HyperTransport™
C O N S O R T I U M

HyperTransport™ Consortium

HyperTransport™ I/O Technology Overview
An Optimized, Low-latency Board-level Architecture

White Paper
HTC_WP02

June 2004

The HyperTransport Consortium

www.hypertransport.org

The HyperTransport Technology Consortium disclaims all warranties and liability for the use of this document and the information contained herein and assumes no responsibility for any errors that may appear in this document, nor does the HyperTransport Technology Consortium make a commitment to update the information contained herein.

DISCLAMER

This document is provided "AS IS" with no warranties whatsoever, including any warranty of merchantability, non-infringement, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification or sample. The HyperTransport Technology Consortium disclaims all liability for infringement of property rights relating to the use of information in this document. No license, express, implied, by estoppels, or otherwise, to any intellectual property rights is granted herein.

TRADEMARKS

HyperTransport is a licensed trademark of the HyperTransport Technology Consortium.

Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

I - HyperTransport – A Powerful Board-level Architecture

The need to support higher speed processors and memories and to integrate slow-speed and communications-based I/O devices have led to the development of several new technologies for I/O interconnect. AMD, Alpha Processors and API Networks developed HyperTransport Technology as a result of those companies' and their partners' efforts to simplify and integrate high-speed data traffic between high-speed processors, memories and I/O. The initial 2001 HyperTransport 1.03 specification defined a bandwidth of up to 12.8 Gigabyte/second aggregate bandwidth, far exceeding any other I/O technology at that time. The latest HyperTransport 2.0 specification supports a 22.4 Gigabyte/second aggregate bandwidth yielding the highest bandwidth in the industry as of mid-2004. When the open HyperTransport Technology Consortium was formed in the summer of 2001, many leading systems and communications equipment companies joined and participated in shaping the direction and profile of the final specification. Specifically, communications mechanisms such as sophisticated virtual channels and user packet handling capabilities were added. The resulting HyperTransport Technology specifications, Releases 1.03, 1.05, 1.1(DirectPacket™) and 2.0, define a practical, high performance, highly optimized board-level architecture, ideally suited for applications ranging from consumer, embedded systems, personal computers, portable computers, servers, network equipment, and even supercomputers.

HyperTransport Technology Attributes			
Feature	**Specification**	**Release Version**	**Comment**
Maximum link transfer rate	2.8 Gigatransfers/sec., Scalable from 400 Megatransfers/sec.	HyperTransport Specification Release 2.0 – Maximum	Highest bandwidth available in chip-to-chip link. NOTE: transfers/sec. is used as the measurement instead of bits/sec., because the links may vary in bit width.
	1.6 Gigatransfers/sec.	HyperTransport Specification Release 1.x – Maximum	
Maximum aggregate bandwidth	22.4 Gigabytes/sec., scalable from 200 megabytes/sec.	HyperTransport Specification Release 2.0 – Maximum	Highest throughput available in chip-to-chip link
Data link	2, 4, 8, 16, or 32 bits wide	All	Can be asymmetrical

	Parallel point-to-point	All	
	Dual, unidirectional links	All	Eliminates sideband signals
	Sideband signals: clock, control, reset, and powerok	All	Fewest sideband signals of any parallel interconnect
Electrical protocol	1.2 Volt LVDS	All	Low power, high noise immunity
Clock	Dual-data rate (data exchanged on rising and falling clock edge)	All	Moderate speed clock forwarded interface yields an inherently reliable interface
Packet protocol	4 – 64 byte data payload with 8 or 12 byte header	All	Low protocol overhead relative to data payload
	Command, Address and Data carried in data payload	All	
PCI Support	PCI software transparent	All	100 percent compatible
Link Protocol	Host-reflected link traffic	All	PCI compatible
	Peer-to-peer link traffic	HyperTransport Specification 1.1 and 2.0	
Communications Support	Maximum of 6 virtual channels with flow control	HyperTransport Specification 1.03, 1.05	
	Maximum of 9 virtual channels with flow control; 16 streaming channels with flow control	HyperTransport Specification 1.1 and 2.0	
Native User Packet Handling	Yes	HyperTransport Specification 1.1 and 2.0	
Error Handling	Automatic error detection	All	
	Automatic error detection and recovery (optional)	HyperTransport Specification 1.1 and 2.0	
Address Range	40-bits	HyperTransport Specification 1.03	
	64-bits (optional)	HyperTransport Specification 1.05, 1.1 and 2.0	
Concurrent outstanding transactions	32	HyperTransport Specification 1.03 and 1.05	
	128	HyperTransport Specification 1.1 and 2.0	

Internetworking Support	PCI-X Mapping	HyperTransport Specification 1.03 and up.	
	PCI Express Mapping	HyperTransport Specification 2.0	

The development of HyperTransport technology was influenced by two important factors: the need for PCI (Peripheral Component Interconnect) compatibility and a requirement for low cost implementation. The widespread use of the PCI bus made PCI software compatibility an important advantage. This helped maintain the tremendous investment made by the industry in this popular interconnect bus technology. The fact that HyperTransport devices appear to be PCI devices and conform to all ordering and enumeration properties of the PCI standard played a large factor in the rapid and widespread adoption of HyperTransport technology throughout the industry. Equally as important, the developers of HyperTransport realized that the economic impact of silicon technologies would rapidly bring down the cost of very high performance processors, memories and systems. Consequently, they made low cost of implementation a requirement for HyperTransport technology. This was partially achieved through the point-to-point link structure and partially through the use of enhanced 1.2 Volt LVDS (low voltage differential signaling) that was adopted as the physical electrical link. The choice of LVDS technology reduced system power consumption, increased noise immunity, simplified printed circuit board manufacture and lowered system cost.

But, in addition to delivering raw bandwidth, PCI compatibility and low cost, HyperTransport provides the integral interconnect backbone structure that links all of the core functional units (processor, memory and I/O elements) in a board-level system. Thus, HyperTransport is actually an optimized board-level architecture that delivers the highest effective bandwidth throughout a board-level system. As an optimized board-level architecture, HyperTransport delivers the highest throughput of any standard solution, provides the lowest possible latency, harmonizes interfaces, simplifies implementation, reduces software overhead, enables the intermixing of load/store traffic with packet-bus traffic and supports scalable performance.

