

Study on Engineering Flight Simulator Portable Tester

Cheng Luo, Weiqun Shen

School of Automation Science and Electrical Engineering, Beihang University, Beijing
Email: luochg@qq.com

Received: Jun. 20th, 2014; revised: Jul. 19th, 2014; accepted: Jul. 30th, 2014

Copyright © 2014 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Engineering flight simulator is a complicated semi-physical simulation system, which is the indispensable equipment for developing new aircraft. It has high cost of development, space site, and does not facilitate offsite migration. Especially, when failure occurs at the plane test site, engineering flight simulator cannot play the role well in the complex environment of the scene. Engineering Flight Simulator portable digital test instrument of this research is a sub system of engineering flight simulator. It has the support of flight test on site; general; easy to carry. This paper mainly discussed the self-design of test interface of a portable test instrument, problems of integrated test control design, a variety of data bus interface transitions, etc.

Keywords

Engineering Flight Simulator, Tester, General, Portable

某工程飞行模拟器中便携式数字测试仪的研究

罗 成, 沈为群

北京航空航天大学自动化科学与电气工程学院, 北京
Email: luochg@qq.com

收稿日期: 2014年6月20日; 修回日期: 2014年7月19日; 录用日期: 2014年7月30日

摘要

工程飞行模拟器是一个复杂的人机半实物仿真系统，是研制新型飞机不可缺少的设备。开发成本高；占用空间场地大；不便于异地迁移。尤其当试飞现场飞机发生故障时，无法在复杂的现场环境发挥工程飞行模拟器的作用。本文研究的工程飞行模拟器便携式数字测试仪，是工程飞行模拟器的一个子系统。它具有支持试飞现场测试；通用；携带方便等特点。文中重点讨论了便携式测试仪的自定义测试界面设计；集成式测试控件设计及多种数据总线接口转换等问题。

关键词

工程飞行模拟器，测试仪，通用，便携

1. 引言

近几年，国家大力支持高端先进飞行器的设计与研制，开发研制了多种工程飞行模拟器。设计人员在使用中不仅需要安装在固定厂房中的工程飞行模拟器，而且需要该工程飞行模拟器能够具有支持试飞现场对新机的测试功能。如在试飞现场对机上部件进行测试，及时发现设计中的瑕疵。但现在的工程飞行模拟器，不便于异地迁移，无法在复杂的现场环境发挥其作用。因此，迫切需要进一步完善工程飞行模拟器的功能。

本文所述，提出工程飞行模拟器中的便携式测试仪，正是旨在解决这一问题。通过自定义测试界面的设计，界面及其元素进行描述，界面控件进行分类与设计，数据的交换及对接口系统中总线格式的数据转换进行了设计，开发研制一种支持用户动态需求、便于携带的工程飞行模拟器测试仪。

2. 工程飞行模拟器与便携式测试仪

工程飞行模拟器，为飞机设计人员提供了一个研制新飞机的良好环境。具有试验任务管理控制；实时监控测试数据；测试数据分析、测试信息管理等功能。从而为飞机系统的研制提供充分的技术储备。使飞机研制工作在确保高逼真度，优化设计、系统安全和低能高效的运行环境中工作。

某工程飞行模拟器系统结构图如图 1 所示[1]。

该工程飞行模拟器由控制台；飞行仿真系统；机上试验部件组成。控制台主要由总控计算机，数据/参数显示计算机，曲线显示计算机，数据记录计算机组成。其中总控计算机是整个飞行仿真平台的大脑，负责控制仿真过程、设置初始参数、监控其他各模块运行情况；数据/参数显示计算机与曲线显示计算机共同组成飞行仿真平台的监控系统，负责整个仿真过程的监控，具有实时、通用、图文并茂、扩展容易等特点；数据记录计算机负责将仿真数据保存为文件或数据库表格，并在仿真后可以随时从文件或数据库调用实验记录数据进行曲线绘制等，方便研究人员分析仿真数据。

