航天器自动化测试与航天器测试语言

孙 波 马世龙 余 丹

(北京航空航天大学 软件开发环境国家重点实验室, 北京 100191)

摘 要: 通过对航天器自动化测试需求的分析,总结出面向航天器自动化测试语言应具备的特征;通过对航天器测试业务中测试操作、测试流程、测试层次和体系结构的分析,给出了航天器测试语言的两种类型的语句,常规测试语句和专用测试语句,并将航天器专用测试语句分为操作类语句、数据类语句、时间类语句和控制类语句,给出了这4种专用测试语句的定义及其形式化描述. 该语言对提高我国航天器测试业务的效率和测试人员的工作效率、促进航天器测试自动化研究的发展将起到一定的推动作用.

关 键 词: 航天器; 自动化测试; 测试语言; 常规测试语句; 专用测试语句

中图分类号: TP 273^{+}. 5

文献标识码: A **文 章 编 号**: 1001-5965(2009)11-1375-04

Spacecraft automatic test and spacecraft test language

Sun Bo Ma Shilong Yu Dan

(State Key Laboratory of Software Development Environment, Beijing University of Aeronautics and Astronautics, Beijing 100191, China)

Abstract: By the analysis of the spacecraft automatic test demands, the characteristics of test language for spacecraft automatic test were summed up, and two types of statements of spacecraft test language, that is, the regular test statements and the special test statements were given based on the analysis of test operations, test processes and test architecture. In addition, in the spacecraft test language the special test statements were classified into four kinds of statements, including operation statements, data statements, time statements and control statements. The definitions and their formal descriptions of the four kinds of the special test statements were presented in detail. This language will improve the efficiency of spacecraft test business and testers of our country, and promote the researches on spacecraft automatic test.

Key words: spacecraft; automatic testing; test language; regular test statement; special test statement

为适应不同用途航天器测试的需求,缩短研制周期,各国航天界已经或正在开发通用的航天器测试系统. 根据卫星总装和在轨测控系统软件设计开发中的经验,以及对用户在实际使用中的技术支持及系统维护,具备以下8条功能的系统可以认为是功能齐全的系统: ①多进程多平台分布式运行机制; ②多平台多操作系统可移植性; ③高度可配置功能模块; ④功能的可扩展性,可重用性; ⑤完备的测试语言; ⑥归档储存及回放运行; ⑦集中多用途数据平台; ⑧图形化的人机交互界面.

在上述功能中,设计并构建完备的测试语言是成为通用测试系统中极为关键的一项[1-2].

航天器测试语言是用于航天器综合测试的计算机语言,由于它对提高航天器测试的自动化水平具有重要的意义,世界各航天大国都在对其进行研究,而其发展也是随着计算机水平的提高以及软件的发展而向前发展的[3].

本文通过对航天器自动化测试需求的分析,总结出面向航天器自动化测试的语言所应具备的特征,并给出了相应航天器测试语言中的测试语句及其形式化定义.

收稿日期: 2008-10-28

基金项目: 国家973资助项目(G2005CB321902); 软件开发环境国家重点实验室探索性自主研究课题资助项目(SKLSDE-2009ZX-13)

作者简介: 孙 波(1974-), 男, 北京人, 博士生, sunboo2002@sina.com.

1 航天器自动化测试系统层次结构

在体系结构上,航天器自动化测试系统可分为 3个层次,通过对每个层次进行结构与功能优化,明确各软件的部署情况,可以建立起自动化测试的协同工作环境.

航天器自动化测试系统结构如图 1所示,各层次的主要功能如下:

第 1层为测试资源层,部署与各分系统设备相关的仪器服务器.这些服务器连接通用或专用的测试设备,接收并处理上层发送过来的相关命令与指令,根据命令和指令管理和操作底层设备,同时接收各专用测试设备采集的卫星测试数据信息与指令执行信息,将这些结果数据与信息返回至上层.所有对设备的底层操作均在这一层完成.

第 2层为测试支持层,部属总控服务器.总控服务器主要完成对测试资源层提供的测试数据与测试指令的转发服务,同时完成测试数据的自动存储及测试过程的监视.所有测试应用层软件只需要连接总控服务器就可以发送遥控指令、获得遥测数据并控制和管理测试设备.

第 3层为测试应用层,航天器自动化测试的应用流程分为 3个阶段的工作,即测试准备阶段、测试运行阶段,测试评估阶段,分别完成对应测试阶段的测试工作.因此,航天器自动化测试系统主要以测试数据资源为核心,以测试计划为测试过程的组织原则,各测试阶段应用子系统相互配合完成整个测试过程的组织与实施[4].

