一种基于 IP 的适于数据中心网络的高速通信协议的设计*

王继晨 邹垚昭 江 愿 潘志浩
(上海大学通信与信息工程学院 上海 200072)

摘 要：设计了一种 C/S 结构的、基于 IP 的自定义快速文件传输协议 CFFTP (customized fast file transport protocol)协议，简述了协议交互流程，以及协议的软件实现流程。该协议能充分利用数据中心局域网下的丰富带宽资源，实现数据、文件的可靠快速传输。仿真结果显示，数据传输速率可达近 10Gbps 的线速，实验结果显示，虽然受到 Server 端磁盘文件读取速度的影响，数据传输速率仍然可以达到 6Gbps 的最高速率。

关键词：CFFTP；快速可靠文件传输；数据中心；数据存储
中图分类号：TP393.04 **文献标识码**：A **国家标准学科分类代码**：510.4050

Design and implementation of IP-based High-speed and reliable communication protocol

Wang Jichen Zou Yaozhao Jiang Yuan Pan Zhihao
(School of Communication and Information Engineering, Shanghai University, Shanghai 200072, China)

Abstract: This paper design and implement a new kind of communication protocol CFFTP (Customized Fast File Transport Protocol) which is high-speed and reliable. We propose the interactive process of CFFTP, and how to realize it on server and client. This protocol can make full use of the LAN bandwidth to increase data transmission speed over network. Simulation results shows that the data speed is about 10 Gbps. We also test this new protocol on our TestBed, due to the restriction of the reading rate on RAID(600－800MB/s), the transmission rate is 6 Gbps.

Keywords: CFFTP; high-speed and reliable file transmission; data center; data storage

1 引 言

近年来，随着互联网以及云计算的发展，人们对数据的存储有了巨大的需求，越来越多的数据存储于云数据中心里[1]。同时我们发现在云数据中心中使用 TCP/IP 协议实现高速文件数据传输的研究较少，这是因为 TCP/IP 协议是针对低带宽、低时延的广域网设计的，考虑了广域网中的多种不可靠因素，添加了很多握手、重传的机制以及相关报头信息，使得协议开销大，传输时延高、带宽利用率低。

本文提出了一种 C/S 结构的，基于 IP 的[2]高速可靠的数据传输协议，用于数据中心中磁盘阵列的快速文件访问及数据传输。经过仿真及实验证明该协议有较高的传输速率。

2 CFFTP 协议介绍

2.1 CFFTP 协议 Server-Client 数据交互流程

如图 1 所示，CFFTP 协议的处理流程主要分为 3 部分：文件打开处理、读文件内容处理、文件关闭处理。

1)文件打开处理

Client 端根据上层的需求向 Server 端发起文件读请求，然后等待 Server 的应答；Server 收到文件读请求帧后，在磁盘阵列中搜索该文件，若文件存在，Server 发送文件读请求应答帧通告 Client；当 Client 收到文件读请求应答帧后，文件打开处理结束。若文件不存在，Server 直接发送文件内容完成帧通告 Client 所请求文件不存在，若 Client 在未收到文件内容传送帧的情况下就收到了“文件内容完成帧”，则认为请求文件不存在。

收稿日期：2015-01
* 基金项目：国家自然基金国际合作重点(61420106011)、国家自然科学基金资助重点(61132004，61275073)、上海市科学技术委员会科研计划重点(14511100100)、上海市科学技术委员会科研计划重点(11511502500，11510500500)项目

