

# 改进型 UDP 组播的性能分析及其在实时电源控制系统中的应用

陈海春 傅鹏 程敏

(合肥中国科学院等离子体物理研究所二室,合肥 230031)

E-mail:chenhaichun@sina.com

**摘要** 文章通过采用带连接的 UDP、I/O 复用技术、关闭 UDP 校验和功能、负载均衡等技术对普通的 UDP 组播技术进行改进,使之在每个控制周期(1ms)内满足了分布式实时控制系统的实时性和可靠性要求。在将改进型的 UDP 组播技术应用于实际控制系统之前,文中构建了一个模拟局域网进行测试,对比结果表明改进后的组播技术无论在数据包延迟、丢包率等方面较改进前的 UDP 组播性能有较大幅度的提高。

**关键词** UDP 组播 QNX 分布式实时控制系统

文章编号 1002-8331-(2004)20-0201-04 文献标识码 A 中图分类号 TP273

## Performance Analysis of the Modified UDP Multicast Technology and Which Apply to Realtime Control System for a Power Supply

Chen Haichun Fu Peng Cheng Min

(Institute of Plasma Physics, Chinese Academy of Sciences, Hefei 230031)

**Abstract:** This paper discusses a modified UDP (User Datagram Protocol) multicast technology via connected UDP, I/O multiplex technology, and closes the check-sum function in UDP header, etc. This modified UDP multicast is perfectly satisfy the special requirement of the Poloidal Field Power Supply in every control cycle-1ms. However, before apply this technology into the real system built a simulation LAN for proving its performance, as the result, the optimized UDP multicast is more better than the original UDP multicast not only in the packages delay, but the percentage of packages lost.

**Keywords:** UDP multicast, QNX, distributed and realtime control system

为达到 HT-7U 极向场分布式实时电源控制系统的通信实时性、可靠性的要求,文中选用了 UDP 协议,但是在使用和测试过程发现普通的 UDP 还有很多可改进的地方。因此,首先构建了一个局域网 UDP 测试系统对比分析了改进后和改进前的 UDP 组播性能;然后将改进后的 UDP 组播技术用于国家“九五”重大科学工程“HT-7U 超导托卡马克装置”上极向场控制系统——由 12 套软、硬件系统组成,这些装置在空间上具有较强的分散性,构成了一个硬实时和软实时共存的分布式控制系统,系统设定的控制周期为 1ms。测试平台和实时工作平台选用 QNX6.20 实时操作系统(一种符合 POSIX1003 标准、支持内核抢占、微内核、基于消息传递的 RTOS),其 TCP/IP 堆栈为 4.4BSD-Lite 版。

### 1 UDP 协议的可改进之处

在无法修改 TCP/IP 堆栈源代码、重新编译 OS 的情况下,采用一些简单易行(却往往被忽略)的方式来改进 UDP 的性能,如采用带连接的 UDP、I/O 复用技术、关闭 UDP 校验和功能、负载均衡等。

#### 1.1 可连接的 UDP

普遍认为 UDP 是一种非链接、无状态的传输层协议,其实在使用 UDP 协议中最常用的发送函数 `sendto()` 时,系统都认为它是一种连接套接字的特例<sup>[1]</sup>,UDP 协议发送函数 `udp_output()` 自动先连接远程套接字、发送数据、然后断开连接;当然这里的‘连接’指的不是 TCP 协议中经典的三次握手过程,在 UDP 协议中由 `connect()` 实现的连接仅仅是将远程的端口、IP 地址填充到 UDB 结构<sup>[2]</sup>(UDP 的协议控制块)中。每次发送数据报时刻的隐含连接过程几乎占用了整个 UDP 传输时间的 1/3,这主要是由于系统在发送每个数据报时调用 `connect()` 引发临界区竞争、查找路由表和连接操作本身的未优化而引起的<sup>[1]</sup>。在使用连接的 UDP 发送数据报时应该使用 `write()` 函数发送,因为它已经不需要填充每个数据报的目的地址了。

