

基于命令驱动的通用测试系统研制

朱望纯[†], 陈玉霞

(桂林电子科技大学电子工程与自动化学院, 广西 桂林 541004)

摘要:研制了一套武器电子组件产品通用测试系统。通过二次开发可以针对不同的测试任务进行测试界面和测试流程自定义,减少硬件资源堆砌,缩短测试开发周期。使用开放式、可扩展/剪裁的整体架构搭建硬件平台,利用虚拟仪器技术、脚本技术、编程语言技术以及分层软件架构完成通用测试系统软件的设计与开发。设计了满足测试开发需求的功能命令集和脚本开发工具,构建了较为完备的命令驱动体系,达到了测试流程自定义和缩短测试程序开发周期的目的。经过不同的实验验证,通用测试系统可以完成不同电子组件的测试任务,通用性、灵活性、可扩展性较好,可长期适用于军工厂中各种电子组件产品的测试。

关键词:通用测试;二次开发;虚拟仪器;命令驱动;脚本

中图分类号:TP311.521

文献标识码:A

Development of General Test System Based on Command Drive

ZHU Wangchun[†], CHEN Yuxia

(School of Computer Science and Engineering, Guilin University of Electronic Technology,
Guilin, Guangxi 541004, China)

Abstract: A general test system for weapon electronic component products is developed. Through secondary development, the test interface and test flow can be customized for different test tasks, reducing hardware resources and shortening test development cycle. The hardware platform is built with an open, extensible/tailored overall architecture, and the design and development of the general test system software is completed with virtual instrument technology, script technology, programming language technology and layered software architecture. The functional command set and script development tools are designed to meet the requirements of test development, and a relatively complete command-driven system is built, which achieves the purpose of self-defining test flow and shortening the development cycle of test programs. Through different experiments, the general test system can complete the test tasks of different weapon electronic components. It has good versatility, flexibility and scalability, and can be applied to the test of various electronic components in the military factory for a long time.

Key words: general test; secondary development; virtual instrument; command driven; script

随着科学技术的不断发展,以及现代军队作战方式的升级,对武器装备的功能要求不断提高,因此现代武器装备的综合化程度与日俱增^[1-2],为保

证武器装备的质量,需要对武器电子组件的各项技术指标进行严格测试。根据传统的装备保障经验^[3],会对每个不同种类的武器电子组件或武器设

收稿日期:2024-02-28

作者简介:朱望纯(1976—),男,湖南娄底人,硕士,研究员,研究方向:自动测试总线 VXI/PXI,自动测试系统及软件,虚拟仪器及可互换式虚拟仪器(IVI)。

[†] 通信联系人, E-mail: 1679079510@qq.com

备专门研发一套自动测试系统,有研发成本高、研发周期长,并且硬件资源无法复用等缺点^[4-6],因此研究一种通用化的自动测试系统具有很大意义。

国外自动测试系统(ATS)的发展较早,以美国为代表的诸多国家提出了自动测试系统通用化的设计需求,旨在降低测试开发成本,为此也提出了不少规范自动测试系统设计的标准^[7]。目前国际上比较流行的自动化测试系统通用化设计的方案^[8]有:基于 ABBET 标准的自动化测试系统软件框架,该框架致力于自动测试系统软件的可移植性、可重用性、互换性和互操作性;基于 STD 标准实现测试程序可移植,面向信号的设计模式,该模式将底层信号进行抽象和封装,针对不同测试需求,自由配置相应信号资源,具备较强的灵活性和通用性;基于 IVI 技术实现测试仪器可互换性,在兼顾 VPP 规范的虚拟仪器架构的前提下,构建仪器标准化编程模型,可无需更改或再次编译程序即可更换仪器设备,很好地提高了自动测试系统的硬件设备通用性^[9-10]。

国内关于自动测试系统通用化设计的研究也不在少数,如李涛涛提出的基于 ATML 的电子装备自动化测试技术及系统应用研究^[11],肖勇展开了测试程序集组件化设计的研究^[12],应用虚拟仪器技术,张波等对海军舰艇信息装备的通用化测试展开了研究^[13]。

在现有自动测试系统通用化设计相关技术的基础上,以增强测试系统通用性、提高测试开发效率为目的,围绕通用化、资源可扩展、测试流程可自定义的设计要求出发,提出了基于命令驱动的通用测试系统构建方法,为通用化测试系统的设计提供一种方案。

