基于 Myrinet 的高性能 VIA 设计与实现*

陈 渝[1+], 焦振强[2], 谢 军[2], 都志辉[1], 李三立[1]

[1](清华大学 计算机科学与技术系 高性能计算研究所,北京 100084)

[2](中国科学院 研究生院 电子学部,北京 100039)

Design and Implementation of a High Performance VIA Based on Myrinet

CHEN Yu[1+], JIAO Zhen-Qiang[2], XIE Jun[2], DU Zhi-Hui[1], LI San-Li[1]

[1](Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

[2](Institute of Electricity, Graduate School, The Chinese Academy of Sciences, Beijing 100039, China)

+ Corresponding author: Phn: 86-10-62782530, Fax: 86-10-62771138, E-mail: chenyu@hpclab.cs.tsinghua.edu.cn

http://hpclab.cs.tsinghua.edu.cn/~chenyu/myvia.html

Received 2002-04-01; Accepted 2002-08-14

Chen Y, Jiao ZQ, Xie J, Du ZH, Li SL. Design and implementation of a high performance VIA based on Myrinet. *Journal of Software*, 2003,14(2):285~292.

Abstract: Virtual interface architecture (VIA) established a communication model with low latency and high bandwidth, and defined the standard of user-level high-performance communication specification in cluster system. In this paper, the current development, the principle and implementations of VIA are analyzed, and a user-level high-performance communication software MyVIA based on Myrinet is presented, which is comfortable with VIA specification. First, the design principle and the framework of MyVIA are described, and then the optimized technologies for MyVIA are proposed, which include UTLB, continued physical memory and varied NIC buffer, the pipelining process based on resource and DMA chain, physical descriptor ring and dynamic cache. The experimental results indicate that the bandwidth of MyVIA for 4KB message is 250MB/s, the lowest one-way latency is 8.46μs, which show that the performance of MyVIA surpasses that of other VIA.

Key words: virtual interface architecture; user-level communication protocol; parallel process; cluster computing

摘 要: Virtual interface architecture(VIA)建立了一种低延迟、高带宽的通信模型,定义了集群系统中用户层高性能通信规范的标准.通过分析 VIA 的进展、原理及当前的实现,在 Myrinet 上设计并实现了一个符合 VIA 规范的高性能用户层通信软件 MyVIA.首先定义了 MyVIA 的设计原理和框架;然后针对 MyVIA 实现的不同层次,通过与 BerkeleyVIA 的比较,提出了 UTLB、连续物理内存和可变长 NIC 内存管理、基于资源和 DMA chain 的流水线处理、物理描述子环和物理描述子动态缓存等多项优化技术.通过性能的分析比较表明,MyVIA 发送 4KB 数据包时的带宽可达到 250MB/s,最小单边延迟为 8.46μs.与目前其他 VIA 实现相比,MyVIA 的性能有了较

* Supported by the National Natural Science Foundation of China under Grant No.60203024 (国家自然科学基金); the Postdoctoral Science Foundation of China (博士后科学基金); the 985 Foundation of Tsinghua University of China under Grant No.985 information-38-importance-03 (清华大学 985 项目基金)

第一作者简介: 陈渝(1972－),男,广东湛江人,博士,主要研究领域为高性能并行通信,并行编译.

为显著的提高.

关键词: 虚拟接口结构;用户层通信协议;并行处理;集群计算

中图法分类号: TP393 **文献标识码**: A

近年来,高性能并行集群系统机群间的进程通信对网络性能的要求越来越高,虽然网络硬件的性能大幅度提高,但过多的通信软件开销无法发挥通信硬件的最大性能.为了实现机群系统内消息的快速传递,INTEL,COMPAQ,Microsoft 公司及其他研究机构深入分析了用户层通信的原理,并归纳各种已有的用户层通信协议[1]的优点,联合制定了虚拟接口结构 (virtual interface architecture,简称 VIA[2])规范.作为用户层通信的工业标准,VIA 定义了规范的数据操作集进行消息传递,其目的是在机群中的任意两个节点之间实现高带宽低延迟的通信和数据交换,并尽量减少主机 CPU 的占用率.VIA 是下一代 I/O 互联结构 InfiniBand 中的核心技术之一.