#### 1.2 I/O 复用技术

使用这种方式主要是考虑减少组播发送端用于接收返回信息的线程/进程,通常在实际工作中使用了一个线程/进程接收一个特定的回答信息,这样需要的线程/进程数目就等于发送回答信息机器的数目。由数据链路层的 802.3 协议可知在任何特定的时刻只能某个发送端独占链路,再者大部分情况下使用的都是单 CPU 系统,因而可知,整个通信链路、处理软件

基金项目:国家发展计划委员会“投资(1998)1303 号项目”(子项目)

作者简介:陈海春(1979-),男,硕士研究生,主要研究方向为实时控制、TCP/IP、实时操作系统 QNX 的应用。傅鹏,男,研究员、博导,主要研究方向为核聚变装置电源、电力电子及自动化。

都是串行工作,因此这里采用 I/O 复用技术减少服务器端接收的线程/进程数量。

I/O 复用技术的工作过程如图 1 所示,在测试系统中采用 2 个线程接收 6 台机器的回答信息,从而减少了 4 个线程(需要接收的机器数越多性能的提升也越明显)。实现 I/O 复用技术可以使用 `select()` 和 `poll()` 两个系统调用来实现,该文使用 `select()`。

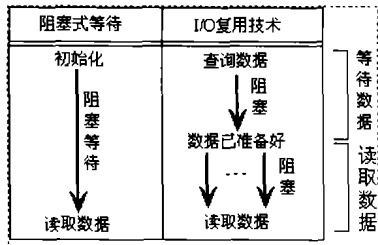


图 1 I/O 复用技术工作过程

### 1.3 关闭 UDP 校验和功能

UDP 校验和需要计算 UDP 头部、UDP 数据包(如果数据包大小不是 16 的整数倍,那么将会在数据包的尾部填充 0,填充的 0 不发送)、16 字节的 UDP 伪首部(源 IP,目的 IP 等信息)。计算过程是每 16bit 反码和,如果数据稍大那么计算的花费是非常可观的。由于在一个局域网中传输的数据包由于不需要经过路由器的转发、不同链路协议之间的转换,因而关闭此校验和不会对 UDP 数据的正确与否造成影响<sup>[9]</sup>。关闭 UDP 校验和的做法很简单,只需要在发送端将 UDP 头部的“16 位 UDP 校验和”置 0 即可,接收端看到此项为 0 后将不会进行验证。但是,设置该选项需要创建一个原始套接字,用户填充 UDP 头部各项(所有的字节都是大端对齐的(big-endian),即低位在左高位在右),而 IP 的头部一般由系统填充。

### 1.4 负载均衡

如果有多块网卡工作在同一子网中,TCP/IP 内核在发送数据包时选择的默认出口是第零块网卡(`eth0, en0`),并且每次发送数据的时候都需要查询路由表进行填充 IP 的首部。因而为了减小冲突、做到负载均衡,该文采用对不同数据传输通路绑定不同网卡的工作方式,这步工作主要是使分布式实时控制系统中的备份数据与控制信息相分离的情况。文中使用 `bind()`

函数将特定的套接字与对应的 `sockaddr_in{}` 结构体内容相绑定。

## 2 测试系统的网络拓扑结构和硬件配置

实时控制信号通过粗线缆传输、备份数据通过细线缆传输,并且通过两个以太网交换减少以太网帧的冲突。其中组播发送端也通过蓝色线缆来接收各工作站的回答信息。

## 3 普通 UDP 组播和优化后的组播性能对比

在同等硬件配置、同样的网络拓扑结构、相同的进程优先级、相同的进程调度算法(FIFO)的情况下两种 UDP 组播通信的双向时间、包丢失率、包延迟情况以及同步性。组播发送端每毫秒组播 1024 字节的数据,然后就启动两个接收线程进行监听接收(回答数据包的大小为 100 字节)。第一到第六台工作站每毫秒接收组播数据,并立即对之进行回答,然后每 16 毫秒将共享内存中的所有数据传至数据库服务器进行备份。双向时间测试在组播发送端进行,包丢失率和延迟情况、同步性在各组播接收工作站测量。

