NEXTSTEP In Focus, Summer/Fall 1994 (Volume 4, Issues 3 and 4).

Bus Architectures: An Overview

Brandon Greimann

Setting up and configuring peripherals for personal computers has traditionally been difficult, but new bus architectures promise to make it easier. Here's a historical perspective on buses, including your options with NEXTSTEP.

Contentious Configuration

When NeXT committed itself to creating NEXTSTEP for Intel-compatible processors, NeXT engineers had to very sadly say goodbye to the smooth integration of NeXT™hardware and software and face the wild and contentious guts of the PC.

The PC design allows variation at the lowest hardware level. PC users can change keyboards, displays, hard disks, mice, and any other component at will, at any time. And they often take advantage of this, especially to get the benefits of newer, faster technology. So PC operating systems must be able to withstand such amputation and reconstruction on the fly, and users must know what they're doing.

This article takes a look at the heart of the confusion, in and around the PC bus.

Communicating among Devices

A bus is hardwareÐessentially a set of wires, each with a specified function. Buses allow devices you connect to the computer to talk to the processor and to each other. That is, a bus connects the CPU to memory and to the display, SCSI, floppy, IDE hard disk controllers, and so on.

Over the last 50 years, engineers have developed a lot of different buses. The buses in today's late-model desktop PCs are beginning to use featuresÐsuch as bus masteringÐthat have existed in other buses for years. Buses include SCSI, networks (Ethernet and token-ring are buses), and I/O interfaces such as Apple's Desktop Bus. All are means of managing the transfer of data between devices and the main processor or CPU.

The specification of a bus tells a lot about its performance and convenience. Primary ingredients of a specification include the bus width in bits (a 32-bit bus is twice as wide as a 16-bit bus), data transfer rate (132 megahertz is lots faster than 8 megahertz), configuration (to let programmers predict the kind of information stored, where, and in what format), and the physical and electrical characteristics of the hardware.

The Good Old DaysÐISA

The original IBM PC set the world on electronic fire. Businesses bought them in bulk. Competing manufacturers lived and died in the compatibility flames of PC DOS.

In the course of these changes, a whole new class of engineer came into being: the PC guru, doer of miracles. Every office had a local magician who could fix and tweak computer configurations. These wizards' magic incantations recited resource attributes: IRQ, I/O port, DMA, and memory. These terms are still with us today.

The original PC sported a 20-bit address bus, topping out at 1 megabyte of main memory, with an 8-bit data path word width. It all lumbered along at 4.77 megahertz.

Later, the 1985 IBM PC/AT featured an updated version of the 8-bit PC bus, with added functionality for full 16-bit technology. This bus physically differed from the PC bus in that it added a second connector slot behind the 8-bit PC bus slots. This distinguished it from the 8-bit PC bus, yet made it possible to use 8-bit cards in a 16-bit slot, with all of the 16-bit specific functionality driven through the added connector. Thus 8-bit cards could be used in a 16-bit slot, and some 16-bit cards would work in 8-bit slots.

Even when this bus design was first introduced, it imposed constraints. Engineers all over the world were already familiar with the design principles and real-world hardware requirements of modern 32-bit operating systems; yet, they couldn't take advantage of them in the 16-bit environment.

To get around these limitations as much as possible, manufacturers developed the PC/AT. Coming standard with a whopping 10 megabyte hard disk and a 24-bit address bus, this new computer expanded the DMA channels from 4 to 7 and IRQs from 8 to 15, raised clock speed from 4.77 megahertz to 6 megahertz, and extended bus word width from 8- to 16-bits. But its main memory was still hobbled by the 1 megabyte architecture of MS-DOS. (Fortunately, updates to the character-based display were fast.)

A standard is born

Meanwhile, more manufacturers were making more devices, with more expansion cards, all competing for the same old resources. The longer the IBM PC/AT remained the standard, the more products appeared, and the longer the computer would remain the de facto standard. Finally, without much enthusiasm, the PC community acknowledged it as the *Industry Standard Architecture* standardÐISA for short.

The good news that came from this standardization was the free market effect on prices: Economies of scale made things cheaper by an order of magnitude. The bad news was the 8 megahertz throughput of the ISA bus, the limited system resources, and the learning curve that continued to discourage all but magicians from attempting to configure PCs.