目前 VIA 有代表性的实现主要有两种:由美国 Lawrence Berkeley 国家实验室(LBNL)所支持的 Modular VIA(M-VIA)是 VIA 标准在 Linux 操作系统下的一个软件 VIA 实现版本[3],它在不同的网卡上部分实现了 VIA;Berkeley VIA Project[4]是美国加州大学 Berkeley 分校计算机系进行的基于 Myrinet[5]的固件 VIA 研究项目,已经发布了基于 Solaris,Linux 和 Windows NT 下的代码发行版本.其他 VIA 的实现还包括 GigaNet 等.通过对 VIA 规范和各种 VIA 实现的深入分析,我们设计并实现了基于 Myrinet LANai4 和 LANai9[6]的高性能 VIA——Myrinet VIA(简称 MyVIA).它是清华大学 THVIA 工程中的一个研究项目.通过性能分析和比较表明,与 M-VIA 和 Berkeley VIA 相比,MyVIA 的带宽最高、延迟最小,并且能够很好地支持并行编程标准——MPI.

1 VIA 的原理分析

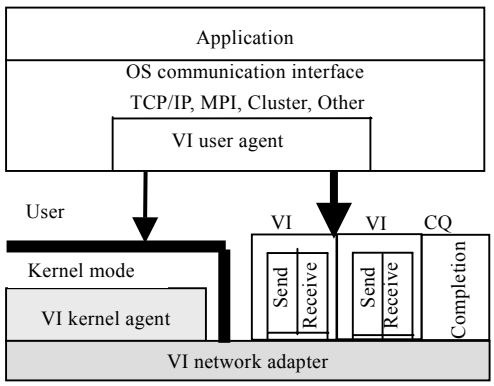

Fig.1 Architecture of VIA
图 1 VIA 体系结构

VIA 是用户层通信协议进一步发展的产物.VIA 的逻辑结构包含 4 个部分:VI 生产者(VI provider)、VI 消费者(VI consumer)、虚拟接口(virtual interface)、完成队列 CQ(complete queues),如图 1 所示.其中 VI 是整个通信结构的核心组成部分,所有的数据传输操作都是通过 VI 进行的.VI 生产者包括支持 VI 的网卡和系统核心代理(OS kernel agent)两部分,主要为 VI 消费者提供各种底层服务.系统核心代理运行在核心态下,并以网卡驱动程序的形式体现,它的任务主要包括网卡打开/关闭、VI 建立/拆除、VI 连接/断开等操作.VI 消费者包括应用程序和用户层代理(user agent)两部分.用户层代理主要向用户提供编程接口,隐藏 VI 系统核心代理和底层硬件的实现细节,方便上层用户(MPI,TCP/IP 等)使用.VI 处于 VI 生产者与消费者之间,提供了二者之间数据传输的接口,每个 VI 由两个工作队列组成:发送请求队列和接收请求队列.VI 消费者以描述子的形式将数据传输请求放入工作队列,并通过工作队列的门铃机制通知 VI 生产者完成数据传输,VI 生产者完成一次传输操作之后,将相应描述子的标识位置,同时在完成队列中添加标识记录,以通知 VI 消费者完成相应的操作.通常一个完成队列可以对应多个 VI 工作队列,使得 VI 消费者可高效地对 VI 生产者的完成情况进行查询.有关 VIA 的详细规范说明见文献[2].

VIA 规范的核心思想是,在通信过程中的旁路操作系统,由用户程序直接对网络接口进行访问,并尽量减少不必要的软件通信开销,减少对主机 CPU 资源的占用,缩短通信操作的关键路径.而且,VIA 规范通过保证 VIA 体系结构一定的抽象性还可通过软件、固件或硬件对其进行具体的实现.

2 MyVIA 设计与实现

2.1 通信硬件简介

Myrinet 是美国 Myricom 公司的高性能网络产品,主要针对大规模并行计算方面的应用.其主要技术特点包括:可变长数据包传输、wormhole 路由、可编程的 RISC CPU、同步发送异步接收、支持切通(cut-through)路由、支持通信链路硬件流控等.由于可对 Myrinet 网卡编程,因此国内外许多用户层通信协议的研究都选择 Myrinet 作为硬件平台,研究各种新型通信协议.MyVIA 目前支持 Myrinet 的 LANai4 和 LANai9,其中 LANai4 不支持 DMA Chain 操作和硬件门铃机制;LANai9 支持 DMA Chain 操作和硬件门铃机制.

2.2 MyVIA的设计思想

计算节点间的进程通信性能受到许多因素的影响,但在具有高性能 CPU 和通信硬件的情况下,其性能主要取决于主机或通信硬件上消息发送或接收的软件开销等.而软件的开销主要来源于通信协议的复杂性、触发的中断数、进程的上下文切换、路由、组通信以及数据拷贝等各种因素.评价通信性能必须与通信流量模型相结合,如果通信流量的组成主要是大数据量消息,那么网络硬件的带宽将起到重要的作用.但当在网络上传递小消息时,每个字节的平均软件开销就会相对地比较高,网络硬件的高带宽一般无法充分利用.

