

# 基于FPGA的千兆以太网数传系统设计

李洋<sup>1,2</sup> 禹卫东<sup>1</sup> 胡晓<sup>1</sup> 刘霖<sup>1</sup> 张彪<sup>1,2</sup>

(1. 中国科学院电子学研究所 北京 100190; 2. 中国科学院大学 北京 100039)

**摘要:** 利用 Xilinx 公司的 VC707 开发板设计了基于 FPGA 的千兆以太网数据传输系统。开发板上集成了以太网的 MAC 层硬核以及 PHY 层芯片, 提供了实现以太网数据传输的基础。该系统同时实现了 UDP 与 IP 协议, 允许上位机与系统之间进行基于 UDP 协议的数据交互, 并可以利用板卡的 DDR3 实现数据的缓存。实践证明, 该系统可以实现数据高速、准确的传输。

**关键词:** FPGA; 千兆以太网; UDP/IP; 数据传输

**中图分类号:** TP393.1 **文献标识码:** A **国家标准学科分类代码:** 520.304

## Design of gigabit ethernet data transmission system based on FPGA

Li Yang<sup>1,2</sup> Yu Weidong<sup>1</sup> Hu Xiao<sup>1</sup> Liu Lin<sup>1</sup> Zhang Biao<sup>1,2</sup>

(1. Institute of Electronics, Chinese Academy of Sciences, Beijing 100190, China;

2. University of Chinese Academy of Sciences, Beijing 100039, China)

**Abstract:** An implementation of data transmission system based on FPGA gigabit Ethernet is designed with Xilinx VC707 evaluation board. The board integrates MAC hardcore and PHY chip, therefore supplying the foundation for Ethernet data transmission. The design also implements the UDP/IP protocol stack and allows data transmission based on UDP protocol between the host computer and the system. Besides, the data could be cached with the DDR3 that evaluation board possesses. Practices certify that the system is capable of implementing high-speed and accurate data transmission.

**Keywords:** FPGA; gigabit ethernet; UDP/IP; data transmission

## 1 引言

近几年来,随着半导体产业的快速发展,FPGA 芯片的性能也得到了提升,由于其具有密度高、速度快、功率小等优点,因此已经成长为提高系统集成度与可靠性的最佳选择之一。同时,FPGA 并行运算的特点使得其在雷达、遥感等涉及数字信号处理的领域得到了极为广泛的应用<sup>[1]</sup>。为此,实现计算机与 FPGA 的通信,将数据高效的传输到 FPGA 上来进行运算就显得非常重要。计算机与 FPGA 的数据交互一般可以通过串口、PCIE 总线以及以太网来实现。但是由于串口传输的速率一般较慢,并且串口和 PCIE 均无法实现远距离传输,所以为了兼顾传输速率与系统使用的灵活性,一般选择以太网接口来实现 FPGA 与上位机的数据传输<sup>[2]</sup>。千兆以太网作为局域网标准的一种重要形式,可以提供高速、远距离的数据交互通道,已经在局域网实现中得到广泛的应用,并且在嵌入式系统上的实现也逐渐受到关注。

研究了基于 FPGA 的千兆以太网数据传输解决方案,提出了一种实现 FPGA 与上位机通过以太网接口进行点对点通信的方法,通过例化乒乓 FIFO 来提高以太网接口的发送速率,测试表明该系统充分利用并发挥了千兆以太网的性能。

## 2 千兆以太网系统实现

### 2.1 数据传输方案设计

本系统数据传输的硬件平台使用 Xilinx 公司的芯片型号为 xc7vx485T 的 FPGA 开发板 VC707。选择该开发板的原因主要有以下 3 点:

1) 该开发板集成了 MAC 硬核以及 SGMII (serial gigabit media independent interface) 接口,可以完成 MAC 层功能以及其与 PHY 层接口的实现;

2) 该开发板集成了 PHY 层芯片,型号为 Marvell M88E1111-BAB1C000,可以完成 PHY 层的功能;

3) 该开发板集成了 1 GB 的 DDR3 资源,可以利用其来

完成对上位机发送至 FPGA 的数据的缓存。

因为该开发板已经嵌入了 PHY 层芯片,所以直接将板卡与上位机通过 RJ45 对双绞线相连即可<sup>[3]</sup>,系统的模块化设计框图如图 1 所示。其中 MAC 层以及 MAC 层与 PHY 层的 SGMII 接口均可以直接调用 IP 核来完成,通过使用 IP 核可以大大缩短开发周期并保证正确性。

