Intel® Ethernet Switch Family Memory Efficiency

Non-blocking Fabric Architecture

White Paper

May, 2009

Legal

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Intel Corporation may have patents or pending patent applications, trademarks, copyrights, or other intellectual property rights that relate to the presented subject matter. The furnishing of documents and other materials and information does not provide any license, express or implied, by estoppel or otherwise, to any such patents, trademarks, copyrights, or other intellectual property rights.

The Controller may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Table of Contents

Overview

The Holy Grail in switch fabric design is an output queued architecture. This has been difficult to achieve in the past due to the high bandwidth required between the switch inputs and the output queues. Because of this, most vendors implement a combined input-output queued (CIOQ) architecture, which needs less core switch bandwidth, but requires extra features to avoid blocking. Most vendors compromise somewhere between core bandwidth and ingress complexity, but corner case blocking still remains. They also must store the packet at both ingress and egress, adding to latency and memory requirements.

The Intel®Ethernet Switch Family RapidArray memory and Nexus crossbar technology provide, for the first time, the capability to support a fully non-blocking output queued, shared memory architecture with extremely low latency. By providing a high-bandwidth core, the switch architecture can be made simpler than competing devices. This eliminates the complexity of ingress VoQs and the extra memory they require. In addition, multicast packets are only stored once, further reducing on-chip memory requirements. This efficient Intel®Ethernet Switch Family memory architecture means competing devices need larger internal packet memory to compensate, and even then cannot provide low latency solutions. Test results show that the Intel®Ethernet Switch Family architecture can remain non-blocking in the face of disruptive background traffic.

Traditional Switch Architecture

Memory access bandwidth has been a thorn in the side of switch chip architects. When using traditional cross bar and memory designs, there is insufficient on-chip bandwidth to allow every input port to write into the same output queue simultaneously. To get around this blocking issue, also known as Head of Line (HOL) blocking, chip architects include Virtual Output Queues at every switch input. This is also known as a Combined Input/Output Queued (CIOQ) architecture as shown in Figure 1.

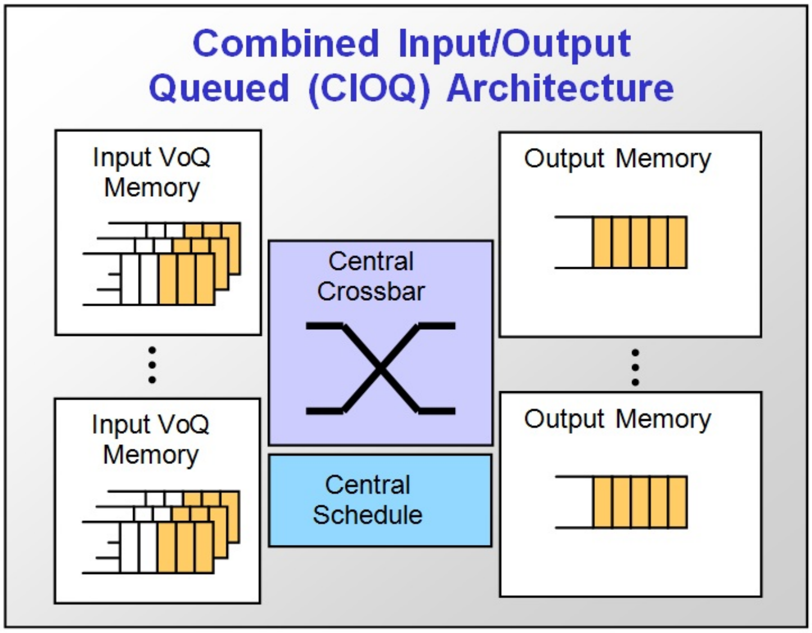

Figure 1. Combined Input/Output Queued (CIOQ) Architecture

Virtual Output Queues provide at each ingress port, a single queue for each switch output (egress) port. If a particular egress queue is temporarily blocked, the matching ingress queue will be flow controlled, but packets destined for other egress ports can bypass this blocked queue and send data to other non-blocked egress ports. But for an N-port switch, this means N*N input queues and associated schedulers which add significant complexity. This also adds to packet latency since each packet must be queued twice through the switch. Because of the complexity of VOQs and associated schedulers, many switch designs trade-off complexity at the expense of some level of internal blocking.