通过 Gusella 等人的研究[7]以及我们对大量并行程序的测试表明,典型的局域网中并行计算的通信流量模型呈现双峰状态,即多于 80%的消息大小在 128 字节~4 096 字节之间,大约 8%的消息大于 8 192 字节.针对上述分析,由于 MyVIA 是基于 Myrinet 且符合 VIA 规范的高性能用户层通信软件,因此在设计 MyVIA 时,我们并没有盲目追求带宽的最大化,而是在充分利用网络硬件带宽的同时,支持 128 字节~4 096 字节之间的高效数据传输.我们的设计原则是:在遵循 VIA 规范的基础上,平衡 VIA 各层次实现的功能,缩短通信关键路径,实现 VIA 通信性能的最大化和良好的可扩展性.基于上述分析和设计原则,我们所设计的 MyVIA 系统具有下述特点:

(1) 高性能:针对 MyVIA 的系统框架,我们通过分析其他 VIA 实现的不足,在多个层次上提出了多种优化技术,可以实现高性能的 VIA 数据传输.这些技术包括:

- 基于主机的 User TLB:减少内存地址虚实转换开销,减少网卡固件的复杂度;
- 连续物理内存和可变长内存管理:减少 DMA 传输开销,支持变长数据传输;
- 基于资源和 DMA chain 的流水线处理:提高网卡各功能部件的利用率和并行度,优化短信息传输;
- 物理描述子环和动态缓存:缩短通信关键路径和数据传输开销,提高 VIA 的节点扩展性.

(2) 透明性:虽然 Myrinet 的硬件总体结构没有改变,但由于硬件实现上的功能改进(DMA chain 功能等),LANai9 比 LANai4 在各方面都有很大的提高.为了充分发挥各自硬件的性能,并减少对软件的大量改动,我们对二者分别设计了网卡固件代理,并在网卡固件代理上建立了一层抽象接口,使得核心代理层和面向用户的 VIA 库层对不同网卡的固件代理基本上透明.

(3) 可扩展:为了在将来进一步支持其他的硬件和技术,我们定义了一个可扩展、模块化的系统框架.它主要由面向用户的 VIA 库层、核心代理层和面向网卡的代理层 3 部分组成,其中在 VIA 库层和核心代理层进行了划分,在各个层次之间提供一个完善的接口.这样,针对不同的实现,只要完成对应的接口,就可以支持不同的硬件.

2.3 MyVIA的组成框架

MyVIA 主要由 3 部分组成,各部分功能划分如图 2 所示.一是面向用户的 VIA 库层,简称 User Agent(UA),用来向用户提供标准的 VIA 函数接口,包括数据放送、数据接收、通信轮询等操作;其次是面向网卡的代理层,简称 NIC Agent(NA),主要负责 VIA 在网卡上的各种资源(完成队列、数据缓冲区、物理描述子等)管理以及 DMA 发送接收管理等;还有一个是核心代理层,简称 Kernel Agent(KA),它是 VIA 规范中运行于系统核心态下的部分,其地位类似于 VIA 的核心驱动程序.KA 提供的功能包括:

(1) 连接管理:由于 VIA 是面向连接的,VI 之间连接的建立需要 KA 的干预.由于连接管理面向所有的用户进程和创建的 VI,所以在核心态之下实现.

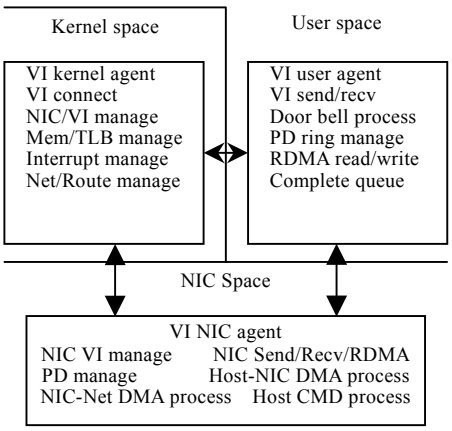

Fig.2 Function structure of MyVIA

图 2 MyVIA 的功能结构图

(2) VI 管理:VI 是 VIA 规范中进程通信的接口,它由 KA 统一管理.用户通信要申请 VI,通信结束后,还要销毁 VI.VI 的申请、创建和销毁以及其 VI/NIC 信息的修改和查询都是由 KA 来完成的.

(3) 内存注册与保护:用户通信使用的缓冲区就是用户向 KA 申请的内存区域,因此 KA 要提供内存区域的限制访问和虚拟地址的转换.这些都属于特权操作,需要在系统的核心态下完成.

(4) 设备驱动:作为 KA 功能的一部分,设备驱动程序提供对不同的网络设备的管理和向 KA 提供统一的设备操作接口.

