

Abstract

Along with the rapid development of computer science, large-scale integrated circuit technique and communication technique, the region of instrument technique takes place enormously, and virtual measurement technique has developed in the region. The appearance of related software makes virtual instruments to replace traditional instruments gradually and to become a kind of main trend in instrumental region.

The technique of web breaks through restricts of the traditional communication method and makes people more convenient. By the modification on the software, some different virtual instruments can be developed, which increases the system flexibility greatly. With the gradual opening of electronics laboratory, traditional management mode has been incompetent with large numbers of data, such as user information, experiment data. When web technique expands to the region of instrument, information system and measurement system can be linked with no seam, so current electronic laboratory can be developed to be a kind of virtual web laboratory, which makes the limited resources can be shared furthest.

A series of virtual instrument has been designed successfully based on PCI-6024E DAQ board and LabVIEW 7 Express software. The main work of my project is to get some solution of experimental management system, based on virtual instruments with the combination of database technique and web technique. The system can manage user information and experimental data effectively, record the experimental process, and make teachers read the reports more convenient. The article gives the detailed introductions of the hardware system, software system and development environment of virtual instrument. With a series of virtual instruments based on LabVIEW 7 Express, the process of designing a virtual instrument has been introduced briefly. Two different scheme and their advantages or disadvantages have been proposed to realize experimental management system with a application server of Tomcat and a database of MySQL. Taking advantage of network technique and virtual instrument technique, network extension of virtual instruments can make more resources to be shared on the web. At last, author gives the detail solutions of virtual web laboratory and proposes some notice-needed questions.

第一章 虚拟仪器技术的现状及发展

1.1 虚拟仪器

1.1.1 虚拟仪器技术的由来

电子技术的飞速发展及其在各方面的广泛应用，对仪器的“智能”要求越来越高，仪器中微机的任务不断加重，仪器在很多方面逐渐向微计算机靠拢。此外，随着微计算机和智能仪器的普及，测试系统中包含的重复部件越来越多，而冗余的部件往往不能容错。因此，需要统筹地考虑仪器与计算机之间的系统结构。在这种背景下，1982 年出现了一种新型的、与 PC 机配合使用的模块式仪器，自动测试系统结构也从传统的机架层迭式结构发展成为模块式结构。与传统仪器不同的是，模块式仪器本身不带仪器面板，因此必须借助于 PC 机强大的图形环境和在线帮助功能，建立图形化的“虚拟的”仪器面板，完成对仪器的控制、数据分析与显示。这种与 PC 机结合构成的，包含实际仪器使用与操作信息软件的仪器，称为“虚拟仪器”。它是计算机技术介入仪器领域所形成的一种新型的、富有生命力的仪器种类。

1986 年，美国 NI 公司（National Instrument）提出了虚拟仪器的概念，提出了“软件即仪器”的口号，彻底打破了传统仪器只能由生产厂家定义，用户无法改变的局面，从而引起仪器和自动化工业的一场革命，代表着从传统硬件为主的测量系统到以软件为中心的测量系统的根本性转变。简单的说，一套虚拟仪器系统就是一台工业标准计算机或工作站，配上功能强大的应用软件、低成本的硬件（例如插入式板卡）及驱动软件，它们在一起共同完成传统仪器的功能。

从虚拟仪器概念提出至今，有关虚拟仪器技术的研究方兴未艾。研究人员在虚拟仪器硬件接口、虚拟仪器软件及其设计方法等方面做了许多有意义的研究工作，并已开发了许多实用的虚拟仪器系统，如卡式仪器、总线式仪器、计算机化仪器等，其共同点是大多强调其软件面板、虚拟界面、控制环境以及数学模型和软件方法。典型的虚拟仪器模式可以理解为，除了信号的输入和输出以外，仪器的其他操作，如测量、控制、变换、分析、显示等功能均由软件来实现，它们依据某种通用或专用总线标准或规约，或以某种接口形式，与计算机进行通信，由计算机统一进行调度和管理的一种数字化仪器。

最初 NI 公司提出的虚拟仪器概念实际上是一种编程思想，这种思想可简单地表述为：一个 VI 可以由前面板、数据流框图和图标连接端口组成，前面板相当于真实物理仪器的操作面板，而数据流框图就相当于仪器的电路结构。随着现代测试水平与仪器技术的发展，日

前虚拟仪器概念已经发展成为一种创新的仪器设计思想,成为设计复杂测试系统和测试仪器的主要方法和手段。先进的计算机总线技术有力地促进了虚拟仪器技术的发展,包括高速总线技术 VXI (VMEbus eXtensions for Instrumentations)、PXI (PCI eXtensions for Instrumentations)、USB (Universal Serial Bus)、IEEE 1394 (Firewire)、基于网络的远程测量技术、智能虚拟仪器驱动技术 IVI (Interchangeable Virtual Instruments) 等。采用这些新技术的测试仪器,预计其技术性能可比传统独立的台式仪器提高 10 倍以上,而且系统互换性和互操作性显著提高。未来的电子测试仪器除了拥有更强大和更完善的功能之外,体积将会更小,甚至可以移植到被测试的设备或电路中去,并且具有自我诊断、自我校准和自我感知的能力^[1]。

1986 年 10 月,美国 NI 公司推出了图形化虚拟仪器专用开发平台 LabVIEW,它采用独特的图形化编程方式,编程过程简单方便,是目前最受欢迎的虚拟仪器主流开发平台。在软件上,为了兼顾其他高级语言开发者的习惯,NI 还推出了 LabWindows / CVI、ComPonentworks 等交互式开发平台。经过 18 年的发展,NI 公司从正式发布 LabVIEW 1.0 到目前的 LabVIEW 7.1,几乎不到两年就推出一个新版本,可见虚拟仪器技术进步的迅速。美国 HP 公司的 HP VEE、Tektronis 公司的 Ez Test 和 Tek TNS 以及美国 HEM Data 公司的 Snap-Marter 平台软件,也是国际上公认的优秀虚拟仪器开发平台。目前,虚拟仪器技术已经在军事、航空、航天、通信、测试与测量、现代汽车、半导体、生物医学等世界范围的众多领域得到广泛应用,如图 1 所示。

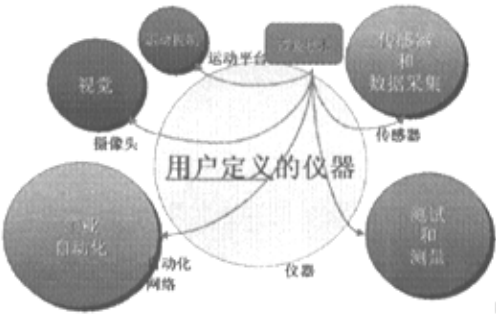


图 1 虚拟仪器技术的应用领域

1.1.2 虚拟仪器定义

所谓虚拟仪器,是指在以通用计算机为核心的硬件平台上,由用户设计定义,具有虚拟面板,其仪器的大部分测试功能由测试软件实现的一种计算机仪器系统。仪器的面板由显示在计算机上的软面板来代替,信号的获取和信号的分析、处理、存储及打印等功能完全由软

件来实现。其实质是利用计算机显示器的显示功能来模拟传统仪器的控制面板，以多种形式表达输出检测结果；利用计算机的软件功能实现信号数据的运算、分析和处理；利用 I/O 接口通信设备完成信号的采集与传输，最终完成各种测试功能^[2]。

1.1.3 虚拟仪器的发展前景

虚拟仪器作为现代仪器仪表发展的方向，已迅速成为一种新的产业，尤其在发达国家中发展更快，其设计、生产和使用已经十分普及。美国是虚拟仪器的诞生地，也是全球最大的虚拟仪器制造国。到 1994 年底，虚拟仪器制造厂已达 95 家，共生产 1000 多种虚拟仪器产品，销售额达 2.93 亿美元，占整个仪器销售额 73 亿美元的 4%。到 1996 年，虚拟仪器已在仪器仪表市场中占有 10% 的份额。生产虚拟仪器的主要厂家 NI、HP 等公司，这些公司生产有数百个型号的虚拟仪器产品。同时，虚拟仪器系统及其图形化编程语言，在美国已作为各大学理工类专业学生的一门必修课程。

国内虚拟仪器研究的起步较晚，最早的研究也是从引进消化 NI 的产品开始。但从 90 年代开始，国内的一些大学相继开展了虚拟仪器系统的研究与开发工作，如：哈尔滨工业大学、重庆大学、国防大学、成都电子科技大学、中国科技大学等，正在积极开展虚拟仪器的研究工作，研制具有自主知识产权的虚拟仪器软件平台，并开发相应的虚拟仪器设备。国家自然科学基金委员会已将虚拟仪器研究作为现代机械工程科学前沿学科之一，并被列为“十五”期间优先资助领域。国内专家预测，未来的几年内，我国将有 50% 的仪器为虚拟仪器。不久，我国将会有大批的企业使用虚拟仪器系统对生产设备的运行状况进行实时监测。随着微型计算机的发展，各种有关虚拟仪器开发软件不断诞生，虚拟仪器将会逐步取代传统的测试仪器而成为测试仪器的主流。

1.1.4 虚拟仪器技术的优点

虚拟仪器技术就是基于 PC 技术的、用户自定义的测试和测量解决方案，其 4 大优势在于：性能强性价比高、扩展性强、开发时间短，以及出色的集成功能^[3]。

性能强性价比高

以软件为主的测量系统，充分利用了常用台式计算机及工作平台的计算、显示和互联网等的强大功能。虽然 PC 机和集成电路技术在过去的 20 年里有着显著的发展和提高，但是，软件才是在功能强大的硬件基础上创建虚拟仪器系统的真正关键所在。虚拟仪器是在 PC 技术的基础上发展起来的，所以降低了系统的开发成本和维护成本，而且完全“继承”了以现

成即用的 PC 技术为主导的最新商业技术的优点，包括功能强大的处理器和文件 I/O，在数据导入磁盘的同时就能实时地进行复杂的分析。随着数据传输到硬驱功能的不断加强，以及与 PC 总线的结合，高速数据记录已经较少依赖大容量的本地内存。以一台 60G 的示波器为例，在采用虚拟仪器技术的情况下，只要将一台基于 PC 的数字转换器放置在 PC 机中，就能以高达每秒 100MB 的速度将数据导入磁盘。

另外，由于传送的信号大多是数字信号，数据的处理几乎是软件实现的，从而大大降低了环境干扰和系统误差的影响。

扩展性强

虚拟仪器的硬、软件具有开放性、模块化、可重复使用及互换性等特点。为提高测试系统的性能，可以方便地加入一个通用仪器模块或更换一个仪器模块，而不必购买一个全新的系统，大大有利于测试系统的功能扩展。

虚拟仪器改变了传统仪器由厂家定义、用户无法改变的模式。系统的功能、规模等可以根据用户的需要，通过软件的修改、增减，方便地同外设、网络及其它应用连接，为虚拟仪器加入新的测量功能，故虚拟仪器可作为许多仪器设备来使用。

开发时间短

在驱动和应用两个层面上，优秀虚拟仪器开发平台已经将其高效的软件构架与计算机、仪器仪表和通讯方面的最新技术结合在一起，给用户提供最方便的、最灵活的操作以及强大的功能，让用户轻松地配置、创建、部署和维护高性能、低成本的测量和控制解决方案。

另外，由于充分利用了计算机技术，将信号的分析、显示、存储、打印和其它管理集中交由计算机来处理，完善了数据的传输、交换等性能，使得组建系统变得更加灵活和简单，增强了数据处理能力。

出色的集成

虚拟仪器技术从本质上说是一个集成的软硬件概念。随着产品在功能上不断地趋于复杂，工程师们通常需要集成多个测量设备来满足完整的测试需求，但是这些不同设备间的连接和集成总需耗费大量时间，并不是轻易可以完成的。虚拟仪器软件平台为大部分 I/O 设备提供了标准的接口，例如数据采集、视觉、分布式 I/O 等等，帮助用户轻松地将多个测量设备集成到单个系统，减少了任务的复杂性。

1.1.5 虚拟仪器的硬件系统

虚拟仪器的硬件平台一般由两部分组成：计算机和 I/O 接口设备。计算机，一般为一台

PC 机或工作站，它是硬件平台的核心。I/O 接口设备，它主要完成对被测输入信号的预处理（如采集、放大、模数转换）。根据不同的总线及其相应的 I/O 接口设备，利用 PC 机总线的数据采集卡版 (DAQ)、GPIB 总线仪器、VXI 总线仪器模块、PXI 总线仪器和串行总线仪器等，虚拟仪器硬件系统构成方式大致如图 2 所示^[4]。

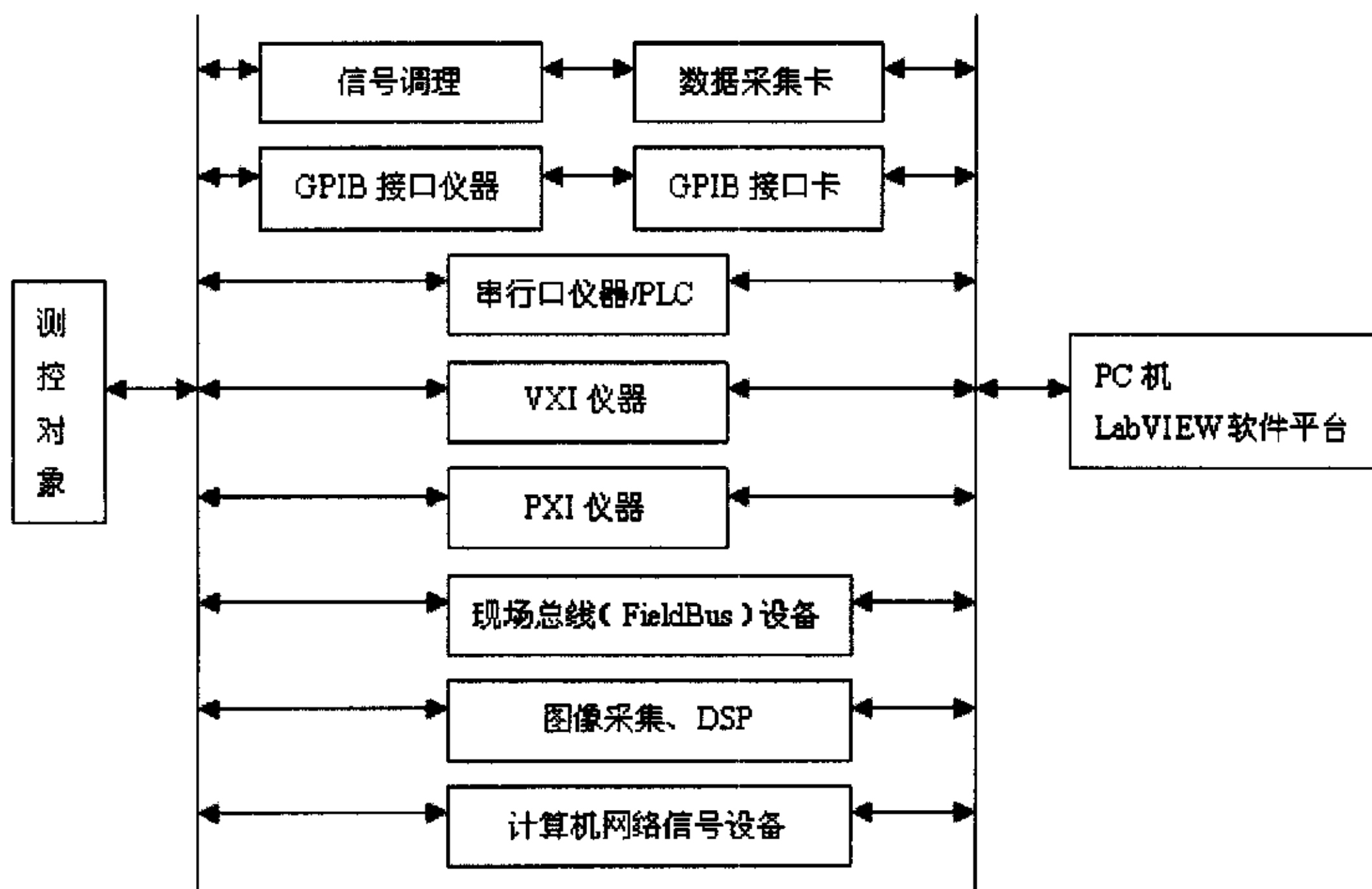


图 2 虚拟仪器的硬件系统

1.1.5.1 计算机系统

PC 机或工作站是虚拟仪器系统的核心，它完成数据的处理和结果的显示。利用计算机图形显示技术和多媒体技术，将复杂的数据计算和数据处理推向后台，把测控的结果用数字、曲线、图象、图形、声音等形式提供给用户。

1.1.5.2 接口硬件设备

DAQ 系统

它是以数据采集板、信号调理电路及计算机为仪器硬件平台组成的插卡式仪器系统，这种系统采用 PCI 或 ISA 计算机本身的总线，将数据卡 / 板插入计算机的相应插槽中即可，容易普及使用。

