

数据采集系统网络传输平台设计

章涛, 王永纲, 陈俊, 都军伟

(中国科学技术大学近代物理系, 合肥 230026)

摘要: 通过以太网传输数据具有组网方便, 扩展性好, 传输距离远等优点, 因此很多地方采用。目前实现以太网传输一般是使用 CPU 运行操作系统来完成, 设计较为复杂。并且 CPU 在产生接口时序方面较为不便, 很多时候需要用 FPGA+CPU 的方式。我们设计了一个网络传输平台, 使用 FPGA 来控制硬件化的 TCP/IP 协议栈芯片来完成网络传输, 避免了由于使用 CPU 带来的一些问题。同时平台具有大容量的存储器, 剩余丰富的 IO 管脚, 适合数据采集系统中的数据传输的应用。

关键字: 数据采集、网络传输、W5300

1 背景介绍

数据采集很多时候需要将数据传输到远程计算机, 也有些采集系统有很多采集点, 比如火灾传感器, 这就需要数据传输部分能够方便地组网。在短距离内 USB 传输和 1394 火线传输[1]都是一个较为理想的方法, 但涉及到远距离传输和组网时, 它们就不是那么方便了。所以很多时候通过以太网传输采集到的数据是一种较好的方式。目前设备实现联网的做法一般是使用 CPU 运行操作系统, 由操作系统里的软件协议栈完成对 TCP/IP 协议的处理。这种方式需要编写 CPU 与数据采集部分接口的驱动和网络通讯的传输软件, 开发工作量较大, 设计较为复杂。并且 CPU 在产生接口时序方面较为不便, 很多时候需要用 FPGA+CPU 的方式: FPGA 完成与数据采集部分接口时序的产生和数据的缓存, CPU 完成数据传输[2]。因为数据采集系统中经常会使用 FPGA, 如果能用 FPGA 和少量易用的外围器件实现网络传输则是一种较为理想的方式。我们设计了一个网络传输平台, 主要包括 FPGA、DDR 芯片和硬件化的网络协议栈芯片, 可以通过以太网和计算机通讯, 将数据传输到计算机中。

2 平台结构

平台结构包括硬件上的结构和 FPGA 逻辑结构。其结构如图 1 所示。其中数据采集模块是数据采集系统的传输平台需要的一个模块, 但是目前没有, 可以根据实际需要进行开发。

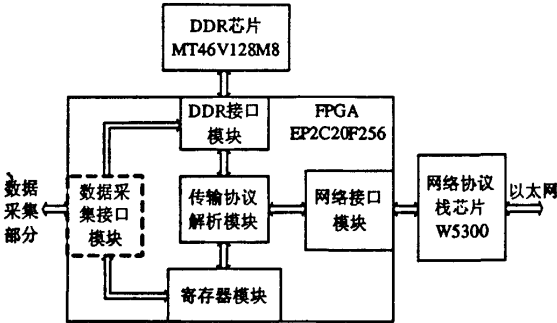


图 1 平台结构图

2.1 硬件结构

平台主要包括三个部分: FPGA, DDR 存储芯片和网络协议栈芯片。FPGA 为平台的主控芯片, 使用 Alter Cyclone II EP2C20F256, 该 FPGA 具有 18,752 个 LE, 239,616 位存储器空间, 26 个嵌入式乘法器, 4 个 PLL, 152 个用户可用管脚, 封装采用 256 脚 Fineline BGA[3], 体积较小。

DDR 存储芯片使用 MICRON MT46V128M8-6T, 容量 128MB, 数据线宽度为 8bit, 该容量是 DDR 单片芯片中容量最大的。芯片最高可工作在 167MHz。使用 DDR 对布板有较高的要求, 我们在布板时考虑到了过孔和焊盘长度对走线长度的影响, 保证了对同一组数据线和时钟的精确等长布

线。通过估计信号的回流路径进行走线和参考平面分割，以保证信号回路阻抗的连续性，减小信号之间的串扰。选择了适当的线宽、叠层方式及叠层厚度，使得单端走线的阻抗控制在 50 欧姆，差分走线阻抗在 100 欧姆，以确保传输线的阻抗匹配。这些措施保证了良好的信号的完整性，提高了时序余度，降低了对 DDR IP Core 的时间参数精确性的要求，为平台的稳定运行打下了基础。