(5) 网络管理: KA 根据需要,还将负责主机的网络配置、路由信息管理和网络状况的监测.

3 MyVIA 的性能优化技术

Berkeley VIA 也在 Myrinet 的 LANai4 和 LANai9 上实现了 VIA 规范,但通过分析我们发现,它并没有充分发挥 Myrinet 的硬件性能.在 VIA 实现上,Berkeley VIA 把大量的 VIA 工作放在网卡上完成.由于制造工艺和成本决定了 Myrinet LANai 上的嵌入式 CPU 比主机上的通用 CPU 要慢许多,所以简单地把所有的 VIA 处理工作都交给网卡完成不一定就能达到性能最佳.而且 Berkeley VIA 没有充分利用 Myrinet 中各功能部件,使得实现的效率受到影响.

在深入研究 VIA 规范和 Myrinet 硬件的基础上,我们认为提高通信性能的关键是要精简数据传输在主机和网卡上关键路径的指令流,并合理分配主机和网卡的工作量,达到一个最佳性能的平衡点.我们在 MyVIA 的实现上提出了多种性能优化技术,在 KA 上的优化技术主要包括:基于主机的 User TLB(UTLB)技术、基于 Linux 内核的连续内存管理、物理描述子动态缓存技术;在 NA 上的优化技术主要包括基于资源的流水技术、基于硬件 DMA Chain 的流水技术、物理描述子环技术、变长 buffer 管理技术.

通过采用这些优化技术,MyVIA 发送小消息和大消息的性能相对 Berkeley VIA 有较大提高,并在 Myrinet LANai9 上发送 4KB 数据包时,带宽达到 250MB/s,可达到硬件链路最大性能的 96%以上,最小单边延迟为 8.46μs.下面我们通过与 Berkeley VIA 的实现进行比较,对 MyVIA 所采取的性能优化技术进行分析.

3.1 UTLB地址转换技术

目前,合理的地址转换机制主要有:(1) TLB 保存在主机上,由主机进行地址转换;(2) TLB 保存在网卡上,由网卡进行地址转换,其缺点是空间开销大,无扩展性;(3) TLB 保存在主机上,由网卡进行地址转换.Berkeley VIA 采用的是类似第 3 种的地址转换机制,即基于网卡和主机互补的地址转换技术,并在网卡中建立 TLB cache 以缓存常用的虚实地址.它的好处是,避免了在数据传递中主机处理虚实地址转换的开销,有利于局部性好的应用程序.它存在的问题在于可扩展性不强,会产生中断,地址转换性能依赖于网卡 TLB cache 的大小和程序的局部性.

通过分析和多次比较试验,我们在 MyVIA 中采用了类似第 1 种的地址转换机制,即基于主机的 User

TLB(UTLB)技术,其基本思想是在应用程序注册内存时建立一个用户层可访问的 TLB 表,这样,在以后用户进程进行数据传输时可在用户层执行虚实地址转换.对于 VIA 注册内存,MyVIA 在内核中建立了一张数据表,称为注册内存管理表(registered memory manage table,简称 RMMT),注册内存的所有信息都保存在 RMMT 里.在用户打开设备时,就把 RMMT 表区域以只读方式全部映射到用户空间中.这样,当应用程序进行发送、接收操作时,就可在用户态下直接访问 RMMT 的内容,进行 VIA 描述子到物理描述子的转换.

基于主机的 User TLB(UTLB)技术的优势主要有 3 个方面:首先,它的扩展性最好,由于主机上的内存远大于网卡上的内存,在主机建立 TLB 表可有效处理大规模的应用程序;其次,它不受应用程序局部性的限制,如果应用程序的局部性不好,Berkeley VIA 的 MCP(Myrinet control program)会产生大量查询主机 TLB 表的中断,性能通常会降低 10%~40%左右,但采用 UTLB 技术以后,MyVIA 的 MCP 不会产生类似的中断,可提高性能;第三,由于 TLB 表可在用户层访问,所以在进行数据传输时完全在用户态执行,可最大化旁路操作系统对性能的影响.

3.2 基于资源和DMA Chain的流水线处理

Myrinet 网卡在硬件上为通信流水化提供了丰富的支持,除了 CPU 和高速卡上的内存以外,还有一个 Host-NIC DMA 双向通道,一个 NIC-NET DMA 通道和一个 NET-NIC DMA 通道.Berkeley VIA 在 Myrinet LANai4 上的 MCP 程序基本上没有采用流水线技术,所以在发送 2KB 以上大小的数据包的情况下,其带宽性能较差.它在 Myrinet LANai4 上没有充分利用硬件 DMA Chain,而且流水线分段不是很合理.MyVIA 与其相比,在性能提高上做了较大改进.

