

Re: PCI bus enumeration differences between Geode and ARM

Source:

<http://www.tech-archive.net/Archive/WindowsCE/microsoft.public.windowsce.platbuilder/2004-05/0652.html>

From: Ten (*anonymous_at_tentechologies.com*)

Date: 05/18/04

Date: Tue, 18 May 2004 17:08:48 -0400

K.S.,

Thanks for the information. NoConfig=1 in my platform. I found this a few days ago but was not eager to change it because PCIconfig.c in my platform hard codes values to some of the PCI devices (mem and IO). I thought this might break the whole platform so I decided to research the problem a little more before trying the change.

I'll give the change a try now that I have collected some more information.

Much thanks!

Nick.

"K. S. Huang" <ks_huang@AlphaNetworks.com_remove.this> wrote in message news:%23EpWFiJPEHA.268@TK2MSFTNGP11.phx.gbl...

> *PCI-based device need to be configured and dispatch resources before they can be used.*

> *On most x86 system, device enumeration and resource dispatch is done by the BIOS.*

> *But there is no BIOS in ARM-based system, so the resource dispatch will be handled by the PCIBUS.DLL or EBoot*

> *if the EBoot dose not dispatch the resource, it is resanable that in OEMInit will get nothing.*

>

> *Event after booting up, the PCibus.dll might not do the resource dispatching*

> *if the*

>

> *[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\PCI]*

> *"NoConfig"=dword:1*

>

> *is setting to one, to enable resource dispatching by PCIBUS.DLL*

> you 'll need to set the
> [HKEY_LOCAL_MACHINE\Drivers\BuiltIn\PCI]
> "NoConfig"=dword:0
>
> And there might be another possibility, even the PCIBUS.DLL is trying to
> dispatch the resource
> but the
>
> MemBase, MemLen, IoBase and IoLen in
> [HKEY_LOCAL_MACHINE\Drivers\BuiltIn\PCI]
> dose not setting properly, the PCIBUS.DLL will think that there is NOT
> enough resource, so that the dispatching will still be given up!
>
>
> "Ten" <anonymous@tenttechnologies.com> ¼¶¼g©ó¶I¥ó·s»D
> :#n\$RHdFPEHA.3328@TK2MSFTNGP09.phx.gbl...
> > Hi,
> >
> > I have a Geode based system + a custom PCI device that works like a
champ.
> > I'm now creating a new platform for an ARM based SBC. My PCI driver
> > compiles and loads on the ARM. During XXX_Init I'm using
> > DDKReg_GetWindowInfo to obtain the Memory and IO maps for the PCI
device.
> > Under the Geode, I get valid IO and Memory windows. On the ARM, I get
> > empty
> > structures returned.
> >
> > I noticed that my command register is 0. On the Geode it contains 7.
Am
> > I
> > supposed to write this value from the driver (in the call defined by
> > ConfigEntry, similar to the sample in gdcfg.c) or is this supposed to be
> > written by the bus enumerator? If me, why does the bus enumerator not
> > handle my device? Is there another way to configure the device?
> >
> > Here's the dump of PCI Enum on my ARM platform. My PCI device is last
in
> > the list, 0/18/0.
> >
> > Bus/Device/Function = 0/1/0, Host / PCI Bridge
> > VendorID = 0x1283, DeviceID = 0x8152, Command = 0x0006, Status =
0x2200
> > RevisionID = 0x20, ProgIf = 0x00, SubClass = 0x00, BaseClass = 0x06
> > CacheLineSize = 0x00, LatencyTimer = 0x00, HeaderType = 0x80, BIST =
> > 0x00
> > BaseAddresses:
> > 0xA0000008 0xBCF00001 0x00000000 0x00000000 0x00000000
0x00000000
> > CIS = 0x00000000, SubVendorID = 0x0000, SubSystemID = 0x0000
> > ROMBaseAddress = 0x00000000

```
> > InterruptLine = 0x00, InterruptPin = 0x00, MinGrant = 0x00, MaxLatency
=
> > 0x00
> >
> > Bus/Device/Function = 0/15/0, Ethernet Network Controller
> > VendorID = 0x10EC, DeviceID = 0x8139, Command = 0x0007, Status =
0x0290
> > RevisionID = 0x10, ProgIf = 0x00, SubClass = 0x00, BaseClass = 0x02
> > CacheLineSize = 0x00, LatencyTimer = 0x00, HeaderType = 0x00, BIST =
0x00
> > BaseAddresses:
> > 0xBCE13101 0xB9001000 0x00000000 0x00000000 0x00000000
0x00000000
> > CIS = 0x00000000, SubVendorID = 0x10EC, SubSystemID = 0x8139
> > ROMBaseAddress = 0x00000000
> > InterruptLine = 0x24, InterruptPin = 0x01, MinGrant = 0x20, MaxLatency
=
> > 0x40
> >
> > Bus/Device/Function = 0/16/0, Ethernet Network Controller
> > VendorID = 0x10EC, DeviceID = 0x8139, Command = 0x0007, Status =
0x0290
> > RevisionID = 0x10, ProgIf = 0x00, SubClass = 0x00, BaseClass = 0x02
> > CacheLineSize = 0x00, LatencyTimer = 0x00, HeaderType = 0x00, BIST =
0x00
> > BaseAddresses:
> > 0xBCE13001 0xB9000000 0x00000000 0x00000000 0x00000000
0x00000000
> > CIS = 0x00000000, SubVendorID = 0x10EC, SubSystemID = 0x8139
> > ROMBaseAddress = 0x00000000
> > InterruptLine = 0x22, InterruptPin = 0x01, MinGrant = 0x20, MaxLatency
=
> > 0x40
> >
> > Bus/Device/Function = 0/18/0, Other Bridge
> > VendorID = 0x10B5, DeviceID = 0x9054, Command = 0x0000, Status =
0x0290
> > RevisionID = 0x0C, ProgIf = 0x00, SubClass = 0x80, BaseClass = 0x06
> > CacheLineSize = 0x00, LatencyTimer = 0x00, HeaderType = 0x00, BIST =
0x00
> > BaseAddresses:
> > 0x00000000 0x00000001 0x00000000 0x00000000 0x00000000
0x00000000
> > CIS = 0x00000000, SubVendorID = 0x10B5, SubSystemID = 0x9054
> > ROMBaseAddress = 0x00000000
> > InterruptLine = 0x00, InterruptPin = 0x01, MinGrant = 0x00, MaxLatency
=
> > 0x00
> >
> >
> > Thanks!
```

microsoft.public.windowsce.platbuilder: Re: PCI bus enumeration differences between Geode and ARM

> > *Nick.*

> >

> >

> >

> >

>

>