ISA-architecture-6.eps ¬

Figure 1: *ISA bus architecture: devices soldered on-board or in expansion cards*

The official specification

The ISA specification, as written by IBM and approved by the IEEE, describes the mechanical, physical, and electrical characteristics of slots and expansion cards. It specifies 62 pins for 8-bit cards and an extra 38 pins for 16-bit cards located on a separate connection area toward the front of the computer.

With its 8 megahertz limitation, ISA data throughput is typically good enough for floppy disks, serial and parallel devices, and anything else that doesn't require a lot of bandwidth. But it's a bottleneck for fast graphic updates, hard disk transfers, network access, and other activities that need to move a lot of data fast.

Nonetheless, the ISA bus specification defines the ªclassicº AT bus, the most widely understood computer bus in history. Today, the PC/AT ISA bus is the baseline against which the industry measures the performance of other buses (see Figure 1).

Configuration woes

To configure an ISA computer, you have to break out the screwdriver and the socket wrench, open the box, and hope you don't drop anything down those dark crevices. Then you must crack open the manual, search through the index, and pray the writers use some of the same terms you've learned. And when you turn on the computer, you must watch for which devices hang. Better get out your hanky when it's installation time.

Connecting devices to a particular bus, especially in the PC world, has been a thorny issue. For one thing, there has to be a mechanical connectionÐtypically a slot with a particular dimension, shape, and a certain number of pins that complete an electrical path to any expansion card that fits.

An expansion card is an interface between the bus and a device, such as a printer, hard disk drive, or video display monitor. The card itself is a printed circuit board, typically containing logic chips, memory, and specialized chips that are particular to the card; for example, a card might have a UART chip for serial communications. Sometimes the functionality for a device is built directly onto the motherboard, so there's no need for an expansion card.

Putting an expansion card in a slot is just one step. Typically you must also alter switches and jumpers to affect the way the card interacts with the computer system. Additionally, you might need to install and use device driver software that accompanies the device.

Until recently, setting switches and jumpers and manipulating device drivers has been the art of configurationÐthe judicious allocation of system resources.

Arbitration woes

For most users, the main problem with the old style of configuring systems has been that there's no arbitration. Before you put an expansion card in a slot, you have to know what resources are allocated to all the other cards in your computer. Your new card may need a DMA channel, I/O port, IRQ level, and a memory range.

If the new card claims the same resource as another card, things will become confusedÐin the computer's circuits and in the system software. For example, two devices claiming the same IRQ may receive an arbitrary message when the CPU sends an IRQ signal. There's almost no configuration that works ªout of the boxº if you have more than two or three cards in a system. At best, the net result of adding a card that conflicts with another card is that you'll experience random problems. At worst, your computer will hang.

Fortunately, the majority of expansion cards these days come configured for standard, assumed settings for the card's type. For example, serial, parallel, floppy, keyboard, and IDE controllers are typically configured to almost-industry-standard settings, so you don't need to do much to configure them. However, you may have to fine-tune the settings if you add a sound card, network card, or anything with resource needs beyond those of a standard PC.

Still, you can't use two cards in the same system if their conflicts can't be resolved.

Bus-at-a-Glance: ISA
Summary General-purpose data transfer for modems, floppy disks, mice, graphic tablets, keyboards, and other relatively low-speed devices for Intel-compatible PCs. De facto specification, but one that's well-known and stable.
Highlights Runs at a clock rate different from that of the CPU. Uses 8-bit and 16-bit data paths and a data transfer rate of up to 8 megabytes per second.
NEXTSTEP support Release 3.3 has full support.

Modern Times

Almost from the first day the PC/AT was introduced, the needs for faster data throughput, faster display, and faster hard disk access were apparent. Yet, industry response was to keep MS-DOS compatibility and provide ad hoc workarounds to solve problems.

During these years the UNIX community kept proclaiming the advent of UNIX on the PC. SCO estimated a quarter of a million XENIX sites, mostly single users. Concurrent CP/M was born and died in a backwater. Other competing forces were afoot, inspiring jealous knowledge of multitasking, memory management, and other benefits of larger linear address spaces.