由于这些设备均建在固定的厂房，当在试飞现场用于机上实物部件测试时，便需要异地迁移。现在在现有的这套系统上加入便携式数字测试仪，便携式数字测试仪可随身携带，用于在试飞现场直接进行机上实物部件的测试。图 2 描述了便携式数字测试仪的工作环境及其在整个系统结构中的位置。它由便携式测试台与通用接口转换器组成。便携式测试仪由于它便携方便的特性，可直接用于新机部件外场的测试工作。同时它也可与该工程飞行模拟器的主控台通过无线系统进行通讯。实时将现场测试结果传送到总控台，以便分析处理[2]。由于便携式测试仪面向的是机上所有部件。因此测试仪中自定义测试界面；集成式测试控件及多种数据总线接口转换成为我们急待解决的关键问题。

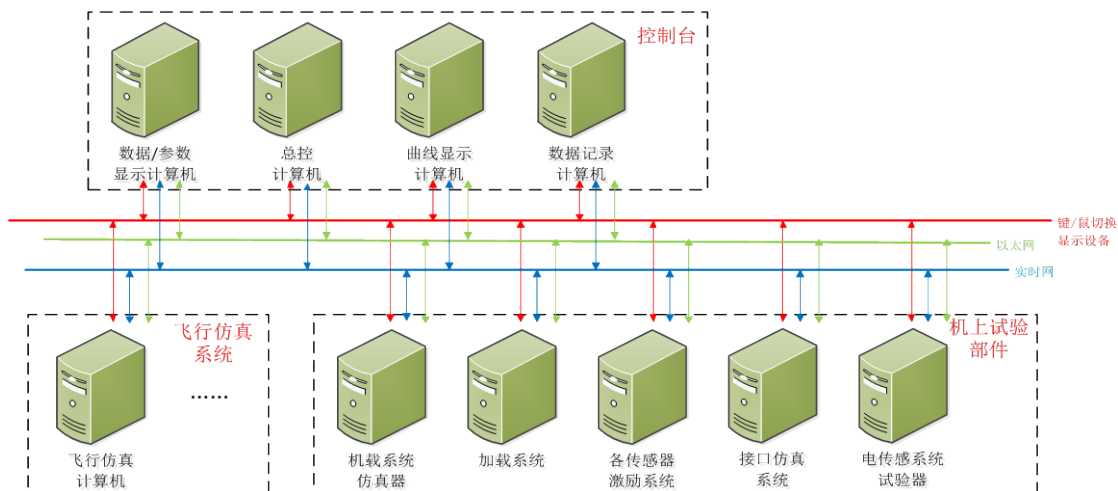


Figure 1. An engineering flight simulator system structure diagram
图 1. 某工程飞行模拟器系统结构图

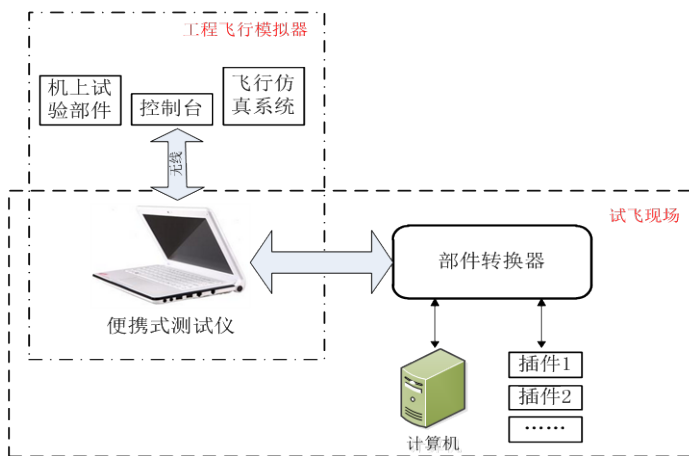


Figure 2. Working environment of the portable tester
图 2. 便携式测试仪的工作环境

3. 自定式测试界面设计

在面向对象的设计中，类代表了某一批对象的共性和特征，类是对象的抽象，用来定义对象的一种抽象数据类型，它是对象的一个模板。便携式测试仪界面元素分为四类：树形测试目录类、输出类、输入类及编辑订制类。每个类具有不同的对象名，相同的操作集合；相同的属性集合。类描述详见表 1 所示。