GPIB 系统

GPIB 技术是 IEEE488 标准的虚拟仪器早期的发展阶段，它的出现使电子测量由独立的单台手工操作向大规模自动测试系统发展，典型 GPIB 系统有一台 PC 机，一块 GPIB 接口卡和若干台 GPIB 式的仪器通过 GPIB 电缆连接而成。GPIB 测试系统的结构和操作命令简单，

适合在要求高精度，但不要求对计算机高速传输的情况下应用。

串口系统

它是以 Serial 标准总线仪器与计算机为硬件平台的系统。包括符合 RS-232 标准的 PLC 系统和单片机系统。

VXI 系统

它是以 VXI 标准总线仪器模块与计算机为仪器硬件平台的系统。VXI 总线是高速计算机总线 VME 总线在仪器领域的扩展。由于它具有标准开放、结构紧凑、数据吞吐能力强、定时和同步精确、模块可重复利用、众多仪器厂家支持等优点，很快得到广泛的应用。

PXI 系统

PXI 总线在 PCI 总线内核技术上增加了成熟的技术规范和要求形成的，增加了多板同步触发总线的参考时钟、用于精确定时的星形触发总线以及用于相邻模块的高速通讯的局部总线。标准的 PXI 模块化仪器系统有 8 个插槽，还可与 CompactPCI 交互操作，可与 GPIB 或 VXI 集成，组成大规模、多用途的系统。

1.1.6 虚拟仪器的软件系统

虚拟仪器软件体系结构主要包含两个层次：用户应用程序和接口程序，其中接口程序是联系用户应用程序和底层硬件设备的基础。每一种接口程序都是为了提高编程灵活性和数据吞吐量而设计的。不管仪器使用的计算机或者操作系统是什么，最终编写的用户应用程序都是可移植的。

当给定计算机的计算能力和必需的仪器硬件平台后，构造和使用虚拟仪器的关键是应用不同软件实现不同的功能。虚拟仪器的应用软件包含以下四个方面：集成的开发环境、仪器的硬件接口和虚拟仪器的用户界面、仪器数据库。虚拟仪器最核心的思想：是如何利用计算机的软件和硬件资源，使得硬件技术软件化和虚拟化，最大限度地降低系统成本，增强系统的功能和灵活性^[5]。虚拟仪器软件系统框架从底层到顶层包括三个部分：VISA 库、仪器驱动程序、应用软件，如图 3 所示。

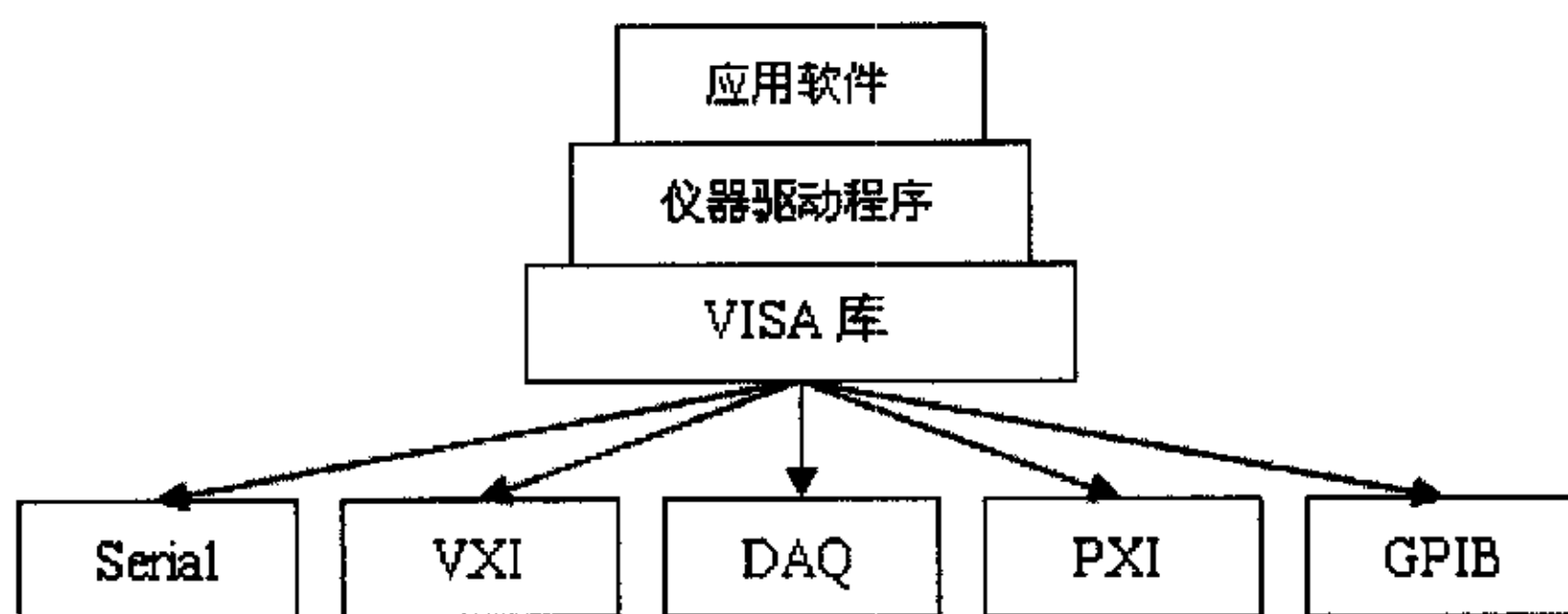


图 3 虚拟仪器的软件结构

1.1.6.1 VISA 库

VISA (Virtual Instrumentation Software Architecture) 库实质就是标准的 I/O 函数库及其相关规范的总称，一般称这个 I/O 函数库为 VISA 库。它驻留于计算机系统之中，执行仪器总线的特殊功能，是计算机与仪器之间的软件层连接，用来实现仪器的控制。对于仪器驱动程序开发者来说，VISA 库是一个可调用的操作函数库或集合。

1.1.6.2 仪器驱动程序

仪器驱动程序是完成对某一特定仪器的控制与通信的软件程序集合，是连接上层应用软件和底层输入/输出 (I/O) 软件的纽带和桥梁。每个仪器模块都有自己的仪器驱动程序，仪器厂商将其以源码的形式提供给用户，用户在应用程序中调用仪器驱动程序。

1.1.6.3 应用软件

应用软件建立在仪器驱动程序上，主要包括仪器面板控制软件 and 数据分析处理软件，完成利用计算机强大的图形功能实现虚拟仪器面板，给用户提提供操作仪器、显示数据的人机接口，及数据采集、分析处理、显示、存储等任务。

虚拟仪器的应用软件由用户编制，可采用各种编程软件，如 C、BASIC、Delphi 等。在当今这个信息时代，提高软件编程效率的关键是采用面向对象的编程方式，可视化编程语言环境 Virtual C++、Virtual Basic 的推出，在软件开发上迈出了可喜的一步。目前，基于 C 语言的 LabWindows/CVI、加载在 Virtual Basic 下的 Component Works 以及 Electronics Workbench V4.0 和 VisualLab 等，都使 VC 和 VB 成为强大的虚拟仪器开发平台。此外，NI 公司还和 Microsoft 合作开发了 Measurement Studio，专门为熟悉 VB 或 VC++ 语言的工程技术人员所设计的，用户可以根据自己的需要选用 VB 或 VC++。Measurement Studio 集成了开发虚拟仪器所需的控件对象、过程、方法或函数，这为具有 VB 或 VC 编程经验的用户开发虚拟仪器提供了极大的方便^[6]。同时，LabVIEW、HPVIE 等 G 语言的推出也使得虚拟仪器开发更加简单易行。

虚拟仪器软件的一个关键要求是能开发可重用的代码模块,并允许用户从其它代码模块中分级调用。

1.2 应用软件解决方案

1.2.1 Visual C++环境下的虚拟仪器软件设计

Visual C++提供了能使高级语言与汇编语言混合编程的编译系统,使得编程人员对接口软件的开发较为方便^[7]。而且,用 C++编写的接口软件能够不作任何修改地被其它任何语言编写的用户应用程序所调用,这非常有利于用户应用程序的开发,也便于接口软件的维护与升级。C++与其他高级语言有良好的接口,便于程序间的相互调用。

以数据采集卡为例,简述 Visual C++环境下的虚拟仪器软件设计原理,如图 4 所示:

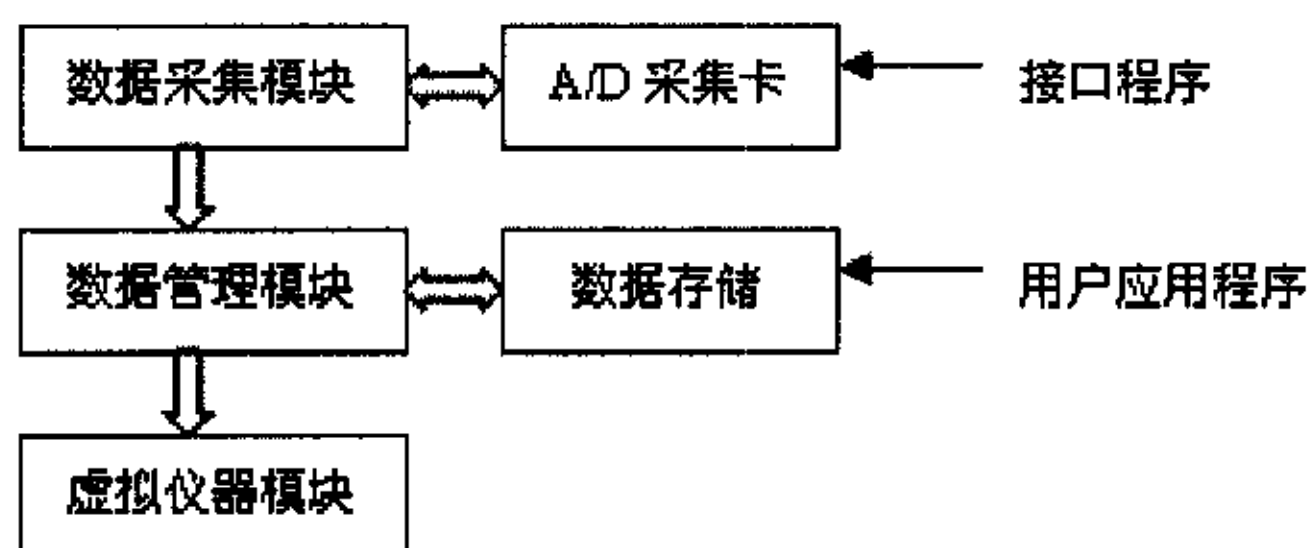


图 4 虚拟仪器软件总体设计框图

数据采集模块的主要任务是获取初始化采样卡、给定采样卡基址、定义控制字节,然后启动采集,对数据口进行读、写操作,最后将采集的数据送入数据管理模块。

数据管理模块接受来自 A/D 采样卡或采样数据文件的采样数据,将数据进行预处理和变换后送入虚拟仪器模块,经过处理后的采样数据如何显示的任务交给虚拟仪器模块的来完成。

虚拟仪器模块对应于具体的虚拟仪器。

1.2.2 Visual Basic 环境下的虚拟仪器软件设计

Visual Basic 是一种具有很好图形用户接口 (GUI) 的程序设计语言,同时又是一种完全结构化程序设计的面向对象的程序设计语言,综合运用了 Basic 语言结构化特点及 GUI 可视化设计工具,各种控制器、下拉式菜单、窗口、对话框等,非常适合虚拟仪器前面板的设计开发。

VB 提供了丰富的 API (应用程序接口) 函数。数据的采集由硬件完成,通过接口程序在 VB 下形成数据文件,运用各种不同的数据处理软件,对数据进行处理,然后利用 VB 丰

富的绘图功能在虚拟界面上显示动态波形。

1.2.3 基于 ActiveX 控件的虚拟仪器开发

简单地说, ActiveX 是一种体系结构, 它允许使用不同编程语言开发的软件组件在网络环境中相互操作。ActiveX 体系结构使用了 Microsoft 的组件对象模型及分布式组件对象模型标准——COM, 允许不同的组件对象实现本地相互交谈, 而 DCOM 则提供在网络上对象之间的通信^[8]。

ActiveX 引入了“软件组件”的概念。因此, 软件不必再编写成一个大型完整且独立运行的系统。为了能使软件完成更大更复杂的任务, 将其模块化、独立化, 这种模块就是“组件”。各组件完成特定的任务, 其发展方向是能跨应用程序、跨平台、跨网络运行。组件对象技术彻底改变了目前软件生产开发的模式, 这种技术节省了开发时间、资金, 使程序的二次开发变得更加简单高效。

基于软件组件的概念, 将虚拟仪器按功能划分成若干模块, 如采集模块、分析模块、显示模块。将仪器软面板视作一系列开关、旋钮、波形显示面板等的集合。将各个实现不同功能的模块创建成 ActiveX 控件, 于是, 构造虚拟仪器的工作便是按测试要求将各种 ActiveX 控件组合在一起的过程。当测试系统要求改变时, 只需将 ActiveX 控件重组。这一技术充分体现了虚拟仪器不同于传统仪器的一些优点: 用户自己定义、可重复利用。

目前, 开发 ActiveX 控件工具有 VB, VC++, Delphi 等。利用 ActiveX 控件可创建基于网络平台的虚拟仪器。

1.2.4 LabWindows/CVI 下的虚拟仪器开发

LabWindows/CVI 是 NI 公司推出的一套面向测控领域的软件开发平台。它以 ANSI C 为核心, 将功能强大、运用灵活的 C 语言平台与数据采集、分析和表达等测控专业工具有机地结合起来。它的集成化开发平台, 交互式编程方法, 丰富的控件和库函数大大增强了 C 语言的功能, 使熟悉 C 语言的开发人员在组建检测系统、自动测量环境、数据采集系统、过程监控系统时有一个理想的软件开发环境。LabWindows/CVI 也是开发 LabVIEW 驱动程序的最好工具^[31]。

LabWindows/CVI 本身虽不能提供用于现代数字信号分析和处理技术的功能函数和模块, 但它是一种文本式编程语言, 可利用向导开发 IVI(Intellectual Virtual Instruments, 智能虚拟仪器)仪器驱动程序和创建 ActiveX 服务器, 如建立 LabWindows/CVI 与 MATLAB 之间

的接口,调用 MATLAB 功能函数,运行 MATLAB 环境下的程序,以实现 LabWindows/CVI 与 MATLAB 的混合编程,同时,它可以为其它工程开发 C 目标模块、动态链接库(DLL)、C 语言库。其次,可生成脱离 LabWindows/CVI 开发环境的虚拟仪器,用户最终看见的是和实际的硬件仪器相似的操作面板;LabWindows/CVI 具有强大的网络功能,支持常用的网络协议,方便网络仪器、远程测控仪器的开发;对于经典数字信号处理技术,LabWindows/CVI 提供了用于仪器控制、数据采集和分析的交互式 ANSI C 编译软件包,用于快速样机开发的代码生成工具和内部编译器以及包含 DAQ、GPIB、PXI、VXI、RS232/485 等各种仪器通讯总线标准的所有功能函数。

1.2.5 基于 G 语言的虚拟仪器软件设计

以上虚拟仪器软件设计方案虽然行之有效,但是对于普通的计算机用户或工程师来说,使用起来有一定的难度。然而,相应的测试软件开发环境长期以来并不理想。用户花在编制测试软件上的工时与费用相当高,即使利用 VC、VB、Delphi 等工具软件也会感到开发周期偏长。以图形化软件编程方法和集成开发环境为标志的虚拟仪器开发环境是虚拟仪器技术研究内容之一。近年来,各大仪器公司开发出了许多高效的基于图形编程的虚拟仪器开发平台,其中最优秀的当属 NI 公司的 LabVIEW 和 HP 公司的 HPVee,它们都是面向仪器、检测、工业监控的图形化通用编程语言。

HPVee 是采用行业人员所熟悉的图标等图形化符号来代替文本程序语言,把复杂繁琐、费时的语言编程简化成简单、直观、易学的图形编程。与传统的编程技术相比,它可节省 80% 的编程时间,程序运行速度却几乎不受影响。Vee 编程语言还允许调用 VC、VB 等语言编写的程序,具有一定的开放性。HPVee 提供了四十多个厂家制造的 450 种仪器驱动程序库,还支持用户根据需要用驱动器写入工具(DWT)编写新的仪器驱动程序。这些仪器包括 GPIB 仪器、VXI 仪器,用户可随意调用仪器驱动器图象组成的方框图,以选择任一厂家的任一仪器。HPVee 还具有一组数学运算及分析目标模块库,如信号处理、信号发生、数字滤波、曲线拟合等目标模块。编程时,只需调用所需要的驱动器图象、目标模块等,输入相关参数,并按人们所熟悉的测试流程用线条连接起来,就基本上完成了编程工作^[9]。

