Advanced Switching Extends PCI Express

Advanced Switching supports flexible interconnects, distributed computing, and multicast options. Teamed with Xilinx FPGAs and RocketIO transceivers, it will dominate next-generation embedded computing and communications platforms.

by Kiran S. Puranik
Staff Design Engineer
Xilinx, Inc.
kiran.puranik@xilinx.com

The PCI Special Interest Group's (PCISIG) ratification of the PCI Express™ Base specification in July 2002 was a major milestone in the widespread adoption of serial system interconnect architectures. Now, the combination of Xilinx Virtex-II Pro™ Platform FPGAs, RocketIO™ multi-gigabit transceivers, and embedded IBM PowerPC™ 405 processors is making high-performance PCI Express designs easier and faster.

PCI Express is suitable for a wide range of computing and communications applications because it is a low-cost, high-performance, interoperable, and scalable solution. An important adjunct to the base standard is the Advanced Switching (AS) technology. It enhances the base standard to address the needs of the communications infrastructure for the next decade.

Single-chip AS solutions for server, storage, and communications applications can be configured with a Virtex-II Pro Platform FPGA, on-chip RocketIO multi-gigabit transceivers, and embedded IBM PowerPC 405 processors. The extensive Xilinx intellectual property (IP) offerings – including the world's first Real-PCI™ Express solution – will accelerate product design cycles and time to market.

PCI Express Architecture

PCI Express is a layered architecture consisting of physical, data link, and transaction layers. Device drivers and application software constitute upper layer protocols (ULPs).

PCI Express Base features include:

- LVDS, 2.5 Gbps, serial links scalable up to 32 lanes in each direction
- Embedded clock and 8b/10b transmission code
- Native hot attach/detach capability for high-availability applications
- Robust and efficient link protocol
- Classes of traffic and support for isochronous applications
- Packet-based protocol and credit-based flow control per virtual channel
- Link level and end-to-end data integrity (CRC-32)
- Completely in-band signaling support.

Advanced Switching

Advanced Switching leverages the physical and data link layers of the PCI Express Base specification. Communication and embedded computing enhancements are added at the transaction layer to address chip-to-chip, backplane, and inter-chassis data communication requirements. The AS fabric architecture is designed for both control and data-plane switching applications.

AS components can be broadly classified as being either switches or end systems. Figure 1 shows the AS protocol stack on an end system. The transaction layer creates an interface between ULPs and the data link layer. It serves as the tunnel for ULP encapsulation and extraction at an end system.

The AS transaction layer packet is shown in Figure 2. It contains an AS route header followed by a payload section. Ingress end systems provide route and encapsulation payload, while egress end systems extract payload. In order to perform packet switching, the AS route header is completely agnostic to the contents of the payload section.

An entire AS transaction layer frame is shown in Figure 3. The physical and link layer portions of the frame are added and removed at each hop between fabric components.

Important Features of AS

Legacy PCI platforms require strong parent-child relationships between connected components. Tree topologies are necessary for hardware and software compatibility. In contrast, AS fabric system topologies can be described as a graph of connected switches and end systems. Switches constitute internal nodes of this graph, providing interconnects with other switches and end systems. End systems, on the other hand, are the edge nodes, representing data ingress and egress

points. The ability to support many topologies gives platform architects enormous flexibility regarding placement of critical resources.

Supports Distributed Computing Architecture

AS enables distributed processing systems, which results in multiple memory-address domains within a platform. This is in sharp contrast to the single flat address domain found in legacy PCI platforms. AS platforms allow load-store and messaging protocol interactions between concurrent hardware and software processes on end systems. This facilitates peer-to-peer-based applications commonly found in server, storage, and communications arenas. Distributed processing offers better scalability and ultimately a lower cost of operation.

Path-Based Unicast, Multicast, and Broadcast Packet Routing

Unicast packet switching is based on path information embedded in the AS route header and takes the form of a turn pool. Path routing requires no up-front programming of switches. Switches simply look at a packet's route header to determine the egress port. This simplifies switch design and platform configuration enormously. A packet's forward route turn pool can also serve as a backward route to the source. Completions for read requests and event notifications to a packet's source end system are examples of backward-routed packets.

The multicasting feature allows an end system to target a packet to multiple end systems. A multicast group index is carried on each multicast packet's route header. A multicast group uniquely identifies a set of switch egress ports for each switch on a multicast packet's path. At a switch, a multicast group table is found in a lookup table using the packet's multicast group index.

