

· 电路与控制 ·

光电伺服系统的多串口通讯软件设计

夏 妍, 牛 啸

(东北电子技术研究所, 辽宁 锦州 121000)

摘要:对光电伺服控制系统通讯设计中的硬件选择和软件编程进行了分析和优化。硬件上采用了16C554异步通讯芯片,减少了CPU的中断数量,选用RS422通讯方式,提高了串口效率,实现了伺服控制系统的实时控制。分析了控制软件的功能和设计流程,提出了软件设计中出现的问题,并给出了解决方案。针对通讯设计过程中出现的数据查询等待耗时过长、数据不能实时处理等问题,阐述了多串口高速率数据通讯的软件编程方法,解决了循环等待问题,实现了四路串口高速率数据通讯。

关键词: 串口通讯; 软件编程; 异步通信

中图分类号: TP273+.5

文献标识码: A

文章编号: 1673-1255(2012)06-0065-05

Design of Multi-serial Communication Software for Servo Control System

XIA Yan, NIU Xiao

(Northeast Research Institute of Electronics Technology, Jin zhou 121000, China)

Abstract: Hardware selection and software programming in the communication design of electro-optic servo control system are analyzed and optimized. 16C554 asynchronous communication chips are used in hardware and the interruptions of CPU are reduced. RS422 communication mode is chosen to improve efficiency of serial ports and realize real-time control of servo control system. The functions and design processes of control software are analyzed. The problems in software design are put forward and relative solutions are given. According to the issues such as too long time consuming of data query and non-real time data process during communication design process, software programming methods of multi-serial port high rate data communication are introduced. The problem of circular wait is solved and high rate data communication of four road serial ports is realized.

Key words: serial port communication; software programming; asynchronism communication

随着计算机CPU处理能力的提高,伺服系统的控制逐渐由数字控制取代模拟控制。文中所述的伺服转台控制系统采用数字控制方法。伺服控制器采用PC/104结构,选用500 MHz主频的CPU,主板上采用16C554扩展了四路高性能通讯串口,最高波特率达460.8 Kb/s。16C554是TI公司生产的4通道异步收发器集成芯片,在FIFO模式下,传输和接收前将数据缓冲为16字节数据包,减少了CPU的中断数量。内部包含4片改良的16C550异步传输器件,使得串行I/O更加可靠。

由于大部分的PC/104软件的设计都是基于DOS

平台,而DOS操作系统是单任务操作系统,而实时多任务是控制的基本要求,因此在DOS操作系统中完成多任务是要解决的主要技术问题^[1]。

1 多串口通讯硬件工作方式的选择

串口通信方式包括RS422、RS485、RS232等多种形式。为了提高串口效率,串口硬件采用四路光电隔离异步串行口,采用RS422通讯方式,RS422通过两对双绞线传输数据可以全双工工作,而RS485通过一对双绞线只能半双工工作,收发不能同时进

行。由表1所示,可以看出RS422/RS485相对RS232来说,驱动电平明显下降,这样不易于损坏接口电路芯片;传输速率的大大提高,有利于高更新率,大数

据的传输;差分工作模式,抗干扰能力提高;传输距离提高了几十倍^[5]。

表1 RS-232、RS422、RS485 电气参数对比

规定	RS232	RS422	RS485
工作方式	单端	差分	差分
节点数	1收1发	1发10收	1发32收
最大传输电缆长度/英尺	50	4 000	4 000
最大传输速率	20 Kb/s	10 Mb/s	10 Mb/s
最大驱动输出电压/V	+/-25	-0.25~+6	-7~+12

2 软件设计

2.1 控制软件的功能

转台伺服控制系统需要实现的功能主要包括:CPU定时中断、数模模数功能调试、通讯功能调试、测角信号采集、手柄操控、电限位功能调试。控制系统功能框图如图1所示^[2]。

在这5项功能模块中,CPU 1ms定时中断是控制的主要部分,完成三轴伺服稳定平台的陀螺模拟速度环,数字速度环,数字位置环的校正算法,输出控制电压,驱动电机电作,实现三轴稳定。系统中通讯功能模块的四个串口设置和功能见表2。