LabVIEW 采用数据流框图编程,适合任何编程任务,也是一种通用编程语言。LabVIEW 强大的硬件驱动、图形显示能力和便捷的快速程序设计为过程控制和工业自动化应用提供了优秀的解决方案。目前,已经成为测试与测量领域的工业标准,通过 GPIB、VXI、PLC、串行设备和插卡式数据采集板构成实际的数据采集系统。它具有工业界最大的仪器驱动程序

库，同时还支持通过 Internet、ActiveX、DDE 和 SQL 等交互式通信方式实现数据共享，使得复杂的测试任务变得简单易行^[5]。

1.3 LabVIEW 简介

1.3.1 基于图形的集成化程序开发环境

LabVIEW 是一种基于图形的集成化程序开发环境，实现了虚拟仪器的概念，它是一套专为数据采集与仪器控制、数据分析和数据表达而设计的图形化编程软件，增强了用户在标准的计算机上配以高效经济的硬件设备来构建自己的仪器系统的能力。LabVIEW 是一种结构化解释型开发平台。结构化是指 LabVIEW 的程序完全支持顺序结构、循环结构和条件结构 3 种标准结构^[38]。同时又是由模块化的形式组成的，它的每一个子程序都称为一个 VI，子程序间可以互相调用。所谓解释型是指用 LabVIEW 开发的软件、无法在 Windows 操作系统下直接运行，所以软件必须在 LabVIEW 的平台支持下运行，也就是说 LabVIEW 不能生成真正的可执行(.exe)文件。

1.3.2 面向对象的编程语言

LabVIEW 又是一种面向对象的编程语言——G 语言，程序代码是框图的形式。它像 JAVA 或 C++ 等其它计算机高级语言一样，是一种通用编程系统，具有各种各样、功能强大的函数库，包括数据采集、GPIB、串行仪器控制、数据分析、数据显示及数据存储，甚至还有目前十分热门的网络功能。

LabVIEW 具有完善的仿真、调试工具，如设置断点、单步等，其动态连续跟踪方式，可以连续、动态地观察程序中的数据及其变化情况，比其它语言的开发环境调试更方便、更有效。与其它计算机语言相比，LabVIEW 有一个特别重要的不同点：其它计算机语言都是字符代码行，而 LabVIEW 采用的是图形化编程语言。对于熟悉仪器结构和硬件电路的硬件工程师、现场工程技术人员及测试技术人员来说，编程就像设计电路图一样。

另外，LabVIEW 支持多种系统平台。在任何一个平台上开发的 LabVIEW 应用程序可直接移植到其它平台上。

1.4 VI 程序简介

使用 LabVIEW 开发平台编制的程序称为虚拟仪器程序，简称为 VI。一个 VI 由交互式用

户接口、数据流框图和图标连接端口组成，各部分功能如下^[5]：

1. VI 的交互式用户接口与真实物理仪器面板相似，称为前面板，如图 5 所示。前面板由控制、指示和修饰（Decoration）构成，可以包含旋钮、刻度盘、开关、图表和其他界面工具，允许用户通过键盘或鼠标获取数据并显示结果。控制是用户设置和修改 VI 程序中输入量的接口，在某种意义上相当于 C 语言中的输入语句 `scanf`；指示则用于显示由 VI 程序产生或输出的数据，相当于 C 语言中的输出语句 `printf`；修饰的作用仅是将前面板点缀得更加美观，并不能作为 VI 的输入或输出来使用。

2. VI 从数据流框图接收指令。框图是一种解决编程问题的图形化方法，如图 6 所示，事实上，它就是 VI 的程序代码。每一个前面板都有一个框图程序与之对应。框图程序由节点（Node）和数据连线（Wire）组成。节点是 VI 程序中的执行元素，类似于文本编程语言程序中的语句、函数或者子程序。节点之间由数据连线按照一定的逻辑关系相互连接，可定义框图程序内的数据流运动方向。

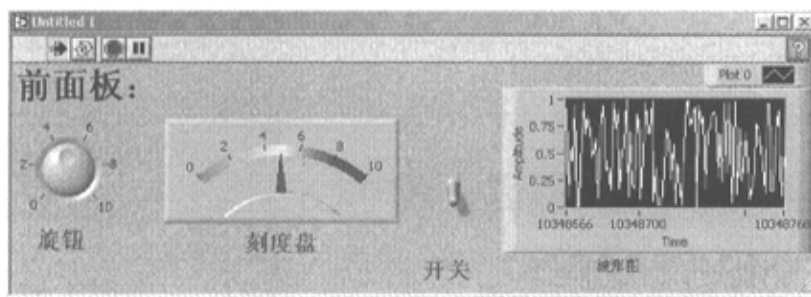


图 5 前面板

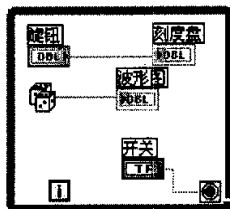


图 6 数据流框图

3. VI 模块化特性。一个 VI 既可以作为上层独立程序，也可以作为其他程序（或子程序）。当一个 VI 作为子程序时，称作 SubVI。VI 图标（如图 7 所示）和连接端口（如图 8 所示）的功能就像一个图形化参数列表，可以在 VI 与 SubVI 之间传递数据。

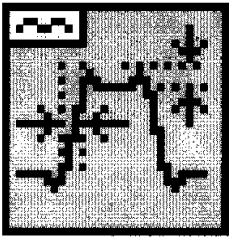


图 7 VI 图表

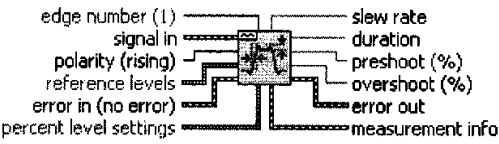


图 8 连接端口

第二章 基于 LabVIEW 的实验室管理系统设计

2.1 课题研究的内容和意义

随着测控技术和计算机技术的不断发展,越来越复杂的测试条件、高度自动化的工业化大生产迫切需要功能更强大、成本更低廉、系统更灵活的新一代测试仪器。从模拟技术向数字技术过渡、从单台仪器向多种功能仪器组合过渡、从完全由硬件实现仪器功能向软硬件结合方向过渡、从简单的功能组合向以个人计算机为核心的通用虚拟测试平台过渡、从硬件形式向软件包形式过渡,代表了今后电子测试仪器的发展方向。

实验是大学理工科教育的一个重要环节,是提高学生动手能力,培养创造能力和综合素质的一个最为有效的手段,很多学科都是以实验课程为基础的。缺少了实验的支持,工科学科的科研活动就无法进行。然而,实验系统是一个资源高度密集的系统,这包含三个方面:

1. 实验设备需要大量资金的投入。
2. 实验体系的建立和维护,需要大量的人力成本投入。
3. 实验教学需要大量的教师资源。

在个人计算机出现以前,高校实验室一般采用功能单一、价格昂贵的传统仪器来供学生实验中使用,投资费用较大,仪器的维护和保养也比较麻烦。随着高等教育的不断普及和发展,对工科专业学生的实验教学要求进一步的提高,加剧了实验系统不足的现状。目前,在我国的不少高等院校中所使用的教学实验仪器,仍然是传统的物理实验仪器为主。自 20 世纪 90 年代以来,在计算机技术的推动下,以虚拟仪器为标志的通用化、智能化和网络化测量仪器及测试系统得到了迅猛发展,使得测量仪器和数据采集系统的设计方法和实现技术产生了深刻的变化。为了改善实验条件、改革实验教学方法和实验手段、更新实验教学内容、提高实验教学课程的开设水平,利用计算机技术和网络技术,把虚拟仪器引入实验教学已经成为一种必然趋势。

学校的规模不断地扩大,学生的数量相应地增加,学生实验的管理是一项比较复杂的工作。传统的实验管理模式已经不能适应现代化管理的需要,手工的数据收集和处理显得非常困难。随着电子实验室的逐步开放,虚拟网络实验室的建设,实验者不仅仅只局限于在校的

本专业学生，或只限于在实验室本地进行实验，这样，用户信息的收集和管理、实验数据的管理已经不是传统管理模式所能胜任的工作。因此，结合数据库技术和网络技术建立一套实验管理系统是非常必要的。实验管理系统是一个基于网络的自动化信息系统，对用户信息、实验数据进行有效的管理，实时地记录用户的实验过程，方便实验教师管理实验、上交实验报告，批阅实验报告，学生也可以随时随地（有计算机和网络的地方）的进行实验，使实验室实现真正意义上的开放。

2.1.1 研究内容

1. 以 LabVIEW 7 Express 为平台开发出一系列虚拟实验仪器，如虚拟函数信号发生器、虚拟双通道示波器、虚拟毫伏表、数字 I/O 等等。
2. 结合虚拟仪器，用 Tomcat 作为 Web 服务器，后台使用 MySQL 数据库，进行实验管理系统的开发。

2.1.2 方案制定

应用程序之间的通信模式主要有三种：客户机/服务器(C/S)模式、浏览器/服务器(B/S)模式和 C/S/B 模式。C/S 模式主要由客户应用程序 (Client)、服务器管理程序 (Server) 和中间件 (Middleware) 三个部件组成，它是一种“胖客户端/瘦服务器”模式，多用于集散控制系统。客户应用程序是系统中用户与数据进行交互的部件，服务器程序负责有效地管理系统资源，中间件负责联结客户应用程序与服务器管理程序，协同完成作业。系统的主要负担由客户端来完成，在不同的客户机端可以安装不同应用程序。从网络分层体系结构出发，C/S 只是应用层的一种标准，它需要其他低层结构的支持，才能完成网络通信。这种模式有着安全性高、网络通信负荷小、速度快等优点。但是，更新和维护的工作量较大，扩展性和灵活性较差。如果系统要进行升级，所有的客户端程序都需要进行修改，系统规模较大时，通知所有的用户升级他们的程序简直就是一场噩梦。

B/S 模式是一种以 Web 技术为基础的新型网络模式，其主要的是用于在 web 网页上发布数据。它有三层体系：第一层是用户和系统的接口程序，一般为通用的浏览器软件，如 Microsoft IE 和 Netscape Navigator 等等。在这一层，通过表单实现和服务器之间的交互。第二层 Web 服务器将启动相应的进程来响应浏览器的请求，通常还要与数据库服务器进行交互处理，然后将请求的结果返回给客户机浏览器。第三层的任务类似于 C/S 模式，负责协调不同的 Web 服务器发出的请求。通常用户以此扩展虚拟仪器的网络化功能。这种模型下，用

户端只需要安装浏览器，就可以直接通过访问远端地址。开发人员的主要工作集中在服务器端的开发上，不必对客户端进行开发和设计，极大地简化了开发和设计难度。

C/S/B 模型，以上两种模型虽然能够满足大多情况的要求，但是对于大型集散控制的发布和远程控制，还是存在一些不足。为弥补这种不足，以适应测控系统信息化和网络化的需要，自然出现了客户/服务器/浏览器模型，它是 C/S 模型和 B/S 模型的结合和扩展^[32]。

针对本课题研究的内容，选用 B/S 模式，提出以下两种方案：

方案一：利用 LabVIEW 自带的 Server 或利用第三方软件 LabVNC 进行网页发布。整套系统由 LabVIEW 语言实现，后台数据库使用 LabSQL。

优点：易于 LabVIEW 程序与服务器、数据库之间的通信。

缺点：开发成本高，NI 公司的工具包价格昂贵，不利于程序的扩展。

LabVIEW 7.0 Express 自带的 Server 完全版允许同时连接到 Web Server 上的 Client 端的默认数目为一个，如果要使更多的 Client 端同时连接到 Web Server，必须向 NI 公司购买授权。

利用第三方软件 LabVNC 配合 LabVIEW 进行网页发布，目前同样不能解决多用户连接的问题。LabVNC 是基于 VNC (virtual network computing, 虚拟网络计算作业) 协议的、免费、跨平台、开放源代码的 Remote Panels 发布工具。但是 LabVNC 的缺点是，只能打开主 VI 的前面板窗口，若主 VI 之下还有 SubVI 窗口，则只能在服务器上打开，在客户端看不到。多个客户机同时访问服务器上的 Remote Panels 时，只有第一个获得 LabVNC 访问权的客户机成功连接，其他的客户机只能等待，直到第一个客户机与服务器断开连接为止，然后动作快的那台客户机可以获得 LabVNC 的访问权^[1]。

方案二：在 LabVIEW 平台上进行虚拟仪器开发，利用 Tomcat 应用服务器进行网页发布，后台使用 MySQL 数据库进行数据管理。

优点：易于网页发布，有利于管理系统的功能扩展。

缺点：LabVIEW 与服务器通信困难。

目前，比较流行的 Web 服务器有 Apache 公司的 Tomcat、Apache，Microsoft 公司的 IIS 等。最常用的三种动态网页语言：JSP (Java Server Pages)、ASP (Active Server Pages) 和 PHP (Person Home Pages)。它们都属于服务器脚本语言，而 JSP 是一种服务器端嵌入 Java 代码的脚本语言，拥有 Java 语言的优点：“一次编写，随处运行”，它比 ASP、PHP 执行速度更快^[10]。JSP 与其他各种类型的数据库连接方便。Sun 公司和其它数据库厂商共同开发了各种数据库的连接桥，JDBC API 提供数据库编程接口，非常适合管理系统的开发。

针对这种方案的缺点,提出的解决方法是:利用 LabVIEW 强大的文件读写能力将实验数据、实验过程中的实时波形以文件的形式输出到本地机。实验结束以前,通过 Tomcat 服务器,将这些文件上传到服务器,这样就避免了 LabVIEW 语言在实现管理系统方面的不足。

鉴于目前的软、硬件条件,选用第二种方案来实现基于 LabVIEW 的学生实验管理系统。

虚拟仪器的功能如同传统物理仪器,可以显示检测点的电压波形,对波形进行分析,学生可以根据分析结果记录波形的特征值。实验电路板由硬件电路实现,将待测点的信号通过数据采集卡引入计算机,虚拟双通道示波器对待测点的电压进行测量,显示其波形。在学生实验进行的过程中,利用 LabVIEW 强大的文件读写功能,将实验数据及波形实时地写入本地机器的文件中。整个系统结构如图 9 所示:

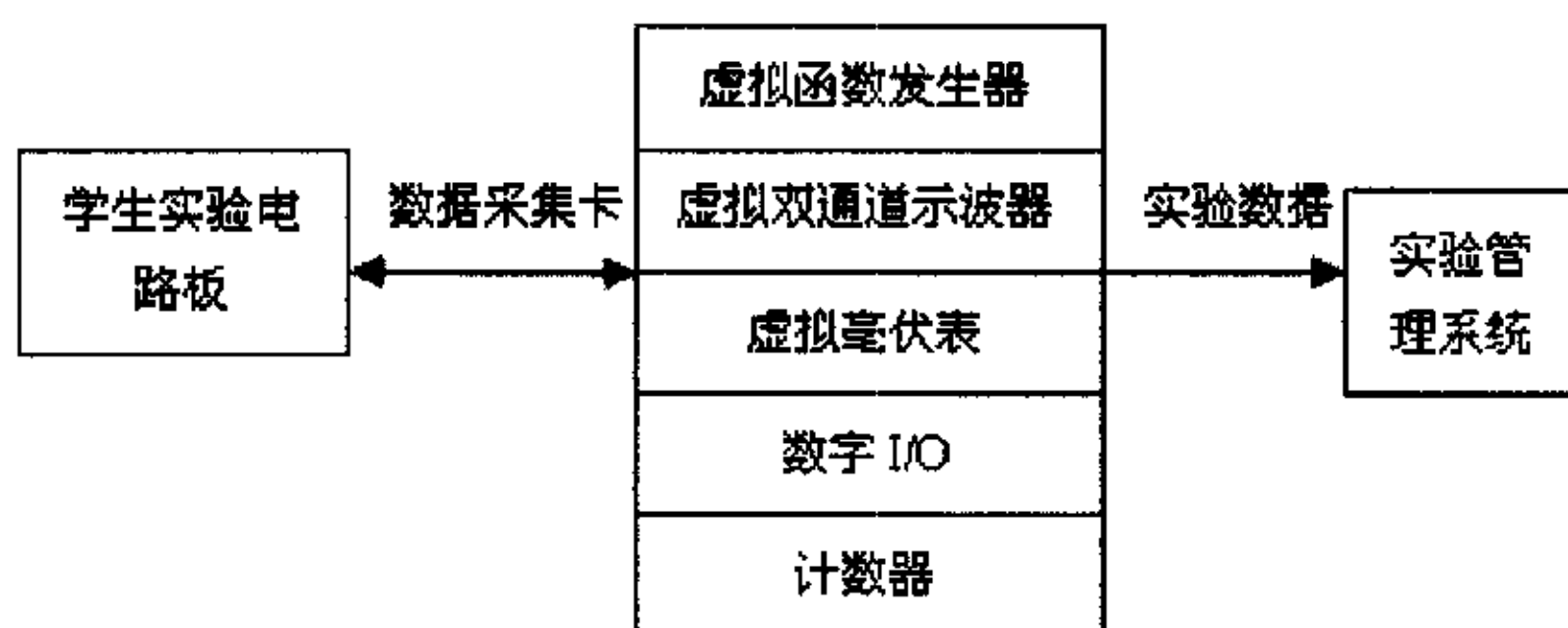


图 9 基于 LabVIEW 的实验室管理系统结构

实验管理系统采用的是 B/S/D 结构(浏览器/服务器/数据库),如图 10 所示。学生从学生客户端登陆进入实验,在实验进行的过程中,利用 LabVIEW 强大的文件读写功能,将实验数据及波形实时地写入本地机器的文件中。实验完成时学生按照实验要求填写实验数据,上传实验波形,提交实验报告给应用服务器。后台 MySQL 数据库配合 JSP 实现实验管理。