Some switch fabrics are designed using chip-sets, which have separate ingress/egress chips, sometimes called Fabric Interface Chips (FICs), along with central crossbar and scheduler chips. Because of the limited bandwidth provided by off-chip interfaces compared to on-chip interfaces, blocking can occur at the crossbar outputs, requiring a CIOQ architecture. This configuration also requires a complex central scheduler and the associated flow control and grant signaling that must be communicated between devices. All of this adds significantly to cost, power and area.

Intel®Ethernet Switch Family Switch Fabric Architecture

The Intel®Ethernet Switch Family provides a true output queued shared memory architecture. This is enabled by several Intel® patented technologies, which include the Nexus crossbar and the RapidArray memory. By providing full bandwidth access to every output queue from every input port, no blocking occurs within the switch, eliminating the need for complex VOQs. The block diagram of the Intel®Ethernet Switch Family shared memory architecture is shown in Figure 2.

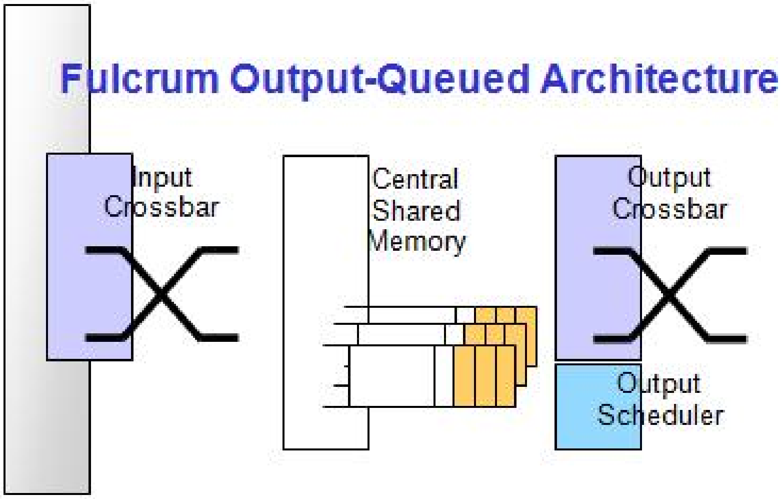

Figure 2. Intel® Output-Queued Architecture

With the Intel®Ethernet Switch Family, all packets arriving at any ingress port are immediately queued at full line rate into shared memory. Packets are then scheduled from shared memory to the egress ports. Multicast packets are de-queued multiple times to each egress fan-out port. Each egress port has an independent scheduler design that is much simpler than a central scheduler. In addition, since the packet is queued only once, cut through latencies of a few hundred nanoseconds can be achieved independent of packet size.

Multicast Capability

Multicast adds complexity to CIOQ designs due to its blocking nature. Figure 3 and Figure 4 below shows a multicast example with a fan-out of 4. With the CIOQ architecture, both egress buffers must accept the packet before it can be de-queued from the ingress buffer, which adds to ingress congestion. Also, the packet must be stored multiple times per switch adding to both the overall memory requirements and to the latency. This also adds to latency jitter due to different physical egress queues.

The Intel®Ethernet Switch Family stores the packet only once per switch and de-queues it multiple times to each egress port. This reduces on-chip congestion, reduces overall memory requirements and provides low latency transmission. Also, port-to-port skew is minimized which is important in applications such as video distribution.

