8 Lane, 5 Port PCI Express® Switch Performance Report

89PES8T5A

Notes

Overview

This document presents performance measurements and benchmarking results for IDT's 89PES8T5A 8-lane, 5-port peripheral chip, a member of IDT's PRECISE™ family of PCI Express Switching solutions. The PES8T5A has one x4 upstream port and four x1 downstream ports. The switch is compliant with PCI Express (PCIe®) base specification revision 1.1.

The test vehicle for the PES8T5A is the evaluation board IDT89EBPES8T5 which hosts the PES8T5A. Accompanying the throughput and latency performance metrics are descriptions and methodologies outlining the test setup and procedures.

The nature of tests and the equipment used for these tests varies significantly across the spectrum of tests performed. In the interest of readability and searchability the document is divided into various sections. Each section represents a single test suite that employs a single test setup. A single test suite is capable of highlighting several features of the switch device under test.

Section I provides some insight into issues that can affect the performance of a PCIe device. This includes overhead derived from the protocol, as well as the architectural decisions made while implementing the PCIe device.

Section II describes the performance of the PES8T5A with Gigabit Ethernet endpoints attached to its downstream ports. Bidirectional performance comparisons with and without the PCIe switch in the traffic path are provided for both Windows-XP and Linux environments. SmartBits™ SMB600 is used to generate controlled Ethernet traffic which is looped back between the two GE NICs.

Appendix A gives a brief introduction to the SmartBits traffic generator and analyzer and the SmartFlow™ test software package used in conjunction with this test equipment.

Revision History

August 3, 2007: Initial version.

SECTION I: PCIe Performance Basics

The PES8T5A primarily serves the purpose of high-performance I/O connectivity expansion in a typical system. Simply put, the PES8T5A uses one existing PCIe port in a system and offers four ports in its place. Given that nothing ever comes for free, it is presumed that the addition of a port has some "cost" associated with it in the form of real estate on the system board, power/heat, design complexity, support circuitry/devices (clocks, hot plug controllers, EEPROMs, power regulators, jumpers, etc.), signal integrity, or adverse effects on throughput/latency. All but the last item in this list are unavoidable to some extent. It is the impact on throughput and latency (system performance in general) that is the least intuitive to predict without a reasonable understanding of the system and switching device architecture, the usage model of the switching device, and some basic understanding of the PCIe protocol itself. In this section, some of these elements are introduced to the users of the PES8T5A, specifically those users who are new to PCIe and switching. Advanced users of PCIe and switches may skip the reminder of this section.

What Does Performance Mean?

PCIe switch performance can mean different things to different users. The following is an introduction to some basic terminology.

"Raw bits" refers to the total number of bits that go through the switch in any given period of time, regardless of function, source, or destination. The PES8T5A is designed to handle 2.5 Gigabits per Second of raw throughput in each direction on each of its lanes. This results in (2.5 Gbps) x (2 directions) x 8 (lanes) = 40 Gbps of raw switching capacity.

"Switch throughput" is calculated as the useful bits passing through the switch per second after subtracting the 8b/10b encoding/decoding overhead from the total raw bits. Discussion of the 8b/10b mechanism is beyond the scope of this document. It is sufficient to note that two out of every ten bits passing across a PCIe link do not contribute to any meaningful user data and are, therefore, subtracted from the throughput measurement. It must also be noted that this overhead is a feature of the PCIe protocol itself and is not uniquely associated with a switch device per se. For the PES8T5A, the "switch throughput" becomes (40 / 10) x 8 = 32 Gigabits per second.

"Switch utilization" is the "switch throughput" less the overhead associated with the PCIe protocol infrastructure. Examples of this type of overhead traffic are TLPs containing no user data (messages related to interrupts, errors, hot plug, power management, vendor defined messages, etc.) and eight types of DLLPs (Ack/NAK, flow control, etc.). This overhead is variable in nature and can sometimes be fine-tuned to meet system requirements by modifying the switch settings, such as the ratio of ACK/NAKs to total packets, etc. In general, however, expect this overhead to be about 15% of switch throughput for the majority of real life systems. "Switch utilization" brings us one step closer to estimating how much user data goes through the switch in a given period of time, but there is one more overhead to consider.

