

一种实时系统中实现可视化仿真模拟测试的方法

周国祥, 石 雷, 韩江洪

(合肥工业大学计算机与信息学院, 安徽 合肥 230009)



摘 要: 探讨了一种为复杂的嵌入式系统搭建软件仿真平台的方法, 给出了详细的系统架构和前后台的设计。该方法通过网络协议将界面开发工具和程序开发工具联系起来, 从而实现了仿真平台的界面直观、操作方便。这种方法的特点是实现简单, 具有通用性, 可应用于一般生产监控实时系统中。

关键词: 仿真; 可视化; 客户端/服务器架构; XML 套接字

中图分类号: TP391.9 **文献标识码:** A **文章编号:** 1004-731X (2006) 12-3561-03

A Test Method for the Real-Time Embedded System with the Visual Simulate

ZHOU Guo-xiang, SHI Lei, HAN Jiang-hong

(School of computer Science and Information Engineering, Hefei University of Technology, Hefei 230009, China)

Abstract: A kind of method was discussed to put up the software simulating platform for complicated embedded systems, and the systematic framework and detailed design of platform were provided. The method links up the developing instrument of interface and developing instrument of procedure ingeniously through the internet protocol, and it has beautiful interface, easy operating and easy realizing. The method is common in use, and it can apply to various kinds of real-time embedded systems.

Key words: Simulation; Visualization; C/S(Client/Server); XMLSocket

引 言

在实时嵌入式系统的设计与测试中, 常常会涉及到多方面软硬件的协同与联调。通常先对各个子模块分别进行测试, 子模块测试完成后再实现各子模块的协同联调。这样对于每个子模块就要分别在 PC 机上设计仿真软件模拟实际的环境协助测试。通常对实时嵌入式系统的测试方法是采用专门仿真测试软件或搭建仿真硬件电路^[1,2]。但在这样的针对子模块的小型测试环境中, 用专门的或者大型的仿真测试软件显然是不现实的, 而是应该根据各子模块要求开发简单易用的仿真测试系统。而如果开发专门的硬件仿真电路, 如仿真调试器, 虽然能起到很好的效果, 一劳永逸, 但考虑到开发成本和开发周期, 也不是最好的解决办法。

因此可以想到的快速有效的解决办法是采用计算机编写仿真测试系统实现计算机模拟。对于情况简单的子模块, 开发一个文字界面的仿真测试系统是容易的。但对于一些情况复杂的子模块, 如果单纯用文字界面的测试系统, 往往很难切合实际情况。此时仿真软件的界面一般需采用图形可视化的方式表达, 显得很复杂, 界面设计的要求也很高。例如在汽车电子中, 要设计一套汽车仪表仿真软件来测试汽车仪表盘车身控制系统的正确性, 这就要求仿真软件的界面与实际的汽车仪表尽量一致。

此外, 需要测试的系统往往不是直接运行于 PC 机上, 而是各种嵌入式系统中, 这就需要通过各种底层接口与 PC 机进行通信。例如汽车电子仪表盘车身控制系统就是经 USBCAN 通过 USB 接口与 PC 机通信的。使用 VC++, Delphi 等这样的开发工具, 虽然其功能强大, 可以实现对 PC 机各种底层接口的编程, 但做界面的能力都不强, 要想做出如汽车仪表这样的复杂软件界面很困难。而使用图形界面强大的工具如 flash 等可开发出复杂的图形界面, 但编程能力弱, 更无法进行底层接口的编程。如何采用开发工具既能设计出模拟实际环境的真实、美观的图形界面, 又能对 PC 机各种底层接口进行有效的编程, 这是开发嵌入式应用系统可视化仿真测试软件有待解决的问题。

本文探讨了一种方法, 利用这种方法做出的可视化仿真测试软件, 既可以做到界面美观, 操作方便, 又很好的解决了底层接口编程问题, 而且实现简单, 对一般的生产监控系统具有通用性。