The widespread adoption of HyperTransport across a broad spectrum of high performance product sectors ranging from consumer devices to personal computers, network equipment and supercomputers, is tangible proof of the power and flexibility of the HyperTransport architecture. In many cases, HyperTransport's integration extends into the processor, such as in AMD's Opteron and Athlon64 64-bit x86 processors, Transmeta's Efficeon x86 processor, Broadcom's BCM1250 64-bit MIPS processor, and PMC-Sierra's RM9000 64-bit MIPS processor family. In these instances, HyperTransport operates as a fully integrated front-side bus and the traditional NorthBridge-SouthBridge structure is eliminated. In other instances, such as in Apple's G5 PowerMac, HyperTransport is used as an integrated, high performance I/O bus that pipes PCI, PCI-X, USB, Firewire and audio/video links through the system. In all cases, HyperTransport replaces the overlapping processor and local I/O buses of earlier generation systems with a unified, high bandwidth, low latency, and low-cost architecture that is scalable, low-cost and extensible to future product generations.

II - HyperTransport Basics

HyperTransport technology is defined by channel topology, signal electrical characteristics, and command/address/data packet protocols. Topology defines the structure of the HyperTransport link. Electrical protocols define the physical characteristics of the HyperTransport signal interface. The packet protocols define how data is organized and transferred across the HyperTransport link.

Point-to-point Link Topology

HyperTransport employs dual, point-to-point unidirectional data links – one for input and one for output – with a concise signal set using 1.2V LVDS and carrying both standard computer-based load/store data and communications-oriented packet data in HyperTransport packets and streaming channels.
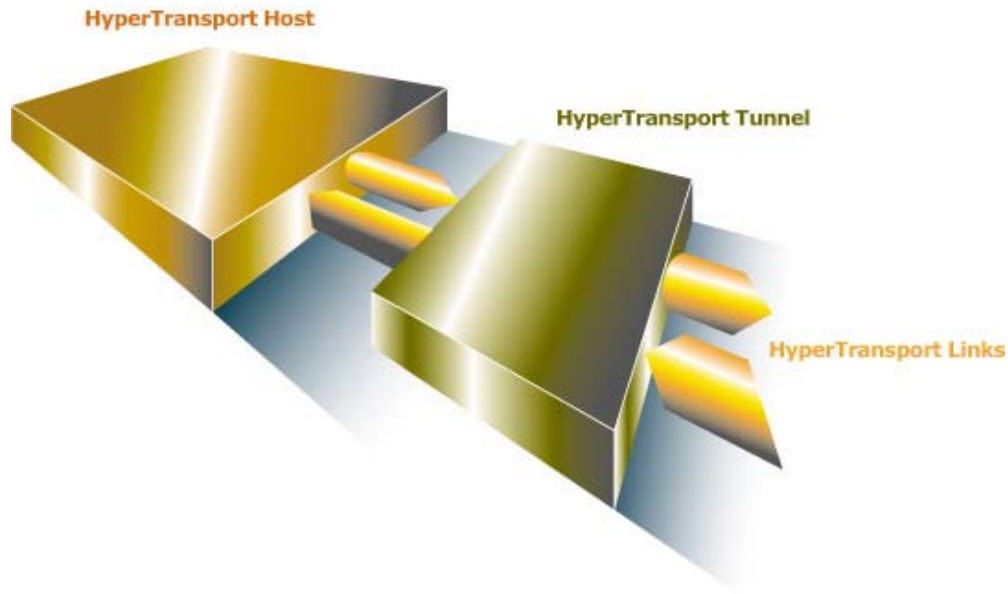
Figure 1 - The HyperTransport Link consists of at least one host (often a HyperTransport-enabled CPU) and one tunnel/endpoint. A tunnel enables the HyperTransport link to be passed from one HyperTransport-enabled device to another in daisy-chain fashion.

The HyperTransport daisy-chain topology includes a required "host" device, at least one end-point or "cave," optional "tunnel" devices that connect the link to other HyperTransport devices and optional "bridge" devices that interface with non-HyperTransport interconnect technologies. There can be a maximum of 32 HyperTransport-enabled devices in a single daisy chain, although tunnels and bridges can be employed to create tree structures where additional, but separate HyperTransport daisy chains can be linked to the first chain. HyperTransport switch devices, defined in Specification 1.05 can be deployed to create switched topology systems.

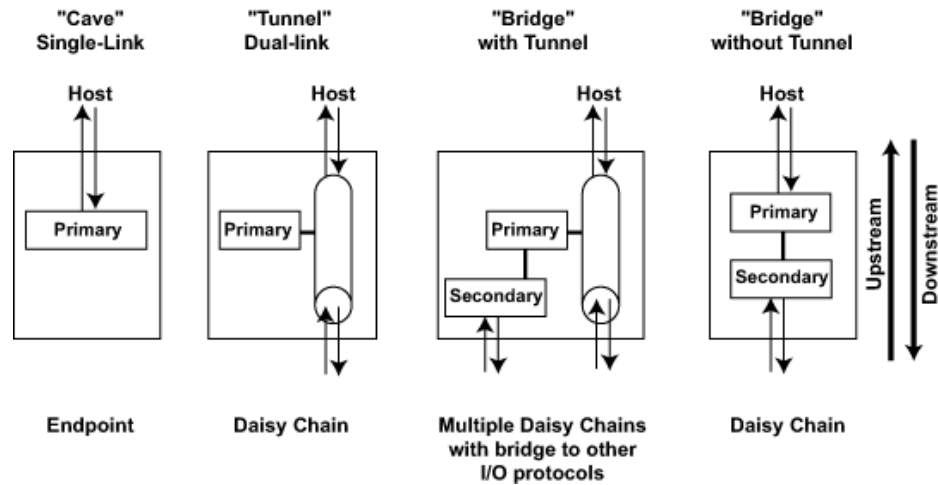**HyperTransport I/O Device Configurations**

Figure 2 – HyperTransport-enabled devices are configured to be one of four types. The host, often a CPU, is always considered the top of the link and traffic from the host is downstream while traffic to the host is upstream. A cave serves as an endpoint to a HyperTransport link. A bridge can be configured as an endpoint with a secondary interface to other interconnect technologies, or as a tunnel with an interface to other technologies.

