

viapropm doesnt like sys/dev/pci.c rev 1.214

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/current/2003-05/1024.html>

From: David P. Reese Jr. (*daver_at_gomerbud.com*)

Date: 05/31/03

Date: Sat, 31 May 2003 14:06:00 -0700

To: current@freebsd.org

In rev 1.214 of sys/dev/pci/pci.c, we have started checking if a pci_set_command_bit() was successful with a subsequent PCI_READ_CONFIG and comparing the results. For some odd reason, this doesnt work when my viapropm tries to attach. Allocating its port resources fails in pci_enable_io_method(). The code in question is:

sys/dev/pci/pci.c:pci_enable_io_method()

[snip]

```
pci_set_command_bit(dev, child, bit);
command = PCI_READ_CONFIG(dev, child, PCIR_COMMAND, 2);
if (command & bit)
    return (0);
device_printf(child, "failed to enable %s mapping!\n", error);
return (ENXIO);
```

What is annoying is that by changing the last line to return a zero instead of ENXIO, the viapropm successfully attaches. The smbus interface even works. Everything else that tries to claim port resources succeeds.

Is it my chipset's fault for not reading back the correct register value? The board is a SOYO K7VTAPRO-2AA6. What other info would be helpful in this situation?

```
viapropm0@pci0:7:4: class=0x068000 card=0x30571106 chip=0x30571106 rev=0x40 hdr=0x00
  vendor = 'VIA Technologies Inc'
  device = 'VT82C686A/B ACPI Power Management Controller'
  class = bridge
  subclass = PCI-unknown
```

--

David P. Reese Jr.

daver@gomerbud.com

It can be argued that returning a NULL pointer when asked to allocate zero bytes is a silly response to a silly question.

-- FreeBSD manual page for malloc(3)

freebsd-current@freebsd.org mailing list

viapropm doesnt like sys/dev/pci.c rev 1.214

freebsd-current: viapropm doesnt like sys/dev/pci.c rev 1.214

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

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