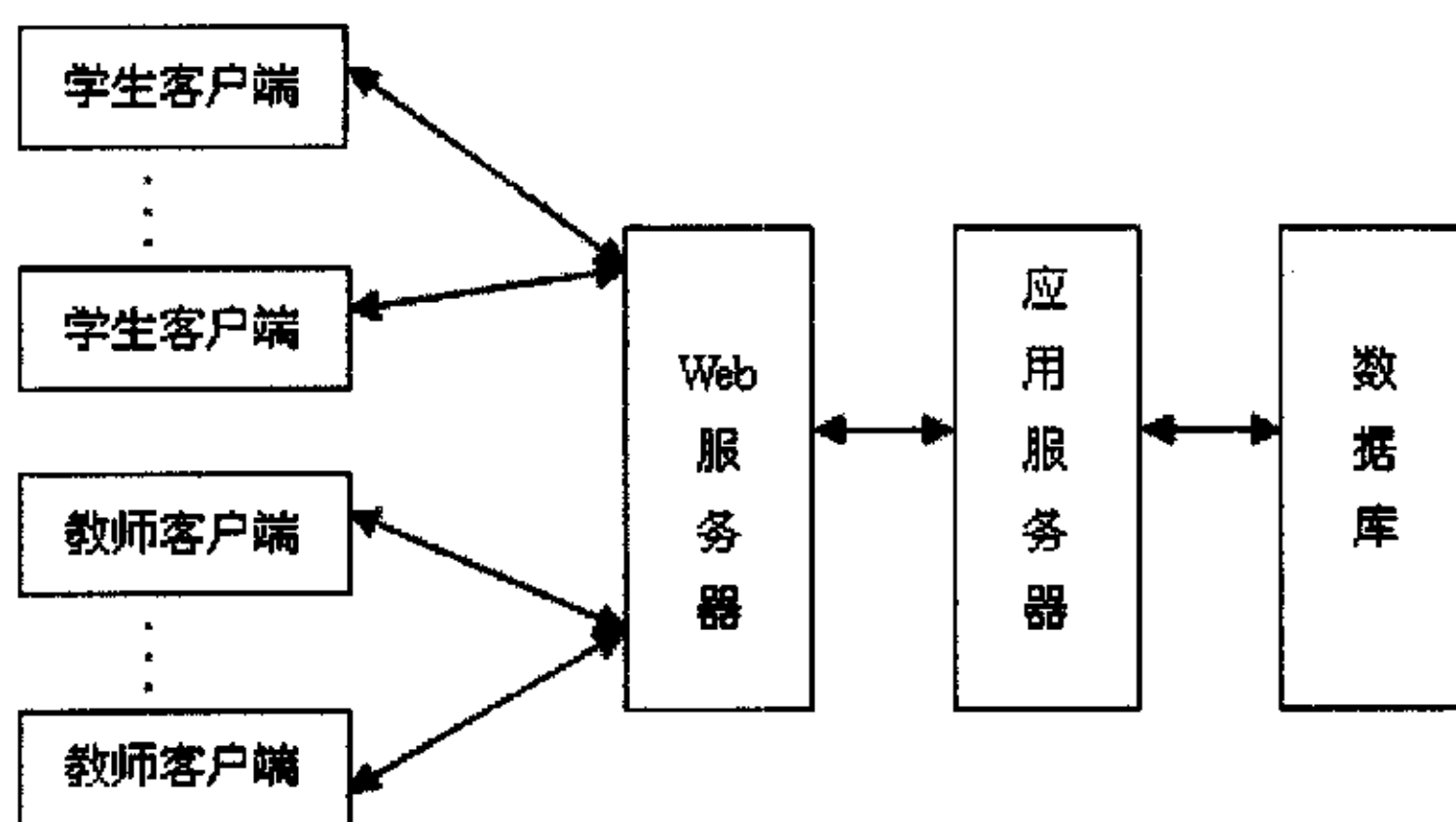


图 10 B/S/D 管理系统

2.2 虚拟仪器的实现

2.2.1 硬件系统

该项目所用的硬件有：学生实验电路板若干块、NI 公司的 PCI-6024E 数据采集卡一块、信号调理电路板一块、PC 机一台。

2.2.1.1 PCI-6024E 数据采集卡

PCI-6024E 数据采集卡是一块基于 32 位 PCI 总线的多功能数据采集卡，支持 DMA 方式和双缓冲方式，保证了实时信号的不间断采集与存储。支持单端和差分模拟信号的输入，信号的输入范围为 $\pm 10\text{V}$ 。提供了 16 路单端/8 路差动输入通道、2 路独立的 D/A 输出通道、8 线的数字 I/O、2 个 24 位的定时器/计数器等多种功能。附件包括一根 68 芯的数据线，一个型号为 CB-68LP 的转接板^[1]。

PCI-6024E 数据采集卡参数：

Analog Inputs: 16 SE/8 DI

Input Resolution: 12 bits

Sampling Rate: 200KS/s Polarity: Bipolar

Input Range: $\pm 10\text{V}$ (MAX) 和 ± 0.05 (MIN)

Analog Outputs: 2

Output Resolution: 12 bits

Output Rate: 10KS/s Polarity: Bipolar

Output Range : $\pm 10\text{V}$

Digital I/O: 8

Counter/Timer: 2, 24-bit

Triggers: Digital

Available DMA Channels: 1

Measurement Precision

Gain	Input Range	Precision ¹
0.5	-10 to +10 V	4.88 mV
1.0	-5 to +5 V	2.44 mV
10.0	-500 to +500 mV	244.14 μ V
100.0	-50 to +50 mV	24.41 μ V

¹ The value of 1 LSB of the 12-bit ADC; that is, the voltage increment corresponding to a change of one count in the ADC 12-bit count.

2.2.1.2 信号调理电路

信号调理电路首先对输入的信号进行放大或衰减，抬高电位和电位降低等调理，使待采集信号变换到 6024E 采集卡所要求的-10~10V 电压范围以内，以免电压过高或过低，损坏 6024 数据采集卡。信号由采集卡送出后，经输出保护电路后输出。

输入、输出保护电路分别如图 11-14 所示。

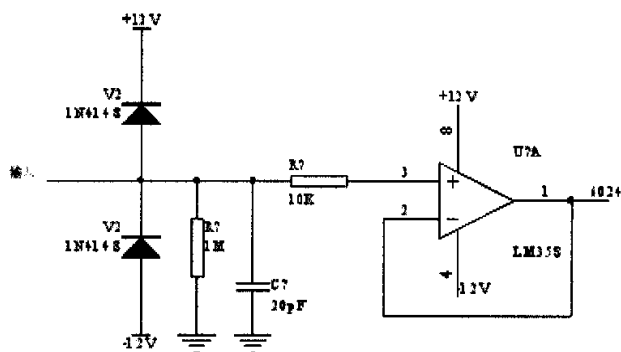


图 11 模拟输入保护电路

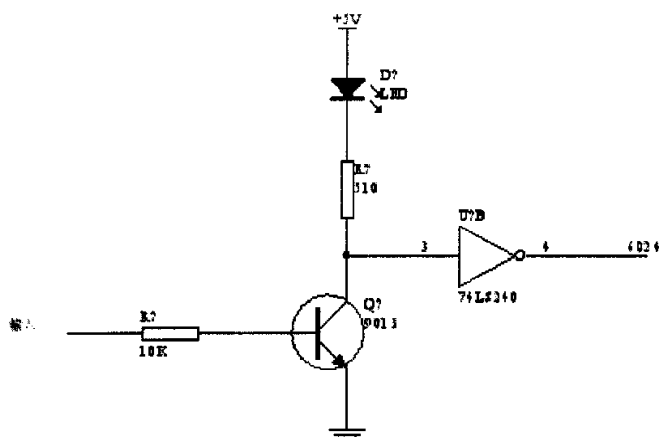


图 12 数字输入保护电路

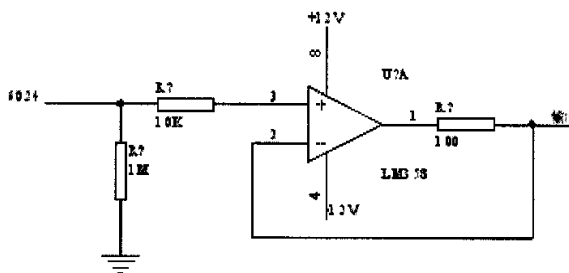


图 13 模拟输出保护电路

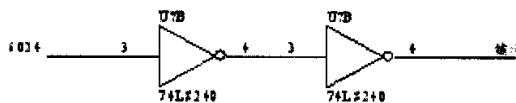


图 14 数字输出保护电路

2.2.2 软件编程

2.2.2.1 虚拟函数信号发生器

前面板设计：

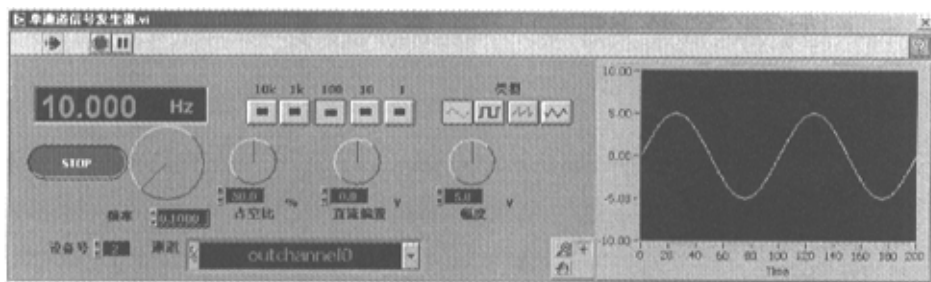


图 15 虚拟函数信号发生器前面板

虚拟函数信号发生器如图 15 所示，仪器主要功能包括通道选择、波形选择、频率调节、波形显示等。函数信号发生器可以产生四种类型的信号：正弦波、方波、锯齿波和三角波。

前面板上的四个旋钮分别用来控制输出波形的频率、方波的占空比、直流偏置和输出波形的幅度。频率旋钮从 0.1~1 连续可调，方波的占空比 0~100% 连续可调，直流偏置 -10V~10V 连续可调，波形的幅度 0~10V 连续可调；

说明：从前面板上看，输出波形的频率有 1Hz、10 Hz、100 Hz、1K Hz、10K Hz 等五个档位可调，因此，理论上输出波形的频率范围可以从 0.1Hz 至 10KHz。由于数据采集卡的限制，实际输出波形的频率达不到上述范围。由于 NI 公司的 PCI-6024E 型数据采集卡的输出采样速率最大只能达到 10KS/s，这样在设定输出波形数据长度时就只能限定在 10K Hz

以内的频率范围了。否则，频率调节旋钮将不能满量程可调。例如，如果选定输出波形数据长度为 20K，运行此 VI，调整频率旋钮时，你会发现当频率调节旋钮超过 0.5 的位置时，运行出错，LabVIEW 自动弹出一错误提示对话框。这样，输出波形的频率将不能连续。选择 1K Hz 档位时，采集卡输出波形的一个周期只有 10 个点；选择 10K Hz 档位时，一个周期只有 1 个采样点，所以，当选择 1K Hz 或 10K Hz 的档位时，输出波形严重失真。选择另外三个低档位时，具有较好的输出波形。

框图程序设计：

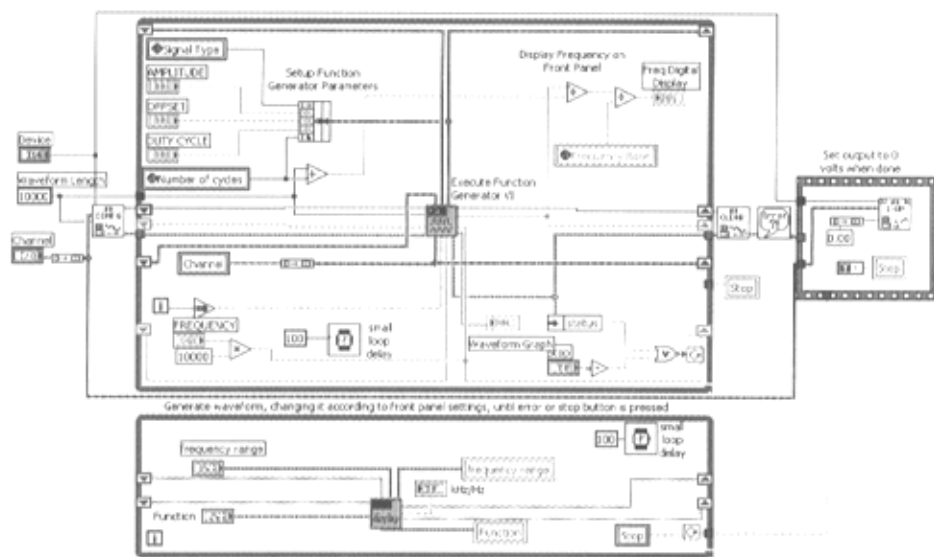


图 16 虚拟函数发生器程序框图

根据信号发生器的功能，框图程序包含以下几个模块：

1. I/O 配置模块

采用 DAQ 卡输出波形时，首先要进行输出通道的配置。如设置通道名称、通道类型、设备号、通道号、比例类型和输出范围等参数。框图程序中首先调用 AO Config.vi 进行软件配置，如设置缓存的大小，选择 DAQ 设备，选择输出通道等等。输出通道配置完毕，产生一模拟输出操作的任务号。

2. 波形输出模块

根据前面板输入的信号类型、频率、幅度、直流偏置、方波信号的占空比等参数来产生波形数据并经采集卡送到所选通道。

3. 频率显示模块

负责前面板频率档位调节按钮、波形选择按钮的切换。

波形输出频率的计算公式：
$$F = \frac{\text{实际刷新速率 (ticks per second)}}{\text{每个周期的数据点数}}$$

2.2.2.2 虚拟双通道示波器

前面板设计：

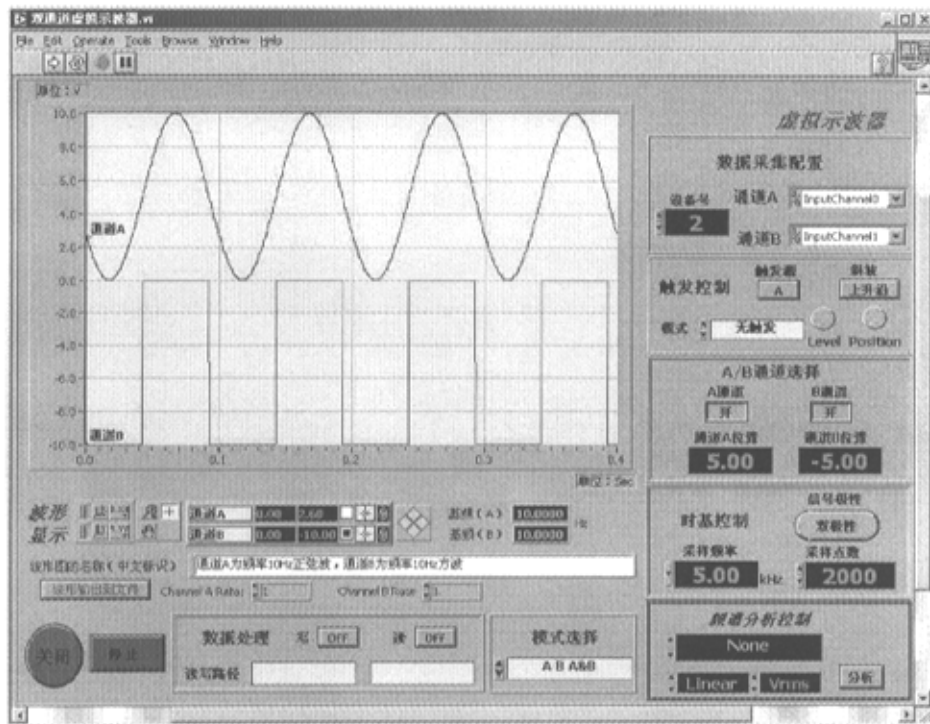


图 17 虚拟双通道示波器前面板

参照实际的物理示波器，所设计的虚拟双通道示波器前面板如图 17 所示。其中，图中所采集的信号由虚拟双通道函数信号发生器送出，通道 A 为频率 10Hz 的正弦波，通道 B 为频率 10Hz 的方波。仪器主要功能包括双通道信号输入、触发控制、通道控制、时基控制、波形显示、脉冲参数测量、频谱分析^[5]。

