

分布式加速器控制系统时间同步系统的研究

沈佳蓉 沈立人 刘亚娟

(中国科学院上海应用物理研究所 上海 201800)

摘要 现代大型加速器控制系统大都采用分布式控制系统,系统使用网络交换各类数据。各种控制动作及数据获取要求各智能控制器的CPU基于时间进行调度,因此时标系统在分布式控制系统中具有极其重要的作用。

本文论述了采用软件实现的时间同步系统及其所采用的时间协议和原理以及达到的精度等。

关键词 分布式控制系统, 时间同步, 时间协议, 简单网络时间协议, 自适应通信环境

中图分类号 TP393

分布式控制系统网络上存在各种各样的智能控制设备,各种控制动作都要求各CPU基于时间对设备进行调度,各类数据获取都依赖于时标进行工作。如果各系统之间时间不一致,会产生各种严重的后果,影响整个控制系统的工作,因此一个精确的时标系统直接关系到控制系统的成败。

本文论述了基于以太网的分布式控制系统采用软件实现的时间同步系统,设计精度误差小于10ms。由于简单网络时间协议(SNTP)既能达到我们的精度要求而且复杂性也不是很高,系统采用SNTP协议来实现。SNTP协议在UDP端口通信,是无连接的,因此每个客户端不会占用服务器过长时间,对于有很多个客户端的情况不会造成等待很久的情况,符合时间同步的要求;但同时由于它是无连接的,具有不可靠性,不能保证每个消息都能到达目的地,因此要弥补这一不足,在这里采用了重发和日志记录的方式,在消息丢失的情况下进行重发。并且无论是客户端还是服务器对于消息的丢失这一事件都会记录在日志中,留待系统管理员检查。

系统用C++语言开发,采用了自适应通信环境(ACE)网络编程工具包,ACE是开放源代码的自由软件,它有许多先进的设计和框架,并且最主要的原因是它具有可移植性,它消除了C++编译器之间的差异,可以方便地移植到各种操作系统,这对于使用不同的操作系统的大规模分布式控制系统是很有用的。

1 时间协议

时间协议用在互联网上传递统一标准的时间,

目前网络上使用的时间协议有好几种,包括: Day Time Protocol (RFC-867)、Time Protocol (RFC-868)、Network Time Protocol (RFC-1305)、Simple Network Time Protocol (RFC-2030)。

目前使用最广泛的是NTP(Network time protocol)协议,它的时间戳使用port 123, 64位的二进制码,前32位和Time Protocol一样,后32位用以表示s以下的部分,并加上网络传播延迟量的估计,理论上可以精确到 2^{-32} s,实际使用上只能有1—50 ms。NTP是由美国德拉瓦大学的David L. Mills教授于1985年提出的,除了可以估算封包在网络上的往返延迟外,还可独立的估算计算机时钟偏差,从而实现在网络上的高精度度计算机校时,它是设计用来在Internet上使不同的机器能维持相同时间的一种通讯协议。SNTP是为了那些不需要完整NTP实现复杂性的主机,它是NTP的一个子集,通常让局域网上的若干台主机与其他的NTP主机同步时钟,接着再向局域网内其他客户端提供时间同步服务。

鉴于大部分分布式控制系统基于局域网交换数据,我们采用SNTP协议来实现时钟同步,并采用特定的端口通讯。

1.1 SNTP的报文格式

NTP和SNTP是基于用户数据报协议(UDP)的,SNTP报文格式的描述紧跟在IP和UDP报头之后。SNTP的消息格式与NTP的格式是一致的,不同的地方是:一些SNTP的数据域已被封装,也就是说已初始化为一些预定的值。图1是SNTP报文格式的描述^[1]。

第一作者: 沈佳蓉, 女, 1980年出生, 2006年于中国科学院上海应用物理研究所获硕士学位

通讯作者: 沈立人

收稿日期: 2006-03-15, 修回日期: 2006-04-28

0								1								2								3							
0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
LI			VN			Mode		Stratum								Poll								Precision							
Root delay																															
Root dispersion																															
Reference identifier																															
Reference timestamp(64)																															
Originate timestamp(64)																															
Receive timestamp(64)																															
Transmit timestamp(64)																															
Key identifier (optional)(32)																															
Message digest (optional)(128)																															