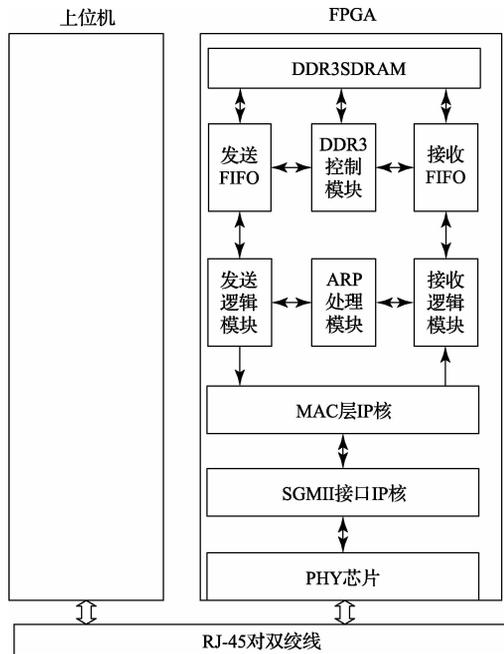


图 1 数据传输方案总体设计

系统中各个模块的主要作用为:

1)接收逻辑模块:接收上位机的 ARP 请求,提取上位机的 MAC 地址与 IP 地址,以及接收上位机发送的 UDP 数据报并提取数据。

2)发送逻辑模块:发送 ARP 应答给上位机,以及将要发送的数据封装 UDP 首部、IP 首部以及以太网帧首部,并传递给 MAC 层 IP 核。

3)DDR3 控制模块:调用 DDR3 用户接口,将接收 FIFO 得到的数据依次存入 DDR3,以及在发送时将数据从 DDR3 读出,送入发送 FIFO。

4)发送/接收 FIFO:由于 FPGA 以太网的工作时钟为 125 MHz,DDR3 用户接口的工作时钟为 200 MHz,因此本系统需要利用 FIFO 来解决以太网与 DDR3 的跨时钟域数据传输的问题。

5)MAC 层 IP 核:完成以太网帧的前导码、帧起始定界符的封装(发送)与解封(接收)。

6)SGMII 接口 IP 核:完成 GMII (gigabit media independent interface)接口与 SGMII 接口的桥接功能。

系统在发送数据时,将 DDR3 中的数据通过发送逻辑以及 MAC 层 IP 核进行数据封装,再通过 SGMII 接口 IP 核输出数据送入 PHY 芯片,在 PHY 层完成数据编码后经

由 RJ45 接口发送给上位机;而系统接受数据就是发送数据的逆过程,上位机发送的数据经过 RJ45 接口到达 PHY 芯片,在 PHY 层完成解码后通过 SGMII 接口将收输出数据送入 MAC 层 IP 核以及接收逻辑模块,完成数据提取或指令执行的工作,并将数据存入 DDR3。

该系统收发数据时选择了面向无连接的 UDP 协议,原因主要有两点。

1)UDP 协议以及 IP 协议都有校验和,有一定的检错能力,并且相对于面向连接的 TCP 协议来说,UDP 协议不需要在通信时进行三次握手等工作,因此拥有更高的效率与更低的延时<sup>[4-5]</sup>;

2)UDP 协议是传输层的协议,在发送 UDP 数据时需要选择源端口号与目的端口号,这样可以通过改变端口号来告知开发板该帧发送的是数据还是指令,便于 FPGA 进行下一步的工作。

接下来将分别介绍 ARP、UDP/IP 协议以及系统中核心的发送逻辑与接收逻辑模块。

## 2.2 ARP 协议

ARP(address resolution protocol),即地址解析协议,为 IP 地址到对应的硬件地址之间提供动态映射。在 TCP/IP 网络环境下,网络层会为每个主机分配一个 32 位的 IP 地址,但是要完成物理链路上数据的传输,首先必须知道对方主机的 MAC 地址,这样就引出 IP 地址转换成 MAC 地址的问题。如本系统所应用的以太网,数据是以 MAC 帧的格式在物理链路上传输,帧头部需要填充 48 位的目的 MAC 地址,如图 2 所示。下面是以太网环境中 ARP 具体的协商过程:

1)首先主机以广播的形式发送一份称作 ARP 请求的以太网数据帧,该数据帧会到达连接到这个网络中的每个主机。其目的 MAC 地址为 FF-FF-FF-FF-FF-FF,并且该帧中也包含了目的 IP 地址的信息。其隐含的意义为网络中主机与目的 IP 地址相同,就请发送 ARP 应答帧。

2)目的主机在收到该 ARP 请求后,就立即发送包含有自己 MAC 地址信息的 ARP 应答帧。

3)发送主机收到 ARP 应答数据帧后,提取目的主机的 MAC 地址,实现该 IP 地址和 MAC 地址的映射。此时,就可以发送数据到目的主机了。

目的 MAC 地址	源 MAC 地址	帧类型	硬件类型	协议类型	硬件地址长度
协议地址长度	操作	源 MAC 地址	源 IP 地址	目的 MAC 地址	目的 IP 地址

图 2 ARP 协议帧格式

## 2.3 IP 和 UDP 协议

为了完成数据的封装,需要在以太网帧的基础上,将 IP 和 UDP 的首部添加在图 3 所示的位置<sup>[6]</sup>。首先介绍 IP

首部的格式,这里对协议部分、校验和进行说明。协议部分字长为 8 位,这个字段用来标明数据报的携带的传输层数据使用的是何种协议,以便目的主机的 IP 层将数据上交给指定的进程。常用的一些协议和相应的协议字段值是:UDP(17),TCP(6),ICMP(1);就本系统而言,使用的传输层协议是 UDP,因此在此字段应该填写 0x11。首部校验和字段只校验数据报的首部,不包括数据部分。IP 校验和

的计算方法是:将 IP 数据报首部看成 16 位字的序列,先将校验和字段置零,然后依次将每个 16 位字进行相加,需要注意的是,如果相加的结果产生进位,则将进位加到末位,按照这样的计算方法将所有 16 位字序列相加后,最后再取反就得到了校验和<sup>[7]</sup>。收到数据报后,将首部的 16 位字的序列再相加一次,若首部未发生任何变化,则和必为全 1,否则即认为出差错,并将此数据报丢弃。

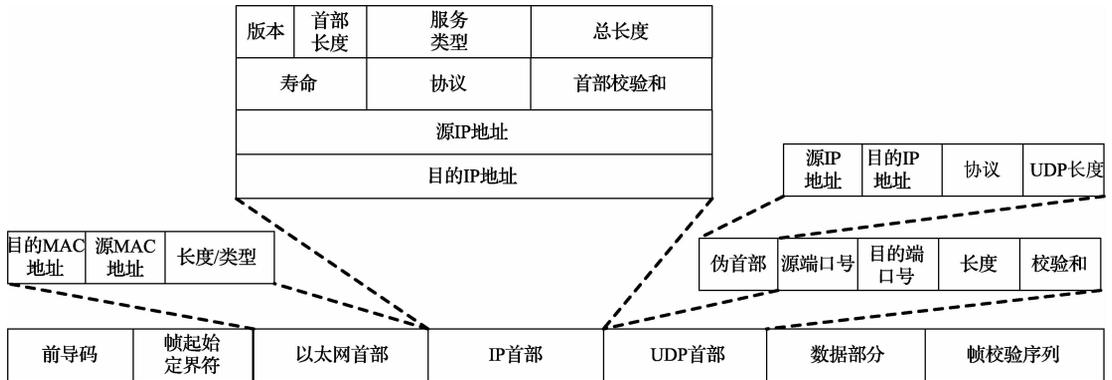


图 3 以太网数据封装格式

用户数据报协议 UDP 在 IP 的数据报服务之上增加了端口的功能,这样上位机与板卡就可以通过端口号来表明进程。在本系统中,当传输的 UDP 协议的目的端口号为 8080 时,则视为数据,当目的端口号为 50000 时,则视为发送完成的指令。UDP 的首部字段较为简单,只有 8 个字节,分别为源端口字段、目的端口字段、长度字段以及校验和字段。下面对校验和字段的计算方法进行说明。UDP 数据报在计算校验和之前要增加 12 个字节的伪首部,所谓“伪首部”是因为这种首部并不是 UDP 数据报真正的首部,其组成如图 3 所示。与 IP 校验和计算不同的是,UDP 首部校验和的计算需要考虑数据。在计算时,仍然将伪首部、首部和数据部分看成一个 16 位字的序列,若 UDP 数据报的数据部分不是偶数个字节,则要填入一个全零字节,然后同 IP 数据报的计算方式相同,按照二进制反码的

