谈谈DDIO你该知道的事

前言

当今时代，随着大数据和云计算的爆炸式增长，宽带的普及以及个 人终端网络数据的日益提高，对电信服务节点和数据中心的数据交换能力和网络带宽提出了更高的要求。并且，数据中心本身对虚拟化功能的需求也增加了更多的网络带宽需求。电信服务节点和数据中心为了应付这种需求，需要对内部的各种服务器资源进行升级。在这种环境下，英 特尔公司提出了Intel® DDIO(Data Direct I/O)的技术。该技术的主要目的就是让服务器能更快处理网络接口的数据，提高系统整体的吞吐率，降低延迟，同时减少能源的消耗。

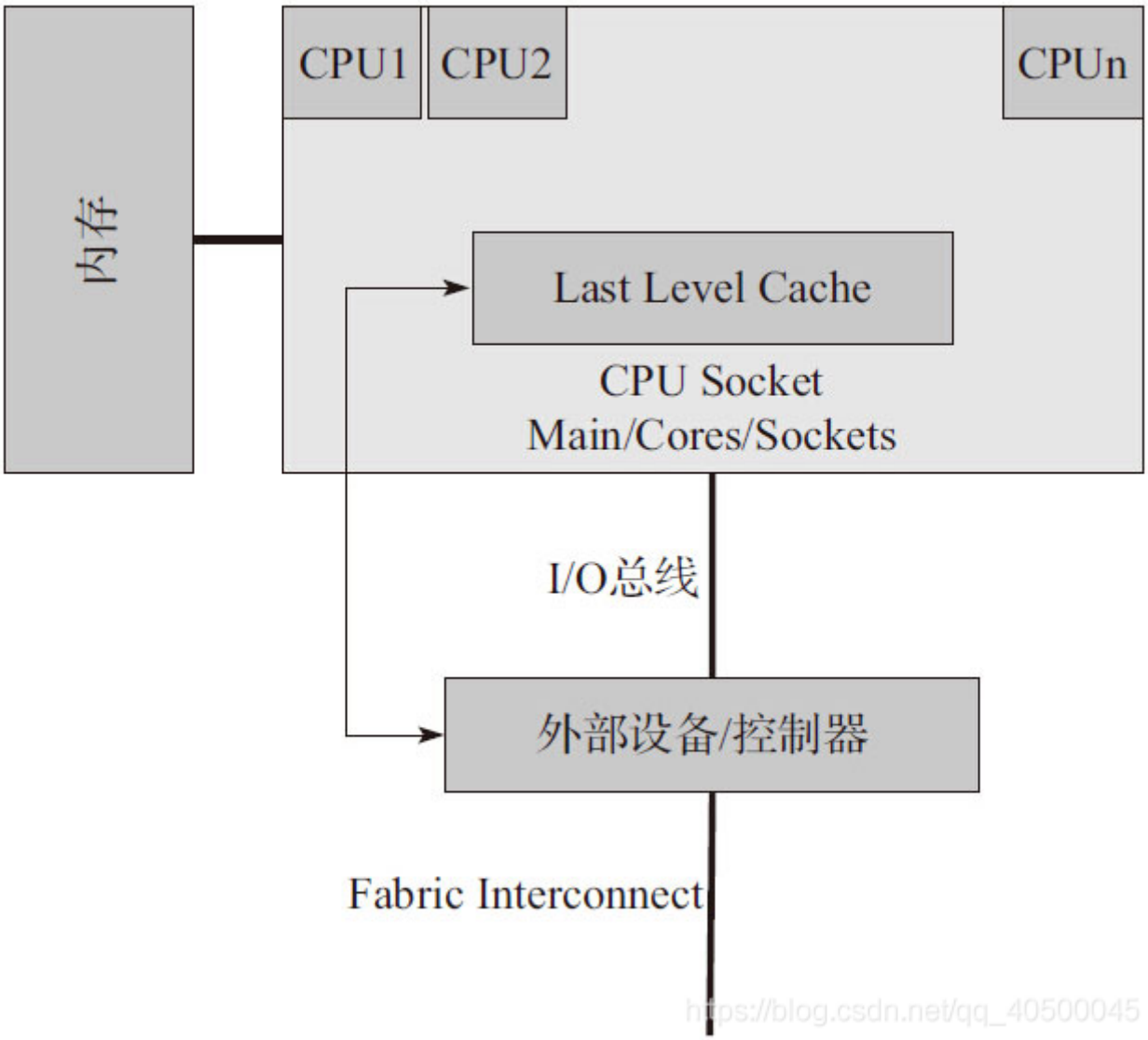
一、服务器是如何处理从网络上来的数据？

1. 当一个网络报文送到服务器的网卡时。
2. 网卡通过外部总线(比如 PCI总线)把数据和报文描述符送到内存。
3. CPU从内存读取数据 到Cache进而到寄存器。
4. 进行处理之后，再写回到Cache，并最终送到内存中。
5. 最后，网卡读取内存数据，经过外部总线送到网卡内部，最终通过网络接口发送出去。

可以看出，对于一个数据报文，CPU和网卡需要多次访问内存。而内存相对CPU的使用寄存器来讲是一个非常慢速的部件。CPU需要等待数百个周期才能拿到数据，在这过程中，CPU什么也做不了。

二、DDIO技术是如何改进的呢?

这种技术使**外部网卡和CPU通过LLC Cache直接交换数据**，绕过了内存这个相对慢速的部件。这样，就增加了CPU处理网络报文的速度(减少了CPU和网卡等待内存的时间)，减小了网络报文在服务器端的处理延迟。这样做也带来了一个问题，因为网络报文直接存储在LLC Cache中，这大大增加了对其容量的需求，因而在英特尔的E5处理器系列产品中，把LLC Cache的容量提高到了 20MB。