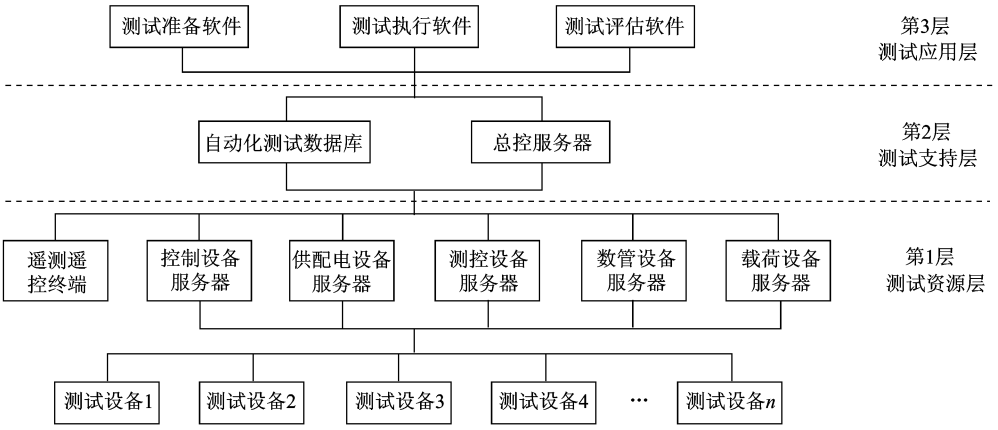

图 1 航天器自动化测试系统结构

2 航天器测试语言

航天器测试语言是支撑航天器自动化测试的形式体系及航天器测试工作标准.测试人员通过航天器测试语言,利用测试准备软件编写出测试程序,然后将其交给测试执行软件运行,并对运行结果进行处理与评估,就可实现测试程序自动执行.

航天器测试语言类似于高级编程语言,测试人员可以方便直观地用来编写测试程序.因此除了具备与一般高级编程语言相同的语句指令和执行控制流外,测试语言还必须能读取遥测参数及遥控命令数据,共享并交换图像显示功能模块的数据,前者可以方便对遥测参数监控,对遥控命令内容赋值以及发送,后者则直接将数据作图像或数字显示.

航天器测试语言和高级编程语言相似,但又有其特殊性.由于航天器测试语言是面向航天器的分系统和测试专家,而不是程序员.与高级编程语言相比,应该在语言词法、语法和语句结构上具有航天器测试业务上的直观性和测试操作上的简便性[5].

2.1 航天器测试语言的语句类型

航天器测试语言主要包括 2种类型的语句:

1) 常规测试语句.这类语句类似于通常的过程语言,句法要简化到覆盖以下范围:全局变量区、声明区、主编码区;赋值和表达式计算;程序流控制;通用数学函数;时间函数、文件存取;过程调用;绘图语句等.

2) 专用测试语句.这类语句应易于维护和扩充对航天器综合测试系统发出命令、数据交换和设备管理等功能,如:数据归档、授权发送遥控或激励命令、连接设备、断开设备连接、改变参数期望值、禁止监视、获取参数原码、获取遥测包原码、遥控命令、激励命令、执行控制等.

2.2 航天器专用测试语句

航天器专用测试语句可以分为以下 4类:操作类语句、数据类语句、时间类语句和控制类语句.

2.2.1 操作类语句

1) 遥控指令语句

①语句 Lstc

功能:发送离散遥控指令

范式:

Lstc ::= “lstc” <Lsid> “ ” <Rt>

Lsid ::= letter <letter|digit> +

Rt::= digit <digit> +

解释:Lsid表示离散指令号; Rt表示等待回令时间

②语句 Bltc

功能:发送带比例字参数遥控指令

范式:

Bltc::= “bltc ” <Blid> “ ” <Blz> “ ” <Pc> “ ”<Num> “ ”<Period> “ ”<Width> “ ”<Rt>

Blid::= letter <letter|digit> +

Blz::= (“00000”|(<digit> <digit> <digit> <digit> <digit>))

Pc::= (“0”|“1”)

Num::= (“1”|(<digit> <digit> <digit>))

Period::= (“0”| <digit> <digit> <digit> <digit>)

Width::= (“0”| <digit> <digit> <digit> <digit>)

Rt::= <digit> +

解释:Blid表示比例指令号; Blz为比例字; Pc为脉冲选择; Num为个数; Period为周期; Width为宽度; Rt为等待回令时间

③语句 Kutc

功能:发送数据块遥控指令

范式:

Kutc::= “kutc ” <Zlid> “ ” <Zldes> “ ” <Plist> “ ” <Sbid> “ ” <Zlbn> “ ” <Rt>

Zlid::= letter <letter|digit> +

Zldes::= <letter|digit> +

Plist::= (letter <letter|digit> +) {“ ”letter <letter|digit> +}

Sbid::= (<digit> <digit> <digit> <digit>)