网络协议栈芯片使用的是 WIZNET W5300，芯片内部集成了以太网物理层（PHY）、802.3 以太网 MAC 和 TCP/IP 协议栈[4]。该芯片可用在 10M/100M 以太网中，完成 ISO 七层模型中的物理层、数据链路层、网络层和数据传输层的任务，将控制器从复杂的网络协议处理中解脱出来，提供了一个易于实现的网络传输方式。芯片提供了并行接口和串行接口，可工作在直接寻址模式和间接寻址模式。因为本设计目的之一是为了提供高性能的网络传输，同时 FPGA 具有丰富的管脚资源，因此我们选取了并行接口，工作在直接寻址模式。

2.2 FPGA 逻辑结构

在进行 FPGA 逻辑编写时采用了模块化设计，结构主要包括四个部分：DDR 接口模块、网络接口模块、传输协议解析模块和寄存器模块。DDR 接口模块对 DDR 芯片进行上电初始化，产生 DDR 的读写时序并控制 DDR 刷新。网络接口模块对网络协议栈芯片进行上电初始化，产生芯片的读写时序，并控制芯片发送和接受数据。传输协议解析模块对网络接口模块送来的数据进行解析，转换成对 DDR 或寄存器的读写操作，并将 DDR 模块或寄存器模块读出的数据送给网络接口模块，通过网口发送出去。寄存器模块包括 16 个 16 位的寄存器，存放参数设定和状态信息，远程用户通过读写寄存器实现对数据采集的控制。

在进行 FPGA 逻辑编写时采用了层次化的设计，每层完成其特定工作，使得设计清晰明了。系统的层次结构如图 2 所示

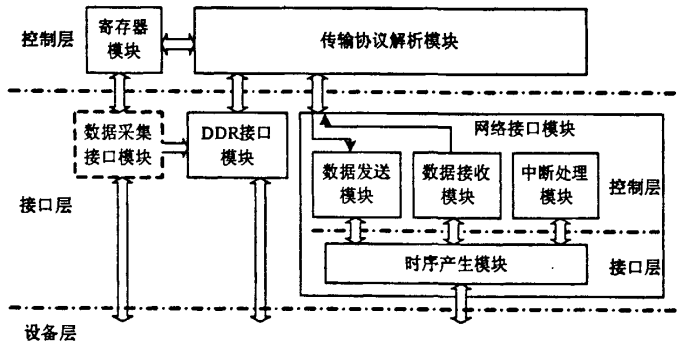


图 2 平台层次图

从整体上来看，平台结构分为设备层、接口层和控制层。设备层为 FPGA 外围芯片，接口层作为设备层和控制层的桥梁，面向设备发送相应指令，产生特定时序的信号；面向控制层提供数据。控制层将接口层送来的数据进行解析，转化成对接口层的操作，控制整个平台的工作。在网络接口模块中，同样也可分为接口层和控制层。接口层将上层所有对 W5300 访问的操作按照其时序要求产生相应地信号，控制层则控制 W5300 芯片完成以太网通信。

2.2.1 DDR 接口模块

DDR 接口使用 Altera 提供的 IP core。该 IP core 包括控制逻辑和数据通道两部分，如图 3 所示。控制逻辑面向 DDR 芯片执行总线的操作，对内提供一个简单的接口。数据通道在内部读写数据总线与 DDR 芯片的双向数据总线之间提供一个接口，因为 DDR 芯片的数据总线是上升下降沿都传送数据，内部总线仅在上升沿传送数据。所以内部数据总线宽度是 DDR 芯片数据总线的两倍。本设计中内部总线的宽度为 16 位。DDR 突发数据传输长度选择 2。

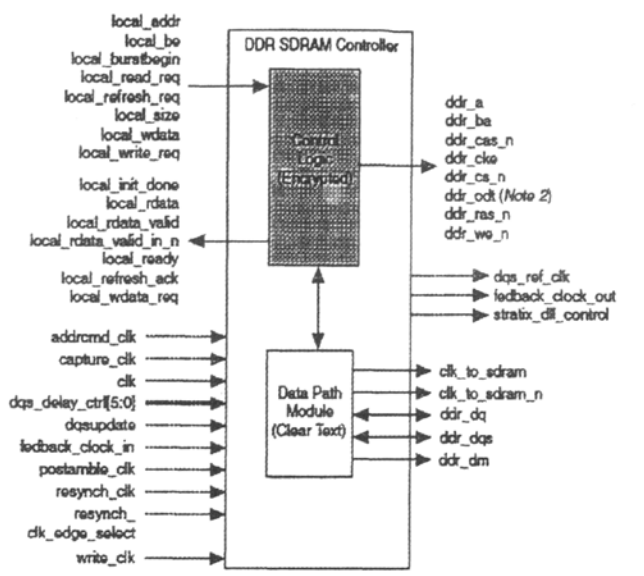


图 3 DDR 控制器结构图

FPGA 逻辑中还包括一个随机数测试模块，可以产生伪随机数写入 DDR 芯片，然后将其读出，通过比较两者是否相同，可以判断芯片的好坏。随机数测试在上电时会执行一次，执行结果由板上 LED 灯指示。