Engineers in Intel and DOS coding rooms were toiling away. Their job was to pressure-fit bigger programs into a smaller space, crowded by more and larger device drivers and other programs sharing memory. Thus, in the mid-eighties were born the Terminate and Stay Resident (TSR) and Expanded Memory Specification (EMS) ªstandardsº as attempts to provide some of the benefits of multitasking and larger memory space.

The whole thing would have exploded into a variety of premature 32-bit operating systems and applications had it not been smothered by the weight of 20 million users.

But the software kept jamming at the bus. The industry finally cracked, and out popped MCA.

The MCA bus

Clinging to MS-DOS compatibility, the PC rulersÐIBM, Intel, and MicrosoftÐconceived the idea for a new operating system for PCs, a true multitasking operating system running on a proprietary bus that would let IBM reclaim control of the desktop and hook a Fortune 500 mainframe into a PC slot.

IBM introduced OS/2 and the MicroChannel Architecture (MCA) bus in 1987. The MCA bus was the first attempt to bring greater functionality to Intel-based PCs. It introduced bus mastering, which allows a dynamic, hierarchical method of assigning priorities and bus access to each master. The idea was to decrease system overhead so the microprocessor had more available cycles for use elsewhere.

MCA provides a 32-bit data path and 4-gigabyte memory addressing with a clock rate of 10 megahertz, a little faster than ISA. The peak data transfer rate is 20 megabytes per second. Also, matched memory cycles let 32-bit memory boards run at 16 megahertz and allow the bus to quicken the pulse of data transfers between the motherboard and the expansion cards, with a 25 percent improvement of data transfer. Other MCA features include interrupt sharingÐthe ability for hardware devices to share the same IRQ if their device drivers can arbitrate conflicting interrupt calls.

However, the bus architecture was different from the ISA standard; even the pins on the expansion cards were spaced differently. Because MCA was deliberately incompatible with older bus designs and was a proprietary standard requiring PC vendors to pay IBM licensing fees, it didn't go over as well as its technical superiority might have deserved.

Bus-at-a-Glance: MCA

Summary Introduced bus mastering, matched memory cycles, and interrupt sharing to improve PS/2 computer performance over ISA computers. Expansion card pin-out dimensions are incompatible with ISA, with no provision for compatibility. Each card stores a device ID, dispensed by the Micro Channel Developer's Association. Designed for the PS/2 ROM BIOS and the OS/2 operating system. Developed by IBM.

Statistics 10 MHz clock rate, with data transfer up to 20 MB/sec on a 32-bit data path.
Supporting organization Micro Channel Developer's Association, 22280 N. Bechelli Lane, Suite B, Redding, CA 96002; (916) 222-2262.
NEXTSTEP support None in Release 3.3.

The EISA bus

In 1988, the industry retaliated against IBM's proprietary, licensed MCA bus by creating the Enhanced ISA (EISA) bus, a pin-compatible, enhanced, faster version of the ISA bus. This standard has been more widely accepted in the industry than has MCA, but hasn't reached sufficient acceptance to lower costs for its use.

Physically, EISA slots are the same size as ISA slotsÐyou can push an ISA card into the socket of the slot and it works. To use an EISA card, you keep pushingÐit plugs further down in the socket, where the EISA pins are. Thus both EISA and ISA slots use the same motherboard real estate. Each EISA slot has a unique number, whereas all ISA slots are identical.

The EISA bus incorporates a lot of the same features found in the MCA bus. It provides a 32-bit data path with a peak transfer rate of 33 megabytes per second. Advanced transfer modes reduce the number of clock cycles required to move a byte of data.

You don't typically configure EISA cards by moving jumpers. Instead, you use an EISA Configuration Utility (ECU), which is an application that configures EISA cards; ECUs are usually provided by computer manufacturers.