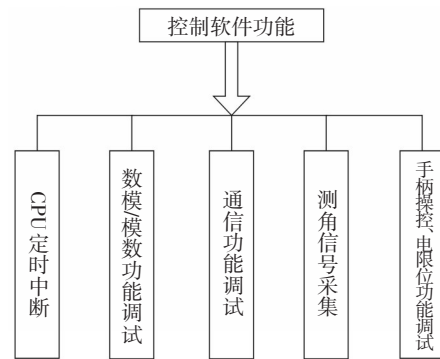


图1 控制系统功能框图

表2 通讯功能模块设计

基地址	波特率	数据包长/字节	发送更新率/Hz	接收更新率/Hz	传输方式	功能
0x310	115 200	收:10 发:47	100	800	RS422	得到设备三轴角速率 发送模拟设备信息
0x318	115 200	不定长 收:约20 发:约20	100	1以下	RS422	与控制设备通讯,接收控制 命令,回报工作状态
0x330	320 400	收:19 发:13	200	200	RS422	得到设备三轴角位置;发送设 备姿态解算所需平台运动信息
0x338	115 200	不定长约20	约1	无	RS422	发送测试信息

2.2 软件的流程设计

由图1和表2所示,基于系统的数据处理量很大,如果程序编制不合理,可能会导致系统实时控制无法实现,出现系统死机。要保证多串口通讯的更

新率和准确性是系统软件设计的最大难题。系统软件设计在结合常规编程思路的同时,针对系统特殊情况,在解决调试中出现的问题时,改进了常规思路,通过实验验证了其可行性。软件的优化设计框图如图2所示^[3]。

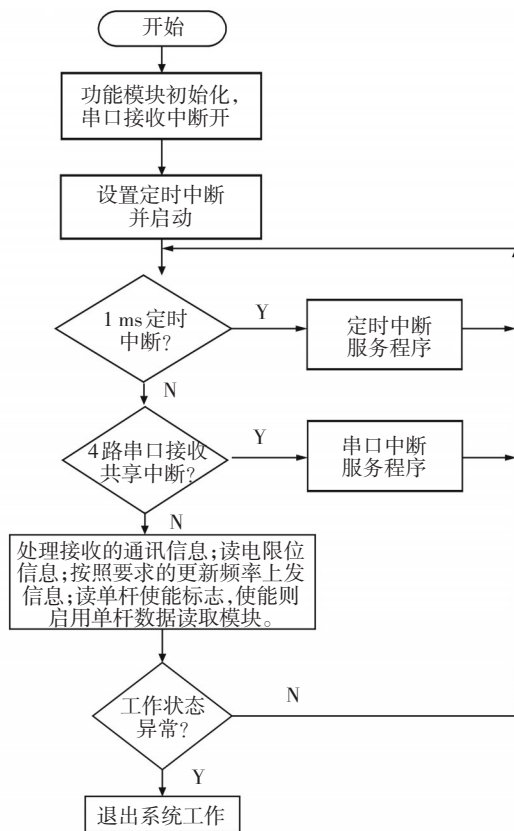


图2 软件设计框图

2.3 软件设计出现的问题及解决方法

2.3.1 串口通讯编程容易出现的问题

考虑到伺服稳定平台的实时性控制, 选用最高优先级中断, 具体工作方式不在此详述。在这里重点阐述常规串口通讯编程容易出现的问题^[4]。

(1) 串口发送数据时数据查询等待耗时过长, 数据不能实时处理的问题;

系统采取了通常发送数据实时处理, 串口发送数据到发送保持寄存器前需要查询线路状态寄存器(16C554的LSR)的第6位确定前一个字节已经发送完毕, 等待发送的过程是一个if循环, 而这个时间会是几十到几百微秒; 当一个串口的数据包发送完毕才开始进行第二个串口的数据包发送, 以此类推每次判定各个串口的工作状态。而如果对于较高更新率的串口, CPU在这时除了响应中断, 处理中断处理程序, 不能做其他任何事情, 其他进程就无法得到服务。CPU的大部分时间没有执行有效的程序代码, 完全耗费在等待之中。

(2) 容易丢包的问题

一般串口程序习惯性认为没有从接收缓冲器中取回数据, 则返回0xff, 易与实际数据包信息中的0xff混淆, 导致丢包。

2.3.2 出现问题的解决方法

(1) 针对串口发送数据时数据查询等待耗时过长, 数据不能实时处理的问题的解决方法;