方式求这些 16 位字序列的和,再将结果写入 UDP 的检验和字段后,就可以发送这样的 UDP 数据报了。

### 2.4 接收逻辑模块

接收逻辑模块接收的数据有两种,即上位机发送的 ARP 请求与 UDP 数据报。因此该模块要对在网络传输中接收到的数据进行识别。当上位机发送 ARP 请求时,根据之前介绍的 ARP 协议的格式,采用的 ARP 请求判别方法为:首先,判断目的 MAC 地址字段是否为 FF-FF-FF-FF-FF-FF;其次,再判断帧类型字段是否为 0x0806(ARP 数据报的帧类型为 0x0806),最后再判断操作字段是否为 1(在 ARP 协议中,1 表示 ARP 请求)。若全部满足,则表明上位机发送了 ARP 请求,此时产生一个高脉冲 ARP\_request,并将从 ARP 请求中提取到的上位机的 MAC 地址与 IP 地址输出到 ARP 处理模块。当上位机发送 UDP 数据报时,

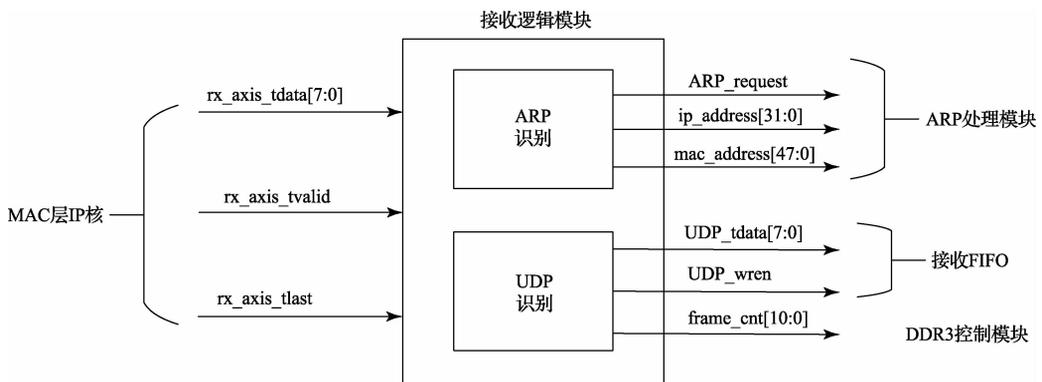


图 4 接收逻辑模块

根据之前介绍的以太网数据封装格式,得到的UDP数据报判别方法为:首先,判断IP数据报首部的协议字段填写的是否为0x11,若是,则表明发送的是UDP数据报;其次,再判断IP数据报部分的目的IP地址是否为FPGA开发板的IP地址,若是,表明发送的数据是传输给FPGA的;最后,再判断UDP首部的目的端口号是否与FPGA开发板设置的相同(本系统设置为8080),若结果一致,则将UDP数据报的数据部分提取出来并写入接收FIFO。同时,接收逻辑模块还需要将每一帧收到的UDP数据报的数据字节数frame\_cnt发送给DDR3控制模块,以便将该帧数据通过FIFO正确的写入DDR3,实现数据的缓存工作。接收逻辑模块的框图如图4所示。

### 2.5 发送逻辑模块

发送逻辑模块发送的数据有两种,即ARP应答,以及从DDR中读出并封装成基于UDP协议的MAC帧数据。发送逻辑模块框图如图5所示。整个发送逻辑模块的核

心是一个独热编码的状态机<sup>[8]</sup>,其状态转移图如图6所示。开始状态机处于空闲状态,当FPGA接收到上位机发送的ARP请求后,ARP处理模块输出的ARP\_reply使状态机进入写ARP应答帧状态,其中ARP应答需要的上位机MAC地址以及IP地址由ARP处理模块提供。在ARP应答发送完成后,状态机回到空闲状态。由于FPGA发送的数据来源于DDR3的缓存,因此当状态机检测到由DDR3用户接口模块输出的DDR\_RD\_CPL的高电平时,表明一帧的数据已经由DDR3读入到FIFO中,此时可以开始依次添加MAC帧头部、IP数据报首部与UDP数据报首部,最后将数据从FIFO中读出并添加,同时状态机回到空闲状态,等待下一次DDR\_RD\_CPL高电平。这里为了减小FPGA发送数据的帧间隔,本系统例化了两个FIFO工作在乒乓模式下,即当FPGA发送逻辑在读取并发送一个FIFO中数据的同时,DDR3将下一帧要发送的数据存入另外一个FIFO中,这样可以减小发送逻辑等待一帧发送数据读取完成的时间,提高发送效率。

