## CAN EISA LIVE UP TO THE MICRO CHANNEL'S POTENTIAL?

*Michael Goodwin and Karl Koessel*

For two-and-a-half years, IBM has been making big claims for MCA—increased system performance, higher reliability, easier configuration, support for multiple intelligent devices, and so on. But so far, the only obvious benefit has been the ability to install and configure expansion boards using a clever utility instead of DIP switches.

Now EISA has arrived, proffering all the advantages of MCA plus the ability to accept old-fashioned PC/XT/AT expansion boards. Will EISA do any better at fulfilling its promises, especially higher performance?

The answer may be yes, simply because the industry consortium that proposed EISA has focused on an application where the high capacity and intelligence of the bus will really matter: high-speed mass storage to serve multiple users. At this writing, we're still waiting for an application that makes the MCA bus strut its stuff. On paper, at least, some of the peripherals being released with EISA systems look like they deliver blazing server performance.

### A Tale of Two Philosophies

With the appropriate peripherals, an MCA machine should make as good a server as an EISA machine. Both—especially in a high-volume, transaction-oriented environment—could beat an AT-bus system handily. The differences lie in the master plans.

The first round of EISA systems is optimized for client/server networking in which most of the network resources reside on a massive 80486 file server that will also handle transaction-oriented computing with back-end applications like *SQL Server.* In this scenario, the EISA server offers minicomputer-like processing and storage, while standard desktop PCs concentrate on running the front end.

IBM, however, has no reason to cultivate MCA systems that compete with the company's own minicomputers. Instead, it envisions MCA as part of a peer-to-peer local area networking scheme, in which processing power and network resources (disks, printers, modems, and so on) are shared across the LAN—from larger IBM systems to MCA workstations.

Without much help from IBM, third-party board manufacturers have been hard-pressed to create peripherals that demonstrate MCA's advantages over the standard AT bus. But both EISA *and* MCA have the features to handle tomorrow's high-throughput, mission-critical needs.

### Accelerating I/O

EISA and MCA buses have the same basic function: providing an I/O conduit through which a system can shovel more data faster and with greater reliability than through an AT bus.

The AT bus is an 8-MHz, 16-bit bus. At a time when 386 and 486 processors shuttle data in and out of memory in 32-bit chunks at speeds as high as 33 MHz, I/O has some catching up to do. That's why both EISA and MCA support 32-bit I/O.

Full 32-bit I/O requires a new slot design with more pins. Furthermore, to enable systems to use faster peripherals without compromising data integrity, bus data transfer rates must be accelerated without producing unacceptable electromagnetic interference. To accomplish these goals, IBM chucked backward compatibility and started from scratch; the EISA consortium, on the other hand, created a connector that accepts either old-fashioned or EISA boards.

The pins in MCA slots are twice as close together as the pins in AT slots, and every signal pin is adjacent to a ground pin—enabling smaller circuit loops that minimize generation of (and susceptibility to) electromagnetic interference. Furthermore, inserting or removing an MCA board takes less force than an AT board does, easing the rigors of installation and upgrading.

The EISA solution is an ingenious, two-tier slot. The top tier of contacts maintains compatibility with old-style boards, while the lower tier provides MCA-like extensions (interspersed grounds, 32-bit support, and so on). EISA's 32-bit slot is the same size as an AT's, and the force needed to insert or remove boards is also similar.

New slots are only part of the story. To further boost throughput, both EISA and MCA need a way to circumvent their relatively slow clock speeds (8.3 MHz and 10 MHz, respectively). That's why both MCA and EISA provide a burst mode that permits data to move almost three times faster than in an AT. This feature becomes more important when you install several ultrafast I/O devices in one system.

### Smart Subsystems

To exploit the high-throughput virtues of MCA and EISA, you need devices that—collectively or individually—move more data than an AT bus could handle. Currently, the most taxing devices are hard disk subsystems with large, controller-based disk caches. Couple one of these with a high-speed network adapter and a souped-up printer controller, and the new buses earn their keep where an AT bus would fail.

But managing so many high-powered peripherals can overburden the CPU, slowing system throughput. That's why both the MCA and EISA buses support bus masters, intelligent peripherals that can take control of the bus in order to read from or write to system memory or another I/O device—without involving the CPU. Compaq, Zenith, NEC, and other vendors promise EISA bus masters that control multiple drives and use the burst mode to transfer large quantities of data to and from system memory.