通常设计中每组数据都需判断状态寄存器数据第六位与上0X40, 若不为0等待寄存器再次发送数据, 若为0数据发送完毕。这样的数据发送毫无传送效率。系统设计采用16C554连接到总线, 将常规程序设计中CPU处理串口通讯的功能, 由16C554代替进而解决CPU等待寄存器发送, 充分利用554FIFO模式下, 传输和接收前将数据缓冲为16字节数据包, 减少了CPU的中断数量。16C554提供4信道, 可实现串行和并行两种连接方式的转换, 每个信道的状态可以通过CPU的操作读取。这4路信道被称作异步通信单元(ACE asynchronous communications element), 16C554可以工作在FIFO模式下。在FIFO模式下, 每路ACE有16字节大小的先进先出缓存, 以减少引起CPU的中断次数, 优化系统的效率。每路ACE可以对波特率进行设置。首先CPU向554发送字符, 由于异步串行通信中, 字符到达时间及间隔时间往往无法预测, 此时如果采用查询方式将会浪费大量的CPU时间, 降低系统性能, 在实际设计中采用中断方式, 当串行口接收到一个新的字符或者前一个字符发送结束时, 通用异步接收器以中断方式通知CPU这样大大提高系统的实时性。写入554的数据以16位数据通过近端232或远端485标准进行传送。常规思路和系统设计程序见表3。

(2) 易与实际数据包信息中的0xff混淆, 导致丢帧问题解决办法。

通讯协议需要包括以下几方面;

① 帧头: 区别于数据信息, 如果无法分别, 应该引入透明符;

② 非定长数据帧尾: 如果帧定长, 可以无帧尾, 同样, 帧尾应该区别于数据信息, 若无法区分, 引入透明符;

③ 校验: 如果有奇偶校验等硬件校验, 则在用户层可以无校验字, 否则应该添加校验字(和校验、异或校验等), 来保证数据的正确性。

系统的四个串口有两种通讯格式, 一种是定长, 一种是非定长。其中非定长数据帧的格式包括帧头、帧尾、透明符、校验字。具体拆包流程如图3所

表3 数据处理常规思路和系统设计程序

常规思路	系统设计
<pre>void SendcharCOM1(unsigned char unch) { while((inportb(COM1BASE+5)&0x40)==0); outportb(COM1BASE,unch); } void send_fun1() { int i=0; for (i=0;i<=ssflag1;i++) { SendcharCOM1(com1_send[i]); } } void main() { send_fun1(); }</pre>	<pre>unsigned int com1_i=0,count_com1=0,com1_i=0; void send_fun1() { outportb(COMPORT3,0x7e); for(num=0;num<6;num++) outportb(COMPORT3,SendData [num]); outportb(COMPORT3,0xe7); } Viod send_fun2() { } void main() { {send_fun1(); else {send_fun2(); }</pre>

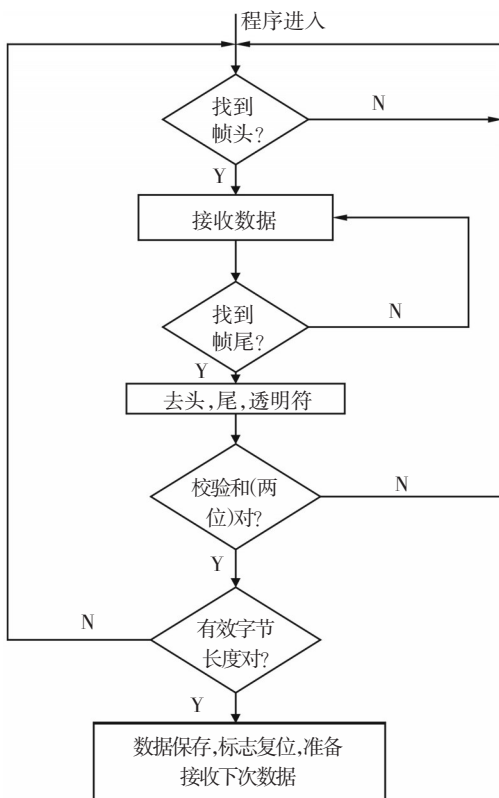


图3 串口接收函数处理流程

示。在检验帧头、帧尾、校验字的同时将剔除透明符的有效字长也进行了检验,这样得到的数据信息准确度得到进一步保障。

在缓冲区对 0xff 进行判断,如果发现了 0xff,则直接跳到下一数据进行处理,同时,在通讯协议中规定,如果数据中有 0xff 则自动在有效数据前增加一个 0xff 作为透明符。常规思路和系统设计程序如表 4 所示。