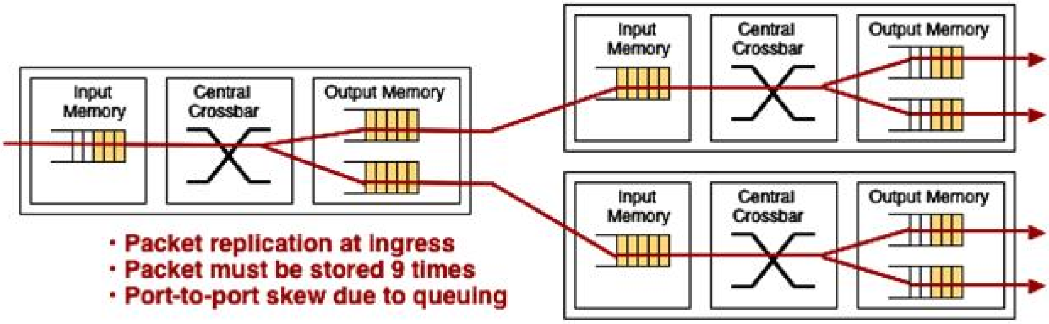

Figure 3. Two-stage CIOQ Multicast Implementation Example

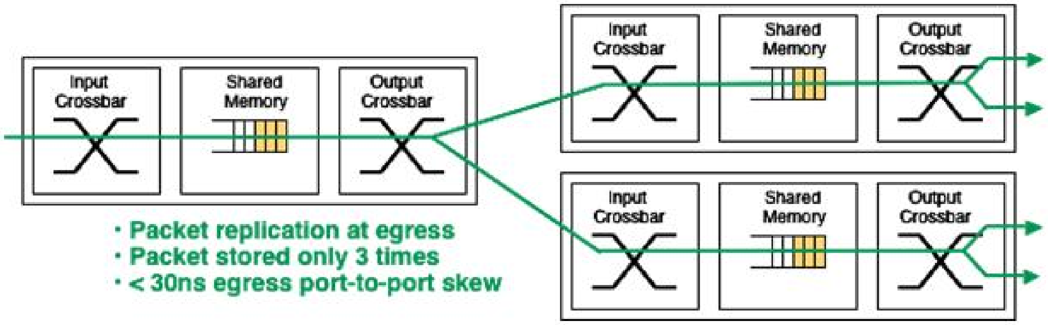

Figure 4. Two-stage Intel®Ethernet Switch Family Multicast Implementation Example

Memory Size Comparison

This section provides a comparison of memory requirements for a CIOQ architecture compared to the Intel®Ethernet Switch Family in a typical implementation. As described above, due to internal blocking and multicast inefficiencies, additional ingress memory will be required. Also, multicast replication requires additional egress memory on a CIOQ switch. There are also inefficiencies due to transferring packets between input queues and output queues, which is not covered in this comparison.

Assume the design goal is to provide enough on-chip packet memory to support 1000 2K packets, which is close to the typical FCoE packet size. The Intel®Ethernet Switch Family architecture allows storing all of these packets in a single shared memory buffer, and therefore requires 2MB of on-chip memory.

Lets assume a CIOQ switch would normally assign 1MB to the ingress queues and 1MB to the egress queues. Also assume due to the design trade-offs discussed above, that there is a 20% blocking probability at the ingress. This means the CIOQ switch would actually need 1.2MB of ingress memory. Lets assume 20% of the traffic is multicast with an average fan-out of 4. This requires 60% more egress memory or 1.6MB. So for this example, the CIOQ switch would need 2.8MB of memory to come close to the performance available with 2MB of Intel®Ethernet Switch Family memory. Keep in mind that the CIOQ architecture can never match the Intel®Ethernet Switch Family latency and low multicast fanout jitter due to the multiple queue hops.

Measured Results

In this section, we evaluate the performance of the Intel®Ethernet Switch Family FM4000 switch, including individual performance tests and comparisons of latency and losslessness, flow control and prioritization in port-to-port, hot spot and multicast scenarios. The FM4000 provides line rate performance in all of these cases and very low latency. Flow control effectively partitions fairness in 23-to-1 hot spot regression tests, and prioritization proves effective for multicast traffic under the stress conditions of full-rate background traffic.

Figure 5 below shows the basic port-to-port latency measured from an experimental setup. The latency per hop is as low as 300 nanoseconds, and in all cases is below 400 nanoseconds. At 100% utilization of the switch, the performance is almost the same as the performance at 10% utilization, where utilization refers to the percentage of available data rate, or line rate.

