

Re: timer counter chip access mystery]

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/performance/2003-07/0038.html>

From: Petri Helenius (*pete_at_he.iki.fi*)

Date: 07/17/03

To: "Jin Guojun [DSD]" <j_guojun@lbl.gov>

Date: Thu, 17 Jul 2003 09:50:55 +0300

I thought you have to use ACPI-safe if you have broken hardware and otherwise you'll have ACPI-fast. (which my boxes default to)

Pete

----- Original Message -----

From: "Jin Guojun [DSD]" <j_guojun@lbl.gov>

To: "Petri Helenius" <pete@he.iki.fi>

Cc: "Steven Hartland" <killing@barrysworld.com>; <freebsd-performance@freebsd.org>

Sent: Thursday, July 17, 2003 2:22 AM

Subject: Re: timer counter chip access mystery]

> 5.1 defaults to use ACPI-safe, which calls read_counter() at least
> three times. The total cost is 3098 ns. So, it means that read_count()
> cost is about 1000 ns, which is close to what the Linux does.

>

> The rest question is how much error will be if not to use
> ACPI-safe? i.e., sysctl kern.timecounter.hardware=ACPI?

>

> -Jin

>

> /*

> * Fetch current time value from reliable hardware.

> */

> static unsigned

> acpi_timer_get_timecount(struct timecounter *tc)

> {

> return (read_counter());

> }

>

> /*

> * Fetch current time value from hardware that may not correctly

> * latch the counter.

> */

> static unsigned

> acpi_timer_get_timecount_safe(struct timecounter *tc)

> {

freebsd-performance: Re: timer counter chip access mystery]

```
> unsigned u1, u2, u3;
>
> u2 = read_counter();
> u3 = read_counter();
> do {
> u1 = u2;
> u2 = u3;
> u3 = read_counter();
> } while (u1 > u2 || u2 > u3 || (u3 - u1) > 15);
> return (u2);
> }
>
> Petri Helenius wrote:
>
> > 5.x defaults to ACPI or TSC timers. So your wish has been transported
> > by a timemachine into the past.
> >
> > With 4.x you have to set it manually.
> >
> > Pete
> >
> > ----- Original Message -----
> > From: "Steven Hartland" <killing@barrysworld.com>
> > To: "Petri Helenius" <pete@he.iki.fi>; "Jin Guojun [DSD]" <j_guojun@lbl.gov>;
<freebsd-performance@freebsd.org>
> > Sent: Thursday, July 17, 2003 12:57 AM
> > Subject: Re: timer counter chip access mystery]
> >
> > > If this could be changed / fixed it would have a marked performance increase
> > > for all game server code which use this function heavily from what I've seen
> > > and would explain the sometimes huge performance differential between
> > > FreeBSD and Linux.
> > >
> > > Steve / K
> > > ----- Original Message -----
> > > From: "Petri Helenius" <pete@he.iki.fi>
> > > To: "Jin Guojun [DSD]" <j_guojun@lbl.gov>; <freebsd-performance@freebsd.org>
> > > Sent: Wednesday, July 16, 2003 10:44 PM
> > > Subject: Re: timer counter chip access mystery]
> > >
> > >
> > > > This access happens over ISA bus and thus happens at the speed the bus
> > > > operates at. Use TSC or ACPI for faster gettimeofday.
> > > >
> > > > Pete
> > > >
> > > >
```

freebsd-performance@freebsd.org mailing list

Re: timer counter chip access mystery]

freebsd-performance: Re: timer counter chip access mystery]

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

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