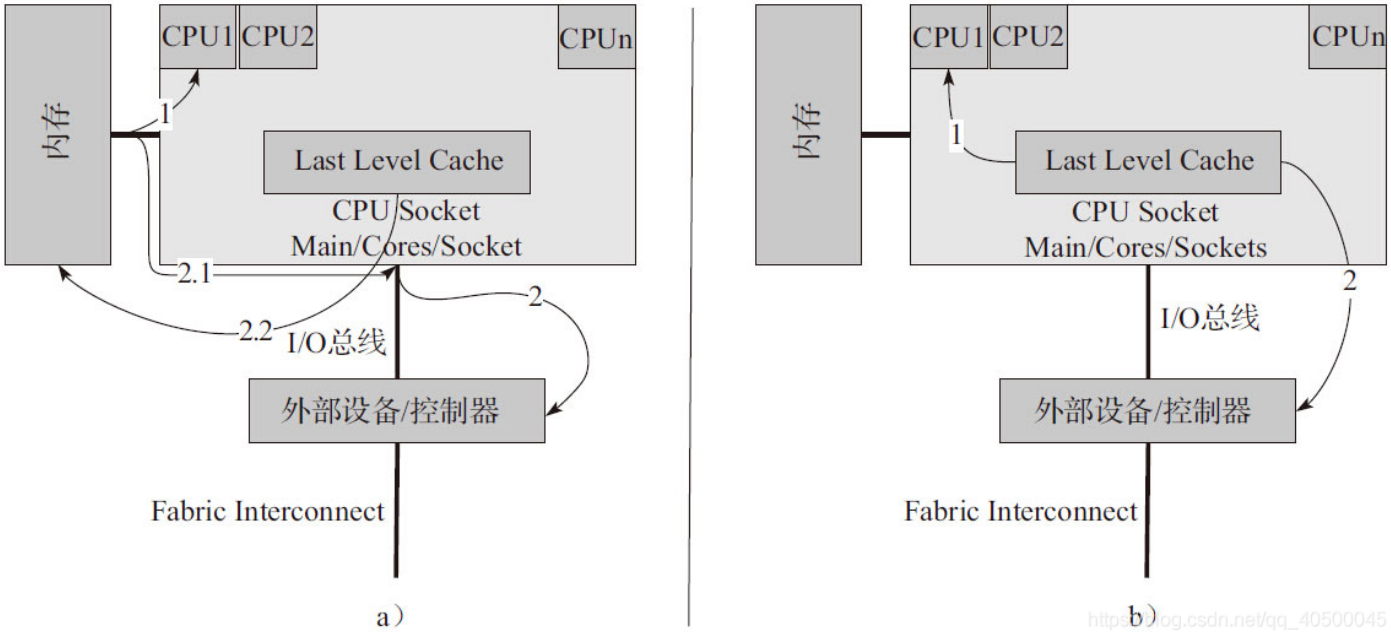


DDIO功能模块会学习来自I/O设备的读写请求，也就是I/O对内存的读或者写的请求。例如，当网卡需要从服务器端传送一个数据报文到网络上时，它会发起一个I/O读请求(读数据操作)，请求把内存中的某个数据块通过外部总线送到网卡上;当网卡从网络中收到一个数据报文 时，它会发起一个I/O写请求(写数据操作)，请求把某个数据块通过外部总线送到内存中某个地址上。

接下来的章节会详细介绍在没有DDIO技术和有DDIO技术条件下， 服务器是如何处理这些I/O读写请求的。

1.网卡的读数据操作

通常来说，为了发送一个数据报文到网络上去，首先是运行在CPU 上的软件分配了一段内存，然后把这段内存读取到CPU内部，更新数据，并且填充相应的报文描述符(网卡会通过读取描述符了解报文的相应信息)，然后写回到内存中，通知网卡，最终网卡把数据读回到内 部，并且发送到网络上去。但是，没有DDIO技术和有DDIO技术条件的处理方式是不同的。



