

**自动化测试软件设计方案**

编 写：冯思墨

校 对：隋佳林

审 核：于劲松

批 准：

北京航空航天大学自动化科学与电气工程学院

自动测试设备实验室（ATE）

2023年4 月10 日

|  |
| --- |
| 文件状态：  [√] 草稿  [ ] 正式发布  [ ] 正在修改 |
| 文件起草分工：  于劲松  冯思墨 |

|  |  |  |  |
| --- | --- | --- | --- |
| **编制** |  | **日期** |  |
| **校对** |  | **日期** |  |
| **审核** |  | **日期** |  |

状 态 页

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **更改人** | **更改描述** | **更改日期** |
| 1.0 |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目 录

[1. 概述 1](#_Toc27021)

[1.1项目简介 1](#_Toc6214)

[1.2适用范围 1](#_Toc5825)

[1.3术语与缩略语 1](#_Toc10316)

[1.4引用文件 2](#_Toc1509)

[2 设计要求 2](#_Toc19164)

[2.1总体要求 2](#_Toc25957)

[2.2功能要求 3](#_Toc4465)

[2.3性能要求 3](#_Toc12931)

[2.4组成要求 3](#_Toc22295)

[2.4.1测试平台软硬件组成情况 3](#_Toc11179)

[2.4.2测试平台软硬件组成要求 6](#_Toc6639)

[2.5软硬件其他要求 6](#_Toc12883)

[2.5.1开发环境要求 6](#_Toc2568)

[2.5.2通用性要求 6](#_Toc249)

[2.5.3扩展性要求 7](#_Toc16360)

[2.5.4可靠性要求 7](#_Toc32256)

[2.5.5维护性要求 7](#_Toc24698)

[2.5.6安全性要求 8](#_Toc31350)

[2.5.7信息安全要求 8](#_Toc24887)

[3 设计内容 9](#_Toc25610)

[3.1系统架构 9](#_Toc30400)

[3.1.1 自动化测试平台逻辑层次结构 9](#_Toc27582)

[3.1.2自动化测试平台运行与功能架构 10](#_Toc18770)

[3.2自动化测试平台功能设计 13](#_Toc28440)

[3.3测试设备主控端设计 21](#_Toc9649)

[3.3.1测试资源管理与监控模块设计 22](#_Toc8154)

[3.3.2测试用例开发与管理模块设计 24](#_Toc22133)

[3.3.3静态激励与管理模块设计 26](#_Toc1679)

[3.3.4测试数据管理模块设计 27](#_Toc28912)

[3.3.5数据库功能模块设计 28](#_Toc7791)

[3.3.6 ICD导入与加载模块设计 30](#_Toc31229)

[3.4测试设备服务端设计 32](#_Toc32099)

[3.4.1测试资源驱动模块设计 33](#_Toc10865)

[3.4.2用例测试服务执行模块设计 34](#_Toc20789)

[3.4.3静态激励执行模块设计 35](#_Toc6162)

[3.4.4测试对象数据获取模块设计 35](#_Toc814)

[3.4.5模型加载与驱动功能模块设计 36](#_Toc28472)

[3.5通信中间件及交互共享数据池设计 37](#_Toc7595)

[3.6自动化测试平台测试资源 39](#_Toc2696)

[3.6.1BST35701 40](#_Toc11135)

[3.6.2 BST35702 41](#_Toc24932)

[3.6.3 BST34211 41](#_Toc18186)

[3.6.4 BST31701 41](#_Toc19207)

[3.6.5 AECDA-PXI-20 42](#_Toc32628)

[4 方案创新点 43](#_Toc17261)

[5 方案成果与进度安排 43](#_Toc11383)

[6 方案与技术协议对照 44](#_Toc2154)

[7 方案总结 46](#_Toc19851)

# 概述

## 1.1项目简介

自动测试软件可单机或联网多机运行，支持半测、成测、二筛、常温和高低温等多种测试应用选项，用户可根据实际情况在软件内对测试类型和测试环境进行选择，进行相应的测试。

测试软件具有完善的测试显示和测试结果输出保存、查询功能。测试数据显示信息主要是根据产品的详细规范（出厂标准），按照测试指标，根据被测产品的测试指标不同，显示的表头不同，有单位、最大值、典型值、最小值等信息。将采集的信息存储到数据库中，对应的存储结构不同（不同温度点测试、不同工序）。

测试结果数据查询主要是对存储在数据库中的产品数据进行查询并导出到 Excel 表中。表头数据显示不同，查询字段不同。每个产品根据产品类型、测试类型、测试温度，有相对应的 Excel 表单。导出的时候将查询的内容对应到表单中数据位置进行写入。

自动化测试软件平台支持用户对现有测试功能进行封装、免编程二次开发，实现面向测试功能的测试用例开发及重用。并支持用户根据新的测试任务进行仪器和测试功能的封装。

## 1.2适用范围

本文档适用于自动化测试软件设计评审，指导自动化测试软件开发、为安装调试自动化测试软件以及自动化测试软件平台交付验收提供依据。

## 1.3术语与缩略语

1. **术语与缩略语表**

|  |  |  |
| --- | --- | --- |
| 序号 | 术语与缩略语表 | 含义 |
|  | ICD | 接口控制文件 |
|  | ATML | 自动测试标记语言 |
|  | XML | 可扩展标记语言 |
|  | B/S | 浏览器/服务器结构 |
|  | I/O | 输入/输出 |
|  | 自动化测试 | 测试平台 |

## 1.4引用文件

* 《自动测试系统软件平台架构要求的说明》
* 《IEEE ABBET Guide for the Understanding of the 'A Broad-Based Environment for Test (ABBET) Standard》
* 《IEEE Guide for the Use of IEEE Std 1641, IEEE Standard for Signal and Test Definition》
* GJB 1622-1993 《航空电气和电子设备的测试设备通用规范》
* GJB 1132-1991 《飞机地面保障设备通用规范》
* GJB5109-2004 《装备计量保障通用要求检测和校准》
* GB/T11918 《工业用插头插座盒耦合器规范》
* GB/T11919 《工业用插头插座盒耦合器检测用量规技术标准》
* Q/AVIC 30108-2015 《航电软件测试报告编写指南》

# 2 设计要求

## 2.1总体要求

设备总体要求：建设一套自动化测试平台，可以通过甲方提供计算机控制终端相关软硬件进行测试任务的控制与交互，满足单机或分布并行运行要求，可以包括主控端与多个测试服务节点，具备与其他平台交互的功能软硬件接口。具备基本的ICD工具以及相关管理功能，测试资源管理，测试用例开发与管理功能，测试数据监控、采集与管理功能，用户权限管理。

自动化测试软件平台包含测试主控端、单个或多个测试服务节点以及配套的硬件被测对象，测试平台所需满足的要求分为功能要求、性能要求、组成要求以及软硬件其他要求。

## 2.2功能要求

自动化测试平台主要功能包括:

（1）包括主控端与单个或多个测试服务节点，具备与其他平台交互的功能软硬件接口;

（2）具备基本的ICD 工具以及相关管理功能;

（3）具备测试资源管理功能;

（4）具备测试用例开发与管理功能;

（5）具备测试数据监控、采集与管理功能;

（6）具备用户权限管理功能。具备用户管理功能，支持设置的操作人员角色应当包括∶管理员、开发与调试人员、测试使用人员等，根据操作角色不同，对操作人员进行权限划分的功能;

（7）具有良好的人机界面，并提供指令发送宏和在线帮助功能;

（8）提供可视化、易于操作的控制工具，以满足用户对上述功能的控制;

（9）集成测试环境具有可扩展性。

## 2.3性能要求

自动化测试软件平台应当满足的性能指标如下：

（1）测试平台的启动时间应当不超过30秒；

（2）测试平台的操作-响应时间应当不超过1秒；

（3）测试平台的激励-响应时间应当不超过5秒；

（4）测试平台的数据监控刷新周期应当不高于500毫秒；

（4）测试平台应当持续正常工作至少8小时；

（5）支持不少于10项测试前端实时运行。

## 2.4组成要求

### 2.4.1测试平台软硬件组成情况

（1）测试资源驱动至少应包括并不限于各类设备如电源、电子负载、示波器、数字多用表、信号源等。

（2）设备应选用成熟可靠，技术选型的通用仪器设备。元器件、原材料均应选用“工业级”产品并经广泛应用。

（3）该平台应包括一台服务器，多台服务节点。配置显示交互模块。

### 2.4.2测试平台软硬件组成要求

（1）外观：设备表面不应有损伤、变形、掉漆和锈蚀；紧固件应齐全、连接牢固，并具有可靠的防松措施；插头、插座、电缆标识清晰，无缩针、缩孔、倒针。

（2）铭牌和标识：设备应有项目标识和铭牌，铭牌采用中文标识，位置便于识别，内容包括名称、型号、出厂编号、出厂日期、制造厂等内容，其中设备名称、型号、制造厂必须与合同和技术协议一致，标识和铭牌应字迹清晰、安装牢固、横平竖直。设备上如有英文标识，应在英文标识旁应附有相同内容的中文标识。

（3）硬件工作环境要求：

地面电源电压：5/12V；贮存温度：－10℃～50℃ 工作温度：-55℃～125℃）；工作湿度范围：10%～95%（无凝露）；提供可靠地接地线。

## 2.5软硬件其他要求

### 2.5.1开发环境要求

（1）Python3.8及以上版本，以及其他主流相关开发工具；

（2）服务器端主控软件的人机界面采用B/S架构开发，采用模块化的开发框架，便于扩展人机接口组件与控件；

（3）文档版本号齐全，相关代码注释清晰；

（4）在Windows操作系统下开发。

### 2.5.2通用性要求

测试平台应当具有通用性，具体应符合：

（1）测试平台应当能够根据应用场景的不同进行灵活的构型裁剪，以满足集成验证及环境试验的要求；

（2）测试平台的应用场景可以包括：

a）被测硬件设备装调测试系统级动态联试与集成；

b）被测硬件设备分系统级的装调测试与半实物仿真测试；

c）被测硬件设备部件级入厂验收试验；

d）分布测试资源自检与校准测试。

### 2.5.3扩展性要求

集成测试环境具有可扩展性，具体应符合：

（1）测试服务节点与客户端的扩展

测试平台采用面向服务的软件架构，通过软件与测试物理资源的隔离，通过网络配置可快速地扩展测试服务或客户端节点，从而满足较强的节点扩展能力的需求；

（2）面向测试功能的关键字库开发及扩展

测试平台应能提供面向甲方特定需要测试功能的关键字库的管理功能，并提供关键字的开发环境，便与关键字的扩展，从而实现测试平台测试能力的扩展；

（3）测试资源的扩展

测试平台应采用分层的软件设计，来隔离测试资源变更带来的影响，从而便于资源扩展；

（4）测试模型库的扩展

测试平台应具备测试仿真模型的开发和运行环境，并通过信号通道的配置的方式实现模型与资源解绑，从而便于测试模型库的扩展。

**2.5.4可靠性要求**

完成系统间的测试，平均修复时间不长于24小时。

### 2.5.5维护性要求

测试平台应当充分考虑维护性，方便软件维护，测试平台应当采用模块化设计思想。设备软硬件采用模块化设计，硬件部分便于拆装，每个软件模块都具有相对独立的功能，结构紧凑，可靠性高，并具有较强的扩展能力；设计完善的防差错措施和识别标志。同时，需保证提供保证设备正常运行两年所必备的易损件；提供设备操作和维修的专用工具。采用标准化、模块化进行设计以便升级维护。

