

基于嵌入式网络协议栈的多轴运动控制卡通信系统设计

杨亮亮^{1,2}, 黄 坤¹, 张莉敏³, 史伟民¹

(1.浙江理工大学机械与自动控制学院, 浙江 杭州 310018;

2.杭州汇萃智能科技有限公司, 浙江 杭州 311121;

3.中南财经政法大学, 湖北 武汉 430073)

✉ yangliangliang@zstu.edu.cn; 2522823811@qq.com; zhanglimin1979@163.com; swm@zstu.edu.cn



摘 要: 针对现代工业生产中运动控制卡高速化、高精化的技术要求, 设计了一种基于小型嵌入式网络协议栈LwIP的多轴运动控制卡网络通信系统。系统以STM32H7微控制器为主控芯片, 以低功耗芯片LAN8720A作为外部以太网PHY层芯片, 采用TCP/IP网络协议栈中的UDP协议作为上下位机间通信协议。通信系统的下位机程序采用基于LwIP协议栈的RAW API方式编写, 上位机程序基于SOCKET套接字接口编写。测试结果表明: 系统能达到既定要求的通信响应速度, 可实现单台工控上位机与多块运动控制卡之间“一对多”的主从式高速通信, 具有较高的工业实用价值。

关键词: 嵌入式; LwIP; UDP; 网络通信; 多轴运动控制卡

中图分类号: TP23 **文献标识码:** A

Design of Communication System for Multi-axis Motion Control Card based on Embedded Network Protocol Stack

YANG Liangliang^{1,2}, HUANG Kun¹, ZHANG Limin³, SHI Weimin¹

(1. Faculty of Mechanical Engineering & Automation, Zhejiang Sci-Tech University, Hangzhou 310018, China;

2. Mstar Technologies, Inc., Hangzhou 311121, China;

3. Zhongnan University of Economics and Law, Wuhan 430073, China)

✉ yangliangliang@zstu.edu.cn; 2522823811@qq.com; zhanglimin1979@163.com; swm@zstu.edu.cn

Abstract: This paper proposes to design a multi-axis motion control card network communication system based on the small-scale embedded network protocol stack LwIP (Light Weight Internet Protocol), in order to meet the high-speed and high-precision technical requirements of motion control cards in modern industrial production. The system uses STM32H7 microcontroller as main control chip, low-power chip LAN8720A as external Ethernet PHY (Physical Layer) chip, and UDP (User Datagram Protocol) in TCP (Transmission Control Protocol)/IP (Internet Protocol) network protocol stack as communication protocol between upper and lower computers. Lower computer program of the communication system is written in RAW API method based on the LwIP protocol stack, and upper computer program is written based on the SOCKET interface. Test results show that the proposed system can reach the required communication response speed. It also realizes the one-to-many master-slave high-speed communication between a single industrial control host computer and multiple motion control cards. The proposed system has high industrial practical value.

Keywords: embedded; LwIP; UDP; network communication; multi-axis motion control card

1 引言(Introduction)

随着现代工业信息技术的发展, 国内中低端运动控制领域对控制的响应速度及精度的要求也越来越高。传统的运动控制卡采用的是DSP+FPGA的硬件架构, 如国内的固高科技GT系列、研华集中式运动控制卡系列等。随着工况多样化与外设丰富性要求的不断提高, DSP虽运算能力强大, 但在外

设控制、事务处理、信息管理等方面不如高性能ARM灵活便捷、性价比高。另外, 传统的工控上位机与运动控制卡之间通常采用RS-232/485串口或PCI并行总线通信, 其中串口方式通信速率较低, 传输距离短, 受干扰影响大^[1]; PCI总线方式不够灵活便捷, 成本较高, 只能嵌入在工控机上使用。针对上述问题, 我们设计了一种基于ARM与LwIP协议栈的多轴

运动控制卡以太网通信系统。在STM32主控芯片上应用轻量型LwIP协议栈,采用LwIP RAW API编程接口搭建运动控制卡的网络通信框架,SOCKET套接字方式编写上位机UDP服务端程序,并针对UDP协议不可靠传输的特性,设计了自定义协议头及对应的数据收发处理状态机对数据帧进行封装与解封处理,实现一台工控上位机控制多块运动控制卡的“一对多”网络通信系统。

2 系统架构(System architecture)