Bus-at-a-Glance: EISA
Summary Provides full ISA compatibility with additional features similar to those in the MCA bus: synchronous data transfer, automated configuration, enhanced DMA modes, and hardware mediated bus arbitration. EISA slot sockets are numbered, with enhanced pin configuration.
Highlights Requires a chipset including EISA Bus Controller (EBC), Integrated Systems Peripheral (ISB), and EISA Bus Buffers (EBB). Also requires EISA ROM BIOS and an EISA Configuration Utility (ECU) to allocate resources. Expansion cards must include a configuration file and have an assigned device ID.
Statistics 32-bit data path with a burst data transfer rate of 33 MB/sec.
Supporting organization CPR Services, Inc., 1400 L Street NW, Washington, DC 20005.
NEXTSTEP support Full support in Release 3.3.

The VESA Local Bus

Introduced in 1992 in conjunction with the Video Electronics Standards Association (VESA), the VESA Local Bus is usually referred to as the VL-bus. This bus adds yet another set of connectors on an extended portion of the expansion card. This extension provides a 32-bit direct connection to the CPU, with data transfers at the CPU's top speed, 132 megabytes per second for reading, 66 megabytes per second for writing. A computer with a VL-bus also provides either an ISA or an EISA bus as well.

The most common use for the VL-bus is fast display updates on high-resolution screens. For example, a 24-bit color image at 1,024 x 768 resolution is roughly 2.3 megabytes. The VL-bus offers the minimum bandwidth necessary for the high-quality display performance typical of NEXTSTEP systems; in comparison, neither ISA nor EISA is sufficient. Figure 2 shows what this architecture looks like.

VL-Simplev1.eps ¬

Figure 2: *The VL-Bus is a separate, complementary bus*

Bus-at-a-Glance: VESA Local Bus
Summary Provides an auxiliary, high-speed data path from the CPU to VL-bus devices. Practical performance considerations limit the number of VL-bus devices to one, usually the display.
Statistics 32-bit data path, with burst data transfer rates of 132 MB/sec for data reads and 66 MB/sec for data writes.
Supporting organization Video Electronics Standards Assoc., 2150 North First Street, Suite 440, San Jose, CA 95131-2020; (408) 435-0333.
NEXTSTEP support Full support in Release 3.3.

Access to Resources

Every computer has memory spread out into defined blocks, a means to interrupt then return control to programs, alternative fast paths from devices to memory, andÐif the computer's a PCÐspecial memory locations for communication between and control of devices. These are system resources.

System resources are specified or referred to by the bus specification, taking into account the software embedded in the computer's ROM BIOS.

The big issues in configuring PCs are how to avoid running out of system resources and how to assign resources to devices without conflicts. For example, you can't load more programs than will fit in memory, and you can't have two programs occupying the same area of memory. Underlying these considerations is the concept of arbitration, distributing resources among devices without contention. In ISA systems to date, arbitration has depended on the user's understanding.

Resources There are four major attributes that make up the resources of an Intel-based PC computer system: interrupts, DMA, I/O ports, and system memory.

Interrupts The processor may be interrupted every so often; for instance, the computer may need to halt a text conversion routine so that it can respond to keyboard activity. Interrupts tell the processor to stop and pay attention to the bus for a moment. Each device on the bus is assigned an IRQ (Interrupt Request Line). Some of the most common problems configuring PC systems arise when two devices use the same IRQ.

Direct Memory Access (DMA) lets devices transfer data directly to memory without going through the CPU, an optimization used by high-speed communication devices such as Ethernet and hard disk controllers. DMA channels are numbers that are unique to each device, but they can be shared as long as two devices don't use the channel at the same time.

An *I/O port* is the lowest level of resource that the CPU uses to talk to a particular device. In the PC, communication with devices is generally through I/O memory, a separate memory area from main memory.

I/O space is divided into areas for various devicesÐthe speaker has its special area, the keyboard has its area, serial communications has two areas, and so on. The I/O port number tells the CPU which area in I/O memory to look in for information about a particular kind of device.

The most essential system devices have I/O port addresses assigned. Other peripheral devices allow you to change the addresses. These can be configured through the system BIOS (or PCI BIOS), the cards' BIOS, or hardware jumpers or DIP switches. On EISA systems, each slot is assigned its own private I/O range, so you don't have to worry about this. On PCI systems, each device or slot has its own number, and all data is transfered via the PCI bus.