由于 Myrinet LANai4 不支持硬件 DMA Chain,因此为了充分利用 Myrinet 网卡的硬件资源,我们把网卡上的发送和接收过程进行了逻辑划分,在设计上采用了基于资源的流水通信机制,在实现上采用双 buffer 结构和基于资源的有限状态自动机控制流,并结合物理描述子缓冲机制,极大地提高了网卡的吞吐量.

通过分析可知,对于网卡发送过程,主要包括:物理描述子的处理、接收主机数据缓冲区的处理、发送数据缓冲区的处理和状态字返回处理.对于网卡发送过程,主要包括:接收网络数据包的预处理、物理描述子的处理、发送主机数据缓冲区的处理和状态字返回处理.

根据上述发送和接收的情况,我们对基于 Myrinet LANai4 的 MyVIA 中的优化设计思想是:把发送过程和接收过程分别处理,简化通信执行路径的逻辑复杂性;再把发送过程根据资源的占用阶段划分为 7 段流水线,增加并行性,充分利用资源;最后把接收过程根据资源的占用阶段划分为 5 段流水线,增加并行性,充分利用资源.

由于 Myrinet LANai9 支持硬件 DMA Chain,可减少对不同物理空间的数据进行 DMA 传输的启动开销,简化 MCP 的程序逻辑.所以我们对基于资源的流水线处理技术进行了粗粒度化,把发送过程和接收过程都分为结合双 buffer 结构的两段流水

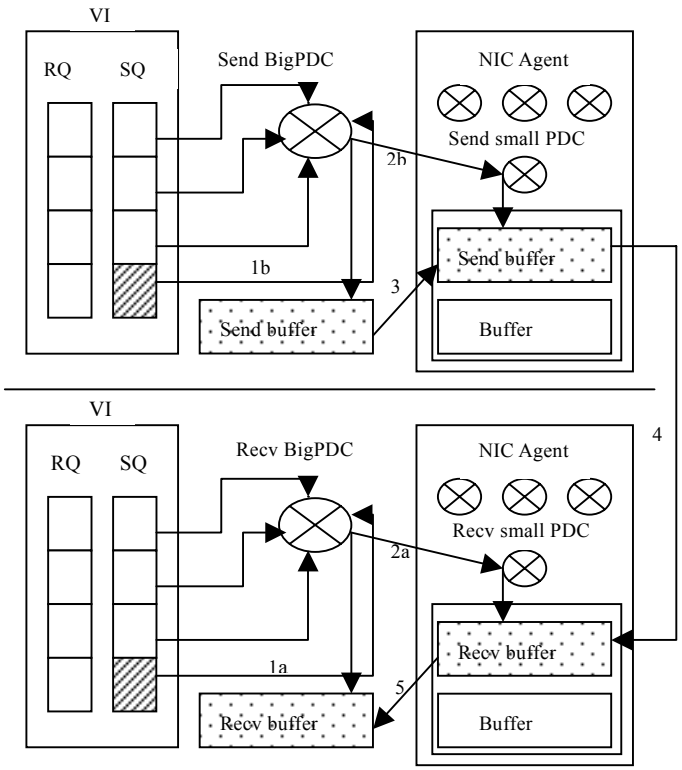

Fig.3 Logical communication of MyVIA
图 3 MyVIA 的通信传输过程

线.发送过程的第 1 阶段是用一个 DMA Chain 操作完成从主机得到要发送的数据并把状态字写回主机,发送过程的第 2 阶段是把数据传输到网络中;接收过程的第 1 阶段是从网络中得到数据,第 2 阶段是把数据和状态字写回主机.

测试数据表明,采用上述技术后,极大地提高了 VIA 的带宽性能.虽然二者的流水线实现机制不同,但 MyVIA 通信传输的总体思路是一致的,通过建立一层抽象接口,保证对上层 Kernel Agent 和 User Agent 透明.

MyVIA 的简化通信流程如图 3 所示,RQ 表示接收描述子队列,SQ 表示发送描述子队列,BigPDC 表示主机上的物理描述子环,SmallPDC 表示 NIC 上的物理描述子环.由于 VIA 的接收操作要在发送操作之前发生,所以图 3 中的(2a)处理过程要在(2b)处理过程之前发生.

3.3 门铃机制的实现

门铃机制用于通知网卡 VI 消费者有发送或接收请求.在 Berkeley VIA 中,VI 的发送及接收队列是在主机内存中实现的,并在网卡内存中开出一块空间实现 VIA 的门铃.当主机发出通信请求时,Berkeley VIA 中的 UA 要执行 PIO 操作对 NIC 写入的 2 字节的门铃信息;而后 MCP 程序根据门铃中的信息,以 DMA 方式从主机内存的队列中取 64 字节 VI 描述子进行下一步处理.对于小数据量的数据,DMA 方式的效率不高,而且 VI 描述子的部分信息对网卡而言是多余的,这都带来了额外的通信开销.