1 硬件总体结构

通用测试系统硬件平台如图 1 所示,硬件平台支持 LAN、PXI、VXI 和 USB 等测试总线。连接仪器设备进行数据通信,以当前驱动程序支持为前提,在不同测试场景可根据要求对仪器进行扩展和剪裁。硬件平台设计有标准接口的信号调理单元对被测对象输出信号进行调理,使得被测信号满足可测量要求。不同的测试任务通用同一个硬件平台,通过系统软件控制硬件资源的重配置,实现通用测试系统硬件资源的重复利用^[14]。

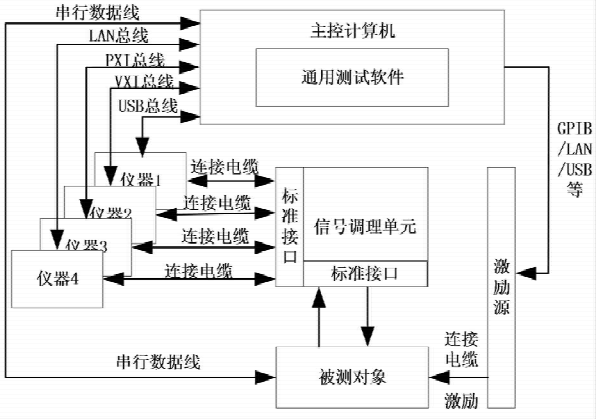


图 1 硬件总体结构图

2 软件体系结构

软件平台具有灵活、可扩展、可重用和低成本等特性,面对多种测试对象和测试任务需求,在一定硬件基础上,通过二次开发手段可以快速设计出满足测试要求的测试程序,这是研究测试软件通用化的目的。本文基于以上目的,提出一种软件体系,结构如图 2 所示。

测试需求层:该层通过分析测试任务,确定测试需求。

测试脚本开发层:该层开发满足测试需求的测试脚本。首先根据测试需求判断软硬件资源是否满足要求,若当前资源无法满足测试,则进入资源管理子系统扩充软硬件资源。并根据测试特征,进入仪器配置子系统完成所需硬件设备的配置信息描述,使用可扩展标记语言(XML)记录仪器配置信息并输出仪器配置描述文件,其次进入测试开发子系统,通过脚本开发工具,遵循脚本开发规则,将测试逻辑和具体的测试步骤用脚本命令形式表现,最后输出测试方案脚本文件。该层根据测试需求自定义测试程序,体现系统的灵活性。

脚本执行层:该层设计了运行时服务引擎,将脚本命令转换成具体的测试动作。在脚本执行子系统按照一定规则解析脚本文件中的测试命令和参数,驱动对应功能函数,执行具体的测试步骤。具体测试步骤的执行,取决于测试脚本,不针对任何测试对象,体现系统的通用性。

仪器驱动层:仪器设备的多样性是实现通用测试系统研制的硬件基础,因此该层封装了不同厂家提供的设备驱动程序,向上层提供统一的调用接口^[15]。执行脚本命令时根据配置文件匹配仪器专

