

嵌入式新型机载实时数据处理系统的设计与实现

祁春, 聂瑶佳, 周训强

(中国飞行试验研究院, 西安 710089)

摘要: 为满足 C919 机上实时监控要求以及对试飞结果数据快速获取的迫切需求, 针对现今机载网络数据包大流量、高速率的特点, 通过对传统应用模式和系统结构的分析, 指出传统机载实时数据处理系统在功能和性能上的不足, 并提出了一种基于嵌入式结构、具有实时监控和重要参数快速处理复合功能的新型机载实时数据处理系统设计方法; 相比基于工控机的系统结构, 采用嵌入式结构和实时操作系统工作模式可极大提高系统多任务实时数据处理能力; 同时, 在系统中增加对重要参数在线处理和结果数据存储功能, 可使试飞工程师在任务结束后的便可拿到结果数据, 以解决以往地面数据处理等待时间长的难题; 实际应用表明, 新型系统很好满足了 C919 型号任务需求, 可显著加快试飞决策和提高试飞效率; 该设计方法也可为其它机载实时数据处理系统的设计提供方法借鉴。

关键词: 新型; 机载; 实时; 快速处理; 分组存储

Design and Implementation of a New Embedded Airborne Real-time Data Processing System

Qi Chun, Nie Yaojia, Zhou Xunqiang

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

Abstract: Aiming at the requirement of real-time monitoring and the urgent need for the rapid acquisition of results during the flight test of C919, in view of the characteristics of high flow and high rate of today's airborne network data packets, on the analysis of traditional application patterns and system structures, the shortcomings of the traditional airborne real-time data processing system in function and performance are pointed out and a new design method of airborne real-time data processing system based on embedded structure, real-time monitoring to process complex functions quickly is put forward. Compared with the system based on IPC, embedded structure and real-time operation system can greatly improve the system's multitask real-time data processing capability. At the same time, the online processing of important parameters and the storage of result data are added in the system, which makes the test engineer get the result data after the task immediately, so as to solve the problem of long waiting time for ground data processing. The practice shows that the new system meets the requirements of the C919's test, and can significantly accelerate the flight test decision and improve the efficiency of the flight test. This method provides a reference for the design of other airborne real-time data processing systems.

Keywords: late model; airborne; real-time; immediate processing; grouping storage

0 引言

飞行试验是在真实飞行条件下进行科学研究和产品鉴定试验的过程, 具有风险高、投入大、周期紧^[1]的特点。随着近年来新机及改型机的大量涌现, 以及新技术、新材料在新机上的应用, 使得试飞风险随之增加, 实时安全监控倍受各方的关注。而针对民用中大型飞机试飞, 通常利用飞机的空间优势, 建设机载实时数据处理系统以供试飞工程师、局方人员进行空中实时监控, 及时调整飞机状态^[2], 以提高试飞安全和质量, 这在我国 ARJ21-700 飞机的飞行试验中得到了充分验证与肯定。但随着机载测试系统全面网络化和新型航空总线的应用, 机载测试参数量也呈几何倍数增加, 从 ARJ21-700 到 C919, 测试参数从近 8 000 个增加到了近 40 000 个^[3], 网络数据包流

量更是高达 50 Mbps 以上, 传统的基于工控机服务器模式的机载实时数据处理系统无法满足如此高速率的实时数据处理要求。因此, 本文提出了一种新型机载实时数据处理系统的架构与实现方法, 不仅满足机上大流量、高速率网络数据实时监控需求, 更是在任务结束后便可立即对重点关注的参数进行结果快速检视, 有效解决了长期困扰的地面海量数据处理速率的瓶颈问题, 可为后续试飞科目的决策下达快速提供数据支持。该系统已成功应用于 C919 型号试飞, 在提高试飞安全、缩短试飞周期上发挥着重要作用。

1 机载实时数据处理系统

1.1 传统机载实时数据处理系统

传统的机载实时数据处理系统通常由实时数据处理单元、网络交换单元和实时监控单元组成^[4]。其中实时数据处理单元是整个系统的核心部分, 通常采用机载级工控机和桌面操作系统完成对机载网络数据包的实时接收、解包、分析和参数工程量转换, 并通过网络交换单元分发至各个监控显示单元来实现

