

Re: PCI bus numbering and orphaned devices

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/current/2003-06/0691.html>

From: Bernd Walter (*ticso_at_cicely12.cicely.de*)

Date: 06/11/03

Date: Wed, 11 Jun 2003 01:16:50 +0200

To: "M. Warner Losh" <imp@bsdimp.com>, ticso@cicely.de, ticso@cicely12.cicely.de, freebsd-current

On Tue, Jun 10, 2003 at 03:34:36PM -0700, John-Mark Gurney wrote:

> M. Warner Losh wrote this message on Tue, Jun 10, 2003 at 08:27 -0600:

> > > > *hdrtype = REG(PCIR_HEADERTYPE, 1);*

> > > *This needs to be tested on that given hardware.*

> > > *I don't know if REG will work as expected because it asks function 0,*

> > > *which is disabled.*

> > :

> > : *I've reread John-Mark's last mail about the readable registers.*

> > : *So - yes it should work.*

> >

> > *That's what inspired me. Also, I'd expected that we'd need some kind*

> > *of tweaking to make it actually compile and be neat.*

>

> *Ok, attached is a patched I tried, but sad to say, this doesn't work*

> *out to well. I added a printf before and after the REG statement, and*

> *a boot with the kernel give this output:*

> *found-> vendor=0x108e, dev=0x5000, revid=0x13*

> *bus=0, slot=1, func=1*

> *class=06-04-00, hdrtype=0x01, mfdev=1*

> *cmdreg=0x0147, statreg=0x02a0, cachelsz=16 (dwords)*

> *lattimer=0x50 (2400 ns), mingnt=0x02 (500 ns), maxlat=0x00 (0 ns)*

> *about to read HEADERTYPE*

> *panic: trap: data access error*

> *cpuid = 0;*

> *Uptime: 1s*

> *panic: Assertion mtx_unowned(m) failed at ../../kern/kern_mutex.c:956*

> *cpuid = 0;*

> *Uptime: 1s*

> *panic: Assertion mtx_unowned(m) failed at ../../kern/kern_mutex.c:956*

> *cpuid = 0;*

> *Uptime: 1s*

>

> *the last three lines repeat for a while, but this is because of:*

> *psycho_read_config(...)*

> *[...]*

> */**

> ** The psycho bridge does not tolerate accesses to unconfigured PCI*

freebsd-current: Re: PCI bus numbering and orphaned devices

> * devices' or function's config space, so look up the device in the
> * firmware device tree first, and if it is not present, return a value
> * that will make the detection code think that there is no device here.
> * This is ugly...
> */
> if (reg == 0 && ofw_pci_find_node(bus, slot, func) == 0)
> return (0xffffffff);
>
> Which obviously will fail if reg != 0 which we try to do when reading
> the PCIR_HEADERTYPE..
>
> So, the question is, does other arch's do something nasty like this
> too? Should I change the check to just do ofw_pci_find_node? Is this
> why pciconf -r is returning 0xffffffff when reading the ebus and firewire
> parts of the SME2300BGA? Simply because it isn't in the ofw tree?

Possible – in fact I was very surprised that a disabled device was not readable on some registers.

We have a similar situation on alpha, where we get traps for reading non available devices.

It's handled in that we tell the system to expect traps before accessing registers.

This is done by reading with the badaddr function, which sets a flag for our trap handler so it can continue in case the device doesn't exist.

badaddr() returns a flags which tells if reading was successful.

> I don't have any data sheets or the PCI spec, so making heads or tails
> of this is going be hard.

It's OK to get errors when reading locations that aren't available.

Some chipsets never trap, others only trap for type2 access (behind Bridges) and some always trap.

--

B.Walter
ticso@bwct.de

BWCT

<http://www.bwct.de>
info@bwct.de

freebsd-current@freebsd.org mailing list

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

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