Bridges connect HyperTransport links to other interconnect technologies, such as PCI, PCI-X, PCI Express or AGP. HyperTransport devices linked in daisy-chained topologies support PCI ordering and host-based traffic routing conventions natively, or optionally, with specification 1.1 and up, support peer-to-peer (peripheral chip-to-peripheral chip) data routing.
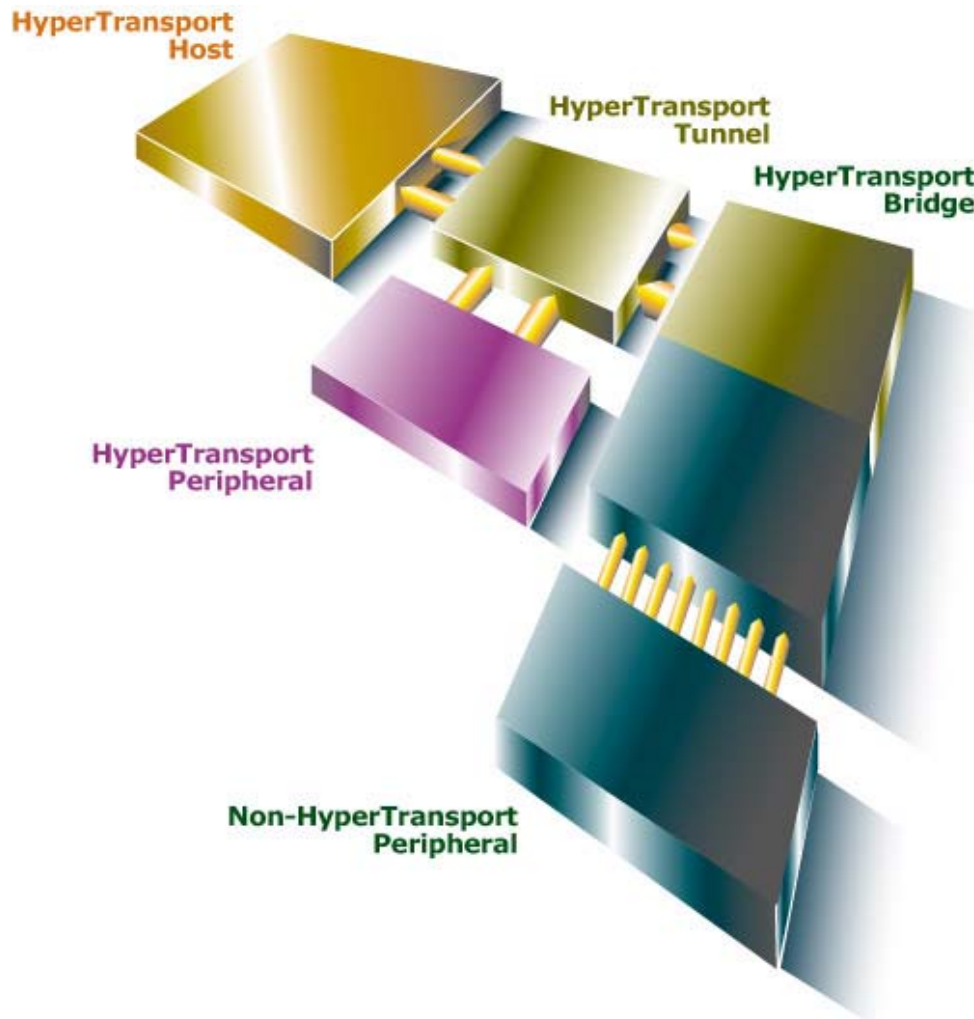
Figure 3 - The HyperTransport bridge device enables HyperTransport links to connect to other I/O technologies such as PCI, PCI-X, PCI Express or AGP devices.

HyperTransport Signals and Electrical Characteristics

The HyperTransport dual, point-to-point unidirectional links include a data path (2-, 4-, 8-, 16- or 32-bit wide), one or more clock lines (one per each 8-bit data path) and a single control line. Commands, addresses, and data are carried in packets over the data path, eliminating many sideband control signals needed in traditional multiplexed, multi-drop bus standards such as PCI and PCI-X. System level control lines RESET# and PWROK complete the required set of signal lines. LDTSTOP# and LDTREQ# are optional control lines with power

management functions. For instance, LDTSTOP# can be used to put the HyperTransport link into a virtual zero power consumption state.
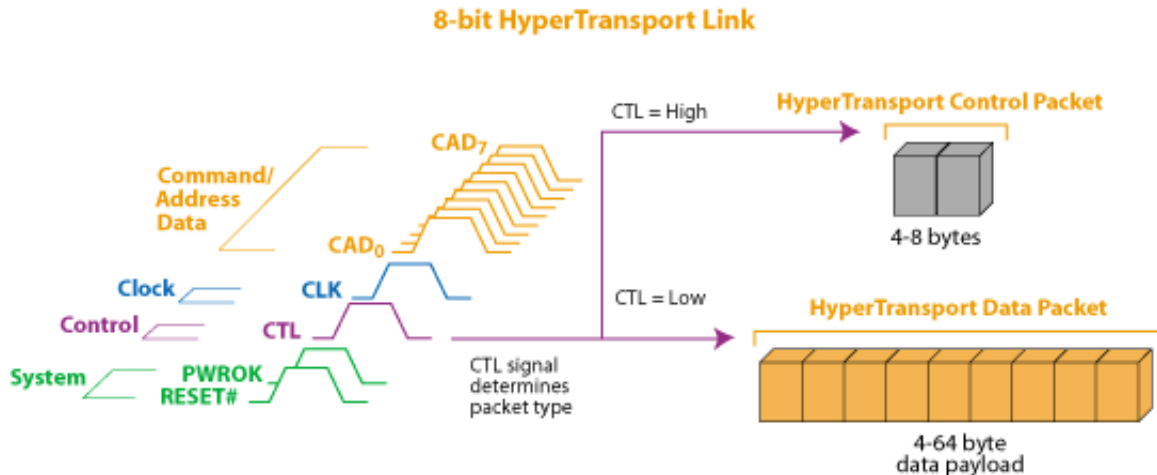
8-bit HyperTransport Link