### 2.5.6安全性要求

测试平台应当考虑安全性，具体应符合：

（1）测试平台应当具备处理因人员误操作引起的系统危害性错误的能力；

（2）测试平台应当具备处理软件严重故障的能力；

（3）某一模块的故障不应影响平台主要功能的正常运行；

（4）设备应符合的安全环保要求（防爆、噪音、废液/废气、固废、安全警示标识等）；

（5）该设备不会导致人员伤亡、危害健康及环境，也不会给设备或财产造成破坏或损失。

### 2.5.7信息安全要求

测试平台应当考虑信息安全性，具体应符合：

（1）测试平台应当在启动时设置密码保护，并根据产品所涉及信息的密级，采取相应级别的密码保护措施；

（2）测试平台应当限定数据系统的访问权与被访问权，并采取必要的访问控制手段；

（3）不得安装使用国家保密管理部门或上级机关单位明令禁止安装使用的软件或系统；

（4）不能存在监听与设备无关的多余模块；

（5）所配操作系统应安装正版操作系统软件；

（6）专用计算机、特殊设备等信息设备所配置和使用的操作系统和杀毒软件等软件应与安全防护产品具有兼容性，并提供兼容性测试报告。（公司安全产品版本：a）三合一（品牌是：鼎普公司）。b）主机审计单机版（品牌是：北信源公司）。c）瑞星杀毒软件（版本是：ESM 3.0企业版））；

（7）内部开口机须安装：主机审计、杀毒软件（多台计算机组成局域网，其中只能有1台计算机用于导入导出，这台计算机就叫开口机）；内部计算机须安装：杀毒软件。涉密计算机须安装：三合一、主机审计、杀毒软件；

（8）操作系统采用正版操作系统，不能存在病毒木马及恶意代码程序。

# 3 设计内容

## 3.1系统架构

### 3.1.1 自动化测试测试平台逻辑层次结构

依据IEEE1226标准（广域测试环境），自动化测试平台的设计应降低测试服务、测试资源和被测系统之间的物理相关性，提高测试平台的复用率和通用性；分布式概念引入自动化测试领域，也可使得测试人员依据测试策略和需求，灵活配置测试平台测试模块，避免了传统通用测试平台规模大、复杂度高、配置能力差、资源使用率较低等缺点；层次化测试平台结构设计能够降低测试平台与测试仪器的耦合度，提高测试平台的扩展性、移植性与可重配置性。



1. 自动化测试设备逻辑层次结构

自动化测试平台总体结构参照IEEE1226标准，采用层次化结构及面向对象模块化设计思想，主要分为测试策略层、测试过程层、资源管理层和仪器控制层。测试信息描述遵循IEEE 1641（ATML）标准，策略信息、测试信息、测试资源及其配置采用ATML描述，便于后续不同型号的测试实验的移植与重用。

1. 测试需求与测试策略层

测试需求与测试策略层上进行测试信息的配置，如测试需求、测试策略等。根据产品测试的手段和需求，制定测试进行的顺序，驱动测试过程层进行测试操作，获得测试数据之后，给出测试结果；

1. 测试过程层

测试过程层即测试程序集与测试执行逻辑，测试程序集是根据测试需求开发的测试用例与静态激励的集合，测试执行逻辑是测试程序中对测试关键字的控制执行逻辑，包含判断与循环等逻辑。运行时，测试平台根据测试策略和需求执行不同的测试程序集，获取测试数据。

1. 资源管理层

资源管理层建立在仪器控制层之上，是整个测试系统仪器资源、配置资源管理的中枢，它统筹管理虚拟模型库资源，直接为测试程序集服务。资源管理包括对数据资源的管理和仪器资源的管理。数据管理实现对测试过程中所得到的数据进行存取，对配置数据进行存取等；仪器资源管理实现对系统仪器资源的统一调度，根据测试程序集的需求分配仪器，执行仪器功能，结束后回收仪器资源；

1. 仪器控制层

仪器控制层是整个软件的最底层，该层直接与设备硬件进行交互，是虚拟资源与真实设备交互的桥梁。该层软件建立在仪器厂家提供的驱动、API等资源之上，对底层仪器控制函数进行基于功能的二次封装。新封装的函数具有统一的函数形式和明确的函数名称，对外提供规范的调用接口，供资源管理层调用，以此来补偿不同厂家仪器之间的差异，提高测试系统的仪器互换和资源扩展能力。

### 3.1.2自动化测试平台运行与功能架构

测试平台架构包含一个测试主控端与一个/多个测试服务节点所构成的软件测试平台以及多个功能板卡作为硬件被测对象。

基于以太网的分布式测试平台架构，通过资源管理实现了测试能力的综合，并采用基于B/S平台架构进行搭建，设计一套自动化、通用化的软件测试平台。测试平台提供一套统一的测试环境与可重配置的I/O接口资源，所有系统功能模块根据需求部署于测试主控端与测试服务节点，在局域网内共同构建一个基础化、标准化系统架构，使得不同型号产品的测试项目能够基于测试平台进行搭建与扩展。为了满足自动化测试设备测试需求，本项目构建了一个服务器节点和多个测试资源服务节点的灵活配置扩展的平台构型，架构如图2所示。其中，测试服务节点可根据被测对象的需求灵活配置多个。硬件测试资源的功能板卡通过PXI总线、GPIB总线等多种仪器总线在测试服务节点集成。



自动化测试平台运行结构

自动化测试平台基于“请求-响应”交互模式。所谓“请求-响应”交互模式，是指主控端与服务节点的每次交互分为两步，首先是一方向另一方发送请求命令或请求信息，另一方根据接收到的不同请求给予不同的响应回复。例如，测试前，服务节点向主控端发送登录连接的请求后，主控端经过分析与决策，给予同意或拒绝的回复响应；测试中，主控端将测试用例打包成测试请求命令，服务节点给待测的设备或系统提供一个激励信号，并触发其内部的一系列运算、状态切换或内部的数据传递，最后对相应结果进行采集或测量，并将数据打包发给主控端。

在自动化测试平台分布式软件架构中，测试主控端作为自动化测试设备平台的逻辑控制中心与人机交互接口，部署测试资源管理功能模块、测试用例开发与管理功能模块、静态激励开发与管理功能模块、ICD导入加载功能模块、测试数据监控模块、试验记录与用户权限管理模块。其中，测试资源管理模块支持对当前测试环境中所有的I/O接口资源的集中配置与管理；测试用例开发与管理模块支持对测试用例的开发、维护与管理，实现在不同测试需求下对测试用例的快速配置；静态激励开发与管理功能模块，支持对静态激励快速配置，实现灵活直接的激励响应测试与故障排除；测试数据监控功能支持实时监控与被测对象交联的I/O资源的底层信号；测试主控端的数据库模块即试验记录与用户权限管理模块，对试验记录与数据进行存储，并支持回放导出等功能，同时对用户权限进行管理。

测试服务节点则集成测试资源驱动功能模块、模型加载与驱动模块、测试服务执行模块。本项目硬件测试资源主要为板卡仪器测试资源。测试服务节点通过解析来自主控端的数据收发以及测试执行控制命令，驱动硬件测试资源或仿真模型，实现自动化测试设备平台测试服务的具体执行。

测试平台软件功能结构如图3所示。



**测试设备软件功能结构图**

硬件被测对象即相关功能板卡、台式仪器。各硬件被测对象板卡可通过线缆、接线调试板组合构成拓扑级联结构，并通过PXI总线、GPIB总线及LAN等各类总线与测试服务节点通信。

## 3.2自动化测试平台功能设计

自动化测试设备平台从功能角度主要分为测试主控端与测试服务节点两部分。测试主控端软件部署于高性能计算机Linux系统环境中，通过测试主控计算机实现现场无线人机交互。测试服务节点软件部署于PXI测试系统、台式仪器、传感器网络等测试资源，与外部硬件被测对象功能板卡通过PXI总线连接。测试主控端与测试服务节点上分别部署多个功能模块；多个硬件被测对象功能板卡通过线缆、接线调试板组合构成拓扑级联结构，并通过PXI总线与部署在PXI控制器中的测试服务节点通信。自动化测试设备平台功能组成如下：



**图4 自动化测试设备功能组成图**

（1）测试执行功能

测试平台在执行测试项目/测试用例中所描述的测试过程为：从测试主控端到测试服务节点仪器驱动实现测试执行，且实时反馈测试结果。

测试平台针对不同测试需求提供多种测试模式（如单步、多步、循环等），且支持自动化测试，支持在不同测试模式下对系统进行调试，测试主控软件应当能连续正常工作至少8小时。