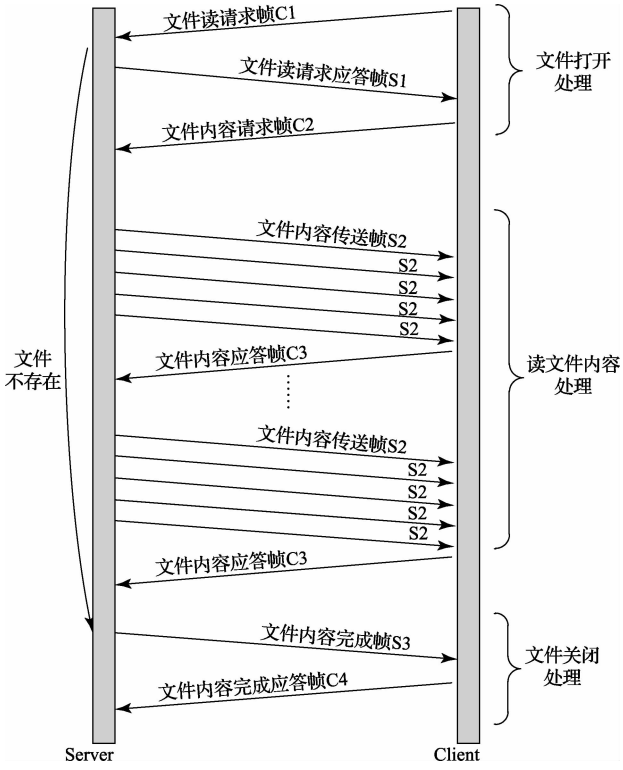

图 1　Server 与 Client 数据交互流程

2)读文件内容处理

当 Client 收到 Server 的文件读请求应答帧后，根据应答帧中的文件存储信息如：文件名、文件大小及对应的端口号向 Server 发送文件内容请求帧，并等待对方的文件内容传送帧；Server 收到文件内容请求帧后，将文件内容切分为多个分片，每次发送多个分片，并等待 Client 的文件内容应答帧，重复此过程，直到文件发送完成。

3)文件关闭处理

当 Server 收到 Client 对最后一组文件内容帧的 ACK 后，向 Client 发送文件内容完成帧，通知 Client 文件内容发送完成，并关闭此文件的读操作，释放端口资源。

2.2　协议可靠性保证及数据速率控制

考虑到局域网网络环境相对可靠，Server 在每个周期发送多个分片，并在最后一个分片中设置标志位，标识该分片为最后一片；若 Client 收到具有该标志位的分片后且收到的分片数量正确，则向 Server 发送文件内容应答帧作为 ACK 通知 Server，完成该周期的数据传送；若在计时时间内 Server 未收到 ACK 或 Client 收到分片数量不足，则启动超时重传机制，Server 端重新发送该周期内的所有分片。

这种方式一方面减少了控制报文的数量，进一步提高了数据传输速率；同时通过控制每次发送分片的数量可以控制数据的发送速率，避免网络的拥塞。

2.3　协议帧格式

协议帧格式如图 2 所示，其中包括 14 字节的以太网首部、4 字节的以太网尾部、20 字节的 IP 首部、12～30 字节的 CFFTP 协议首部，文件内容存放在 CFFTP 首部之后的数据内容部分，协议中负责可靠通信以及超时重传机制都建立在 CFFTP 首部之中。

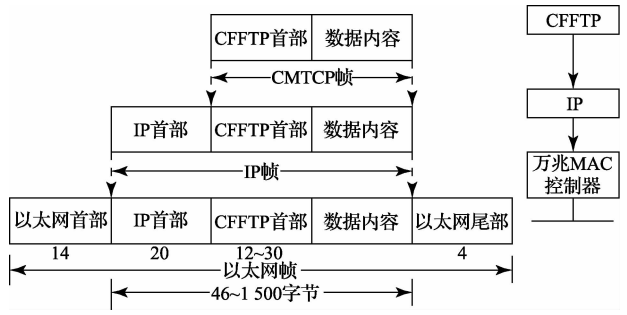

图 2　CFFTP 数据帧格式

在 CFFTP 协议的首部中，一般要包括请求文件名、请求文件大小、数据帧标识、该帧所含数据内容大小等相关信息，用于在 Server 或 Client 进行协议的解析。图 3 为文件内容传送帧的声明格式。

```
struct CFFTP_FILE_CONTENT_SEND{
  char     filename[6];      //文件名
  uint8_t  op;               //帧标识
  uint16_t pkt_len;          //数据包大小
  uint16_t file_no;          //
  uint8_t  reserved[2];
  uint64_t memaddr;
  char     content[BUF_MAX];//文件内容
};
```

图 3　文件内容传送帧格式

3　仿真与实验

3.1　系统结构

系统物理结构如图 4 所示，整个系统由 N 台组件、万兆交换机及盘阵服务器 3 部分组成。每台组件由 4 片 Xilinx XC6VLX550T FPGA 组成，作为 Client 端向磁盘盘阵请求文件，每个 FPGA 都有两个万兆以太口，通过光电转换模块及多模光纤接入到万兆交换机中；