图1 SNTP协议的报文格式

Fig. 1 SNTP message format

1.2 SNTP 基本原理

网络延时与时间偏差测量的有关说明见图2。客户端封装SNTP报文，在传送时间戳字段写入它的发送时间，然后发送给服务器，服务器收到后在接收时间戳字段写入它收到报文的时间，将传送时间戳字段的值拷贝到原始时间戳字段，然后再在传送时间戳字段写入它的发送时间，将报文回复给客户端，客户端收到后再在目的地时间戳字段写入它的收到时间。

t 为服务器和客户端之间的时间偏差； d 为两者之间的往返时间，即网络延时。

$$\text{由于} \begin{cases} T2 = T1 + t + d/2 \\ T4 = T3 - t + d/2 \end{cases} \quad (1)$$

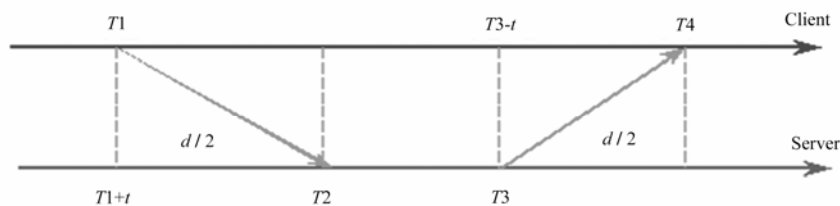
$$\text{所以有} \begin{cases} T2 - T1 = t + d/2 \\ T3 - T4 = t - d/2 \end{cases} \quad (2)$$

$$\begin{cases} t = [(T2 - T1) + (T3 - T4)] / 2 \\ d = (T2 - T1) - (T3 - T4) \end{cases} \quad (3)$$

由公式(3)可以看到， t 值和 d 值只与 $T2 - T1$ 和 $T3 - T4$ 值有关，而与 $T2 - T3$ 值无关，即与服务器处理请求的时间无关。根据上述公式，客户端就可以算出时差 t 去调整本地时钟^[1]。

2 系统的软件实现

系统的结构：在局域网内使用一个时间服务器，这个时间服务器直接从GPS获得精确的时间。局域网内的各个客户端CPU同这个服务器进行同步。目前实现的这个时间服务器使用的是Linux平台，时间服务客户端有Linux和Windows两个平台。为了系统的可靠性，还可以在系统内建立一个冗余时间服务器，在主时间服务器不能工作时，客户端和冗余服务器保持同步。



Timestamp Name	ID	When Generated
Originate Timestamp	T1	Time request sent by client
Receive Timestamp	T2	Time request received at server
Transmit Timestamp	T3	Time reply sent by server
Destination Timestamp	T4	Time reply received at client

图2 同步时序图

Fig. 2 Synchronous time figure

实现方式：系统实现同步的方式是在后台持续运行一进程，客户端周期性地主动发送 SNTP 报文给服务器，服务器收到后回复 SNTP 报文给相应的客户端，客户端再根据 SNTP 报文里的时间戳计算两者之间的时钟偏差，如果计算得到的时钟偏差大于 0.5 ms 就修正客户端的时钟。这里的周期是一个灵活的周期，客户端会记录下上次对时的时钟偏差，周期会随着时间偏差的多少来调整，如果偏差大相应的周期就会短，如果偏差小则下次对时的周期就会长^[2]，采用这样的灵活周期是为了既要达到同步的精度，又不会占用太多的网络带宽。目前的设计是时钟偏差大于 0.5 ms 则 60 s 后进行下次对时，小于 0.5 ms 则 120 s 后进行下次对时。同时为了结合控制系统对于稳定性的要求，还结合了心跳机制，对于各个 CPU 的通信状态进行了检测，在客户端和服务端都有日志记录，对于收不到报文等异常情况都有记录，管理员可以迅速地了解到各个 CPU 的状况，作出反应。

上海光源控制系统使用了 Linux、VxWorks、Windows 操作系统。系统原型是在 Linux 上实现的，现在已将客户端移植到了 Windows 上，并将很快移植到 VxWorks 操作系统上。目前还可在 HP-UX，Sun Solaris 等 UNIX 系统上运行。