自动化测试设备平台的测试执行功能为测试平台较为重要的功能，其用于执行相关硬件设备的测试用例，支持基于产品软硬件通讯规约的测试，且实时反馈测试结果，具体要求如下：

a）支持相关硬件设备的测试验证需要；

b）支持测试项目/测试用例中所描述的测试过程的执行，从测试服务节点仪器驱动实现测试执行；

c）支持测试资源、仿真模型的驱动，并与被测对象实现动态的数据交互，在测试平台、测试资源与被测对象间实现数据流通；

d）针对不同测试需求为用户提供多种测试模式（如单步、多步、循环等），且支持自动化测试，支持用户在不同测试模式下对系统进行调试；

e）支持测试项目/测试用例的灵活加载，实现在被测产品、应用场景改变等其他导致测试项目/测试用例发生变化的情况下，只需更改测试项目/测试用例加载配置的可重配置功能实现，提高自动化测试设备平台的复用性；

f）支持基于软、硬件通讯规约对接收的动态数据进行实时解析，对数据进行显示与存储；

g）支持较复杂的测试逻辑模型的执行，实现对用例中复杂测试过程的准确解析与驱动执行；

h）支持测试执行的进程、结果、故障、告警等信息的实时上报及显示，并能够提示测试操作信息和被测产品的状态信息，为用户提供准确实时的测试执行状态信息；

i）支持测试执行随时开始、中止、停止的功能，提供用户在测试主控端对测试执行全过程高效快速的控制能力，实现测试执行全过程的完全受控；

j）提供可视化、易于操作的控制工具，以满足用户对上述功能的控制。

k) 具备测试故障中止功能，按照要求中描述的，连续三个不合格则停止测试。

l) 有多工位测试执行功能，按照多线程的形式实现并行的测试，对仪器进行分时复用。

m) 测试资源能够自动分配，当多工位执行测试任务时，能够按照一定的机制尽可能的减少测试的总时间，充分利用设备性能。

n) 支持序列号手动界面输入、自增序列号、扫码枪、序列号导出等功能。

o）支持计算公式管理，具有自定义录入、数据实时计算等功能

p）支持在云端汇集统一型号产品的数据进行一致性分析

根据测试执行功能的要求，本方案采用分布式架构能够凭借通信网络实现从测试主控端到测试服务节点的测试执行，主控端对测试用例中测试过程进关键字匹配解析，经通讯服务将测试执行指令传输至测试服务节点，测试服务节点匹配测试执行指令与仪器控制驱动，具体执行测试服务。测试主控端为用户提供测试执行全方位的控制能力，胜任多种测试模式、支持测试执行的随时控制，并对策是服务节点接受的动态数据进行动态解析。测试平台的支持测试项目灵活加载，具有良好的重配置性。

（2）测试资源管理与驱动功能

测试平台需具备测试资源管理功能，自动识别硬件资源，获取相关的硬件资源信息，进行硬件资源与ICD关联，支持各测试资源自检控制，与自检结果上报。

同时测试平台需实现测试资源的驱动，并与被测对象实现动态的数据交互，在测试主控软件、测试资源与被测对象间实现数据流通，测试主控软件的操作-响应时间不超过1秒。

自动化测试设备平台的资源管理与驱动功能用于实现对当前测试环境中所有I/O接口资源的集中配置与管理，并依据测试执行功能下发的测试命令驱动资源完成相应的测试行为，其具体要求如下：

a）支持当前环境下测试资源的自动识别，即无预装软件的仪器地址自动识别或手动地址录入，提供用户以所有可用测试资源的信息，并供测试用例开发使用；

b）支持各测试资源的状态及使用情况的监控，提供模型开发规范，包括接口要求、交互操作要求、关键字管理要求等。与非正常工作时故障信息的实时上报，通过底层仪器驱动反馈信息来实时向用户提供测试资源状态信息；

c）支持各测试资源自检控制，与自检结果上报，以此保证测试平台中测试资源的有效性与可靠性；

d）支持各测试资源的各个通道所连接设备的标记和查看，以及真件与仿真件的识别；

e）支持I/O资源的配置信息固化功能；

f）能够依据配置文件、通讯规约等，调用仪器驱动或数据交互操作接口，驱动测试资源进行数据收发、信号激励与测量等测试行为；

g）支持测试主控端与测试服务节点的数据交互功能，主要实现测试资源信息在服务器端的汇总；

h）支持基于多种常用测试行为执行；

i）能够依据开发人员的配置信息或ICD配置文件的信息对测试资源进行配置和管理；

j）提供可视化、易于操作的控制工具，以满足用户对上述功能的控制。

k) 支持分选机驱动，能够通过TCP/IP等向PLC发送指令、初始化等。

根据用户关于资源管理与驱动功能的需求，本方案依据测试资源描述文件进行当前环境所有测试资源的自动识别；通过自检函数完成上电自检与手动自检；通过异常函数实现实时获取各测试资源的运行状态。

（3）模型开发与运行服务

测试平台的模型开发运行服务为平台与被测件的通讯、测试过程提供一个模型开发和运行的环境，规范系统内部、外部仿真模型架构，并提供数据通道和运行机制，提高模型开发效率，具体需求如下：

a）支持模型的加载与运行；

b）提供模型开发规范，包括接口要求、互操作要求、关键字管理要求等；

c）提供模型数据接口与ICD消息、物理通道直接的绑定设置能力；

d）支持依据自动化测试用例执行指令或测试人员的手动指令，实现对模型的运行、停止控制功能；

e）支持仿真模型运行状态的监控功能；

f）提供可视化、易于操作的控制工具，以满足用户对上述功能的控制。

根据模型开发与运行服务的需求，测试平台提供外部数字仿真模型加载与运行环境，为实现模型灵活加载与可重配置，模型加载与运行环境具有规范的数据接口、互操作要求与关键字等。外部仿真模型应根据提供的模型规范进行封装，并匹配测试平台模型控制接口，强化用户对模型的控制与交互数据监听功能。

（4）测试用例开发与管理功能

测试平台需提供各级别的测试用例开发与管理功能，建立装调测试用例规范，支持基于测试关键字库的编辑、复用和生成功能；支持多种测试行为的编辑，如启动测试资源、启动收发、根据合格判据或判读逻辑进行结果判断等；支持串行、并行、循环等多种测试行为运行模式；支持测试用例的编辑、校验；支持测试报告生成。

测试平台的测试用例开发与管理功能用于实现测试用例的开发、维护和管理，便于用户快速配置测试用例，以配合不同的测试需求，其具体要求如下：

a）建立测试用例规范，提供各硬件设备的测试用例开发与管理功能；

b）支持基于测试关键字库的编辑、复用和生成功能；

c）支持用户通过测试关键字库中的关键字，将测试行为、测试逻辑与测试数据组合成测试用例；

d）支持模拟量、离散量等类型测试数据的内容编辑；

e）支持多种测试行为的编辑，如启动产品、启动收发、启动仿真模型、根据合格判据或判读逻辑进行结果判断等；

f）支持串行、并行、循环等多种测试行为运行模式；

g）支持依据用例开发人员建立并导入的ICD配置文件来组织测试用例中所需的数据激励的功能；

h）支持通过属性表实现用例的标签化，用于标签化的属性有：项目、产品、验证阶段、用例名称、用例描述、创建日期、创建人等；

i）支持根据测试用例项目、产品、验证阶段等维度的属性对现有的测试用例进行分类管理；

j）为后期的可视化、表格化的用例开发功能模块提供必要的软件接口，并提供上述模块的运行环境；

k）提供可视化、易于操作的控制工具，以满足用户对上述功能的控制。

L）支持的测试用例环节应该至少包括初始化、测试、终止等部分。

根据用户关于测试用例开发与管理功能的需求，本方案依据关键字库的描述文件搭建关键字库，其中包括行为关键字、数据关键字、行为模型关键字和判读模型关键字；在关键字库的基础上引导编译环境进行测试用例开发与编辑；通过测试用例属性的标签化实现对测试用例的分类保存与管理。

（5）静态激励功能

测试平台的静态激励功能多用于调试等非正式测试验收场景下，是与被测产品的静态数据通信过程，可分为单条测试消息激励与测试序列（多条测试消息按一定逻辑组成）激励。与测试用例相比，静态激励保留所有测试数据，但对测试行为与测试逻辑的要求较低，其具体要求如下：

a）支持以软硬件通讯规约为基础的单条测试消息激励与测试序列（由多条测试消息按照一定顺序组成）激励的功能；

b）支持基于变量的测试消息内容编辑及属性（如事件/周期发送、发送次数、发送周期等）编辑功能；

c）支持单条测试消息激励的情况下在线调参的功能；

d）支持测试序列激励的情况下，在激励前对组成该激励序列的各测试消息的内容与属性（如事件/周期发送、发送次数、发送周期等）编辑功能；

e）支持测试序列的保存功能，并添加相应属性实现测试序列的标签化，用于标签化的属性有：项目、产品、序列名称、序列描述、问题描述、创建日期、创建人等；

f）支持根据测试序列项目、产品等维度属性对现有测试序列进行分类管理的功能；

g）支持激励消息的自动打包与发送能力，测试平台自动将用户编辑好的激励消息通过通信网络传输至测试服务节点，而后通过硬件资源发送至测试对象；

h）支持根据当前发送的激励消息，驱动测试资源的能力，实现将数据发送至被测产品的功能；

i）支持静态激励的开始、中止、停止的控制功能；

j）提供可视化、易于操作的控制工具，以满足用户对上述功能的控制。

根据静态激励功能需求，测试平台关键字库配置文件为用户提供静态激励编译环境，通过属性参数的设置定义特定静态激励的属性，静态激励用于标签化的属性设置为测试序列的保存分类与查询提供了基础；测试平台支持对静态激励编辑环境中用户以添加的激励消息进行自动打包发送，由测试执行功能支持周期型或事件型静态激励测试。

（6）ICD导入与加载功能

测试平台需具备ICD导入和管理功能，提供模型数据接口与ICD消息、物理通道直接的绑定设置能力；支持多种总线ICD协议；提供图形化的编辑界面，支持数据的检索和查找。

测试平台的ICD导入与加载功能用于测试主控端与测试服务节点交互协议封装与解析，其具体要求如下：

a）支持各类ICD配置文件（xml文件）的灵活加载功能；

b）能够根据ICD配置文件的建立规范对测试消息进行结构还原与显示；

根据ICD导入与加载功能的需求，本方案采用基于ICD接口控制文档的通讯交互协议动态加载机制，以统一的接口控制文档为基础，建立多种测试主控端与测试服务节点交互协议管理系统，使测试软件开发者不直接接触交互协议。避免程序中硬编码通讯交互协议带来的修改、调试的困难，采用软编码方式实现通讯交互测试，能够最大限度地减少由于ICD协议变更对测试软件的影响，避免程序中硬编码通讯交互协议带来的修改、调试的困难。