Figure 4 - The HyperTransport Link consists of a set of command/address/data (CAD) lines (ranging from 2, 4, 8, 16, or 32 bits wide), one control line per link, and one clock line per eight bits of CAD. Command and data information carried on the CAD lines is assembled into HyperTransport control and data packets. Control packets are 4 or 8 bytes long and 4- to 64-byte data packets follow read and write control packets. The state of the CTL line determines what type of packet, control or data, is being carried on the CAD lines.
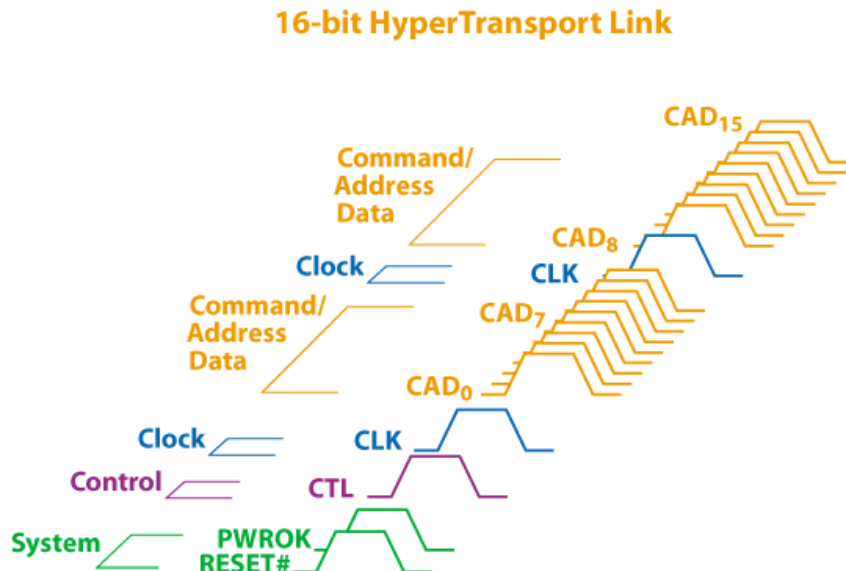
16-bit HyperTransport Link

Figure 5 – A 16-bit HyperTransport link shows that an additional 8 CAD lines require only one additional clock signal. Likewise, a 32-bit link will require only 2 further clock signals. Regardless of HyperTransport link width, the HyperTransport packet format remains the same, enabling the design of asymmetrical links (one link wider than the other).

HyperTransport Electrical Characteristics

As previously noted, the HyperTransport specification calls for 1.2V LVDS signals instead of traditional single wire, single-ended signals. Each LVDS line is implemented by means of twin wire lines – otherwise called balanced, or differential line – carrying electrical signals that are equal in amplitude and timing but with opposite polarity. The specular nature of the signals carried over the balance line prevents electrical noise within the system from affecting and potentially corrupting the signal detection process at the receiver end – noise would affect both signals in equal measure thereby nullifying the effect –, thus ensuring a high degree of architectural noise immunity, as well as a maximized transmission range. Even though there are two wire lines per signal line, requiring a second printed circuit board (PCB) trace for each data pin/pair, the multiple advantages of higher operational speeds, higher noise immunity and lower power consumption more than make up for this minor disadvantage. In addition, since the HyperTransport protocol of packet-based traffic greatly reduces the number of total signal lines required for a given bandwidth, HyperTransport architectures can be implemented with fewer PCB traces and require less signal shielding precautions as compared to traditional single-ended buses.
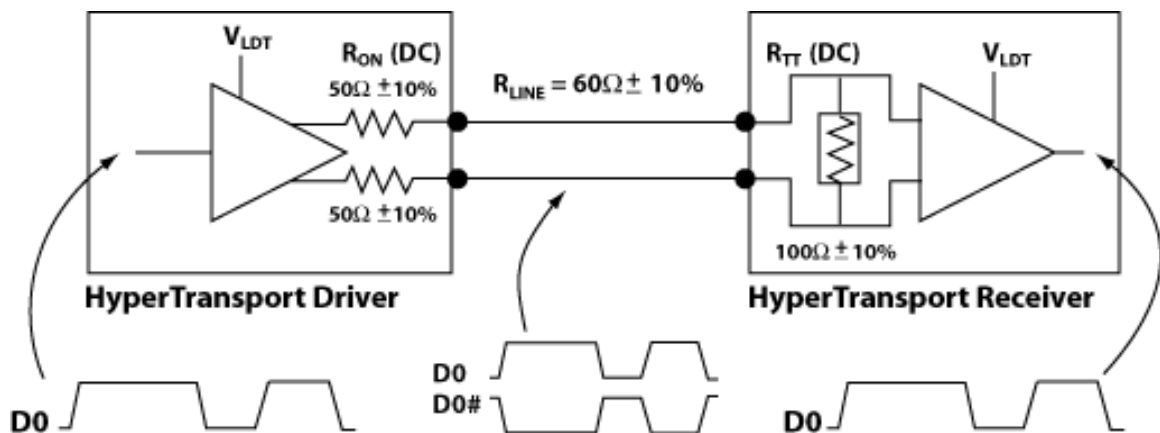

Figure 5 – The HyperTransport link uses balanced LVDS lines. Signal voltage on each wire line is symmetrical and of opposite polarity, ensuring the highest noise immunity.

A HyperTransport Specification 1.x VLDT supply voltage of 1.2 volts +/- 5% results in a differential output of only 600 mV (typical) and a differential voltage at the receiver inputs that can be as low as 200 mV. A 60 ohm differential

impedance enables the use of low-cost printed circuit board materials. LVDS traces can be up to 24 inches.

HyperTransport Specification 2.0 boosts speeds up to 2.8 Gigatransfers/second and requires that a simple signal de-emphasis scheme be used for clock speeds over 1.0 GHz. De-emphasis uses a 1 bit "history" to de-emphasize the differential amplitude generated by the transmitter when transmitting a continuous run of 1's or 0's.
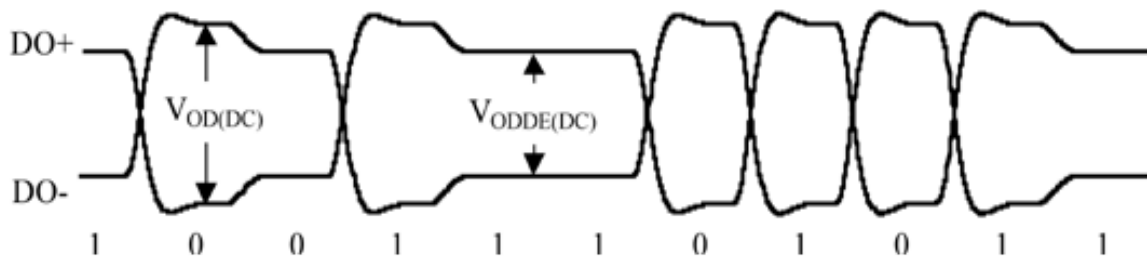

Figure 6 – A simple de-emphasis technique is used to support the 2.8 Gigatransfers/second links of HyperTransport Specification 2.0. The eye of the signal uses reduced amplitude for sequential bits of the same value. High-pass filtering on the output counteracts the printed circuit board material's low-pass filtering roll-off above 2 Gigabits/second by extending the signal eye's width and reducing the number of signal transitions.