System memory To conform to DOS compatibility needs, a particular area of main memory is dedicated as a system resource. Most often, programs use this memory area as a buffer to store a copy of bits to be sent to the display screen. This is usually done to gain higher performance, although it slightly complicates device driver software design. Sometimes an expansion card contains software on its own ROM BIOS that must be mapped into system memory.

ROM BIOS Every PC has on-board a ROM BIOS, a chip that is specific to every vendorÐCompaq's BIOS is different from DELL's, and so on. These chips control the lowest level operations. Typically the ROM BIOS provides a user interface to set up hard disks, the system clock, and other goodies.

Increasingly, operating systems are taking over the functions of ROM BIOS. NEXTSTEP pretty much sidesteps the use of the on-board system BIOS entirely. It only uses the BIOS at boot time to get information about the system and permit rudimentary keyboard and display activity during the boot time. SoftPC uses the BIOS in some special instances. Only DOS and some other "real mode" operating systems rely on BIOS as they run.

Summary Assigning resources to devices has been the big headache of the last decade. What if documentation for a device isn't complete? Usually the manufacturer provides technical support, but you have to make the call and get through. What do you do if two devices seem to need the same resource? Sometimes you can override a resource requirement and provide an alternate; sometimes you must choose one device or the other. The trends toward devices that don't make specific resource requirements and toward software control over resource allocation are real improvements.Ð*James Stockford*

Bus Memory Spaces

ConfigurationSpace.eps ¬

The various memory spacesÐI/0 channel, main memory, and configuration spaceÐare structured very differently.

The **ISA I/O** channel is divided into various memory ranges dedicated to a particular function such as the Timer, speaker, and so on. A COM1 device, for example, is designed to work within I/O addresses 3F8 to 3FF. Each of these I/O channel memory ranges is a port. A port left empty is unused.

The structure of **main memory** loads software in at the bottom, piling toward the top. The Intel iAPX86 in real mode for MS-DOS loads the Interrupt Vector Table at the bottom of memory, then adds other memory components one on top of another (ROM BIOS resides just below the one megabyte boundary). The NEXTSTEP kernel blows away all but the IVT and some system configuration information, piling processes into extended memory as the users requests them.

Configuration space is implemented in a variety of waysÐPlug and Play is different from PCMCIA, which is different from PCI. In general it can be thought of as a set of identical memory areas that may or may not be filled, the internal structure of which is defined to store such information as manufacturer ID, device ID, and so on.

PCI configuration space is structured as a possible hierarchy of buses, each with up to 32 devices which may have up to eight functions. The zero device on bus zero is the PCI host bridge. Additional bus bridges may reside at any device location on a PCI bus. A PCI device, such as a hard disk controller or video, may be soldered into the motherboard, and is wired to a particular device number (from 0 to 31). A PCI slot socket soldered into the motherboard also is wired to a particular device number, and the expansion card that resides in the socket there by takes on the device number of the slot's socket.Ð*James Stockford*

Today and Tomorrow

In the face of more competing standards, vendors have tried to simplify and improve products at all levels.

For example, new configuration software provides a user interface to help manage installation of new devices. For instance, Microsoft has come up with an Intel Configuration Utility (ICU) for the Windows environment that is similar to the NEXTSTEP Configure application. You still have to understand and allocate resources, but the software helps you by reporting which resources are already taken and which of the available set are appropriate to the device you're installing.

Additionally, on many computers ROM BIOS provides a user interface that you can use to set up and modify system services for essential devices such as disk drives, the system clock, settings for on-board controllers, and system resources such as IRQs and port addresses.

New kinds of buses

The hallmark features of a modern computer bus are a wide word width and high throughput. The standard is a 32-bit-wide bus with throughput varying from 66 megabytes to 132 megabytes per second.

Additionally, the specifications allow for multiple buses on the same computer. For instance, an EISA computer also supports an ISA bus. A manufacturer can create a computer with EISA, ISA, VL, and PCMCIA buses. Other combinations are also possible.

One of the biggest breakthroughs has been the definition of configuration space. Modern bus specifications dictate special memory that's separate from main system memory or I/O memory and dedicated to storing information about devices and their resource requirements. They also specify software that inspects the configuration space for each device, arbitrating and allocating resources automatically based on the priorities, availability, and competing needs of other devices. These techniques free users from the messy and usually intimidating job of installation and configuration.

