Intel® Data Direct I/O Technology (Intel® DDIO): A Primer

>

Technical Brief

February 2012

Revision 1.0

Legal Statements

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.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document 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.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Intel, Xeon, and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Contents

Revision History

Date	Revision	Description
February 2012	1.0	Initial release.

1 Background

Data center bandwidth requirements continue to rise, driven by growth in server virtualization, unified data and storage networking, and other bandwidth-intensive applications. Many IT organizations are deploying 10 Gigabit Ethernet (10 GbE) to meet these increasing needs. While 10 GbE provides greater, more cost-effective bandwidth, it also places a greater burden on server resources. This is a result of architecture and design decisions made when I/O was slower and processor caches were small. For that environment, it was reasonable for systems to be designed in such a way that the main memory was the primary destination and source of I/O data rather than the scarce resource of cache. This resulted in I/O data transfers requiring many “forced” trips to the memory subsystem for data consumed and delivered by I/O devices. It also kept the cache free of I/O data not currently needed by the CPU. These trips loaded the memory subsystem up to five times the link speeds, forcing the CPU and the I/O subsystem to run slower and consume more power.

The environment has changed; the Intel® Xeon® processor E5 product family supports up to 20 MB last-level cache, so cache resources are no longer scarce. With Intel® Data Direct I/O Technology (Intel DDIO), Intel has updated the architecture of the Intel Xeon processor to remove the inefficiencies of the classic model by enabling direct communication between Intel Ethernet controllers and adapters and host processor cache. Eliminating the frequent visits to main memory present in the classic model reduces power consumption, provides greater I/O bandwidth scalability, and lowers latency. Intel® DDIO is a platform technology that enables I/O data transfers that require far fewer trips to memory (nearly zero in the most optimal scenarios). In doing so, Intel DDIO significantly boosts performance (higher throughput, lower CPU usage, and lower latency), and lowers power.

2 Intel® DDIO: How does it work?

Simply put, Intel DDIO is a platform technology that improves I/O data processing efficiency for data delivery and data consumption from I/O devices. With Intel DDIO, Intel® Ethernet Server Adapters and controllers talk directly to the processor cache without a detour via system memory. Intel DDIO makes the processor cache the primary destination and source of I/O data rather than the main memory. By avoiding multiple reads from and writes to system memory, Intel DDIO reduces latency, increases system I/O bandwidth, and reduces power consumption. Intel DDIO is enabled by default on all Intel Xeon processor E5 based servers and workstation platforms.

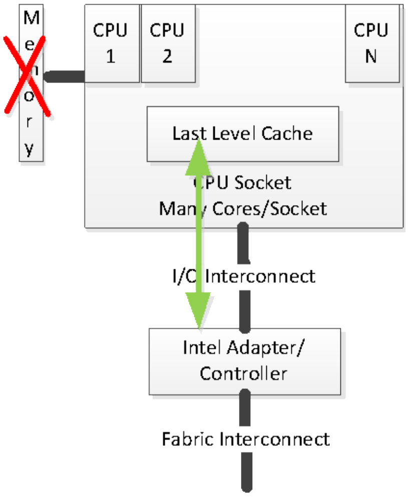

Figure 1. With Intel DDIO, Last-Level Cache Is the Main Target for I/O Data

Intel DDIO functionality is best summarized by studying a data consumption (read) and data delivery (write) operation from an I/O device; these are operations that an I/O device initiates to read and write to host memory. For example, in the case of a network interface card (NIC), an I/O read operation is initiated when that NIC performs a transmit operation, and an I/O write operation is initiated when it receives data from the fabric it is attached to and has to be transferred to the host's memory. In the case of storage, a request generates an I/O read transaction and a response generates an I/O write transaction.

The following sections describe these interactions for a converged network adapter (CNA). In general, these interactions can be generalized for other I/O devices.