2.2.2 网络接口模块

网络接口模块分为控制层和接口层。控制层由多个处理模块组成，包括：复位模块、初始化模块、数据接收模块、数据发送模块和中断处理模块，每个模块完成一种情况的处理。这些模块受一个状态机的控制，状态机根据不同的输入分别执行相应的模块。状态机的跳转如图 4 所示：

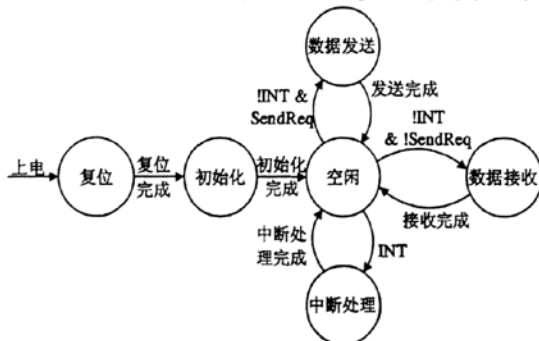


图 4 网络接口模块状态机状态跳转图

系统上电开始，状态机进入复位 W5300 状态，在其复位管脚上产生一个 2 微秒的负脉冲以复位芯片，复位完成后进入初始化阶段。初始化阶段设置芯片工作方式、平台网络接口的 MAC 地址、源 IP 地址、源端口号、掩码、网关和协议类型等。本系统采用的是直接寻址方式，协议类型选择 UDP。初始化完成后状态机进入空闲状态。在这个状态下，先检测是否有中断(INT)产生，如果有中断则进入中断处理状态，执行中断处理模块，如果没有中断则检测是否有发送数据请求(SendReq)，如果有则进入数据发送状态，执行数据发送模块，如果既没有中断也没有发送数据请求，则进入数据接收状态，执行数据接收模块。数据接收模块执行时，首先要读取 W5300 内部的接收数据字节长度寄存器，如果不为 0 表明 W5300 接收到数据，则读取数据，否则直接结束数据接收过程，状态机返回到空闲状态。

网络接口模块的接口层包括时序产生模块，它将控制层对 W5300 的访问转换成其所需的时序。

2.2.3 协议解析模块

网络接口模块从 W5300 中读取到数据后，将数据发送到协议解析模块。协议解析模块根据这些数据解析出相应的操作，读写寄存器或 DDR 存储器。因为数据写入寄存器只需要一个时钟周期，所以写入寄存器的数据可以被即时写入。而 DDR 有突发数据传输长度的限制和刷新的存在，使得数据无法即时写入，因此写入 DDR 的数据先缓存在一个写入 FIFO 中，当该 FIFO 中的数据大于突发数据传输长度时，向 DDR 接口模块请求一次数据写入。读取数据时，因为读寄存器和 DDR 的速度远大于网络的传输速度，因此读取的数据需要存放在读出 FIFO 中。读取 DDR 时，采用了预读方式，持续读取 DDR 直到读出 FIFO 满，但实际发送的数据个数仍然为实际需要的数据量，这种方式降低了对 DDR 读取的复杂度。

3 传输协议设计



图 5 传输协议

我们定义了一个传输协议如图 5 所示，协议包括 16 位命令、32 位地址、32 位长度和可选的数据部分。命令表示本次操作要完成的任务，目前定义了四个，分别是：CMD_RD_REG（读寄存器）、CMD_WD_REG（写寄存器）、CMD_RD_MEM（读 DDR 存储器）、CMD_WD_MEM（写 DDR 存储器）。地址表示本次操作的起始位置，长度表示本次操作的数据长度。数据是可选部分，当命令字段为 CMD_WD_REG 或 CMD_WD_MEM 时，表明本次为写操作，此时有数据部分，长度为长度字段值，平台将把该数据写入到寄存器或 DDR 存储器指定的地址位置；当命令字段为 CMD_RD_REG 或 CMD_RD_MEM 时，表明本次为读操作，此时没有数据部分，平台将从寄存器或 DDR 存储器指定的地址读取指定的长度数据发送至远程计算机。