收稿日期: 2017-11-06; 修回日期: 2017-12-15。

作者简介: 祁春(1983-), 男, 湖北黄冈人, 高级工程师, 主要从事飞行试验数据处理技术方向的研究。

参数的机上实时监控, 系统结构如图 1 所示。

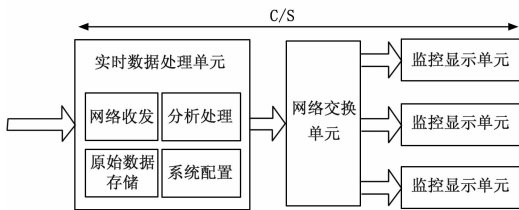


图 1 传统机载实时数据处理系统结构

传统的机载实时数据处理系统采用 C/S 架构实现监控显示单元与实时数据处理单元之间的数据通信, 因此需要在每台监控显示单元上部署监控软件。受工控机自身性能及操作系统资源调度限制, 对机载网络数据流的接收和实时处理能力一般不超过 40 Mbps, 由于以往机载测试参数相对较少, 机载网络数据包流量相对较低, 这种模式的机载实时数据处理系统还能够满足当时型号的需求, 但显然不具备对 C919 大流量、高速率机载网络数据包的处理能力。

1.2 嵌入式新型机载实时数据处理系统

ARINC664 等新型航空总线的应用和机载网络化测试技术的发展, 使得测试参数量成指数级增加, 机载网络数据流已超过 60 Mbps 甚至更高, 不仅对机载实时数据处理能力带来巨大挑战, 同时地面试飞工程师对数据结果快速获取的需求与海量数据处理时间长的现状形成尖锐矛盾, 俨然成为制约试飞效率的重要因素。为了解决这些问题, 本文提出了一种嵌入式新型的机载实时数据处理系统模式, 其系统结构如图 2 所示。

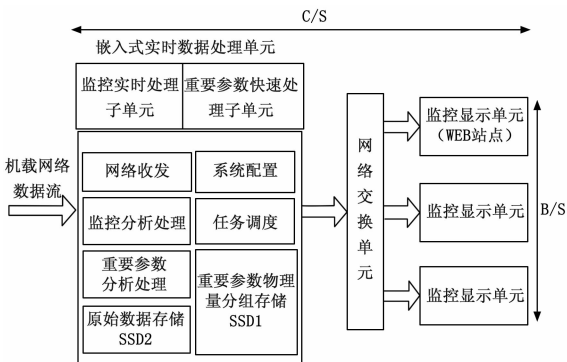


图 2 新型机载实时数据处理系统结构

嵌入式新型机载实时数据处理系统由嵌入式实时数据处理单元、网络交换机和监控显示单元组成。其中, 嵌入式实时数据处理单元是系统的核心, 也是在实时数据处理技术上的重要突破, 采用了基于 FPGA + DSP 架构的嵌入式多核处理器设备, 并使用 VxWorks 嵌入式操作系统作为其运行环境。采用嵌入式系统结构使得功能针对性更强, 可以充分利用嵌入式多核处理器资源和高速处理性能, 可极大提高对机载网络数据包实时接收和分析处理能力。

为了充分开发实时数据的价值, 解决以往地面数据处理等待时间长的难题, 将嵌入式实时数据处理单元的功能划分成两部分: 监控实时处理子单元和重要参数快速处理子单元。其中, 监控实时处理子单元主要根据机上实时监控需求, 对监控

参数进行预处理, 并将结果的工程量数据打包后通过网络分发至监控显示单元, 实现机上的实时监控; 重要参数快速处理单元则主要根据快速处理参数组, 对专业课题重点关注的参数进行实时在线处理, 并将工程量结果文件按照任务要求分组存储在 1 号固态硬盘, 试飞任务结束后便可立即将 SSD1 固态硬盘内容交给专业课题人员进行数据快速检视, 在短时间内可为下一架次的试飞决策提供数据支持, 省去了地面漫长的数据处理等待时间, 显著提高试飞效率。

同时, 整个系统采用了 C/S+B/S 的混合架构模式, 数据收发采用 C/S 方式以保证数据的实时性, 而在画面调度和管理上则采用 B/S 方式, 通过在其中一台监控显示单元上建立 WEB 站点, 实现对监控软件的集中管理。监控显示单元通过浏览器请求站点方式实现监控软件的调度与显示, 简化监控软件的部署和提高画面更新与维护效率。

嵌入式新型机载实时数据处理系统可实现不低于 110 Mbps 的机载网络数据接收及分析处理能力, 能够完成近 15 000 个测试参数 (32 点/秒) 的在线预处理和多采样率分组存储, 是对现有数据处理功能的重要扩充和能力提升。