由于软件的实现使用了 ACE，ACE 提供了一系

列的类的可跨平台的调用，并且这些类的构造函数已经消除了一些编程中的常见错误^[3]，所以源代码简短清晰并且具有很强的可移植性和可靠性。

3 系统的精度测试

为了确保时钟同步能够达到比较好的精度，系统采用了 QoS 技术，对自定义端口的 SNTP 报文设置一定的优先级，确保它不会在网络上拥堵。精度测试仍采用 SNTP 协议的原理即公式(3)来计算，客户端每 30 s 发送报文给服务器，收到服务器答复后计算两者之间的时间偏差取绝对值，再将测试保存到指定文件中。

测试环境：上海应用物理研究所局域网络，千兆网络，网络负载约在 30%—50%，运行各类应用，其中包括各种 P2P 协议的应用。

测试结果：每 0.5 min 测试一次，测试 24 h 后得到 2880 个数据，其中最大的精度误差是 3896 μ s，最小的精度误差是 11 μ s，平均精度误差是 341 μ s，达到了 ≤ 10 ms 的设计要求。将测试得到的 2880 个数据分组并取每组精度误差平均值(每 10 个数据为 1 组，共得 288 个平均精度误差值)，也就是通过计算得到每 5 min 的平均对时精度误差，得到的平均精度误差曲线见图 3。

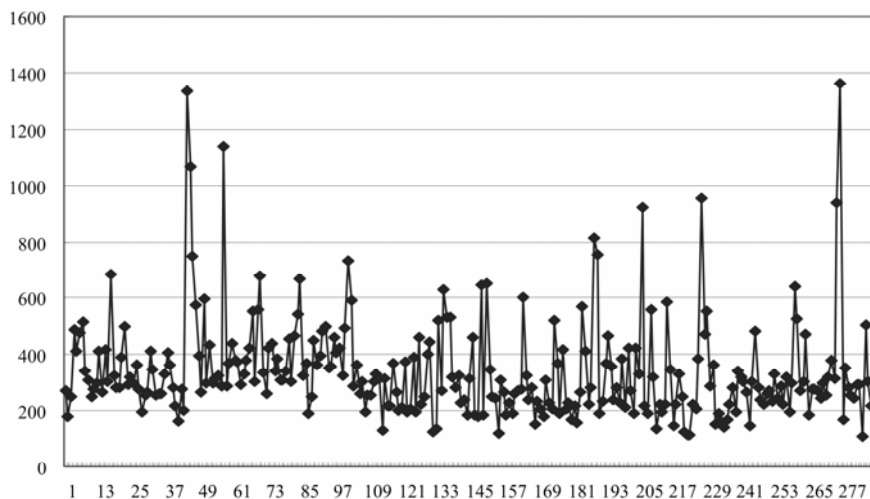


图 3 精度测试结果
Fig. 3 Result of precision test

4 结束语

现在这个时钟同步系统已经在小范围内的 Linux 和 Windows 两个平台的机器上运作，并即将

移植到 VxWorks 上。由于使用的是纯软件实现，以及 SNTP 本身能达到的精度有限，因此下一步将结合 NTP 的一些算法设计，对系统进行改进，以更好地满足分布式控制系统的需求。

参考文献

- 1 Mills D L. RFC2030 Simple Network Time Protocol (SNTP) Version 4, October 1996
- 2 CERNET 网络中心时间服务网 <http://www.time.edu.cn/ntp.htm>
- 3 Huston S D, John J C E, Syid U 著(马维达译). ACE 程

序员指南, 网络与系统编程的实用设计模式. 北京: 中国电力出版社, 2004. 1—165

Huston S D, John J C E, Syid U (Ma Weida trans.). The ACE Programmer's Guide: Practical design patterns for network and system programming. Beijing: China Electric Power Press, 2004. 1—165

Implementation of time synchronization in distributed control system of large accelerators

SHEN Jiarong SHEN Liren LIU Yajuan

(Shanghai Institute of Applied Physics, the Chinese Academy of Sciences, Shanghai 201800, China)

Abstract Nowadays the distributed control system is used in most of large accelerator control systems. The system exchanges various kinds of data on the network. Since the control operation and data access require the CPUs to be regulated based on time, a time system is very important for the distributed control system. This paper describes the time protocol and its principle used in the time synchronization system, and introduces how to use software to build the system and to test its precision.

Key words Distributed control system, Time synchronization, Time protocol, Simple network time protocol, Adaptive communication environment

CLC TP393