

API change for bus_dma

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/arch/2003-06/0277.html>

From: Scott Long (scott_long_at_btc.adaptecc.com)

Date: 06/21/03

Date: Fri, 20 Jun 2003 20:21:35 -0600

To: freebsd-arch@freebsd.org

All,

As I work towards locking down storage drivers, I'm also preening their use of busdma. A common theme among them is a misuse of bus_dmamap_load() and the associated callback mechanism. For most, the consequence is harmless as long as the card can support the amount of physical memory in the system (systems with IOMMU's not withstanding). However, in cases such as PAE where busdma might have to use bounce buffers, most drivers don't handle the possibility of bus_dmamap_load() returning EINPROGRESS. The consequence of this is twofold: bus_dmamap_load() returns without the callback being called, but the driver doesn't detect this and merrily goes on its way. Later on the callback does get called, and any state that was shared with it gets corrupted. This is a problem even for drivers that are under Giant.

The solution for this is mostly a mechanical cut-n-paste of the code dealing with the callback. However, locking down the drivers presents a new problem with the callback. Since the callback can be called asynchronously from an SWI, it needs some way to synchronize with the driver. Adding code to each callback to conditionally grab the driver mutex incurs a penalty (albiet small) and requires more effort. The better solution is to export the driver mutex to busdma and have the SWI that runs the callback lock the mutex before calling the callback.

This requires adding a 'struct mtx *' argument to bus_dma_tag_create() to hold the mutex to be exported. For drivers that are under Giant and/or decide not to use this functionality, passing NULL for this argument is accepted. Therefore, the change is fairly low-impact, though it incurs an API change. Since locking the peripheral drivers is a major goal for 5.2 and 5-STABLE, it is time to bite the bullet and do this now. A few 3rd-party drivers stand to be affected, and hopefully their maintainers will react accordingly.

Comments?

Scott

freebsd-arch: API change for bus_dma

freebsd-arch@freebsd.org mailing list

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

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