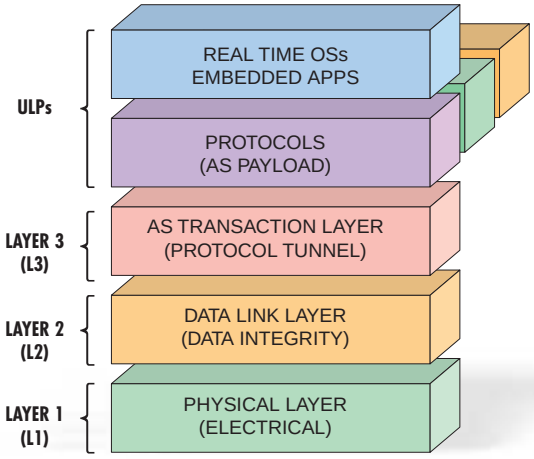

Figure 1 - PCI Express AS protocol architecture

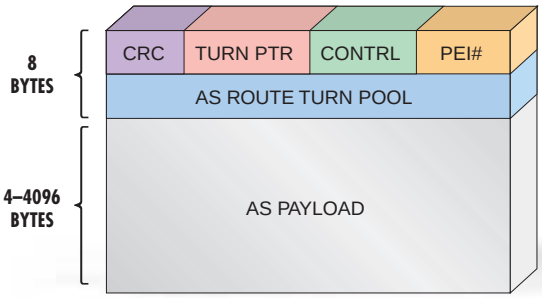

Figure 2 - AS transaction layer packet

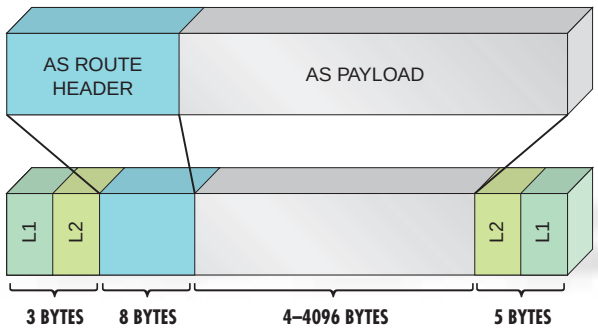

Figure 3 - AS transaction layer frame

The packet is then replicated on each port contained in the multicast group.

When a switch detects a broadcast route header on a packet, it is required to replicate it on all ports except the packet's ingress port.

Congestion Management and Quality of Service

Bandwidth provisioning is an important aspect of AS fabric management because it helps guarantee Quality of Service (QoS) levels to applications. End systems must operate within prescribed bandwidth limits by metering the rate of data transfer.

Congestion in AS fabric can be caused by unexpected transient events such as component failure, accidental removal, or errant end-system behavior. Congestion causes packets to suffer excessive latencies, resulting in a loss of expected levels of service.

AS congestion management mitigates fabric congestion and maintains QoS levels. Congestion is detected at switches; when detected, Backward Explicit Congestion Notification (BECN) messages are sent upstream to end systems contributing to congestion. End systems respond to BECN messages, reducing the rate of data injection by a specified amount.

Congested packets may also be marked with a Forward Explicit Congestion Notification (FECN) bit. FECN notifies downstream end systems of congestion on a certain path. Some AS applications may also set a discard bit on AS route headers, allowing downstream switches to selectively discard packets to alleviate local congestion. In the absence of BECN messages, end systems restore normal traffic flow in specified increments. Fabric management may use congestion notifications to initiate corrective action and activate built-in fail-over using alternate paths.

Differentiated Classes of Service

Class of Service (CoS) mechanisms reduce the complexity of maintaining QoS by mapping multiple traffic flows into a few service levels. AS fabric resources are allocated based on as many as eight service levels called Traffic Classes (TCs). Traffic flows are aggregated and forwarded by fabric components based on the TC of these packets.

Within a TC, AS fabric preserves the ordering of packets end-to-end, with the exception of those marked as bypassable at

the source. There is no such ordering requirement across TCs. AS components map TCs to Virtual Channels (VCs) corresponding to hardware channels within the components. AS components must implement at least two VCs. All AS link partners must share the same TC to VC mapping, downshifting if necessary to the smallest common number of VCs supported between the two link partners.