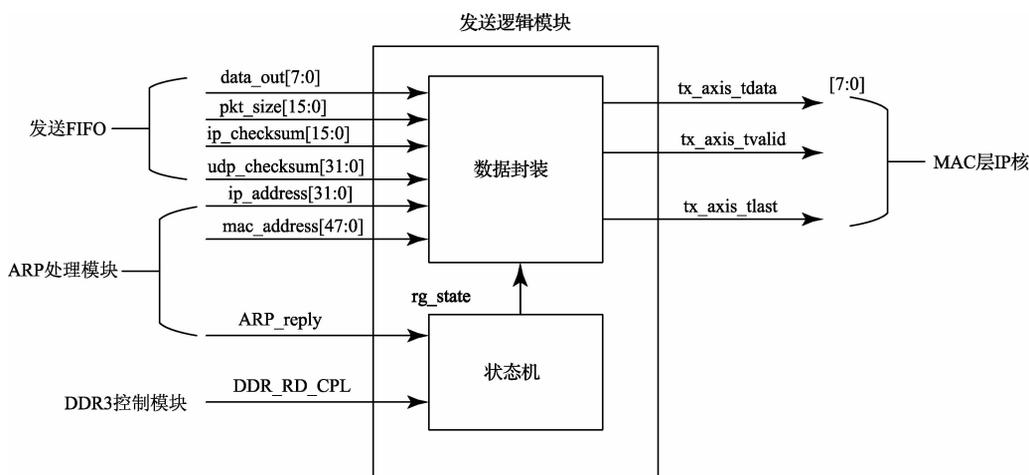


图5 发送逻辑模块

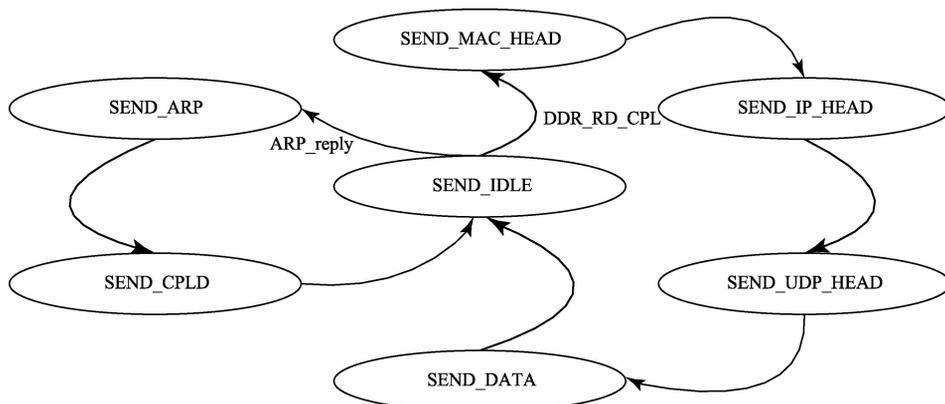


图6 发送逻辑模块的状态机状态转移

需要注意的是,IEEE802 标准规定,在全双工的工作模式下,以太网帧的数据部分长度不能超过 1 500 个字节。通过图 3 可以看出,本系统的以太网帧数据部分由 IP 首部、UDP 首部和数据组成,其中 IP 首部的长度为 20 个字节,UDP 首部的长度为 8 个字节,因此有效数据部分长度应该小于 1 472 个字节。为了保证发送的效率并方便测试,本系统选取以太网帧的有效数据部分默认长度为 1 000 个字节,即有效净荷为 1000 B。当 DDR3 中需要发送的数据多于 1000 B 时,DDR3 用户控制接口保证每次只读出 1 000 个字节,就产生一个 DDR\_RD\_CPL 高脉冲;当 DDR3 中需要发送的数据少于 1000 B 时,就将全部数据读出再产生 DDR\_RD\_CPL 高脉冲。当 FPGA 向上位机发送数据时,一般会有一帧的数据长度不足 1000 B,而其他所有帧的数据长度都是默认长度 1000 B。由于数据的长度和每一帧数据都是不固定的,需要 FPGA 在将数据由 DDR3 读出并写入 FIFO 的同时,完成数据长度 pkt\_size 的统计、IP 首部校验和 ip\_checksum 的计算以及 UDP 首部校验和 udp\_checksum 的计算工作,并将结果输入给数据封装子模块。

