

Application Note: PCI32/Host PC Compatibility Issues

DATE: January 14, 1999
TO: PCI32 Users
FROM: Technical Staff
RE: Compatibility issues between the PCI32 and certain host systems
CC:

Innovative Integration has discovered two compatibility issues between the PCI32 and certain host computer systems which can keep the PCI32 from operating correctly. The symptoms and their causes are described below, along with fixes.

Problem 1: Improper host read caching of dual port RAM memory

Certain host system BIOSes appear to determine that the dual port memory of an installed PCI32 is cacheable ROM memory and perform a single read of the dual port RAM at system powerup, then use that cached information for all subsequent reads. This caching interferes with proper communication between the host system and the PCI32 since after powerup host software never has access to the current state of the information stored in the dual port RAM. The caching occurs despite disabling the BIOS cache feature in the setup program.

Unfortunately, Innovative does not have a comprehensive list of BIOSes which perform this improper caching. There is a test which will allow the individual user to determine if the caching is taking place. First, power the host system up and start Windows. Start the VIEWER application from the Zuma Toolset (C:\PCI32CC directory) and type the following commands:

```
0 OPEN [enter]
```

```
0 20 DPDUMP [enter]
```

The DPDUMP command will produce a dump in table form of the first 20 32-bit words of the dual port memory as viewed by the host. Write a test value to dual port RAM using the following command:

```
12345678 0 DP! [enter]
```

The DP! command will write the hexadecimal value 0x12345678 to location 0 of the dual port RAM. Next, dump the dual port again with the DPUMP command:

```
0 20 DPDUMP [enter]
```

Make note of the zeroth location's value. If the value changed to 0x12345678 then the host system is not caching the dual port RAM memory and this problem report does not apply.

If the value in the zeroth location did NOT change as a result of the DP! command, then we need to perform one additional test to determine if the BIOS is caching the dual port RAM memory. Make a note of the

values stored in the first three or four locations in the dual port RAM, then close the VIEWER program by typing the BYE command. Reset the host system by selecting Shutdown from the Windows Start button and selecting the Restart radio button in the dialog box, then clicking OK. Allow Windows to restart, then once again start VIEWER and type the following commands:

0 OPEN [enter]

0 20 DPDUMP [enter]

If the zeroth location now reads as 0x12345678, then the host system IS improperly caching the dual port RAM memory. The VIEWER program is requesting memory reads from the dual port RAM area, but the host chipset does not actually perform the reads when requested and instead responds with data read when the system was restarted. After the DP! command was performed, the data was written correctly (writes are not affected by the BIOS caching bug: only reads), but VIEWER couldn't see the affect of the read since it was receiving data read by the BIOS BEFORE the DP! write actually occurred.

Problem Solution: Innovative has found that a small modification to the PCI32's PCI plug-and-play configuration information will alleviate this problem. By changing the memory allocation request from the PCI32 to the host to force the host to map the dual port RAM memory high (i.e. above the 1M memory address point) the BIOS will no longer cache reads from the dual port RAM. The problem apparently stems from the PCI32's request that its dual port RAM be mapped into low memory (nominally the 0xC8000-0xEFFFF range), which the BIOS apparently considers cacheable. High memory mapping avoids the problem.

One side effect of this fix is that the dual port RAM is mapped into 32-bit addressable memory, which makes the PCI32 incompatible with older 16-bit DOS software. Innovative no longer produces DOS host software for the PCI32 so this is not an issue with Innovative software, but individual users may still be running DOS-based applications and they should be aware of this fact. The original configuration ROM information is available for users with 16-bit DOS compatibility requirements: contact Innovative for information.

Fixing the problem involves removing and reprogramming the PCI configuration ROM device on the PCI32. Users must return their hardware for the modification. Contact Innovative for a return material authorization to have the PCI32 configuration ROM upgraded to the current revision.

The ROM fix for this problem was placed into production in September 1998. All boards produced after this date should not exhibit this problem.

Problem 2: Intel 440LX/BX chipset incompatibility

Due to a timing difference between the Intel 440LX and 440BX Pentium II motherboard chipsets and previous chipset designs, the PCI32 may act erratically in Pentium II host systems using this chipset. Attempts to start TERMINAL will occasionally produce "Talker didn't start" errors, and program downloads from TERMINAL will generally fail even after the program starts communicating with Talker correctly.

Problem Solution: A modification to the PCI32's onboard logic fixes the problem. Contact Innovative for a return material authorization to have PCI32 hardware upgraded to the current logic revision.

The logic fix for this problem was placed into production in December 1998. All boards produced after this date should not exhibit this problem in Pentium II systems.