

主机控制软件在测发控系统中的应用研究

于大海, 袁杰, 白亮, 向军, 丁景义

(上海航天电子技术研究所, 上海 201109)

摘要: 主机控制软件是运载火箭一体化测发控系统的关键环节, 测试流程可配置是主机控制软件的重要研究方向; 针对现役 CZ-2D、CZ-4B/C 一体化测发控系统主机软件测试流程不可配置、软件代码更改频繁、升级次数多、管理成本高等问题, 新一代主机控制软件以简易、可靠、快速可配置、界面友好为主要特征, 通过采用“抽象一级基础文件、拼装二级测试文件、独立发送器接收器、网络协议映射”的软件架构开发策略, 结合现役主机控制软件优点、新一代测发控软件系统要求、通信类型、接口定义等, 实现了新一代可配置主机控制软件, 为未来主机控制软件的研究打下了技术基础。

关键词: 可配置; 通用化; 主机控制软件; 一体化测发控系统

Application Research of Host Control Software in Integrated Measurement and Control System

Yu Dahai, Yuan Jie, Bai Liang, Xiang Jun, Ding Jingyi

(Shanghai Aerospace Electronic Technology Institute, Shanghai 201109, China)

Abstract: Host control software is the key link of the launch vehicle integrated measurement and control system, Test process configurability is an important research direction of host control software. For the current CZ-2D, CZ-4B/C integrated measurement and control system host control software test process is not configurable, software code changes frequently, upgrade several times, high management costs, etc. The new generation of host control software is simple, reliable and fast Configurable, user-friendly features, Through the use of abstract first-level basic data files, assembled secondary test files, independent transmitter receivers, network protocol mapping software architecture development strategy, combined with the advantages of active host control software, a new generation of measurement and control system requirements, communication types, interface definitions etc, realized a new generation of host control software and laid the technical foundation for the future research of host control software.

Keywords: configurable; generalization; host control software; integrated measurement and control system

0 引言

运载火箭一体化地面测试发射控制系统(简称一体化测发控系统)是现役运载火箭地面测试系统的重要组成部分^[1], 如图1所示。负责完成运载火箭总装厂集成综合试验、发射场试验^[2]。通过一体化测发控系统完成对火箭控制系统、推力系统、遥测系统、外安系统、动力系统等五大系统的控制^[3], 完成信号采样、信号诊断、信号激励、流程控制、点火发射、生成判读报告等。该系统采用集中控制、分布式测试模式, 共同完成一套测试任务^[4]。

主机控制软件是一体化测发控系统软件分系统的重要组成部分, 是所有控制指令的发起端, 也是有回令属性指令的闭合端, 负责完成测试流程文件、控制指令文件创建、流程控制、数据诊断、报表生成等操作。因现役主机控制软件测试流程不可配置, 且测试流程固化在软件代码中, 导致软件代码更改频繁、升级次数多、管理成本高等问题发生, 固新研新一代主机控制软件(简称新型主机控制软件), 该软件有效地将试验过程信息以文件流的形式与软件

架构相分离, 降低了软件模块间耦合度, 提升了软件通用性, 将该软件功能最终转化为测试流程控制器, 通过输入测试流程文件, 最终实现对一体化测发系统的控制。

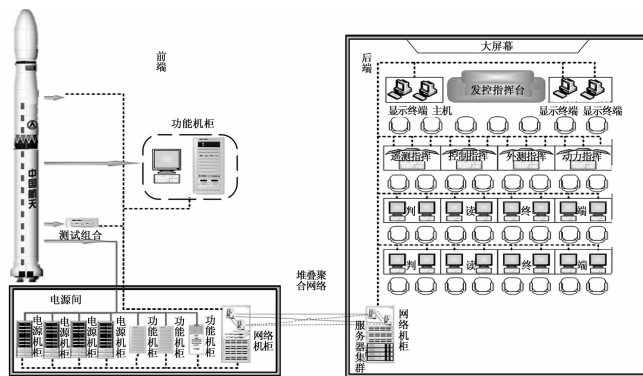


图1 一体化测发控系统框图

1 新型主机控制软件研制的必要性

1.1 现役一体化测发控系统主机控制软件现状分析

软件架构设计不合理。现役一体化测发控系统主机控制软件采用在软件代码中固定不同类型编码顺序, 形成不同测试函数模块, 通过用户界面选择, 执行不同函数功能模块分支, 完成测试流程, 实现对一体化测发系统的控

