

嵌入式软件测试平台的设计与实现

金成杰¹, 烜涛², 张焕欣¹

(1.上海中广核工程科技有限公司, 上海 200241; 2.北京火龙果网络技术有限公司, 北京 100088)

摘要: 嵌入式软件因为运行环境的差异性造成测试工具各不相同, 各个测试工具独立使用, 缺乏完整的工具方案。针对该问题, 研制了嵌入式软件测试平台, 可提供嵌入式软件测试的完整工具支持, 包括全周期测试工具链、测试 workflow 建模、测试 workflow 自动执行、将各测试工具输出的测试结果合并为完整的质量报告、存储测试工件的工程仓库。该平台把各种测试工具集成为一个完整的测试平台, 提高了嵌入式软件测试的完整性和效率。用户还可以根据需要扩展测试工具、工作和质量报告, 实现嵌入式软件测试的复用机制。

关键词: 嵌入式软件; 集成测试; 持续集成; 测试平台

中图分类号: TP311.5

文献标识码: A

DOI: 10.19358/j.issn.2096-5133.2022.04.016

引用格式: 金成杰, 烜涛, 张焕欣. 嵌入式软件测试平台的设计与实现[J]. 信息技术与网络安全, 2022, 41(4): 102-107.

Design and implementation of embedded software test platform

Jin Chengjie¹, Xu Tao², Zhang Huanxin¹

(1. Shanghai CGN Engineering Technology Co., Ltd., Shanghai 200241, China;

2. Beijing Pitaya Network Technology Co., Ltd., Beijing 100088, China)

Abstract: Embedded software test tools are different because of the differences of operating environment, and each test tool is used independently, lacking a complete tool solution. To solve this problem, an embedded software testing platform is developed in this paper, which can provide a complete tool support for embedded software testing. It includes full-cycle test tool chain, test workflow modeling, automatic execution of test workflow, integration of test results output by each test tool into a complete quality report, and engineering warehouse for storing test artifacts. Embedded software testing platform integrates various testing tools into a complete testing platform, which improves the integrity and efficiency of embedded software testing. Users can also extend test tools, work and quality reports as needed to achieve the reuse mechanism of embedded software testing.

Key words: embedded software; integration testing; continuous integration; test platform

0 引言

嵌入式软件常用于汽车、飞机、工业控制等领域, 很多与人身安全密切相关, 为了保证软件的质量, 会做尽可能多的测试: 代码检查、单元测试、集成测试、系统测试等。这些测试都有相应的测试对象、工具、质量报告, 而且不同的嵌入式软件常常具有不同的运行环境。以上因素造成了嵌入式软件测试的复杂性和挑战性。下面就对嵌入式软件测试存在的问题进行解读, 并提出相应的解决方案。

1 嵌入式软件测试存在的问题

为建立完整的嵌入式软件测试工具方案, 应全面分析其过程、角色、工具、数据, 发现各环节存在的问题^[1]。目前嵌入式软件测试过程中存在以下问题:

(1) 测试工作效率低: 涉及多种测试工作, 主要靠人工。

(2) 测试工具匮乏: 嵌入式软件的测试工具与测试环境密切相关, 经常找不到特定的测试工具。

(3) 难以获得完整的质量视图: 不同测试工具

输出的报告各不相同,没有一个综合的质量报告。

(4)测试工件缺乏管理:各种测试对象、测试环境、测试工具、测试脚本、测试报告,缺乏集中管理。

(5)复用率低:不同的测试项目具有类似的测试流程、工具、工件,但是需要重复搭建。

2 嵌入式软件测试解决方案

嵌入式软件测试的5大问题是密切相关的,应该提供一个完整的嵌入式软件测试平台,目标如下:

(1)全周期流程建模:代码检查、单元测试、集成测试、系统测试^[2];

(2)提供完整工具链:代码检查工具、单元测试工具、集成测试工具、系统测试工具;

(3)自动化执行全周期测试流程^[3];

(4)把各测试工具输出的报告合成完整质量报告;

(5)建立测试工程仓库,集中管理各种被测程序、环境配置、测试用例、测试脚本、测试报告;