Zlbn::= (<digit> <digit> <digit> <digit>)

Rt::= <digit> +

解释:Zlid为指令号; Zldes是指令描述; Plist为参数列表; Sbid为起始块号; Zlbn为指令块数; Rt为等待回令时间

2) 设备操作语句

①语句 Setp

功能:发送通用测试设备设置命令字

范式:

Setp::= “setp ” <Tdtsid> “ ” <Tdid> “ ” <StrCommand>

Tdtsid::= letter <letter|digit> +

Tdid::= letter <letter|digit> +

StrCommand::= letter <letter|digit> +

解释:Tdtsid表示测试设备转发服务器标识; Tdid表示测试设备标识; StrCommand表示命令字,使用 SCPI标准格式字符串

②语句 Setk

功能:发送专用测试设备设置命令字

范式:

Setk::=“setk ” <Tdtsid> “ ” <StrCommand>

Tdtsid::= letter <letter|digit> +

StrCommand::= letter <letter|digit> +

解释:Tdtsid表示测试设备转发服务器标识; StrCommand表示命令字,是使用 SCPI标准格式字符串

3) 连接操作语句

①语句 Connect

功能:连接某个测试设备

范式:

Connect::= “connect ”“ ”<Did>

Did::= letter <letter|digit> +

解释:Did为设备号

②语句 Disconnect

功能:断开某个测试设备

范式:

Disconnect::= “disconnect ”“ ”<Did>

Did::= letter <letter|digit> +

解释:Did为设备号

2.2.2 数据类语句

1) 遥测数据语句

①语句 Getvalue

功能:获得参数遥测工程值,返回字符型变量

范式:

Getvalue::= <Variable> “ = ”“getvalue ” “ ”<Pid>

Variable::= letter <letter|digit> +

Pid::= <letter> + <digit> +

解释:Variable为返回值标识; Pid为参数代号

②语句 Getrawvalue

功能:获得参数的遥测源码值,返回字符型变量

范式:

Getrawvalue::= <Variable>“=”“getrawvalue”“”<Pid>

Variable::= letter <letter|digit> +

Pid::= <letter> + <digit> +

解释:Variable表示返回值标识; Pid表示参数代号

③语句 Getframevalue

功能:获得整格式遥测源码值,返回字符型变量

范式:

Getframevalue::= <Variable>“=”“getframevalue”“” <Tpid|Tfid>

Variable::= letter <letter|digit> +

Tpid::= <digit> +

Tfid::= <digit> +

解释:Variable表示返回值标识; Tpid表示的是遥测包号; Tfid表示的是遥测帧号

④语句 Verify

功能:判别参数在上下限范围内,返回布尔变量值

范式:

Verify::= “verify”“” <Pvalue>“” <Up>“” <Dp>

Pvalue::= letter <letter|digit> +

Up::= <digit> +

Dp::= <digit> +

解释: Pvalue表示参数值; Up表示上限; Dp表示下限

2) 设备数据语句

①语句 Getp

功能:获得通用测试设备数据

范式:

Getp::=“getp”“” <Tdtsid>“” <Tdid>“” <Floatarray|Binaryarray>

Tdtsid::= letter <letter|digit> +

Tdid::= letter <letter|digit> +

Floatarray::= letter <letter|digit> +

Binaryarray::= letter <letter|digit> +

解释: Tdtsid表示测试设备转发服务器标识; Tdid表示测试设备标识; Floatarray | Binaryarray表示浮点数组型或二进制型数组标识

②语句 Getk

功能:获得专用测试设备数据

范式:

Getk::=“getk” <Dataid>

Dataid::= letter <letter|digit> +

解释: Dataid表示数据标识

2.2.3 时间类语句

1) 等待语句 Wait

功能:等待某段时间

范式:

Wait::=“wait”“” <secnum>“s”

secnum::= <digit> +

解释: secnum为等待的时间

2) 时间条件语句 Until

功能:直到某规定时间或满足某条件后继续执行

范式:

Until::=“Until”“” <secnum>“s”| <atime>|condition

secnum::= <digit> +

atime::=“0”<digit>|“1”<digit>|“2”(“0”|“1”|“2”|“3”)“h”“:”(“0”|“1”|“2”|“3”|“4”|“5”) <digit>“m”“:”(“0”|“1”|“2”|“3”|“4”|“5”) <digit>“s”

condition::= expression statement

解释: secnum表示相对时间; atime标识绝对时间; condition表示需要满足的条件

2.2.4 控制类语句

1) 执行语句 Perform

功能:执行一个或多个测试程序

范式:

Perform::=“Perform”“” <pid> +

pid::= letter <letter|digit> +

解释: pid为程序标识

2) 开始语句 Start

功能:启动某测试程序

范式:

Start::=“Start”“” <pid>

pid::= letter <letter|digit> +

3) 暂停语句 Halt

功能:暂停某测试程序执行

范式:

Halt::=“halt”“” <pid>

pid::= letter <letter|digit> +

(下转第 1407页)

lindrical tank with elastic spacer [J]. Communications in Nonlinear Science & Numerical Simulations, 1996, 1(2): 66 - 69

[4] 陈铭年,林永智,徐建全. 椭圆矩形油罐车转弯横向稳定性的计算分析 [J]. 专用汽车, 2003, 5: 3 - 5
Chen Mingnian, Lin Yongzhi, Xu Jianquan. Calculate analysis of transversal stabilization for turning of oil tank truck with irregular elliptical section [J]. Special Purpose Vehicle, 2003, 5: 3 - 5 (in Chinese)

[5] 陈志伟. 移动式压力容器介质晃动数值模拟及防波装置研究 [D]. 浙江:浙江大学材料与化学工程学院, 2006
Chen Zhiwei, Numerical simulation of liquid sloshing in transportable pressure vessel and research on the baffles [D]. Zhejiang: School of Material and Chemical Engineering, Zhejiang University, 2006 (in Chinese)

[6] 李松,高芳清,杨翊仁,等. 液体晃动有限元模态分析及试验研究 [J]. 核动力工程, 2007, 28(4): 54 - 57
Li Song, Gao Fangqing, Yang Yiren, et al. Finite element modal analysis and dynamic experimental for liquid sloshing [J]. Nuclear Power Engineering, 2007, 28(4): 54 - 57 (in Chinese)

[7] 刘奎,康宁. 罐车制动时液体晃动的仿真分析 [J]. 北京航空航天大学学报, 2009, 35(7): 799 - 803
Liu Kui, Kang Ning. Simulation of liquid slosh in braking process of tank truck [J]. Journal of Beijing University of Aeronautics and Astronautics, 2009, 35(7): 799 - 803 (in Chinese)

(编　辑:张　嵘)

(上接第 1378页)

4) 结束语句 End
功能:停止某测试程序
范式:
End::= “end”“ ”<pid>
pid::= letter <letter|digit> +

5) 继续执行语句 Continue
功能:继续执行某暂停的程序
范式:
Continue ::= “ continue ”“ ”<pid>
pid::= letter <letter|digit> +

6) 调用语句 Call
功能:调用某测试子程序
范式:
Call::= “call ”“ ”<pid>
pid::= letter <letter|digit> +

3 结　论

航天器测试语言是支撑航天器自动化测试的形式体系及航天器测试工作标准. ①本文总结出面向航天器自动化测试语言应具备的特征; ②通过对航天器测试业务中测试操作、测试流程、测试层次和体系结构的分析,给出了航天器测试语言中专用测试语句的定义及其形式化描述. 该航天器测试语言的实际试用表明,对提高航天器测试业务的效率和测试人员的工作效率、促进航天器测试自动化研究的发展起到一定的推动作用.

参考文献 (References)

[1] Clancy D A, Clyde B A, Mirantes M A. The evolution of a test process for spacecraft software [C]//Proc of the 2nd IEEE International Conference on Space Mission Challenges for Information Technology. Washington, DC, USA: IEEE Computer Society, 2006

[2] Yamada T. Generic software for spacecraft testing and operations based on a spacecraft model [C]//Proc of Space Ops2006. USA: American Institute of Aeronautics and Astronautics, 2006

[3] 王庆成. 航天器电测技术 [M]. 北京:中国科学技术出版社, 2007
Wang Qingcheng. Electrical test technology of spacecraft [M]. Beijing: Chinese S & T Press, 2007 (in Chinese)

[4] 余丹,叶钢,李先军,等. 基于工作流的航天器测试信息化系统的建模与分析 [J]. 武汉大学学报, 2009, 55(2): 211 - 215
Yu Dan, Ye Gang, Li Xianjun, et al. Modelling and analysis of spacecraft testing informatization system based on workflow [J]. Journal Wuhan University, 2009, 55(2): 211 - 215 (in Chinese)

[5] 王宪文. 航天器测试控制语言设计与实现方法研究 [J]. 航天器工程, 2007, 16(2): 57 - 63
Wang Xianwen. Design and realization method of the test and control language for spacecraft overall checkout [J]. Spacecraft Engineering, 2007, 16(2): 57 - 63 (in Chinese)

(编　辑:赵海容)