### 3.1 双向时间

双向时间见图 3、图 4。

### 3.2 包丢失率和延迟情况

ST1~ST6 表示 station1~station6。表 1 的测试结果是每次 10 万次,测量 3 次后的平均值。

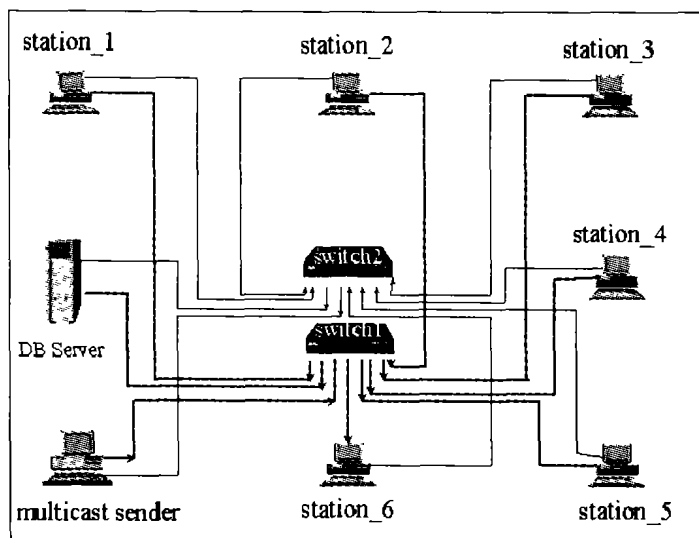
优化后的工作对改善超时性能效果明显, 超时率低于 0.1%。优化后的包丢失率均为零。

### 3.3 各组播接收端的同步性(示波器测量)

图 5 和图 6 显示的第一路波形是组播发送端的毫秒定时器的定时波形。从图中可以看出各接收端基本上是同时收到组播数据包的,也可知优化工作对各接收端的同步性影响甚微(因为 1.1,1.3,1.4 的优化项只影响各接收端的处理和发送数据包的情况)。但由于使用的示波器只有 4 个通道,因而在此也只能观察到 4 路信号波形。

## 4 优化的 UDP 组播技术在实时电源控制系统中的应用

HT-7U 极向场分布式实时控制电源系统的拓扑结构如图 7 所示。



硬件配置:

所有网卡皆为  
3COM905B-TX,

station1~station\_6  
内存皆为 256M  
CPU C-800M  
C-633M  
P4-1.7G  
P3-600M  
P4-1.0G  
P3-866M