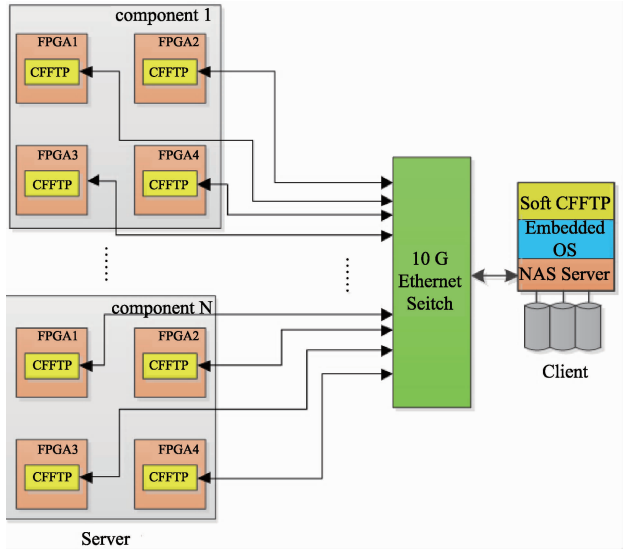

图 4　多 FPGA 访问磁盘盘阵结构

磁盘盘阵作为 Server 端,采用 NAS 存储方式,容量为 416 TB(STATA)或者 124.8 TB(SAS)的 D-Fusion 3500 服务器,其上层运行 CentOS 6.2 64 位操作系统。并运行 CFFTP Server 端软件用于响应 FPGA 端的盘阵文件读取请求,并从磁盘阵列中读取相应文件发送出去;

Server 与 Client 间通过具有 288 口以太网 10 G 的 Cisco 交换机相连。

3.2　CFFTP 协议 Server 端软件实现流程

Server 端软件实现流程如图 5 所示。在 Server 端,使用 Socket 编程的方式,使用原始套接字实现基于 IP 的 CFFTP 协议[3-6];首先解析收到的数据包报头,根据帧标识区分收到的数据包,按照图 5 的软件实现流程做相应包处理。

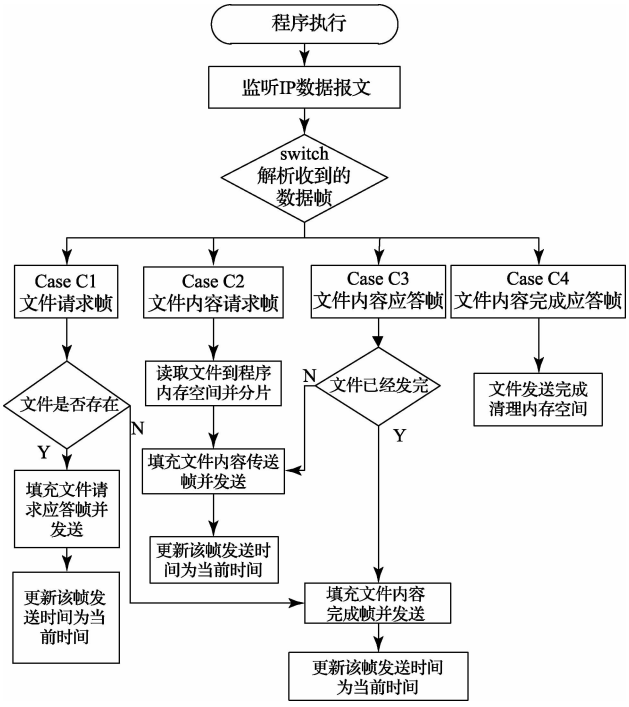

图 5　Server 端软件实现流程

3.3　CFFTP 协议 Client 端软件实现流程

由 FPGA 实现的 Client 端框架如图 7 所示[7-8],图 6 为组件内部图。Client 端把接收到的光信号转化为电信号,用 Xilinx 提供的 XAUI IP 核(Intellectual Property)把 4 路串行数据转化为 64 位的 XGMII 信号;XGMII 信号通过 Xilinx 提供的 10G MAC IP 核对接收到的以太帧进行 CRC 校验以及前导码过滤;帧分类模块即识别协议类型判断接收到的以太帧是什么类型的帧如 ARP 请求帧、ARP 应答帧及 IP 帧;如果是 IP 帧则继续分类是 ICMP 请求/应答帧还是我们自定义的传输层协议 CFFTP,收到的文件信息也可通过内存存储管理模块存入本地 DDR3 中[9]。

