

Re[4]: Modem + Network in Xircom cards, and maybe others

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/current/2004-05/0269.html>

From: Bruce Evans (*bde_at_zeta.org.au*)

Date: 05/06/04

Date: Thu, 6 May 2004 16:44:42 +1000 (EST)

To: Carlos Velasco <freebsd@newipnet.com>

On Wed, 5 May 2004, Carlos Velasco wrote:

> *On 06/05/2004 at 0:26 Bruce Evans wrote:*
>
> *>It seems like a patch for MFC since it has mfc's in it (what is a MFC?).*
>
> *MFC == Multi Function PCMCIA cards.*
> *FreeBSD has yet support for REAL MFC cards. However Xircom and maybe other*
> *cards don't work as REAL MFC.*
> *Right now only the last function (usually Network function) is the one that*
> *works in these cards with CURRENT.*
>
> *I'm patching it to activate both functions (usually modem/serial and*
> *network).*
> *Network is in xe driver, however I'm passing modem to sio.*
> *It works, however sio reports interrupt-level buffer overflows when I'm*
> *above 2400bps, losing characters, and here is where I'm lost.*
>
> *>However, you may need only this part of it. This part permits software*
> *>interrupts to be delayed by 38 ticks instead of the expected maximim*
> *>of 2 ticks. I keep forgetting to fix this. Scaling by hz is not quite*
> *>right, because hz may be set to values that are too small to work all*
> *>the time or even most of the time. There is a clamp to 128 (min), but*
> *>this is a bit small. E.g., with hz = 1000 and speed = 115200, the*
> *>original code in the above gives cp4ticks = 44 and ibufsize = 128. If*
> *>hz = 1000 actually worked, then the buffer would be drained every 1*
> *>msec and would never have more than 12 characters in it, but the*
> *>software interrupt latency is apparently sometimes >= 12 msec so the*
> *>buffer overflows. There are some mii Giant hogs which sometimes delay*
> *>timeout processing for that long or nearly so (each).*
>
> *Lost here... I think understand that problem is related to GIANT driver*
> *delaying this, right?*

It's more a general issue that that. sio's SWI handler[s] need to run

freebsd-current: Re[4]: Modem + Network in Xircom cards, and maybe others

often enough. For that, it needs to have high enough priority. Giant just delays things generally.

The broken interrupt priorities are easy to see in "ps laxw | sort -n +4" output. Note that the highest priorities are numerically lowest:

%%%

```
UID PID PPID CPU PRI NI VSZ RSS MWCHAN STAT TT TIME COMMAND
0 6 0 0 -84 0 0 12 actask IL ?? 0:00.00 (acpi_task0)
0 7 0 0 -84 0 0 12 actask IL ?? 0:00.00 (acpi_task1)
0 8 0 0 -84 0 0 12 actask IL ?? 0:00.00 (acpi_task2)
```

[acpi tasks with high priority above. It's not clear that acpi tasks should have the same high priority as clock interrupt handlers or any other hardware or software interrupt handler. Interrupts generally need to be able to preempt tasks.]

```
0 12 0 0 -84 0 0 12 - WL ?? 0:00.00 (irq0: clk)
0 19 0 0 -84 0 0 12 - WL ?? 0:00.00 (irq8: rtc)
```

[Bogus ithreads for clock interrupt handlers above. Clock interrupt handlers are fast, so these threads are never used. Since they are fast, they effectively have higher than the highest priority (sic), so they can interrupt the acpi tasks. Perhaps acpi depends on this.]

```
0 16 0 0 -68 0 0 12 - WL ?? 0:00.07 (irq5: fxp0)
0 21 0 0 -68 0 0 12 - WL ?? 0:00.00 (irq10: bge0)
0 17 0 0 -64 0 0 12 - WL ?? 0:00.00 (irq6: fdc0)
0 25 0 0 -64 0 0 12 - WL ?? 0:00.01 (irq14: ata0)
0 26 0 0 -64 0 0 12 - WL ?? 0:00.00 (irq15: ata1)
0 13 0 0 -60 0 0 12 - WL ?? 0:00.00 (irq1: atkbd0)
```

[Normal ithreads with correct priorities above.]

```
0 14 0 0 -60 0 0 12 - WL ?? 0:00.00 (irq3: sio1)
0 15 0 0 -60 0 0 12 - WL ?? 0:00.00 (irq4: sio0)
```

[Bogus ithreads with wrong priorities for serial drivers above. The interrupt handlers are fast, so these threads are never used. If they were used, then their low priority would cause problems.]

```
0 18 0 0 -60 0 0 12 - WL ?? 0:00.00 (irq7: ppc0)
```

[Another normal ithread above. Its priority is adequate but not quite right. This is for the parallel port, and its interrupt pretends to be for a (slow) tty, but the parallel port is not quite a slow tty and a higher priority interrupt would be better. Hopefully the priority increases when the parallel port is used for PLIP.]

```
0 22 0 0 -60 0 0 12 - WL ?? 0:00.00 (irq11: cy0)
```

freebsd-current: Re[4]: Modem + Network in Xircom cards, and maybe others

[Another bogus ithread for a serial driver above (bogus as above).]