组播发送端:  
CPU-866M, MEM-256M

数据库服务器:  
CPU-2.4G, MEM-512M

图 2 测试系统的网络拓扑结构

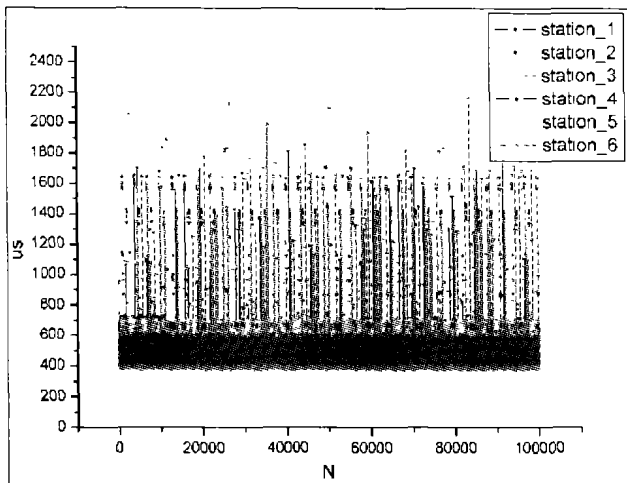


图3 优化后的 UDP 双向时间

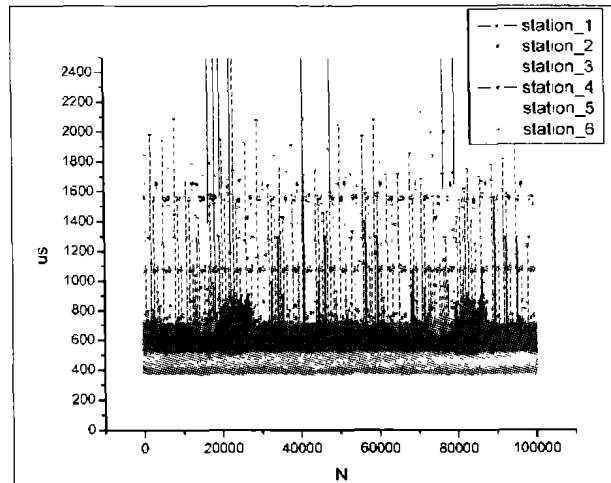


图4 优化前的 UDP 双向时间

表 1

	测量次数	超时次数						包丢失率					
		ST1	ST2	ST3	ST4	ST5	ST6	ST1	ST2	ST3	ST4	ST5	ST6
优化前	3*100,000	720	753	700	657	741	1274	0%	0.005%	0%	0.004%	0%	0%
优化后	3*100,000	27	90	37	46	53	133	0%	0%	0%	0%	0%	0%

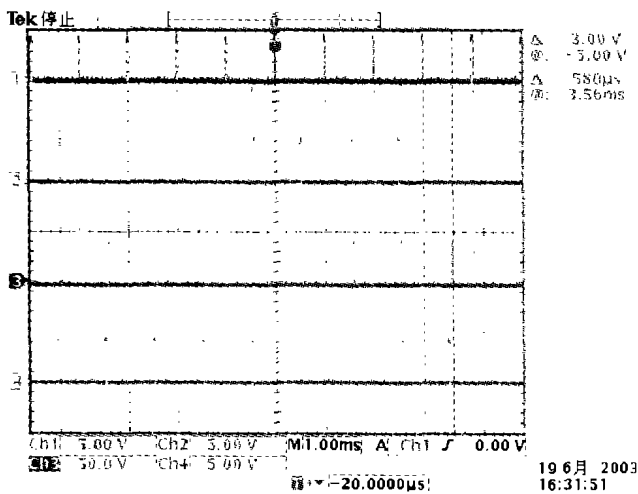


图5 优化后的工作站 1~3 的同步图

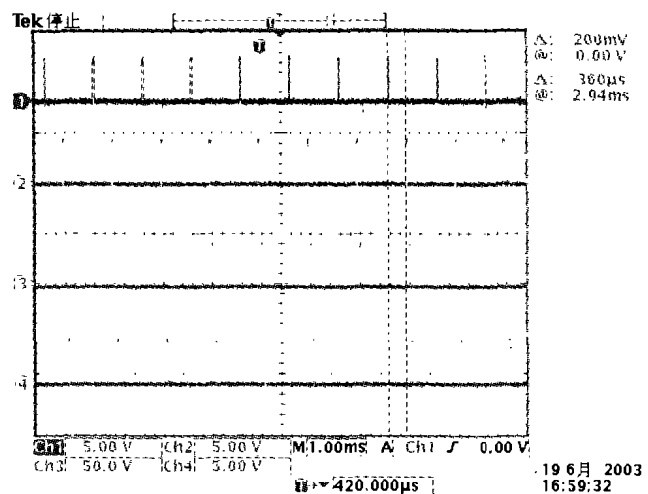


图6 优化前的工作站 1~3 的同步图

根据系统硬件的位置、核聚变实验的要求、系统模块化和便于将来系统的升级以及开发的方便,在此采用层次化的软件体系结构,将整个系统分为三层:Windows 监测层、QNX 实时控制层、现场总线执行层。

