

Re: [patch/rfc 2.6.19-rc5] arch-neutral GPIO calls

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-11/msg03449.html>

- *From:* David Brownell <david-b@xxxxxxxxxxxxx>
 - *Date:* Sat, 11 Nov 2006 19:04:34 -0800
-

On Saturday 11 November 2006 5:27 pm, H. Peter Anvin wrote:

If this is done, I think it's essential that a "high-level" API (one that supports message-based GPIO) is provided at the same time.

It's not a "high level" vs "low level" issue, though. There are dozens to hundreds of drivers that know they're using "real GPIOs", and which **need** to do that. For example, because the signals must be accessed while holding a spinlock, or in_irq().

I'd like to see some "message based GPIO" API soonish, but can't buy the notion that lack of one should be a blocking issue. In fact, if you go reread that LKML thread I referenced originally, you'll notice that a key "why this can't fly" issue was that it didn't have support for irq/spinlock safe GPIOs ... which make up by far the majority of current GPIO users in Linux.

Lack of arch-neutral "real GPIO" calls is more or less a blocking issue, given the merge of the AVR32 architecture ... unless you'd like to argue it's a good thing to make drivers for integrated controllers be (otherwise needlessly) be arch-specific? One current example is the "atmel_spi" driver, sharable between AVR32 and AT91; Atmel reused a lot of the controllers from its AT91 ARM series when they designed the AVR32 SOCs, and it doesn't make sense to have both AT91 and AVR32 versions of the same drivers.

The "high-level" API should be able to address the GPIOs addressed by the low-level API. What we do **not** want is a bunch of stuff using the low-level API when the high-level API would work.

If there were many (any?) drivers that didn't need to care, I might be tempted to agree. But there aren't ...

Re: [patch/rfc 2.6.19-rc5] arch-neutral GPIO calls

Another way to look at it is to notice how many spinlock-safe GPIO APIs are in use *right now* on ARM ... that's on the order of a dozen, which is a kind of existence proof that switching to the API in the patch I sent would be a useful cleanup. (And switching would be mostly just syntax tweaks ... the primitives are all the same, except for error checking; their names are just spelled differently.)

Even if there were only four platform specific drivers per architecture which use GPIOs, that'd still