

一种测控通信平台的数据发送机制

葛方振

(淮北煤炭师范学院 计算机科学与技术系, 安徽 淮北 235000)

摘要:根据UDP协议的特点,介绍了测控通信平台的功能和逻辑结构。通过对数据通信流程的分析,提出了数据发送机制设计的新方法,同时,针对发送定时器带来的负面影响,提出可行的处理方法。本设计有效地提高了测控通信系统的可靠性和实时性。经过测试,本设计能够满足测控通信系统功能要求。

关键词:测控通信平台;数据发送机制;相对定时

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2007)09-0196-03

Data Transmission Mechanism of a Monitoring and Communication Platform

GE Fang-zhen

(Dept. of Computer Sci. and Tech., Huaibei Coal Industry Teachers' College, Huaibei 235000, China)

Abstract: Introduces the function and logic structure of a monitoring and communication platform based on characteristic of UDP protocol. Proposes a new design method for data transmission mechanism through the analysis of the data communication process, meanwhile, provides a feasible solution in view of the negative impact of transmission timer. This design effectively improves reliability and real-time of the monitoring and communication platform. After testing, this design can meet the functional requirements.

Key words: monitoring and communication platform; data transmission mechanism; relative timing

0 引言

目前国内绝大部分测控系统采用的还是现场总线或RS-232、RS-485,并且都直接与终端设备发生联系,这样的系统不但要完成自己的逻辑功能,还要负责与终端设备通信的实时性、可靠性、通信协议的解析等一系列复杂问题,而且每个子系统都实现这样一整套机制,造成大量重复劳动,且硬件的改动会导致各个子系统的软件都需要修改,这样牵一发而动全身,无疑使系统的开发难度和成本大大增加。

所以,本测控通信平台的目的是把测控系统的监控、通信功能独立出来,专门负责数据安全、可靠的传输和通信协议的解析等一系列工作。本测控通信平台是基于商用以太网(即IEEE802.3标准)^[1],采用UDP(User Datagram Protocol,用户数据报协议)^[2]作为数据承载,建立了完整的、有效的通信服务模型,制定了有效的实时通信服务机制,协调好现场控制系统中

实时信息的传输服务,形成在通信确定性、实时性、稳定性和可靠性等方面较好的通信平台^[3,4]。

测控通信平台的功能,一方面是监控系统中的各种设备,以保证数据传输、数据处理正确,并对设备进行远程操作控制,这些设备可能运行在以太网,也可能运行在串口或其他总线(如:RS485)上;另一方面是与上层软件交互,通知上层软件终端设备的一些工作情况或传输业务数据,同时为上层软件提供终端设备的操作接口。监控设备主要是一些智能协议的设备,它们自身有一套基本通信协议和RS232、RJ45等接口,通过这些接口,设备能进行数据输入输出。

从功能上分析,平台逻辑结构关系如图1所示,各模块之间以消息的形式交互。提取接收缓冲区数据比较容易,按照先进先出原则进行就可满足性能要求。

1 平台的数据发送机制

根据传输数据的特征看,系统发送的数据可分两类:一类是对终端设备请求的应答数据。当数据正确到达终端后,终端设备不再给任何形式的回应,而当数据没有及时到达对方或传输过程中出现某种错误时,终端设备将重发该请求,如图2(a)所示;另一类是平

收稿日期:2006-11-15

基金项目:安徽省高校青年教师科研资助计划项目(006ql137)

作者简介:葛方振(1975-),男,安徽砀山人,硕士,讲师,研究方向为分布式数据库和现场总线。

台或上层软件主动发起的数据传输,即平台或上层软件向终端发送指令,要求终端设备执行某种操作,并向平台或上层软件返回其执行的状态或结果,如图2(b)所示。

2 平台的数据发送缓冲区

为了解决数据发送出现问题后,重新发送数据,或者上层软件与终端设备之间的数据缓冲,创建一个发送缓冲区来存放数据副本和起数据缓冲作用^[5,6]。发送缓冲区是个先进先出队列,每个元素由五个字段组成,如图3所示,分别为 DevNo 字段、Data 字段、DataType 字段、SendTimes 字段和 SpareTime 字段,发送缓冲区的大小由上层软件来决定。

