PCI Express (PCIe) technology is now emerging as the interconnect standard for both chip-to-chip and backplane-interconnect designs, with broad applicability in PCs, servers, communications, storage, and embedded systems. Unlike several other potential interconnect solutions, it is based on the ubiquitous PCI bus specification and allows for extensive reuse of existing PCI software.

PCIe is distinct from the 32-bit PCI and 64-bit PCI and PCI-X shared parallel buses (collectively PCI/X) by providing a high-performance serial, packet-based, scalable architecture designed to overcome the limitations of PCI/X, dramatically improve time to market, and deliver new standardized features. PCIe switches provide standardized solutions utilizing existing code yet allow advanced system designs to be developed.
**PCI, PCI-X interconnect limitations**

Currently, many backplane designs use PCI/X as the interconnect technology. PCI bus is limited in throughput to one gigabit per second (1 Gbps) and involves a shared multi-drop bus. With its multiplexed address and data, PCI is slowed by wait states added by both master and target devices. As data transfer sizes are unknown at the start of a burst cycle, PCI buffer management is highly inefficient. With no split-level data transfers, delayed transactions are also inefficient since the master must guess when to attempt retransmissions. With a shared bus such as PCI/X, interrupt signals may come from a variety of devices, so additional software overhead is needed to determine which device generated the interrupt. When parity or other errors are detected in PCI, an interrupt occurs and causes a system shutdown, for which no error detection and recovery is provided.

PCI-X was developed as a speed enhancement to PCI and has been designed into servers, communication solutions, and, more recently, embedded systems, and is fully backward compatible with PCI at both the hardware and software levels. With PCI-X 2.0 and its double and quad data rate clocking modes, data rates are able to reach 32 Gbps. PCI-X provides for several enhancements over PCI: PCI-X signals are registered (vs. unregistered for PCI), providing smaller setup time to sample a signal. Split transactions are employed, wait states are not allowed, data is transferred in blocks, typically 128 bytes in size, and most transfers are burst cycles, enabling higher bus utilization. The resulting data transfer efficiency approaches 85 percent, far higher than PCI’s worst-case 50 percent. Other PCI-X advantages include more efficient interrupt handling with mandatory message-signaled interrupt support, improved error handling with attempts to recover data parity errors, and the addition of relaxed ordering capability. But despite these improvements, performance is not truly scalable, clock skew issues arise, fan-out limitations become severe, and board-layout space concerns occur at these large pin counts and higher frequencies. Yet, throughput demands march forward, beyond the scope of PCI-X. Neither PCI nor PCI-X supports peer-to-peer transfers without tying up the main bus; this prevents host-based services, thus causing a potential system bottleneck.

PCIe technology overcomes these limitations through scalable throughput and advanced features.

**PCI Express: The new interconnect standard**

In PCIe technology, the physical layer is based upon a dual simplex serial PHY and provides for switched point-to-point links, with only four wires per lane total, and no sideband signals. PCIe layout simplifies root complex chip designs, reduces the pin-out required, and allows more direct PCIe ports than does PCI/X. With the minimized routing, layout “hotspots” around the platforms CPU/memory/root complex can be avoided (see Figure 1). The fast-wide PCI-X traces can be moved to the board periphery with a 20-inch routing length permitted from PCIe root complex to PCIe bridge/switch devices. Clocking is embedded with the familiar 8 B/10 B encoding scheme. Each link can scale up to 32 lanes, and lanes transfer data at 2.5 Gbps in each direction for a potential total throughput of 160 Gbps. Future PHY layer enhancements will be boosted to 5 Gbps per direction per lane with scalability beyond this rate, while retaining the same higher-layer functions.

PCIe technology adds improved support for power management, hot plug, fault isolation, and error reporting. However, these enhancements don’t conflict with PCI/X. For instance, PCIe bridges natively support hot plug on the PCIe side and through a Standard Hot Plug Controller (SHPC) on the PCI/X side. Power Management Events (PMEs) using the PME# signal are taken from the PCI/X interface of a bridge and converted to in-band PME messages on the PCIe side and routed to the system controller. Bridges use the PCIe Requester ID to identify the source of these PMEs. Bridges also support L0 and optionally L1 low-power states.