收稿日期:2018-09-27; 修回日期:2018-10-22。

作者简介: 于大海(1986-), 男, 山东黄县人, 硕士, 工程师, 主要从事运载火箭地面一体化测试发射控制系统技术方向的研究。

制。这种定制化软件架构, 对未来软件需求发生变更方面适应性较差, 软件架构不合理, 没有将软件功能模块、接口定义、模块间从属关系划分清晰, 当一级基础文件编码发生变更、或增加、缩减控制指令、测试流程顺序发生调整时, 必须更改软件代码, 升级软件版本, 验证测试后, 才能符合用户需求。这种实现方式导致代码功能模块耦合度高、后期维护管理成本高, 无法满足用户多样性需求。

通用化设计欠缺。由于软件架构设计的不合理、网络通信协议不兼容, 导致不同型号运载火箭都必须独自开发与之型号相适应的主机控制软件, 进而导致各型号间通用化程度低, 实现方式原理各不相同, 研制、维护成本增加, 人员利用率低。

1.2 新型主机控制软件研制方向

通用化是新型主机控制软件研制的必然发展方向^[5], 只有实现最大限度的通用, 才能有效降低人力成本、维护成本、缩短相同类型软件的研发周期, 以最小的更改适应更广阔的市场变化^[6]。

三层 C/S 结构提供了一种思路, 即分离出业务处理规则, 使之独立出来, 这种独立的特点使得一个领域内最容易导致系统无法复用的部分变得相对容易修改^[7]。新型主机控制软件三层结构如图 2 所示。

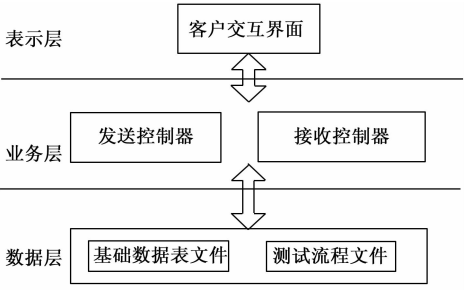


图 2 新型主机控制软件 C/S 三层体系结构

新型主机控制软件与现役主机控制软件相比, 具有如下特点:

- 1) 业务层与数据层更好的分离。业务层各功能模块逻辑关系划分清晰, 耦合度低; 数据层以 XML 文件的形式作为软件的输入, 该模式可有效地提升软件通用性;
- 2) 测试流程文件以 XML 文件形式与软件进行剥离, 并可根据用户需求实时调整测试流程或基础数据表文件, 适应变化;
- 3) 增加标签概念及标签处理, 可有效提升软件功能性;
- 4) 软件结构更加模块化, 边界清晰, 降低模块间耦合度;
- 5) 以 XML 文件形式保存新型主机控制软件数据层内容, 可通过调整文件属性列方式, 满足用户多样性需求。

2 新型主机控制软件设计方法

2.1 总体结构

新型主机控制软件由基础数据表模块、测试流程模块、

接口协议转换模块、发送控制器模块、接收控制器模块、单步测试模块组成, 其中, 发送控制器包括 SOCKET 发送控制器和 COM 发送控制器; 接收控制器包括 SOCKET 接收控制器和 COM 接收控制器, 分别用于完成对网络信号和串口信号控制; 一级基础文件通过与一体化测发控系统不同功能子系统进行一一对应, 依托功能号进行不同编码设计, 生成不同功能一级基础指令编码表文件; 测试流程文件依据任务需求通过不同基础数据表文件中的控制编码指令拼装而成, 并以不同名称独立 XML 文件形式保存。试验过程中, 发送控制器对测试流程文件内容进行解析, 将当前步骤中解析好的指令数组传送给 SOCKET 功能模块或 COM 功能模块, 依托通讯模块返回结果控制任务流程进度, 显示执行结果, 提示错误信息, 等待用户选择。SOCKET 功能模块或 COM 功能模块最终实现控制指令的发送与接收, 并对接收到的返回指令进行解析、判定, 将判定结果返回给发送控制器, 辅助测试流程控制, 软件结构如图 3 所示。