a图没有DDIO技术的处理流程图

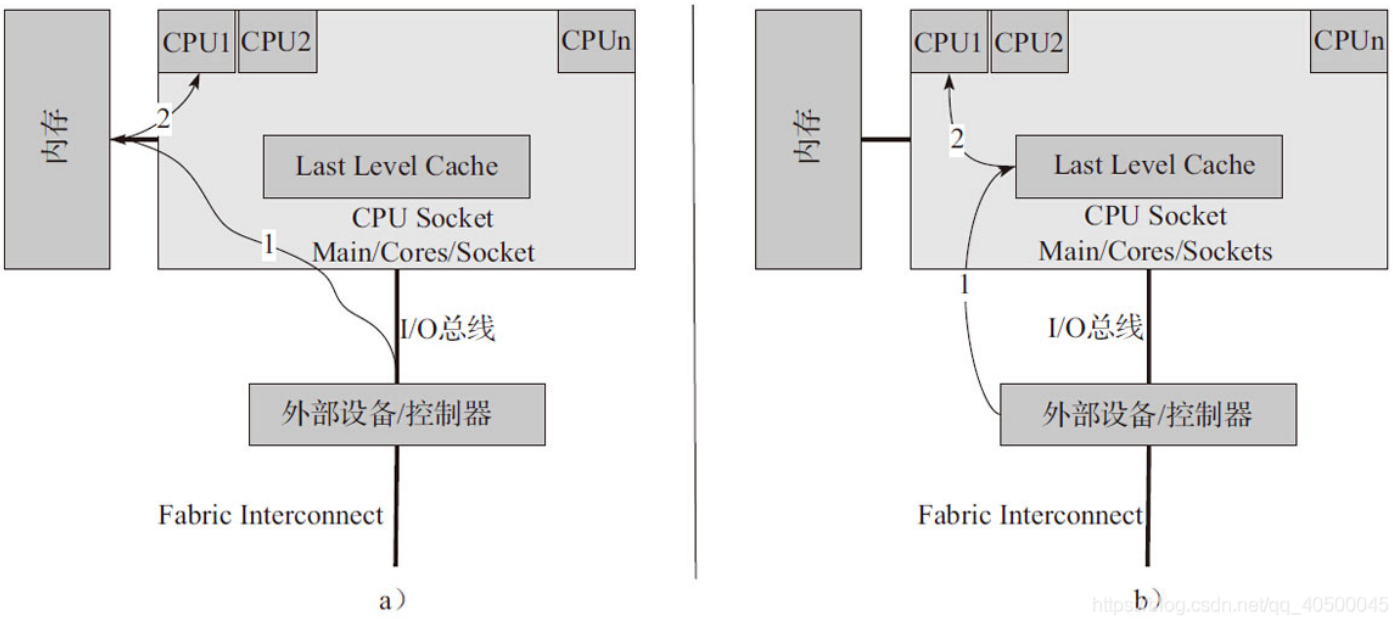
1. 处理器更新报文和控制结构体。由于分配的缓冲区在内存中， 因此会触发一次Cache不命中，处理器把内存读取到Cache中，然后更新控制结构体和报文信息。之后通知NIC来读取报文。
2. NIC收到有报文需要传递到网络上的通知后，它首先需要读取控制结构体进而知道从哪里获取报文。由于之前处理器刚把该缓冲区从内存读到Cache中并且做了更新，很有可能Cache还没有来得及把更新的内容写回到内存中。因此，当NIC发起一个对内存的读请求时，很有可能这个请求会发送到Cache系统中，Cache系统会把数据写回到内存中，然后内存控制器再把数据写到PCI总线上去。因此，一个读内存的操作会产生多次内存的读写。

