Tech Info Library

Macintosh IIci: Causes for IIci Incompatibilities (Part 1 of 4)

Revised: 7/9/92
Security: Everyone

Macintosh IIci: Causes for IIci Incompatibilities (Part 1 of 4)

==========

Article Created: 30 March 1990
Article Last Reviewed: 8 July 1992
Article Last Updated:

TOPIC ----------

This is part one of a four part article detailing the changes which caused the compatibility problems the Macintosh IIci faced. A significant number of Macintosh IIci compatibility problems were related to the improvements outlined below. Keep in mind that the majority of applications were not affected by these changes and that most of those that were have been updated.

DISCUSSION ----------

Leading Causes for Macintosh IIci Incompatibilities

1) Non-contiguous memory

On all Macintosh systems prior to the Macintosh IIci physical memory is represented as one "contiguous" range of addresses. Despite the fact that memory is typically located in two separate banks with separate physical address ranges, these two ranges are presented as one "contiguous" range. The knitting together of bank A and bank B is taken care of by the memory controller. During startup the memory controller determines the amount of memory in bank A (i.e. 1MB or 4MB in case of Macintosh II) and then maps the starting address of the bank B range on top of the final address of the bank A range, which makes the two physical address ranges contiguous.

Making memory's physical addresses contiguous allows the contiguous logical memory map which is presented to software to be identical to the physical memory map. All Macintosh systems up to the Macintosh IIci utilized this 1:1 translation which means that starting at address 0, all logical and physical addresses are the same and an application knows where in physical memory an address is located.

On the Macintosh IIci the lowest physical address of bank B is fixed at physical address $0400 0000, regardless of where the highest physical address of bank A is located. This leaves a gap between the top of bank A and the bottom of bank B which is a condition referred to as non-contiguous.

The move to non-contiguous physical memory on the Macintosh IIci was made to reduce the clock cycles associated with address translation and provide greater flexibility when configuring the system. By making physical memory non contiguous you may have as many banks of memory as you like and you can put whatever density RAM in whichever bank you like.

Non-contiguous memory requires the use of the PMMU on the 68030. The PMMU performs a translation of logical to physical addresses which permits the two to be different. And, in fact on the Macintosh IIci they are different. The address ranges of the two RAM banks are made logically contiguous even though the physical address ranges of the two banks are not contiguous. The PMMU plays the role of a dispatcher. On the one side it is accepting logical address requests made by software. On the other side it is mapping these logical address requests to physical locations in RAM.

This means that the operating system must keep track of the physical location it assigns a particular logical address. It also means that an application no longer knows where a given logical address is located in physical memory. While this change did not have an impact on most developers, it did have an impact on Nubus master card developers.

A Nubus master card is any Nubus card which possesses the intelligence necessary to take over either Nubus or the CPU bus. Examples of Nubus master cards include communications co-processor cards (i.e. Apple's MCP card), high-end video overlay and graphics cards, and intelligent I/O cards (i.e. DSP or SCSI DMA). One of the standard operations a Nubus master card performs is the reading and writing of data stored in the CPU's main memory.

When a Nubus master card wants to utilize main memory it must create a buffer within main memory, which it requests from the memory manager. This request specifies not only the size of the buffer but also that it be contiguous, non-cacheable and non-relocatable. In all Macintosh II systems before the Macintosh IIci, after the memory manager returned a logical address range that met these criteria the Nubus master card could immediately begin directly addressing physical memory. This was possible because in all designs up to the Macintosh IIci a given logical address range had an identical physical address range, which meant that a Nubus Master card could determine the location of the physical addresses by virtue of the logical address range it was provided.

With the Macintosh IIci design a Nubus master card must take some additional steps before it can begin accessing physical memory. The first step is the same, the card requests a contiguous, non-cacheable and non-relocatable chunk of memory from the memory manager, and in turn the

memory manager will return a logical address range. However on a Macintosh IIci, because the logical addresses are not the same as the physical addresses the master must make two additional requests. First, it must request that in addition to the logical address range being contiguous, the physical address range must also be contiguous. Second, it must request the location of the physical address range. Only after these requests are returned may the Nubus master begin to access physical memory directly. These two additional steps meant that all Nubus master card developers were forced to re-write their drivers to work with the Macintosh IIci.

Tech Info Library Article Number:5430