TECHNOLOGY®
THE I/O INTERCONNECT SOLUTION

Read Pacing

Pace Read Requests to Optimize Fan-out Applications

White Paper

October 2007

Website: www.plxtech.com
Technical Support: www.plxtech.com/support

Pace Read Requests to Optimize Fan-out Switch Applications

Fan-out is the most common use of PCIe switches. In the DMA I/O model prevalent in workstations and servers, DMA controllers in the I/O device endpoints both write blocks of data to host memory and read from it. The host connection is a point of aggregation that is usually wider than any of the endpoint connections without necessarily being as wide as the sum of the widths of all them. If not, then congestion and bandwidth sharing are primary concerns. Even in the ideal case of host bandwidth equaling or exceeding the sum of all the devices' bandwidth, read completions are often delivered to an endpoint faster than it can consume them, leading to congestion and to the starvation of other endpoints.

The PCIe specification prohibits endpoints from flow controlling completions. An endpoint is required to reserve buffers in advance for all the read data that it requests so that it can consume the data from the PCIe link at wire speed when it returns. This eliminates queuing in the interconnect when the source and the sink are the same bandwidth but doesn't begin to address the problem when, as is so often the case, the source (host) has a wider, faster connection than the endpoint.

The potential problem is compounded by the observed behavior of root complexes and the endpoints themselves. RCs typically service read requests in order (FIFO) instead of round robin among queued requests from different devices to avoid choking them. Endpoint designs often spring from a PCI legacy where it was necessary to read ahead very aggressively in order to gain a fair share of bandwidth. What can happen is that a device with a narrow PCIe link can request a large block of data from the host. This set of read requests blocks all other read requests in a queue at the RC. When served, the resulting completions back up in the switch, blocking other downstream completions until sunk by the endpoint. This throttles the downstream throughput of the RC's link to the often fractionally slower rate of the endpoint link (see Figure 1).

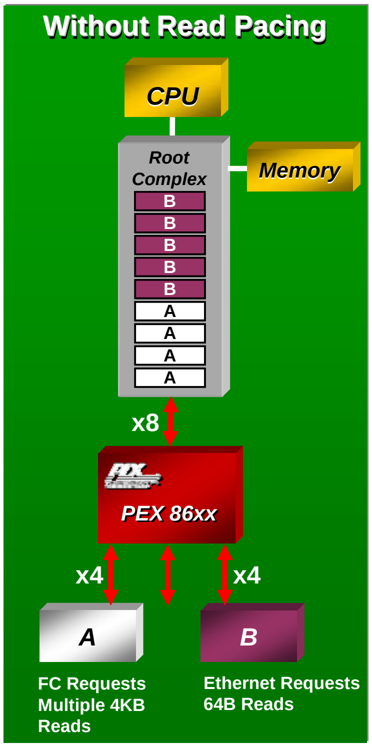

Figure 1

To combat downstream congestion, the user can try to configure the read behavior of the endpoints. Unfortunately, no PCIe architected mechanisms exist for traffic shaping or rate limiting. The desired device knobs often don't exist. Ultimately, the only thing that one is guaranteed to be able to do is to reduce the maximum read request size. This may help but is not a complete solution.

PLX has implemented the Read Pacing feature in its Gen 2 ExpressLane switch family (PEX 8600) to solve the problem described above. The PCIe specification allows posted writes and completions to bypass read requests. In read pacing, excess upstream read requests are delayed in the switch to avoid both blocking other device's read requests in the RC and to limit completion queue size in the switch. Excess requests are for an amount of data in excess of that required to mask the roundtrip latency between the endpoint and the RC. Upstream request flow is metered so that completions arrive at the rate at which they may be sunk. A small completion queue is allowed to develop but never so much as to block completions to other devices.

All the user needs to do is enable the read pacing feature; it is disabled by default. While some tunable thresholds are provided, the defaults automatically adapt to link-width. Simulations have shown that the Read Pacing feature does not reduce the throughput of the "paced" device but can significantly increase the throughput of other devices previously victimized by it.

Conclusion

In host centric applications, PLX's Gen 2 ExpressLane switch family's (PEX 8600) Read Pacing feature enables fair allocation of upstream port bandwidth between I/O devices to avoid one port occupying the read bandwidth of the upstream port. Additionally, it helps avoid head of line blocking between ports and hence starvation of ports. With Read Pacing, each port gets its share of memory read requests sent to the root complex and fair usage of shared buffers within the switch (see Figure 2).

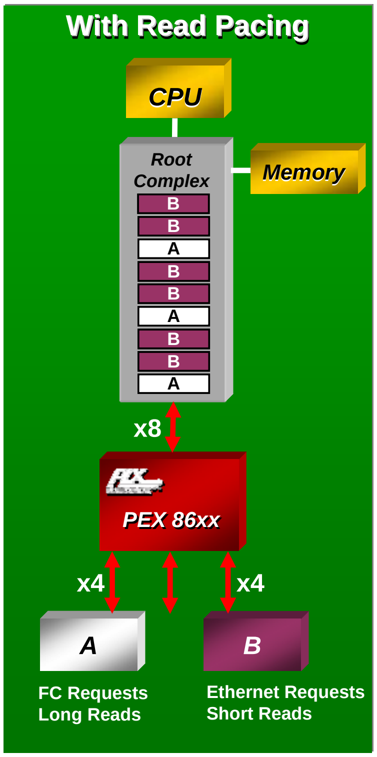

Figure 2