QNX 实时控制层的反馈控制节点(Feedback node)相当于图 2 测试系统中的组播发送节点,采用改进后的 UDP 组播技术在每个控制周期(1ms)内向 PF1~PF6 组播发送运算后的控制分量并且接收各子控制器的回答数据;数据库节点(Data Server)始终处于 UDP 接收状态(接收 QNX 实时层的各种备份数据)。Windows 监测层通过 UDP 单播或者 ODBC 方式从数据库节点取得当前需要察看的数据。PF1~PF6 子系统控制器也采用 UDP 单播方式将各种现场开关动作信号、水温水压调节等信号发至现场总线的控制接收模块。

## 5 结论

测试结果表明改进后的 UDP 组播性能无论在实时性、可靠性完全满足系统的要求,都在系统允许的误差范围之内(系统要求数据延迟率小余 0.01%,丢失率小余 0.001%),并且已经成功地应用于极向场控制系统,实际运行表明改进后的 UDP 组播通信在 LAN 实时控制系统中是可靠、可行的。当然为了进一步改善 UDP 的双向时间,可以采用软件方法模拟 Token Ring 的运行方式,即图 2 中的 ST1~ST6 不是接收到组播包以后马上回答,而是每个节点在系统运行范围内等待不同的时间后进行回答,这样将会进一步的减少图 3、图 4 的‘毛刺’情况。(收稿日期:2003 年 11 月)

## 参考文献

- 1.Craig Partridge,Stephen Pink.A Faster UDP[J].IEEE/ACM Trans on Networking,Volume:1 Issue:4,1993:429~440
- 2.W Richard Stevens.陆雪莹等译.TCP/IP 详解卷 2[M].机械工业出版社

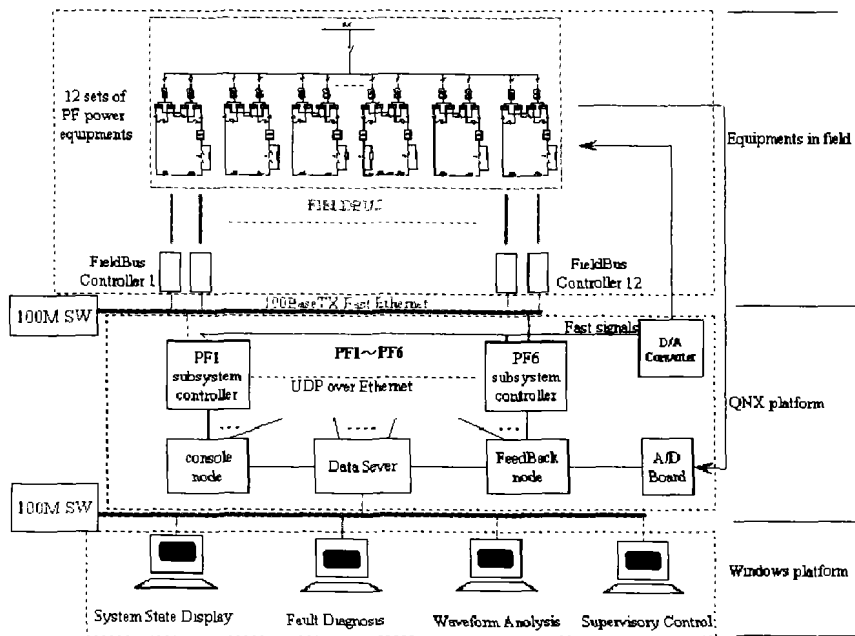


图7 极向场分布式控制系统拓扑结构图

- 社,2000
- 3.W Richard Stevens.范建华等译.TCP/IP 详解卷 1[M].机械工业出版社,2000-04
  - 4.杨清宇,施仁.基于因特网的工业控制网络体系结构研究[J].信息与控制,2002;(5):466-472
  - 5.Feng-Li Lian,James R Moyne,Dawn M Tilbury.Performance Evalua-