Since the de-emphasis creates a smaller, but cleaner eye signal at the receiver end, the receiver requires a higher sensitivity.

It is important to note that HyperTransport 2.0 devices seamlessly interoperate with HyperTransport 1.0 devices. Since speed and width are automatically negotiated at system startup, links will operate at the highest rate that each end can support.

HyperTransport Packet Format

HyperTransport is an efficient data transport mechanism with the least overhead of any modern I/O interconnect architecture. Command information is carried as a control packet of four or eight bytes. HyperTransport data traffic is carried as a data packet that consists of an 8- or 12-byte header (one 8-byte control packet for writes or two control packets, one 4-byte and one 8-byte, for reads) followed by a 4-64 byte data payload. All HyperTransport information is carried in multiples of four bytes (32-bits).
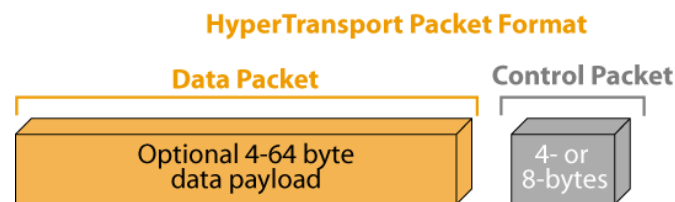

HyperTransport Packet Format

Figure 7 – HyperTransport control packets consist typically of 4 to 8 bytes of command information. With optional 64-bit extended addressing, control packets can be 12 bytes. Data packets consist of 4- to 64-byte data payloads (in increments of 4 bytes) and directly follow either 1) an 8-byte read request followed by a 4-byte read response or 2) an 8-byte write request control packet.

Command packets are 4 or 8 bytes and include all of the information needed for inter-device or system-wide communications except in the case of reads and writes, when the data packet is required for the data payload. HyperTransport writes require an 8-byte Write Request control packet, followed by the data packet. HyperTransport reads require an 8-byte Read Request control packet (issued from the host or other device), followed by a 4-byte Read Response control packet (issued by the peripheral or responding device), followed by the data packet.

HyperTransport Data Write Sequence

Write Request
Control Packet   **Data Packet**

8-bytes   4-64 bytes of data

HyperTransport Data Read Sequence

Read Request   Read Response
Control Packet   Control Packet   **Data Packet**

                                    4-64 bytes of data
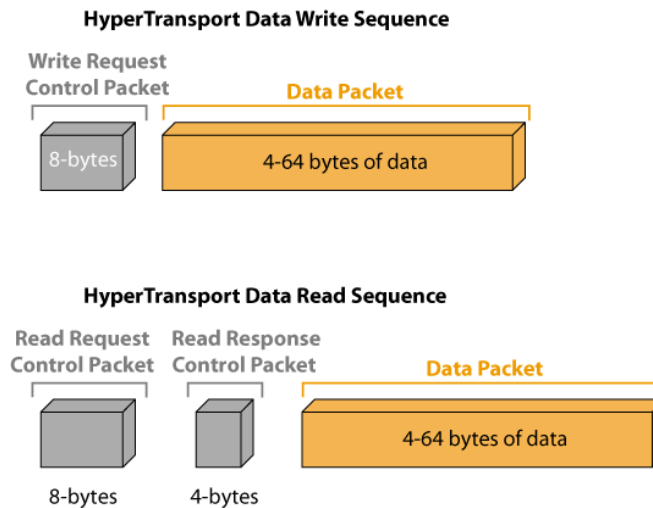
8-bytes   4-bytes

Figure 8 – HyperTransport reads and writes are very low overhead. Writes require only an 8-byte Write Request control packet followed by the data packet. Reads require an 8-byte Read Request control packet, followed by a 4-byte Read Response packet from the receiver, followed by the actual read data packet.

HyperTransport uses a single control line to determine when the link is carrying a control packet (the control signal is asserted) or a data packet (the control signal is de-asserted). This deterministic control of packet type is a significant feature of the link because the control signal can be used to insert control packets in the middle of a long data packet. This Priority Request Interleaving™ feature, unique to HyperTransport technology, contributes to the very low latency characteristics of the HyperTransport link by allowing, for example, a memory read request to be initiated in the middle of a data packet.

HyperTransport commands and data are separated into one of three types of virtual channels: non-posted requests, posted requests and responses. Non-posted requests require a response from the receiver. All read requests and some write requests are non-posted requests. Posted requests do not require a response from the receiver. Write requests are posted requests. Responses are replies to non-posted requests. Read responses or target done responses to non-posted writes are types of response messages.

With this concise, minimal set of data and control lines, and a straightforward packet format, the HyperTransport specification provides a powerful, high

bandwidth, easily implemented chip-to-chip communications channel capable of delivering up to 22.4 Gigabytes/second aggregate bandwidth for both standard computing-oriented applications (all HyperTransport specification levels) and communications-oriented, packet stream applications (HyperTransport specification levels 1.1 and above).

III - A Low-latency Solution

All of the newer chip-to-chip and slot-based architecture communications technologies, such as RapidIO, PCI Express, and PCI Express AS among others, seek to replace the multitude of low bandwidth and complex standard and proprietary processor and local I/O buses. They aim to support processors, memories, graphics and I/O devices with a single unified serial or parallel bus that exhibits high bandwidth and simplified electrical characteristics. Based on their application focus, some of these solutions pack a number of features to support slot-based communications into their specifications and introduce a number of protocol layers on top of the basic protocol to handle anticipated system or network operations. HyperTransport, in contrast, has been intentionally focused on creating a unified chip-to-chip communications channel that exhibits the lowest possible latency and introduces the lowest possible overhead in supporting packet-based data streams.

One aspect of HyperTransport's low latency capability is the parallel nature of its link structure. A single forwarded clock is used per set of 8 data path bits, enabling a very low latency point-to-point data transfer. In contrast, serial links, such as Serial RapidIO and PCI Express, eliminate the single clock signal by adding extensive clock encoding/decoding requirements at both ends of the link. This introduces significant clocking overhead, forces the addition of serializing/deserializing logic and increases latency time well beyond ideal levels for high performance, chip-to-chip communication.

