1.0 Summary

The InfiniHost III HCA introduces a “mem-free” mode of operation, which eliminates the need for dedicated HCA context memory. In this mode, HCA control information (connection context, translation tables, etc.) is stored in system memory. InfiniHost III Ex includes substantial context caches; memory accesses for HCA context are issued only if internal caches do not satisfy the request.

Due to highly-pipelined operation of the HCA and high hit rate of the HCA control access to internal caches, the performance impact of mem-free mode is minor. Further performance improvements of mem-free mode will be available in future FW and driver releases.

This memo compares initial performance results of an HCA with context in attached memory versus a mem-free mode.

2.0 Basic Low Level Tests

2.1 Basic Bandwidth Test

When HCA context accesses are served by internal caches, mem-free mode delivers similar results as using local memory. When HCA context accesses cannot be served by internal caches, highly-pipelined operation of the HCA overlaps cache miss processing and wire delay of the subsequent packet.

Figure 1 and Figure 2 on page 2 show bandwidth test results for SEND and RDMA-WR tests respectively.

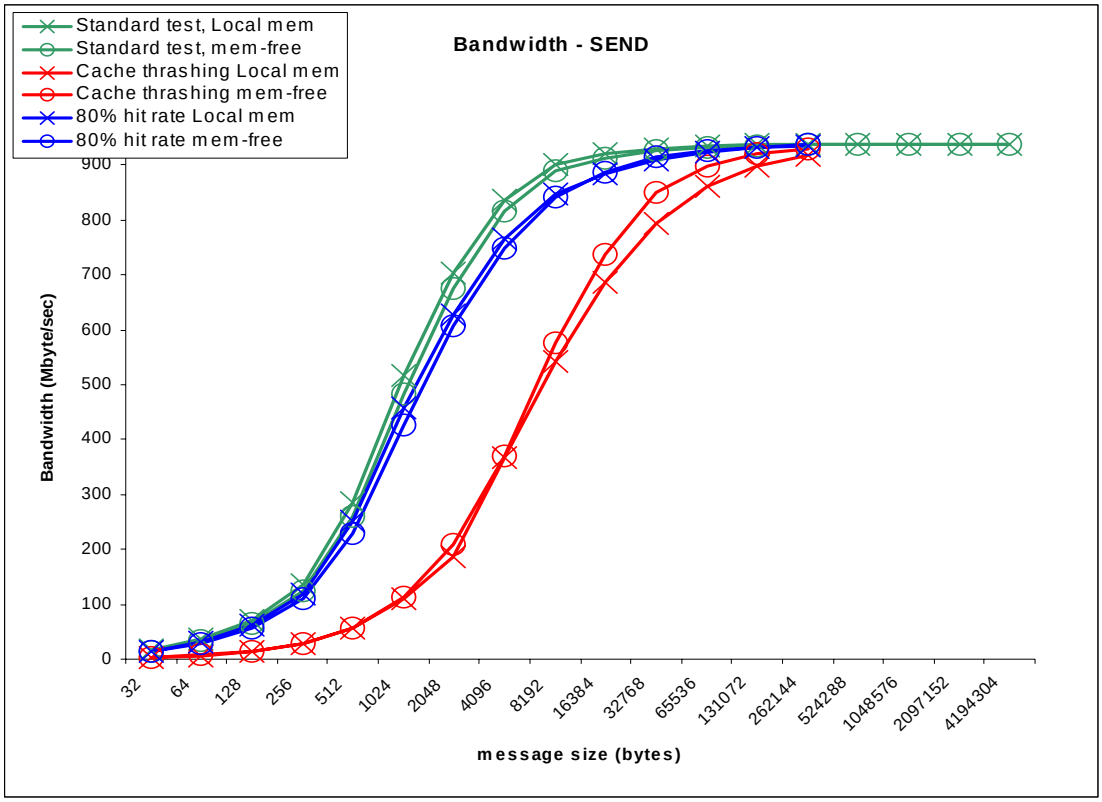

Figure 1. Bandwidth as a Function of Message Size - SEND

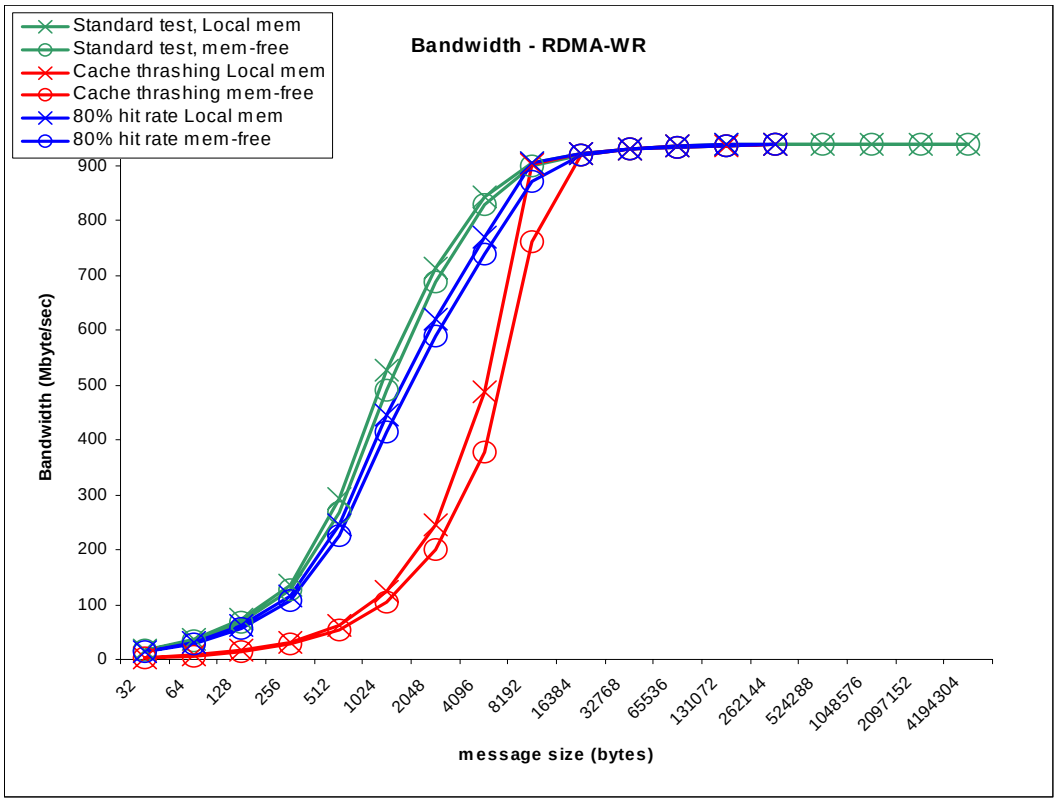

Figure 2. Bandwidth as a Function of Message Size - RDMA-WR

2.2 Basic Latency Test

Mem-free operation has no impact if HCA context access hits internal caches. If access to the HCA context misses the internal caches, an external memory access is generated by the device. The read latency from system memory across the chipset is measured to be about 200nSec higher than a read from HCA-attached memory, which bounds the latency impact of mem-free operation. Figure 3 below and Figure 4 on page 4 show the latency of mem-free and local memory configurations for various cache hit rates. The cache thrashing case (red lines) is an unrealistic worst-case scenario, where all control accesses miss the internal caches (QP Context cache for both send and receive operations and translation caches for all memory access operations). Under high-load conditions, context caches hit rate is measured to be about 80%, which is illustrated by the blue graph on the chart.