（7）测试数据监控功能

具备数据监控、支持，以列表、波形等形式显示监控结果，并以固定速率（不高于500ms）刷新当前监控结果；支持监控历史数据的保存与显示功能；实现测试执行的进程、结果、故障、告警等信息的实时上报及显示，并能够提示测试操作信息和被测产品的状态信息，为用户提供准确实时的测试执行状态信息。支持监控数据txt/Excel格式的保存与导出功能。

测试平台的测试数据监控功能用于监控所有与被测对象交联的I/O资源底层信号，其具体要求如下：

a）支持以一个完整的消息包为单位，对当前环境下所有与产品通讯的测试资源接收的消息的监控功能；

b）支持监控筛选功能，具体功能为：按照被测对象类型筛选：运行时只监控选定被测对象类型的数据，并可以支持多个被测对象数据同时监控；

c）支持数据监控的开始、暂停等控制功能, 实现测试执行的进程、结果、故障、告警等信息的实时上报及显示；

d）支持以列表、波形等形式显示监控结果，并以固定速率刷新当前监控结果；

e）支持监控历史（本次软件运行的全周期）的保存与显示功能；

f）支持监控数据txt/Excel格式的保存与导出功能；

g）提供可视化、易于操作的控制工具，以满足用户对上述功能的控制；

根据用户关于实验数据记录功能的需求，本方案搭建Access和Mysql数据库，将原始数据及解析数据实时记录存储，通过在数据列表中添加被测对象类型等属性，可以支持数据的筛选。同时支持数据有选择回放及导出功能以方便用户有针对性地分析与排故。

（8）试验记录管理功能

测试平台的试验记录管理功能用于存储及管理所有在试验前、试验过程中及试验后产生的操作记录、测试结果等信息，其具体要求如下：

a）与数据库进行关联，以完整保存所有的与测试试验有关的信息；

b）存储如下几种类型的信息：被测产品信息、操作人员信息、操作记录信息、测试结果信息、故障及问题信息；

c）支持测试结果信息与故障及问题信息的查询功能，查询维度可以包括操作人员、被测产品等；

d）支持测试结果信息、故障及问题信息的保存与导出功能，导出形式可以为txt、xml、xls、word等；

e）提供可视化、易于操作的控制工具，以满足用户对上述功能的控制。

f) 利用Access核MySql数据库进行数据存储，对超差数据进行单独存储。

根据试验记录管理功能需求，测试平台存储试验记录信息，提供用户以历史记录为后续验证服务，在每次测试平台使用中，将操作人员、被测产品、操作记录、测试结果、故障与问题等信息存储于数据库中，支持相关权限用户根据多维度的信息对实验记录进行查询与特定文件格式（如txt）导出。

（9） 用户权限管理功能

测试平台需具备用户管理功能，支持设置的操作人员角色应当包括：管理员、开发与调试人员、测试使用人员等，根据操作角色不同，对操作人员进行权限划分的功能；

测试平台的用户权限管理功能用于对所有使用该测试平台的人员进行角色分配、权限设置及管理，其具体要求如下：

a）与数据库进行关联，以完整保存所有的操作人员信息，可以以工号作为权限账号；

b）支持设置的操作人员角色应当包括：管理员、开发与调试人员、测试使用人员等；

c）支持根据操作角色不同，对操作人员进行权限划分的功能；

d）支持根据操作人员的权限不同，禁用或启用相关功能界面/模块，或可以根据操作人员的权限不同，调显不同的平台使用界面的功能；

e）支持为最高权限用户提供用户的注册、编辑与删除功能；

f）为不同的用户或操作场景定义不同的人机交互界面，提供各个操作场景下的操作导向，并有效组织各个功能模块；

g）提供可视化、易于操作的控制工具，以满足用户对上述功能的控制。

h) 支持加密安全相关操作。

根据用户关于用户权限管理功能的需求，本方案在平台启动后测试执行前，会对用户信息进行采集，该用户信息与数据库、测试用例库、平台各模块相关，根据用户的不同权限加载不同的功能模块，保证平台对不同用户的适用性，同时数据库和测试用例库根据不同权限暴露给用户的程度也不同，保证数据库和测试用例库的安全性。

## 3.3测试设备主控端设计

测试设备主控端设计如图5所示。人机交互界面为表示层对象，它提供了用户操作界面，用户可通过该界面控制测试主控节点的启动与关闭以及设置相关参数，用例生成、静态激励、关键字库、资源管理、ICD加载与数据监控与存储等功能模块的上层应用程序界面也集成于此。在测试过程中，界面上会显示测试主控端与测试服务节点之间的会话内容，包括已经连接到测试主控端的测试服务节点的基本信息。

逻辑控制是人机交互界面与各功能模块交互的桥梁，实现用户对各模块的有效调度。向下给各个模块传递来自人机交互界面的控制信息，向上传递来各模块的信息与结果，进行实时显示，供用户决策。

资源管理模块是测试主控端对测试服务节点当前拥有所有测试资源的统筹与管理。本方案将现有测试资源全部反射到主控端，在主控端产生一一对应的虚拟测试资源对象，通过测试资源列表进行管理，并将所有测试资源呈现给用户，方便用户对现有资源的掌控。关键字库模块是用户进行测试用例开发的前提与基础。本方案为用户提供了常用的关键字，同时提供关键字库的编辑功能，便于用户随时添加或修改关键字库。测试用例的开发通过可视化用户二次开发交互接口进行串行、并行、循环等编辑，开发的依据是关键字库。测试用例的管理是对编辑开发好的用例进行规范化的保存、修改等，方便用户后期重复利用。静态激励的开发直接在主控端界面输入参数进行编辑，静态激励的管理与测试用例管理基本一致。数据的实时监控支持多重数据筛选选择。原始数据的解析用到ICD，ICD模块主要是针对ICD工具产生的XML文件进行加载。数据库模块主要对测试数据、操作信息以及用户信息进行保存，并支持数据有选择回放。

通信服务为隶属于通信网络功能模块，它与测试服务节点中的通信模块一起，构成订阅/发布的通信架构，从而实现了测试服务节点和测试主控端间的通信。

**图5 测试主控端功能及逻辑交互设计**

### 3.3.1测试资源管理与监控模块设计

根据装调测试需求，该系统中的测试资源主要包括PXI测试系统、台式仪器、传感器网络、大屏幕显示输出等，有些部署在远端，有些部署在本地，而用户对这些测试资源的分布并不敏感，因此主控端需对当前所有测试资源进行管理。资源管理功能隶属于资源管理层。考虑到用户的需求，该功能模块主要分为测试资源识别、测试资源自检与测试资源监控三部分。



图6 资源管理功能模块示意图

（1）测试资源识别

测试资源识别的任务有两个：识别当前环境中所有测试资源。

测试资源识别的技术实现主要依托两方面：对测试资源的描述以及对资源识别函数的设计。

对测试资源的描述是资源识别的依据，需进行规范化设计，为方便后续测试资源的扩展，本方案充分考虑各测试资源的共性，提取测试资源名称、测试资源类型作为对测试资源的描述。

资源识别函数的设计是资源识别的手段，各测试资源对资源识别命令做出响应，资源识别函数处理这些响应，可得到测试资源的识别结果。

测试资源识别的具体流程为：测试主控端发出测试资源识别命令后，平台将读取测试资源响应结果，调用资源识别函数进行处理，将各自的识别返回值进行界面显示。

主控端的测试资源管理模块将识别到的测试资源以表格的形式显示，显示的项包括当前换进中所有的测试资源、每个资源所有通道的状态及使用情况、每个资源所有通道连接设备的名称及识别真件和仿真件。同时支持用户对该表格中的资源进行保存、删除及修改等管理功能。

（2）测试资源自检

测试资源自检的任务为：检查当前所有测试资源是否可以正常使用。

测试资源自检的具体实现主要依托统一的自检函数。测试主控端发送已识别资源的自检指令，相应资源的反馈信号经过自检函数处理，可判断资源是否可以正常使用。

测试资源自检的具体流程为：在测试资源识别完成后，发送需要进行自检的测试资源的自检指令，相应资源的反馈信号作为自检函数的输入，并将自检函数的返回值返回到界面，供用户决策。

（3）测试资源监控

测试资源监控的任务为：实时监测当前环境中所有测试资源的运行状态，包括是否正在运行，是否出现故障等。

测试资源监控的具体实现主要依托该测试资源的所有驱动函数。本方案在每个驱动函数执行完成后对返回值进行判定是否为执行异常，并将异常值通过异常函数赋给异常属性。该异常属性包含三种状态值，0代表该资源未被使用，1代表正常，2代表异常，并将返回异常的函数名作为返回值，便于用户明确是哪个函数异常。

测试资源监控的执行贯穿于测试的全过程，只要对某个或某些测试资源进行驱动，都会有异常检测，用户界面都会有实时的测试资源当前状态。

### 3.3.2测试用例开发与管理模块设计

测试平台的测试用例开发与管理功能用于实现测试用例的开发、维护和管理，便于用户快速配置测试用例，以配合不同的测试需求。该功能模块隶属于测试过程层，根据测试需求，本模块主要分为关键字库、测试用例开发和测试用例管理三部分。

图7 测试用例开发与管理模块组成

（1）关键字库

关键字库模块的主要任务为：关键字库的生成、修改与管理。

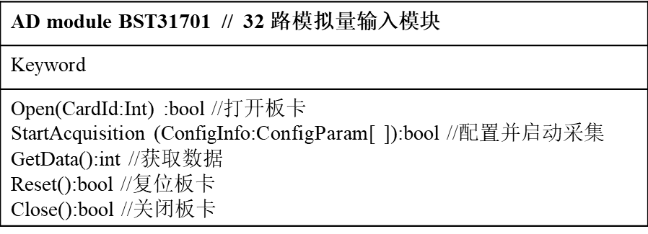
关键字库中的关键字库类型主要为：关键字主要完成测试数据收发、测试资源自检、板卡切换以及板卡功能控制等行为。