1.3 新型系统处理流程设计

嵌入式新型机载实时数据处理系统同时完成机上实时参数监控和重要参数快速处理两种功能, 以满足现有型号任务及数据结果快速获取的需求, 其处理流程如下:

- 1) 加载试验工程文件, 获取数据包 KEY 值、长度、参数个数、采样率等信息; 获取参数名称、包中地址、有效位、字长等信息;
- 2) 加载参数组文件, 获取监控参数列表和快速处理参数列表。监控参数列表发送到监控实时处理子单元; 快速处理参数列表发送到重要参数快速处理子单元; 在两个子单元内部建立对应的参数索引和内存映射表;
- 3) 对机载网络数据包进行实时采集、解包、参数挑选和工程量转换, 并将采样率不同的数据作数据时间对齐处理后存入对应缓存;
- 4) 监控实时处理子单元按照要求数据更新率向监控显示单元发送监控参数工程量, 为试飞工程师提供机上实时监控;
- 5) 重要参数快速处理子单元按照要求的数据更新率以数据块形式进行在线存盘;
- 4) 和 5) 多线程并行进行。
- 6) 试飞任务未结束, 则跳转至 3), 否则结束。

2 系统软件设计与实现

硬件是软件行的平台, 软件是系统功能的具体实现, 两者相辅相成。嵌入式新型机载实时数据处理系统软件主要包括任务管控任务、网络数据实时处理软件、基于 B/S 的画面调度与管理软件、系统管理与维护软件。

2.1 任务管控软件

任务管控软件主要用于对机载测试系统中的采集器配置文件 (又称机载格式格栅文件) 节点内容进行解析, 获取机载采集器设置、网络数据包及参数的详细信息^[5-6], 包括数据包的 KEY 值、长度、参数个数、采样率等信息, 以及参数的名称、地址、有效位、字长等信息, 通过导入参数的校线、有效位等信息, 生成嵌入式实时数据处理单元识别的试验工程文件 (*

.hea) 和参数组文件 (*.par), 以此作为嵌入式实时数据处理单元进行监控参数处理和重要参数快速处理的依据性文件。任务管控软件功能组成如图 3 所示。

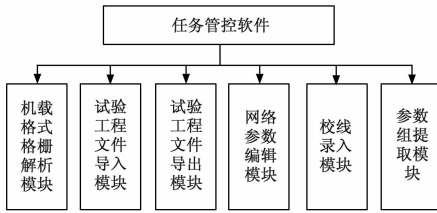


图 3 任务管控软件功能组成

- 1) 机载格式格栅解析模块：对机载格式格栅文件各节点进行分析，提取网络包、参数等信息，生成原始的试验工程文件；
- 2) 试验工程文件导入模块：用于导入工程文件，为参数查看或编辑做准备；
- 3) 参数编辑模块：实现对参数的名称、字长、地址、有效位等信息的编辑功能；
- 4) 校线录入：为参数的校线录入提供人机接口；
- 5) 参数组提取模块：从总的参数列表中提取部分参数形成一组或者多组文件，用于参数处理与结果分组。

2.2 网络数据实时处理软件

2.2.1 功能结构

网络数据实时处理软件是整个软件的核心，运行在嵌入式实时数据处理单元上，采用 VxWorks 嵌入式操作系统作为开发环境，依据任务管控软件生成的试验工程文件，完成对机载网络数据包进行实时接收、解包、监控参数提取、快速处理参数提取、工程量转换、网络分发和快速处理参数结果分组存储等。其功能组成如图 4 所示。

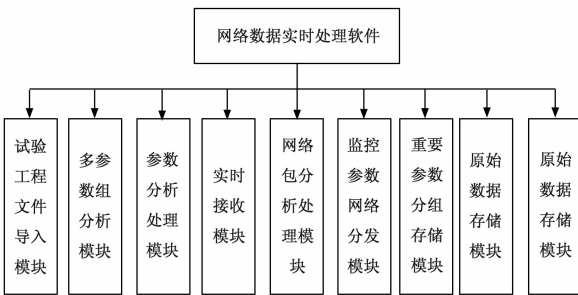


图 4 网络数据实时处理软件功能组成

2.2.2 工作流程

网络数据实时处理软件首先加载试验工程文件获取网络数据包信息(包 ID、包长度、参数个数、包采样率以及组播地址和端口号等)、参数信息(参数所在包 ID、参数名称、字长、校准类型、拼接和取位信息、校准值以及参数采样率等)和参数组信息(监控参数名、重要参数快速处理参数名及分组信息等)，并将信息综合后存放在内存的表结构中。然后创建监控实时处理环形缓存、原始数据存储环形缓存以及重要参数快速处理环形缓存，用于后续的数据处理和存储。