A second aspect of HyperTransport's low latency capabilities is the low data packet overhead. As compared to other packet-based approaches, HyperTransport provides the lowest packet header overhead: 8 bytes for a read operation and 12 bytes for a write operation (for a read request, there is an 8-byte Read Request control packet followed by the data packet; for a write request, there is an 8-byte Write Request control packet followed by a 4-byte

Read Response control packet, followed by the data packet). This compares very favorably to PCI Express for example as shown below:

HyperTransport Packet Format

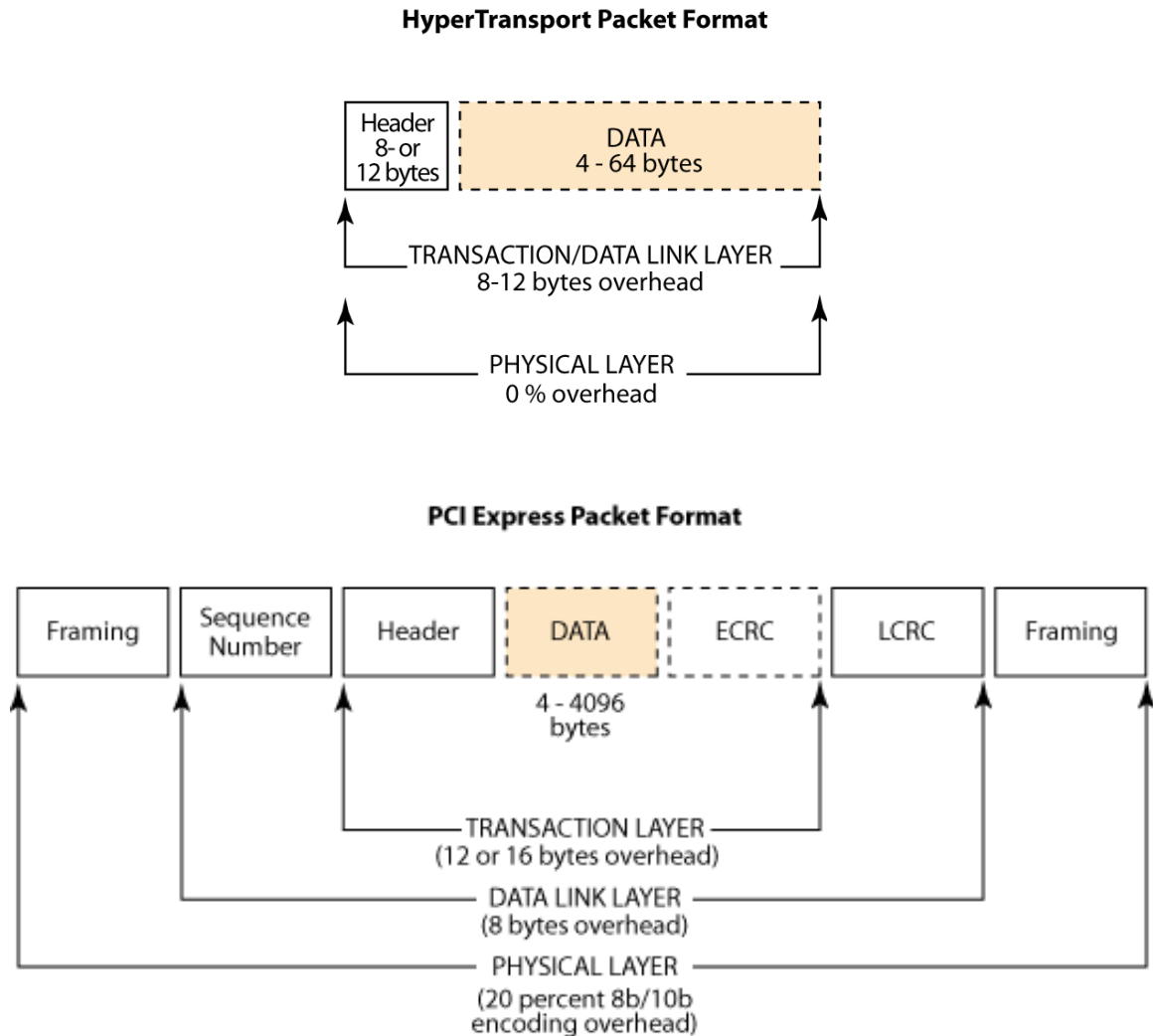**PCI Express Packet Format**

Figure 9 – Packet overhead comparison between HyperTransport and PCI Express. HyperTransport needs only an 8-byte header (control packet) per packet data payload, while PCI Express uses multiple layers of encoding with 20 to 24 bytes of overhead to move even a small command or data payload. This multi-layer overhead is on top of the 20 percent clock encoding/decoding overhead of the link serializing/deserializing circuitry.

Another aspect of HyperTransport's low latency feature is the provision of a native mechanism, Priority Request Interleaving™, or PRI, that enables a high

priority request command (only 8-byte long) to be inserted within a potentially long, lower priority data transfer. A typical use is shown in the figure below. While data transfer 1 is underway between peripheral B and the host, the need arises for peripheral A to start a data transfer from the host. Without PRI, transfer 2 would have to wait until transfer 1 completes and, should transfer 1 be the answer to a cache miss, for instance, latency for transfer 2 would become prohibitive. With PRI, a control packet is promptly inserted within transfer 1's data stream, instructing the link to initiate data transfer 2 on the other link channel concurrently with the completion of data transfer 1. This mechanism, unique to HyperTransport technology, greatly reduces latency of HyperTransport-based systems.
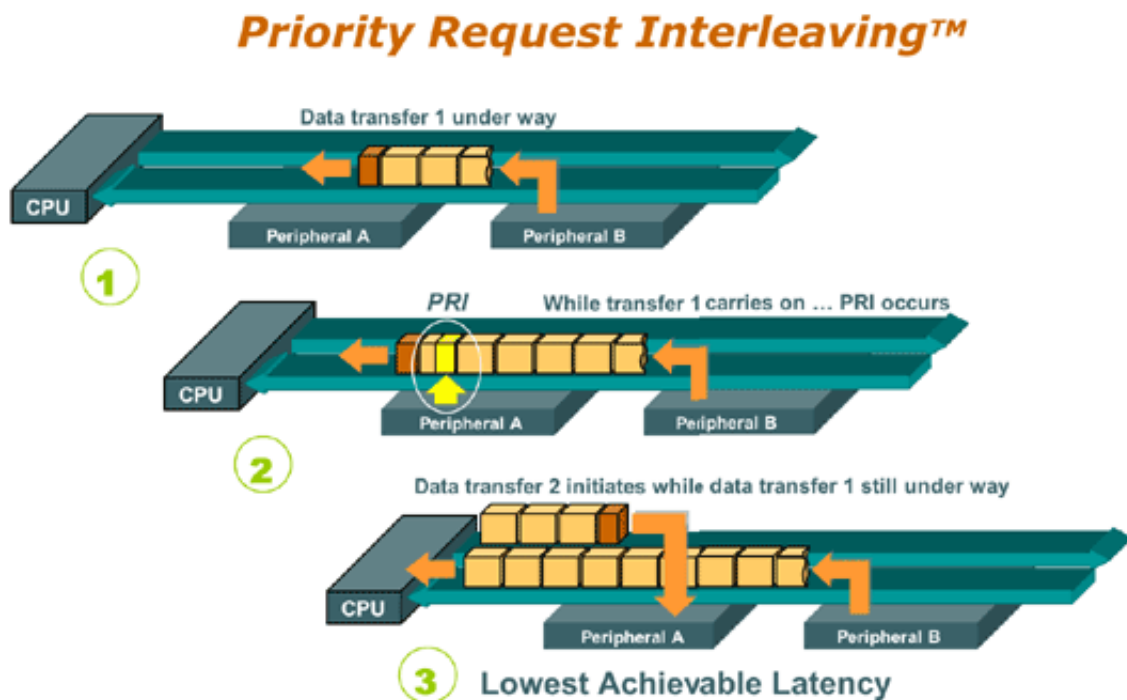