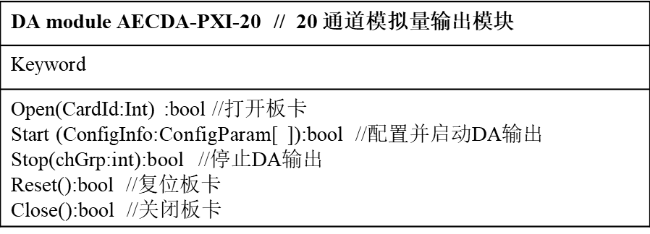
关键字库的实现主要依托：关键字构建规范、关键字类、关键字库管理列表类。

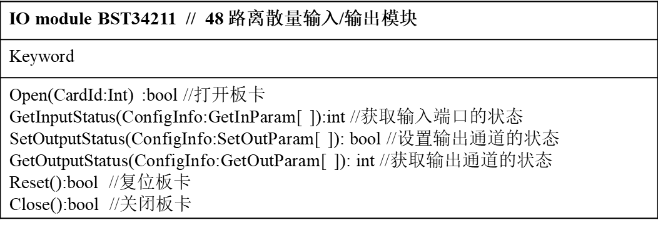
根据关键字的主要类型及共同特点明确关键字构建规范：关键字取关键字类型、关键字名称、关键字参数、测试资源名称等特征，组成关键字库，供用户进行测试用例的开发；同时用户可以根据关键字库的管理界面，进行手动选择添加时是否对关键字库进行添加或修改等，修改时回弹出测试资源的管理界面，供用户选择需要添加或修改的关键字所用到的那些测试资源及测试资源的驱动函数（此处由平台已导入的ICD决定）。当然关键字库管理界面会显示当前平台中包含的所有关键字。

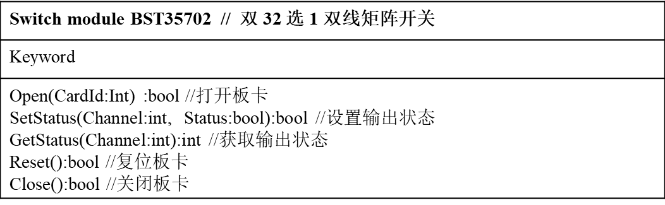
关键字类是关键字对象实例化的依据与前提。根据关键字构造规范，该关键字类主要包含关键字名称、关键字类型、关键字参数、测试资源名称等属性。

关键字库管理列表类是关键字库管理的依据与前提。该类中主要包含了对当前平台中所有关键字的一些操作，包括获取关键字各种属性、添加或修改关键字库等方法。

****

****

****

****

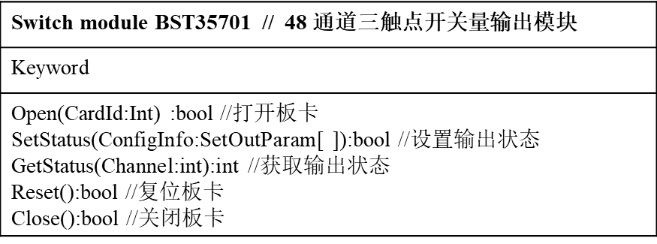
****

图8 板卡基础功能关键字示例

（2）测试用例开发

测试用例开发的主要任务为：测试用例的开发支持串行、并行、循环等多种测试运行模式；用例编辑能力必须支持各种被测对象的测试报文的编辑。

测试用例开发的实现主要依托：关键字库、编译环境，其中关键字库已在上节介绍。本方案对测试用例开发拟在图形化环境下编辑，用户调用关键字库中的测试方法，结合基本用例运行逻辑种类实现测试逻辑，将测试行为、测试模型、测试数据组合成测试项目，编译生成动态链接库由测试平台解析执行测试服务，能很好地完成测试用例的并行、串行、循环等运行逻辑的实现。提高扩展能力，方便无编程基础用户使用自动化测试设备平台开发测试用例。

（3）测试用例管理

测试用例管理的主要任务为：对现有测试用例项目进行分类管理，实现测试用例的复用。

测试用例管理的实现主要依托：测试用例类的属性标签化。用于测试用例开发编辑的类中定义多个能够表征测试用例的属性，主要包括测试资源、用例名称、用例描述、创建日期、创建人等。根据测试用例的属性实现对测试用例进行分类保存。

### 3.3.3静态激励与管理模块设计

尽管该测试平台已经设计开发了测试用例功能模块，但是根据测试特殊需求，例如某场景需要快速激励或迅速排故，还需设计开发静态激励，因为静态激励比测试用例更直接、更快速、周期短以及开发成本低等优点。根据被测硬件设备测试的需求，静态激励开发与管理模块主要包括静态激励开发与静态激励管理两大模块。

（1）静态激励开发

静态激励开发的主要任务为：静态激励应具备基于变量的测试消息内容和属性的编辑；单条测试消息激励时能在线调参；对测试消息进行结构还原与显示。

静态激励开发的实现依托：资源管理模块和ICD模块。静态激励的编辑与调试基于人机交互的界面，实现静态激励的开发。静态激励报文要手动编辑，为了方便用户编辑，将激励报文与ICD信息进行关联，用户根据ICD信息直接输入状态量或物理量等业务数据。系统自动编排整个报文的二进制字符串，在激励过程中，自动打包数据，保证数据的正确性。