b图是有DDIO技术的处理流程

1. 处理器更新报文和控制结构体。这个步骤和没有DDIO的技术类似，但是由于DDIO的引入，处理器会开始就把内存中的缓冲区和控制结构体预取到Cache，因此减少了内存读的时间。
2. NIC收到有报文需要传递到网络上的通知后，通过PCI总线把控制结构体和报文送到NIC内部。利用DDIO技术，I/O访问可以直接将Cache的内容送到PCI总线上。这样，就减少了Cache写回时等待的时间。
   由此可以看出，由于DDIO技术的引入，网卡的读操作减少了访问内存的次数，因而提高了访问效率，减少了报文转发的延迟。在理想状况下，NIC和处理器无需访问内存，直接通过访问Cache就可以完成更新数据，把数据送到NIC内部，进而送到网络上的所有操作。

2.网卡的写数据操作

网卡的写数据操作和上节讲到的网卡的读数据操作是完全相反的操作，通俗意义上来讲就是有网络报文需要送到系统内部进行处理，运行的软件可以对收到的报文进行协议分析，如果有问题可以丢弃，也可以转发出去。其过程一般是NIC从网络上收到报文后，通过PCI总线把报文和相应的控制结构体送到预先分配的内存，然后通知相应的驱动程序 或者软件来处理。和之前讲到的网卡的读数据操作类似，有DDIO技术和没有DDIO技术的处理也是不一样的，以下是具体处理过程。



a图没有DDIO技术的处理流程图

1. 报文和控制结构体通过PCI总线送到指定的内存中。如果该内存恰好缓存在Cache中(有可能之前处理器有对该内存进行过读写操作)，则需要等待Cache把内容先写回到内存中，然后才能把报文和控制结构体写到内存中。
2. 运行在处理器上的驱动程序或者软件得到通知收到新报文，去内存中读取控制结构体和相应的报文，Cache不命中。之所以Cache一定不会命中，是因为即使该内存地址在Cache中，在步骤1中也被强制写回到内存中。因此，只能从内存中读取控制结构体和报文。

b图是有DDIO技术的处理流程

这时，报文和控制结构体通过PCI总线直接送到Cache中。这时有两种情形:

1. a)如果该内存恰好缓存在Cache中(有可能之前处理器有对该内存进行过读写操作)，则直接在Cache中更新内容，覆盖原有内容。
   b)如果该内存没有缓存在Cache中，则在最后一级Cache中分配一块区域，并相应更新Cache表，表明该内容是对应于内存中的某个地址的。
2. 运行在处理器上的驱动或者软件被通知到有报文到达，其产生一个内存读操作，由于该内容已经在Cache中，因此直接从Cache中读。

由此可以看出，DDIO技术在处理器和外设之间交换数据时，减少了处理器和外设访问内存的次数，也减少了Cache写回的等待，提高了系统的吞吐率和数据的交换延迟。