合理使用多线程技术可以有效提高程序的运行性能^[7]。网

络数据实时处理软件在获取处理信息及开辟缓存后，创建了实时数据接收线程、监控参数处理及快速处理线程、监控参数工程量转发线程、重要参数分组存储线程、原始数据存储线程以及系统配置及维护线程等，线程间通过信号量等多线程同步技术来进行线程协调运行。

同时，为了防止各线程同时对共有缓存区进行读写操作而出现不可预知的结果，采用关键代码段对共有缓存区进行保护，确保数据的可靠性和软件的稳定性。软件的处理流程如图 5 所示。

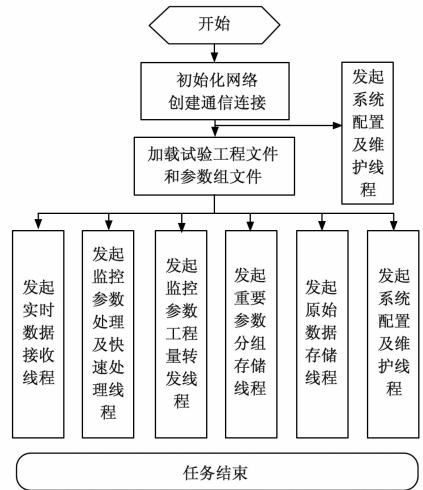


图 5 软件处理流程图

2.2.3 关键技术

在对机载网络数据包进行实时处理的整个过程中，数据接收、监控参数与重要参数快速处理是过程的核心部分，直接关系到系统功能的正常性和数据处理结果的正确性。数据接收、监控参数与重要参数快速处理的工作流程分别如图 6 和图 7 所示。在实时接收流程中，首先对接收到的机载网络数据包的 KEY 值进行判断，如果需要处理则将网络包推入处理数据包环形缓存中，当环形缓存达到满状态后，就释放处理信号量对网络包进行解析。同时对接收到网络包进行原始数据存储。

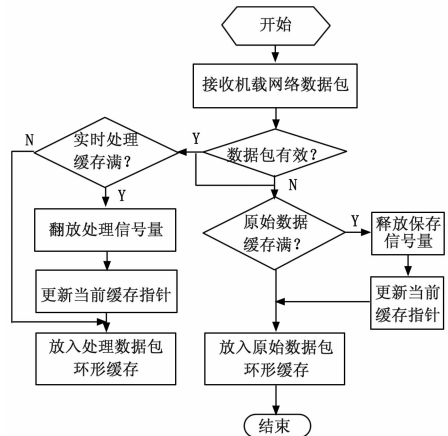


图 6 机载网络数据包接收流程图

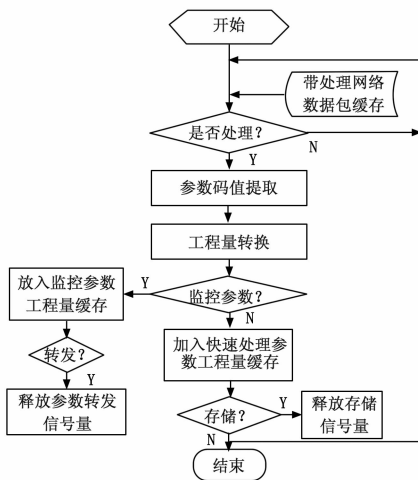


图7 监控参数与重要参数快速处理流程图

对于机载网络数据包的接收,需要在配置机载采集器时合理设置网络包的大小。包太小,将造成数据包的频率过高,在数据的接收和分析过程中容易丢包;包太大,每个包的参数过多,增加对参数的分析与处理时间,也会因处理不及时而丢包。对于不需要处理的小包数据,在数据接收前可以通过设置网络交换机进行过滤。

监控参数与重要参数快速处理线程得到信号量后,根据已获知的参数信息从网络包中进行提取和工程量转换,然后根据参数的性质(监控参数、快速处理参数)推入相应的缓存当中,当缓存达到满状态后,进行监控参数转发或者重要参数的存储。

