

基于 VxWorks 的网络数据实时处理软件的设计与实现

王 灏, 郝 朝, 张 国 旺

(中国飞行试验研究院 测试所, 西安 710089)

摘要: 机载实时数据处理可以大幅提高数据处理的效率; 为满足机载网络化测试系统架构下的实时数据处理的需求, 设计并实现了一种基于 VxWorks 的网络数据实时处理软件; 该软件运行在多核嵌入式数据处理系统上, 通过多任务并发执行的方法保证实时性; 软件运行过程中利用零拷贝技术实时接收机载测试系统发送的网络包数据, 然后按照自定义的文件格式存储原始数据, 同时将需要处理的网络包存入数据处理缓存中, 根据网络数据协议实时解析网络包数据, 最后利用预留缓存的办法将处理结果写入结果文件; 实践证明, 该软件能快速接收并实时处理网络数据, 同时将原始数据和结果数据进行存储, 该方法有效提高了数据处理的效率, 为飞行试验数据处理提供了新的途径。

关键词: 网络数据; 多核多任务分配; 双缓存区; 预留缓存

Design and Implementation of Real-time Network Data Processing Software Based on VxWorks

Wang Hao, Hao Zhao, Zhang GuoWang

(Chinese Flight Test Establishment, Xi'an 710089, China)

Abstract: Airborne real-time data processing can greatly improve the efficiency of data processing. In order to meet the needs of real-time data processing under the airborne networked test system architecture, a real-time network data processing software based on VxWorks was designed and implemented. The software runs on a multi-core embedded data processing system and ensures real-time performance through the multi-task concurrent execution method. During the software running process, the zero-copy technology is used to receive the network packet data sent by the test system in real time, and then the original data is stored in a custom file format. At the same time, the network packets that need to be processed are stored in the data processing buffer. Parse the network packet data, and finally use the reserved cache method to write the processing result to the result file. Practice has proved that the software can quickly receive and process network data in real time, while storing raw data and result data. This method effectively improves the efficiency of data processing and provides a new way for flight test data processing.

Keywords: Network data; multi-core and multi-tasking allocation; dual cache area; reserved cache

0 引言

数据处理是飞行试验过程中的关键一环, 对于飞行试验的效率有着非常重要的影响^[1-2]。通常情况下, 数据是在飞行试验结束后通过“卸载、分流、处理、分发”的方式发送至课题, 这种数据处理方式的效率较低, 课题往往需要等到一段时间才能得到结果数据^[3]。

目前在飞行试验中广泛使用网络化测试架构, 在该架构下机载测试系统将采集的数据通过网络包的形式发送至存储设备、遥测发射设备等^[4-5]。如果在机载网络交换机上连接一台数据处理设备, 该设备就可通过以太网接收机载测试系统发送的所有网络包数据, 然后对数据进行解析并存储处理结果。该方法在飞行的过程中可以实时处理网络数据, 在飞行结束后可通过拷贝的方式直接得到结果数据,

这就省去了传统数据处理中的卸载、分流、处理的过程, 大幅提高了数据处理的效率。

当前各类军民用飞机试飞时对于机载实时数据处理都有着广泛的需求, 如何设计一个可靠的数据处理架构, 在满足机载网络网络化测试环境的前提下, 实现数据的高效处理, 是数据处理工程师必须要解决的一个问题。

1 机载实时数据处理介绍

随着增强型网络化遥测系统的广泛使用, 机载测试系统由之前的 PCM 架构逐渐转化为网络化测试架构^[6]。相比于之前的 PCM 测试系统架构, 网络化架构配置更灵活、使用更方便、产品更丰富^[7]。网络化机载测试系统架构下, 系统采集的参数量大、采样率高, 采集的数据以网络包的形式在测试系统上传输^[8-9]。机载实时数据处理设备通过网络的方式实时接收网络包数据, 将接收到的网络数据存储至硬盘中, 同时根据网络包的数据格式对数据进行解析, 并将处理结果存储在结果文件中。机载实时数据处理极大地提高了数据处理的效率, 真正实现了“边飞行、边处