Every data packet is preceded and followed by a variable number of bytes. These bytes include the frame K-code, sequence number, TLP header, ECRC, and LCRC. Once this "framing" overhead (see Figure 1) is deducted from the "switch utilization" number, the resulting performance metric is called the "switch efficiency".

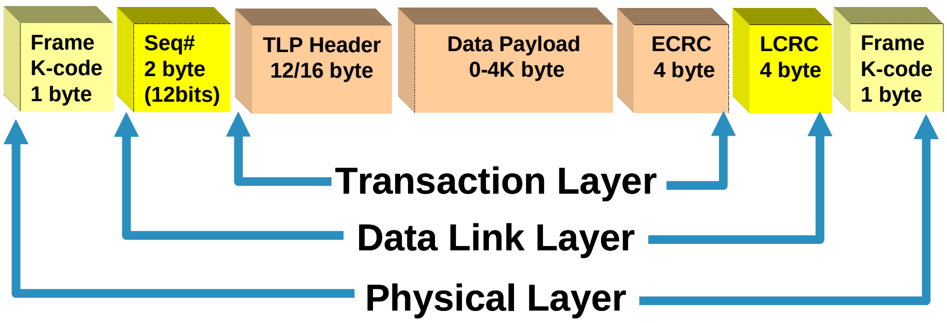

Figure 1 Framing Overhead in a Typical Transaction Packet

A different indicator of the performance of a switch is the switch "latency", which is defined as the time spent by a bit within the switch from the moment it enters the switch to the moment it exits. The latency number, typically low hundreds of nanoseconds, can be affected by several parameters including, but not limited to, switch architecture, traffic pattern, state of the switch in terms of loading, width of the ingress port, and width of the egress port.

Impact of Architecture on Switch Performance

Two high-level architectural decisions which will have the biggest impact on switch performance are "how" the data is forwarded from one port to the other within a switch and "when" the data is forwarded. System designers must make these decisions at the very beginning of the design process. The architectural choices available for the "how to forward" question are: Shared bus, Crossbar, and Shared memory, or a hybrid of some combination of the above. The PES8T5A is implemented in a shared bus style architecture. Explanation of these different types of switching architectures is beyond the scope of this document.

The architectural choices available for the "when to forward" question are: Cut-through (start forwarding a packet while it is being received) or Store and Forward (start forwarding only after an entire packet is received). The PES8T5A uses the Cut-through forwarding method.

There are several other micro-architectural features or implementation details of a switch that can also have noticeable impact on the performance of a switch. Discussion of the relationship between a feature choice and its impact on performance are beyond the scope of this document. It is relevant to note that several implementation details, such as the transmit retry buffer sizes, ingress buffer sizes, flow control mechanism, allowable maximum payload size (MPS), and controllable frequency of DLLPs including flow control updates and ACK/NACK, have an impact on the performance of the switch. Specifications related to these implementation details for the PES8T5A are found in the 89HPES8T5A User Manual, available by contacting IDT.

SECTION II: GE Throughput Measurements

The goal of this set of tests is to demonstrate the behavior of the PES8T5A with Gigabit Ethernet endpoint devices. Test results are obtained both with and without the PES8T5A device in the data path, so as to measure the impact of the switch on data throughput.

Hardware Setup

Following is a summary of the hardware used in the system:

- Tyan Thunder K8QE (S4885)
  - Four (quantity) - AMD Opteron 852 CPU's (64-bit)
  - 4GB of DDR-RAM
  - 4 available PCIe slots - Two PCIe x16 and Two PCIe x4
  - Fedora Core 3 - Linux Kernel 2.6.9 - 1.667 SMP
  - Windows 2003 Server
- IDT PES8T5A - PCIe upstream (x4), PCIe downstream (four x1)
  - Max Payload Setting 128 bytes
- Broadcom BCM5751 NetXtreme® Gigabit Ethernet Controllers

Figure 2 is a logical representation of the hardware setup used for GE throughput measurements with the PES8T5A. In this case, only one PCIe slot on the motherboard is used.

