Cahners
MICROPROCESSOR REPORT
www.MPRonline.com
THE INSIDER'S GUIDE TO MICROPROCESSOR HARDWARE

# RapidIO Expands Narrow-Bus Options

## *New Standard Will Compete With PCI-X and Infiniband*

*By Peter N. Glaskowsky {5/8/00-01}*

Into a market crowded with new point-to-point switched interconnects, Motorola and Mercury Computer Systems have introduced yet another: RapidIO. The new standard offers the same basic benefit—high bandwidth over a narrow interface—claimed for FibreChannel, FireWire, the HotRail Channel, InfiniBand, AMD's Lightning Data Transport, Rambus memory, and similar specifications.

Despite the plethora of alternatives, RapidIO is likely to see widespread use. The new standard is meant to be used as a chip-to-chip and backplane interconnect within networking equipment, a market dominated by RapidIO's key supporters, which include Cisco, Lucent, and Nortel. These companies alone control enough of this market to justify the design of RapidIO-equipped embedded microprocessors and network interfaces. These chips today are often designed with PCI-bus interfaces, but RapidIO will offer better performance than PCI at a comparable cost.

RapidIO should also prove attractive to network-hardware makers that today use high-performance proprietary backplane architectures. Compared with these proprietary solutions, RapidIO should offer similar throughput at a lower cost, with the added benefit of improved standardization.

The RapidIO standard will be managed by the RapidIO Trade Association (*www.rapidio.org*). In addition to the previously mentioned companies, the association boasts a number of other influential founding members: Galileo Technology, Fujitsu System Technologies (a business unit of HAL Computers), PLX technology, Seagull Semiconductor, Sky Computers, Tundra Semiconductor, and Xilinx. HAL and Sky are computer systems manufacturers; the others are chip vendors.

### Specifications Sound Familiar

Initial RapidIO implementations will use either an 8- or a 16-bit parallel interface operating at speeds from 250MHz to 1GHz, with two data transfers per clock cycle. The maximum configuration yields 4GB/s of peak throughput.

The interface uses low-voltage differential signaling (LVDS), compatible with the widely used IEEE 1596.3 LVDS standard. A source-synchronous clock signal is used; the 16-bit interface provides one clock pair for each 8 bits of data to reduce skew. The interface also includes a FRAME signal, which identifies the start of a packet or a control symbol.

Each RapidIO signal is unidirectional, so two sets of signals are required for each bidirectional link. For the 8-bit interface, one complete link requires 40 signal wires. The 16-bit version uses 76 signal wires. Because all signals are differential, no additional ground wires are required. This physical interface is called an "8/16 LP-EP" (for link protocol end point).

Error coverage for the physical layer is provided by 16-bit cyclic-redundancy codes (CRCs) in request and response packets; control symbols are transmitted twice (once in inverted form) and protected by error-correcting codes. An error in a request or response packet is handled by resending the packet; errors in control symbols can usually be corrected. Additional error coverage may be provided at the application layer if desired.

The resulting channel is similar to those described by HotRail (see *MPR 7/12/99-05*, "HotRail Rides With New

Core Logic") and AMD (see *MPR 10/25/99-06*, "AMD Shows Big Server Plans"). The HotRail and AMD proposals are aimed at multiprocessor servers, which require high-bandwidth connections for interprocessor communication and memory access.

A switching fabric—one or more chips equipped with multiple RapidIO interfaces—connects all of the RapidIO devices in the system. Switch chips can be very complex. HotRail's 14-port switch chip for 8-CPU servers, which supports a protocol that is conceptually similar to that of RapidIO, has about one million gates of logic plus 616 signal pins.

Though RapidIO's initial LVDS physical layer is expected to meet the needs of a wide variety of applications, the RapidIO protocol supports almost any signaling interface. Motorola says that, over time, the RapidIO protocol may be implemented on serial links, optical fiber, or other connections. Serial links would have to run at much higher speeds to match the throughput of the initial parallel configurations, but the technology for high-speed serial connections is already being developed by the same networking companies that are backing RapidIO.

### Protocol Optimized for Networking, Storage

RapidIO is designed to transfer large amounts of data with low software overhead. The protocol uses a simple send/receive model based on a shared physical memory map of the entire system. Most data transfers are initiated by read and write requests that specify an address and data. Each request is acknowledged by a response packet.

Each packet carries an 8- or 16-bit target-device address as well as a 32-bit device-offset address that specifies a 32-bit-word boundary. Taken together, these address fields define an address space as large as 50 bits. The protocol allows multiple target-device addresses to be assigned to a single physical device. Multiple transactions (up to 256 between each pair of end points) may be pending at any time. Special target-device addresses may also be used to implement multicast and broadcast addressing modes, but these modes are not defined in the specification. Customers that require multicast or broadcast support must develop appropriate extensions to the RapidIO standard. Extensions are also required in applications that require larger address spaces.

Global shared-memory architectures are supported by a directory-based cache coherency scheme that holds each memory controller responsible for tracking the location of the most current copy of each data element in the system. The RapidIO specification does not define the nature of data elements; this too is left as an implementation option.

The RapidIO approach greatly reduces the amount of coherency traffic between devices compared with that of snoop-based coherency schemes. Each element in each controller's directory is tagged as modified, shared, or local. All controllers must be notified before the state of any shared element can be changed. Though more efficient schemes are possible, this solution was relatively easy to implement and adds little extra logic to RapidIO interface chips.