本设计的系统总体框架如图1所示,将工控上位机与多块运动控制卡通过网线连入同一路由器或交换机下的局域网内实现“一对多”通信的运动控制系统。

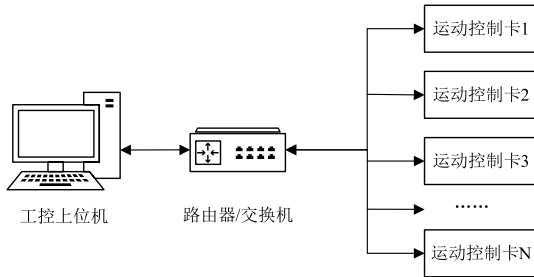


图1 控制系统总体框架

Fig.1 Overall framework of control system

3 通信系统硬件设计(Hardware design of communication system)

运动控制卡上的主控芯片采用基于ARM Cortex M7内核的32位高级MCU—STM32H743IIT6,该芯片集成了大量的外设接口及通信接口,且其稳定运行主频高达400 MHz,能满足嵌入式运动控制系统的各项性能要求。芯片上的介质访问控制器MAC通过符合IEEE 802.3标准的精简介质接口RMII与外部PHY芯片LAN8720A相连接^[2]。LAN8720A再与带有网络隔离变压器的RJ45网络接口模块相连接,利用网络线缆实现板卡与工控上位机的网络连接。

本设计采用的LAN8720A芯片是一款体积小(24pin QFN封装,4 mm×4 mm×0.85 mm)、功耗低的全能型10/100 Mbps以太网物理层收发器,符合IEEE 802.3/802.3u(快速以太网)标准,支持精简介质RMII接口,能大幅减少所需引脚数。

本设计采用的RJ45网络接头型号为HR911105A,该接头带有网络屏蔽功能,能有效地减少外部干扰造成的影响,同时内部带有变压器^[3]。网络通信模块的接线示意图如图2所示。

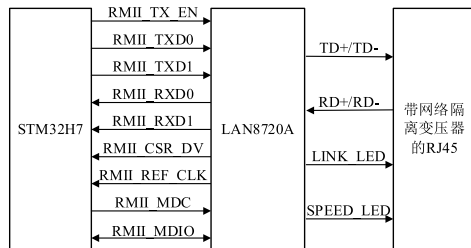


图2 通信模块接线图

Fig.2 Communication module wiring diagram

4 通信系统软件设计(Software design of communication system)

4.1 系统软件框架

系统软件框架如图3所示,工控上位机端通过创建SOCKET套接字编写UDP服务端程序接入路由器,运动控制卡端通过应用轻量型LwIP协议栈编写相对应的UDP程序来与

外部PHY芯片LAN8720A进行数据交互。LwIP协议栈是瑞典计算机科学院(SICS)的Adam Dunkels等开发的一个小型开源的TCP/IP协议栈,其在保持TCP/IP协议栈主要功能的基础上减少了对RAM的占用,适用于小型嵌入式系统^[4]。本设计中下位机程序采用基于无操作系统的LwIP RAW API编程接口方式设计UDP客户端程序。

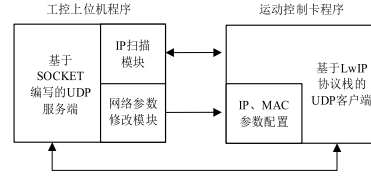


图3 系统软件框架

Fig.3 System software framework

4.2 工控上位机通信程序

上位机程序采用VS2012作为开发环境,结合C与C++编程搭建基本框架并实现MFC人机交互界面,通信部分采用SOCKET套接字接口编写UDP服务端。上位机程序在网络通信方面需要实现的功能为:

(1)扫描工控上位机所连入的局域网,实现对该网内运动控制卡的检测、连接、通信

检测板卡设备IP的方式采用TCP/IP协议中的Ping命令。Ping服务命令位于TCP/IP网络体系结构中的应用层,其主要作用是向特定的目的主机发送ICMP(Internet Control Message Protocol,因特网报文控制协议)回显请求报文,并得到目的主机的回显应答报文。通过Ping命令可以确定两台主机之间是否成功交换数据包,再根据返回的信息就可以确定目的主机与本地主机间网络通信链路是否通畅等信息。

网络通信中第一部分功能实现的工作原理为上位机程序检测主机所联入局域网内的所有设备(同一局域网内每个设备的IP地址与MAC物理地址要求是唯一的),并将所有检测到的设备IP号组成列表列举在MFC交互界面中,通过点选MFC界面IP列表框进行点对点连接通信。另外,上位机程序采用多线程并发执行机制实现同时控制多块运动控制卡的功能,即“一对多”通信。获取设备IP列表的过程如图4所示。