环形缓存由16个大小为8MB的缓存组成,重要参数快速处理结果文件存储在SSD1固态硬盘,原始数据存储在SSD2固态硬盘。值得注意的是,在转发频率的控制上,软件采用的是频率参照法实现的,即预先选取2~3个频率相同的网络包作为参照,当接收到该网络包时就进行监控参数工程量转发,可以解决软件定时误差大、硬件定时中断处理繁琐的问题。选择2~3个同频率网络包作为数据发送参照,是为了防止某一参照包因自身丢包而造成缺失数据发送参照。

2.3 画面调度与管理软件

画面调度与管理软件采用B/S架构实现,将其中一台监控显示单元设计为WEB站点服务器,实现对监控画面的统一管理调用。WEB站点采用SQL Server数据库和APS.NET实现,监控计算机使用浏览器访问请求实现监控软件的加载和运行,可简化机上软件部署,提高任务更新与维护效率。

2.4 系统管理与维护软件

系统管理与维护软件主要为嵌入式实时数据处理单元提供可视化的配置与维护界面,以命令字的方式实现软件与实时数据处理单元间的交互。主要实现网络通信设置、任务设置、实时数据处理单元工作状态监视以及系统控制等。其中工作状态监视包括接收数据包速率(流量)及丢包统计、转发监控参数数据包流量及包速率、固态硬盘文件记录及磁盘空间信息等,便于对系统工作状态及时掌握。

3 试验结果与分析

嵌入式新型机载实时数据处理系统主要实现机上实时监控和对重要参数的在线快速处理。为了验证数据的正确性,将嵌入式新型机载实时数据处理系统生成的结果数据与地面对机载记录数据处理的结果数据进行对比分析,例如图8所示的某参数数据曲线图,数据结果值的误差在课题人员进行数据分析允许的范围之内,可以直接作为试飞任务结束后的快速数据分析和为下一次架次试飞决策提供数据支持。

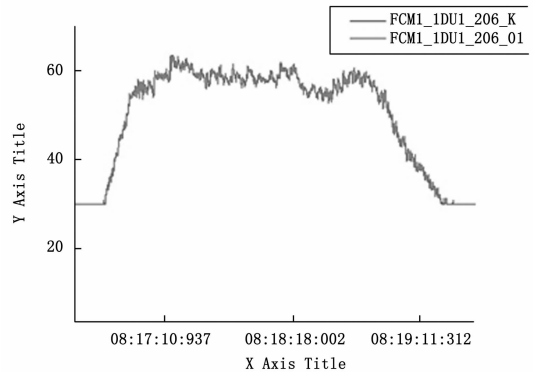


图8 数据结果对比图

4 结论

嵌入式新型机载实时数据处理系统以嵌入式实时数据处理单元作为数据处理核心,并在功能上分化为监控实时处理子单元和重要参数快速处理子单元两部分,既满足了机上对大流量、高速率机载网络数据包的实时处理与监控需求,又通过对重要参数的在线实时处理与分组存储功能,极大缓解了地面对海量数据处理的时间压力,实现了80%以上测试参数的在线预处理,试飞任务结束后便可将记录结果文件交给专业课题进行分析,能够在短时间内为下一架次试飞决策提供数据支持,进而提高试飞效率。新型机载实时数据处理系统已经成功应用于我国C919大型客机上,在保障试飞安全、提供试飞效率上发挥着重要作用。同时,本系统模式及实现方法也可为其其它机载实时数据处理系统的建设提供参考和借鉴。

参考文献:

- [1] 党怀义. 云技术在飞行试验数据处理中的应用[J]. 测控技术, 2014(3): 49-51.
- [2] 王彦庆, 陈彦强, 刘磊. 机载实时数据处理系统设计与实现[J]. 军民两用技术与产品, 2017(14): 79-81.
- [3] 戴卫兵, 王文丽. 网络技术在ARJ21试飞测试中的应用[J]. 测控技术, 2010, 29(12): 42-43.
- [4] 房瑾, 霍朝晖, 聂睿, 等. 机载网络数据实时处理软件的设计与实现[J]. 电子设计工程, 2015, 23(6): 52-54.
- [5] 李五一, 闫楚良, 赵丽娜. 飞行试验机载网络数据实时处理技术研究[J]. 计算机测量与控制, 2015, 23(10): 3515-3517.
- [6] 房瑾, 张国旺, 等. 民机机载实时处理系统的设计与实现[J]. 科技信息, 2014(5): 130-131.
- [7] 孙鑫, 余安萍. VC++深入详解[M]. 北京: 电子工业出版社, 2006.