框图程序设计：

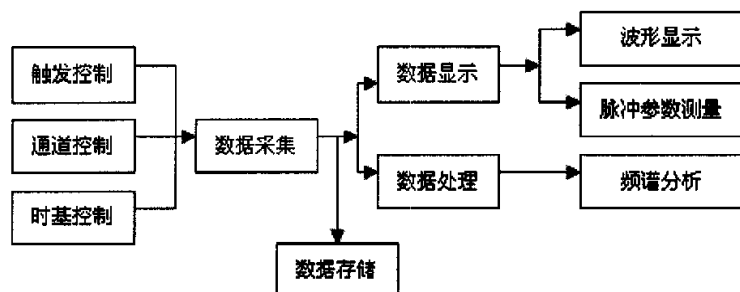


图 18 虚拟双通道示波器的功能结构图

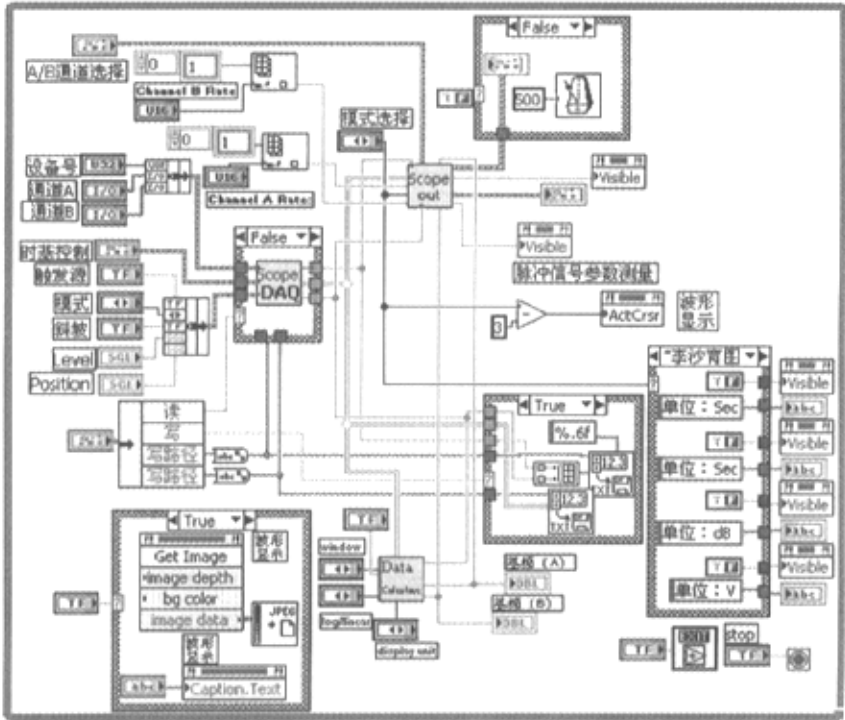


图 19 虚拟双通道示波器程序框图

根据双通道示波器的功能，框图程序包含以下几个模块：

1. 数据采集模块

数据采集模块是虚拟示波器的核心，主要完成数据采集的控制，包括触发控制、通道控制、时基控制等。触发控制包括触发源、触发模式、触发斜坡和电平控制。通道选择主要控制单通道或双通道测量。时基控制主要控制采集卡扫描率、每一通道扫描次数。

2. 数据显示模块

数据显示模块提供数据的三种显示方式，A B A&B 模式，李沙育图形模式和脉冲参数测量模式。前两种模式显示波形，后一种模式显示被测信号参数值。

3. 数据处理模块

对被测信号进行频谱分析，可实现的频谱分析控制包括：窗函数选择，提供了 9 种频谱分析窗；Log/Linear 选择，提供 2 种坐标显示模式；Display Unit 选择，提供 8 种单位。上述信号的频谱分析图如图 20 所示：

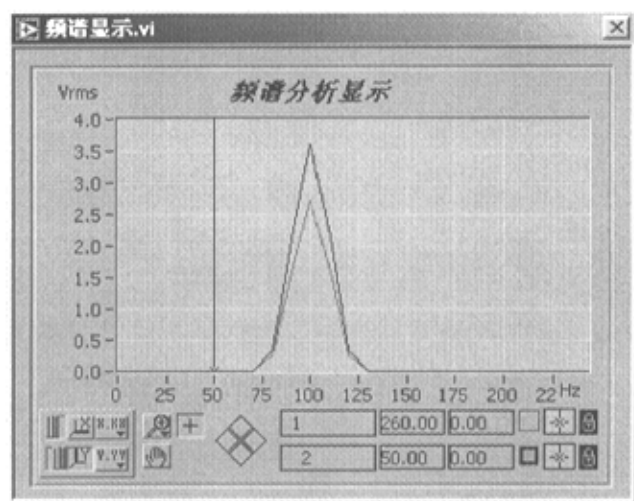


图 20 频谱分析

4. 文件处理模块

文件操作部分的“写”按钮按下时可将采集得到的数据输出到指定的文件，“读”按钮按下时可从指定的文件中读取数据，并在虚拟示波器上进行显示。

运行 VI 期间，点击“波形输出到文件”按钮，可以将采集所得到的波形以图片的形式保存。

2.2.2.3 虚拟毫伏表

前面板设计：



图 21 虚拟毫伏表前面板

程序框图设计：

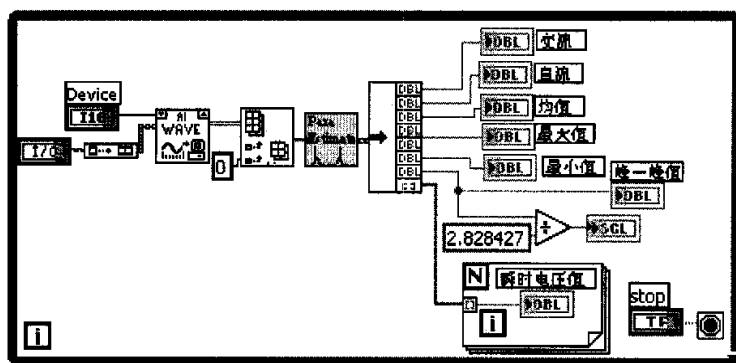


图 22 虚拟毫伏表程序框图

根据虚拟毫伏表的功能，框图程序包含两个模块：

1. 数据采集模块

数据采集模块采用 LabVIEW 函数库里的 AI WAVE 模块，按 1000scan/second 的速度采集 1000 个点，然后对其进行数据处理。

2. 数据处理模块

对采集的数据进行频谱分析，计算出瞬时电压值，最大值，最小值，峰-峰值，均值，交流有效值，直流偏置。

2.2.2.4 计数器

计数器前面板：

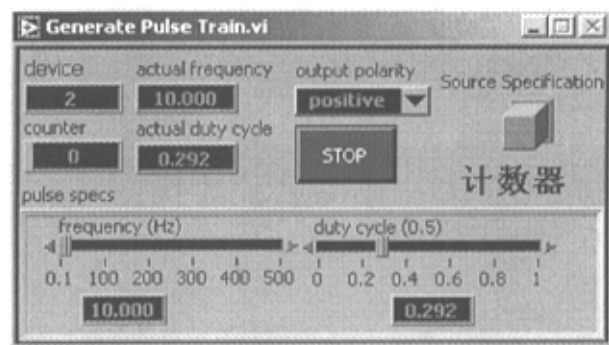


图 23 计数器前面板

NI 公司的 PCI-6024E 数据采集卡采用 DAQ-STC (System Timing Controller, 系统定时控制器) 芯片来实现与定时相关的功能，DAQ-STC 内部共有 7 个 24 位、3 个 16 位的计数器，能够产生 13 种定时信号，最大分辨率能达 50ns，其中包含两组通用定时信号，分别通过 GPCTR0_SOURCE Signal, GPCTR1_SOURCE 端送出，它们分别反应了通用计数器 0, 1 的实际时钟信号，可以软件配置。

虚拟计数器前面板如图 23 所示，它可以产生频率可调、占空比可调的方波信号。

计数器程序框图：

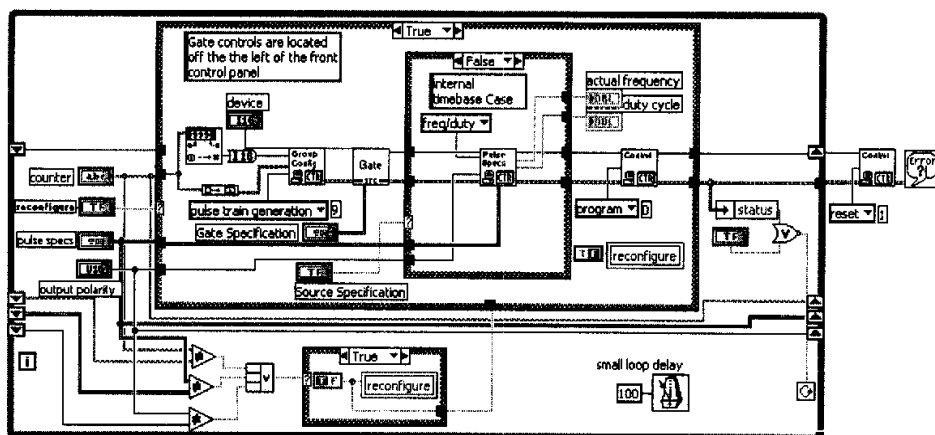


图 24 计数器程序框图

框图程序包括以下几个模块，调用 LabVIEW 函数库自带的子函数：

1. 分组函数：Group Config 函数对计数器进行分组。
2. 选通信号设置：设置选通信号源（Gate），设置其极性。
3. 脉宽配置：设置输出信号的频率、脉宽以及占空比。
4. 控制函数：控制计数器的启动和停止。

2.2.2.5 数字 I/O

数字 I/O 前面板：



图 25 数字 I/O 前面板

PCI-6024E 数据采集卡有 1 个 8 线的数字 I/O 通道，其中每一个引脚都可以作为数字信号的输入或者输出。当作为输出时，可以设置 0 为高电平或低电平。

数字 I/O 前面板如上图所示，按要求将低 4 位设置成输入，高 4 位设置成输出。

数字 I/O 程序框图:

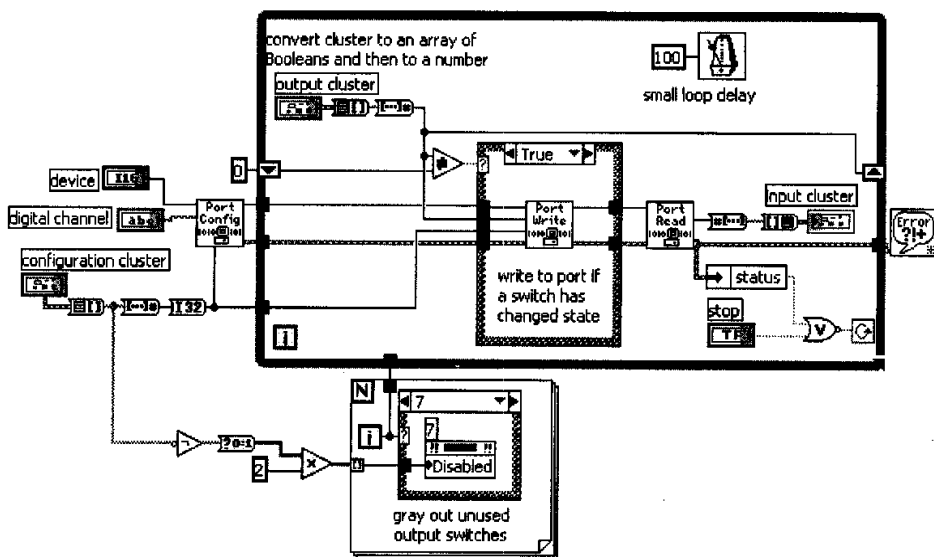


图 26 数字 I/O 程序框图

框图程序包括以下几个模块，调用 LabVIEW 函数库自带的子函数：

1. 端口配置模块: Port Config 函数配置端口为输入或输出。
2. 端口写函数: 向输出端口写“0”或“1”。
3. 端口读函数: 从指定的端口读信号。

2.3 实验管理系统

2.3.1 实验管理系统的功能

开放式实验室管理系统功能模块如图 27 所示:

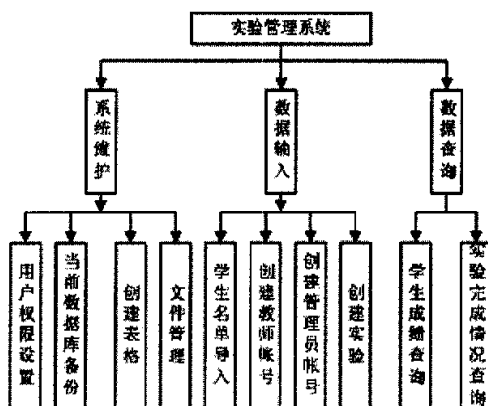


图 27 实验管理系统功能模块

1. 用户权限管理

实验管理系统用户分为四类：高级管理员、实验管理员，实验指导教师和学生。

(1) 高级管理员是最高一级的管理员，他保证系统的正常运行，负责数据库中相应表格的创建。他可以添加或删除学生账号、实验管理员账号、教师账号，同时还具有实验管理员的所有权限。

(2) 实验室管理人员按照实验指导教师的要求负责创建实验、删除实验，可以查看所有实验的完成情况。

(3) 实验指导教师仅能查看自己所授课程的实验完成情况，负责检查本班学生的实验完成情况，批阅实验报告并给出成绩。

(4) 学生用户根据自己所选课程完成相应的实验。学生仅能查看自己的实验完成情况和成绩。允许根据完成情况选择重做该实验。

2. 当前数据库备份

系统采用 MySQL 数据库。高级管理员对该系统的数据库进行备份，在系统崩溃时数据不丢失。

3. 创建表格

数据库中用于用户管理的表格属于基本表格，高级管理员在该系统投入使用之前应当将其创建好，以便对不同用户的身份进行验证。目前，系统中的基本表格有 5 个，administrator 表、student 表和 teacher 表，分别用于存储管理员信息、学生信息和教师信息，exp_list 表用于存储实验信息，score 表用于存储学生的实验成绩。他们的结构如下列表格所示：

administrator 表的结构 (a_id 为索引项)：

a_id	a_pw	a_name
qushizhu	6122916	李翼
linpin	linpin	林平

student 表的结构 (s_id 为索引项)：

s_class	s_name	s_id	s_pw	s_teacher
机械工程及其自动化	薛志刚	3010811119	3010811119	林平
机械工程及其自动化	刘晓洪	3020621069	3020621069	林平
机械工程及其自动化	许立辉	3020811011	3020811011	林平
机械工程及其自动化	刘翔	3020811020	3020811020	林平
机械工程及其自动化	姜红刚	3020811021	3020811021	林平
机械工程及其自动化	范翔	3020811025	3020811025	林平
机械工程及其自动化	谢佳斌	3020811043	3020811043	林平
机械工程及其自动化	王鑫非	3020811044	3020811044	林平
机械工程及其自动化	王良俊	3020811045	3020811045	林平

teacher 表的结构 (t_id 为索引项):

t_id	t_name	t_pw
wxh	王小菊	wxh
fxc	付晓程	fxc
fwm	裴伟献	fwm
zdh	张德华	zdh
lp	林平	lp

exp_list 表的结构 (exp_id 为索引项):

exp_id	exp_teacher	exp_name
005	林平	单管放大实验
003	王小菊	模拟电路实验
004	林平	数字电路实验
002	林平	负反馈实验
001	王小菊	单管放大实验
007	张德华	单管放大实验

score 表的结构:

s_id	exp_id	exp_score
3010811119	001	75
3010811119	002	80
3010811119	003	80
3010811119	004	80
3020621069	001	80
3020621069	002	70
3020621069	003	75
3020621069	004	80

4. 文件管理

学生可以提交任何格式的文件到服务器, 将学生提交的文件保存在 C:\Program Files\Apache Software Foundation\labadmin\data\expresult 目录中, 每个学号对应一个子目录, 子目录下保存该学生不同实验的实验信息。

5. 学生名单导入数据库 student 表中

由于学生用户较多, 单独创建学生账号来增加 student 表中的记录工作量很大。MySQL 提供了批量装载数据的方法, 指令格式如下: load data local infile "文件名" into table "表名";, 该语句读取位于客户机上当前目录中数据文件中的内容, 并将其发送到服务器装入相应的表格。数据文件要按照相应表格的格式来组织数据, 数据项之间用“Tab”键间隔, 记录之间用“回车符”间隔, 一般以“.txt”文件保存。但是, “.txt”文件中的“回车符”与 MySQL 中的“回车符”编码不一致, 装载数据之前必须对“回车符”进行转化。为此, 开发了一个名为“LabAdmin-loadData”的小程序, 可以直接将“.txt”中的数据导入库。