为了减小与门铃相关的通信开销,我们在 MyVIA 中提出了物理描述子环技术和物理描述子动态缓存技术.通过把 VI 描述子转换成物理描述子,使其从原先的 64 字节减少到 16 字节.我们用物理描述子环来完成门铃机制,其基本思想是:把 VI 发送队列和 VI 接收工作队列用网卡上的物理描述子环实现,当 VI 描述子转换为物理描述子以后,直接用主机 CPU 以编程 IO(program IO)的方式写到网卡上的物理描述子环中,把 Berkeley VIA 一次 PIO 和一次 DMA 操作完成的工作用一次 PIO 操作就完成了.通过合理调整环大小可保证主机与网卡处理的并行程度.

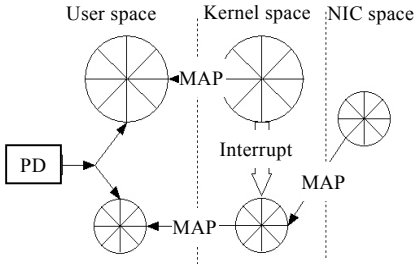

Fig.4 Physical descriptor ring

图 4 物理描述子环

Myrinet LANai4 上只有 1MB 内存,为了支持多个 VI,则网卡上每个 VI 的发送及接收队列的物理描述子环所开辟的缓冲区不能很大.为此,我们提出物理描述子动态缓存技术(如图 4 所示),其基本思想是:在主机内存中开辟一块内存区域作为物理描述子的缓存(简称大环),由于主机的处理速度比网卡上 CPU 的处理速度快,所以当网卡上 VI 的物理描述子环(简称小环)写满后,后续的物理描述子就暂存到主机内存的缓存中.当小环写满,暂时把 PD 缓存在大环的时候,主机把一个标志位置位,通知网卡大环中有数据,且主机暂停向小环写入;当网卡上小环中未处理的物理描述子个数到达一个下限值,并检测到大环中有数据的标志位时,就向主机发出中断;主机在中断处理过程中把大环中的物理描述子搬移到小环中去.小环中剩余物理描述子的下限值应仔细选取,它不能取得太大,否则会产生太多中断,对主机造成影响.但也不能取得太小,应能保证 MCP 在处理余下的 PD 过程中,主机中断处理能够把缓存在大环中的 PD 写到小环中,这样才能使 MCP 程序全速进行数据的发送、接收,不影响效率.

小环中剩余物理描述子的下限值通过具体实验确定,而且大环长度和小环的长度也是根据具体实验来确定的.实验发现,当小环长度和下限值调整为一定值时,如果使用 Myrinet LANai4,则为了减少中断次数而进一步增加小环长度的实际效果对通信性能的影响基本可忽略不计;如果使用 Myrinet LANai9,则基本不会产生中断.

3.4 内存管理

在内存管理中,Berkeley VIA 一次 DMA 操作最大只可传递 64KB 的数据,且相关开销较大.为了提高主机与网卡间 DMA 相关操作的性能,我们还在 MyVIA 中首次提出了基于 Linux 内核的连续内存管理技术和变长 buffer 管理技术,其基本思路是在 Linux 内核中设计了一个大块连续物理内存管理 Agent-kmem Agent,使用类 Buddy 算法来有效地分配与回收页面块,负责完成 VIA 用户程序的内存申请、注册和释放等;NIC Agent 负责完成对 NIC buffer 的 DMA 操作和管理,NIC buffer 的长度只受 NIC Agent 的 buffer 池大小的限制.Myrinet 网卡和

其他大多数高性能网卡不支持虚拟内存地址的 DMA 操作,但应用程序在用户层访问的是虚拟内存地址空间,其在一般情况下物理地址不连续,所以必须把虚拟内存地址转变成物理内存地址,再进行多次 DMA 传输.通过基于 Linux 内核的连续内存管理技术,保证申请的虚拟内存在物理空间上连续,这样对于多页面的传输只进行一次 DMA 操作即可,从而大大减少了 DMA 传输带来的的相关开销.通过进一步在 NIC Agent 上采用变长 buffer 管理,使得一次 DMA 操作最大可传递 120KB 的数据,且在实现上只受网卡硬件的限制.关于基于 Linux 内核的连续内存管理和变长 buffer 管理的实现细节描述见文献[8].