1 系统总体设计

1.1 系统架构

考虑到软件开发工具和图形开发工具各自的优缺点, 如果能将二者有机结合起来, 则可以取长补短。因此, 本系统的基本设计思路是利用图形开发工具如 flash 等开发前台界面, 用软件开发工具如 VC 等开发后台接口, 二者通过网络协议交流信息。整体架构如图 1 所示。

整个系统采用 C/S (Client/Server) 架构。即处理底层硬件接口的程序为服务器端程序, 处理前台界面的程序为客

收稿日期: 2006-02-15 修回日期: 2006-08-29

基金项目: 安徽省自然科学基金项目 (050420202)

作者简介: 周国祥(1956-), 男, 安徽合肥人, 教授, 硕导, 研究方向有科学计算与可视化技术、网络管理信息系统、智能建筑; 石雷(1980-), 男, 安徽合肥人, 硕士, 助教, 研究方向有嵌入式系统。

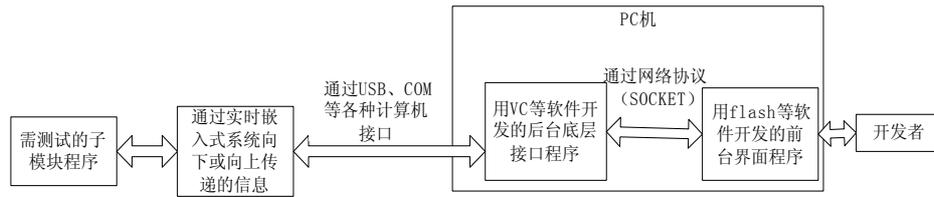


图 1 系统架构

户端程序。服务器端有一线程监听接口，当接口有数据到来时，就会将数据打包发往客户端，随后在前台界面上做出相应的显示。客户端也可通过服务器端向需测试的子模块发送数据。

下面结合汽车电子车身仪表仿真软件来对各部分作具体探讨。

1.2 网络连接模型

该方法的前台界面程序和后台接口程序是通过网络协议进行连接的。由于几乎所有的开发工具都支持网络协议，因此选择网络协议实现二者通信是最直接最有效的方法。

而实现网络编程最直接的方法是采用 Socket 套接字。Socket 是一种网络编程接口，可以适用于不同的网络协议，如 TCP/IP 协议、APPLE TALK 协议等。Socket 最早诞生于 Unix，后来微软、SUN 联合其他计算机软硬件厂商开发了 Windows Socket，使程序员可以在 Windows 下方便地编写基于图形界面的网络程序。

Socket 的实现机制是一端启动一个服务器 Socket，一端启动一个客户端 Socket，连接以后传输数据可通过 Socket 的输入输出流来实现。总的说来，其接口模型采用的是“打

开—读—写—关闭”方式。常用的 Socket 有两种：一种是字符流 Socket，建立在面向连接的协议上(TCP)，是一种可靠的传输方式。另一种是数据包 Socket，建立在非面向连接协议上(UDP)，其传输速率比前者高，但是一种不可靠的传输方式。另外还有一种直接面向 IP 协议的 Socket，一般很少采用[3]。前两种传输方式的模型分别如图 2 和图 3 所示。

1.3 WinSock 编程模型

具体说来，程序中使用 Socket 包括以下几个步骤^[5,6]：

建立 Socket：使用 Socket 进行网络通信时，必须先建立 Socket。

配置 Socket：包括本地和远地机的 IP 地址、本地和远地进程的协议端口、连接上使用的协议。

连接 Socket：首先确定采用的是面向连接的协议还是无连接的协议，采用面向连接的协议时这一步必不可少。该协议将在连接端点之间建立一个虚电路，客户端程序只需提供原地主机的 IP 地址和协议端口。

使用 Socket：程序能够使用 Winsock API 在一个配置好的 Socket 上传送和接收数据。Winsock 有四个函数用于数据的传送和接收，分别为 Send、Sendto、Recv 和 Recvfrom。