用驱动。

仪器设备层:该层通过使用总线接口及仪器设备的驱动程序实现对仪器设备的数据通信及控制。

按照功能划分,软件系统可分为仪器配置子系统、测试开发子系统、脚本执行子系统、资源管理子系统。

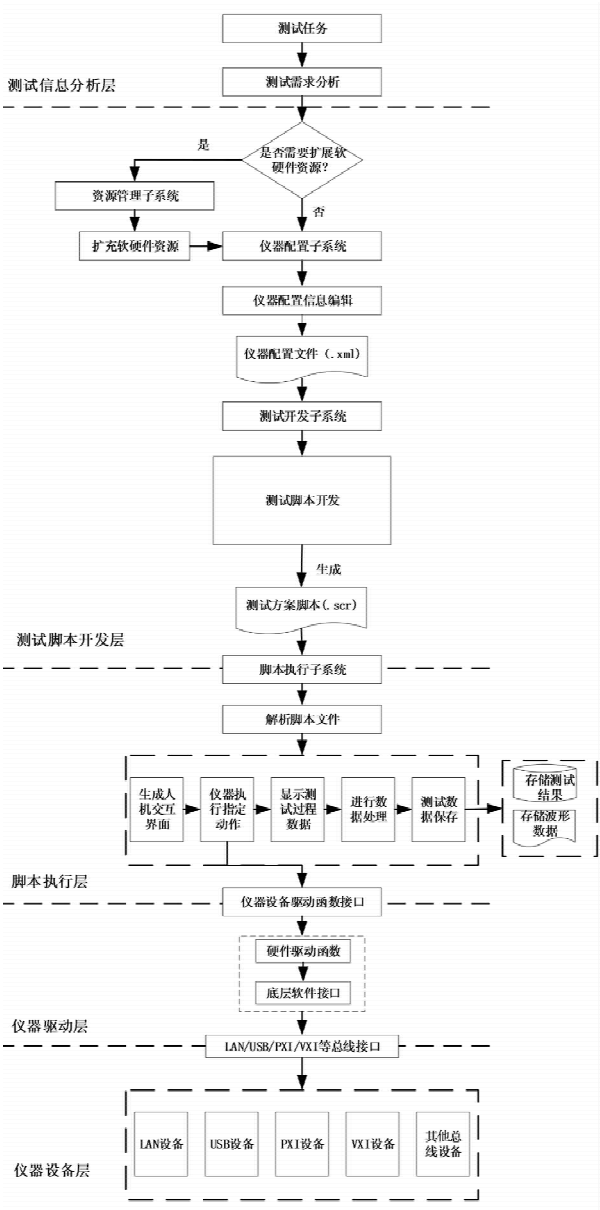


图 2 系统软件分层结构图

3 软件设计

主要对系统的 4 个子系统设计进行介绍。

3.1 资源管理子系统设计

资源管理子系统分为测试资源管理和历史记录管理,可以查看系统软硬件资源、测试脚本以及历史测试记录。测试资源主要为硬件资源和软件资源,其中硬件资源表现为当前系统所拥有的硬件

测试设备,通过信号接口和硬件接口标准化实现硬件资源的可扩展。软件资源则表现为用于测试流程开发的命令资源,通过注册用户命令实现软件资源可扩展。资源可扩展是提高测试系统通用性的重要手段,是测试系统可以应用在不同测试场景的前提。

本文软件资源扩展即为命令集的扩展。在无需编译测试系统软件的情况下,可自行另外设计数据处理算法函数,生成动态链接库的形式,并在系统中注册一条用户命令关联该数据处理算法。命令注册成功系统命令集新增一条以 USER_开头的命令,同时注册信息保存为 xml 文件格式。编辑测试任务时可像其他命令一样正常使用。命令解析执行时,根据命令读取注册信息以动态加载方式显式调用动态链接库中的数据处理函数。

3.2 仪器配置子系统设计

针对不同的测试任务,根据测试需求灵活配置硬件资源,使得不同的测试任务可以分时共用一套硬件设备,减少硬件资源堆砌。

仪器配置子系统设计用于编辑该测试任务中所需使用的硬件设备的描述信息,包括硬件型号信息、地址、能力范围和配置参数等,其中配置参数是指仪器启动工作时所必须设置的参数信息,完成所有硬件设备的描述信息编辑之后,保存为 xml 文件格式。

3.3 测试开发子系统设计

测试开发子系统为用户进行测试脚本开发提供一个开发工具。通过开发工具,按照测试需求从命令集中选定特定功能的测试命令进行组合,并遵循设定的规则为每条命令设置参数,进而完成测试脚本的开发。

结合自身武器装备专用测试系统设计经验,并充分分析一般自动测试系统共有功能,针对测试流程开发设计了较为全面的测试命令集。命令集包含六类测试命令组,分别为 UI 命令组、仪器控制命令组、数据处理命令组、数据存储命令组、流程控制命令组、其他命令组,各命令组的部分命令如表 1所示。命令集涵盖了设计一个测试软件所需的人机界面设计资源、仪器调配控制资源、数据处理资源和数据存储等。测试脚本开发示意图如图 3 所示,在六类测试命令组中选取测试命令并配置命令参数,组合形成脚本文件。

为了提高测试开发的成功率和效率,结合软件命令资源的特点,设计使用选择代替输入的形式编辑测试脚本命令,消除测试开发时命令拼写错误的可能性。设计命令参数格式提示框,选中命令时,提示框提示命令参数格式和命令说明,降低参数格

式出错率,设计参数格式检查功能,若命令参数不符合规则,无法写入并弹出提示框。为方便用户在编辑命令脚本时查看硬件配置信息,设计了一个硬件配置信息显示框。脚本编辑界面如图 4 所示。