收稿日期: 2019-12-12; 修回日期: 2020-01-13。

作者简介: 王 灏(1992-), 男, 陕西渭南人, 硕士研究生, 助理工程师, 主要从事飞行试验实时监控与数据处理技术方向的研究。

理”, 飞行结束后课题可以快速地得到数据处理结果。

机载数据实时处理系统有如下几个方面的特点:

1) 实时接收的网络数据流比较多, 数据流量比较大。在当前部分型号上, 机载网络数据流可达上百条, 数据的位速率可达 60~70 Mbps;

2) 实时数据处理软件在数据处理过程中要保证数据处理的实时性, 否则会由于网络数据流过大而发生丢包、错误处理等方面的问题;

3) 除了进行实时数据处理, 系统还需要完整记录原始数据, 所以需要定义数据存储格式, 方便飞行结束后数据卸载;

4) 为了提高处理器的利用率, 减少频繁的 I/O 操作, 数据处理过程中需要设计缓存来存储中间数据, 提高系统的运行性能。

2 数据处理软件架构设计

由于需要对机载测试系统采集的数据进行实时处理, 而 VxWorks 是业内公认的实时性好、可靠性高的系统^[10], 所以在 VxWorks 平台的基础上, 设计并实现了网络数据实时处理软件, 并在实验中测试了该软件的有效性。为了满足机载网络数据实时处理的需求, 该软件需要设置网络数据接收、原始数据存储、实时数据处理、结果数据存储四个任务, 同时设置原始数据缓存、处理数据缓存、结果数据缓存三个缓冲区以及原始数据存储信号量、数据处理信号量、结果存储信号量三个二进制信号量, 系统运行时先初始化三个数据缓存以及三个二进制信号量, 然后发起网络数据接收、原始数据存储、实时数据处理、结果数据存储四个任务, 之后这四个任务在多核多任务环境下执行, 软件运行流程如图 1 所示。

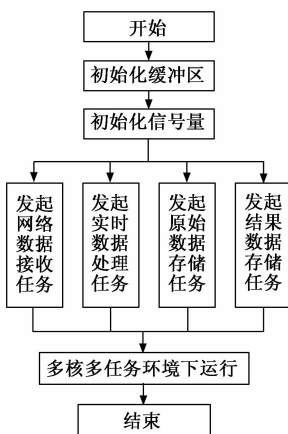


图 1 软件运行流程

网络数据接收任务实时接收来自机载测试系统发送的网络包数据, 然后将接收到的网络数据存入原始数据缓存及实时处理缓存中; 当原始数据缓存存满之后, 通过原始数据存储信号量通知原始数据存储任务进行存储, 当实时处理缓存存满时, 通过实时数据处理二进制信号量通知实时处理任务从缓存队列中读取网络包数据进行处理, 实时

处理任务根据网络数据协议对数据进行解析, 然后将处理结果放入结果缓存中。当结果缓存存满时, 通过二进制信号量通知结果数据存储模块将处理结果写入到硬盘。软件的整体架构如图 2 所示。

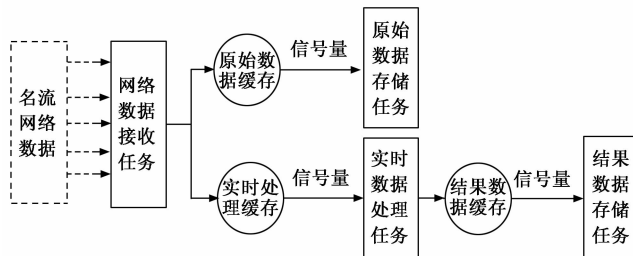


图 2 网络数据实时处理软件架构

为了提高数据处理的效率, 机载实时数据处理设备采用了双核处理器, 所以在设计数据处理软件时需要多核多任务进行分配, 分配的原则是负载均衡, 即软件运行的过程中每个核的利用率基本保持一致。设置网络数据接收和原始数据存储任务运行在一个核上, 实时数据处理和结果数据存储运行在另外一个核上。另外由于 VxWorks 是基于优先级任务调度的实时操作系统, 所以需要设置各任务的优先级, 优先级设置的原则一是保证数据接收、处理、存储时各任务之间协同工作不发生错误, 二是根据任务的重要程度, 对系统运行越重要的任务优先级越高, 基于此设置该系统四个任务的优先级从高到低依次为: 网络数据接收任务、实时数据处理任务、原始数据存储任务、结果数据存储任务。