3.1. 自定式测试界面描述

一直以来针对每一个工程飞行模拟器型号，都需要重新进行大量的测试页面设计。这需要大量的时间进行开发设计，而没有一个通用的模板可以借鉴，使得开发各类功能时，开发周期过长，并且各个软件的通用性不强[3]。

使用基于 XML 的通用软件优势在于，不需要在有编译环境的计算机上就可以对软件进行编辑、修改及调试，这样编程人员就可以在任意一台有记事本功能的计算机上修改及调试软件项目，并且高度集成了定制软件所需要的主要功能，不需要对其核心部分进行修改[4] [5]。

3.1.1. XML 对树形测试目录的描述

一个完整的测试系统包含树形测试目录，测试页面和控件。一个测试目录对应着若干测试页面，每个测试页面内又包含若干个测试控件，从而形成一棵树状结构。

测试目录包含两层结构，一是测试项目的名称，二是子系统所包含的页面的名称，它们是嵌套关系，使用 XML 描述如表 2 所示。

3.1.2. XML 对页面的描述

每个测试页面内均可包含若干个控件类，每个类的对象的个数原则上均不受限制。一个页面的描述，等于是若干控件描述的集合，表 3 给出了测试页面的 XML 描述，至于控件内部的描述，将在后一小节进行介绍。

3.1.3. XML 对页面的描述

一个完整的控件，包括其参数信息、位置信息和状态信息。参数信息包括反射内存中变量的信息，如地址、名称、类型、上下限等；位置信息包括控件本身处于页面中确切位置的信息及控件内部排版相关的信息，如行间距、列间距等；状态信息保存控件的设置状态，如信号发生器中设置的曲线及曲线的参数，使得每次打开文件可直接操作而毋须进行重复的设置。将组成一个控件的所有信息描述为 XML，表 4 所示。

3.2. 界面控件的分类与设计

前面，提到了对测试元件的分类。通过对已有测试台内所用到的功能进行研究与分析，将其分为两

Table 1. System resulting data of standard experiment

表 1. 便携式测试仪界面元素分类及描述

类名	类功能	类内部表示	主要操作实现
树形测试目录类	显示测试目录及切换测试页	目录项，目录项状态	切换显示当前测试页
输出类	显示反射内存中的信息	反射内存地址、名称、数据类型、数据范围等	数据转换；数据处理；显示数据
输入类	将数据输入到反射内存中	反射内存地址、名称、数据类型、数据范围等	输入数据；数据处理；数据转换
编辑订制类	编辑界面，添加、删除、编辑控件	位置坐标、布局信息	添加控件；删除控件；移动控件

Table 2. XML description and meaning of test catalog

表 2. 测试目录的 XML 描述及含意

XML 中文件标识	标识的含意
Level 1	对子系统的描述，当子系统中只含有一个页面时，也可以是对页面的描述。
Level 2	对子系统内页面的描述，一个子系统可以含有若干个页面。

Table 3. XML description and meaning of test pages

表 3. 测试页面的 XML 描述标识及含意

XML 中文件标识	标识的含意
root	根节点，代表整个页面
Group	组，内可存放若干个 GroupItem
GroupItem, BitSet, SelectBits, Signal Generator	控件
.....	其它控件(可增加新的控件类型)

Table 4. XML description and meaning of one control

表 4. 某控件的 XML 描述及含意

XML 中文件标识	标识的含意
addr	反射内存地址
addrName	反射内存地址中数据的名称
dataType	反射内存地址中数据的类型
itemType	控件的类型
dataUpLimit	反射内存地址中数据的上限
dataDownLimit	反射内存地址中数据的下限
dataUnit	反射内存地址中数据的单位
x, y, numInRow, gap, distanceBetweenRow	某控件的属性, 不同控件类型可以有不同的属性
.....	(针对不同的控件类型所特有的项目, 包括参数信息及状态信息)