2.1 NIC Data Consumption/Read Operations

Data consumption or read operations are initiated by a CNA/NIC when it has a packet to transmit and/or control structures, for example, descriptors to fetch from memory. Intel DDIO technology enables these data consumption operations to be completed with as few trips to memory as possible; in the most optimal case, no trips will be required.

The sequence of operations begins with SW running on the CPU creating a packet and associated data for the NIC to transmit. This involves bringing the data into the CPU's caching hierarchy and updating it with data that the NIC eventually reads. This happens when the NIC is notified by SW, once a packet is ready to be transmitted. The underlying steps for this sequence of operations, is different in a system with and without Intel DDIO technology; the following paragraphs describe these differences.

Figure 2a and **Figure 2b** illustrate the differences in the sequence of operations that occur for I/O initiated data consumption or read operations on systems with and without Intel DDIO.

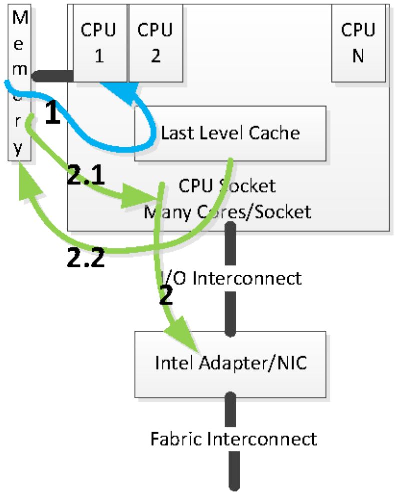

Figure 2a. NIC Reads without Intel® DDIO

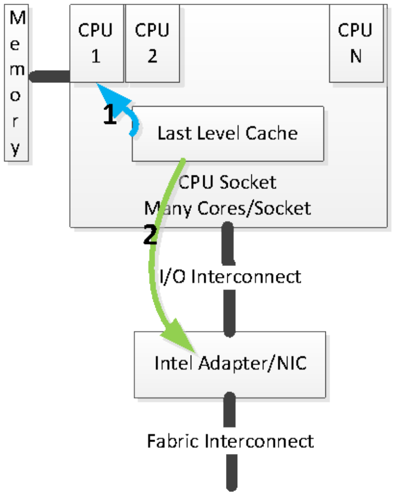

Figure 2b. NIC Reads with Intel® DDIO

(bold numbers refer to steps below)

Figure 2a, shows a data consumption or read operation without Intel DDIO technology:

> **Step 1:** Software running on the CPU creates a packet, which involves filling an allocated buffer (memory addresses) with new information, as well as control structures with commands for a NIC. These data access operations result in “cache misses” (see Step 2.2 below for explanation) and cause data to be read from memory and into the CPU’s caching hierarchy. Once the data is created, the NIC is notified to begin a transmit operation.
>
> **Step 2:** When the NIC receives notification for starting a transmit operation, it first reads control structures and subsequently the packet data. Since the data was very recently created by software running on the CPU, there is a high probability of the data being resident in its caching hierarchy.
>
> > **Step 2.1:** Each read operation triggered by the NIC on its PCIe interconnect (for example) causes data to be forwarded from the cache, if present; however, it also causes the data to be evicted from cache, that is, the act of reading by an I/O device causes data to be evicted.
> >
> > **Step 2.2:** In earlier generations, this read operation also causes a speculative read to be issued to memory in parallel, while the coherency protocol checks if the data happens to be in the CPU’s caching hierarchy.

So, at a minimum, a single read operation caused two or three trips to memory, depending on the platform.

Figure 2b illustrates the same set of operations on a platform with Intel DDIO technology.

Step 1: Data access operations associated with creating a packet are satisfied from within the cache. Thus, software running on the CPU does not encounter cache misses, and, therefore does not have to fetch data from memory.

Step 2: The read operation initiated by the NIC is satisfied by data from the cache, without causing evictions, that is, the data remains in the cache; since this data is re-used by software, it stays in the cache.