Each AS VC contains two independent queues – the main queue and a bypass queue. AS link flow control manages flow credits for each queue independently. A packet marked bypassable must enter the bypass queue if it causes VC head-of-line (HOL) blocking. The bypass queue is serviced as soon as bypass credits become available. This capability is compatible with legacy bus protocols that require write transactions to pass HOL-blocked read transactions, to avoid possible deadlocks.

Multi-Protocol Support

Protocol Encapsulation Interfaces (PIs) represent fabric management and application-level interfaces to the AS fabric. Table 1 details a list of currently supported PIs. PIs 0-7 represent fabric management interfaces, while PIs 8-254 are application-level interfaces. As shown in Figure 4, AS supports the tunneling of virtually any protocol. This makes AS platforms modular and cost-effective, as well as easy to deploy and support.

Application PI implementations mandate an efficient encapsulation, without losing a tunneled protocol's semantics, which enables effective extraction at the fabric's egress. A PI may be custom-tailored to suit a specific requirement or can tunnel standard protocols to support a broad range of applications, such as SPI, ATM, Ethernet, or TDM.

Support for Segmentation and Reassembly

The Maximum Payload Size (MPS) of the AS platform is the least common denominator of MPS supported by all components within the platform. All PIs must restrict AS Transaction Layer Packet payload size to platform MPS. End systems that need to encapsulate larger-than-MPS protocol packet sizes must split these into sub-MPS-sized segments. This also requires keeping track of multiple segments of the original packet and reassembling them at the fabric egress end. AS supports a standard mechanism to perform segmentation and reassembly (SAR) via the SAR PI. Individual PIs may choose other SAR implementations.

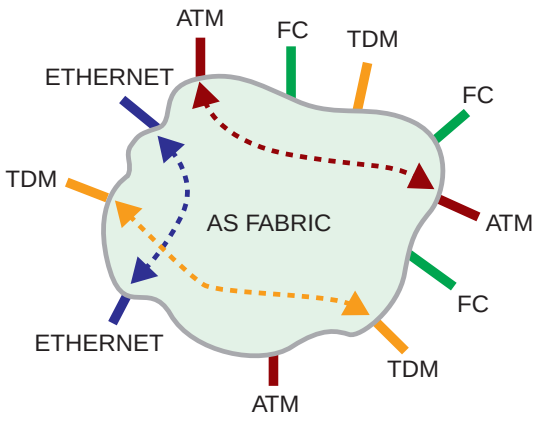

Figure 4 - AS tunneled protocols

PI Index	Protocol Encapsulation Identity
0	Fabric Discovery
1	Multicasting
2	Congestion Management
3	Segmentation and Reassembly
4	Configuration Management
5	Fabric Event
6	Reserved
7	Reserved
8	PCI-Express Base
9-223	Unassigned, Future PEIs
224-254	Vendor-Defined PEIs
255	Invalid

Table 1 - AS protocol encapsulation interfaces

Xilinx Real-PCI Express

The Xilinx Real-PCI Express solution is a combination of two leading technologies: the PCI Express specification and Virtex-II Pro FPGAs. It provides user-configurable options, excellent flexibility, and Xilinx Smart-IP™ technology guarantees critical timing. Key benefits include:

Availability

The world's first PCI Express solution is available for download today. It enables your compute and communication systems to achieve the highest level of performance using serial I/O technology.

Performance

The RocketIO 3.125 Gbps-capable transceivers on Virtex-II Pro FPGAs enable multiple 2.5 Gbps lane implementations on a single chip.

Flexibility

The inherently programmable nature of FPGAs allows you to continually tune your design to changing platform performance and functional requirements, reducing your risk in adopting the standard.

Faster Time to Market

Today, there is simply no easier way to develop PCI Express applications with minimal impact to your overall system development cycle. Xilinx will provide updates for any specification changes.

Conclusion

Advanced Switching architecture addresses all of the major requirements for next-generation system interconnect solutions, such as scalability, expandability, modularity, high availability, and peer-to-peer capability, with built-in QoS and CoS support.

Xilinx Virtex-II Pro Platform FPGAs, together with the Real-PCI Express solution, are well-positioned to provide compliant Advanced Switching solutions, enabling rapid product deployments. More information about the Xilinx Real-PCI Express solution may be found at *www.xilinx.com/pciexpress.* Σ