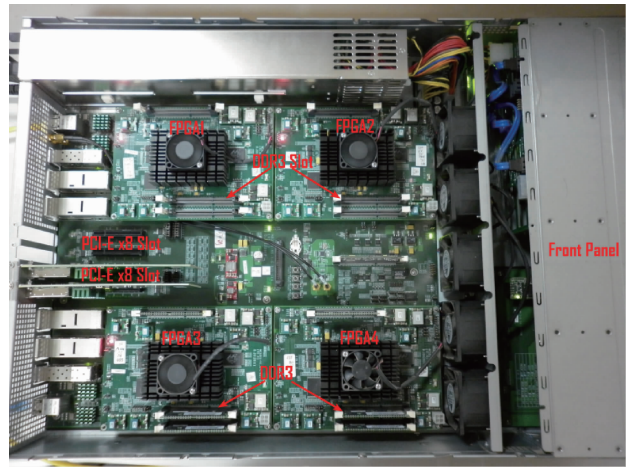

图 6　Client 组件内部

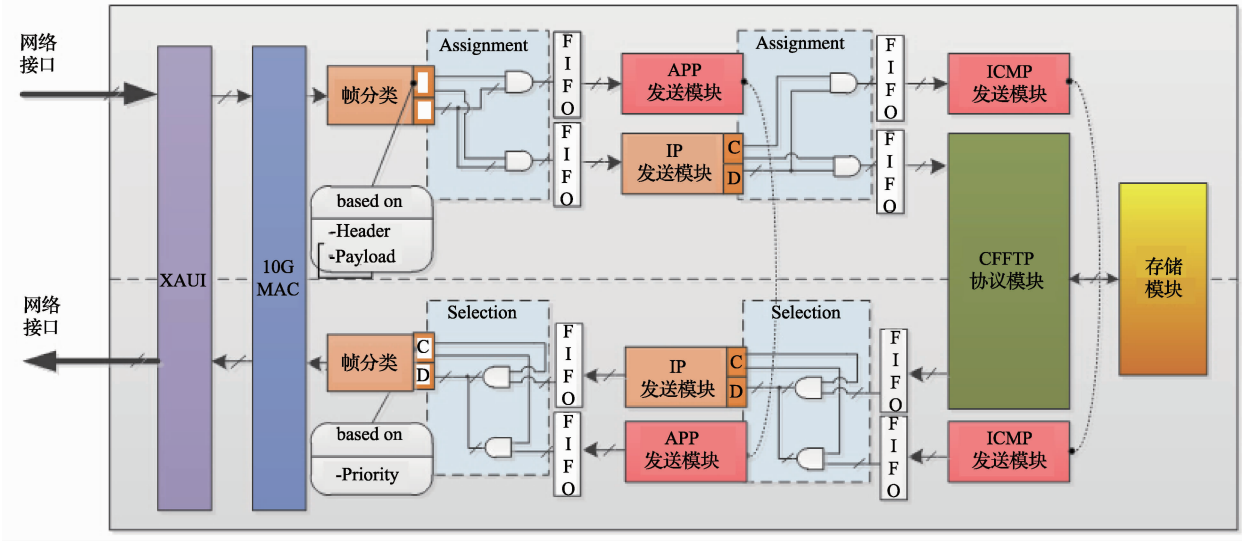

图 7　Client 端软件实现流程

3.4　仿真结果

使用 Modulsim 10.1 进行 FPGA 端的仿真，在文件内容发送阶段，编写模拟磁盘阵列的 Testbench，不断发送 1 460 Byte的内容帧，每发 64 个帧等 FPGA 盘阵访问逻辑过来的一次内容应答帧然后继续发送下一个组 64 帧，直到发满 1.864 MByte。Modelsim 10.1 仿真测试结果如图 8 所示。