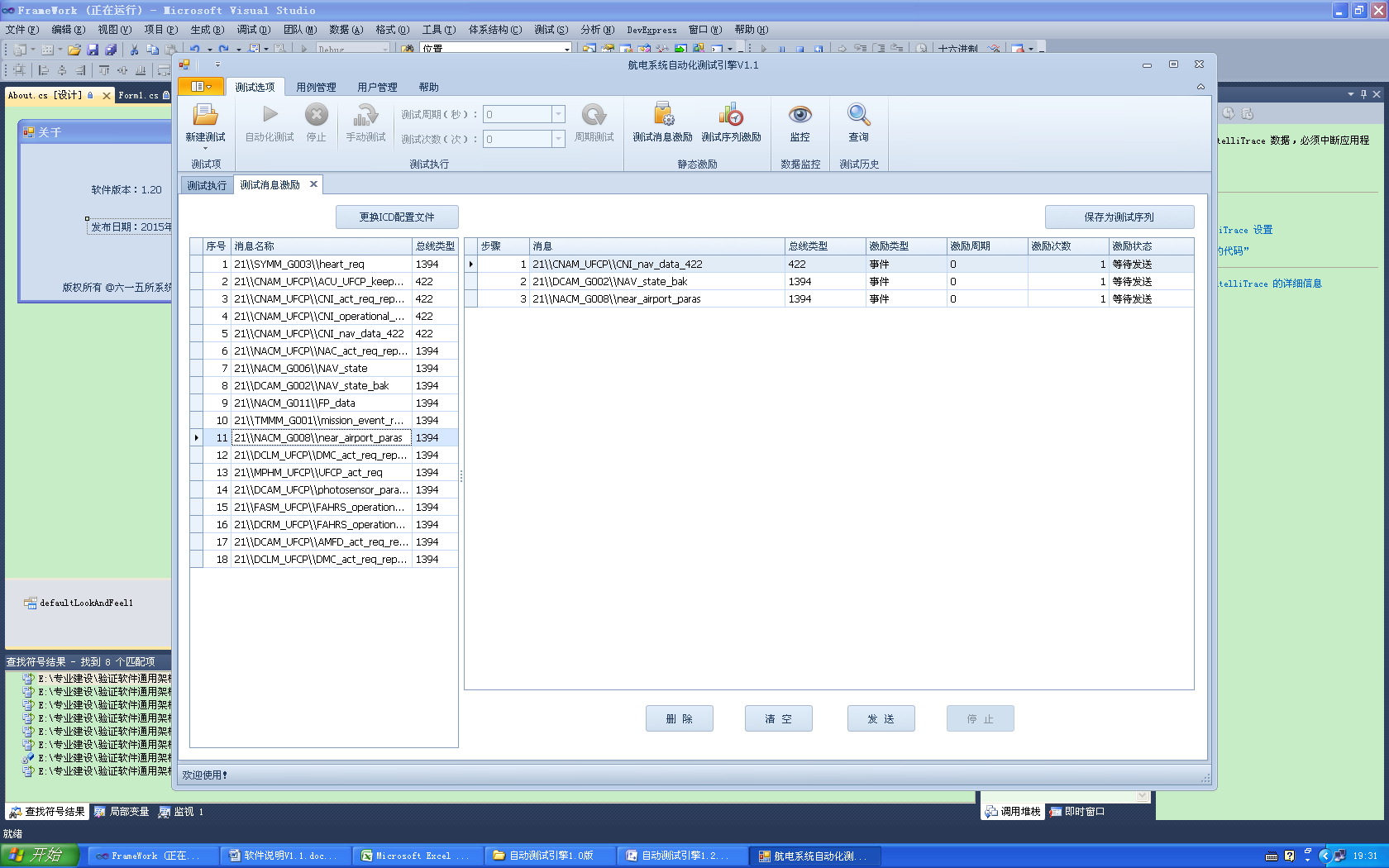


图9静态激励开发界面样张

（2）静态激励管理

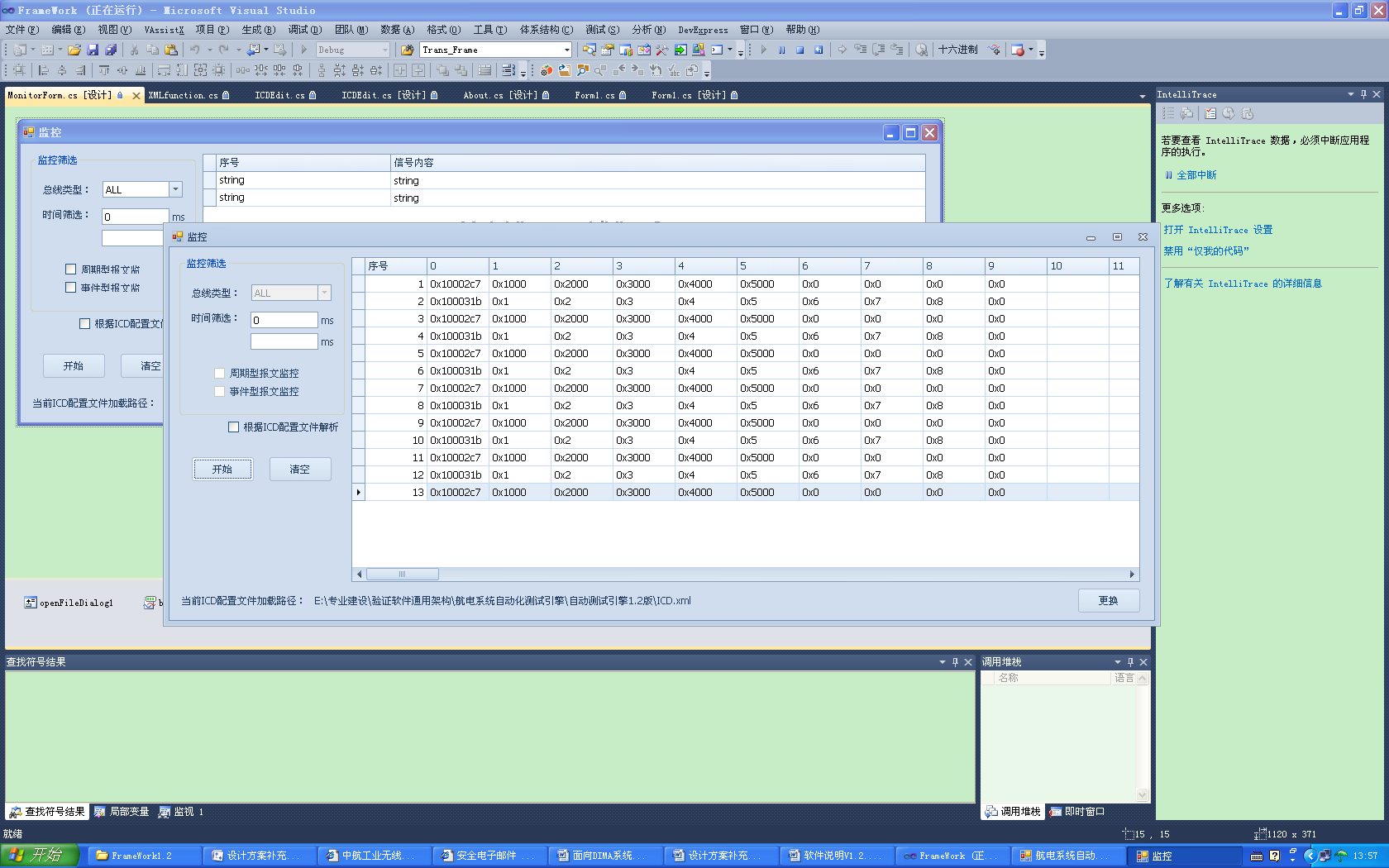
静态激励管理的任务主要为：对现有静态激励条目进行分类管理，实现静态激励条目的复用。

静态激励管理的实现主要依托：静态激励的属性标签化。用于静态激励开发编辑的类中定义多个能够表征静态激励的属性，主要包括测试资源、验证阶段、静态激励名称、静态激励描述、创建日期、创建人等。根据静态激励的属性实现对静态激励进行分类保存。

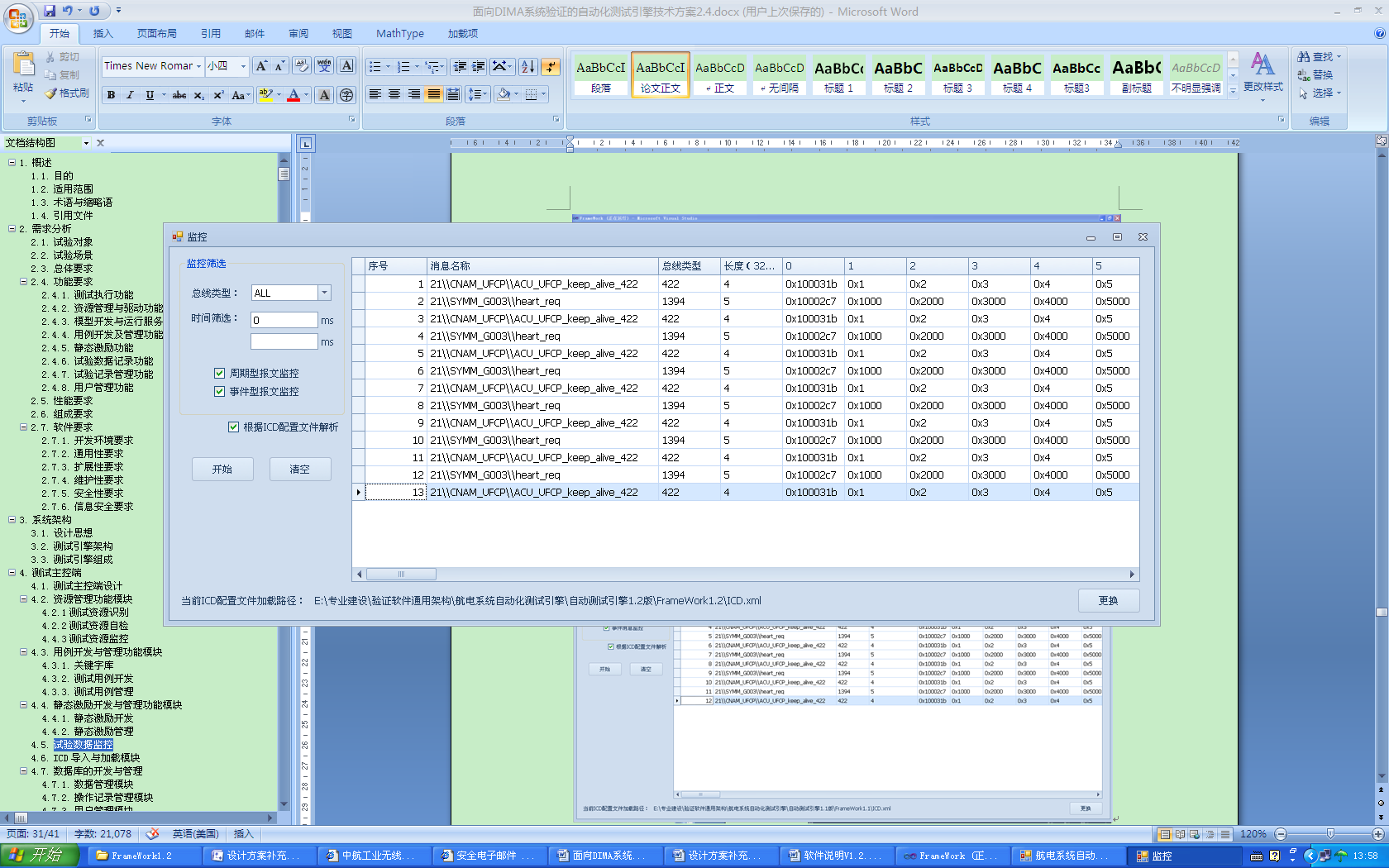
### 3.3.4测试数据管理模块设计

系统运行时主要产生两种数据：原始的采集数据和解析后的数据。每个测试服务节点在采集后会把原始数据打包发给测试主控端，主控端在接收后一方面按用户的需求先将部分被选择的参数在界面予以显示，另一方面将所有数据存储在数据库里。

时钟拟定在主控端，主要作用是驱动整个测试平台，保障主控端和各个测试服务节点的有序运行。在时钟的每次驱动下，各测试服务节点会主动所有对象的参数都采集一遍，然后发给主控端。在发给主控端之前需要对数据进行加前缀处理，前缀是不同的被测对象的名字，这样当数据发送到主控端时就能按照不同的被测对象前缀进行分类。据监控用户界面如图10所示。



**（a）原始数据显示**



**（b）解析数据显示**

图10 数据监控用户界面样张

测试数据管理功能模块提供数据的筛选、解析与导出功能。数据的筛选同时支持按照被测对象筛选和时间筛选，当用户选择按被测对象筛选时，界面会给出目前所有的被测对象名供用户选择，然后只显示所选的被测对象类型的数据；当用户选择按照时间筛选时，只要给出起止时间段就可以。用户数据的显示支持只原始数据显示、只解析数据显示和两种数据同时显示三种显示模式。数据的解析也可根据被测对象类型筛选条件与时间筛选条件对选择的原始报文进行解析。数据导出功能支持用户预先定义导出数据筛选条件与导出路径及文件等，这样在测试执行过程中，符合条件的数据会自动保存到指定文件中。

### 3.3.5数据库功能模块设计

数据库存储和管理着在系统运行前、运行中以及运行后的所有操作记录和测试数据，包括测试的原始数据、解析后数据、用户的操作记录、用户信息等，是总控软件乃至整个分布式实时仿真平台正常运行的前提和保证。

考虑到在分布式系统对数据库访问的安全性、便捷性和集中性的需求，本系统的数据库应搭建在服务器端，这样有利于用户在系统登录时能够快速链接访问到数据库。同时，因为系统采用的客户端/服务器模式是星型结构，每个客户端都会与服务器时刻保持通讯，这样既能在测试运行中方便地观测和储存每个客户端的实时测试数据和用户的操作记录，也能在测试运行后在服务器端进行数据的筛选、显示和导出等功能。数据库在系统中的位置和形式图11所示。



图11 数据库模块信息交互图

本系统的数据库包含以下功能模块：数据管理模块、操作记录管理模块和用户权限管理模块。

（1）数据管理模块

系统运行时主要产生两种数据：原始的采集数据和解析后的数据。数据的采集和解析都在客户端完成，每个客户端在采集和解析后会把这两种数据打包发给服务器，服务器在就收后一方面按用户的需求先显示部分被选择的参数在界面予以显示，另一方面将所有数据存储在数据库里。

数据在数据库中以一张表的形式存储，时间作为表的主键，这样每个客户端的数据就能按照时间的顺序在表中进行存储，也方便后续以时间作为筛选条件进行数据的回放。同时每一个参数都会用相邻两列来存储，列名分别为“参数名\_Raw”和“参数名\_processed”，分别为原始数据和解析后的数据。

数据管理模块的具体功能包括：支持测试运行中每个客户端数据的监控和录入、支持按照时间和被测对象进行过滤和筛选；支持按照不同被测对象或时间进行分块显示；支持数据的有选择的导出。

（2）操作记录管理模块

本模块旨在将用户在测试前、测试中和测试后的所有操作记录以日志的形式存储下来，为后期进行测试的分析提供有效的支持。操作记录在数据库中的存储形式也是一张表，这张表只有操作时间和字符串形式的操作记录，操作记录包含测试前登录时的操作人员信息、测试中服务器端的控制信息和每个客户端产生的测试结果，以及故障信息等所有和操作相关的信息。

同数据管理模块一样，操作记录模块也同样支持录入、筛选、显示的导出等功能。

（3）用户权限管理模块

该模块旨在对测试平台的用户权限进行管理,用于保证对本系统的合法和安全的使用。用户权限管理模块保证系统安全性的具体功能表现在：支持用户权限管理模块和数据库的关联，保障用户在登陆系统时将用户信息录入数据库的操作记录模块；支持对操作人员的权限划分，不同的权限的用户所能行使的操作命令不同以及访问的数据库级别的不同；支持用户的注册和删除功能，注册后会将用户名、用户密码、用户级别存储进数据库，方便以后登录时的便捷性。