NICs 1 through 4 are plugged into the downstream port slots of the PES8T5A evaluation board (89EBPES8T5) hosting the PES8T5A switch. The upstream port of the PES8T5A is at the x8 edge connector of the PES8T5A evaluation board and is plugged into a x16 port slot of the motherboard. The PES8T5A switch uses one PCIe slot on the motherboard and creates a fan-out of two slots where four GE NIC endpoints can be used in this system.

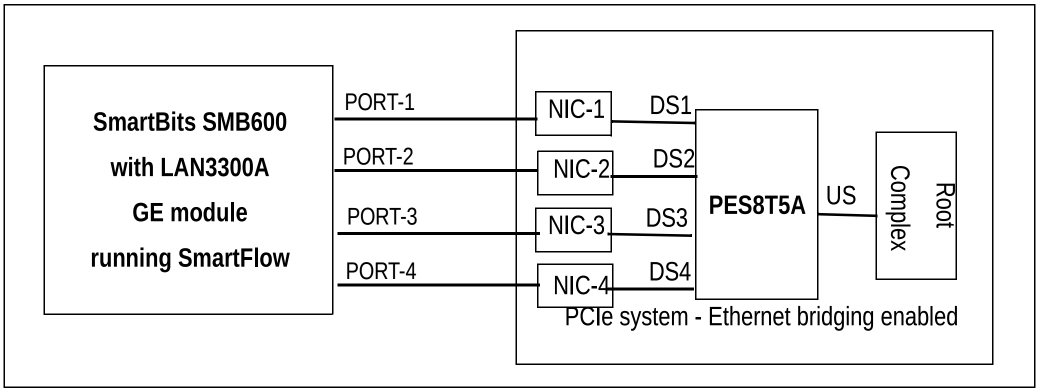

Figure 2 GE Throughput Measurement Setup with the PES8T5A

Figure 3 is a logical representation of the hardware setup used for GE throughput measurements without the PES8T5A in the data path. In this setup, two PCIe slots on the motherboard are used by the endpoints since the fan-out provided by the PCIe switch is no longer available. The GE NIC cards are plugged directly into the PCIe slots on the motherboard.

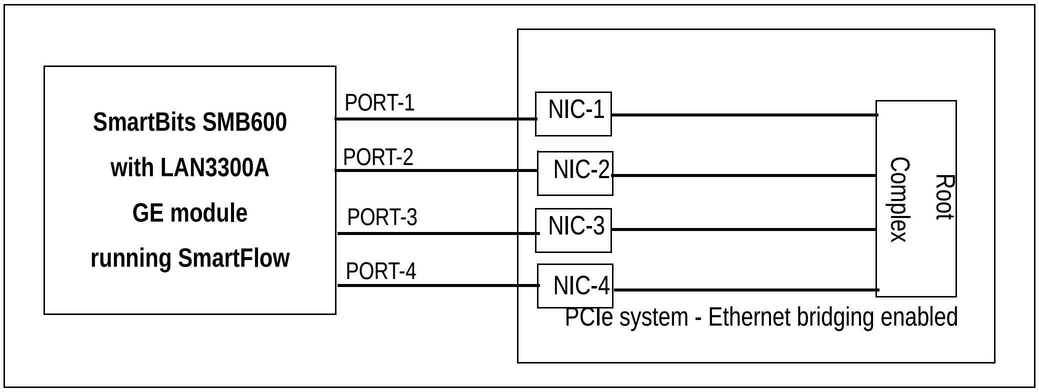

Figure 3 GE Throughput Measurement Setup without the PES8T5A

Software Setup

The SmartBits 600 Gigabit Ethernet traffic generator is controlled by the SmartFlow software package to generate and sink Ethernet traffic in a loopback mode. Details related to SmartBits setup can be found in Appendix A. The PCI Express-enabled server system is controlled by the operating system (Linux or Windows) and implements bridging of Ethernet traffic from one Ethernet port to another.