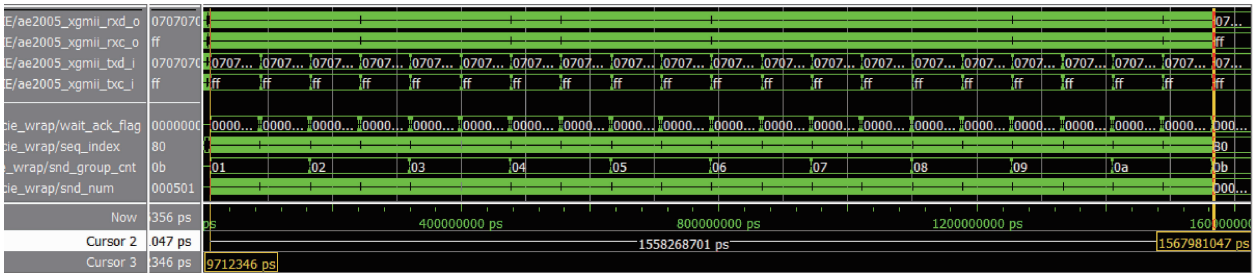

图 8　Modulsim10.1 仿真结果

整个文件处理请求处理共用了 1.558 ms，计算得知数据传输速率可达 9.57 Gbps，接近 10 Gbps 的线速。

3.5　实验结果

选取不同大小的文件进行测试，FPGA 向磁盘盘阵请求大小分别为 1 KB、10 KB、100 KB、1 MB、10 MB、100 MB、1 G 的文件。在每次实验中，设置每次 Server 端发送不同的分片数，并统计文件完成时间，计算文件传输速率，将多次实验结果绘制了如图 9 所示。

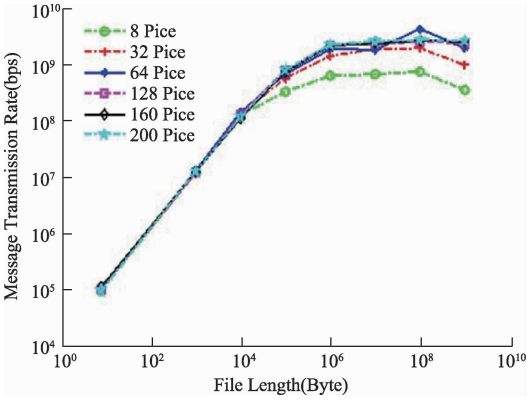

图 9　不同分片、不同文件大小数据传输速率

结果显示，越大的文件传输速率越高，同时每组的分片数越多，传输速率也越高但最终都会趋近一个最高值。同时传输速率只能达到 6.1 Gbps。

该结果表明 CFFTP 协议对大文件的传输有较高的传输速率，因为当文件较小时，每一数据帧中包含的有用信息相对更少，带宽利用率不高；

实际传输速率与仿真速率有一定差距，磁盘阵列的读取速度限制了系统的整体表现，在后来的测试中，发现磁盘的读取速率基本在 600 MB/s 到 800 MB/s，因此理论上磁盘阵列的最高访问速率不高于 6.4 Gbps，这直接限制了整个系统的传输速率，如果采用读写速率更高的磁盘阵列设备，相信该协议的传输速率会有较高的提升。

4　结　　论

该文提出了一种基于 IP 的自定义高速、可靠的数据传输协议，将该协议部署与磁盘阵列与 FPGA 上，实现了二者的高速文件传输，仿真结果显示该协议能以接近 10 Gbps的线速传送文件，而实验结果显示，虽受到磁盘阵列读写速率的限制，磁盘阵列与 FPGA 间仍能以最快的速率实现数据传输。

参考文献

[1]　陈小军，张璟. 面向高效能计算的虚拟化技术研究综述[J]. 系统仿真学报，2012，24(4)：741-747.

[2]　刘蓟铭，朱杰，许智斌. FPGA 在 TCP/IP 实现中的应用[J]. 电子测量技术，2006，29(2)：83-84.

[3]　孙德辉，王科，史运涛. 基于 SOCKET 编程实现 IP 传感器的网络通信[J]. 电子测量技术，2008，31(9)：97-99.

[4]　王渊，赵宇. 嵌入式 Linux 网络通信的实现[J]. 电子测量技术，2007，30(6)：94-96.

[5]　李彦，李锵. 基于嵌入式 Linux 系统的双网卡大数据传输[J]. 电子测量与仪器学报，2014，28(9)：1027-1032.

[6]　张志博，孙长瑜. 基于 TCP/IP 的浮标网络通信系统设计[J]. 电子测量技术，2008，31(2)：7-11.

[7]　周云霞，赵跃龙，杨希. 智能网络磁盘通信协议的研究[J]. 通信学报，2007，28(4)：101-107.

[8]　NORONHA R，CHAI L，TALPEY T，et al. Designing NFS with RDMA for security，performance and scalability [C]//2007. ICPP 2007. International Conference on Parallel Processing. IEEE，2007：10-14.

[9]　吕烁，文中领，杨帆，等. Infiniband 通道磁盘阵列中目标器的设计与实现[J]. 计算机研究与发展，2012 (S1)：164-171.

作者简介

王继晨，1988 年出生，硕士研究生。主要研究方向数据中心网络拓扑、协议改进等。

邹垚昭(通讯作者)，1992 年出生，硕士研究生。主要研究方向数字通信、宽带接入等。

E-mail：imei857@shu.edu.cn