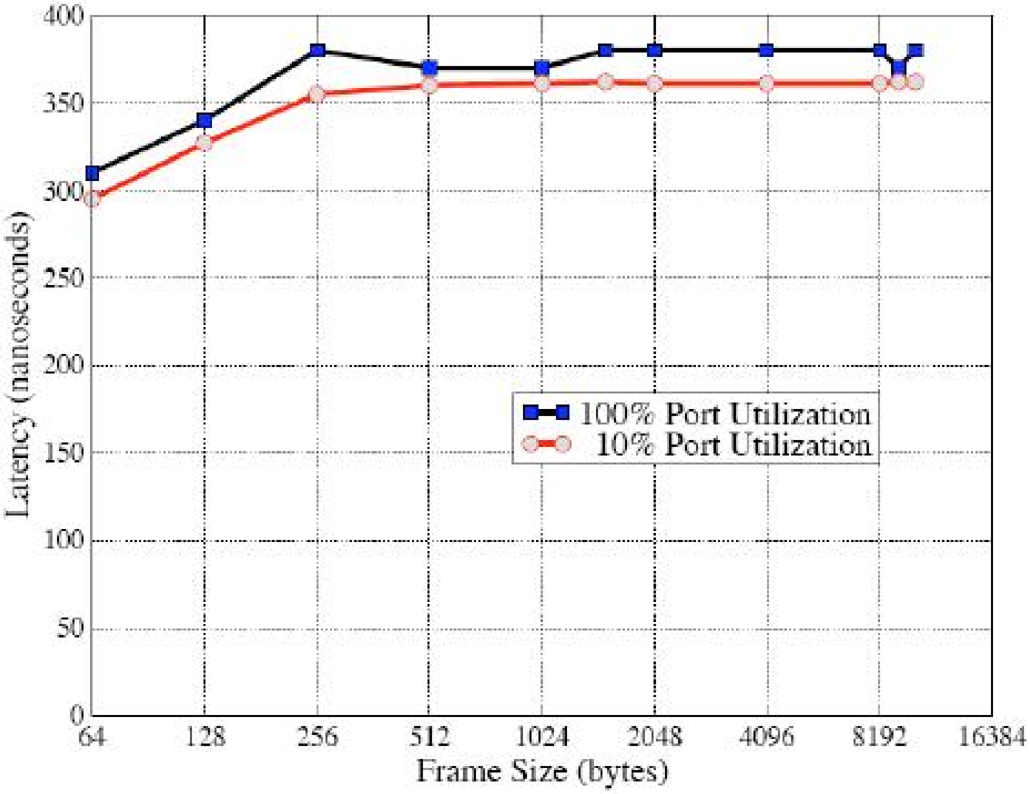

Figure 5. Port-to-Port Latency Measured in an Experimental Setup

The FM4000 is fully provisioned to handle the worst-case congestion scenarios, and in such a scenario is able to operate with lossless operation that is fair between ports. Figure 6 shows fairness in regression tests where twenty-three ports all write to the same egress port. In this scenario the switch uses the Ethernet pause flow control mechanism, which causes the incoming data streams to slow-down to the egress rate of the single port. Fairness measures the difference between how many packets were expected to arrive from each ingress port and how many packets actually arrived.