6. 账号管理

由于实验指导教师和管理员人数不多,教师账号、管理员账号可以由高级管理员从管理员页面动态的创建和删除,也可以从文件中批量转载到相应的表格中。

7. 实验管理

实验由管理员或高级管理员创建或删除,实验指导教师无权创建或删除实验。管理员可以从管理员页面动态的创建和删除,也可以从文件中批量转载到 exp_list 表格中。

8. 数据查询

学生通过自己的学号作为账号登陆成功后,可以查看教师给出的成绩。

实验指导教师根据给定的账号登陆成功后,可以检查自己学生的实验完成情况,批阅实验报告,并给出实验成绩。管理员可以查看所有实验的完成情况。

2.3.2 实验管理系统应用程序的安装

1. 数据库 MySQL, 包括 Server 和 Client

说明: 其中 Server 是必须的, Client 是为了方便直接输入数据到数据库。

默认安装目录: C:\mysql

Client(MySQL Control Center, 客户端, 解压后即得到安装程序): mysqlcc-0.9.4-win32.zip

默认安装目录: C:\Program Files\mysqlcc

a. 数据库全局管理用户名密码: mysql, 08180422

b. 实验管理系统用户名密码: labadmin, 08180422

c. 删除无用用户名密码

d. 设置成 NT service, 去掉 System Tray icon

e. 在 cmd 下本地连接数据库测试

2. JDK

源文件: (Java Developer's Kit, 安装程序) jdk-1_5_0-windows-i586.exe

默认安装目录: C:\Program Files\Java\jdk1.5.0

a. 设置环境变量:

JAVA_HOME=C:\Program Files\Java\jdk1.5.0

CLASSPATH=.;% JAVA_HOME %\lib\dt.jar;% JAVA_HOME %\lib\tools.jar

b. 在 DOS 环境下测试环境变量

3. JDBC connector

源文件：（由 mysql-connector-java-3.0.11-stable.zip 解压得到当中的 jar 文件）

mysql-connector-java-3.0.11-stable-bin.jar

复制到目录%JAVA_HOME%\jre\lib\ext（默认）：C:\Program Files\Java\jdk1.5.0\jre\lib\ext

4. 应用服务器 tomcat

源文件：（安装程序）jakarta-tomcat-5.0.25.exe

默认安装目录：C:\Program Files\Apache Software Foundation\Tomcat 5.0

a. 设置环境变量：

TOMCAT_HOME=C:\Program Files\Apache Software Foundation\Tomcat 5.0

b. 设置 tomcat 管理员用户名密码：admin，08180422

2.3.3 基于虚拟仪器的管理系统的优点

1.以 PC 机作为主平台的虚拟仪器，由于其软硬件维护十分方便，不同仪器采用不同传感器及相应的调理模块和相应的分析软件即可构成不同的测控仪器，缩短了开发周期、降低了开发成本。而且，它在数据处理和保存方面有传统仪器不可比拟的优越性，方便实现实验数据的管理。

2. LabVIEW 作为一种编程语言，支持通过 Internet，ActiveX，DDE 等交互式通信方式进行数据共享，实现代码重用。

3. 实验管理系统采用 B/S/D 结构，方便系统的扩展。为今后实现虚拟网络实验室打下基础。

4. 实验管理系统通过实时提交实验数据实现对实验过程的在线管理，方便教师批阅实验报告，大大节省了实验教师的工作量。

2.4 实验管理系统的使用

如前面所述，根据用户权限的不同，将实验管理系统的用户分为三级，不同的用户从登录页面成功进入管理系统后，将进入不同的页面。学生用户根据所选课程，由相应的实验教师安排需要完成的实验，每个学生需要完成的实验不尽相同。对于某一具体实验，学生通过虚拟仪器将所测得的波形数据先以文件的形式保存在本地机器中，实验完成时再将此波形数据上传到服务器，同时在服务器端记录学生提交波形的时间。具体测量的实验参数在每个实验页面以表单的形式提交到服务器。教师用户可以查看自己所授课程的实验完成情况，负责

本班学生的实验检查，给出实验成绩。管理员用户按照实验教师的要求负责创建实验、删除实验，同时他可以查看所有实验的完成情况。其中，高级管理员还需要进行系统的维护，保证系统的正常运行，负责数据库中相应表格的创建。

第三章 实验管理系统的发展

3.1 实验管理系统可扩展的空间

目前已实现的实验管理系统仅仅是基本框架,功能有限。在此基础上可以无限扩展,不断增加其功能,如增加实验预习模块、实验考试模块等。

虚拟网络实验室的建设与管理是实验管理系统发展的一个方向。当今时代,自动测试仪器的发展方向是:智能化、标准化和网络化。将网络技术与虚拟仪器相结合,构建网络化虚拟仪器,使虚拟仪器的功能大大扩展。通过虚拟网络实验室能够实现跨时空跨学科的设备远程共享,甚至远程控制,很好地满足了科研教学对分布式实验系统的要求,国外的很多科研院所已经在这方面进行了有益的尝试。

3.2 虚拟实验室的研究现状

虚拟实验室及虚拟实验技术是近年来信息技术迅速发展的产物。在 90 年代初,提出了虚拟实验室的最初概念^[12],它的定义与基本内涵主要包括以下两个基本的方面:(1)一个无墙的中心。研究人员能在其中从事科学研究和工程设计,不必顾及地理位置的限制,实现同行间、同事间的互动;共享仪器、设备、数据、计算资源以及数字图书馆的信息。该定义强调三个方面的问题:一是现代实验室应当是一个无墙的中心,这个中心可以有逻辑上的限制,但没有物理空间的限制,其基础是 Internet;二是协同工作环境,即计算机支持的协同工作环境;三是充分地实现资源共享。(2)科学研究与工程设计的新方法——基于计算的方法,这种方法具体表现为虚拟实验、虚拟设计和虚拟制造。

虚拟实验室的研究意义重大,将成为解决我国实验资源紧缺和远程教育实验难等问题的重要途径。目前,大多数的虚拟实验室按照如图 28 所示的架构,客户端通常基于 Web,通过 Web 服务器下载客户端软件,然后客户端程序与服务器端程序建立通讯,进行实验。整个系统由客户端,服务器端和物理设备组成。

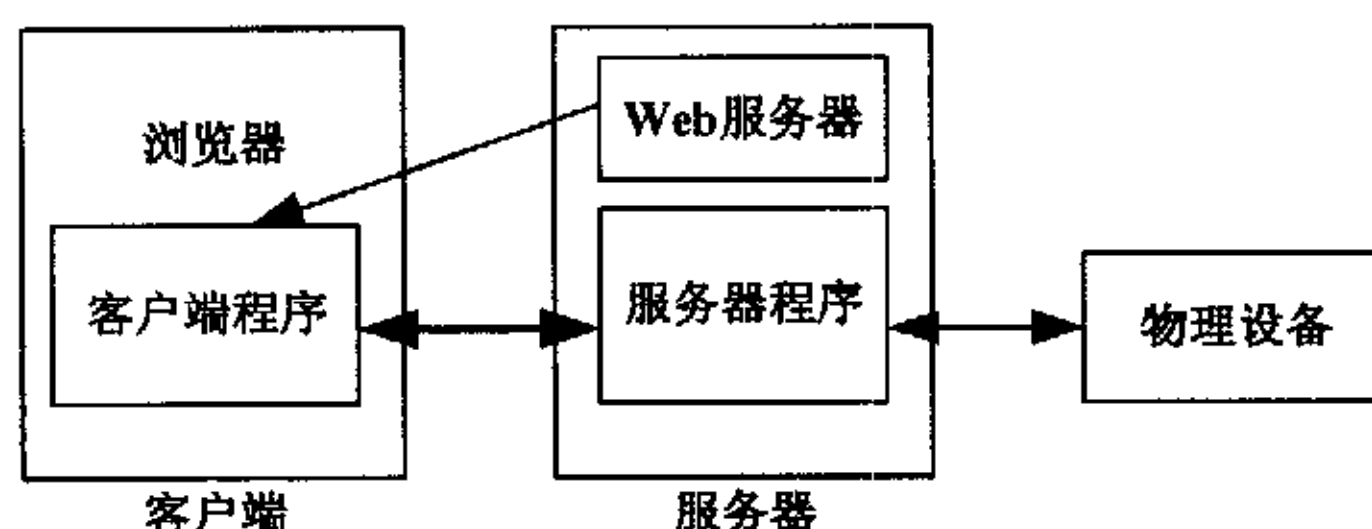


图 28 远程虚拟实验室典型结构图

● 按照客户端软件划分

按照客户端软件可以大致分为 3 种类型的虚拟实验室方案，（1）利用简单的 Http 页面或者使用非浏览器中运行的客户端软件。（2）利用组件技术。（3）利用 Java Applet 技术及 HTML 网页编程技术搭建基于 Web 的虚拟实验系统^[15]。

方案二是基于组件开发的虚拟实验室，可以提供良好的代码重用性和平台无关性，适合于开发复杂的系统，基于 CORBA 组件技术进行开发^{[14][22]}或基于 LabVIEW 提供的 ActiveX 控件直接生成客户端界面^[21]。

由于 Java 的平台无关性和良好的网络编程支持，方案三成为目前各个虚拟实验室的首选方案^{[18]-[20]}。Java Applet 技术能嵌入到网络浏览器中随网页一起下载到客户端，并在浏览器中运行。而且，此种方案系统版本的升级及维护是在 Web 服务器端进行的，用户需要访问时才动态下载，这就保证了用户每次使用的都是最新版本^[15]。如利用 Java Applet 实现电子技术实验箱面板，实验过程中根据用户鼠标点击对面板中的图形进行擦除和重绘，实验电路绘制结束，向服务器提交请求^[18]。

网络实验室客户端的实现还有多种其它选择，如使用各种 Web 技术创建 HTML 页面，如 ASP, JSP, CGI 等等，也可以不通过浏览器，而是直接使用 TCP/IP 技术与服务器相连或直接使用 UDP 协议与服务器相连^[12]。

● 按照服务器端软件划分

按照服务器端软件分类，虚拟实验室方案也可大致分为 3 类，（1）基于 LabVIEW 构造的虚拟实验系统；（2）基于 MATLAB 的 Simulink, RTW 等工具构造虚拟实验室系统；（3）自行编程实现。

LabVIEW 作为虚拟仪器的开发平台，提供了便利的设备访问能力和强大的实时控制能力，使得仅仅使用 LabVIEW 就可以构建一个完整的虚拟实验室^[23]，利用其自带的组件开发的虚拟仪器不仅功能强大，而且外观与真实仪器仪表高度相同，非常适合作为训练学生掌握仪器仪表的辅助工具，因而成为虚拟实验室中设备访问的首选标准。如基于虚拟仪器的远程

NPN 晶体管性能测试实验，使用 GPIB 总线连通具体的仪器和 GPIB 适配器，通过 SCPI 应用程序接口编程，控制实际的仪器进行实验测试^[19]。

Matlab 由于强大的仿真和计算能力，一直是自动控制领域最为重要的工具，自然也被广泛地用于虚拟实验室的开发，使用了 Matlab 作为服务器控制实际对象的引擎，尤其是对于控制类实验的开发。如基于 Matlab 的一整套的网络控制系统解决方案^[26]，包括设备访问，实时控制，具体界面和帮助系统等，经济性也被考虑其中。

● 按照基本功能划分

(1) 软件共享虚拟实验室

其特点为，服务端共享本地的虚拟实验室模拟软件平台，接受客户端发送的实验请求，分析和处理实验参数，经过计算模拟最终将结果返回客户端。整个系统不涉及具体的实验仪器硬件设备，只是利用软件模拟实验的过程。目前，国内外高等院校或者科研机构已建成的虚拟实验室基本上都是仅基于仿真对象，大多处于这种类型。

(2) 仪器共享虚拟实验室

服务端同样接受客户端的实验请求和实验参数，使用实验参数配置与之连接的实验仪器硬件设备，由实验仪器硬件设备进行实验，并将实验结果返回服务端，最后返回到用户端，实现实验仪器和实验数据的共享。

(3) 远程控制虚拟实验室

与仪器共享虚拟网络实验室最大的区别在于，除了实验仪器实验数据的共享之外，其还要实现客户端对实验仪器设备的远程控制。

3.3 虚拟网络实验室的实现

3.3.1 LabVIEW 中的网络技术

充分利用目前较为成熟完善的、基于全球通用 TCP/IP 网络通信协议的 Intranet/Internet 网络，建立统一的总线或网络，无论从成本、管理和维护，都是一个很好的选择。LabVIEW 以 3 种网络模型为参考，通过内建功能强大的 TCP/IP (transmission control protocol/internet protocol, 传输控制协议/互联网络协议) 及 UDP (user datagram protocol, 用户数据包协议) 协议技术、DataSocket 技术、VI Server 和 Web 服务器技术，较好地实现了虚拟仪器的网络化扩展。

3.3.1.1 TCP/IP 或 UDP 协议通信

TCP/IP 或 UDP 协议是目前全球通用的网络协议。IP 是网络层协议，实现的是不可靠无连接的数据包服务。TCP 和 UDP 都是建立在 IP 基础上的传输层协议。TCP 是基于可靠连接的协议；UDP 是基于不可靠无连接的协议。如果希望得到可靠的数据传输，那么使用 TCP 是较好的选择。

利用 LabVIEW 的 TCP 实现网络通信，只是对 TCP/IP 协议的简单包装，虽然原理比较简单，但是，由于涉及到底层 TCP 编程，相对比较复杂。其一般通信方法如下：

1) 在服务器端，利用 TCP Listen 监听指定端口的 TCP 网络连接，连接建立后，利用 TCP Write/Read 写或读数据，最后用 TCP Close Connection 关闭连接。

2) 在客户端，利用 TCP Open Connection 打开 TCP 网络连接，连接建立后，利用 TCP Read/Write 读或写数据，最后用 TCP Close Connection 关闭连接。

3.3.1.2 DataSocket (DS) 技术

DataSocket (DS) 是 NI 公司把 TCP/UDP 协议封装成适用于测控的网络技术^[1]，利用这一技术可以实现跨平台、多种数据传输方式，其优点是不用进行复杂的底层 TCP 编程，就可以同时通过计算机网络向多个终端发布现场测量数据。它包括两个组件：DataSocket Server 和 DataSocket Server Manager。

1) DataSocket Server (DS 服务器) 是一个独立运行的小程序，利用 DS 技术发布数据的机器上必须打开它。

2) Server Manager (DS 服务器管理器) 主要功能是设置 DS 服务器可连接的客户数和用户权限等，用于组成智能的测控网。

DS 传递数据有 3 部分组成：即发布数据程序、DS 服务器和接收数据的程序，通常将发布数据程序和 DS 服务器运行在同一机器上，在其他机器上运行接收数据的程序。发布数据的程序通过 DS 服务器进行数据输出，输入数据的程序通过 DS 服务器接收数据。其关系如图 29 所示。

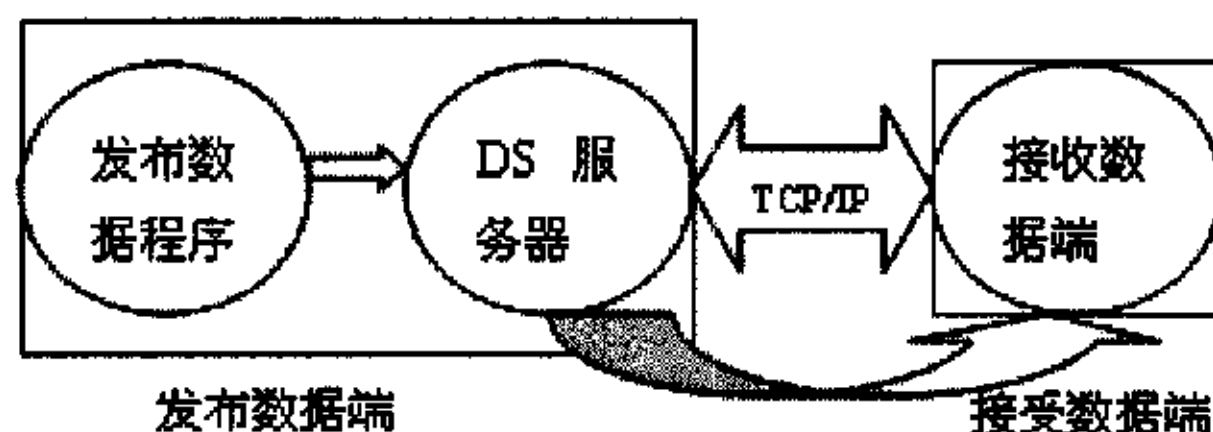


图 29 数据的发布与接收

DataSocket API 提供一个可用于多种编程语言和数据传输格式通信的单一接口，其中发布数据程序用 DataSocket Write 函数自动将用户数据转化为网上传输的字节流，接收数据程