Thus, I/O device data consumption operations with Intel DDIO technology are achieved with fewer trips to memory, and, in the ideal case, no trips to memory. Also, from a CPU caching perspective, the data in the cache is not disturbed by an I/O data consumption operation. It is important to note that an I/O data consumption operation that does not find the data in cache, sources it from memory; this operation does not cause data to be allocated into the caching hierarchy (unlike the case of data delivery operations described below).

2.2 NIC Data Delivery/Write Operations

Data delivery or write operations are initiated by I/O devices when data is transferred to the host from these devices. In the case of a NIC, this occurs when it receives a packet from the network and/or control structures that it needs to pass up to the host software for processing.

Data delivery begins with a NIC receiving a packet from the wire; the received packet and any accompanying control data is transferred to the host for protocol processing. If the protocol processing is successful, the payload from the packet is delivered to a consuming application. This involves bringing the data, as well as any accompanying control structures, into the CPU’s caching hierarchy that the SW running on the host eventually reads. The underlying steps for this sequence of operations, is different in a system with and without Intel DDIO technology; the following paragraphs describe these differences.

Figure 3a and **Figure 3b** illustrate the differences in the sequence of operations that occur for I/O initiated data delivery or write operations on systems with and without Intel DDIO.

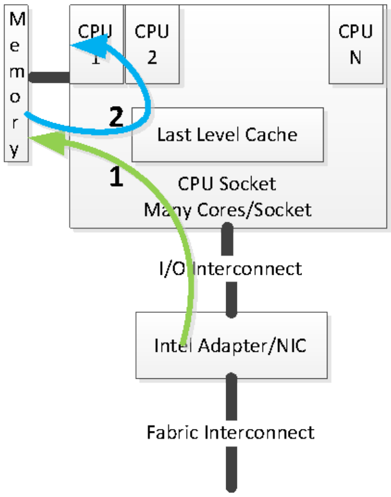

Figure 3a NIC Writes without Intel® DDIO

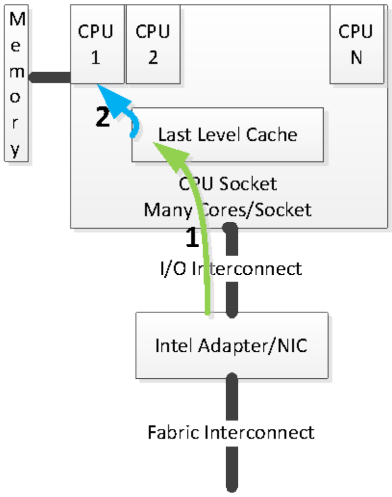

Figure 3b NIC Writes with Intel® DDIO

(bold numbers refer to steps below)

Figure 3a shows a data delivery or write operation without Intel DDIO technology:

> **Step 1:** Data delivery operations have the NIC transferring data (packets or control) to host memory. If the data being delivered happens to be in the CPU's caching hierarchy, it is invalidated.
>
> **Step 2:** SW running on the CPU reads the data to perform processing tasks. These data access operations misses cache (See step 1 above for explanation) and causes data to be read in from memory, into the CPU's caching hierarchy.

Figure 3b illustrates the same set of operations on a platform with Intel DDIO technology.

> **Step 1:** I/O data delivery uses Write Update or Write Allocate operations (depending on cache residency of the memory addresses to which data is being delivered), which causes data to be delivered to the cache, without going to memory.
>
> **Step 2:** The subsequent read operations initiated by SW are satisfied by data from the cache, without causing cache misses.

Intel DDIO technology enables these data delivery operations to be accomplished with as few trips (including none) to memory as possible with two modes of operation:

1. ***Write Update***, which causes an in-place update in the cache. If the memory addresses for the data being delivered already exists in the CPU's caching hierarchy, then no trips to memory are needed.
2. ***Write Allocate***, which causes allocation in the last-level cache of the CPU's caching hierarchy. If the memory addresses for the data being delivered does not exist in the CPU's caching hierarchy, then no trips to memory are needed.