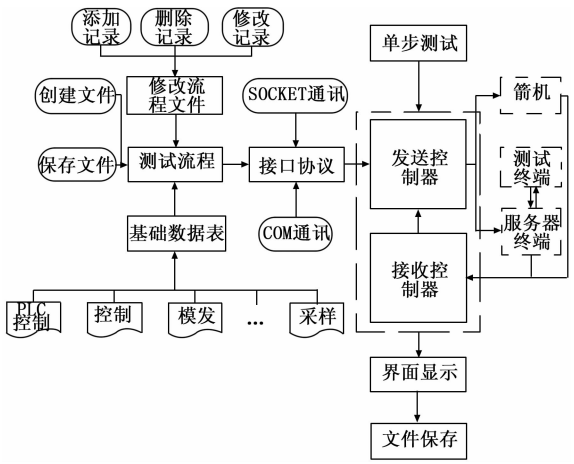


图 3 新一代主机控制软件系统结构

2.2 基础数据表模块设计原理

基础数据表功能模块由基础数据表格信息和网络信息构成, 主要用于完成一级基础文件的创建、删除、保存; 文件记录的增加、删除、更改、搜索、移动等操作。一级基础文件通过对一体化测发控系统不同功能子系统进行编码设计, 生成具有不同功能号的一级基础指令编码表文件^[8-11]。

1) 基础数据表格信息定义如下六元结构:

(编码, 名称, 标准值, 偏差值, 计量单位, 备注)

编码: 同一基础表文件中不同控制指令的唯一数字标识;

名称: 同一基础表文件中不同控制指令的唯一字符串标识;

标准值: 同一基础表文件中不同控制指令的理论值;

偏差值: 同一基础表文件中不同控制指令的误差值;

计量单位: 单位属性。例如, V, mA, A 等;

备注: 字符串标识, 用来记录指令的含义或其他;

2) 网络信息如图 4 所示, 定义如下九元结构, 图片中其他信息作为后续拓展功能使用。

(文件属性、信源、信宿、功能号、帧标识、IP、IP-Port、数值类型、通讯方式)

文件属性: 分发送和接收两种属性。发送表明该基础数据表文件为指令控制文件; 接收表明当接收到该文件中的指令时用于显示, 不进行判断;

信源: 表示控制指令的发送方;

信宿: 表示控制指令的接收方;

功能号: 基础数据表文件的唯一标识, 不同基础数据表文件具有不同功能号;

帧标识: 用于标识此帧为命令帧或为回令帧; 若为命令帧, 主机控制软件只需要发送此帧, 没有回令; 若为回令帧表明发送完此帧后, 还需对随后返回的回令帧进行判别处理。

IP: 网络接收方 IP;

IPPort: 网络接收方网络端口;

数值类型: 表示基础数据表中标准值与偏差值得类型, 分为字符串和数值型两种。在数值型基础数据表文件修改时, 需要对每条编码指令对应的标准值与偏差值的合法性进行判别;

通讯方式: 分 TCP 通讯与 COM 通讯两种。TCP 用于完成网络通讯, COM 通讯用于完成串口通讯, 未来可根据新型主机控制软件应用场景对通讯方式进行扩充;

基础数据表信息		网络信息	
文件名称:	COM控制.xml	文件属性:	发送
帧标识:	0000	功能号:	222
信源系统编码:	4	信宿节点:	3
信宿系统编码:	4	信宿节点:	422
帧标识:	00	数值类型:	字符串
通讯方式:	COM		