4 性能分析

我们分别在两种实验环境中对 MyVIA 进行测试,其中:环境 a 包括支持双 Intel Pentium III 733MHZ 的 CPU,有 512M 内存和 33MHZ/32 位 PCI 总线的主机两台、Myrinet LANai4.1 网卡和相应 16 端口的 Myrinet 交换机、Intel Pro100 百兆以太网卡;环境 b 包括支持双 AMD AthlonXP 1.4GHZ 的 CPU,有 512M 内存和 66MHZ/64 位 PCI 总线的主机两台、LANai9 M3M-PCI64C-2 网卡和对应的 32 端口 Myrinet-2000 交换机.

在环境 a 下,我们对 Berkeley VIA3.0 及 MyVIA 的单边延迟和带宽进行了测试,测试程序为 Berkeley VIA 中的 pingpong 和 window 程序.测试结果如图 5 所示.图 5 中单边延迟的测试结果是用所测大小的数据包在两台主机间往返 1 000 次后求出单方向上的平均时间得到的.可以看到,Berkeley VIA 与 MyVIA 的测试结果曲线几乎重叠,相差不大.图 5 中带宽的测试是由一台主机连续向另一台主机发送 2 000 个所测大小的数据包,而无论接收方是否接收.由图中可以看到,在发送小于约 2 000 字节大小数据包的情况下,MyVIA 的带宽性能比 Berkeley VIA 略低,而在发送 2 000 字节以上大小数据包的情况下,MyVIA 的带宽性能要比 Berkeley VIA 高.通过测试,Berkeley VIA 的带宽最大可达到 47.46Mbytes/sec,而 MyVIA 可达到 76.2Mbytes/sec.

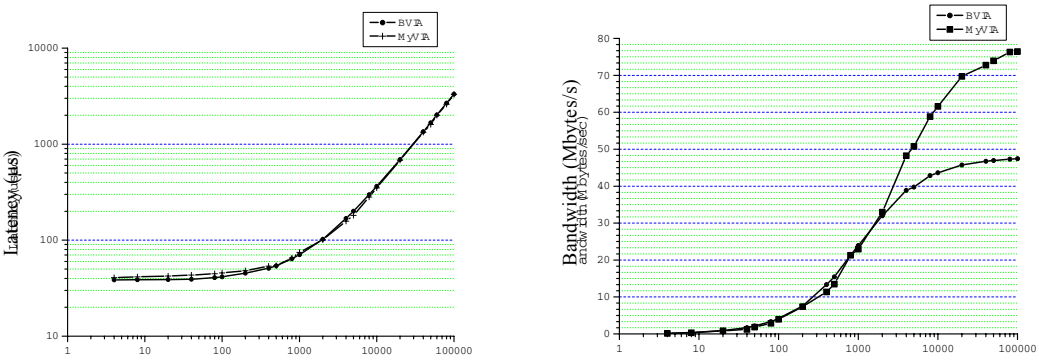

Fig.5 Performance analysis on MyVIA and Berkeldy VIA in Myrinet LANai4

图 5 MyVIA 与 Berkeldy VIA 在 Myrinet LANai4 上的性能分析

在发送 2KByte 下的小数据包时,MyVIA 的带宽性能比 Berkeley VIA 低的原因有多方面,主要是由于 MyVIA 采用了基于资源的流水技术.这导致在数据传输的关键路径上存在对网卡上各资源进行开锁、解锁的竞争操作,使 MCP 程序的控制逻辑复杂了许多.而且网卡上主频为 33MHz 的 CPU 的处理能力较弱,这样在连续发送小数据包的情况下,网卡上主机-网卡间 DMA 控制器与发送 DMA 控制器之间的并行程度不够大.在 2KByte 以上的数据包时,网卡上主机-网卡间 DMA 控制器与发送 DMA 控制器的流水并行处理所带来的好处抵消了 MyVIA 在网卡上流水控制过程的开销,从而使带宽比起 Berkeley VIA 得到了进一步的提升.

在环境 b 下,我们对 GM1.5[9],Berkeley VIA3.0 及 MyVIA 进行了同样方式的测试,GM 的测试程序采用 GM 软件包中的 allsize 程序.测试结果如图 6 所示.可以看出,由于 Myrinet LANai9 的硬件性能出色,在发送 32KByte 大小的数据包的情况下,三者都可以达到 250MB/s 的传输速率.但在发送小于 8Kbyte 大小的数据包时,MyVIA 的性能可比 Berkeley VIA 的性能高出接近一倍,而且 MyVIA 发送 4KB 数据包时的带宽就可达到 250MB/s,最小单边延迟为 8.46μs.这主要是由于 MyVIA 的各种优化技术充分发挥了 Myrinet LANai9 的硬件性能和消息传递的并行性.MyVIA 的性能比 GM1.5 的性能略低,其主要原因是由于 GM 没有按照 VIA 规范实现,去掉了一些

VIA 规范规定的操作,而且充分利用了 LANai9 的某些特性.