- tion of Control Networks: Ethernet, ControlNet, and DeviceNet[J].IEEE Control System Magazine,2001;21(2):66-83
- 6.李炳宇,萧蕴诗.网络控制系统的应用现状与发展方向[J].测控技术,2002;21(2):1-3
- 7.Bushnell L G.Networks and Control[J].IEEE Control System Magazine,2001;21(1):22-23

(上接 176 页)

```
create table test(tno int,tname char(30));
insert into test values(1,"English");
```

该序列种包含的基础操作有 create table 和 insert,对应到 Oracle 是 table 和 insert 两个语句选项。此序列的关键点是 create table 操作和 insert 操作上各配置一种类型的审计;对应到 Oracle 是 table 和 insert 选项上各配置一种类型的审计。对于该操作序列,create table 操作上支持语句审计和用户审计,insert 操作支持语句审计、用户审计和对象审计,因此,此时关键点对应的审计配置饱和度为  $(33.3\%+50\%)/2=41.7\%$ 。

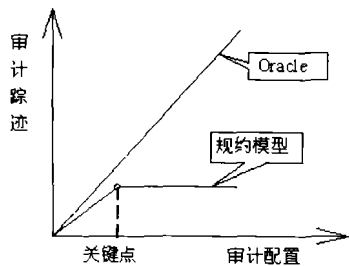


图2 审计踪迹随审计配置的变化规律

在关键点处,执行上述 SQL 命令序列,新模型和 Oracle 各生成两条审计踪迹,其中一条对应于 insert 操作(或 insert 选项),另外一条对应于 create table 操作(或 table 选项)。新模型的审计踪迹规模略小于 Oracle,这是因为前者中对象的信息只记录一次,而后者则记录两次。

当该操作序列的审计配置饱和度增长到 100%时,执行上述 SQL 命令序列,Oracle 系统生成的审计踪迹共有 5 条,其中包括 table 选项的审计踪迹两条、insert 选项的审计踪迹三条;

归约模型仅生成两条审计踪迹,其中包括对应于 create table 操作和 insert 操作的审计踪迹各一条。

综合以上分析,在关键点之后,Oracle 的审计踪迹依旧呈线性增长,而采用归约模型的系统审计踪迹的规模不再增长。审计踪迹规模的降低同时意味着写审计踪迹代价的降低,从而带来整个数据库系统性能的提高。因此,对于给定的应用,审计配置的饱和度越高,归约模型的优越性越明显。在实际应用中,审计配置的交叉是不可避免的,导致对应 SQL 命令序列上的审计配置饱和度高于关键点处的饱和度。因此,归约审计模型有巨大的使用价值。

### 3 结束语

文章在分析当前数据库安全产品的审计模型的基础上,结合审计模块的安全需求,对需要记录的审计信息进行了关联性分析,提出了归约审计模型。该模型能够有效降低审计踪迹的规模,同时为安全管理员提供更丰富的安全信息。实践中,规约模型已经成功应用到中国航天科技集团的数据库系统软件 Oscar 中。(收稿日期:2004 年 1 月)

### 参考文献

- 1.National Computer Security Center.A Guide to Understanding Audit in Trusted Systems.NCSC-TG-01,Version 2,1988-06
- 2.Oracle Corporation.Oracle 9i Database Concepts,Release 1(9.0.1), Part No.A8856-02,2001-07
- 3.Microsoft Corporation.C2 Administrator's and User's Security Guide. http://www.microsoft.com/security/
- 4.DB2 Developer Domain.DB2 security and instances,http://www.ibm.com/software/database/db2