类: 输出型(监控型)元件和输入型控件。具体地, 输出型包括参数显示, 曲线显示, 状态显示等; 输出型包括写参数, 信号发生器, 状态设置等。

3.2.1. 界面元件控件和规范化

在对控件进行分类后, 各控件已经能够完成相互之间比较独立的功能。但是, 当控件的上层对控件进行操作时, 很大情况下是不区别控件是何种类型的。比如说拖拽以调整控件的位置, 程序是不会也没必要区分操作人员拖拽的是何种控件的。并且, 规范化的操作对后续控件的扩展也会大有好处, 使得程序员不用太多地去关心除了控件本身之外的其它代码。因此, 在对控件进行 XML 描述之后, 我们仍然需要对它建立一套机制, 以对其操作进行规范化。

3.2.2. 所有控件类均有的操作

- 保存
将本控件的数据信息及显示信息以 XML 的形式保存。
- 加载
从 XML 中提取控件的相关信息, 包括数据信息及显示信息。
- 显示
将控件类的对象以一定的形式显示在页面上, 主要供新建控件及加载控件后调用。
- 删除
删除本控件类的对象, 清除所有相关数据, 由上层结构的对象进行调用。
- 拖拽
改变控件的位置, 并更新其位置信息。页面内控件布局的主要调用函数。
- 属性设置
进行控件的相关属性设置。因为不同的控件具有不同的属性, 这种“不同”不但体现在不同的控件类之间, 也体现在相同控件类的不同数据之间。所以, 当产生出一个控件类之后, 根据需要, 进行本控件对象的属性相关设置。

3.2.3. 系统的保存与加载

得益于良好的树形结构的设计, 系统的保存与加载均可通过本结构的保存与加载函数及本结构所包含的下一结构的保存与加载函数来实现。这同时也说明, 当新增一个新的控件类时, 只需按照标准定义

新控件类的相关操作即可，而不用去关心其它类的具体实现。

- 系统的保存

从系统的保存函数开始，依次调用子系统、页面、控件(控件也可能调用其它控件的保存函数)的保存函数。

- 系统的删除

与系统的保存调用顺序相反，调用系统的删除函数时，先调用子系统的删除函数，而其又先调用页面的删除函数，其又先调用各控件的删除函数。

- 系统的加载

从系统的加载函数开始，依次调用子系统、页面、控件(控件也可能调用其它控件的加载函数)的加载函数。与通常所见到的加载不同的是，这里使用了一个状态变量，用来标识是否删除已有的系统。选择“是”，则是直观意义上的加载另外一个系统；选择“否”，则相当于将已有系统和被加载系统合并为一个新的系统。这得益于 XML 对整个系统结构完整的良好描述。

3.2.4. 基于 XML 的连接扩展机制

基于 XML 的扩展控件是遵循一定规范所编写的控件类。当现有的控件类不能满足需求时，就需要添加新的控件类，只需要给出 2.2.2 中所提到的所有控件类所均有的操作即可。程序中定义有一个类，类中包含有这些操作的虚函数。所以可以很方便地通过继承这个类和覆写虚函数即可实现。新控件类的对象则是通过抽象工厂产生[6]。

4. 集成式测试控件设计

4.1. 通用监控控件的设计

通用监控控件的主要功能有参数显示、曲线显示、状态显示，在工程飞行模拟器中这三个功能分别由三台计算机完成。在便携式测试仪中把这三个功能集中在一台计算机之中。对应的将三种功能设计成三种控件：参数显示控件，曲线显示控件，状态显示控件。参数显示的主要功能是从设备中(本测试仪是从反射内存卡中，下同)读取参数的值，并给以显示。曲线显示用于显示某个或某些参数在一定时间段内的值的变化，并绘制成曲线。状态显示则是从设备中读取某状态位的值(0 或 1)，并将其译为适合操作人员阅读的信息。

在集成的过程中，要考虑数据流的问题。为了让这些监控控件有条不紊地读取数据，通过对整个过程的分析，将其分为四层：数据读取层，数据转化层，数据处理层，数据显示层。如图 3 所示。第 1 层从设备上读取原始的数据；第 2 层将第 1 层所读取的数据进行格式转化，这并不是必须的；第 3 层将第 2 层转化后的数据根据要求进行必要的处理；第 4 层将处理后的数据以直观的人性化的形式展示给监控人员：参数显示则直接显示参数的值，曲线显示将时间段内的值绘制成曲线后给以显示。