序用 DataSocket Read 函数将字节流还原到用户数据格式。

DS 函数与 DS 服务器的通信采用的是本地协议 DSTP (DataSocket Transfer Protocol)。该协议通过用户熟悉的 URL 标志确定数据源的位置和目标位置,就如同网络浏览器一样方便。

更为简单的方法是利用 DataSocket 技术中的 DataSocket Connection...功能,将两个在不同机器上运行的 Front Panel (前面板)对象连接起来,不需要任何编程的数据传输方式,就可以实现两个连接对象之间的数据发布和接收,方便地实现程序的远程控制和数据的远程采集,这也为远程测控提供了方便。

3.3.1.3 VI Server

LabVIEW VIs 程控是通过 VI Server 技术实现的。利用 VI Server 技术,在 LabVIEW 的 Option 对话框中设置 VI Server 的访问权限,用户就可以编程动态控制 LabVIEW VIs,通过一个调用引用节点(Call By Reference Nodes)方便地运行远程的不同平台上的 LabVIEW VI,就像本地 VI 一样。功能如下:

- 本地动态调用 VIs,动态地将 VIs 加载到内存中。
- 将一台计算机中的 LabVIEW 配置成为一个可以到处 VIs 的 VI Server,在另一台计算机上通过网络远程动态调用这些 VIs。
- 动态更改 VI 属性和 LabVIEW 属性。
- 通过为应用程序创建一个内插结构来增加功能。例如,应用程序可能具有一系列滤波器 VIs,这些滤波器 VIs 都使用相同的参数,可以设计应用程序从一个 Plug-in 目录中动态加载这些 VIs。

3.3.1.4 Web 服务器

Web 服务器是虚拟仪器由 LAN 向 WAN 扩展的,实现其网络化扩展,极具价值的一环。使用 LabVIEW 的 Web 服务器,可以方便简单地实现在 Web 上发布 LabVIEW 程序;同时,网络上的计算机无需安装 LabVIEW,就可以查看以 HTML 文件发布的 Web 网页,甚至通过网页控制程序的运行。使虚拟仪器 Web 化得以实现,借此建立起强大的网络化功能(具体方法在后面章节介绍)。

综合应用 LabVIEW 的网络新技术,设计虚拟网络化方案。

3.3.2 两层 C/S 结构

LabVIEW 作为虚拟仪器平台,既可以开发基于采集卡的真实虚拟仪器,也可以开发一

些纯软件的虚拟仪器，用于仿真实验。LabVIEW 的最新版本 LabVIEW 7.1，包括控制与仿真、数字信号处理、统计过程控制、模糊控制、PID 控制等众多软件包，可以运行于 Windows，Linux，Macintosh，Sun，HPUX 等多种平台。利用 LabVIEW 可以开发不同的仿真实验模块，应用在不同的学科上以辅助教学之用。将这些仿真实验模块推向网络，学生可以随时在宿舍或家里远程进行实验仿真。

在 LabVIEW 中，有两种方法可以远程操作前面板，给网络实验室的实现提供了一些思路。

1. 在 LabVIEW 环境中直接操作 Remote Panels (C/S 结构);
2. 在网页浏览器中操作 Remote Panels (B/S 结构)。

无论采用哪种方法，服务器端都要进行 LabVIEW Web Server 的配置，包含三个方面：

1. 文件路径和网络设置;
2. 客户机访问权限设置;
3. VIs 访问权限设置。

LabVIEW Web Server 默认的端口号为 80，通常情况下，端口号 49152 到 65535 是推荐给用户自定义 TCP/IP 应用程序使用的网络端口。两种方法的区别在于，第一种方法实现远程前面板操作的前提是 Client 端已经安装了 LabVIEW，在 Client 端的 LabVIEW 菜单栏中选择 Operate>>Connect to Remote Panel...，在弹出的对话框中输入 Server IP Address、VI Name 和 Port，然后进行连接。第二种方法是通过网页浏览器进行访问，故使用更加方便，在 LabVIEW 菜单栏中选择 Tools>>Web Publishing Tools...，在弹出的对话框中进行前面板发布。但是，这两种方法的缺点是，多个不同的 Client 端同时连接，给用户的操作带来一定的限制。如果向 NI 公司购买授权，便可以使更多的 Client 端同时连接到 Web Server。这种方案特别适合仿真实验的开发，但它不能对实验数据和用户信息进行管理，开发过程相对简单。

3.3.3 三层 C/S 结构

三层 C/S 结构采用 B/S/C+数据库结构，对用户信息和实验数据进行有效的管理。实际上是 B/S/C/D 结构，如图 30 所示。具体有两种实现方案：

方案一：Browser & NI Web Server 结构，服务器为 Web 服务器，其整体架构如图 30 所示。

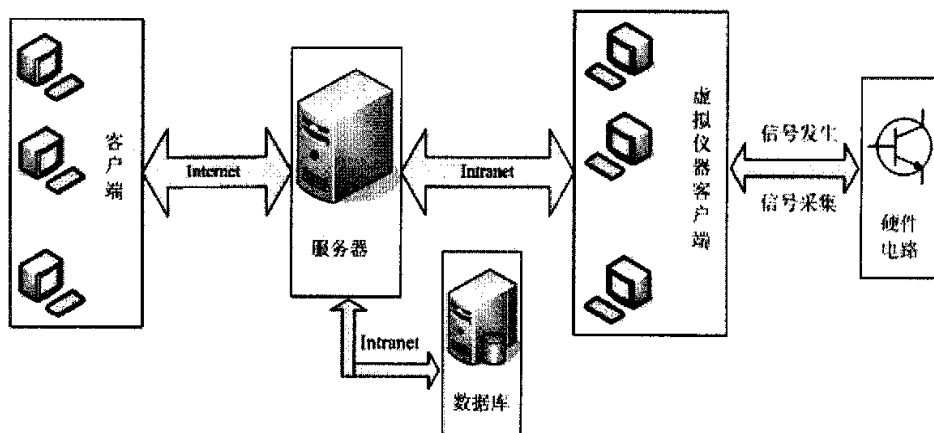


图 30 虚拟网络实验室整体架构（方案一）

以 LabVIEW 内置的网页发布功能为基础，通过 HTML 设计网页，并使用 Web 服务器的网页发布功能，直接在服务器端生成嵌入实验平台的 WWW 网页，用户只需使用网页浏览器进入站点，即可通过 Internet 进行实验。LabVIEW 内置的 Remote Panel Connection Manager 用来对用户的使用情况进行监控与调度。此 Web 服务器可以是 ASP 服务器、JSP 服务器或者 PHP 服务器等等，但必须能够与数据库进行通信，方便保存客户信息和数据信息。客户端用户登陆成功后进入实验，选择相应的实验，了解实验原理，操作虚拟仪器的前面板，共享实验数据。

以单管放大电路实验.vi 作为一实验模块为示例，简述这种方案可操作性：

1. 进入登陆页面，对用户的身份进行验证，确定其权限；

普通用户首先要注册，如图 31 所示，在注册页面将需要完成的实验加入个人实验收藏夹，在登陆成功并进入实验后，实验模块目录栏将显示该用户所需要完成的实验，如图 32 所示：



图 31 虚拟网络实验室登陆页面

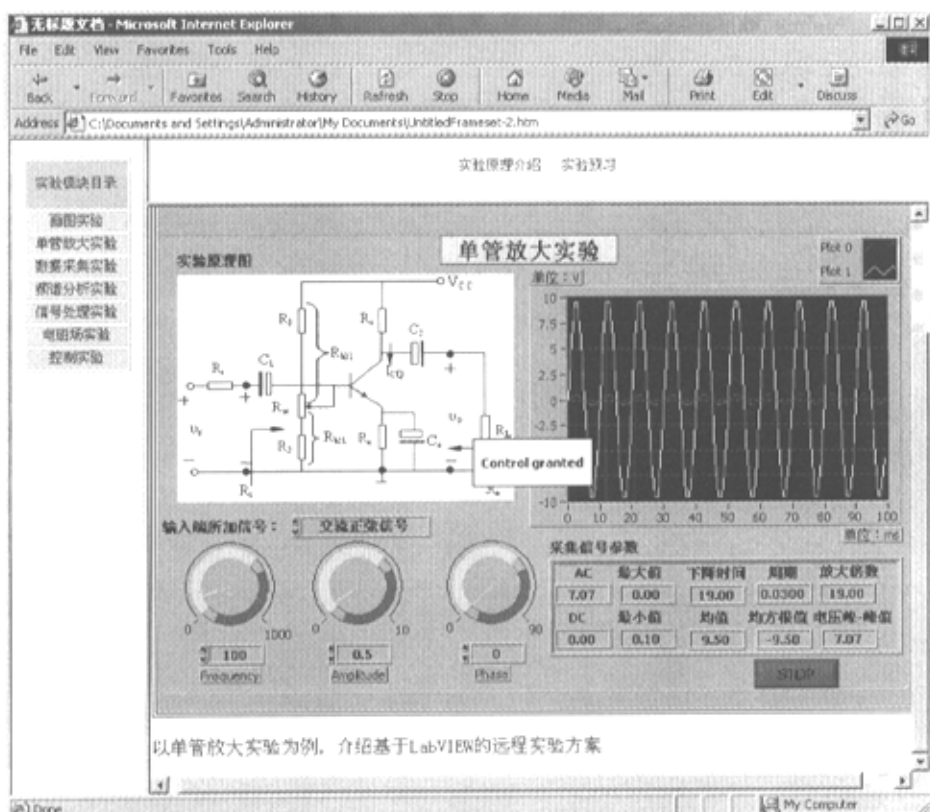


图 32 虚拟网络实验室界面

虚拟网络实验室界面用框架实现，右下方是虚拟仪器客户端 VI Server 发布的虚拟仪器前面板，用户可以获得控制权对前面板进行控制。右上方是该实验的实验原理介绍，实验电路图。虚拟仪器 VI 位于虚拟仪器客户端，当实验模块较多时，可能需要多个虚拟仪器客户端进行网页发布，以减少服务器的工作压力。服务器只负责与客户端进行交互，具体的实验在虚拟仪器客户端进行。每一个独立的虚拟仪器实验是一个独立的 VI，以单管放大电路实验为例，简述客户端 VI 的开发过程：

单管放大电路实验前面板设计:

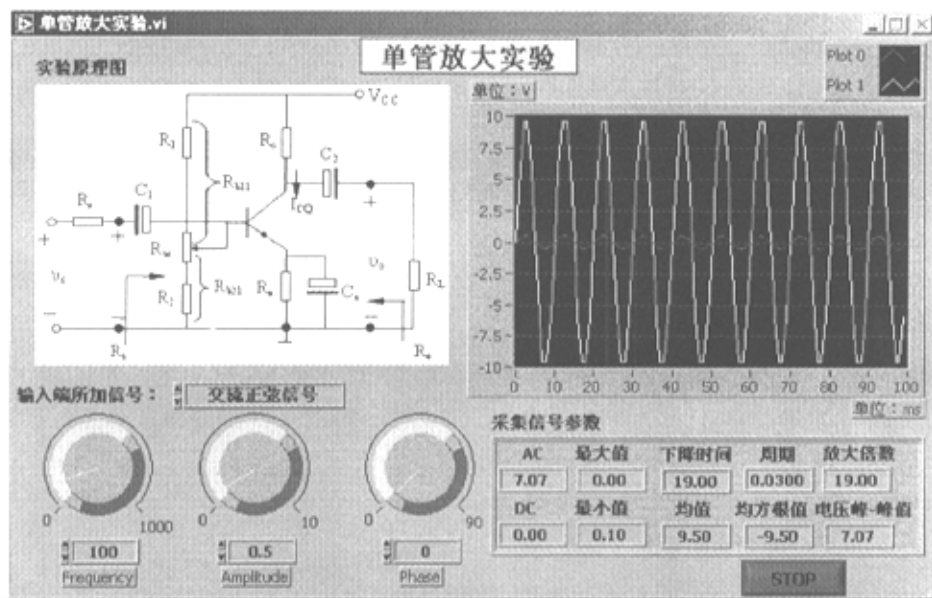


图 33 单管放大电路实验前面板

单管放大电路实验程序框图:

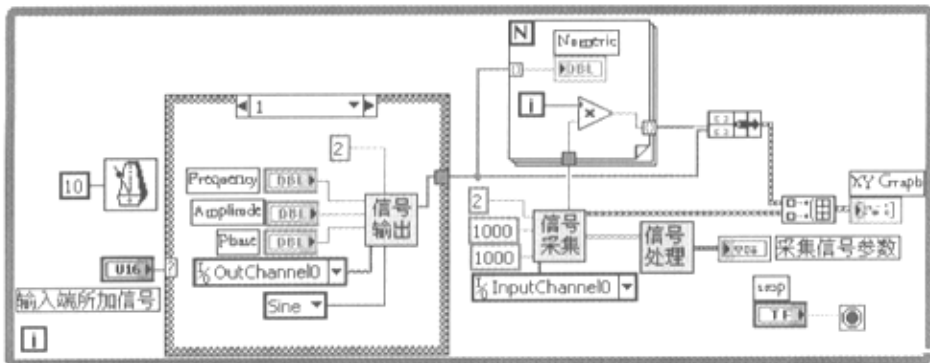


图 34 单管放大实验程序框图

其中,子函数可以是本地调用、网络调用,或者通过 DLL 调用。LabVIEW 中 DLL (动态链接库)调用是通过 CLF(Call Library Function)节点实现的,可以使用此节点实现 windows API 调用,在此不详细介绍其调用过程。

网络调用过程稍稍复杂一些,首先用 Open Application Reference 打开远程应用程序参考,然后用 Open VI Reference 打开远程子 VI(必须保证远程的 LabVIEW 作为 TCP/IP server 在内存中运行);最后通过 Call By Reference Node 动态地调用子 VI。这种方式的子程序调用充分体现了模块化设计的“代码重用”的优势,方便地实现了本地到远程的扩展。

从上面介绍的情况看,三层 C/S 结构较两层 C/S 结构有明显的优势。但是,基于 VI 发布单用户连接的缺点还是存在,而且这种方案仅适合软件共享虚拟网络实验和仪器共享虚拟网络实验。

方案二:客户端应用程序是用 Java 编写的 Applet,采用目前虚拟实验室最为通用的 Web 发布方式。服务器采用 Web+应用服务器(Java 编写),控制端 LabVIEW 编写。整体功能模块架构如图 35 所示,展示了网络实验室各部分功能模块的运行机理。其中 1 代表浏览器与 Web 服务器之间采用 HTTP 协议的通讯,2 代表 Web 服务器下载 Applet 到浏览器,3 和 4 分别代表客户端应用程序与服务器端应用程序、服务器端应用程序与控制器端应用程序采用基于 TCP/IP 协议的自定义协议的通讯。

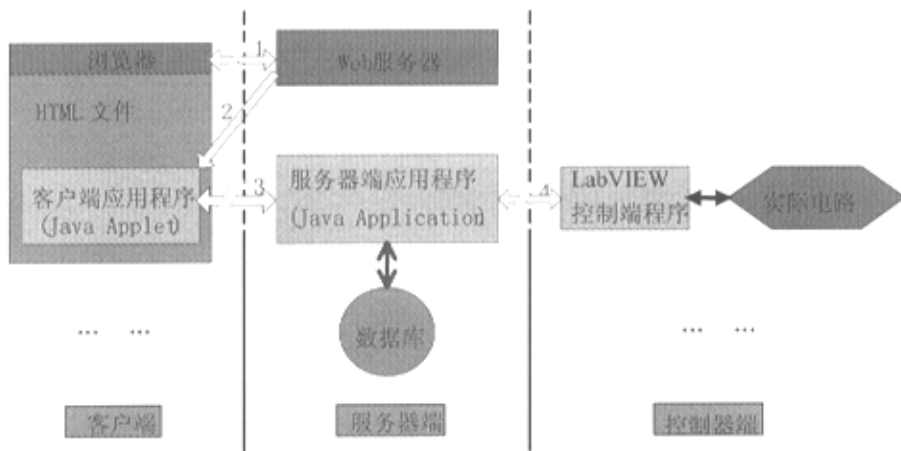


图 35 虚拟网络实验室整体架构（方案二）

方案二避免了此类实验仅仅是简单的校验性实验。如对于电路实验,若在客户端 Applet 中提供电路图组态功能,这样用户就可以自行绘制电路图,增强了系统的灵活性。实验时,用户利用工具条或菜单上的器件,自行绘制电路图,利用客户端程序中的电路图校验工具对绘制电路进行校验,当电路图正确且在实际电路组态范围之内时,电路检测成功,用户可以

