Technical Note TN1135

Dealing with PCI Expansion Chassis Problems

CONTENTS

Defining the Problem

Problem Scenario

Why This Hasn't Been a Problem Before

Coping with Expansion Chassis Issues

Additional Notes & Comments

Summary

References

Downloadables

When designing PCI cards, it's easy to overlook the fact that the card may be plugged into an expansion chassis. Therefore, it is possible for compatibility issues to arise if you do not allow for certain conditions. This note provides information on many of these conditions, and explains how you might avoid some problems.

A PCI expansion chassis can present several problems to the PCI card designer. Variations of interrupt latency, propagation delays, intercard compatibility, and other problems can be introduced by the sub-bridge architecture inherent to this design.

This note is directed at developers of PCI cards so that they will understand the compatibility issues that might arise if their product is used in a PCI expansion chassis. There are both hardware and software issues that need to be considered.

Updated: [Aug 17 1998]

Defining the Problem

You may experience system crashes and/or the inability to access memory or I/O space on PCI cards that are plugged into a PCI expansion chassis. The same cards may work fine plugged into a direct PCI host slot.

Problem Scenario

The following lists possible causes for PCI expansion chassis problems:

- Interrupt hardware sharing: all interrupts funnel into a single hardware interrupt line for an entire PCI

expansion chassis;
- Multiple vendors producing simultaneous interrupts;
- Multiple interrupts from single multifunction cards;
- Initialization problems; or
- Interrupt propagation delays.

Back to top

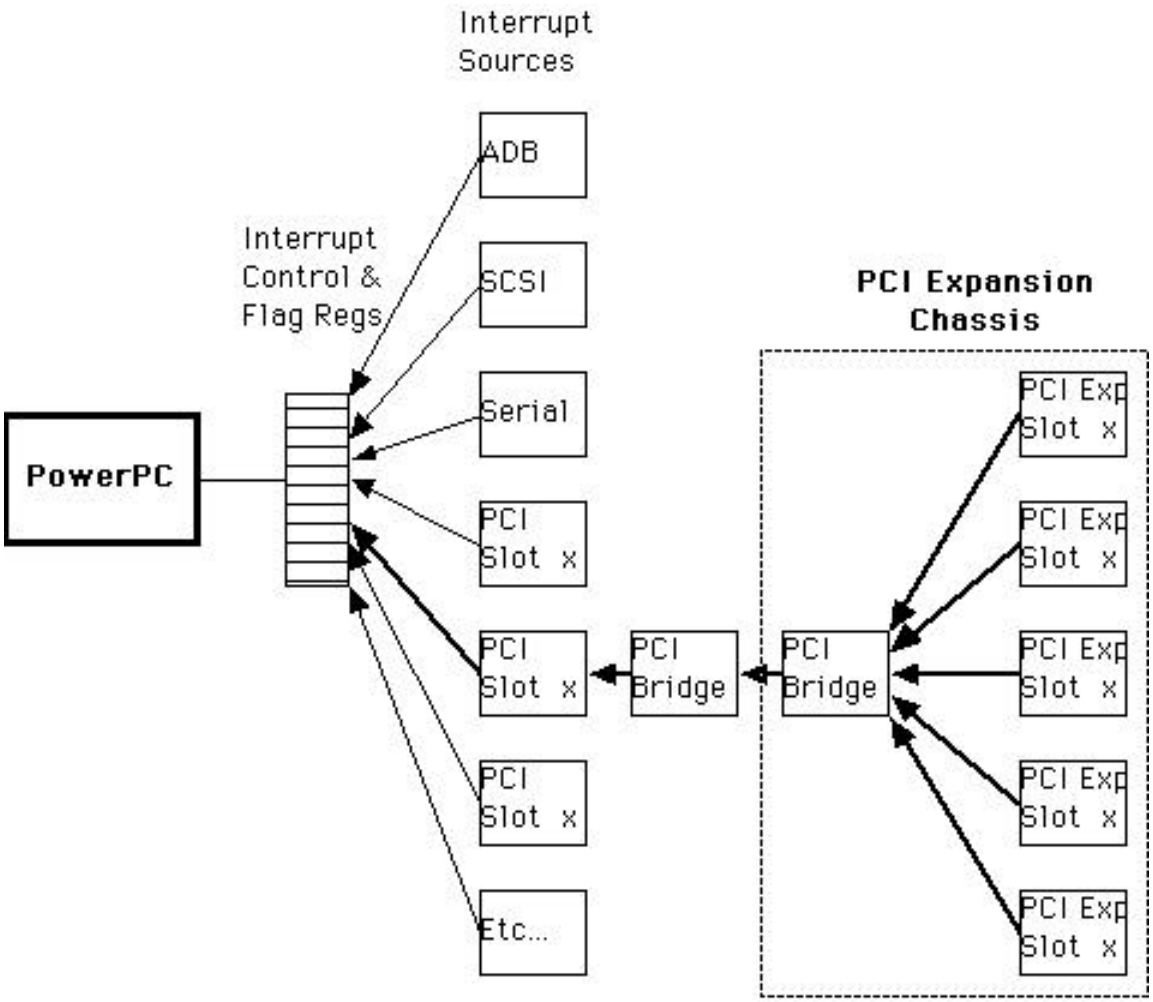
Why This Hasn't Been a Problem Before

Previously, Apple manufactured 6-slot systems (i.e., 95xx/96xx Macs); thus, the need for an external expansion chassis was diminished because 6 slots were already available for PCI expansion in the host system. The Power Macintosh G3 provides only 3 slots; therefore, some applications have migrated to using PCI expansion chassis to contain the hardware configuration.