图 4 网络信息

依托基础数据表功能模块可以创建多项事先按功能号划分的一级基础数据表文件, 以 XML 形式保存。文件中编码与名称必须唯一, 在保存过程中会自动校验, 对异常记录信息进行报错提示。一级基础数据表文件中的控制指令是二级测试流程文件的重要组成部分, 依靠功能号与控制指令编码实现对终端的控制。

2.3 测试流程模块设计原理

二级测试流程文件由来自不同一级基础数据表文件中的控制指令及标签功能码按一定顺序排列而成。测试流程模块负责完成测试流程文件的创建、修改、保存等工作, 具体操作流程如图 5 所示。

测试流程模块执行过程描述如下:

1) 创建测试流程文件, 输入新测试流程文件名字, 生成内容为空的 XML 文件;

2) 从一级基础数据表文件中选择控制指令, 单击“添加记录”按钮, 将已经选择的控制指令添加到测试流程信息列表中。添加过程中, 既可以选择单项, 也可以选择多

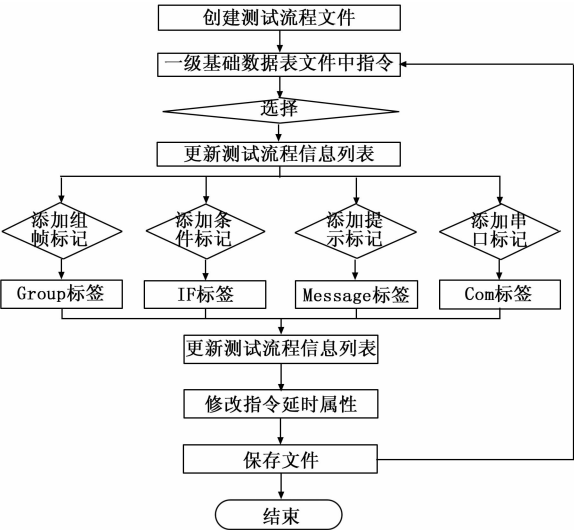


图 5 测试流程模块工作流程

项一起添加;

3) 对已经生成的测试流程信息列表进行添加标签操作。标签分为 Group (组帧) 标签、IF (条件) 标签、Message (提示) 标签、Com (串口) 标签 4 种类别。其中 Group 标签用于将来自同一基础数据表文件的多条连续指令打包为一个组, 一帧发送; IF 标签用于将一种信号或多种信号作为判别条件, 当条件满足时执行特殊代码段; Message 标签用于在特定时间点完成弹框提示功能, 同时可供用户选择继续当前测试流程或者退出当前测试流程; Com 标签用于添加 Com 指令, 可对 COM 指令内容按用户需求设定;

4) 指令延时属性用来标记两条连续指令间执行时间间隔, 满足对一些采样信息的特定时间延时需求;

5) 当测试流程信息列表中所有测试信息已经修改完毕时, 单击保存按钮, 将测试流程信息列表中所有信息保存至已经生成的相应 XML 文件中;

2.4 处理单元设计原理

处理单元负责完成自动测试过程中二级测试流程文件指令解析、标签识别处理、流程控制, 将解析好的指令数组传送通讯模块, 通讯模块完成指令的发送与接收、指令比对校验, 并将处理结果反馈给处理单元, 是新型主机控制软件的调度中心单元。SOCKET 功能模块包含 SOCKET 发送控制器和 SOCKET 接收控制器; COM 功能模块包含 COM 发送控制器和 COM 接收控制器;

发送控制器负责完成指令识别、指令处理、界面更新、流程进度控制等; 接收控制器负责完成指令接收、指令校验、合法性判定、同步互斥变量状态设置等;

同步互斥变量负责协同发送控制器与接收控制器同步工作, 并携带接收控制器判读结果, 发送控制器通过对同步互斥变量进行判断, 依托不同测试结果执行不同程序代码分支。

处理单元执行过程描述如下:

1) 发送控制器判断指令队列是否不为空, 若不为空,

