyms
HOSTCC scripts/conmakehash
CC arch/i386/kernel/asm-offsets.s
GEN include/asm-i386/asm-offsets.h
CC init/main.o
CHK include/linux/compile.h
HOSTCC usr/gen_init_cpio
CHK usr/initramfs_list
UPD usr/initramfs_list
CPIO usr/initramfs_data.cpio
GZIP usr/initramfs_data.cpio.gz
AS usr/initramfs_data.o
LD usr/built-in.o
CC init/do_mounts.o
UPD include/linux/compile.h
CC init/initramfs.o
CC init/calibrate.o
CC arch/i386/kernel/process.o
CC arch/i386/mm/init.o
CC arch/i386/kernel/semaphore.o
CC init/version.o
CC arch/i386/kernel/signal.o
LD init/mounts.o
LD init/built-in.o
CC arch/i386/mm/pgtable.o
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
CC arch/i386/mach-default/setup.o
CC arch/i386/mm/fault.o
AS arch/i386/kernel/entry.o
LD arch/i386/mach-default/built-in.o
CC arch/i386/kernel/traps.o
LD arch/i386/crypto/built-in.o
CC arch/i386/mm/ioremap.o
CC arch/i386/mm/extable.o
CC arch/i386/mm/pageattr.o
CC kernel/sched.o
CC arch/i386/kernel/irq.o
```

Finally here's some information about my system that may be relevant :

```
$ uname -a
Linux dragon 2.6.16-rc5-mm1 #1 SMP PREEMPT Tue Feb 28 19:36:09 CET
2006 i686 athlon-4 i386 GNU/Linux
```

```
$ scripts/ver_linux
If some fields are empty or look unusual you may have an old version.
Compare to the current minimal requirements in Documentation/Changes.
```

```
Linux dragon 2.6.16-rc5-mm1 #1 SMP PREEMPT Tue Feb 28 19:36:09 CET
2006 i686 athlon-4 i386 GNU/Linux
```

```
Gnu C 3.4.5
Gnu make 3.80
binutils 2.15.92.0.2
util-linux 2.12p
mount 2.12p
module-init-tools 3.1
e2fsprogs 1.38
reiserfsprogs 3.6.19
quota-tools 3.12.
PPP 2.4.4b1
nfs-utils 1.0.7
Linux C Library 2.3.6
Dynamic linker (ld) 2.3.6
Linux C++ Library 6.0.3
Procps 3.2.6
Net-tools 1.60
Kbd 1.12
Sh-utils 5.94
udev 064
Modules Loaded snd_seq_oss snd_seq_midi_event snd_seq
snd_pcm_oss snd_mixer_oss uhci_hcd usbcore snd_emu10k1 snd_rawmidi
snd_ac97_codec snd_ac97_bus snd_pcm snd_seq_device snd_timer
snd_page_alloc snd_util_mem snd_hwdep snd agpgart
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
$ cat /proc/cpuinfo
processor : 0
vendor_id : AuthenticAMD
cpu family : 15
model : 35
model name : AMD Athlon(tm) 64 X2 Dual Core Processor 4400+
stepping : 2
cpu MHz : 2200.209
cache size : 1024 KB
physical id : 0
siblings : 2
core id : 0
cpu cores : 2
fdiv_bug : no
hlt_bug : no
f00f_bug : no
coma_bug : no
fpu : yes
fpu_exception : yes
cpuid level : 1
wp : yes
flags : fpu vme de tsc msr pae mce cx8 apic sep mtrr pge mca
cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt
lm 3dnowext 3dnow pni lahf_lm cmp_legacy ts fid vid ttp
bogomips : 4401.74
```

```
processor : 1
vendor_id : AuthenticAMD
cpu family : 15
model : 35
model name : AMD Athlon(tm) 64 X2 Dual Core Processor 4400+
stepping : 2
cpu MHz : 2200.209
cache size : 1024 KB
physical id : 0
siblings : 2
core id : 1
cpu cores : 2
fdiv_bug : no
hlt_bug : no
f00f_bug : no
coma_bug : no
fpu : yes
fpu_exception : yes
cpuid level : 1
wp : yes
flags : fpu vme de tsc msr pae mce cx8 apic sep mtrr pge mca
cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt
lm 3dnowext 3dnow pni lahf_lm cmp_legacy ts fid vid ttp
bogomips : 4399.54
```

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

Odd sched behaviour; It takes 5 threads or more to load 2 CPU cores during kernel build

```
$ cat ./block/sda/queue/scheduler  
noop [anticipatory]
```

```
$ cat /etc/slackware-version  
Slackware 10.2.0
```

Do we have a scheduler problem?
An io-scheduler problem?
Is it "make" that's being difficult?
Is it something in the kernels Makefile that causes the build to
behave this way?
Could it be a bottleneck in my system somewhere?

Anyone got a clue?

--

Jesper Juhl <jesper.juhl@xxxxxxxx>
Don't top-post <http://www.catb.org/~esr/jargon/html/T/top-post.html>
Plain text mails only, please <http://www.expita.com/nomime.html>

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@xxxxxxxxxxxxxxxxx
More majordomo info at <http://vger.kernel.org/majordomo-info.html>
Please read the FAQ at <http://www.tux.org/lkml/>