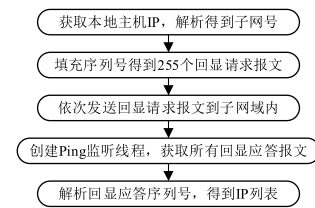


图4 Ping扫描得到IP列表

Fig.4 Ping scan to get the IP list

(2)工控上位机下发设定的IP、MAC等参数,实现在线修改运动控制卡网络参数的功能

网络通信中第二部分功能的实现方式是通过在上位机中使用MFC微软基础类库编写人机界面,使用者按照规定设置所需的IP、MAC等网络参数,上位机程序将该网络参数打包成网络数据帧发到指定IP地址的运动控制卡内。运动控制卡在接收到该网络数据帧后进行相应的命令处理,并通过I2C总线将其写入与板卡外部连接的掉电数据不丢失的EEPROM芯片(AT24C02)内,运动控制卡每次上电时都会通过I2C接口读取AT24C02内最新一次写入的IP、MAC等网络参数并初始化网络配置。此功能可以将给定的运动控制卡设置为指定的网络参数,方便管理局域网内所连入的多块多轴运动控制卡。

该部分功能所对应的上位机软件界面如图5所示。

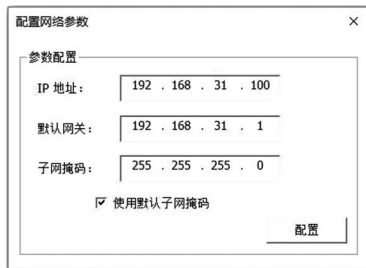


图5 配置网络参数模块

Fig.5 Module for configuring network parameters

4.3 运动控制卡LwIP通信框架

在LwIP网络编程中主要有三种接口编程方式：NETCONN API接口编程、SOCKET API接口编程、RAW API接口编程。本设计采用第三种方式RAW API接口编程，该方式的核心是对TCP/UDP控制块的处理，优点是数据处理效率高，可以基于无操作系统编程^[5]。

LwIP RAW API编程方式是基于回调机制的，即为协议栈内核中不同事件注册相应的回调函数后，当事件发生的时候相应的回调函数就会被调用。基于LwIP RAW API编程接口实现的下位机程序框架如图6所示。其分为三部分：LwIP协议栈的初始化配置、网络数据接收检测及命令处理、数据重组打包发送。

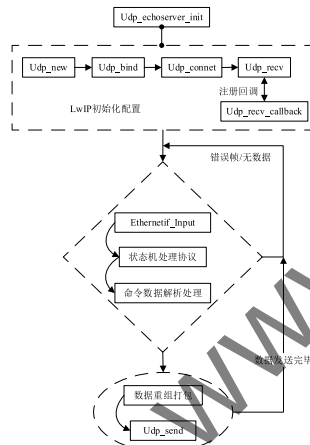


图6 LwIP RAW API接口程序框架

Fig.6 LwIP RAW API interface program framework

下位机程序的网络通信整体是基于LwIP协议栈的，下面为对程序通信框架的总体概述。

(1) 基于LwIP协议栈框架对网络进行初始化配置

Udp_echo_server_init函数用于初始化，该函数调用下面五个函数建立UDP连接并且注册回调函数：

Udp_new函数用于创建UDP控制块；Udp_bind函数用于绑定本机IP地址与端口号；Udp_connect函数用于设置控制块中的远端IP地址与端口号，并将UDP控制块插入控制块链表udp_pcbs中；Udp_rcv函数用于设置UDP控制块中的recv与recv_arg字段，recv是一个函数指针，指向一个udp_rcv_fn类型的回调函数，它是内核与应用程序交互的桥梁，当内核接收到数据时，就会调用这个回调函数，进而将数据递交到应用层处理；Udp_rcv_callback函数是数据接收回调函数，用于将以太网接收到的数据拷贝到设定的全局数据缓冲区以便于后续处理。

(2) 网络数据接收检测及命令处理阶段

程序主体部分采用非阻塞轮询的方式将数据包存放到DMA描述符中，然后拷贝其中的数据到pbuf^[6]结构体中用于后续处理，即在无网络数据接收时会一直处于一个初始轮询状态。在该段程序体内，Ethernetif_input作为数据处理的入口，它在接收到以太网帧时被调用，用于从以太网数据存储区读取已接收到的一个以太网数据帧，并将其送入指定的网卡结构体netif中。