* DevNo 字段:发送数据目的终端的编号。

* Data 字段:系统所要发送的数据,记录内数据格式可以由上层软件规定。

DevNo	Data	DataType	SendTimes	SpareTime
3453	XXXXX	24	0	10
3433	XXXXX	41	1	8
3435	XXXXX	42	2	2
3453	XXXXX	22	4	3
4542	XXXXX	22	2	2
2343	XXXXX	41	1	5
2346	XXXXX	41	2	2

图3 数据发送缓冲区队列

* DataType 字段:记录当前项 Data 字段里的数据的类型,用以加速与应答的匹配速度。

* SendTimes 字段:主要用来记录 Data 字段中的数据可重发的次数,反映了相应数据项的状态,数据每次重发一次,它的值就会被减1。如果该字段的值变为0,则表示数据无法到达目的地,应该丢失该数据,并向上层模块通知出错信息;如果该字段为-1,则说明该数据已经成功到达对方并得到对方的应答;如果数据发送超时,只要该字段不为0,即数据的重发次数没有超过规定的次数(如:3次),数据将被重发,并被重新压到队列尾部。

* SpareTime 字段:说明该项数据距下次发送还剩余的时间单位数。它是一个相对值。如果该字段的值变为0且 SendTimes 不为0,则重发此数据。

3 平台的发送定时器

从图1可以看出,数据发送缓冲区的数据发送和数据接收缓冲区的数据接收都是在定时器的控制下工作的,针对发送缓冲区就是在发送定时器控制之下。发送定时器对发送的数据进行计数,它是一种减法计数器,当计数值变为0时,数据仍没有收到相应的应答,表示发送的数据已经超过定时范围,需要重发或丢弃。从常规上来说,在发送一个数据的同时,需要启动一个数据发送定时器,如果在定时器时限范围内该数据被确认,则关闭该定时器,否则需要重新发送该数据,并且该定时器重新开始计时。

3.1 发送定时器的定时值

从理论上,数据发送定时值是根据网络性能的不断测定,通过一种动态的算法来不断调整,才能达到一种最佳效果,如 TCP 通常采用的 Jacobson 算法,对于每一条连接,它保存一个变量 RTT,存放当前到目的端往返时间最接近的估计值,当发送一个数据段,同时启动相应的定时器后,如果时间太长就触发重发机制。如果在定时器超过定时值之前得到了确认, TCP 测量该确认花费了多长时间(假定为 M),那么根据公式 $RTT = \sigma RTT + (1 - \sigma)M$ 来修正原有的 RTT 值。 σ 是

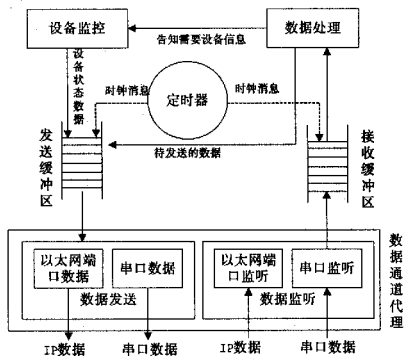


图1 测控通信平台逻辑结构关系图

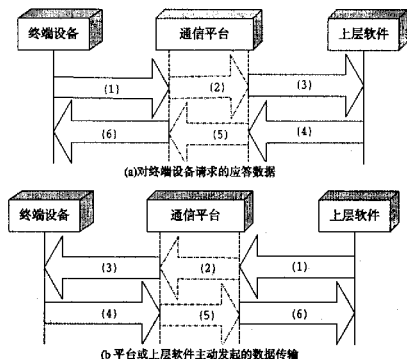


图2 数据通信流程

一个推荐值为 0.9 的平滑因子(smoothing factor),决定以前的 RTT 值的权值(即所占比例)。

由于测控系统一般网络结构不太复杂和本测控通信平台的路由器和网络流量变化很小,同时为了减小系统的复杂度,所以数据发送定时时间设为固定值,那么这个值应该设为多长?经过测试得到数据在网络中一个往返传输时间的概率密度如图 4 所示,显然将定时值设为 40ms 合理,因为在这个时间内绝大部分数据在网络没有故障的情况下都能够收到应答。