此外，用户权限管理模块根据用户角色或权限的不同，为用户配置不同的功能模块组合，提供所需功能模块，并屏蔽用户使用过程完全不需要或无权限使用的模块，为每个用户提供既满足需求又无赘余的自动化测试设备平台，保证自动化测试设备平台对不同用户的适用性，同时数据库和测试用例关键字库根据不同权限暴露给用户的程度也不同，保证数据库和测试用例库的安全性。

### 3.3.6 ICD导入与加载模块设计

测试用例开发将针对抽象的逻辑资源，涉及具体测试主控端与测试服务节点交互协议和信号接口，因此被测硬件设备交互将涉及测试主控端与测试服务节点交互协议封装与解析，而被测硬件设备远程数据采集单元测试必然设计ICD信号接口配置与转换。

ICD数据源

数据合成

数据解析

测试资源

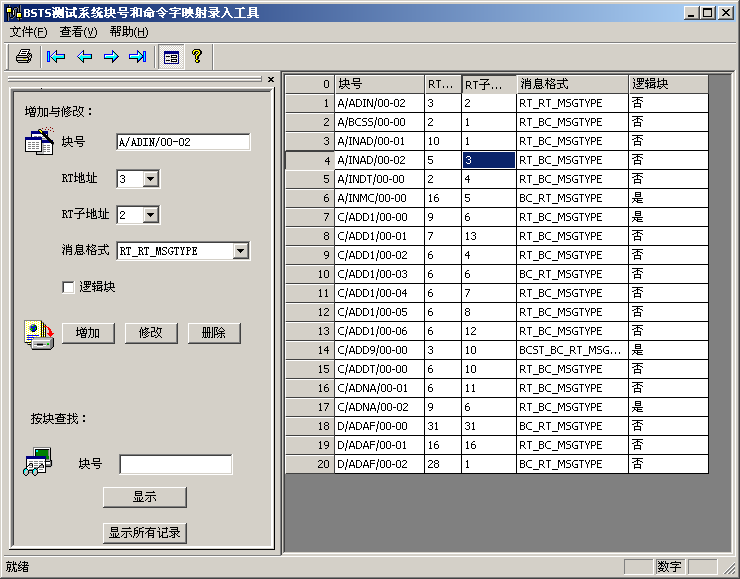
数据

图12 ICD功能模块示意图

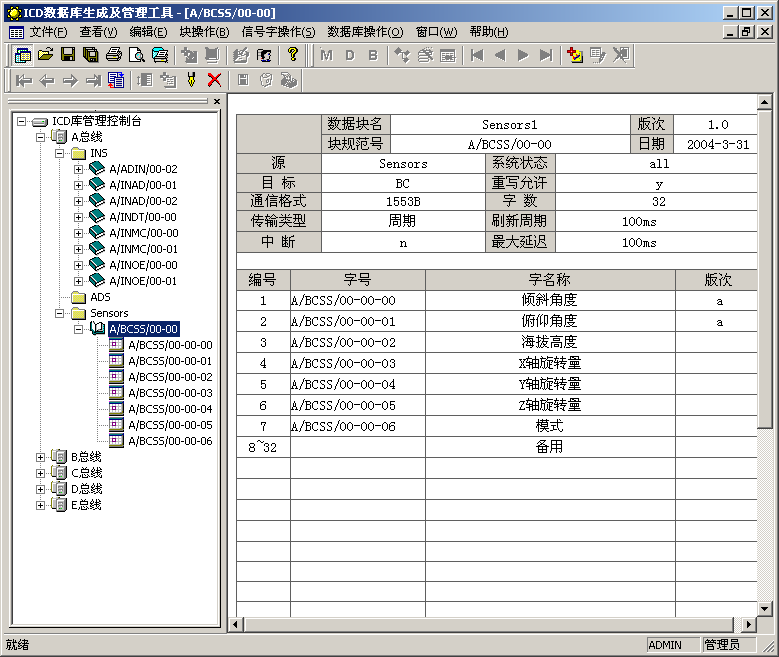
为适应测试主控端与测试服务节点交互协议的变更需要，本方案中采用基于ICD接口控制文档的通讯交互协议动态加载机制，以统一的接口控制文档为基础，建立多种测试主控端与测试服务节点交互协议管理系统，使测试软件开发者不直接接触通信协议。避免程序中硬编码通讯交互协议带来的修改、调试的困难，采用软编码方式实现测试主控端与测试服务节点交互协议通讯测试，能够最大限度地减少由于ICD协议变更对测试软件的影响，避免程序中硬编码通讯交互协议带来的修改、调试的困难。

本方案中需要导入的ICD文件，建议采用XML格式，因此测试平台应专门设计一个ICD加载器，该加载器实际是一个读取XML文件的一个类。测试执行前，根据不同的测试资源加载环境中所有的测试资源对应的ICD文件，并实例化出相应的ICD对象，避免在测试执行过程中不停地加载ICD引起性能的下降。实例化的ICD对象作为数据解析的依据，实现平台对原始数据的解析。同时，根据静态激励小节中介绍，ICD也作为静态激励编辑的依据。

下图13、图14，演示ICD信息管理与配置接口形式与交互。



**图13 ICD信息管理样张1**

 图14 ICD信息管理样张2

## 3.4测试设备服务端设计

测试服务端设计支持测试执行随时开始、中止、停止的功能，提供用户在测试主控端对测试执行全过程高效快速的控制能力，实现测试执行全过程的完全受控。

测试服务端驱动器的结构设计如图15所示。表示层提供了测试服务节点的公共操作界面，它提供以下一些功能：

a）测试资源的加载：通过向导式的页面引导用户将客户端二次封装的测试资源以正确的形式加载到应用程序中；

b）测试运行流程的控制：控制测试的启动、暂停和终止；

c） 测试交互关系的设置：通过操作面板向测试服务器发起连接，设置与其它测试服务节点交互的互操作关系；

d） 显示测试信息：在测试主控端操作面板上显示的信息包括本节点与测试服务器的会话内容，当前的测试（对应测试板卡）、仿真时刻（对应模型）以及所装载的测试服务节点的基本信息。



图15 测试服务端设计

在测试服务节点驱动器的架构中，逻辑控制是位于测试逻辑层的对象，它负责处理本测试服务节点的测试逻辑。与测试服务器中的测试控制器对象相同，逻辑控制在程序内部也起到了中介者的作用。逻辑控制具有如下的功能：

a）获取底层对象提交的测试交互信息，并按照相关的测试逻辑进行处理。

b） 接收来自上层的用户指令，调用底层的功能模块完成对测试资源驱动。

通信服务隶属于通信网络功能模块，它与测试主控端中的通信服务器一起，构成订阅/发布的通信架构，从而实现了测试服务节点和测试主控端间的通信。

### 3.4.1测试资源驱动模块设计

本设计方案在测试资源驱动设计中，在仪器厂家提供的驱动、API等资源之上，对底层仪器控制函数进行基于功能的二次封装，封装函数与被测对象关键字对应结合。新封装的函数具有统一的函数形式和明确的函数名称，对外提供规范的调用接口，同时增加用于资源管理的资源注册、自检等函数接口，供测试资源管理器调用，这样可以屏蔽和补偿不同厂家仪器之间的差异，提高测试系统的仪器互换和资源扩展能力。通过测试资源封装保证了测试驱动接口的一致性，也为后续系统扩展、重用、资源互换奠定了基础。

仪器模块在系统软件中执行具体的仪器操作任务，这些仪器模块以动态链接库的格式存在，软件在初始化时使用测试资源加载模块动态加载，加载后使用统一的访问方式读取模块的各项参数，保存到测试资源列表中。

### 在具体实现中通过测试主控端传来的关键字与测试服务节点的接口函数匹配耦合，实现资源驱动功能。

### 3.4.2用例测试服务执行模块设计

该模块针对不同测试需求提供多种测试模式（如单步、多步、循环等），且支持自动化测试，支持在不同测试模式下对系统进行调试，测试主控软件应当能连续正常工作至少8小时。支持测试执行随时开始、中止、停止的功能，提供用户在测试主控端对测试执行全过程高效快速的控制能力，实现测试执行全过程的完全受控。

测试服务节点对接收到的来自测试主控端的测试命令进行解析，匹配测试资源与通道以及测试驱动函数，实现对应的真实测试资源进行准确驱动。

在具体实现中，测试用例中所描述的测试过程，经过通信网络中测试用例数据包的传输，经过测试服务节点逻辑层处理，最终到达测试服务节点的用例测试服务具体执行模块进行解析。

用例测试服务具体执行模块解析测试用例数据包中的测试过程与测试数据，并根据其中的关键字匹配相应硬件资源与测试行为驱动，在测试用例测试过程描述与测试资源测试行为之间建立直接的对应关系。针对测试过程中的每一个测试行为，使用反射机制调用其特定测试资源的具体测试行为驱动函数，进而完成测试行为的物理实现。

### 3.4.3静态激励执行模块设计

测试服务节点的静态激励执行模块对接收到的来自测试主控端的静态激励进行解析，匹配测试资源与通道以及测试驱动函数，实现对应的真实测试资源进行准确驱动。

在具体实现中，静态激励条目中所描述的测试过程，经过通信网络中静态激励条目数据包的传输，经过测试服务节点逻辑层处理，最终到达测试服务节点的用例测试服务具体执行模块进行解析。

用例测试服务具体执行模块解析静态激励条目数据包中的测试过程与测试数据，并根据其中的关键字匹配相应硬件资源与测试行为驱动，在静态激励条目测试过程描述与测试资源测试行为之间建立直接的对应关系。针对测试过程中的每一个测试行为，使用反射机制调用其特定测试资源的具体测试行为驱动函数，进而完成测试行为的物理实现。

### 3.4.4测试对象数据获取模块设计

在测试过程中，测试平台中测试服务节点应支持对测试对象数据的实时获取与向测试主控端进行数据传输的功能。测试对象数据获取则依托对被测对象板卡进行数据采集。为提高测试平台性能，且满足各测试资源间不互相造成影响的需求，在测试服务节点为每一个测试资源开辟一个新线程用于数据采集与数据侦听。子线程中获取的测试对象数据原始报文，通过主线程控制通信服务执行，传输至测试主控端，并由主控端的测试数据管理模块进行数据储存。