表 1 部分命令说明

命令组别	命令	功能说明	前缀关键字
仪器控制命令组	OPEn	打开仪器或者连接远程设备	
	READ	读取仪器数据或者接收远程通信数据	
	WRITE	往仪器写入数据或者发送数据给远程设备	
	CLOSe	关闭仪器或者断开远程连接的设备	
数据存储命令组	CREATAbLe	在数据库中创建一个表格	
	ADDfIeLd	表格添加一个字段	
	CREAREPort	创建报表	
	SAVeDATa	将数据保存在文件中	
数据处理命令组	ANDCAL	与运算	
	GETMAX	获取一组数据中的最大值	
	SETVALUe	用标记名标记数值	
	FREQCAL	计算一组数据的频率	
UI 设计命令组	SINGOSC	设置一个示波器	
	MESSAgeBox	弹出消息提示框	FUN
	INADDSUB	插入一个带有数值加减功能的控件	FUN
	UPDAteWINDow	更新窗口	
流程控制命令组	IF	判断语句	IF
	WHILe	循环语句	WHILe
	GOTO	跳转语句	
其他命令组	STARThREad	开启一个线程	TSTEP
	DELAy	延时命令	
	GETCURTIme	获取当前系统时间	

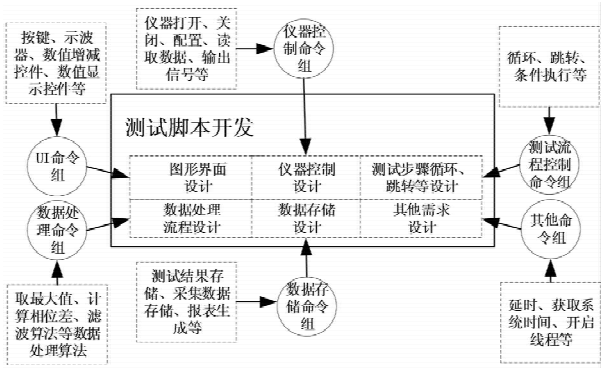


图 3 脚本开发示意图

3.4 脚本执行子系统设计

脚本执行子系统通过解析脚本命令,驱动命令所对应的功能函数,进而执行具体的测试动作。

执行测试脚本时,系统首先载入脚本文件,并对脚本进行编译,检查脚本命令是否规范,若编译通过,则按序将每一条命令进行解析。解析执行一条命令时其流程图如图 5 所示。

若测试命令参数格式不规范,参数解析时会出现错误,为了保护系统安全,会提示出错类型和出错位置,并保存当前已完成的测试数据和终止测试流程。

通过解析测试脚本命令驱动其功能功函数,完成所有测试点测试。仪器操作组命令驱动其相关功能函数,进而调用驱动层所提供统一接口,驱动仪器完成指定操作,特别地,执行仪器打开和配置相关命令时,会根据仪器地址读取其配置文件的描述信息对仪器进行配置。UI 设计相关命令驱动其功能函数完成人机界面的动态生成及测试过程中的数据动态、静态显示设置。数据处理组、数据存储组相关命令驱动其功能函数完成测试过程中数据的计算处理和保存。流程控制组相关命令驱动其功能函数完成测试流程的跳转、循环执行等。

测试执行过程中,涉及数据的存取,系统设计了一个结构体 DataStruct,该结构体包含常用的数据类型如字符串、字符数组、整型、双精度浮点数及其指针、双精度浮点数值型容器等。定义一个类型为 map< string, DataStruct> 的全局变量,用于数据临时存放。测试流程执行时,输出的数据会存入结构体变量,届时系统自动以标记名为键,结构体变量为值,存入系统全局变量 map 容器。同理,查找数据则以标记名为键,从 map 容器中根据键查找数据值。

4 实验验证

4.1 命令扩展的可行性验证

给命令集中的数据处理组扩展一条计算相位差的命令,首先将算法用 C++ 代码实现,函数定义为 vector< double> CalphaseDiff(double * data1, double * data2),并封装成动态链接库,库文件名为 Myfunction.dll。其次是来到资源管理子系统,打开用户命令注册界面,填写注册信息,其中注册的命令为“USER_CALPHDIF”,如图 6 所示。注册完成后,在数据处理命令组新增一条用户注册的命令,如图 7 所示,且在 xml 文件中保存注册信