Figure 10 - HyperTransport Priority Request Interleaving. HyperTransport natively provides a mechanism that enables the insertion of commands within lower priority data traffic so that higher priority traffic can be initiated concurrently.

IV - High-Efficiency Communications Packet Handling

HyperTransport DirectPacket™ (in specification level 1.1 and above) introduced powerful communications protocols that enable HyperTransport links to carry user packet data efficiently. Computer-oriented data transfers use a "load/store" metaphor and require the communications link to instruct each attached device precisely where to store or retrieve the data in system memory. Communications technologies on the other hand, use a "channel" metaphor, where source and destination addresses are specified, data is passed to the channel in packets containing control and data information, and it is the receiver's or transmitter's responsibility to determine where data streams are to be stored. The link is only responsible for providing the source/destination, control information and data payload, and does not have to specify exact memory locations or be concerned at all with memory storage management.

HyperTransport's DirectPacket protocol is the leanest, most efficient protocol to process user packet data and to support native packet handling. DirectPacket makes no assumptions about how the system architecture needs to handle packet data. Because HyperTransport DirectPacket utilizes unused bits in the base HyperTransport packet format, there is no extra overhead for supporting user packets. This makes it possible to support user packets with far less overhead than alternative approaches. This combined with HyperTransport's already low overhead results in HyperTransport's DirectPacket being able to move user packets from point A to point B more efficiently than any other combined load-store plus packet handling protocol.

The assumption of other technologies supporting native packet handling is that the basic specification defines both board-level packet movement and the entire system architecture. Consequently, there are several layers of protocol management that over-burden even the most basic of packet movement commands. This adds additional layers of complexity and overhead to simple data transfers. HyperTransport takes the opposite approach, defining just the level of protocol required to move user packet data from point A to point B and leaving the rest of the system architecture (if needed) to the OEM to implement.

For more information on the HyperTransport DirectPacket protocols, please refer to the other white papers available at www.hypertransport.org, in particular, the

"HyperTransport™ I/O Technology DirectPacket™ Specification: Efficient User Packet Handling Supports Streaming Communications" white paper published in 2004.

V – A HyperTransport High Performance Example

Among the benefits that HyperTransport brings to board-level design are scalability, high bandwidth, low latency and ease-of-implementation. For example, the multiprocessor system shown in the diagram below connects multiple closely coupled processors and memories, legacy I/O devices such as PCI and PCI-X and high performance cluster linking technologies such as InfiniBand and network links such as Ethernet. In the past, such as system would be based on complex, parallel multidrop buses with competing clock circuitry, numerous printed circuit board traces and interbus bridges and links. Maintaining system peak throughput would be problematical at best and system debugging would be a very time-consuming proposition.

In contrast, using HyperTransport technology greatly simplifies system design. One continuous HyperTransport link integrates not only interprocessor communication, but brings both low and high bandwidth devices into an integrated I/O stream accessible by all computing elements. Using wide links between processors and narrow links between I/O devices gives the system designer the ability to apply chip-to-chip bandwidth precisely where needed, without having to run expensive links to every node. With this system architecture, a single processor system can use the same system architecture as a four-way processor system, enabling the benefits of high volume, low cost production. HyperTransport-enabled architectures also allow the linking of multiple multiprocessor clusters in an efficient, low-cost manner. Finally, through the use of HyperTransport-to-PCI or HyperTransport-to-PCI-X bridges and its full PCI compatibility, HyperTransport technology enables full support of legacy PCI-compatible I/O subsystems without the need for system redesign.