Beyond its speed and feature advantages over PCI/X, PCIe technology provides a rich new feature set, including quality of service, flow control, and peer-to-peer transfers that allow for usage in high-speed backplane designs, while maintaining backward-compatibility with PCI/X. With PCIe technology, data can be classified in up to eight classes and is invariant through the fabric. These Traffic Classes (TCs) are then mapped to multiple Virtual Circuits (VCs) in PCIe bridges and switches (see Figure 2). Each VC can in turn be assigned a priority level with various polling techniques to determine the next VC to transfer data. Each VC also will have buffer memory to prevent dropped packets; head-of-line blocking is nearly negated, and time-critical isochronous traffic such as audio and video can now be better supported.
Flow control, which eliminates inefficient polling in PCI/X, is achieved through data link layer message transfers. In PCIe technology, flow control uses an input buffer, credit-based, per VC link flow control method. With this technique, for each VC, data is not sent to a device with input buffer memory that lacks sufficient space for a new packet; rather, data link layer flow control packets are sent to the sending node informing it of the amount of buffer space available. The nearest PCIe device can thus avoid congestion and potentially dropped packets. Peer-to-peer data transfers and message request transactions also are defined in PCIe for multiple hierarchies within a single fabric topology (see Figure 3). While enumeration of switches, bridges, and endpoints are host-based and routing is based on addresses, routing through the host is avoided.

PCIe software compatibility with PCI/X

Just as existing code compatibility is a major determinant in the choice of a microprocessor for a given design, interconnect choices will need to factor in existing board subsections and add-in card device code. The programming model of PCIe software allows backward compatibility with PCI/X software. PCIe is also compatible with PCI/X at the hardware level, with PCIe bridges providing this backward compatibility at the component and slot level. During the transition to PCIe solutions, numerous endpoint devices and chip sets that utilize PCI/X will continue to exist. It is important from both time and cost standpoints that these chip sets, add-in cards, and other peripheral devices be supported.

PCIe software is backward compatible with legacy operating systems, drivers, and PCI software protocol stacks. The 256 byte base configuration address space for PCI/X is the same as for PCIe software, with this memory-mapped space extended to 4 kB for the added functionality associated with the enhanced features of PCIe technology. PCIe software reports class/subclass and device type information, just as in PCI/X. In transition from PCI/X to PCIe, memory, configuration, and I/O transactions are forwarded, and device ID message translation is optional. PCIe completion messages are translated to either delayed (PCI) or split (PCI-X) transactions, and locks are propagated. Data integrity is also supported through these bridges; on the PCIe side, link-level 32-bit Cyclic Redundancy Check (CRC) and optional end-to-end ECRC (generated at Layer 3) provide error checking. CRCs (16-bit) are also generated for messages sent at the data link layer. On the PCI/X side, parity and error checking and correction signals are generated. Errors occurring on the PCIe link can be propagated through the PCI/X side of the system for system controller software present on the PCI/X side. PCI/X interrupts are converted to PCIe assert/de-assert interrupt messages. Through reverse bridging, PCIe-based endpoints can be bridged to PCI/X-based systems, as well.

PCIe momentum

The first PCIe specification was released in April 2002 and since then has been revised twice, with version 1.1 finalized in April 2005. More than 500 companies now support the standard with silicon, software, test tools, IP, and programmable devices. The silicon covers the gamut, from north bridges to switches, bridges, and endpoints for PCI/X, gigabit Ethernet, and InfiniBand. A complete ecosystem is now in place with five specifications released to help with testing and interoperability assurance.

PCIe-enabled personal computers with enhanced graphics are already on the market. Many server designs are transitioning to PCIe technology, with storage systems following right behind. With PC, server, and storage designs as the primary driving forces, along with an emerging new ExpressCard standard, component and system pricing will continue to fall, thus spawning even more designs. Also driving PCIe deployment is a robust interoperability effort, including frequent PCIe “plug fests” that are well attended by vendors and other workshops that help ensure the ability to mix and match components from those vendors. These efforts, combined with technological and market momentum, are helping to guide the ongoing migration from PCI/X to PCIe technology.

John Gudmundson is senior product marketing manager for PLX Technology, Sunnyvale, Calif. Prior to PLX, he served in marketing positions at Sunrise Telecom, Nortel Networks, Integrated Telecom, and Ascend Communications. He is an active member of the IEEE and American Marketing Association, and maintains positions on several industry standards bodies. He holds a BSEE from the University of California, Los Angeles, a BSBA from the University of California, Berkeley, and an MBA from the University of Southern California.

To learn more, contact John at:

**Reprinted from Embedded Computing Design / January 2006**