(6)提供项目级的整体测试资源复用能力。

嵌入式软件测试平台的方案概览如图1所示。

本研究项目在当前已有的嵌入式软件测试工具基础上,研发嵌入式软件测试平台,为测试工作提供集成解决方案。

3 嵌入式软件测试平台的架构设计

为将上述的方案转化并形成最终的产品,需进行完整的架构设计,嵌入式软件测试平台的架构设计包括功能架构和软件开发架构。

3.1 功能架构

嵌入式软件测试平台采用客户端和服务端的部署方式,服务器分为管理服务器和测试服务器,相应的功能分配到各个部署节点,系统功能部署图如图2所示。

对应的子系统和功能分配如表1所示。

3.2 软件开发架构

为了实现开发的复用和扩展,平台采用了基于组件和面向服务的架构模式。根据可复用、可扩展的原则,把公共部分提炼为应用框架和公共组件、公共服务,并提供了定制部分的接口和配置参数文件。如下是开发架构的简要说明:

(1)采用可扩展的应用框架,可以通过配置扩展功能;

(2)客户端和服务端的接口采用 Restful 微服务技术;

(3)基于 GDI+ 开发 workflow 建模,支持 workflow 复用;

(4) workflow 配置用 XML 格式文件存储,易于维护。

下面详细列出了各个子系统的组件设计和功能说明。客户端子系统的组件模型如图3所示,组件说明如表2所示。

管理服务子系统的组件模型如图4所示,组件说明如表3所示。

测试服务子系统的组件模型如图5所示,组件说明如表4所示。

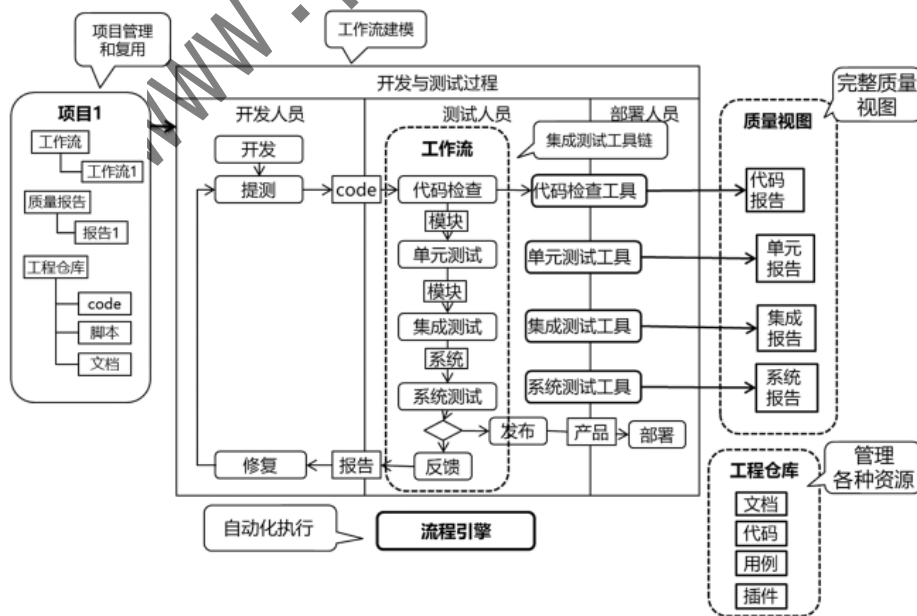


图1 嵌入式软件测试平台的解决方案概览图

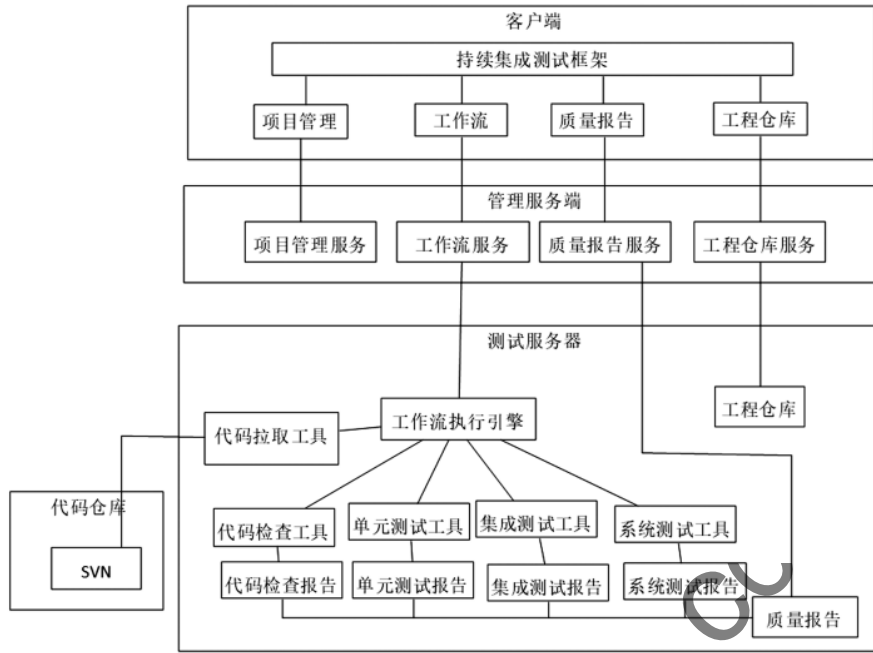


图2 嵌入式软件测试平台功能分配图

表1 嵌入式软件测试平台的子系统和功能描述

子系统	功能描述
客户端	提供用户操作界面,包括:创建项目;配置 workflow;配置 job 和插件;配置质量报告模板;启动 workflow,执行 job,生成质量报告;查看质量报告
管理服务器	提供应用服务功能支持,包括:项目管理、 workflow 管理、质量报告管理、工程仓库管理;提供工程数据库,存储各种工程数据,例如项目信息、 workflow 配置、工具配置、质量报告配置等