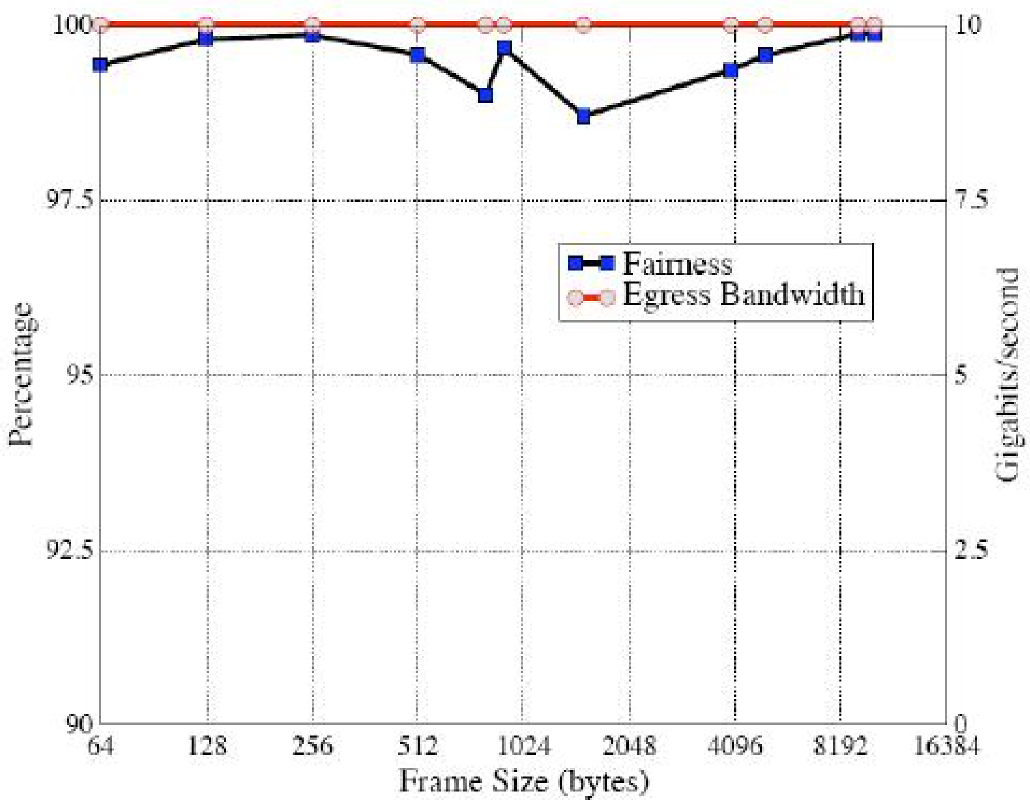

Figure 6. Fairness in Regression Tests for Writing to the Same Egress Port

We also tested the effectiveness of prioritization for multicast messages in the presence of background traffic. Background traffic is generated by creating a persistent loop in the network. This is done by turning the spanning tree off and looping 23 ports back onto themselves. The remaining port is attached to the tester. A single background frame is injected into one of the loop-backed ports after which it is broadcast to all of the ports. This repeats for all of the duplicate packets arriving and the ports quickly reach the full rate for sending and receiving packets. Excess frames are dropped. For the test, 1 Gigabit/sec of multicast traffic and 9 Gigabits/sec of more background traffic are sent on the multicast port.

Figure 7 shows a summary of the multicast results. Prioritized traffic maintains a bounded latency ranging from 1.2 to 4.3 microseconds, depending on the frame size used in background traffic. Note that the synthetic background traffic in these tests represents the extreme worst-case scenario for switch contention. Latency will be much lower if less than 100% of the chip bandwidth is being utilized.

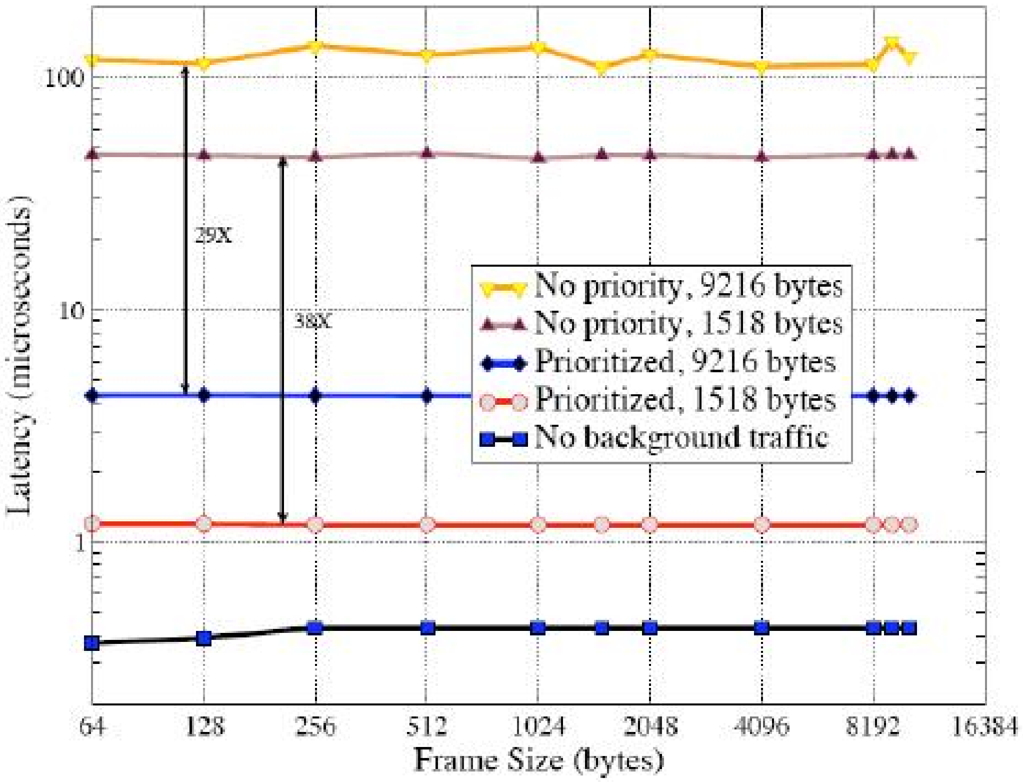

Figure 7. Multicast Results Summary

In the absence of contention from background traffic, multicast is able to achieve even lower latencies on par with the latency of port-to-port traffic, exemplifying how the FM4000's deterministic processing pipeline delivers high performance independent of feature utilization. Traffic prioritization improves latency by 28 to 39 times.

Conclusion

Traditional switch fabric designs utilize a combined input/output queued architecture which requires added complexity. The patented memory and crossbar technology available in the Intel®Ethernet Switch Family , allow the implementation of a true output queued shared memory architecture, which reduces complexity and improves performance. In addition, the Intel®Ethernet Switch Family's non-blocking low latency performance can be achieved with lower on-chip memory requirements than the traditional architectures. Test results show that the Intel®Ethernet Switch Family can provide low latency, fairness and excellent multicast performance in the face of disruptive background traffic.

§ §

NOTE: This page intentionally left blank.