3 数据处理软件功能实现

3.1 网络数据接收任务

机载测试系统的采集器将采集的数据以组播的形式发送至测试网络, 待处理参数就在这些网络包中^[11], 采集器发送的网络包数据有如下两个特点:

1) 数据包之间的目的 IP 不一致, 即不同网络包的目的地址可能不一样;

2) 网络包数据的发送速率比较高, 目前在部分型号上, 机载测试系统发送的网络包数据的位速率一般可达 60~70 Mbits^[12]。

对网络数据的接收一般有两种方式, 一是通过普通套接字接收, 二是通过零缓存套接字接收。在 VxWorks 系统中, 网络层次包含四层: 应用程序层、协议接口层、多路复用层、链路层^[13]。如果使用普通套接字进行网络数据接收, 数据传输过程中就会使用 VxWorks 的各个网络层, 这样编程的优点在于层次分明, 程序的可移植性强, 但是缺点在于这样会导致数据在各层之间进行传输时会层层复制, 浪费了时间和空间资源^[14]。但是 VxWorks 提供了基于零拷贝技术的 zbuf 套接字, 使用该方法接收网络数据时, 传输过程中不需要进行复制, 数据可直接由链路层到应用层, 省去了时间以及空间的开销。理论上, 使用零拷贝技术后

网络数据接收速率可提高至少 50% 以上。网络数据接收任务的处理流程是：

- 1) 初始化网络接收任务，加载机载系统带头文件，系统相关的配置信息存在于系统带头文件中；
- 2) 读取带头文件中的数据包目的 IP 及端口号，将信息存储到相应的结构中；
- 3) 创建 zbuf 网络接收套接字，按照数据包的地址及端口，将套接字加入组播组；
- 4) 等待机载网络数据，当有数据到来时，循环接收网络数据，将所有的网络包数据放进原始数据缓存中，将需要处理的网络包数据放入数据处理缓存中。

原始数据缓存和数据处理缓存在系统中被设置为双缓存。以数据处理缓存为例，当网络数据接收模块接收到机载测试系统发送的数据时，先将数据放在缓冲区 1 中，当数据缓冲区 1 存满之后，通过数据处理信号量通知实时数据处理任务进行数据处理，然后将接下来的数据存入缓冲区 2 中，缓冲区 2 存满之后同样通过数据处理信号量通知数据处理任务，如此交替往复，保证数据接收的可靠性。

3.2 原始数据存储任务

为了方便在硬盘中存储网络数据，并且在飞行结束后通过卸载的方式将数据卸载下来，需要针对网络数据设计一种数据存储的文件格式，自定义的文件存储格式可以方便地进行数据的存储与卸载。根据实际数据的需求，设计图 3 所示的文件格式，该格式包括文件头和数据两个部分：文件头包含文件个数和每个文件的自有格式，包括：文件开始位置、结束位置、文件大小、开始时间、结束时间五个信息。数据存储格式如图 3 所示。

文件个数 N \				
开始位置	结束位置	文件大小	开始时间	结束时间
.....
开始位置	结束位置	文件大小	开始时间	结束时间
文件 1 数据				
.....				
文件 N 数据				

图 3 数据存储格式

在原始数据存储文件中，先是文件个数 N，然后是 N 个描述每个文件信息的文件头，最后是 N 个数据文件。根据实际数据存储的需求，一般将 N 设为 1 000。当原始数据存储任务接收到存储信号量时，开始进行数据存储，具体的方法是：

- 1) 读取文件头中的文件个数 N，并将文件个数加一；
- 2) 获取文件 N 的结束位置，并在该位置的基础上加一，作为新文件 N+1 的开始位置；
- 3) 获取当前系统时间，作为该文件的开始时间；