2 系统的前后台开发

系统前台采用 Flash 开发。Flash 是制作 Web 内容的专业标准创作工具。无论是创建动画徽标、Web 站点导航控件、长篇动画，还是 Web 应用程序，Flash 都是设计者们的首选工具。图 4 是用 Flash 制作的汽车电子车身仪表仿真软件界面。由图可见，用 Flash 制作的界面美观，一目了然，而且实现简单。

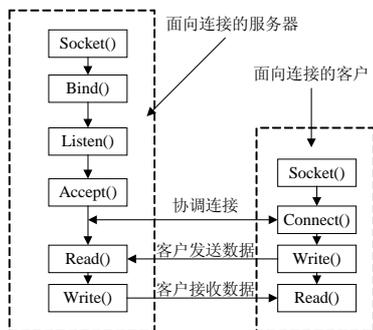


图 2 面向连接 Socket 客户机/服务器模型

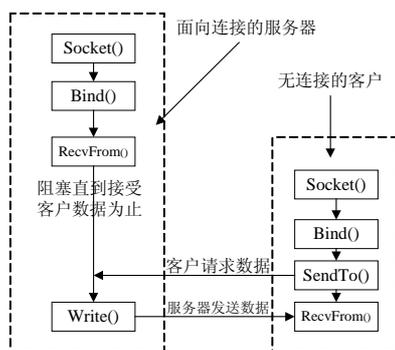


图 3 无连接 Socket 客户机/服务器模型

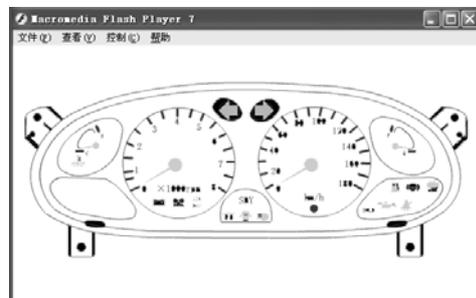


图 4 用 Flash 制作的汽车电子车身仪表仿真软件界面

Flash 从 5.0 版本开始提供了网络应用功能，允许开发人员在以 Flash 为基础的网站应用程序中使用 XML 的结构化语言，以便达到数据安全传输与交互的目的。XML 是一种扩展性标志语言，使用它就可以以容易而一致的方式格式

格式化和传送数据。Flash5.0 以及后续版本提供了一个很好的对象 XMLSocket, 通过该对象可以方便的进行网络连接与交互。该对象还很好的解决了网络体系结构中常见的被动性和滞后性的限制。即在常见的 C/S 架构中, 客户端每隔几秒钟就要主动向服务器端发送请求, 然后获取服务器端返回的信息刷新界面。使用 XMLSocket 后客户端可随时接收服务器端发来的信息, 从而更好的满足了实时系统的要求。

Flash 进行网络编程很简单, 首先创建一个 XMLSocket 连接, 通过 onConnect 即可实现。连接建立成功后, 通过 onData 函数处理从服务器到来的消息, 通过 send 函数向服务器发送消息。

系统后台采用 VC 开发。图 5 是采用 VC 开发的汽车电子车身仪表仿真软件后台控制界面。

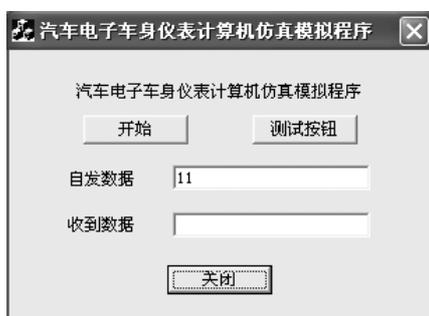


图 5 用 VC 制作的汽车电子车身仪表仿真软件后台控制界面

Microsoft Visual C++ 的 MFC 类库提供了两个 Socket 类: CAsyncSocket 类和 CSocket 类。CAsyncSocket 类封装了 Windows Sockets API, CSocket 类是从 CAsyncSocket 类派生的高级抽象, 支持同步操作。