4.2. 通用输入控件的设计

输入控件的作用是对设备进行写操作，主要分为写参数，信号发生，状态设置。写参数能够将参数的值写入设备中。信号发生能够将某一信号类型的值连续写入设备中。状态设置则是设置状态位。

类似监控控件的分层，也将输出控件的整个操作分为四层：数据获取层，数据处理层，数据转化层，数据写入层。如图 4 所示。第 1 层从界面获取初始数据；第 2 层将第 1 层获取的数据进行必要的处理，例如信号发生器控件从数据获取层获取了输出设备的地址以及输出曲线的类型及相关参数，而数据处理层则产生符合曲线类型要求及参数要求的数据；第 3 层将第 2 层处理后的数据进行转化，这并不是必须的，有些情况下可能并不需要转化；第 4 层将最终的数据写入设备。

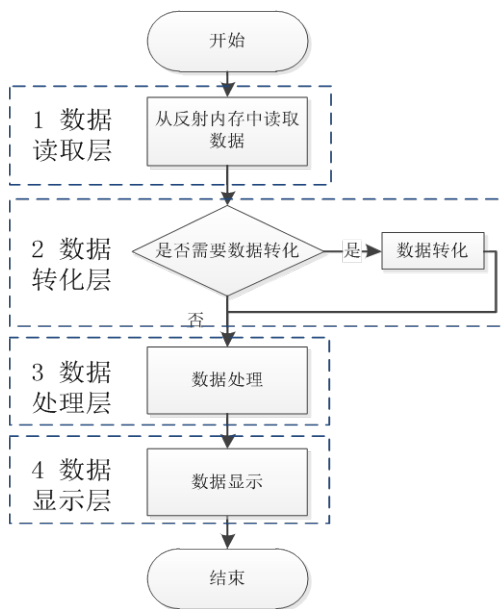


Figure 3. Monitoring controls' four layer structure and flow chart

图 3. 监控控件的四层分层结构及流程图

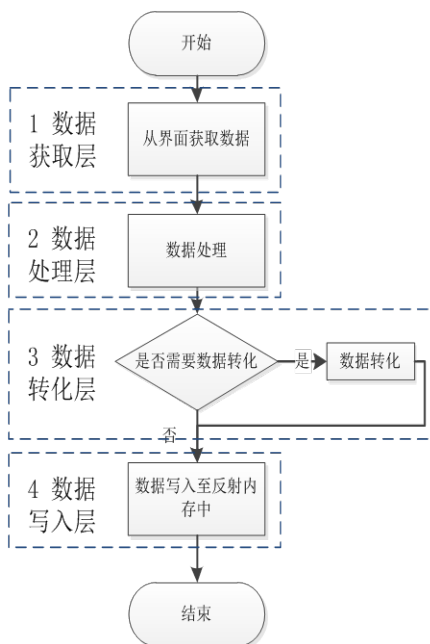


Figure 4. Inputting controls' four layer structure and flow chart

图 4. 输入控件的四层分层结构及流程图

5. 接口系统中总线格式的数据转换

由于便携式测试仪面向机上的多个部件，主要面向的总线数据格式种类主要有 429、422、1553。在把数据从总线格式转换为真实数据格式的时候，我们需要考虑以下几个问题[7] [8]:

- 数据大小端
- 数据范围

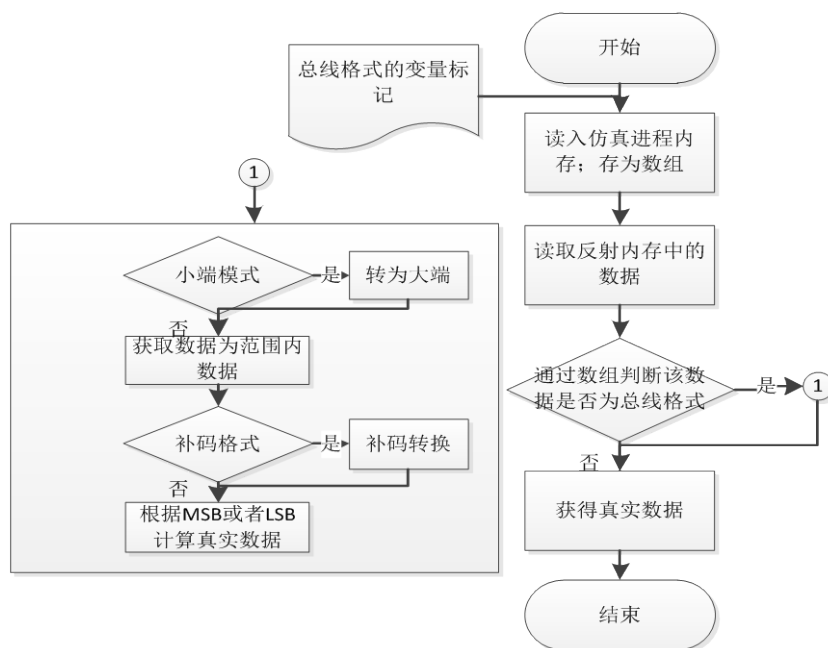


Figure 5. Bus format analysis flow chart

图 5. 总线格式解析流程图

- 是否补码格式
- 数据 MSB 或者 LSB 对应数值

经分析，总线格式的数据一般有 32 位，可以标记为 D0~D31，其中只有一定范围内的数据位表示真实数据值。例如，D0~D14 表示数据位。

数据位进行二值编码，其中除最低位为 1 以外其它位全为 0 表示最小数据，即 LSB；所有数据位全为 1 则表示最大数据，即 MSB。因此，我们需要确定最小数据或者最大数据对应的真实数值。

多字节数据在机器内存中的表示分为两种方式：大端模式和小端模式。多字节数据中高字节在高地址位，低字节在低地址位称为小端模式。反之，高字节在低地址位，低字节在高地址位则称为大端模式。由于总线格式的数据是 32 位，即 4 字节的，属于多字节数据，因此需要考虑数据的大小端问题。

机器中的数据通常是以补码的形式存储的，然而总线格式的数据有些是按照补码格式，有些不是按照补码格式，因而在转换为真实数据的时候需要加以区分。

整个解析流程图如图 5。

6. 结论

本文介绍了某工程飞行模拟器中便携式测试仪的实现方案，包括方案的提出及具体的设计。首先比较了工程飞行模拟器与便携式测试仪，介绍了工程飞行模拟器在试飞现场的应用不足之处，借此提出了便携式测试仪。随后对自定式测试界面进行设计，将便携式测试仪界面内的元素根据其功能分为四类，然后使用 XML 对界面及其元素进行描述并对界面控件进行分类与设计。再后对通用监控控件及通用输入控件的数据交换进行了分层，使数据能够有条不紊地交换。最后对接口系统中总线格式的数据转换进行了设计。

通过使用此测试台，可方便快速的构造出一个对应于具体环境及条件的测试方案。极大的提高了软件开发人员的效率，缩短了开发周期，节省了开发成本。本人已初步完成了一个这样的测试台，其界面