A bus master is itself a computer, an expansion board with its own processor—typically an 80186, a 68000, or another relatively low-power chip. A bus master must work hand in glove with the host system, and ensuring correct interaction between the two is no trivial task.

*(continues)*

# CAN EISA LIVE UP TO THE MICRO CHANNEL'S POTENTIAL?

*(continued)*

So far, MCA hasn't delivered on its bus-master potential, partly because IBM has provided little support for board designers until recently. By contrast, EISA seems to be enjoying a glut of bus-master support, largely due to productive industrywide collaboration, prompt publishing of detailed specs, and the talents of a star member of the EISA chipset—the Bus Master Interface Controller (BMIC).

Designed to reside on bus-master boards and support the processor, the BMIC interfaces between the intelligent devices and the bus, greatly easing the burden on peripherals designers. In conjunction with IBM, Chips and Technologies has developed an equivalent of the BMIC, the Bus Master MicroChip. However, preproduction samples of the chip won't be available until sometime this quarter.

## Handling Heavy Traffic

With so many smart devices clamoring for access to the bus, you need a strategy to ensure that bus masters (including the CPU itself) get their fair share. MCA uses an arbitration scheme that gives control to the requesting device with the highest priority level. A "fairness" algorithm forces devices that have just been serviced to wait until other requesting devices get their turn. EISA also supports bus masters, but it arbitrates bus access among requesting devices on a rotating basis.

A glut of devices also increases the chance of false interrupts and lost interrupt requests—which can crash the system. The AT bus's interrupts, which require the system to detect a sudden change in voltage, are all too prone to such errors. Interrupt problems are even more troublesome in a high-performance PC that supports a work group.

The MCA solution is to use level-sensitive interrupt handling. Requesting devices raise interrupt line voltage and hold it high until they're serviced; the result is a significantly more reliable system. EISA uses level-sensitive interrupts for EISA boards, but if any AT boards are installed, it must use edge-triggered interrupts to service them.

## Painless Peripherals

Most hardware problems are due to improper installation of an expansion board. That's why MCA includes Programmable Option Select (POS), a feature that identifies the type of MCA board you're installing and automatically assigns system resources to it.

No switch-setting is required. POS allows duplicate boards to be installed as active spares that can take the place of a failing board; diagnostics are also simpler since POS can activate boards individually. And because POS can identify the requesting device, MCA allows devices to share interrupt request levels. This is why, for example, an MCA system can support eight serial ports, while the AT supports only two.

EISA systems also support switchless configuration of EISA boards and internal devices. The installation utility even recommends switch settings for many older-type boards. EISA also provides slot-specific I/O address ranges intended to prevent conflicts between EISA boards, allow installation of duplicate boards, and aid diagnostics.

## Which Bus to Board?

On the strength of greater reliability and higher throughput potential—but mostly because of the IBM name—Big Blue has sold millions of MCA boxes. The EISA consortium is less ambitious, positioning the first EISA machines as work-group behemoths, not the average user's workstation.

Deciding which bus to go with requires thinking ahead. IBM's vision of peer-to-peer networking isn't entirely clear, but it holds the promise of exploiting networkwide resources—including existing mainframes and minis—without the expense of a dedicated multifunction server. Recent IBM announcements (particularly those concerning subsystem control-block architecture, a software protocol scheme that enables bus masters to work more efficiently with each other and with the system) indicate that IBM aims to bring peer-to-peer networking to new levels of productivity using MCA-specific features.

Thanks to the coming avalanche of EISA peripherals, it looks as though EISA machines will offer concrete benefits sooner. If you see client/server computing in your business future, then you're a prime candidate for EISA. Currently, the biggest advantage of MCA machines is that they're readily available, and they work. EISA's super servers look great on paper, but MCA has taught us that you can't live on promises.

*Michael Goodwin is an associate editor and Karl Koessel is the technical editor for* PC World.

**POWER TIP NO. 82**

### Printing Multiple Copies of a Q&A Report

Q&A has no menu option for printing more than one copy of a report, but you can create a pair of macros that will do the job. First, from the Report menu, create a macro that prints your report once. To do this, use the Define option on the pop-up Macro menu (<Shift>-<F2>). Next, create a second macro that simply replays the first one as many times as needed.

*George R. Beinhorn*
*Nevada City, California*