3 针对程序所做的其他优化

(1) 中断共享问题和 FIFO 设置。

为了减小 CPU 的负担,串口接收中断采用中断共享方式和带有 16 字节 FIFO 设置。中断共享方式的选择可以节省中断资源,同时方便管理和处理中断服务程序。FIFO 的设置大大降低了串口中断触发频率,提高程序效率;FIFO 的超时中断,保证了接收数据包的准确性。

指针和全局变量的分配不合理容易导致程序跑飞。

测试串口功能的连续拷机时间应该长,因为指针和全局变量的不合理分配容易导致程序跑飞,出现死机或者蓝屏或者程序退出等现象。这些在系统中是不允许

表4 丢帧问题常规思路与系统设计程序

常规思路	系统设计
<pre> unsigned char ReadCOM1(void) { unsigned char ch; if(readnum1!=innum1) { ch=buff_com1[readnum1]; readnum1++; if(readnum1>=IBUF_LEN) readnum1=0; return(ch); } else return(0xff); } </pre>	<pre> char caich1; bool ReadCOM1(char*ch) { if(readnum1!=innum1) { *ch=buff_com1[readnum1]; Readnum1=(readnum1+1)%IBUF_LEN; if(readnum1>=IBUF_LEN) {readnum1=0;} return true; } else return false; } </pre>

的。长时间拷机可以检验串口的合理性同时能保证系统的稳定、可靠性。

根据实际物理值范围,对接收的数据进行限定,增加对实际系统的保护。

接收的数据是系统的测量元件产生的,作为控制系统的反馈或前馈信号,其准确性和快速性直接影响到系统的整个控制。虽然已经有严格的数据帧结构保障,但是对于一个平台控制系统来说,这是远远不够的。所以系统针对其特殊要求和实际物理值范围,对接收处理的有效数据包进行过滤,使系统工作在可靠区域。

4 结束语

针对光电伺服转台设计中4路串口通讯的改进设计提出了新方法,经验证实现了伺服控制系统实时控制,完成了多串口高更新率通讯。选用通用异步通信芯片,将常规程序设计中CPU处理串口通讯的功能,由16C554代替进而解决CPU等待寄存器发送,减少了CPU的中断数量,解决了串口发送数据时查询等待时间过长的问题。对常规程序进行优化,解决了容易丢包、程序跑飞的问题。使控制系统可靠、稳定,为类似系统设计的编程提供了可借鉴的方

式,对编程设计者有一定的启发作用。

参考文献

- [1] 龚建伟,熊光明.Visual C++/TurboC 串口通信编程实践[M].北京:电子工业出版社,2005.
- [2] 梁越,李刚,王晓陵.基于PC/104的多串口通讯的设计[J].应用科技,2004(3):28-29,41.
- [3] 李毅,贺洪江,袁左英.DOS下实时多任务系统的设计方法[J].工矿自动化,2004(2):45-46.
- [4] 张稳稳,欧阳娴,白永林,等.PC104串口通信在工程中的应用[J].微计算机信息,2006(2):57-59.
- [5] 曹义新,江用胜,崔坤理,等.基于PC/104总线的嵌入式控制系统[J].工业控制计算机,2009(1):1-2.
- [6] 刘建军,宋文龙,高建鸣.用软件模拟实现串口通讯的异步传送[J].林业机械与木工设备,2005(5):37-38.
- [7] 姜立志.基于PC104多串口控制系统设计[C]//2011航空试验测试技术叙述交流会论文集,2011.
- [8] 宋刘飞,张蕊.基于DOS平台的串口通讯的实现[J].光电技术应用,2009,24(1):62-66.
- [9] 黎步银,姜胜利,吕文中,等.基于C语言的串口通讯程序设计[J].传感技术学报,2002(1):64-66.
- [10] 葛磊蛟,毛一之,李歧.基于C语言的RS232串行接口通信实现[J].河北工业大学学报,2008(6):12-14.
- [11] 王慧霞,石丽.VC++实现的串口通讯数据处理[J].真空电子技术,2005(5):39-41.