ECR #: 20
Title: MDA Platform Support Requirements
Release Date: Feb. 17, 1997
Impact: Change
Spec Version: A.G.P. 1.0

Summary:

This ECR specifies additional requirements for graphics controllers which support coexistence with monochrome device adapters in A.G.P. platforms.

Background:

The A.G.P. interface specification requires A.G.P. target devices to adhere to the PCI to PCI Bridge Architecture Specification (1.0) for resource configuration, which includes allocation of VGA resources behind the bridge. Older monochrome device adapters (MDA) use a subset of the VGA resource ranges for their operation. The P2P spec does not differentiate between MDA resources and VGA resources. The implication is that the MDA, typically located on ISA, will not work when a VGA device is present behind a P2P bridge. The typical solution in today's systems is to locate the VGA device to the primary PCI bus for debug purposes and then move it back behind the P2P for final testing (without MDA support). This option is not available for the A.G.P. compliant corelogic since the P2P bridge functionality is integrated into the corelogic and therefore relocating the VGA device is not possible.

To support this capability, the platform must differentiate handling of MDA accesses from normal VGA accesses, whenever a MDA is present. Several platform implementations are possible, and the general options are presented. However, to ensure compatibility of these solutions across graphics controller implementations certain requirements are specified for A.G.P. compliant graphics devices which support MDA coexistence.

Change Current Specification as shown:

Add the following text to section 2.4 Platform dependencies (following "Performance" subparagraph):

Monochrome Device Adapter Support: Existing PCI platforms typically support coexistence of a VGA device with a second monochrome device adapter (MDA), for debug and software development. A.G.P. compliant core logic implementations may elect to support this capability depending on their specific market needs. Possible implementations include static detection of MDA adapters present off the PCI bus, and rerouting of MDA accesses to the PCI bus under BIOS control; or snooping of A.G.P. directed accesses to dynamically detect disabling of MDA resources on the A.G.P. compliant device and subsequent re-routing to the PCI bus. To ensure interoperation with possible core logic implementations, additional requirements for A.G.P. compliant graphics controllers with respect to MDA resources are specified in this document (Chapter 6).

Add the following text as a new section 6.2 A.G.P. Master MDA Resource Use Restrictions

6.2 A.G.P. Compliant Master MDA Resource Use Restrictions

A.G.P. platforms may support multiple graphics adapters in the system, including monochrome device adapters (MDA) off the PCI or standard expansion busses. A.G.P. compliant graphics controllers which support coexistence with a MDA controller are required to not use any of the following standard MDA resources, whenever a MDA is present in the system:

Memory addresses:	0B0000h - 0B7FFFh
I/O Addresses:	3B4h, 3B5h, 3B8h, 3B9h, 3BAh, 3BFh,
	(including ISA address aliases, A[15:10] are not used in decode)

A.G.P. compliant graphics controllers may utilize the MDA resources if a MDA is not present in the platform. A.G.P. compliant graphics controllers which do not support coexistence with MDA are not required to follow these MDA resource use restrictions; however such controllers will cause platform failures if MDA devices are present.

During system initialization, software needs to determine if a MDA is present in the system or not. Before the primary graphic device is enable, software will generate a write access to a MDA memory range (B0000h-B7FFFh), then reads it to determine if MDA is present. If a MDA device is present then software may configure the corelogic to pass MDA references to the primary PCI bus (which can then be claimed by the expansion bus bridge). How the corelogic chooses to provide this functionality (if supported) is specific to the implementation of the corelogic and is beyond the scope of this specification. Software completes the initialization process by allocating resources requested by the different agents and enables the devices to operate.