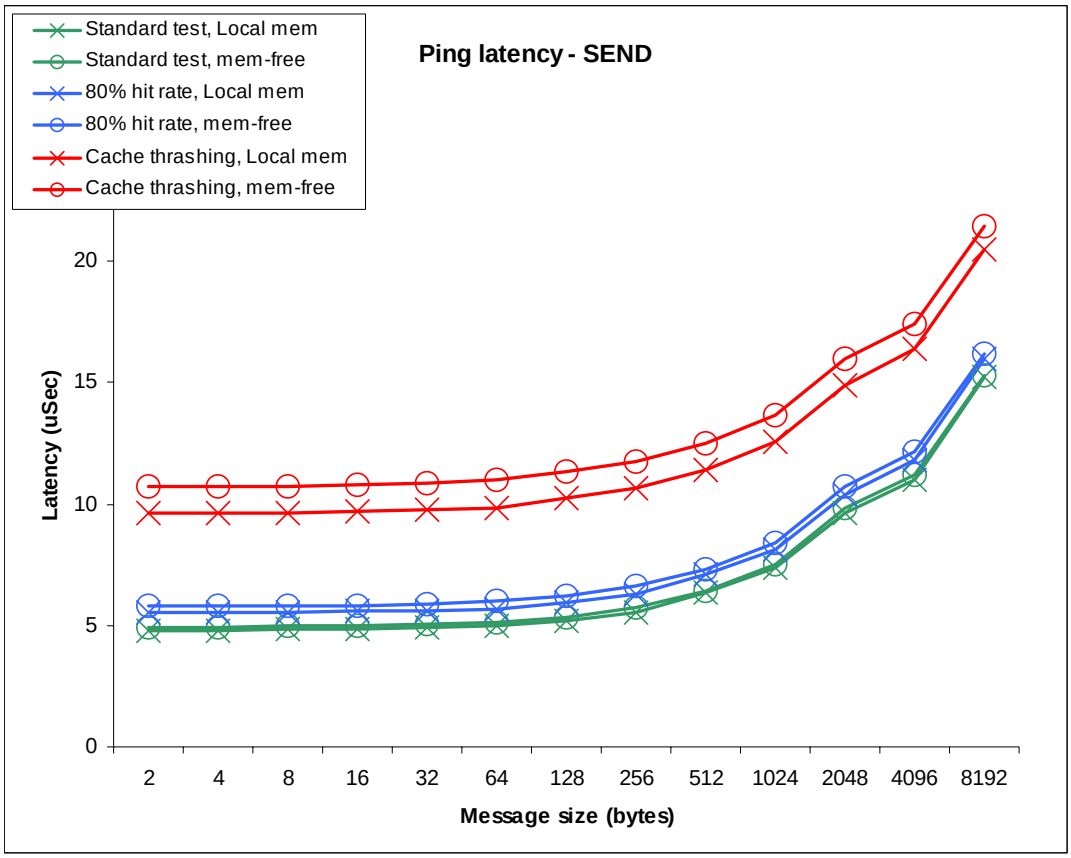

Figure 3. Latency as a Function of Message Size - SEND

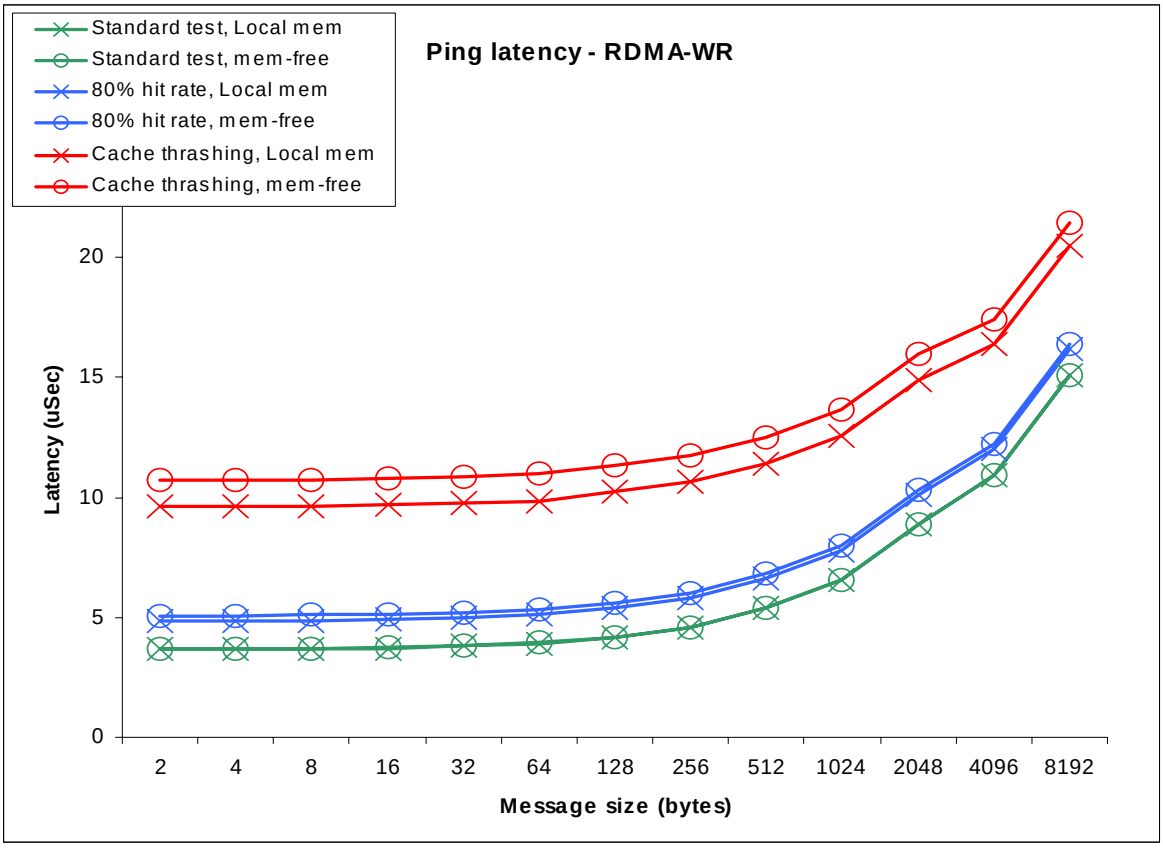

Figure 4. Latency as a Function of Message Size - RDMA-WR

3.0 Basic Tests Description and Setup

All testing is done on the system described in Section 3.4, “System Setup,” on page 6, using the same platform and HCA cards. Mem-free mode is differentiated from operation using local memory by using a different FW version.

3.1 Bandwidth Tests

The standard bandwidth test uses single Reliable Connection QP transferring messages of various sizes. The basic test includes sending a single message. The bandwidth test for each message size uses 1000 iterations of the basic test. Time is measured from the moment when the first send request is posted to the HCA driver until completion of the last request is received. Bandwidth is calculated by dividing the total amount of data sent by the time measured. The connection creation and tear-down is done outside the time measurement periods.

The “cache thrashing” bandwidth test sends messages over multiple Reliable Connection QPs transferring messages of various sizes. The number of connections used exceeds the size of context caches by a factor of four. The basic test includes sending a single message on every connection. The bandwidth test for each message size uses 1000 iterations of the basic test. Time is measured from the moment when the first send request is posted to the HCA driver until completion of all requests is received. A cache miss occurrence for each control access is validated. Bandwidth is calculated by dividing the total amount of data sent by the time measured. The connection creation and tear-down is done outside the time measurement periods.

3.2 Latency Tests

The ping latency test uses single Reliable Connection QP transferring messages of various sizes. The basic test includes sending a single message to a remote peer waiting for the message (ping). Once the message is received by the remote peer, it sends a response message (pong) back to the originator which is waiting (polling) for the response to arrive (pong). Time is measured by the originator from the moment the send request is posted to the HCA driver until a response from the remote peer is received. The reported latency of the basic test is half of the time measured (one-way trip). The latency reported for each message size is the average of 1000 iterations of the basic test. The connection creation and tear-down is done outside the time measurements periods.

The “cache thrashing” latency test sends messages over multiple Reliable Connection QPs transferring messages of various sizes. The number of connections used exceeds the size of context caches by a factor of four. Basic testing includes a ping-pong test on every QP. The latency reported for each message size is the average of 1000 iterations of a basic test. A cache miss occurrence for each control access is validated. The connection creation and tear-down is done outside the time measurements periods.