### 3 测试与分析

为了完成系统测试,首先需要上位机一端的 UDP 数据收发工具,这里选择使用网络调试助手<sup>[9]</sup>;其次需要网络抓包工具来具体分析数据的收发过程,这里选择 Wireshark 软件<sup>[10]</sup>;最后利用 ISE13.4 的 Chipscope Analyzer 来观察 FPGA 的内部信号波形<sup>[11]</sup>。

需要指出,系统硬件平台的 MAC 地址与 IP 地址由用户进行配置,本系统配置的 MAC 地址为 5A-01-02-03-04-

05,IP 地址为 169.254.100.100。上位机的 IP 地址为 169.254.39.65。系统测试的工作流程如下:

- 1) 上位机向 FPGA 端发送 ARP 请求,希望得到其 MAC 地址以便 UDP 数据包的发送。
- 2) FPGA 在接收到 ARP 请求后,立即向上位机发送 ARP 应答,告知上位机自己的 MAC 地址。
- 3) 上位机在接收到 ARP 请求后,就开始发送基于 UDP 协议的数据(其中目的端口号选择为 8080)。便于展示结果,本次系统测试选择发送 8 KB(即 8192 B)的累加数,FPGA 在接收到目的端口号为 8080 的数据后,便将数据写入 DDR3 中进行缓存。
- 4) 发送完成后,上位机再发送一帧基于 UDP 协议的指令,目的端口号改为 50000,此帧的作用是通知 FPGA 数据发送已经完成,可以开始将内存中的数据发送给上位机了。
- 5) FPGA 在接收到目的端口号为 50000 的指令后,便将 DDR3 中的缓存数据读出封装并发送给上位机,完成整个系统测试。

上位机与 FPGA 的 ARP 请求与应答的结果如图 7 所示,Wireshark 的结果表明 FPGA 正确地告知了上位机自己的 MAC 地址。FPGA 接收和发送数据的结果如图 8 与图 9 所示,DDR3 写入数据的计数器结果为 8192,即 8 K,与上位机发送数据大小一致;FPGA 在发送数据时一共发送了 9 帧,其中前 8 帧每一帧的有效数据长度均为 1000 B,最后一帧的有效数据长度为 192 B,即一共发送了 8 KB。在上位机端利用网络调试助手接收数据,经过对比,发送数据和接收数据是一致的。

No.	Time	Source	Destination	Protocol	length	Info
1	0.000000	90:2b:34:3f:79:87	Broadcast	ARP	42	who has 169.254.100.100? Tell 169.254.39.65
2	0.000036	5a:01:02:03:04:05	90:2b:34:3f:79:87	ARP	60	169.254.100.100 is at 5a:01:02:03:04:05

图 7 ARP 请求与应答的数据包

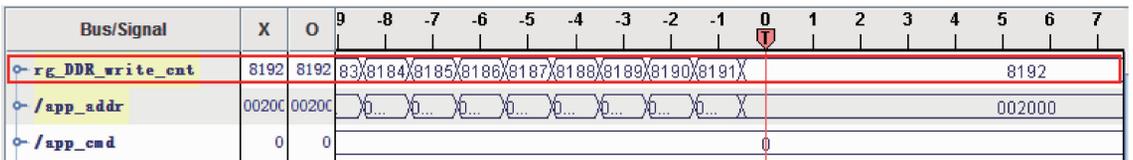


图 8 FPGA 缓存数据的计数器结果



图 9 上位机与 FPGA 端数据传输的数据包

图 10 为在 FPGA 端发送数据时,采用乒乓 FIFO 与未采用乒乓 FIFO 的帧间隔对比。可以看出,采用乒乓 FIFO 可以明显减小帧间隔。利用 Wireshark 观测到,当不使用乒乓 FIFO 的情况下系统在连续发送数据时,上位机每秒接收的数据包为 69 185 个,有效净荷为