息,保存内容如下:

```
<USER_CALPHDIF>
<DLLFile FileName="MyFunction.dll" />
<Funtion Name="CalPhaseDiff" />
  < DataType   ReturnType  = "  vector&&lt;double&>"; />
<FuntionParam ParamList="double * ,double * "
```

```
/>
  < CommandParam  Param = " TagName/Data-
Source1/DataSource2" />
  <CommandExplination Instruction="计算两组数
据之间的相位差" />
</USER_CALPHDIF>
```



图 4 脚本编辑界面

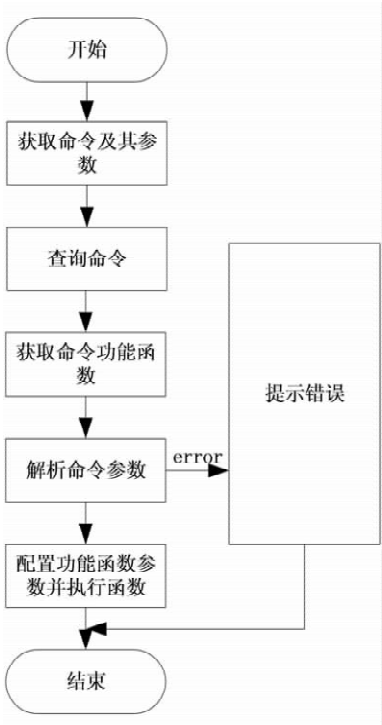


图 5 命令执行流程



图 6 命令注册界面

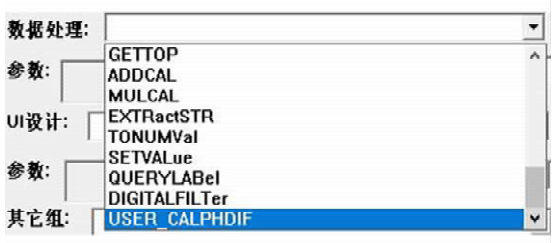


图 7 命令注册成功

4.2 系统整体可行性验证

测试某型号武器产品的一些参数是否合格,包括正负起转电流、陀螺频率、载波有效值等多个参数,并在测试界面显示载波信号、跟踪信号和舵控信号的波形数据,显示方位信号、跟踪信号、噪声信号每一帧的有效值,并将每个参数的测试结果显示出来。下面展示测试开发过程:

经过对测试任务的分析,硬件方面需要一个 8 通道以上的 AD 采集模块,用于采集 8 路模拟信号;需要一个数字 IO 模块,用于测试过程中的状态获取和控制作用。因此,选用 NI 公司的 NI9206 作为 AD 模块,NI9403 作为数字模块。进入硬件配置子系统,根据产品特征对硬件模块进行配置,配置信息保存至 xml 文件,配置内容如图 8 所示。



图 8 AD 模块配置内容

进入脚本开发子系统,开发测试任务的测试策略。

打开 AD 采集模块和数字 IO 模块、读取 AD 模块和数字模块、连接 tcp 通信的命令如下所示:

```
OPEn
@adNiDev1
OPEn
@muldigitalNiDev1
READ
@adNiDev1
READ
@muldigitalNiDev1
OPEn
```

```
@ " TCPIP0.: 192. 168. 203. 1.: 1234.:
SOCKET"
```

测试界面的开发设计,按照测试任务分析,只需要显示 3 个信号的波形数据,分别是通道 5、6、7,还有 3 个通道每一帧数据的有效值显示,分别是通道 4、6、7,所以设置一个示波器即可,测试界面的设计命令如下:

```
SETOSCNUm
@1
SETWAVESHOW
@ adNiDev1(5,6,7);1
SETDATASHOW
@adNiDev1(4,6,7);RMS;1
```

测试过程中,需要对光路进行调整,所以需要 一个控件,让测试员在合适的时间发送数据到给光设备,控制光路的水平(X)方向和垂直(Y)方向的调整。插入 X 方向调整的控件命令如下:

```
INADDSUB
@X 方向光路调整/0/0.1/8/1/-1
FUN:WRITE[ * ]
FUN:@ "0xcc 0x2f 0x55 0x0B"/HEX
FUN:WRITE[ * * ]
FUN:@ "0xcc 0x2f 0x55 0x0C"/HEX
```