0 20 0 0 -52 0 0 12 - WL ?? 0:00.00 (irq9: acpi0)

[Normal ithread with a dubious priority above. Using the lowest priority for the acpi interrupt doesn't seem to go with using the highest priority for acpi tasks.]

0 27 0 0 -48 0 0 12 - WL ?? 0:00.06 (swi8: tty:cy+ clock)

[I think this is supposed to be the low priority softclock ithread (the "slow" cy and sio SWIs attach to it and misname themselves as tty:cy and tty:sio instead of clk:cy and clk:sio). It actually has `_highest_` priority among SWIs, so the problem is sort of the opposite of what I thought, but mostly worse. The "slow" cy and sio SWIs actually have the same bogus high priority, so they don't compete with other SWIs. However, they compete with softclock, and all other SWIs have lower priority than softclock so they don't even compete with it. This is the reverse of what is supposed to happen. I think softclock starts with the correct low priority, but its priority gets clobbered when the cy and sio SWIs attach to it.

0 37 0 0 -48 0 0 12 - WL ?? 0:00.00 (swi0: tty:cy+)

[I think this is the "fast" cy and sio SWI. Verbose names which get truncated complicate debugging.]

0 29 0 0 -44 0 0 12 - WL ?? 0:00.02 (swi1: net)

0 33 0 0 -40 0 0 12 - WL ?? 0:00.00 (swi2: camnet)

0 34 0 0 -36 0 0 12 - WL ?? 0:00.00 (swi3: cambio)

0 28 0 0 -32 0 0 12 - WL ?? 0:00.00 (swi4: vm)

[Finally some examples of SWIs with their indented priorities above.]

0 31 0 0 -28 0 0 12 - WL ?? 0:00.00 (swi5:+)

0 36 0 0 -24 0 0 12 - WL ?? 0:00.00 (swi6:+)

[These seem to be unused space wasters for SWI_TQ_FAST and SWI_TQ_GIANT.]

0 23 0 0 -21 0 0 12 - WL ?? 0:00.00 (irq12:)

0 24 0 0 -21 0 0 12 - WL ?? 0:00.00 (irq13:)

[Bogus ithreads with weird priorities for unused hardware interrupts above. How did their priorities get to be in the middle of SWI priorities?]

0 32 0 0 -20 0 0 12 - WL ?? 0:00.00 (swi7: acpitaskq)

0 35 0 0 -20 0 0 12 - WL ?? 0:00.00 (swi7: task queue)

[Nearly lowest priority SWIs above. The priority seems a little low for acpi, as above. It's not clear whether the generic SWI task queue

freebsd-current: Re[4]: Modem + Network in Xircom cards, and maybe others

should have lowest, highest or average SWI priority.

[softclock is suppose to be here with lowest SWI priority -16.]
%%%

> *I believe this patch to sio.c is only a temporary solution that should be removed when GIANT dissappears in most drivers, am I right?*

No, it (the cp4ticks one) is a more general patch, though it is temporary because it is not in its final form. Best-case interrupt latency cannot be guaranteed for SWIs, and even a saftey margin of a factor of 4 is not enough even without the Giant and priority bugs, since it doesn't scale right when "hz" is configured to large values, and configuring "hz" to too-large values is now encouraged by DEVICE_POLLING.

> *>sio's software interrupt[s] should have a priority higher than timeouts, > >but sio now has 2 software interrupts with one of them having the same > >low priority as timeouts. This priority should work (in fact, old > >versions of sio just use timeouts), but in RELENG_4 the priority is > >much higher than that as a side effect of having only only the correct > >number of SWIs (1). RELENG_4 may now depend on this.*
>
> *Sorry, totally lost here.*
> *Maybe problem is here?*
> *sio4: <Xircom CreditCard Ethernet 10/100 + Modem 56> at port 0x2e8-0x2ef*
> *irq 11 function 0 config 39 on pccard0*
> *sio4: type 16550A*
> *sio4: unable to activate interrupt in fast mode - using normal mode*
>
> *Interrupt is in normal mode that has not a high priority?*

That;s a different problem and not the one here. Hardware interrupts must be working well enough or else you would get silo overflows instead of interrupt-level buffer overflows.

> *I don't know why sio activate interrupt in normal mode and can't do it in > fast, though.*

It is because fast interrupts are not supported at the pccard level or on the same irq as a non-fast interrupt. Multi-function pccards probably cause both problems. The problem corresponding to the first for interrupts layered under puc is hacked around by forcing fast interrupts using PUC_FASTINTR. This only works if all interrupts under puc are fast. pccard is more complicated and handles more devices, so a similar hack would work less well.

Bruce

freebsd-current@freebsd.org mailing list

<http://lists.freebsd.org/mailman/listinfo/freebsd-current>

To unsubscribe, send any mail to "freebsd-current-unsubscribe@freebsd.org"

Re[4]: Modem + Network in Xircom cards, and maybe others