Back to top

Coping with Expansion Chassis Issues

It's not always easy to resolve problems associated with a PCI expansion chassis, and a limited number of developers have need of such an implementation. Although cards plugged into the expansion chassis should, in theory, experience only minimal effect on overall transaction timing, the reality is that there are subtle design issues introduced by the architecture itself that must be addressed.



A Conceptual View Of PowerMacintosh Interrupts

First, additional propagation delay for the sub-bridge levels is introduced by the expansion chassis. A typical implementation is two sub-bridge levels deep, requiring extra PCI clock cycles for each level. Further delays are experienced if the cards themselves possess additional PCI sub-bridges. These delays are a combination of posting operations and actual propagation delays introduced by the hardware itself.

When cards are plugged into a host expansion slot, hardware exists in the host that provides information about which device slot is interrupting. On the other hand, all interrupts from an expansion chassis are funneled into a single interrupt slot line. This concept is known as "interrupt sharing." With interrupt sharing, the driver needs to ensure it does not depend solely on the hardware slot indicator for interrupt association. Instead, a well-behaved driver should follow the rules set forth in Designing Cards and Drivers for Power Macintosh Computers:

- Poll its own hardware to determine whether it is to provide service or not.
- If it is not to provide service, quickly return with `KIsrIsNotComplete`.
- If it is, service the routine then return with `KIsrIsComplete`.

Interrupts should, of course, only be enabled when the card's hardware is ready to be serviced. The expansion chassis will likely have more than one developer's card inserted; thus, if the driver enables interrupts prematurely, and another interrupt is already pending from a neighboring slot, the driver could end up attempting to service the wrong interrupt. Again, following the rules will prevent this. The Power Macintosh does not employ a vectored-interrupt architecture: the interrupt source tree must be traversed for each pending interrupt. This has the advantage of abstracting the interrupt handler from the hardware source, but makes it extremely important to return control immediately if your handler is not the intended service routine in order to minimize interrupt response time.

Extra time should be allowed for the hardware to properly reset interrupt flags that have been asserted, because latencies could make the interrupt look as though it is still asserted for a brief time. This is due to the accumulative full-turn delays of propagating down to the source hardware itself, then back to the host motherboard. The condition may cause the handler to be re-invoked after returning to its caller; however, when the handler regains control, the hardware interrupt has been cleared and the handler will return a `KIsrIsNotComplete` status. This ultimately will cause a spurious interrupt because no proper handler will not be found. The complication in all this is that it is not easy to tell when a device is plugged into a local bus versus an expansion chassis (and sharing interrupts). The device tree could be analyzed to make such a determination, although a good general practice for a handler might be to clear the interrupt before attempting to read the hardware back to ensure it has actually been cleared prior to returning to the caller.

Attempting to make a hypothetical determination as to which cards will work together and in what slot order is extremely difficult due to potential card interaction. This is due in part to loading characteristics and probing order as the cards are initialized. The bottom line is that any given configuration should be empirically tested to see if any problems arise. If problems are encountered, try switching the order of the cards in the slots. At the time of this writing, no known expansion chassis problems are being encountered in `OpenFirmware`, as opposed to the more rigorous thrashing that the cards receive once the system is booted. This could be because of relative slow access rates of `OpenFirmware`. More importantly, this emphasizes that these problems tend to be dynamic-speed related.

Lastly, spurious interrupts are symptomatic of the problems associated with expansion chassis operation, so be on the lookout for them as an indication of a potential interrupt timing problem.

Back to top

Additional Notes & Comments

The following are some important items that you may need to consider when working through the problem of using a PCI expansion chassis:

- The ordering of cards in the slots could affect operation;
- Verify interrupt hardware is properly reset in spite of any delays encountered;
- Interrupt sharing issues; i.e., dependence upon the slot-interrupt hardware as the sole means of determining handler validity;
- Ensure the hardware is ready to produce interrupts prior to enabling; and
- Sub-bridge delays may affect intercard operation and interrupt sharing.

Back to top

Summary

Problems introduced by PCI expansion hardware can be difficult to solve, because the issues involved were generally not a concern prior to the absence of 6-slot systems. Many of the problems can be avoided, however, by strict adherence to the ordering rules laid out in the PCI specification, and re-thinking the methods used in single-card-per-slot implementations.

Back to top

References

PCI Local Bus Specification, revision 2.1

Designing PCI Cards and Drivers for Power Macintosh Computers

Technote 1104: Interrupt-Safe Routines

Technote 1001: On Power Macintosh Interrupt Management

Back to top

Downloadables

Acrobat version of this Note (K). Download

Back to top