- 4) 写入数据时，分析写入数据的大小，每写入一次，更新文件大小、结束时间、结束位置；
- 5) 重复进行步骤 4) 直至系统运行结束。

3.3 实时数据处理任务

机载测试系统发送的数据包格式为网络格式，其数据结构如图 4 所示^[7]。网络数据中先是四个字节的控制域信息；然后是四个字节的包 ID，每一个网络流都有一个 ID 号，该 ID 号是该网络流的唯一标识；接下来是四个字节的序列号，对于每一个网络包数据，序列号从零开始逐渐往上增加，每增加一个网络包序列号加一；序列号之后是四个字节的包长，包长包含从控制域开始到网络包结束的所有字节数；后面是八个字节的包时间信息，网络包时间为 BCD 码格式，可以对此进行解码得到数据包采集时刻的时间。最后是网络包的数据，要处理的参数信息就包含在数据负载里面。

4 字节	4 字节	4 字节	4 字节	8 字节	0—65490 字节
控制域	包 ID	序列号	包长	包时间	数据

图 4 网络数据格式

实时数据处理任务一直等待数据处理信号量，当收到处理信号量时从数据处理缓存中读取网络包数据，根据包 ID 判断该包是否需要处理，若需要处理则对待处理的参数进行码值提取及物理量转化，最后将处理的参数值放入数据结果缓存中，实时数据处理任务的流程如图 5 所示。

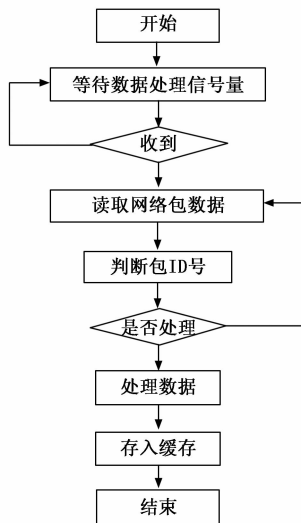


图 5 实时数据处理的流程

3.4 结果数据存储任务

由于 CPU 的处理速度远大于 I/O 的吞吐速度，如果不设置缓存，那么系统就会频繁的进行 I/O 操作，严重影响处理器的性能^[15]。所以需要设置缓存协调高速处理设备与低速存储设备之间的匹配问题。在实时数据处理与结果数据存储之间设置预留缓存，先将处理的中间结果存入预留缓存中，然后在缓存存满之后一次性写入硬盘，这会极大

地减少计算资源的浪费, 提高处理器的利用率。

结果数据存储任务用于将预留缓存中的数据写入到硬盘中, 该任务发起后等待存储信号量, 当预留缓存满之后释放存储信号量, 结果数据存储任务收到存储信号量之后将预留缓存中的数据写入到硬盘中。

具体实现时, 可以巧妙地利用 `fwrite()` 函数自带的缓冲区, 在系统初始化时将该函数的缓冲区设置为一秒钟结果数据的大小, 一般设为 1M。当结果数据缓冲区满时系统自行利用内部信号量的机制写入数据, 这样可有效避免频繁操作 I/O 和硬盘给处理性能带来不利的影响。

4 实验测试与验证

4.1 测试环境搭建

在 WorkBench3.3/VxWorks6.9 环境下实现了网络数据实时处理软件, 在某型号的机载实时数据处理设备上进行测试, 该设备的性能指标如表 1 所示。

表 1 机载实时数据处理设备性能

序号	器件名称	性能指标
1	主机架构	CPCI 3.0
2	CPU	Intel I7
3	内存	8G
4	硬盘	1T
5	网口	10/100/1000M

自适应机载实时数据处理软件需要测试三个方面的内容: 1) 是否能够以同等的位速率接收网络数据; 2) 数据存储格式是否正确且能够 100% 记录; 3) 数据处理的结果准确可靠, 满足课题要求。

以真实的机载网络数据为数据源, 编写机载网络数据发送软件。该软件利用毫秒级高精度定时, 以给定的位速率给机载实时数据处理设备发送网络数据; 在机载实时处理软件中以看门狗定时器的方式定位时间, 计算每秒钟接收的字节数进而计算位速率; 此外编写了数据卸载软件, 以读取硬盘的方式获取文件句柄并按照自定义的文件格式卸载数据。