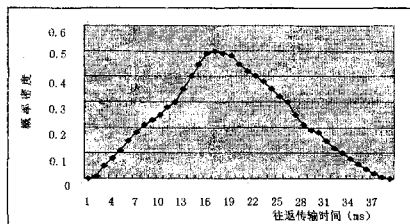


图 4 以太网测控通信系统往返时间概率密度

3.2 发送定时器的定时方法

为了减少定时器计数的维护频率,使用定制时钟的方法,即将队列头指针所对应的数据项的定时时间设置为相对值,其值就是该项数据成为队列头时的时间。也就是说,当一个数据移到队列头时,就可以为其定制一个时钟,其时钟的间隔就是该数据所对应的 SpareTime 字段值。这个时钟一到,队列头所对应数据肯定超时,因此,一个数据在一次发送过程中只会发生一次时钟维护,与每个数据采用一个计数器管理相比,从根本上降低了定时时钟的维护频率,提高平台的效率。每个数据项的实际定时时间是固定的定时值 40 与它前面所有项的定时时间的差值(如图 5 所示)。再压入一个新的数据,则它的定时时间应为 $40 - (10 + 8 + 2 + 3 + 2 + 5 + 2) = 8$ 。图 5 中,数据项的二元组分别表示数据的相对定时时间和数据剩余发送次数,如二元组 (10,0) 表示该数据相对的定时时间为 10 个时间单元,而它的重发机会只有 0 次了,也就是说数据如果在其定时器时限范围内不能收到应答,就可断定数据无法到达目的地,将被丢弃。

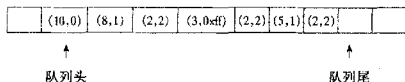



图 5 发送缓冲队列发送定时器实例

4 平台的发送缓冲区空间收集

相对定时的方法有效地解决了为维护多个定时器

而使系统运行压力过大的问题,但也带来了一定的负面作用,发送缓冲区队列是一个有限的空间,而相对定时方法的定时计算与它前面的数据项有着紧密的联系,因此不能随意地增减队列中的数据项,否则将引起系统定时的混乱。也就是说即使某项数据已经收到了应答,也不能马上从队列中删除。因此,在某些情况下会造成严重的资源浪费,甚至会导致队列空间的不足,无法容纳需要发送的数据。缓冲区空间收集就是为解决这个问题而设计的,它在需要的情况下对缓冲区队列的空间进行重新整理,释放不必要的空间,以使缓冲区队列拥有最大可利用空间。

整理缓冲区队列空间最主要的整理对象就是那些已经收到应答,但为了保持定时正确性的而不得不留在缓冲区队列中的数据项,将它们所占用的空间释放出来。图 6 中  为背景的数据项,表示已经收到了回复。当清除一项数据时,该数据所对应的 SpareTime 值必须累加到下一个数据项上。如图 6(a) 中队列的倒数第二项数据原来 SpareTime 值为 10,在空间收集时,其前一数据项的空间被收集,则前一项的相对定时值 2 被累加到该项上,故 $SpareTime = SpareTime + 2 = 12$ 。如果缓冲区队列中所有数据项都收到了应答,则将缓冲队列清空。

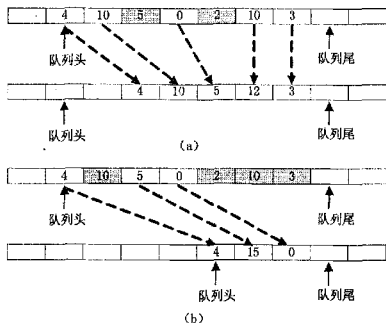


图 6 发送缓冲队列空间收集

5 结束语

文中提出基于 UDP 协议的测控通信平台系统结构,分析了平台的数据通信流程。针对平台发送数据的特点,阐述了发送定时器设计方法。本测控通信平台设计具有以下特点:平台的独立性、扩展性好;减少软件定时器计时的维护频率,提高系统的吞吐量;保证了通信的可靠性,实时性。经过测试,通信次数 80 次/ms,完全能够达到测控通信系统的实时性要求。

(下转第 202 页)