测试服务器	workflow 引擎调度工具执行;代码拉取、代码检查、代码编译、单元测试、部署、集成测试、系统测试;提供各种测试工具输出的质量报告收集服务;提供工程仓库,存储: workflow、工具配置、被测的程序、环境配置、测试脚本、测试报告
代码仓库	开发时提交代码到仓库,测试时拉取代码

表2 客户端子系统的组件说明

组件	功能描述
界面框架	提供界面框架,可以接入各种功能组件
项目管理	创建项目,录入、查询、修改和删除项目信息
工作建模	编辑 workflow 信息,把 workflow 提取到可复用模板库
Job 配置	编辑 job 的信息
工具配置	配置测试工具的命令、参数、工具地址、输出
质量报告	配置质量报告的数据来源、合成关系、显示样式,并可以浏览质量报告
工程仓库	把测试资源放入工程仓库,查询、浏览和使用工程仓库里的资源,支持本地和服务端 2 种工程仓库
Web 服务接口	访问服务端的 Web 服务接口
本地文件接口	访问本地工程仓库的文件接口

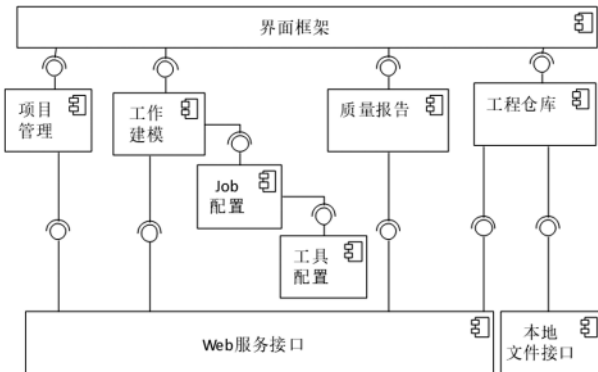


图3 客户端子系统的组件图

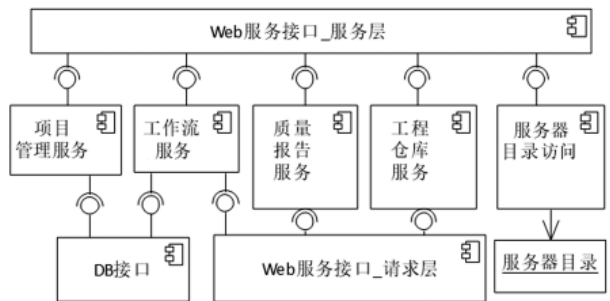


图4 管理服务子系统的组件图

表 3 管理服务子系统的组件说明

组件	功能描述
Web 服务接口	提供对外的 Web 服务框架,可接入各种功能服务
项目管理	访问测试服务器上的项目库,提供项目管理的各种服务;查询、添加、修改、删除项目信息
工作流程	提供 workflow 管理服务:创建、查询、修改和删除 workflow,提供 workflow 调度执行服务
质量报告	提供质量报告服务:配置、生成、查询、删除质量报告
工程仓库	提供工程仓库管理服务:创建工程仓库,查询、修改、删除工程仓库;提供工程仓库的资源服务:导入资源、导出资源、查询资源、删除资源
Web 服务接口	提供测试服务器的 Web 服务接口
服务器	提供测试服务器管理的的服务:查询、打开、关闭到测试服务器的连接