本程序网络通信部分采用LwIP协议栈中的UDP协议。UDP协议是TCP/IP协议栈中的传输层协议，是一个简单的面向数据报的协议，不提供对数据包的分片、组装功能，当报文发送出去后无法知道是否安全、完整到达，所以UDP协议也可称作“无连接的连接协议”。UDP除了这些缺点外也有它自身的优势，由于UDP不属于连接型协议，因而消耗资源少，实时性高，处理速度快^[7]。针对UDP的非连接型协议属性，在程序内制定了一种数据收发处理协议，数据发送前在数据头部封装自定义的一段协议头数据，在数据接收过程中则对其进行逐步解封提取。通过该处理协议来实现对错误帧、非正常帧的处理，以提高UDP协议的数据传输可靠性。数据收发协议头格式如图7所示。

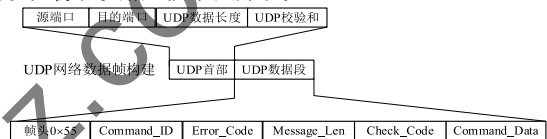


图7 自定义协议头部格式

Fig.7 Custom protocol header format

根据上述处理要求，数据收发部分编写了一个数据收发状态机用于封装与解封网络数据包，并判断接收到的网络数据包是否正确。若数据正确则会按照状态机跃迁顺序重组数据，最后进入switch-case命令分支解析并做出相应的处理；若数据有误或数据包掉帧导致数据不完整，则会在跃迁到相应的状态机后判断处理是否超时进而跳出状态机并在丢弃该部分数据后回复上位机数据接收异常，同时其回到初始轮询状态继续等待接收网络数据包。此处的超时管理机制作用原理为：当数据被接收到并开始进入状态机提取处理时，启动一个计时器机制，在每一个状态机处都会进行计时，计算数据处理时长，若超过设定时间则认为出错，跳出状态机，向上位机报告错误；若未超过设定时间，则认为无错误，处理完成后状态机正常跳转。数据接收处理的状态机跃迁图如图8所示。

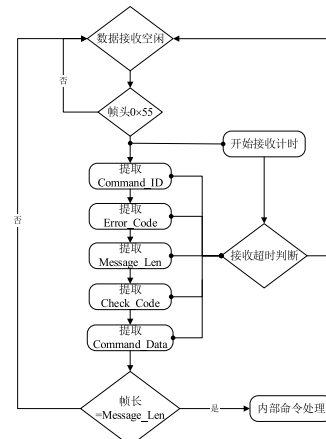


图8 数据处理状态机跃迁图

Fig.8 State machine transition diagram

(3)数据重组打包发送阶段

在数据发送处理中,对于TCP连接,内核会根据用户提供的待发送数据的起始数据和长度,自动将数据封装在合适的数据包中,然后放入发送队列;对于UDP,用户需要手动将数据封装在网络发送buffer中,通过调用发送函数,内核直接发送数据包的数据段^[8]。UDP数据包的最大理论长度为65,535字节,但是UDP不存在数据分片和组装机,只会将数据包直接向下交付给IP层处理,所以它的长度大小受到下层IP层和数据链路层协议的制约。而以太网规定的链路层最大传输单元MTU为1,500字节,去除UDP包头8字节、IP包头20字节,所以数据报文单帧最大长度可采用1,472字节。

运动控制卡在接收到正确的数据包并通过数据接收状态机提取处理后,通过判断命令ID号进入相应的命令ID处理函数,将对应的回复数据通过数据发送状态机重组,若数据长度大于1,472字节,则将数据包分组,每组大小1,472字节,依次将分组数据发出;若小于1,472字节则直接单组发出。

5 实验验证(Experimental verification)

完成控制系统的通信部分程序编写后,在上位机软件端调用一个通信回环测试函数来测试系统性能。通信测试原理是通信回环测试函数在上位机中生成一个指定长度的随机浮点数(0—9)数组,并在该数组前添加自定义的协议头部,然后通过UDP协议下发到指定IP板卡。运动控制卡接收到网络数据包后进行校验处理,再将处理过的数据包返回给上位机,最后上位机对收发数据进行对比,确定通信是否正确,且通信响应时间要求小于2 ms。