Configuration space provides the underpinning for such features as automatic device detection, including systemwide notification of insertion and removal of devices (ªhot-swappingº), as well as automatic configuration. Importantly, configuration space makes possible installation and configuration management across a network, granting new convenience to system administrators. (See ªBus Memory Spacesº to find out more.)

For NEXTSTEP computers, NeXT engineers have identified three important new PC technologies that make use of the configuration space concept: Plug and Play, PCMCIA, and PCI. The following sections describe these alternatives.

Plug and Play

There are two uses of the term *plug and play*, generic and specific.

Generically, plug and play denotes any configuration convention that automates the allocation of resources. This means no more DIP switches or jumpers and, generally, blissful ignorance yet successful configuration. In this general sense, plug and play features show up in modern bus specifications.

Specifically, the Plug and Play specification is designed to improve the process of installation and configuration on PCs while retaining existing ISA equipment. Each device stores configuration information on-board, and the computer provides Plug and Play software either in ROM BIOS, in the operating system, or both. The software inspects the computer to give priority to old-style ISA devices, then allocates resources to Plug and Play devices.

The current state of Plug and Play is nascent. There are few implementations of Plug and Play BIOSs, operating systems, and devices.

However, the future of Plug and Play looks bright. People can retain their investments in ISA devices and upgrade to the benefits of improved configuration. The industry seems committed to implementing the Plug and Play specification. There is development of network software to allow communication across a net, providing system administrators convenient configuration management, update, and control.

PCMCIA

PCMCIA cards integrate controller circuitry with one or more devices in a credit-card-sized package that consumes very little power. Portable computers are the obvious beneficiaries, letting a PCMCIA socket become home to a modem, network interface, SCSI adapter, complete IDE hard disk drive, and memory.

The PCMCIA specification describes a low-power implementation of a 16-bit data path, a 64-megabyte address space, modern automated configuration, and other handy features, all in a tiny form factor.

The specification is in rapid flux, with revisions coming every few months. The result is rampant incompatibility. Nonetheless, the industry is embracing PCMCIA, even to the point of including PCMCIA sockets in desktop computers.

PCI

The PCI specification describes a high-speed bus system that can incorporate multiple other buses, including ISA, EISA, secondary PCI buses, and more.

High bandwidth and multiple isolated buses promise simultaneous fast data transfer among video, hard disk, sound, and network devices, all of which is notably apt for multimedia.

A PCI-only system requires a particular chipset. Adding other buses requires an appropriate chipset controller, if it's available. Chipset compatibility may become a real thorn for developers, requiring multi-architecture binary (MAB) drivers to accommodate proliferate PCI chipsets (see Figure 3).

34330_PCI-Typicalv2.eps ¬

Figure 3: *A typical PCI system: note on-board PCI devices as well as on-board ISA devices*

Configuration across the net

In theory, once configuration is controlled completely by software, you can configure a computer from anywhere on a network. The possibilities promise broadcast upgrades of software, remote diagnosis, and on-the-fly configuration, in addition to improved e-mail, fax, and other communication services.

The Desktop Management Task Force (DMTF), including Microsoft, Intel, IBM, Apple, Hewlett-Packard, Sun, DEC, and others, is promoting Desktop Management Interface (DMI) as a standard for managing computing equipment across a network.

DMI relies on a standard storage mechanism for configuration and other system information in Management Information Format (MIF) files. Thus, if a computer's configuration changes, it's recorded in the MIF files available to the network. The DMI can query the MIF file to readjust network dependencies on that computer's resources.

DMI promises system administrators a standardized vehicle for implementing changes across a network, including management of mixed systems.

NEXTSTEP Support for New Buses

NEXTSTEP Release 3.3 provides limited support for plug and play, with full support for computers with a PCI bus.

This issue of NEXTSTEP In Focus includes three other articles about the details of these newer buses. See ªSimple Configuration: Plug and Play,º ªFlexibility in a Tiny Package: PCMCIA,º and ªThe Need for Speed: PCIº to find out more.