以一个常见的数据采集过程说明使用本协议是如何工作的。远程软件以本平台 IP 地址和端口号为源 IP 地址和源端口号向本平台发送一个写寄存器命令，将寄存器模块中控制寄存器的开始工作位写 1，该位为 1 则触发数据采集接口模块开始工作，其首先要对 DDR 存储器清零，然后等待外部数据的到来，将数据存储在 DDR 存储器中，存储完毕后将寄存器模块中表示状态的寄存器的数据记录完成指示位置 1。远程软件发出开始工作指令后应一直发送读取寄存器命令，读取该位直到其为 1。然后发送读 DDR 存储器命令，地址为当前待读取数据的地址，长度选取合适的数值，从 DDR 中读取数据。读取一段数据后将地址加上长度值读取下一段数据，直到数据读取完成。

从上可以看到远程软件需要一直读取状态寄存器以确定平台工作状态，占用较多的计算机 CPU 时间，同时占用了网络带宽，在多个本平台的存在下，这种情况更加明显。这在多用户的以太网环境里是不允许的。因此我们的平台采用了消息机制，通过这种机制可以避免以上问题。

我们将寄存器模块的第一个寄存器作为消息寄存器，当该寄存器非零时，协议解析模块将该寄存器的值发送到远程软件，发送频率每秒一次，直到该寄存器被清零。在以上举例的数据传输过程中，远程软件发出开始工作指令后即可进入等待状态，数据采集接口模块在数据记录完成后将消息寄存器某位置 1，平台然后开始发送消息寄存器值。软件接收到该寄存器值后，先发送写寄存器命令将该寄存器清零，然后根据消息值进行相应的操作。消息寄存器为 16 位，因此最多可以定义 16 种消息。

4 测试结果

我们将一台计算机和平台通过交叉线连接起来，设定好双方的 IP 地址后，在计算机上通过软件向平台上的 DDR 读写数据来测试平台的数据传输速度。写入时使用的协议长度字段为 1462，读取时为 1472，这样发送的数据长度或读取的数据长度加上 UDP 首部的长度为 1500，是以太网的最大传输单元(MTU)的长度。在这种情况下测得平台接收数据速率在 70Mbps，发送数据速率达到了 30Mbps。

5 结论

我们设计了一个网络传输平台，使用 FPGA 控制硬件化的 TCP/IP 协议栈芯片来完成网络传输，

避免了由于使用 CPU 带来的一些问题，同时平台具有大容量的存储器，剩余丰富的 IO 管脚，在平台基础上再进行少量的开发即可实际使用。我们采用了模块化的设计和层次化的结构，使得平台结构清晰。设计了自定义的传输协议，针对实际环境进行了优化。经测试平台达到了较高的传输速度，其接收数据速率达到了 70Mbps，发送数据速率达到了 30Mbps。本平台具有设计简洁、灵活可靠、性能突出等优点，适合数据采集系统中的数据传输的应用。

参考文献：

1. 郑裕峰，王永纲，红外焦平面阵列测试的任意数据发生器，核电子学与探测技术，2007、27（4）、697-700
2. 程敬原，宋克柱，王砚方，杨俊峰，陆增援，时移地震数据采集和记录系统中数据汇聚和传输子系统的设计，应用基础与工程科学学报，2006、14（2）、181-193
3. Cyclone II Device Handbook，2008、Volume 1、Page 1-4
4. High-Performance Internet Connectivity Solution - W5300，2008、Page 5

论文作者简介：

章涛，27，博士生，主要从事网络安全方面研究，对嵌入式系统和高速电路设计具有较多的经验。

王永纲，44，教授/博师，主要从事核医学成像、网络安全和高速电子学研究，。

陈俊，27，本科生，主要从事高速电子学设计。

都军伟，26，博士生，主要从事核医学成像数据获取与处理。

A Network Transmission Platform for Data Acquisition System

Zhang Tao, Wang Yonggang, Chen Jun, Du Junwei

(Department of Modern Physics, USTC, Hefei 230026, China)

Abstract: Ethernet is widely used for data transmitting in the data acquisition system for it's excellent characters: easy to use, long distance and convenient to create a network. Currently the most popular method to offer a network link is to use CPU to run OS, which needs to write the driver for the interface between the CPU and the data acquisition system. That's a bit more complicated. Besides CPU is week in generating time diagram thus forcing us to use CPU and FPGA together. We design a network transmission platform which avoids using CPU by using FPGA to control a hardware TCP/IP stack chip. The platform also has large memory size and rich IO resources which make it suitable for data transmitting in data acquisition system.

Keywords: Data acquisition, Network transmission, W5300

数据采集系统网络传输平台设计

作者: 章涛, 王永纲, 陈俊, 都军伟

作者单位: 中国科学技术大学近代物理系, 合肥 230026

本文链接: http://d.g.wanfangdata.com.cn/Conference_7226471.aspx