CAsyncSocket 类在很低级别上封装了 Windows Sockets API, 该类使我们使用面向对象的方式来进行 Sockets 编程, 而且可以非常方便的调用其它 MFC 对象。不过这个类要求编程者对 Sockets 编程有较深入的了解, 可能需要编程者编写底层函数来进行通信操作。而 CSocket 类提供了一个高级的 Socket 支持, 它运用了 MFC 的序列化类来提供和传输 Socket 对象。既由 CArchive 管理了很多低层的功能, 使得程序员不需要考虑网络字节顺序和字符串的转换问题, 所以编程的难度得到了降低。但是由于使用 CSocket 和 CSocketFile 类运用了 MFC 的序列化类进行数据传输, 所以 Client 和 Server 双方程序都必须用 MFC 来编程, 或者另一方的程序能够识别 MFC 序列化协议^[6]。由于客户端不能识别 MFC 序列化协议, 所以服务器端需采用 CAsyncSocket 类编写。CAsyncSocket 类中定义的函数都是非阻塞的, 函

数调用后就可以返回, 操作完成后无论成功与否都将有一个相应的网络消息发出。MFC 把这些事件处理函数定义为虚函数, 所以需要继承 CAsyncSocket 类, 重载这些网络事件函数。

CAsyncSocket 类中对网络事件的响应一共定义了六个消息^[6], 分别为 FD_CONNECT、FD_ACCEPT、FD_WRITE、FD_READ、FD_OOB 和 FD_CLOSE。这些消息在建立 Socket 连接时, 请求到来时, 读写时, 收到外来数据时以及关闭 Socket 时会依次激发。每个消息对应一个虚函数, 分别为 OnConnect()、OnAccept()、OnSend()、OnReceive()、OnOutOfBandData() 和 OnClose()。用户只需重载这些函数即可完成相应功能。

3 结论

本方法是一种通用的方法, 其前台界面程序可以采用各种适宜的图形功能强大的工具开发, 只要该工具支持网络协议; 其后台接口程序可以用 VC、Delphi 等多种软件开发。由于其前台界面程序的开发可选用专业的图形开发工具, 使得其界面美观, 操作方便, 便于测试人员使用; 由于其后台接口程序的开发可选用专业的软件开发工具, 使得其后台接口适应面广, 可方便地实现与各种实时系统、嵌入式系统的连接; 由于其前后台的连接选用网络协议, 使得前后台的实时连接通信成为可能。该方法避免了使用专用的大型仿真测试软件, 避免了开发专用的硬件仿真测试器, 开发周期短, 成本低, 并可应用于各种生产监控实时系统的仿真测试中, 无论在界面上, 在操作上, 还是在实现上都具有不可比拟的优势。

参考文献:

- [1] 张翔, 钱立军, 等. 电动汽车仿真软件进展[J]. 系统仿真学报, 2004, 16(8): 1621-1623.
- [2] 陈定君, 郭晓东, 等. 嵌入式软件仿真开发系统的研究[J]. 电子学报, 2000, 3: 137-139.
- [3] 郭正义, 范瑜. 用 Flash 实现的 TCP/IP Socket 网络传输方案[J]. 现代电子技术, 2002, (10): 71-72.
- [4] Chen Xingshu, Zhao Zeliang, Shen Changxiang. Using embedding method to achieve transparent proxy[C]// TENCON '02. Proceedings. 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering, Volume 1, 2002: 228 - 231.
- [5] 刘铭, 施仁. 工业远程监控设计方案探讨[J]. 计算机应用与软件, 2004, (7): 54-56, 92.
- [6] 胡静, 古乐野. 使用 CAsyncSocket 类实现异步网络通信[J]. 计算机应用, 2002, 22(7): 108-109, 112.