4.2 测试结果分析

在实验测试时设定网络数据发送软件的位速率为 65 Mbps, 机载实时数据处理软件接收的位速率也是 65 Mbps, 表明网络数据接收模块能够正常运行。网络数据共发送了 1.98 G 的机载数据, 通过卸载软件能够正常卸载并且卸载的数据文件也是 1.98 G, 表明数据存储任务也能正常运行。将结果存储任务存储的结果文件拷贝出来, 分析其中的真空速参数并与实际的真空速进行比较, 发现该参数处理正确且能正常存储, 表明实时数据处理及结果数据存储任务能正常运行。真空速数值处理结果如图 6 所示。

实验结果表明, 基于 VxWorks 的机载实时数据处理软件能够在机载网络化测试环境下正常接收高速网络数据, 将数据按照自定义的文件格式 100% 写入硬盘中; 同时该软件可以保证数据处理的实时性及准确性, 在给定的位速率

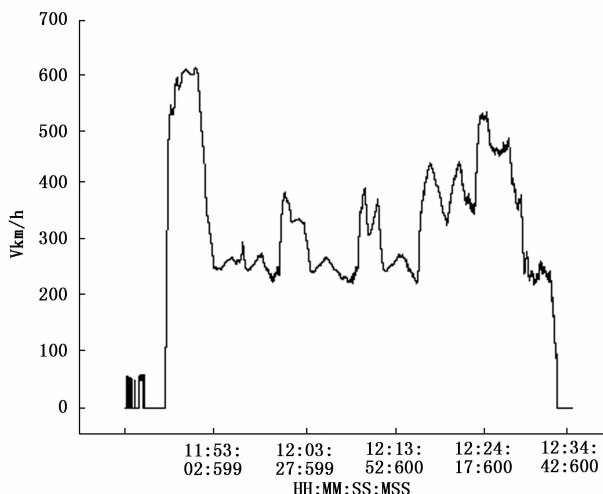


图 6 真空速处理结果

的条件下实时处理数据, 并将结果写入结果文件。该软件可满足机载实时数据处理的需求。

5 结束语

针对当前飞行试验对于机载实时数据处理软件的需求, 设计并实现了一种基于 VxWorks 的网络数据实时处理软件。该软件利用零拷贝技术通过网络的方式接收机载测试系统发送的高速网络包数据, 并将数据存储存储在自定义的文件中; 同时根据网络协议实时解析网络包数据, 将数据结果存储在结果文件中。为了提高软件运行的可靠性, 利用多核多任务分配的方法, 实现了多任务之间的协同工作以及多核之间的负载均衡; 此外在各任务之间设置了缓冲区, 保证了处理器的高效利用。实验测试结果表明, 该软件能实时接收、处理、存储数据, 并得到准确的处理结果。该软件的使用可大幅提高数据处理的效率, 为飞行试验数据处理提供了新的途径。

参考文献:

- [1] 霍朝晖, 覃杨森, 祁 春. 飞行试验机载关键参数快速处理系统设计 [J]. 现代电子技术, 2013 (5): 129-132.
- [2] 吴海东. 基于神经网络算法的飞行数据预测 [J]. 中国科技信息, 2015 (Z2): 41-43.
- [3] 霍朝晖. 关键参数快速处理单元在飞行试验中的应用 [J]. 计算机测量与控制, 2014 (6): 188-190.
- [4] 田方正, 王 锦. 机载网络化测试系统框架设计技术研究 [J]. 测控技术, 2013, 32 (4): 9-11.
- [5] 滕 飞. 基于网络化的机载测试系统关键技术研究 [D]. 西安: 西安电子科技大学, 2013.
- [6] 杨廷梧, 田宝泉. 飞行试验新型遥测机载网络化采集与记录系统架构 [J]. 2013 (5): 59-63.
- [7] 王建军, 彭国金, 陈惠君. 飞行试验多源 iNET 数据精密分析技术研究 [J]. 计算机测量与控制, 2016 (6): 248-250.
- [8] 霍朝晖, 魏建新, 覃 燕. 飞行试验机载网络数据采集与分析技术 [J]. 现代电子技术, 2011, 34 (11): 144-147.