此模块具备数据监控、支持，以列表、波形等形式显示监控结果，并以固定速率（不高于500ms）刷新当前监控结果;支持监控历史数据的保存与显示功能;实现测试执行的进程、结果、故障、告警等信息的实时上报及显示，并能够提示测试操作信息和被测产品的状态信息，为用户提供准确实时的测试执行状态信息。支持监控数据txt/Excel格式的保存与导出功能。

测试服务节点执行测试用例与静态激励后产生大量底层原始数据，测试服务节点此时不进行数据解析，根据不同总线将数据进行打包，在数据包里打上不同总线的标签，然后返回给测试主控端。

## 3.5通信中间件及交互共享数据池设计

拟选用通信中间件DDS（数据分发服务）构成分布式交互测试共享数据池，实现以测试数据为中心的多用户、多测试资源、分布并行测试。  
 DDS规范是由对象管理组织（Object Management Group, OMG）定义的服务，用于部署一个以数据为中心的发布-订阅者模式系统，与数据库对历史数据的检索关注不同，DDS关注的是未来还未生成的数据，只需通过确定数据主题类型便可持续接收对应的数据信息，由此在分布式系统中建立有效可靠的信息分发，在资源处理方面具有如下优势：

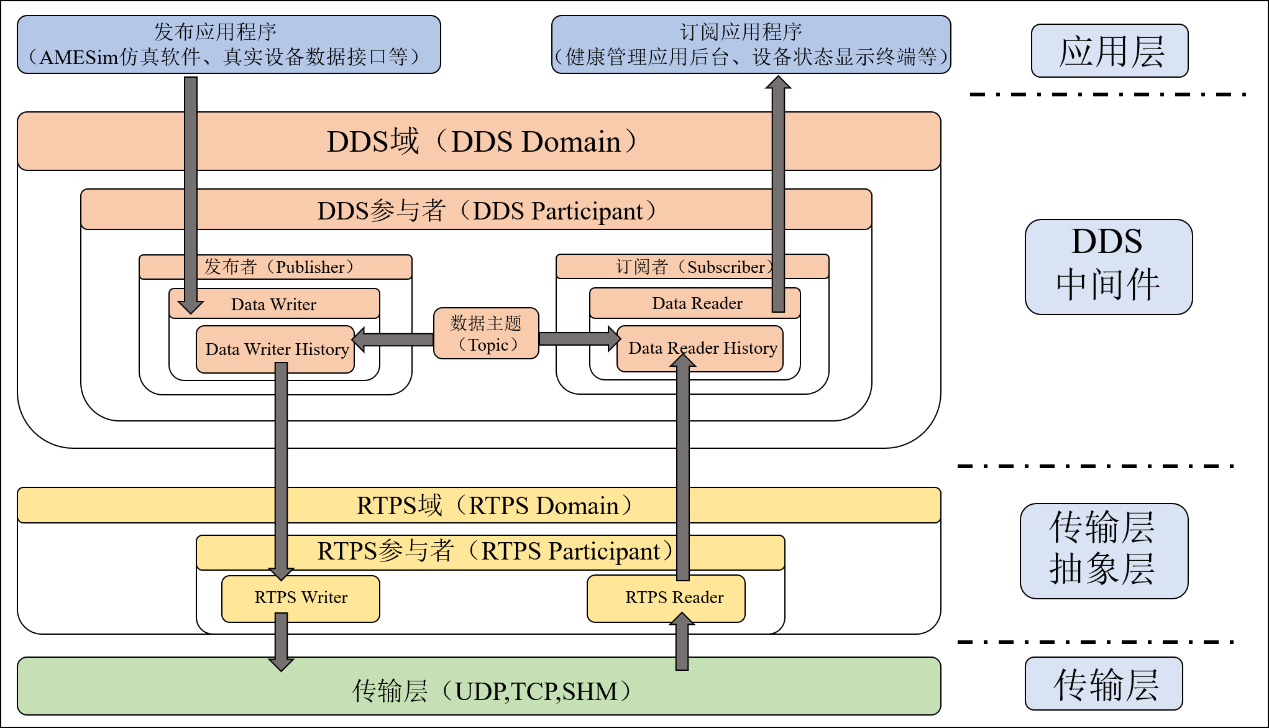
（1）易于集成：以数据为中心，数据模型可扩展，隐藏应用程序连接细节，支持嵌入式、Web等多应用程序；

（2）性能效率：延迟小、吞吐量大，适用于工业物联网、国防、自动驾驶等大数据量场景；

（3）安全性和可扩展性：支持多种通信协议，支持数据加密（TLS）等实现；

（4））支持服务质量策略（QoS）：提供过滤、持久化等数据服务；

DDS作为一个以数据为中心的服务，建立在“全局数据空间”的概念之上，是位于应用层与传输层中的抽象层。在上部应用层，提供信息的应用层程序发布信息到数据空间后，DDS中间件将信息传播给已订阅相关主题的应用层程序，即订阅者。在下部传输层，DDS中间件依托于实时发布订阅（Real-Time Publish Subscribe, RTPS）协议，RTPS协议作为传输层的抽象，为DDS提供支持UDP、TCP和共享内存（Shared Memory, SHM）等多种方式的单播或多播通信方式。下图为DDS中间件架构示意图。



**图17 DDS中间件架构示意图**

测试平台基于DDS通信中间件技术搭建订阅-发布模式数据共享网络，通信网络作为测试主控端与测试服务节点之间的桥梁，其上传输的消息主要分为通讯相关消息与测试相关消息。

通讯相关消息包括测试主控端接收的测试服务节点登录退出等请求信息与该节点上连接的测试资源信息，以及由测试主控端向测试服务节点发送的通讯服务控制命令与时钟同步信息。测试主控端接收服务节点请求信息并对其进行处理，节点登录后，测试主控端将测试服务节点相关信息进行显示、存储与维护，节点退出后，测试主控端则停止对该测试服务节点信息的维护。通信服务控制命令主要为测试主控端通信服务的开启关闭等动作后，向测试服务节点发送的通信服务动作信息。

测试相关消息则包括由测试主控端向测试服务节点发送的测试服务命令数据包与测试执行控制命令，以及测试服务节点向测试主控端反馈的原始试验数据与测试执行状态信息。测试服务命令数据包是测试主控节点逻辑控制层从测试用例功能模块或静态激励功能模块解析出的测试服务命令与数据消息，经通信服务将其传输至测试服务节点，进而实现测试服务的具体执行。测试执行控制命令则为测试主控端对测试执行状态如开始、暂停与终止等的控制，通过通信网络实现在测试主控端对测试服务节点测试执行的远程控制。测试服务节点实时将测试过程中通过测试资源从测试对象接收的数据发送至测试主控端，测试主控端对原始报文进行筛选解析，并根据用户定义存储至数据库中。测试服务节点在测试过程中反馈测试执行状态如通过、失败等给测试主控端，测试主控端在用户界面实时显示，为用户提供测试执行状态提示信息。



图18 基于DDS的通信中间件实现交互网络数据共享

# 4 方案创新点

（1）负载均衡

测试平台采用测试主控端与测试服务节点两级，分布式平台架构。主控计算机不必承担系统所有计算任务，测试平台整体负载均衡，可添加较多的子节点而不显著增加计算负载。

（2）可拓展性

测试平台的子节点的基本硬件结构与接口通用，可在主控端进行子节点增删管理，并设置子节点通信对象与方式。

（3）实时性

系统采用基于订阅发布的通信协议进行通信，可一对多进行通信，提高系统实时性，进而可实现全系统级实时测试。

# 5 方案成果与进度安排

2~4个月

1. **设备交付资料清单**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 资料名称 | 提供要求 | 数量 | 备注 |
|  | **纸质版资料**（中文版或英文版） |  |  |  |
|  | 装箱单 | ■是/□否 | 1套 |  |
|  | 合格证（或质保书、质量证书、检测报告、履历本） | ■是/□否 | 1套 |  |
|  | 设计方案及方案评审纪要 | ■是/□否 | 2套 |  |
|  | 设备验收大纲 | ■是/□否 | 1套 |  |
|  | 出厂检测报告或出厂检测记录或出厂验收试验报告（含设备联试内容） | ■是/□否 | 1套 |  |
|  | 研制技术总结 | ■是/□否 | 1套 |  |
|  | 软件使用说明书（含软件安装环境、安装顺序、安装路径及使用说明等）（中文版） | ■是/□否 | 3套 |  |
|  | 入厂验收测试方法 | ■是/□否 | 2套 |  |
|  | 校准证书或测试报告 | ■是/□否 | 2套 |  |
|  | 软件测试报告 | ■是/□否 | 2套 |  |
|  | 软件测试大纲 | ■是/□否 | 2套 |  |
|  | 设备使用、操作、维护说明书（中文版） | ■是/□否 | 3套 |  |
|  | 培训材料 | ■是/□否 | 1套 |  |
|  | 设备验收纪要 | ■是/□否 | 1套 |  |
| 2 | **电子资料**（中文或英文版） | ■是/□否 | 1套 |  |
| 2.1 | 设备实物外形图片 | ■是/□否 | 1套 |  |
| 2.2 | 设备三维数模 | ■是/□否 | 1套 |  |
| 2.3 | 序号1中所有电子文档 | ■是/□否 | 1套 |  |
| 2.4 | 项目开发过程中形成的程序及软件源代码 | ■是/□否 | 1套 |  |

进度安排：

1. **进度计划表**

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 时间 | 阶段安排 | 研究内容 |
|  | 合同签订1月内 | 评审 | 设计方案评审，提供设计方案报告 |
|  | 合同签订2月内 | 搭建自动化测试设备平台 | 项目开发过程中形成的程序及软件源代码，硬件系统平台 |
|  | 合同签订3月内 | 安装调试自动化测试设备平台 | 软件使用说明书（含软件安装环境、安装顺序、安装路径及使用说明等）（中文版）  入厂验收测试方法  校准证书或测试报告  软件测试报告  软件测试大纲 |
|  | 合同签订4月内 | 自动化测试平台交付验收 | 设备验收大纲  出厂检测报告或出厂检测记录或出厂验收试验报告（含设备联试内容）  研制技术总结  设备使用、操作、维护说明书（中文版）  培训材料 |