X 方向光路调整控件示意图如图 9 所示,其中按下正三角执行带“[*]”的命令,按下倒三角之执行带“[* *]”的命令。



图 9 X 方向光路调整控件

测试噪声时,需要计算噪声信号的有效值,并根据计算值判断噪声测试是否合格,合格则提示测试合格并继续,否则提示不合格,并让测试员选择是否继续测试,相关命令脚本如下:

```
DISPSTATus
@"开始测试噪声"
GETRMS
@rmsValue/adNiDev1(7)
IF
@rmsValue<0.1 || rmsValue>0.4
IF:DISPSTATus
IF:@ "噪声测试不合格"
IF:SHOWSELe
IF:@ "是否结束测试?"/是/否
IF:FUN:STOPTES[ * ]
```

```
IF:FUN:@NULL
ELSe
@NULL
ELSe:DISPSTATus
ELSe:@ "噪声测试合格"
```

完整的测试脚本开发,包含了 UI 设计、仪器控制、数据处理、流程控制、数据储存等,脚本执行完成时其测试界面如图 10 所示,所有参数的测试结果都在“自定义控件”区域显示出来。



图 10 测试界面

目前该系统已经在某工厂中进行了不同电子组件的测试实验,并列出了与专用测试系统相比,本文所设计通用测试系统开发满足相同测试需求的测试程序时所用时长的缩减比,如表 2 所示。

表 2 部分测试项目情况

测试项目	软件开发时长缩减比	测试结果
ZTDD 电阻测试	52%	符合期望
DZC 功放组件测试	55%	符合期望
DZC 载波前放组件测试	51%	符合期望
DZC 计算机组件测试	48%	符合期望
DZC 次级电源组件测试	50%	符合期望
DXJZ 传感器测试	57%	符合期望
DS 电池、插座电缆组件测试	52%	符合期望

5 结 论

探讨了基于命令驱动的通用测试系统的层次结构和系统构建的设计方法,并且通过实验验证了系统的可行性、可扩展性。从实验情况表来看,该系统可以满足不同测试对象和测试任务的测试开发需求,而且开发时间大大缩减,说明系统具有良好的通用性、灵活性,同时可以有效缩短测试开发周期,与传统专用测试系统相比,可以适应更多测

试场景,使用期限也更长,为其他测试系统的构建提供了一种思路与方法。

参考文献

[1] 刘鹏飞,谢森,姚玉山,等. 某型制导武器通用自动测试系统方案设计[J]. 计算机测量与控制,2017,25(6):95—99.

[2] 孙伟. 武器系统测试性评定方法与试验用例设计[J]. 兵器装备工程学报,2016, 37(8):52—55.

[3] 李燕,王海涛,敬宏,等. 某型直升机高可靠集成式通用自动测试系统的研制[J]. 装备环境工程, 2023, 20(5): 104—110.

[4] 刘宪忠,孙妍,张淑舫. 面向弹载计算机的 ATE 通用平台设计与实现[J]. 计算机工程,2018,44(7):60—66+73.

[5] 吴国宝,宋帆,曾俊华,等. 基于 ATML 标准的 SCATS 在机载电子系统通用自动测试设备中的应用[J/OL]. 计算机测量与控制,2023,21 (8):1—11.

[6] 王森,黄思思. 特种装备通用自动测试框架及关键技术研究[J]. 电子设计工程,2023,31(14):142—146.

[7] 金鹏飞. 通用测试系统软件平台设备管理器开发[D]. 杭州:浙江理工大学,2023.

[8] 王丽. 基于脚本的通用测试系统研究[D]. 桂林:桂林电子科技大学,2017.

[9] 吕泉池. 平台管理分系统开发测试平台的设计与实现[D]. 成都:电子科技大学,2015.

[10] 文天柱,许爱强,刘勇,等. 基于 STD 标准的仪器可互换技术[J]. 现代电子技术,2023,46(4):102—106.

[11] 李涛涛. 基于 ATML 的电子装备自动化测试技术及系统应用研究[D]. 北京:北京交通大学,2020.

[12] 肖勇. 测试程序组件化设计研究与实现[J]. 现代电子技术,2023,46(8):154—158.

[13] 张波,李艳青,闫晓萍. 海军舰艇信息装备通用测试平台设计[J]. 计算机测量与控制,2023,31(4):156—160+193.

[14] 朱望纯,王丽,张大伟. 基于脚本的通用自动测试系统设计与实现[J]. 火力与指挥控制,2018, 43(2):168—171+176.

[15] 唐希浪,肖明清,薛辉辉,等. IVI.NET 仪器驱动及其应用研究[J]. 测控技术,2015,34(10):111—114.