Write Allocate operations are restricted to 10% of the last-level cache; this is a trade-off to prevent cache pollution from data streams that are not consumed by threads running on CPUs or for scenarios where the incoming rate is much faster than the thread is capable of handling. This is NOT a dedicated/reserved cache for I/O (or Intel DDIO). It is fully available to applications running on the CPU. This is fixed and cannot be changed.

Thus, I/O device data delivery operations with Intel DDIO technology are achieved with fewer trips to memory, and in the ideal case, without any trips to memory. Also, from a CPU caching perspective, the data in cache is not disturbed by virtue of an I/O data delivery operation, which creates opportunities for intelligent HW/SW design optimizations.

2.3 Intel® DDIO Requires No Industry Enabling

Inte DDIO is enabled by default on all Intel® Xeon® processor E5 servers and workstations.

Intel DDIO has no hardware dependencies and is invisible to software. No driver changes are required. No OS or VMM changes are required. No application changes are required. All industry I/O devices from IHVs benefit from Intel DDIO including InfiniBand Architecture®, Fibre Channel, RAID, and Ethernet. However, Intel Ethernet products with their high performing, stateless architecture excel with Intel DDIO.

2.4 Non-Uniform IO Access (NUIOA) Operations

The Intel® Xeon® processor E5 product family introduces a distinction between whether an

I/O adapter is attached directly to the socket where the I/O is consumed/generated (“local”) or the I/O has to traverse a QPI link to reach the thread where it is used (“remote”). Currently, Intel DDIO affects only local sockets, so its performance improvement is due to the relative difference in performance between the local socket I/O and remote socket I/O.

While Intel DDIO improves overall I/O performance for a server, not all threads will gain the improvement. To ensure that Intel DDIO’s benefits are available to applications where they are most useful, the application can be pinned to particular sockets using Intel DDIO. This arrangement is called socket affinity.

For information about forcing socket affinity, please see the application note, *Process & Interrupt Affinity on Intel® Xeon® Processor E5 Servers with Intel® DDIO Technology.*

3 Intel® DDIO: The Benefits

Applications that push the I/O bandwidth, as is common in telecomm, can see 2x or more increase with Intel® Xeon® processor E5-based servers over the previous Intel® Xeon® processor 5600-based architecture, because memory bandwidth is now no longer a constraint. More common data center applications do not stress I/O; the performance benefit will be relatively minor in general, but they will see a power consumption savings of up to seven Watts per two-port NIC. Applications that are sensitive to latency, like UDP-based financial trading, will see a reduction in latency on the order of ~10-15% due to Intel DDIO. In the lab, the I/O data rates have been pushed to find the absolute limits of the Intel Xeon E5 platform and achieved I/O data rates of ~250 Gbps, three times the maximum ever seen with the previous generation Intel Xeon processor 5600 servers. The previous generation reaches its limit when the maximum memory bandwidth is reached. But much of that memory activity is eliminated with Intel DDIO, removing the bottleneck and unleashing the full capabilities of the Xeon processors, resulting in dramatic improvement in the capabilities of the Intel Xeon processor E5-based platforms in managing I/O. With Intel DDIO, your Intel Xeon processor E5 server or workstation has the headroom to handle even the most extreme I/O loads.

Intel Ethernet products, with their Intelligent Offload architecture, take advantage of host-based processing whenever it makes sense from a system-level perspective balancing performance, power consumption, flexibility, and cost. Intel Ethernet products were designed to take advantage of the improvements in communication between host and network controller that Intel DDIO provides. The industry-leading small packet performance of Intel's Ethernet products gets even better with Intel DDIO. In contrast, competitive Ethernet products that run on-board embedded processors from firmware do not take nearly as much advantage of Intel DDIO or the ability to scale performance with Moore's Law advancements.

###