我们还采用 Intel Pro100 以太网卡对 M-VIA 进行了测试,得到的最大带宽为 11.6MB/s,最小延迟为 22μs,这个性能与慢速的百兆网卡的硬件性能相比无太大差距.但根据文献[4],M-VIA 在 GNIC-II Gigabit Ethernet 网卡上的最大带宽为 59.7MB/s,相对于网卡硬件的最大带宽有较大的差距.通过对 M-VIA 源代码的分析,我们认为其问题主要在于 M-VIA 把 VIA 的大部分工作放在主机上完成,且在数据传输过程中不能避免中断的产生,这些都使其不能很好地发挥高性能通信硬件的性能.

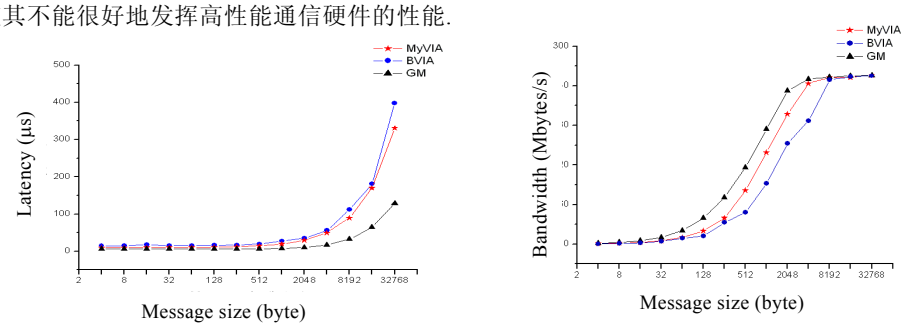

Fig.6 Performance of MyVIA,BVIA and GM on LANai9

图 6 MyVIA,BVIA 与 GM 在 LANai9 上的性能分析

5 结 论

通过对当前的 VIA 实现,尤其是对 BerkeleyVIA 进行深入的分析与比较,本文在通信硬件 Myrinet 上设计并实现了一个高性能、可扩展的 VIA 系统——MyVIA.实验数据表明,MyVIA 充分挖掘了 Myrinet 的硬件优势,发挥了通信过程的流水并行性,减少了通信关键路径.目前,MyVIA 的峰值性能虽然很高,但还有许多需要改进和扩展的地方.下一步我们将在此基础上进一步优化并扩展 MyVIA 的性能和功能,以更好地支持上层应用软件(MPI,TCP/IP 等),完成 MyVIA 的实用化.可访问网址http://hpclab.cs.tsinghua.edu.cn/~chenyu/myvia.html得到有关 MyVIA 和它支持的 MPI 的最新进展.

致谢 在 MyVIA 项目的开发过程中,我们得到了 UC Berkeley 大学的 Philip Buonadonna 和 Myricom 公司的 Patrick Geoffray 博士等人的热情帮助,在此表示感谢.

References:

[1] Bhoedjang RAF, Rühl T, Bal HE. User-Level network interface protocols. IEEE Computer, 1998,31(11):53~60.

[2] Virtual interface architecture specification. Version 1.0, Technical Report, Compaq, Intel and Microsoft Corporations, 1997.

[3] Buonadonna P, Geweke A. An implementation and analysis of the virtual interface architecture. In: Proceedings of the Super Computing'98. Orlando: IEEE Computer Society and ACM SIGARCH, 1998. 7~13.

[4] Bozeman P, Saphir B. A modular high performance implementation of the virtual interface architecture. Technical Report, Lawrence Berkeley National Laboratory, 2000.

[5] Boden NJ, Cohen D. Myrinet——a gigabit-per-second local-area network. IEEE MICRO, 1995,15(1):29~36.

[6] LANai 9. Technical Report, Myricom Ltd., 2000.

[7] Gusella R. A measurement study of diskless workstation traffic on an Ethernet. IEEE Transactions on Communications, 1990, 38(9):1557~1568.

[8] Chen Y. A design scheme of high performance VIA based on Myrinet. Technical Report, Beijing: Tsinghua University, 2000 (in Chinese). http://hpclab.cs.tsinghua.edu.cn/~chenyu/myvia.html.

[9] The GM message passing system. Technical Report, Myricom Ltd, 1999.

附中文参考文献:

[8] 陈渝.一个基于 Myrinet 的高性能 VIA 设计方案.技术报告,北京:清华大学,2000.