1 000 B,有效数据吞吐量为 527.84 Mbps;而当使用乒乓 FIFO 的情况下系统在连续发送数据时,上位机每秒接收的数据包为 106 427 个,有效数据吞吐量为 812.00 Mbps。测试表明,引入乒乓 FIFO 后,极大的提升了系统的发送速率,如图 11 所示。

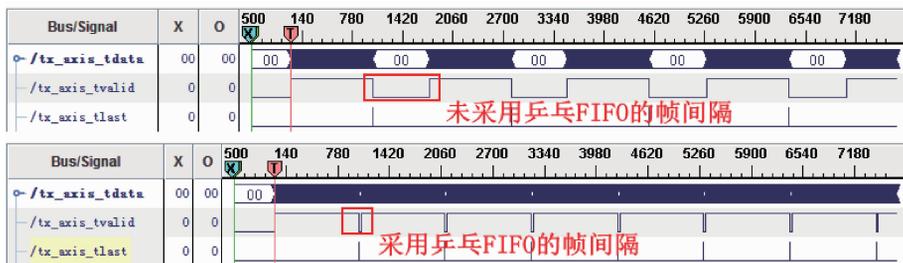


图 10 FPGA 发送数据的帧间隔对比

Description	IP	Packets	Packets/s	
Atheros L1C PCI-E Ethernet Controller	fe80::f4ab:fa5f:9f9f:2741	1246510	69185	未采用乒乓FIFO
Atheros L1C PCI-E Ethernet Controller	fe80::f4ab:fa5f:9f9f:2741	1270841	106427	采用乒乓FIFO

图 11 FPGA 发送数据的吞吐量对比

## 4 结 论

提出了利用 Xilinx 的 VC707 开发板实现基于 FPGA 的千兆以太网数据传输系统的方案。采用基于 UDP 协议的 MAC 帧格式,利用 UDP 数据报的目的端口号来识别数据和指令,并利用开发板的 DDR3 来完成数据缓存的功能。在 FPGA 发送数据时,其最大工作时钟为 125 MB/s,为了减小发送帧间隔,引入乒乓 FIFO 来提高传输效率,实际应用中系统的有效数据发送速率达到了 812.00 Mbps,充分的发挥了千兆以太网的性能。测试表明,该系统可以与上位机进行高速、准确的数据交互。

## 参考文献

- [1] 陈曦,杨亮,李鹏斐,等. 基于 FPGA 的静电测向系统[J]. 仪器仪表学报,2013,34(8):1901-1906.
- [2] 张诚,罗丰. 基于千兆以太网的高速数据传输系统设计[J]. 火控雷达技术,2011,24(1):44-46.
- [3] 王永伟,刘岩俊. 嵌入式网络控制系统设计与实现[J]. 国外电子测量技术,2014,33(9):50-53.
- [4] 崔鹤,刘云清,盛家进. 基于 FPGA 的 UDP/IP 协议栈的研究与实现[J]. 长春理工大学学报(自然科学版),2014(2):133-137.
- [5] 周静雷,贾保军,张宏艳. 基于以太网的声频数据实时

传输[J]. 电子测量技术,2011,34(5):107-110.

- [6] 李彦,李镛. 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.
- [7] 孙海超,陈春宁,田睿,等. 基于以太网的高速图像传输的研究与实现[J]. 计算机技术与发展,2014(3):187-189.
- [8] 宋飞,冯旭哲. 基于硬件协议栈芯片的高速以太网接口设计[J]. 工业仪表与自动化装置,2012(4):57-59.
- [9] 丘一凡. 基于 GPRS 无线数据传输系统的软件设计[J]. 电脑与电信,2013(11):40-41.
- [10] 罗青林,刘金刚,徐克付,等. Wireshark 环境下的网络协议解析与验证方法[J]. 计算机工程与设计,2011,32(3):770-773.
- [11] 杨贤军. 基于 ChipScope 的 EDA 实验平台的设计[J]. 通信技术,2012:101-102.

## 作者简介

李洋,1989 年出生,硕士研究生。主要研究方向为合成孔径雷达相关算法在 FPGA 平台的实现。

E-mail: liyang.sk@163.com