Only HyperTransport provides the low latency link that can enable one processor to access another processor's memory in real-time and to maintain fast processor-to-processor links.
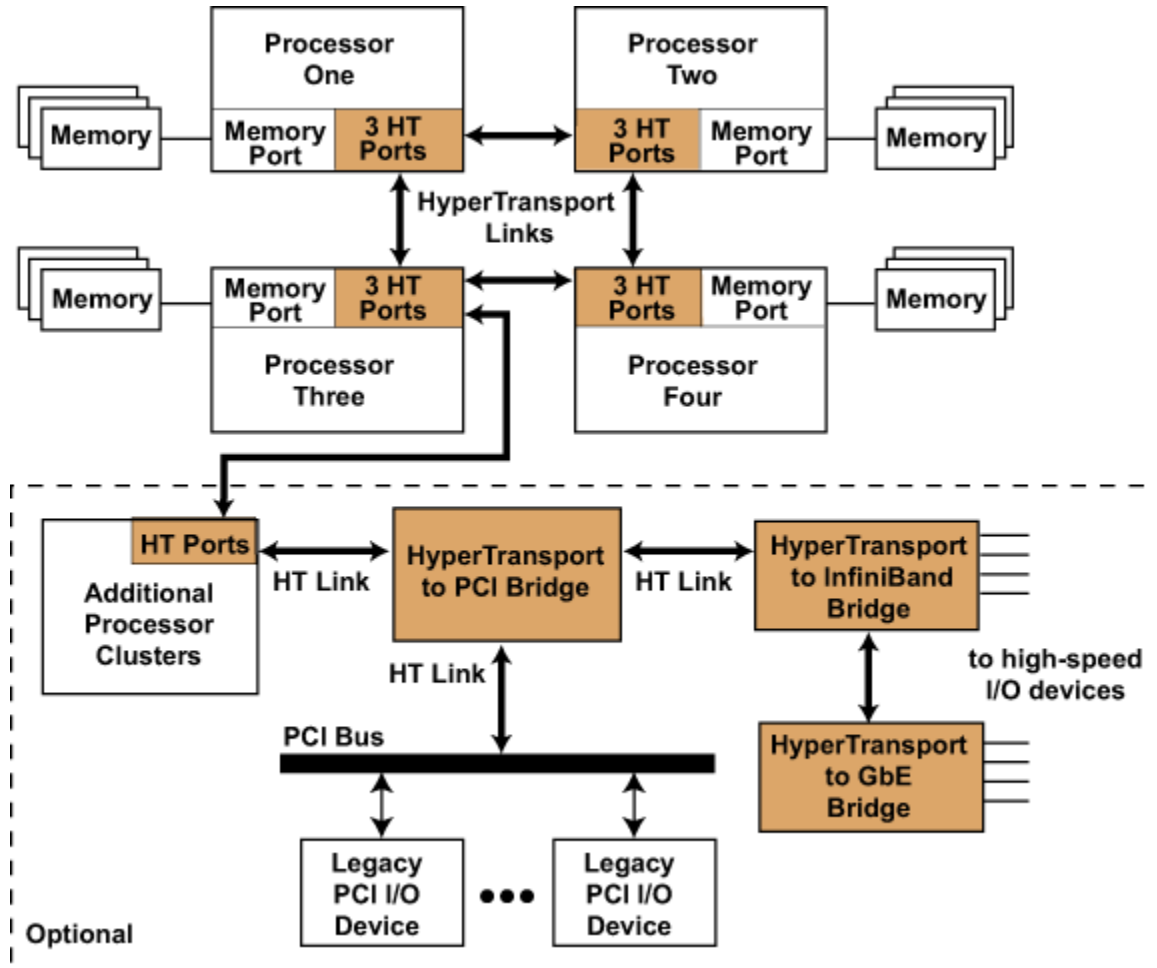
Figure 11 – HyperTransport can be used to integrate processor-to-processor, processor-to-memory and processor-to-I/O data streams as shown in this HyperTransport-based multiprocessor system. Bridges can easily link HyperTransport links to legacy or emerging I/O technologies.

VI - Conclusion

HyperTransport technology is a powerful board-level architecture that delivers high bandwidth, low latency, scalability, PCI compatibility, and extensibility. When processor-native, HyperTransport provides an integrated front-side bus that eliminates custom/proprietary buses and the need for additional glue logic. As an integrated I/O bus, HyperTransport eliminates the need for multiple local buses, ultimately simplifying overall system design and implementation. As a link to legacy PCI and PCI-X subsystems, HyperTransport extends the life of those popular systems and as a link to emerging, high-speed serial buses such as PCI

Express, it becomes a board-level system architecture with ample bandwidth capable of supporting off-board and system-to-system communications.

HyperTransport technology has been deployed in millions of devices used in market leading products such as Microsoft's Xbox, Apple's Power Mac G5, Cisco's routers, IBM's and Sun Microsystems's servers, notebooks and Tablet PC's based on Transmeta's Efficeon-processor, all AMD's Athlon64- and Opteron-based PCs and servers, as well as Cray's and IBM's supercomputers. 2003 industry estimates from market analyst firm IDC projected 30 million HyperTransport port shipments in 2003, rising to over 200 million ports shipped in 2006.

HyperTransport technology has proven to be a powerful, integrated chip-to-chip interconnect technology providing performance benefits that exceed present industry requirements and is ideally positioned to serve future market needs throughout the widest range of applications and industry segments.

End of White Paper HTC_WP02

The HyperTransport Specification Progression
The HyperTransport Specification has evolved since its first introduction in 2001. The current HyperTransport specification is Release 2.0 and it is backward compatible with previous versions of the specification (1.10, 1.03 and 1.05). This means devices in mixed chains can communicate with each other. Of course, a device can only use the specification it supports. Member companies have the option to produce devices to any level of the HyperTransport specification.

The HyperTransport specification compliance of a given device determines its ability to interact with other HyperTransport-enabled devices. During system initialization, the HyperTransport devices communicate with each other and determine speed, width and the highest level of protocol specification that each can support. The link is then set accordingly. A link will be consistent at a given speed, but can support asymmetrical width links and a mix of devices operating at different protocol specification levels.

HyperTransport Specification Release 1.03 – Defines the basic electrical and protocols for HyperTransport including 1.2V LVDS signaling, unidirectional links, 12.8 Gigabyte/second bandwidth, 800 MHz maximum clock frequency (dual data rate transfers for a 1.6 Gigatransfers/second), PCI configuration registers, PCI enumeration protocol, host-based routing (PCI compatible).

HyperTransport Specification Release 1.05 – Adds four capabilities to the basic HyperTransport specification: a HyperTransport switch definition, a larger address space (from 40-bits to 64-bits), an increase in outstanding concurrent transactions from 32 to 128, and enhanced PCI-X 2.0 interworking support.

HyperTransport Specification Release 1.1 (DirectPacket™) – Adds four communications-oriented capabilities: native packet handling, robust error retry protocols, peer-to-peer routing and extra Virtual Channels for streaming data.

HyperTransport Specification Release 2.0 – Adds three speed grades of 2.0, 2.4 and 2.8 Gigatransfers/second for a maximum aggregate bandwidth of 22.4 Gigabytes/second. Electricals are backward compatible, preserving investments made in 1.x devices. Also adds mapping to PCI Express, expanding existing interoperability with PCI and PCI-X.

About the HyperTransport Consortium

The HyperTransport Technology Consortium is a membership-based non-profit organization in charge of managing and promoting HyperTransport Technology. It consists of over 40 member companies including major industry players in the personal computer, server, network equipment, silicon IP, software and supercomputing markets. Founding members include Advanced Micro Devices, Alliance Semiconductor, Apple Computer, Broadcom Corporation, Cisco Systems, NVIDIA, PMC-Sierra, Sun Microsystems, and Transmeta. Membership is open to any company interested in leveraging the HyperTransport technology and is based on a minimal yearly fee that includes the right to royalty-free use of HyperTransport technology and Intellectual Property. For more information, please visit: http://www.hypertransport.org/org_join.html.

The HyperTransport-enabled product portfolio includes tunnel, bridge, and graphic chips; programmable-logic devices; security processors; IP cores; BIOS software; verification and test tools; and training courses and an architecture reference manual. A full product listing can be found at: http://www.hypertransport.org/featuredproducts/products.html.