其中,占用资源有:一个 18×18 位,两个 18×6 位,一个 6×6 位的乘法器,3个48位加法器,7个48位的寄存器。

对于有符号数的相乘结构与此大致相同,只需将符号位考虑在内即可。综合布线后仿真,器件采用 Virtex-II xcv1000 fg456-6,最高频率可达到195.160MHz。

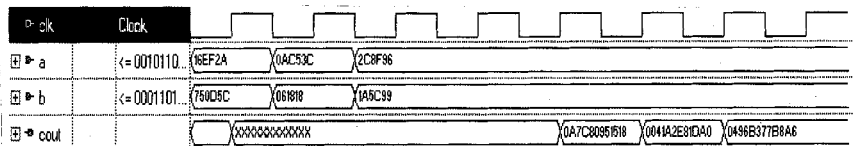


图1 2xn结构乘法器仿真波形



图2 18x18结构乘法器仿真波形

参考文献:

- [1] 梁峰, 邵志标, 梁晋. Radix-16 Booth 流水线乘法器的设计[J]. 西安交通大学学报, 2006, 40(10): 1111-1114.
- [2] Booth A D. A signed binary multiplication technique[J]. Quarterly Journal of Mech & Appl Math, 1951, 4(2): 236-240.

(上接第195页)

幅,不再一一描述。希望文中描述的声卡模式自动切换系统中的进程间通讯机制应用能够给大家带来借鉴作用。

参考文献:

- [1] Richter J. Programming Applications for Microsoft Windows [M]. 4th Edition. [s.l.]: Microsoft Press, 1999.

(上接第198页)

参考文献:

- [1] Wright G R, Stevens W R. TCP/IP 详解 卷2: 实现(英文版)[M]. 北京: 机械工业出版社, 2002.
- [2] Stevens W R. TCP/IP Illustrated Volume I: The Protocols [M]. Beijing: China Machine Press, 2000.
- [3] 李勇, 戴瑜兴. 基于UDP协议的实时监控系统的[J]. 电子技术, 2003(11): 37-40.
- [4] 费仁元, 王民, 徐洪安. 产品生产服务过程中的网络监

4 总结

通过比较, 可以看到采用相同的器件, 该方法已经将运算速度提高了150MHz以上, 大大提高了运算速度, 而且采用流水技术, 实现了并行计算。此外, 这种方法还具有结构电路简单的明显优势。

在工程实践中, 已经将这种方法应用于浮点乘法器的设计当中, 收到了良好的效果。

- [3] 李小进. 定点符号高速乘法器的设计与FPGA实现[J]. 微电子学与计算机, 2005, 22(4): 119-125.
- [4] 常静波, 郭立. 一种3级流水线Wallace树压缩器的硬件设计[J]. 微电子学与计算机, 2005, 22(1): 160-165.
- [5] 侯伯亨, 顾新. VHDL 硬件描述语言与数字逻辑电路设计[M]. 西安: 西安电子科技大学出版社, 2000.

- [2] 吴群飞. Hook技术在监控FSpice中的应用研究[J]. 计算机应用研究, 2005(4): 185-186.
- [3] 杨宁学. 内存影射文件及其在大数据量文件快速存取中的应用[J]. 计算机应用研究, 2004(8): 187-188.
- [4] 王文磊. 多线程编程技术实现经典进程同步问题[J]. 计算机技术与发展, 2006, 16(3): 110-112.
- [5] 谈蓉. Win32系统下进程间通讯[J]. 电脑开发与应用, 2003(10): 19-20.

控技术[J]. 机械工程学报, 2003(9): 35-37.

- [5] Yuan F, Huang C. Performance Analysis of Resilient Packet Ring with Single Transit Buffer [EB/OL]. 2002. [http://www.scc.carleton.ca/faculty/huang/ICT2002 final version. pdf](http://www.scc.carleton.ca/faculty/huang/ICT2002%20final%20version.pdf).
- [6] Francisco M, Yuan F, Huang C, et al. A Comparison of Two Buffer Insertion Ring Architectures with Fairness algorithms, Anchorage [EB/OL]. 2002. [http://www.scc.carleton.ca/faculty/huang/mark-icc03. pdf](http://www.scc.carleton.ca/faculty/huang/mark-icc03.pdf).