Address and control information adds significant overhead to RapidIO packets. A 32-byte read or write operation requires 35 bytes of overhead. The overhead increases only slightly for longer transfers—to 37 bytes of overhead for a 256-byte transfer (the maximum transfer size). RapidIO's connection efficiency is therefore only 48% for 32-byte transactions, whereas 256-byte transfers are 87% efficient. This efficiency is comparable to that of other expansion buses, including PCI.

RapidIO adds only moderate latency to most transactions. A 32-byte read request through a switching-fabric chip to a remote RapidIO-based memory controller experiences about 156ns of extra latency when 16-bit, 500MHz connections are used. (This delay, which includes all sequencing, synchronization, and parsing operations, is in addition to the memory-access latency.)

This figure compares well with the latencies in current PCI-bus systems. A typical 64-bit, 66MHz PCI read operation takes at least 158ns to return 32 bytes. Though PCI-X operates at up to 133MHz, PCI-X chips have extra latch stages, and most read operations are split into two bus transactions, so the effective latency is similar. Bus arbitration, turnaround delays, and other factors, however, can increase the latency of PCI and PCI-X buses to several microseconds. RapidIO's superior predictability allows designers to reduce the size of buffer memories, which helps reduce system cost.

### Competition May Exceed Expectations

The target market for RapidIO overlaps with that of Infiniband, a high-speed serial interface standard supported by Intel, Microsoft, and many PC OEMs (see *MPR 9/13/99-msb*, "NGIO, Future I/O Merge"). Motorola says it doesn't expect RapidIO to compete with Infiniband, but we believe some overlap is inevitable.

Infiniband's primary emphasis is the interconnection of subsystems in large servers, especially when these subsystems reside in separate chassis. Infiniband will be used to connect storage and networking controllers such as RAID arrays and Ethernet switches to enterprise servers and server clusters. RapidIO's initial configuration, which supports a total link distance of only about one meter, is not suitable for such applications.

It seems inevitable that RapidIO will be enhanced over time to allow greater distances between endpoints. Motorola's references to serial and optical implementations clearly anticipate such enhancements, which would bring RapidIO into competition with Infiniband. These implementations are not the current focus of the RapidIO effort, but we view them as a natural next step.

The Infiniband Trade Association (*www.infinibandta.org*), meanwhile, is developing a backplane implementation

of its interface. This version of the Infiniband physical layer will provide immediate competition for RapidIO. Designers of peripheral and network interface chips will be reluctant to support both protocols, and some smaller chip makers may be forced to choose between Infiniband and RapidIO. In most cases, this choice will depend on the vendor's market focus—servers or networking hardware. However, many vendors would like to serve both markets.

There are substantial differences between the RapidIO and Infiniband protocols, but both are probably acceptable for most of the applications envisioned by both camps. Infiniband uses a message-passing architecture that requires much more software support than RapidIO's memory-mapped I/O model, but most network and peripheral interfaces use complex driver models based on message passing, so this is no handicap. (RapidIO provides basic message-passing services to implement "mailbox" and "doorbell" functions, but these are not meant to carry large amounts of data.)

Infiniband imposes more transaction latency than RapidIO, but RapidIO transfers will often involve fairly long data blocks that are split across multiple RapidIO transactions—64 to 1,500 bytes for network packets and 512 to 2,048 bytes for disk sectors. Such long transfers will take appreciable amounts of time, so RapidIO's lower added latency is not crucial to those applications. In applications that rely on many small transactions, however, RapidIO's lower latency will boost sustained throughput.

### Last Call for Fast Buses

If it weren't for the big companies supporting the RapidIO effort, one might reasonably wonder whether it had any chance against its many competitors. The support of these companies makes it likely that RapidIO will be widely used as a local interconnect in networking equipment.

The new standard is likely to compete directly with Infiniband in servers, however. Even RapidIO's biggest boosters are also backing Infiniband. Motorola itself is a member of the Infiniband Trade Association, and Cisco, Lucent, and Nortel are sponsoring members.

We expect to see RapidIO used as a processor-to-memory interconnect in smaller systems, but big iron will need the higher bandwidth and superior coherency support offered by interfaces such as LDT. We believe most high-performance systems will continue to use proprietary solutions optimized for their specific needs.

For applications that need a standardized system-level interconnect, PCI-X, RapidIO, and Infiniband appear to provide an adequate range of choices. There are always new technologies (such as JAZiO's fast, narrow multidrop bus—see *MPR 2/21/00-02*, "JAZiO: Slow Edges Can Run Fast") that offer certain advantages in some applications, but these alternatives are now faced with an uphill battle to prove their worth. Though they may find homes in a few closed-box systems, they are unlikely to be supported by designers of components for open systems.

The inherent scalability of RapidIO and Infiniband likely means that they will be around for a long time. It remains to be seen how well their respective trade associations will manage future transitions to higher data rates and new physical media. The PCI bus, for example, has been adapted to a number of new form factors over the years, but its failure to adapt to the evolving performance requirements of PCs, servers, and embedded systems made the Infiniband and RapidIO efforts necessary. We hope these new standards prove even more flexible and durable.