From the user's viewpoint

The new version of Configure can detect many kinds of devices and present you with just the device drivers for the devices you have installed, rather than a list of every possible driver; the term for this capability is *autodetect*. Configure can also automatically allocate resources for many devicesÐ*autoconfigure*.

Because NEXTSTEP circumvents the ROM BIOS, NEXTSTEP works reliably regardless of BIOS versions and upgrades. Note that NEXTSTEP Release 3.3 does not support automatic removal detection or hot-swapping.

From the developer's viewpoint

The booter, the kernel, and Configure in NEXTSTEP Release 3.3 all include rudimentary support for Plug and Play and PCMCIA specifications as well as full support for the PCI specification.

As the system boots up, the booter queries the hardware, passing information to the kernel. Programs such as Configure use the information in the kernel to match the device drivers to the hardware.

Looking Ahead

Plug and play is coming, one way or another. PCMCIA is already here on laptops and portable equipment, and desktops are including PCMCIA both for power savings and to swap devices with portables. PCI is the real comer. Manufacturers are now releasing PCI devices as their first issue, with little development of devices for other kinds of buses.

Each of these three designs has a place in the market, and NeXT is committed to accommodating all three specifications in future releases of NEXTSTEP.

Bus Memory Spaces

If you plan to purchase a new computer or upgrade what you have with new devices, NeXT has a number of resources to help you out.

Use the HCG
In purchasing new systems, refer to the *Hardware Compatibility Guide* and stick to it. Make purchases with an eye to flash-upgradable BIOSs on the motherboard. Avoid the VL-bus. Consider buying computers supporting the PCI bus for intensive data transfer applications.

Check NeXTanswers
For NEXTSTEP-specific PC configuration information, NeXTanswers is an extremely rich source, ranging from news about NeXT-tested systems to peripheral configuration to bug reports and fixes. It's also a great way to get new and updated device drivers.

NeXTanswers is an extension of NeXT's Technical Support department and is used heavily by NeXT Technical Support staff, making it a very reliable source of information.

The Intel Configuration section of NeXTanswers contains a wealth of configuration information, ranging from how to configure certified systems (systems tested by NeXT) and listed systems (systems tested by hardware vendors) to information about configuring peripherals such as SCSI controllers and Ethernet cards. Also found in these documents are solutions to problems and issues found over time by Technical Support and customers. Current NeXTanswers include advice on setting up specific models of computers, SCSI controllers, network adaptors, sound cards, display adaptors, and many other devices.

Another good place to look for simple answers to common problems is the Urgent Tech Support area of NeXTanswers.To find out more about NeXTanswers, see ªTips & Techniquesº in this issue.

Take a course
NeXT also offers training classes, including one dedicated to configuration: *Configuring NEXTSTEP.* To get information or to schedule a class, call 1-800-955-6398.

Read about it
The *Hardware Bible* is an excellent book on the subject! It's highly recommended for those curious about hardware internals and bus specs, as well as just about everything you'd need to know about PC hardware design, troubleshooting, and system design. It even provides information about case design specifications, ergonomics, and health issues. (*The Winn Rosch Hardware Bible*, Third Edition, by Winn L. Rosch. New York: Brady Publishing, 1994. ISBN: 1-56686-127-6.)

In addition, Mindshare Press publishes excellent books on programming the 486, the Pentium chip, the ISA bus, the EISA bus, the PCI bus, and the PCMCIA bus. To find out more, contact MindShare, Inc., 2202 Buttercup Drive, Richardson, TX, 75082; or call (214) 231-2216.

Finally, check out past issues of *NEXTSTEP In Focus* for articles on PCs and configuration. In particular, look at the Spring 1993 issue. It's available on-line on NeXTanswers. You can also order a back issue directly from NeXT; to find out how, call (415) 780-3769.Ð*BG*

*Brandon Greimann is a Technical Support engineer specializing mainly in system configuration and network troubleshooting. He can be reached via e-mail at **greimann@next.com**.*

Next Article NeXTanswer #1949 **Bus Terminology**
Table of contents http://www.next.com/HotNews/Journal/InFocus/Summer1994/ContentsSummer1994.html