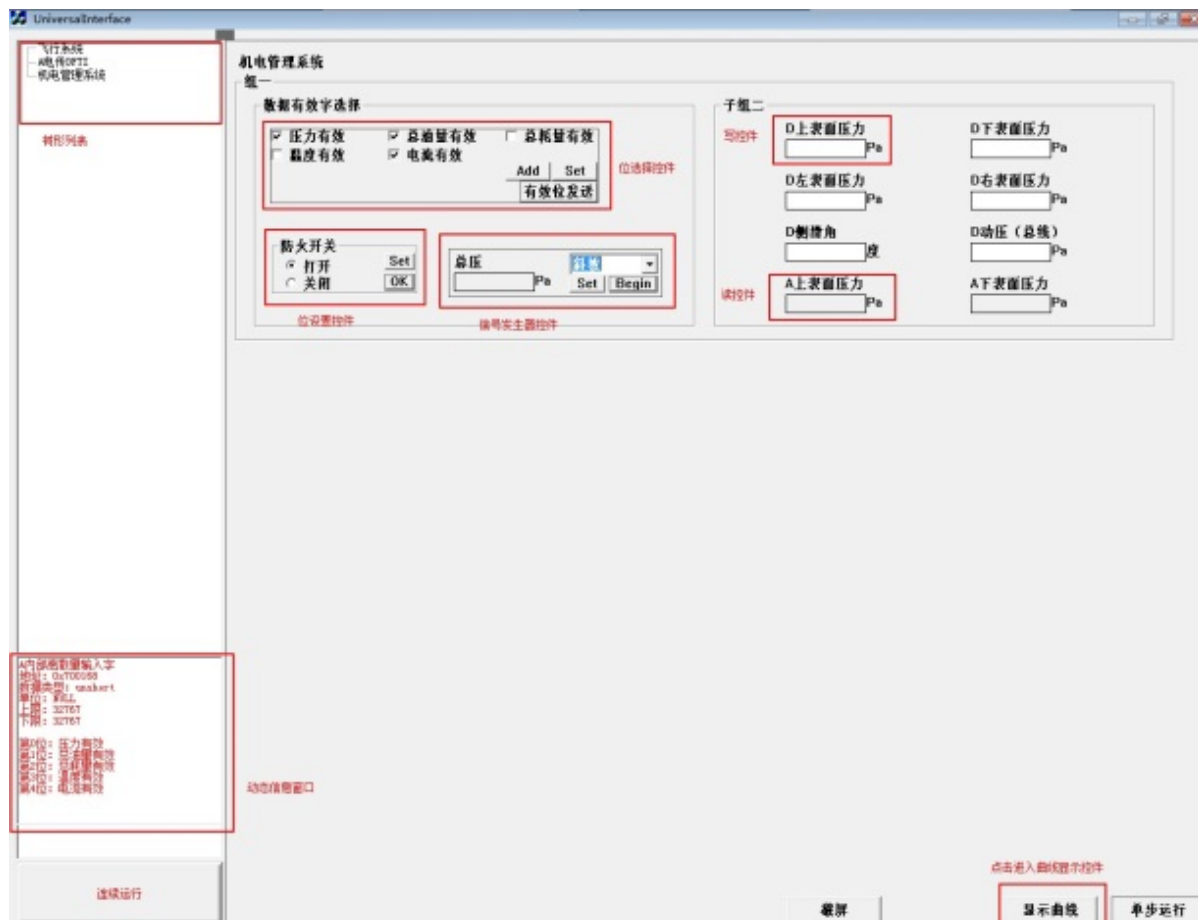


Figure 6. Tester interface example
图 6. 测试仪界面样例

如图 6。当然，还有很多值得改进的地方，后续会进行不断的完善。

参考文献 (References)

- [1] 王行仁 (1998) 飞行实时仿真系统及技术. 北京航空航天大学出版社, 北京, 1-5.
- [2] 徐享忠, 郭齐胜, 张伟等 (2013) 先进分布仿真. 国防工业出版社, 北京, 13-14.
- [3] 李延春, 晏敏 (2003) 软件构件技术的现状与未来. *计算机工程与应用*, **31**, 86-94.
- [4] 张晋, 陈立潮 (2001) 未来基于 Web 的通用数据格式-XML. *华北工学院学报*, **2**, 121-125.
- [5] 周传生 (2006) 基于 XML 的软件总线设计的研究与实现. *计算机工程与应用*, **20**, 3808-3810.
- [6] Gamma, E., Helm, R., Johnson, R. and Vlissides, J. (2013) 李英军, 马晓星等, 译. *Design Patterns*. 机械工业出版社, 北京, 57-62.
- [7] 冯收, 凤雷, 乔家庆, 赵加凤 (2012) 1553B 总线 RT 有效性测试方法研究. *计算机测量与控制*, **10**, 2614-2616.
- [8] 闫晓, 杨福彪, 朱勇 (2011) 智能 1553B 接口软核实现技术. *指挥控制与仿真*, **6**, 112-115.