3.3 Standard Benchmarks

Mem-free mode performance of Pallas benchmarks is shown in Figure 5 below.

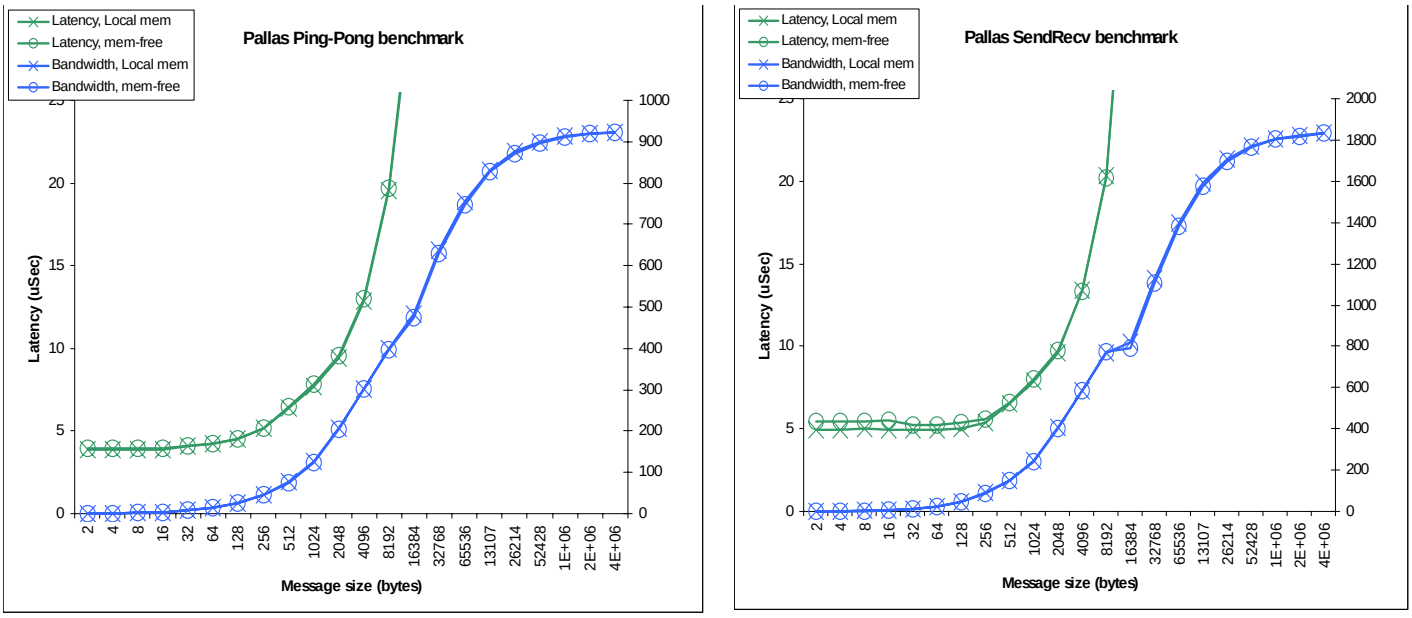

Figure 5. Pallas Benchmarks Results

3.4 System Setup

The setup used for this performance testing is described below.

Platforms: PCI Express IA32 machines with MT25208 InfiniHost III Ex HCA boards - one with attached local memory, and the other in mem-free configuration.

Lab Mach. Names: mtvs40 and mtvs41

Machine: PF400

Chipset: IA32

CPU: 2xGenuine 2.8GHz

Memory: DDR-2G (2064440 KB)

OS: Red Hat Enterprise Linux AS 3 (Taroon Update 2)

Kernel: 2.4.21-15.ELsmp

Firmware: HCA with attached memory: version 25208

HCA in ‘mem-free’ mode: version 25218

VAPI Driver: vapi-linux-4.0-rc6