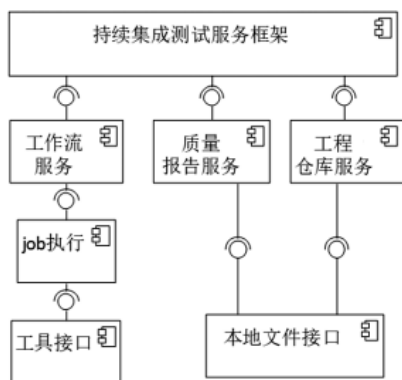


图 5 测试服务子系统的组件图

表 4 测试服务子系统的组件说明

组件	功能描述
测试服务框架	提供测试服务的整体框架,可接入各种测试服务
workflow 服务	提供测试 workflow 的各种服务,包括:创建、查询、修改和删除,workflow 调度执行
Job 执行	提供 job 管理服务:创建、查询、修改、删除 job,提供 job 执行服务,可把参数传递给工具接口
工具接口	提供各种测试工具的统一适配接口,传递工具参数给工具,调用工具执行测试
质量报告服务	访问生成的测试报告,根据质量报告配置合成综合质量报告,并提供测试报告和质量报告查询服务
工程仓库服务	提供工程仓库的访问服务:添加、查询、修改、删除

4 关键技术设计及实现

4.1 嵌入式软件测试工具链与运行环境

嵌入式软件主要是基于 C、C++ 进行开发,运

行在嵌入式操作系统环境之上,而嵌入式操作系统类型众多,包括嵌入式 Linux、VxWorks、Android、RTOS 等。这就需要构建一套能够满足各种嵌入式软件运行环境的测试工具^[4-5]。为此,本文提供了如下工具链:

(1)源代码相关的检查与测试选择了支持 C/C++ 的工具:代码对比工具 Beyond Compare,代码检查工具 CppCheck,单元测试工具 Gtest、C++Test。

(2)运行环境有关的工具提供 2 种支持^[6-8]:

①针对不同的操作系统,提供对应的运行容器。例如,对于 VxWorks 操作系统,提供 Simics 运行容器,虚拟软件运行环境。

②在操作系统之上,构建一个数据总线。在测试的时候,模拟各种通信的数据,屏蔽运行环境。数据总线采用了 DDS 技术框架,如图 6 所示。

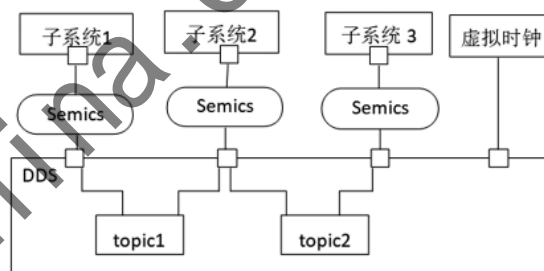


图 6 软件运行容器和数据分发服务总线(DDS)

根据实际工作需要,收集、整理并集成嵌入式软件测试工具链,如表 5 所示^[9-11]。

4.2 可扩展的工具适配接口

嵌入式软件测试平台需要集成各种测试工具:代码检查工具、单元测试工具、集成测试工具、系统测试工具,这就需要可扩展的适配接口,通过配置参数快速引入新的测试工具^[12-13]。测试平台的接口关系如图 7 所示。

各种测试工具有不同的工作界面、命令、参数和报告样式。平台为不同的接口建立一个统一的配置参数结构。当出现新的工具的时候,用户只需为该工具配置对应的接口参数^[14]。

工具的基本描述信息属于基本属性,各个工具的数据结构都是相同的,而输入输出参数具体到不同的工具会有所不同,用户可以根据工具类型自定义输入输出参数。输入和输出参数的定义采用自定义结构的 XML 数据格式,每个 job 的配置信息独立存储为一个 job 的 XML 文件。

表 5 完整的嵌入式软件测试工具链

工作	代表工具	用途
1 版本控制	SVN 插件/Git StarTeam	管理源代码的仓库
2 代码对比	Beyond Compare Diff Merge	对比代码不同版本的差异
3 代码检查	QAC CppCheck	对代码的各种规则进行检查
4 代码度量	Testbed NCSS	度量代码的复杂度
5 编译构建	Workbench Visual Studio	编译源代码为目标程序
6 单元测试	C++Test/GTest	对软件编译单元进行单元测试
7 集成测试	Tessy	基于组件之间的调用接口,进行组件集成测试
	integrator	基于子系统之间的通信接口,进行子系统集成测试
	虚拟路由插件	调用虚拟路由工具,实现虚拟运行环境的总线通信通道配置
8 虚拟运行环境	虚拟时钟插件	调用虚拟时钟工具,实现虚拟运行环境的时间同步
	Simics 插件	构建软件运行容器,实现虚拟运行环境的构建
9 接口总线	DDS 工具	支持应用组件之间的基于消息的数据总线
10 部署	目标文件下载工具	实现指定版本代码的目标文件下载至目标机
11 覆盖率分析	DT10	基于动态运行测试分析代码的测试覆盖率

