

## Re: [RFC] [PATCH 0/3] ioat: DMA engine support

**Source:** <http://linux.derkeiler.com/Mailing-Lists/Kernel/2005-11/8323.html>

---

**From:** Benjamin LaHaise (*bcr1\_at\_kvack.org*)

**Date:** 11/24/05

Date: Wed, 23 Nov 2005 19:17:01 -0500

To: Andi Kleen <ak@suse.de>

On Wed, Nov 23, 2005 at 11:30:08PM +0100, Andi Kleen wrote:

- > *The main problem I see is that it'll likely only pay off when you can keep*
- > *the queue of copies long (to amortize the cost of*
- > *talking to an external chip). At least for the standard recvmg*
- > *skb->user space, user space-> skb cases these queues are*
- > *likely short in most cases. That's because most applications*
- > *do relatively small recvmg or sendmsg.*

Don't forget that there are benefits of not polluting the cache with the traffic for the incoming skbs.

- > *Longer term the right way to handle this would be likely to use*
- > *POSIX AIO on sockets. With that interface it would be easier*
- > *to keep long queues of data in flight, which would be best for*
- > *the DMA engine.*

Yes, that's something I'd like to try soon.

- > *But it's not clear it's a good idea: a lot of these applications prefer to*
- > *have the target in cache. And IOAT will force it out of cache.*

In the I/O AT case it might make sense to do a few prefetch()es of the userland data on the return-to-userspace code path. Similarly, we should make sure that network drivers prefetch the header at the earliest possible time, too.

- > *I remember the registers in the Amiga Blitter for this and I'm*
- > *still scared... Maybe it's better to keep it simple.*

\*grin\* but you could use it for such cool tasks as MFM en/decoding! =-)

-ben

--

"Time is what keeps everything from happening all at once." -- John Wheeler  
Don't Email: <dont@kvack.org>.

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@vger.kernel.org

Re: [RFC] [PATCH 0/3] ioat: DMA engine support

Linux-Kernel: Re: [RFC] [PATCH 0/3] ioat: DMA engine support

More majordomo info at <http://vger.kernel.org/majordomo-info.html>  
Please read the FAQ at <http://www.tux.org/lkml/>