Test Procedure and Methodology

Each port of the SMB600 transmits Ethernet packets of predefined sizes targeted to another port. Each packet transmitted by Port 1 travels through the corresponding NIC in the PCIe system, through the PCIe switch, if present, through the memory in the PCIe system, gets bridged over to the other NIC via the PCIe switch, if present, and returns to Port 2 of the SMB600. Packets starting at Port 2 of the SMB600 traverse the exact opposite path described above. Ports 3 and 4 follow a similar relationship. Combined throughput measurements of these two flows for each packet size, with and without the PCIe switch in the path, are recorded in Tables 1 and 2 below. No data loss is permitted along the entire data path in either direction.

Results

	Throughput in Megabits/Second						
Packet size (bytes)	**64**	**128**	**256**	**512**	**1024**	**1280**	**1518**
Mbits/S Without PES8T5A	85	132	220	484	850	991	1412
Mbits/S With PES8T5A	85	152	265	484	840	980	1350

Table 1 Throughput versus Ethernet Packet Size — Windows-XP

	Throughput in Megabits/Second						
Packet Size (bytes)	**64**	**128**	**256**	**512**	**1024**	**1280**	**1518**
Mbits/S Without PES8T5A	355	597	1047	2140	3944	4000	4000
Mbits/S With PES8T5A	355	597	1050	2160	3662	3780	3850

Table 2 Throughput versus Ethernet Packet Size — Linux

Analysis

The goal of this test was to show the effect of the PES8T5A PCIe switch on Ethernet traffic throughput. A quick review of the results reveals that the bridging performance of the operating system determines how stressful the test will be for the PCIe switch under test. It is clear that Linux offers better Ethernet bridging performance and stresses the switch more than Windows-XP does.

In some cases, the presence of the PES8T5A actually increases the throughput. The buffering capabilities of the PCIe switch allow the endpoints and root complex to send larger bursts of data than they would have been able to send otherwise. Consequently, they are more frequently transmitting data and less frequently waiting to transmit. Another benefit is the coalescing of ACK messages, which frees the return path for data transmissions.

Appendix A Introduction to SmartBits and SmartFlow

Note: Information contained in this section pertains to tools offered by a third party. The information is provided for the convenience of the reader and is not guaranteed to be complete or accurate.

The following document was used for reference while generating this text: Spirent Communications, Inc., 2005. “Introducing SmartFlow.” SmartFlow User Guide (5.0).

SmartFlow is a performance analysis tool to test Layers 2, 3, and 4 on Class of Service devices and networks built with Class of Service priority strategies. SmartFlow allows the setup of multiple flows of IP frames to simulate network traffic and measures latency, frame loss, and throughput. It presents results in charts and tables that include measurements for latency, frame loss, and standard deviation of flows. Results can be tracked by priority or by type of traffic to determine the effect a prioritizing Class of Service device has on the network.

Since our primary goal was to measure throughput through the PCI Express switch, we used the SmartFlow Group Wizard to simply generate flows, track them, and group them. SmartFlow is used in conjunction with a Spirent Communications SmartBits chassis and at least two SmartMetrics or TeraMetrics (or TeraMetrics-based) ports.

SmartFlow includes the following tests:

- Throughput
- Frame Loss
- Latency
- Latency Distribution
- Latency Snap Shot
- Smart Tracker

Below is a general description of the tests that were used for our measurements.

Throughput

Measures the maximum rate at which frames from flows and groups can be sent through a device without frame loss. A sequence of transmissions from one port on the SmartBits chassis to the other port on the chassis is setup. This traffic flows through the device under a test (PCI Express switch) which has Ethernet NICs connected to its downstream ports. An OS-based bridge is created between these two NIC, causing traffic entering one NIC to get forwarded to the other NIC. Bidirectional traffic is used, and each test consists of several sequential transmissions of Ethernet packets varying in size from 64 bytes to 1518 bytes with each type of packets getting transmitted in a single flow for several seconds at a time.

SmartFlow and SmartFlow Demos are available at **support.spirentcom.com**. Path: Self Service Tools –> Download Software Updates –> All Software –> SmartBits –> Applications or Demo. It is necessary to obtain a support account from Spirent to login to this site.