本设计中采用的测试方式:将上位机与三块独立的四轴运动控制卡通过网线连入同一个路由器或交换机下,上位机扫描该局域网内的所有运动控制卡并进行连接,然后分别设置三块运动控制卡网络参数。其中,路由器的IP地址为192.168.1.1,PC上位机的IP地址设置为192.168.31.100,三块运动控制卡的IP地址分别设置为192.168.31.10、192.168.31.20、192.168.31.30。在设置好板卡网络参数后重启运动控制卡,然后同时开启三块板卡通信测试,测试结果如表1所示。

表1 通信性能测试

Tab.1 Communication performance test

发送次数	发送字节数 (Byte/次)	运动控制卡IP	通路时长(ms)	板卡平均通信 速度(Mbit/s)
1	800	192.168.31.10	1.0882	11.7625
		192.168.31.20	1.1135	11.4953
		192.168.31.30	1.0838	11.8102
5	800	192.168.31.10	5.4869	11.6641
		192.168.31.20	5.3965	11.8595
		192.168.31.30	5.4133	11.8227
50	800	192.168.31.10	53.7215	11.9132
		192.168.31.20	54.1341	11.8224
		192.168.31.30	54.9866	11.6391
500	800	192.168.31.10	541.1649	11.8263
		192.168.31.20	550.3129	11.6297
		192.168.31.30	545.1127	11.7406
1000	800	192.168.31.10	1,089.7622	11.7456
		192.168.31.20	1,107.2477	11.5601
		192.168.31.30	1,098.6332	11.6508

通过表1的测试结果可以看出,在“一对多”的通信环境下,工控上位机与连接的每块运动控制卡之间保持了稳定的通信速度——11.7 Mbit/s,通信的平均响应时间为1.09 ms,而且各运动控制卡能相互独立地与工控上位机完成数据传输并确保数据正确性。相较于串口RS232通信传输速度100—200 kbit/s,DSP+专用芯片(W5100S)的以太网传输速度为1.2 Mbit/s^[9],通信测试结果表明,所设计的通信系统能达到要求的精度与响应速度,且在通信速度上有大幅提高。

6 结论(Conclusion)

本文设计并实现了一种基于ARM和LwIP的嵌入式多轴运动控制卡网络通信方案,描述了网络通信系统的硬件设计、软件设计两部分。系统中工控上位机端采用VS2012编写MFC界面方便人机交互,多轴运动控制卡端采用无操作系统的LwIP RAW API编程接口编写适合嵌入式系统的程序。针对UDP连接不安全、不可靠的问题,采用数据收发状态机及处理超时管理机制有效确保数据的正确收发和系统的实时响应性。最后通过实验测试了网络通信性能,实验测试效果良好,可满足实际工业场景的应用要求。本设计方案对嵌入式运动控制系统的以太网通信系统设计具有一定的实际参考意义。

参考文献(References)

- [1] 盛宇波,张成,周澜.一种基于STM32与LWIP的数据采集系统.中国,CN201911218936.0[P]. 2020-02-21.
- [2] 马亚辉,吴凡,李林,等.基于LwIP的热插拔网络嵌入式设计方法[J].单片机与嵌入式系统应用,2020,20(04):10-13,17.
- [3] 王全全,冯博,朱辉.基于STM32和LWIP的井场传感器远程监控系统[J].工业控制计算机,2019,32(04):13-14.
- [4] 武磊,张正炳,胡蓉华.基于Web的家居设备远程控制系统设计与实现[J].微型机与应用,2017,36(19):66-69.
- [5] 张超,王志超,林岩.基于LwIP协议栈的嵌入式网络控制系统设计[J].单片机与嵌入式系统应用,2019,19(02):34-36,40.
- [6] DUNKELS A. Design and implementation of the LwIP TCP/IP stack[D]. Stockholm, Sweden: Swedish Institute of Computer Science, 2001:3-5.
- [7] 朱升林.嵌入式网络那些事:LwIP协议深度剖析与实战演练[M].北京:中国水利水电出版社,2012:320.
- [8] 赵成青,李宥谋,刘永斌,等.LWIP中零拷贝技术的研究与应用[J].计算机技术与发展,2018,28(07):182-186.
- [9] 杨亮亮,李翔,史伟氏,等.基于以太网的运动控制卡通信模块设计[J].工业仪表与自动化装置,2020(03):24-28,59.

作者简介:

杨亮亮(1978—),男,博士,副教授.研究领域:高速高精运动控制。

黄 坤(1993—),男,硕士生.研究领域:嵌入式方向。

张莉敏(1979—),女,博士,副教授.研究领域:逻辑与人工智能。

史伟氏(1965—),男,博士,教授.研究领域:纺织机械自动控制技术。