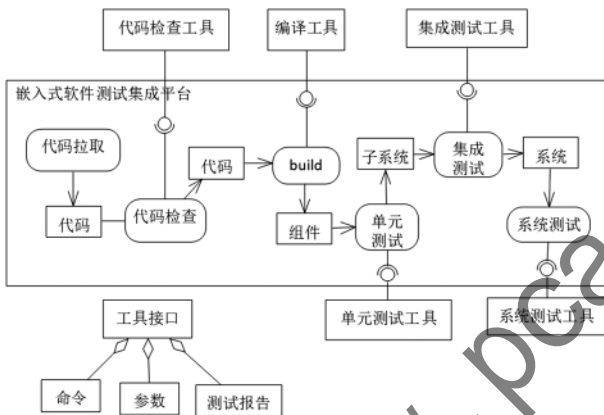


图 7 嵌入式软件测试平台接口关系图

统一的质量报告生成流程图如图 8 所示。

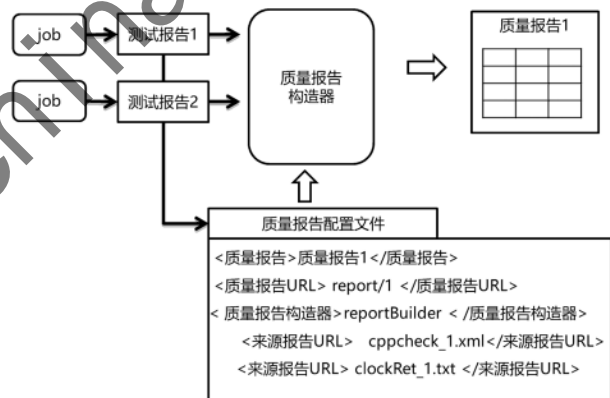


图 8 综合质量报告生成原理示意图

4.3 统一的质量报告

将各个测试工具输出的报告集成到一个质量报告需要具备 2 个功能^[15]：

(1)配置报告:建立各个测试工具输出的测试报告到综合质量报告的映射关系；

(2)报告生成:能够根据配置文件,读取各工具的测试报告,解析其中的数据,根据配置生成质量报告。

质量报告的配置参数包含：

(1)来源报告:来源报告名称和来源 URL；

(2)目标报告:目标报告名称和输出的 URL；

(3)报告构造器:不同的质量报告需要不同的报告构造器,为此抽象一个构造器基类,实现可扩展框架。

5 平台实现与验证

本测试平台软件开发完成后进行了多个嵌入式软件测试项目的验证。用户在测试平台操作界面上建立测试 workflow,配置各个 job 的工具、质量报告的组合关系,然后启动 workflow,依次调用各个 job 的工具、执行测试、输出测试报告,当所有测试都完成后,合成一个综合质量报告。用户可以对测试过程的状态进行监控,浏览最终的质量测试报告。各个测试工作的执行记录如表 6 所示。表 7 是平台使用前后的工作效果对比。

6 结论

嵌入式软件测试平台解决了嵌入式软件测试散乱的问题,提供了完整嵌入式软件测试方案:全

表6 嵌入式软件测试平台应用实例

评价指标	测试工具	测试规模	测试结果
代码检查	QAC	106 379 行代码	131 个问题
代码度量	Testbed	106 379 行代码	度量报告
编译构建	Workbench	106 379 行代码	编译报告
单元测试	C++Test	2 315 个函数	356 个 bug
集成测试	integrator	113 个接口函数	92 个 bug
系统测试	仿真测试环境	58 个功能用例	32 个 bug

表7 嵌入式软件测试平台应用效果表

评价指标	使用平台前	使用平台后
测试工具	代码检查:支持充分 单元测试:支持不充分	代码检查:支持充分 单元测试:支持充分
支持度	集成测试:没有工具支持 系统测试:支持充分	集成测试:工具支持有效 系统测试:支持充分
测试执行效率	一个测试周期走完需要 几周到几个月	一个测试周期走完需要 几小时到几天
质量视图 可读性	没有整体质量视图,对系 统的整体质量评价缺乏 严密性	具有整体质量视图,对系 统质量评价具有完整性 和客观性
测试项目 复用度	复用靠人工完成,包括: 1.测试流程:文档复制 2.测试工具:人工安装 3.测试用例:文档复制 4.测试脚本:人工复制 5.测试数据:人工复制 6.质量报告:人工合成	测试平台支持复用,包括: 1.测试流程:流程模型克隆 2.测试工具:自动安装配置 3.测试用例:自动复制 4.测试脚本:自动复制 5.测试数据:自动复制 6.质量报告:自动合成

周期测试过程建模、测试工具链、测试流程自动执行、合成质量报告,在嵌入式软件测试领域具有先进性和示范作用。本测试平台经过多家企业的实际应用,有效地提升了嵌入式软件测试的效率和质量。同时本平台也发现存在下述待加强之处,例如:应支持更多的嵌入式测试工具,应能够支持更多嵌入式系统的测试环境,能够支持多个测试项目的并行执行。针对以上的改进需求,笔者增强了平台的功能,部分需求已经完成改进,并计划持续提升完善,让测试平台具备更好的实用性和广泛的适用性。

参考文献

- [1] 布鲁克曼.嵌入式软件测试[M].张君施,译.北京:电子工业出版社,2004.
- [2] 杜瓦尔.持续集成:软件质量改进和风险降低之道[M].王海鹏,译.北京:机械工业出版社,2008.

- [3] 蔡建平.嵌入式软件测试实用技术[M].北京:清华大学出版社,2010.
- [4] 王立泽.嵌入式软件测试系统执行框架研究[J].测控技术,2010,29(12):82-86.
- [5] 殷永峰.嵌入式软件系统测试:基于形式化方法的自动化测试解决方案[M].北京:机械工业出版社,2021.
- [6] 奚成义.发射控制软件通用测试平台中支撑软件的设计与实现[D].哈尔滨:哈尔滨工业大学,2013.
- [7] 芦彩林,丁刚毅.嵌入式软件全数字仿真测试环境设计[J].计算机仿真,2007(1):295-297.
- [8] 李伟,程朝辉,王波.仿真环境下嵌入式软件测试技术与应用[J].装甲兵工程学院学报,2004,18(2):57-60.
- [9] 范琳,王忠民.宿主环境下嵌入式软件白盒测试方法[J].西安邮电大学学报,2014(5):106-111.
- [10] 谢东,李昌禧.基于LabVIEW的嵌入式软件黑盒测试系统的研究[J].工业控制计算机,2005,18(12):48-49.
- [11] 呼啸.应用ECU在环仿真的EPS系统嵌入式软件测试研究[D].长春:吉林大学,2012.
- [12] LASTER B.JENKINS2权威指南[M].郝树伟,石雪峰,雷涛,等,译.北京:电子工业出版社,2019.
- [13] Jenkins 开源社区网站.Jenkins developer guide[EB/OL].[2021-11-26].<https://www.jenkins.io/doc/developer/book/>.
- [14] 卢家涛.自动化测试实战[M].北京:机械工业出版社,2021.
- [15] 沮涛.基于模型的质量评测[EB/OL].[2021-11-26].<http://www.uml.org.cn/modeler/202005261.asp>.

(收稿日期:2021-11-26)

作者简介:

金成杰(1982-),通信作者,男,硕士,工程师,主要研究方向:控制系统设计、系统集成。E-mail:jinchengjie@cgnpc.com.cn。

沮涛(1973-),男,硕士,高级工程师,主要研究方向:计算机应用、软件工程。

张焕欣(1974-),男,硕士,高级工程师,主要研究方向:控制系统设计、计算机应用。

版权声明

经作者授权，本论文版权和信息网络传播权归属于《信息技术与网络安全》杂志，凡未经本刊书面同意任何机构、组织和个人不得擅自复印、汇编、翻译和进行信息网络传播。未经本刊书面同意，禁止一切互联网论文资源平台非法上传、收录本论文。

截至目前，本论文已经授权被中国期刊全文数据库（CNKI）、万方数据知识服务平台、中文科技期刊数据库（维普网）、JST 日本科技技术振兴机构数据库等数据库全文收录。

对于违反上述禁止行为并违法使用本论文的机构、组织和个人，本刊将采取一切必要法律行动来维护正当权益。

特此声明！

《信息技术与网络安全》编辑部
中国电子信息产业集团有限公司第六研究所