进行实验。否则,要求用户修改电路图。正确的电路图经服务器发送给控制器,然后控制器按照电路图控制继电器,模拟开关等元件构造实际电路,进行实验。

方案二回避了 VI 网页发布,利用 LabVIEW 的数据采集和数据传输能力将采集到的信号转化为适合于在以太网上传输的数据信号,从而实现在应用程序间共享现场采集的数据。服务器端应用程序与控制器端应用程序通信,获取采集的信号。虚拟网络实验室处于一个多用户,多实验的环境中,如果使用简单的 C/S 结构,用户直接同控制端通讯,简单的结构带来最高的效率,但是由此的弊端也是显而易见的,所有的控制端必须重复实现所有的用户管理,登记管理模块,用户通常无法获得所有实验的状态,缺乏管理性。而这种三层的双 C/S 的结构很好地避免了该问题,用户和实验可以被统一管理,所有的管理记录统一在服务器上完成,服务器处于客户端和控制端之间,除了转发客户端与控制端之间的通讯,也将控制端和客户端相隔离。此时,服务器端也相当于系统的一道防火墙,大大提高了系统入侵的难度,为系统提供了更好的安全性和可靠性。

3.3.4 方案比较

两层 C/S 结构充分利用 LabVIEW 的网络技术,较容易实现,功能相对有限。三层 C/S 结构的第一种方案较两层 C/S 结构有明显的优势,服务器控制所有的连接,负责与数据库通信,用户信息和数据信息得到有效地管理。但是,这两种方案都没有克服多用户同时进行实验的问题。其根本原因在于 LabVIEW 7.0 Express 完全版默认情况下只允许一个用户连接。如果需要多用户连接,需要向 NI 公司购买授权。LabVIEW 虽然是一个强大的软件设计工具,拥有很多的专业设计功能(如:应用程序生成器、数据库连接工具、Microsoft Office 报告生成工具、Internet 工具等),但这些工具都以附加工具包的形式向用户发布,且价格都比较昂贵,从而提高了使用者的设计成本。所以在选择方案时,服务器端应用程序不采用 LabVIEW 编写,避免 LabVIEW 与目前通用的数据库(如 MySQL, ACCESS 等)通信不方便。LabVIEW 本身并不具备数据库访问功能,NI 公司也积极地采取很多措施来解决这一问题,但是其 LabVIEW SQLToolkit 工具包也因为价格昂贵而不能选用。在不使用 LabVIEW 附加数据库连接工具包的情况下,可以利用 ActiveX 技术实现 LabVIEW 对数据库的访问^[27]。如果以其内置的 TCP/IP 模块为基础,应用服务器采用 LabVIEW 编写,客户端可以通过一个 API 用户终端与服务器通信^[28]。Application Server 用来采集传输实验数据、管理用户、记录用户使用情况;而 API 用户终端则提供操作者 GUI 界面,实现数据的获取、分析运算以及显示存储等。

三层 C/S 结构第二种方案中的应用服务器用 Java 编写, Java 语言具有面向对象, 面向网络, 可移植, 与平台无关, 多线程, 安全等特点, 目前在网络环境中得到了广泛的应用。而且 Sun 公司和很多数据库开发商合作, 开发了一系列数据库连接桥, 使得 Java 程序与数据库通信十分方便。但是这种方案较前两种方案, 涉及的技术面较广, 开发难度大, 开发周期长。下面详细介绍一下这种方案的实现过程和需要注意的问题。

3.4 远程控制虚拟网络实验室

3.4.1 常规网络通讯技术

在实际网络编程中, 应用程序员需要面对的是各种网络 API 或将 API 经过再次包装的各种规范。最常用的方式大致可以分为 3 大类进行讨论。他们分别是:

- 基于 SOCKET (套接字) 方式;
- 基于 Web 的通讯方式;
- 基于分布式组件的通讯技术。

3.4.1.1 基于 Socket 方式

20 世纪 80 年代初, 美国政府的高级研究工程机构 (ARPA) 向加利福尼亚大学 Berkeley 分校提供资金, 让他们在 UNIX 操作系统下实现 TCP/IP 协议。在这个项目中, 研究人员为 TCP/IP 网络通信开发了一个应用程序接口, 这个 API 称为 SOCKET 接口。今天该接口成为 TCP/IP 网络最为通用的 API, 也是 INTERNET 上进行应用开发最为通用的 API。20 世纪 90 年代初, Microsoft 联合其他几家公司共同制定了一套 Windows 下的网络编程接口, 即 Windows SOCKETS 规范 (也称为 WINSOCK)。WINSOCK 不仅包含了 BSD SOCKET 的大部分功能, 而且含有 Microsoft 特有的并且继续发展的增强功能。Windows SOCKETS 规范从 1991 年的 1.0 版发展到现在的 3.0 版, 经过不断完善并在 Intel, Microsoft, Sun, SGI 等公司的全力支持下, 已成为 WINDOWS 网络编程的事实上的标准。目前实际应用中 Windows SOCKETS 规范主要有 1.1 和 2.0 版。两者之重要区别是: 1.1 版仅支持 TCP/IP 协议, 而 2.0 版可以支持多协议, 以及 2.0 版本有良好的向后兼容性。

Socket 是一个通信终结点, Socket 接口在计算机中提供了一个通信端口, 可以通过这个端口与任何一个具有 Socket 接口的计算机通信。应用程序在网络上的传输和接收信息都通过这个 Socket 接口来实现。在应用开发中, 就像使用文件句柄一样, 可以对 Socket 句柄进行读写操作。

整个流程如图 36 所示，首先服务器在 Socket 上执行倾听(listen)操作，当收到为套接字支持的连接时，执行接收操作，连接建立后，accept 函数返回一个新的套接字，它表示连接的服务器端，服务器和客户端都通过函数 recv 和 send 执行接收和发送动作。

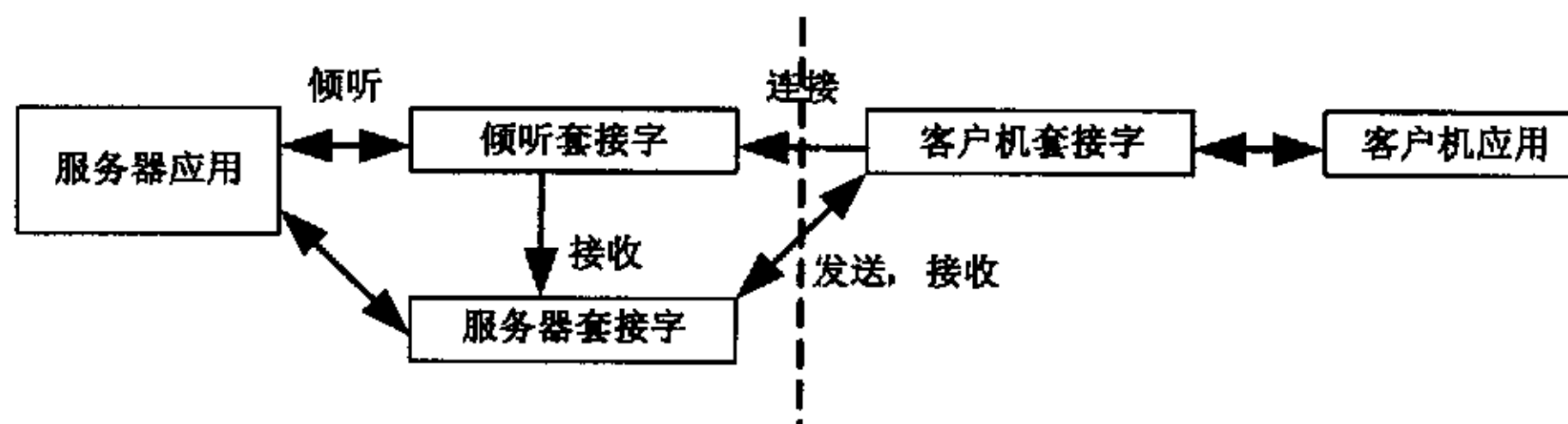


图 36 Socket 通讯示意图

从编程的角度来看，WINSOCK API 突出的特点是它继承了 Windows 消息，可通过标准 Windows 消息函数或执行回调函数来接收操作完成的通知。这一功能简化了 Windows 应用程序的设计。

3.4.1.2 基于 Web 的通讯方式

近年来，Inetnet 以一种前所未有的力量改变着人们的生活。其中最主要的方式就是通过 Web。Web 可以让多客户通过 Internet 或企业内部 Intranet 轻易地访问数据和资源。至今人们已经设计了多种基于 Web 的应用程序开发方案。以下仅介绍最为典型的三种：

- CGI 方式
- IIS 提供的解决方式
- Java 提供的解决方式

CGI 方式是最为传统的解决方案。至今已被几乎所有的 Web 服务器支持。首先输入数据通过 Html 页面得到，提交的数据搭配一个 URL，URL 告诉服务器应该进行哪一类服务，服务器执行的 CGI（Common Gateway Interface）程序位置，并在该程序执行时提供必要的数。CGI 程序通常以 Perl，Python，C/C++等语言完成，只要该语言能够自标准输入设备读取数据并将数据写至标准输出即可。CGI 程序以适当方式处理数据，产生出合适的 HTML 页面，使得 WEB 服务器返回给使用者。

但是，CGI 的执行效率十分低下，正在逐渐地被其他新技术所取代。Mircrosoft 的 IIS（Internet 信息服务器）及其配套的 ISA（Ineternet 应用程序）是当今 WINDOWS 平台上最为高效的解决方案。ISA 是利用 ISAPI 编写的，可以被 HTTP 服务器加载和调用的 DLL。流程与 CGI 方式类似，由服务器调用 ISA，同时提供输入数据，ISA 生成数据，通常是 HTML 页面，由服务器返回。与 CGI 方式不同，服务器与 ISA 的数据交互不是通过标准输入输出

设备，而是直接通过服务器入口点函数。一个标准的 ISA 共包含三个入口点函数：

GetExtensionVersion: 服务器初始化 ISA 时调用该函数，获得 ISA 的基本信息。

TerminateExtension: 服务器结束 ISA 时调用该函数。

HttpExtensionProc: 当服务器收到对 ISA 的请求时，调用该函数，该函数是整个 ISA 的主体。

ISA 提供了 CGI 方式的所有功能，而且由于 ISA 基于 DLL，与 Web 服务器运行于同一进程中，大大降低了通讯的开销，简化了编程。而且 IIS 还提供了 ISAPI 过滤程序，可以实现一系列 CGI 不能实现的功能。

Servlet 和 Applet 提供了 Java 平台下的 Web 应用程序解决方案。客户端使用 Applet 对数据进行检验并送出数据，服务器端以 servlet 接收并处理数据。Servlet 对服务器端程序开发提供了相当出色的解决方案，提供了得以取代 CGI 开发模式的架构，并减少了许多 CGI 的棘手问题。一个最基本的 servlet 从接口 Servlet 派生，存在四个基本函数：

Init: servlet 初始化

getServletConfig: 得到 servlet 初始参数

service: 具体请求处理函数

getServletInfo: 返回 servlet 相关信息，如作者，版本等

destroy: servlet 销毁

servlet 由于完全面向对象编程，所以结构相对清晰，而且由于 Java 的使用，获得了系统的平台无关性^[35]。

3.4.1.3 基于分布式组件的通讯方式

组件技术将单个的应用程序分隔成多个独立的模块，开发者可以随着技术的发展用新的组件取代已有的组件，使系统不断趋于完善。而且利用已有的组件，用户可以快速地建立全新的应用。利用分布式组件，开发者可以按照开发非分布式应用程序那样开发分布式应用程序。组件系统透明地实现了底层的通讯，通常采用 RPC 做为底层通讯的协议。

目前最为常用的组件技术规范有 CORBA (Common Object Request Broker Architecture, 公用对象请求代理[调度]程序体系结构) 和 COM (The Component Object Model, 组件对象模型) 两类。

CORBA 并非某种语言特性，它是一个集成技术，一种规格。CORBA 是 OMG 定义的一套和语言无关的分布式“对象交换标准框架”(standard framework of object interoperability)。CORBA 提供了由各种语言实现的对象之间的远端程序调用(RPC)能力，

以及一种与实际地址无关的原有系统沟通能力。OMG 所开发的对象可交换性规范，通常也被称为“Object Management Architecture”(OMA 对象管理架构)。OMA 定义了两组组成 Core Object Model (核心对象模型) 和 OMA Reference Architecture (OMA 参考架构)。核心对象模型陈述的是对象、接口及操作相关的概念。OMA 参考架构定义底层服务基础建设，以及允许对象相互交换的机制，包括 ORB (Object Request Broker) Object Services, 以及一些常见机制。ORB 是通信进行的管道，对象可以通过它来请求提供的服务，无需考虑对象位置，但是没有任何规格对这些对象规范出 ORBCore 应该如何实现。为了 ORB 之间相容性，OMG 定义了一组可通过标准接口加以访问的服务。IDL 是 CORBA 接口定义语言。名称服务是 CORBA 的基本服务之一。名称服务负责执行“字符串映射至对象”和“对象映射之字符串”的转换和映射动作。

CORBA 实际使用时，先建立 IDL 文件，利用工具生成对应的 stub (客户端存根) 和 skeletons (服务器代理)。服务器 (即组件的提供方) 先初始化 ORB 环境，调用名称服务，绑定对象。客户端 (组件的应用者) 也是先初始化 ORB 环境，调用名称服务，获得对象引用，随后就可以象使用本地对象般的调用。整个过程中应用程序开发者无需关心组件的位置 [36]。

COM 技术由 Microsoft 提出，最先在 Windows 平台上应用。COM 是一个说明如何建立可动态交替更新组件的规范。COM 允许对象向其他组件和宿主应用程序公开其功能。它既定义对象如何公开自身，又定义此公开如何跨越多个进程和网络进行工作。COM 还定义对象的生命周期。

在 COM 中，接口是对象用来公开其功能的机制，接口通过 MIDL 语言编写的 IDL 文件进行描述，对象可以公开它所选择的任意多个接口。每个接口都基于基本的 COM 接口 IUnknown。IUnknown 是作为所有其他接口之基础的基本接口，它定义了三种方法：QueryInterface、AddRef 和 Release。QueryInterface 允许接口用户向对象请求指向其另一个接口的指针。AddRef 和 Release 实现该接口上的引用计数。引用计数是用来决定对象 (严格地说是接口) 何时不再被使用，并因此可以自由地移除自身的技术。

COM 通过包容和聚合机制提供一个对象可以利用其他对象的方法，使得内部对象的接口由外部对象公开。

封送处理机制使对象能够跨越线程、进程和网络边界使用，并允许它具有位置独立性的机制，从而实现了分布式组件技术。实际使用与 CORBA 类似，通过将 IDL 文件编译为存根和代理，屏蔽底层通讯细节。

COM 与 CORBA 相比较,它们的共同点在于都采用了面向对象技术和客户/服务器模式,着重于对象的互操作性和可重用性^[37]。它们的不同点在于:(1)技术基础:CORBA 从 1991 年开始制订,它的发展与 Internet 的基础协议 TCP/IP 的发展密切相关,因此 CORBA 是基于网络的;而 COM 则是基于桌面机,DCOM 是 COM 的延伸,实现了分布式组件,在网络上实现互操作的机制是 RPC。(2)跨平台性:由于 CORBA 和 CORBA 的技术基础不同,导致它们的跨平台性的差别。COM 对 Windows 以外的操作系统平台的支持不够理想,而平台无关性则是 CORBA 的目标之一,并且实现得较好,它支持 DOS, Windows, UNIX (极其变种操作系统)及实时操作系统。(3)对象接口:一个 CORBA 对象的所有操作界面都包含在一个接口里,客户可对对象的整个接口进行存取操作;而 COM 对象则有多个接口,在某一接口里不能直接访问其它接口,但可以通过 QueryInterface 函数查询其它接口的指针。(4)价格:DCOM 已经内嵌入 Windows 操作系统,与其一起捆绑销售,这一点是非常诱人的,因为可以看成是免费的。而使用 CORBA 技术的产品则相对来说比较昂贵。综上所述,CORBA 和 DCOM 各有各的优势。从纯技术上说,CORBA 比 DCOM 要略胜一筹,但 Windows 拥有世界上最大的装机容量,DCOM 是事实上的标准。

3.4.1.4 三种方式的简单比较

基于 Socket 的编程是最容易掌握的,而且由于使用最低层的协议,所以毫无疑问,效率也最高。而且,Java 对 Socket 这个类封装的非常好,建立一个 Socket 只需一句代码。

Web 方式受协议的限制,相对而言,这种方式开发起来比较困难,开发者必须仔细地操纵数据流处理许多复杂的,重复性的工作(如重复地生成 Html 页面的重复部分),必须考虑各种浏览器的兼容性等等。

组件方式对开发者的要求是最高的,组件技术通常处于软件金字塔的顶端。开发者要想学好组件技术并非易事,至少需要两年以上的编程经验,同时必须掌握一系列相关的基础知识,才可以熟练地利用组件开发程序,但是掌握之后,组件技术开发分布式应用程序是最为便利的。在 Windows 平台上,大部分分布式应用程序都是基于 DCOM 技术建立的。

综上所述,采用基于 Socket 的应用程序方式实现的网络通讯,大大地降低了对开发者的要求,也降低了开发的风险,同时缩短了开发周期。

3.4.2 服务器端的开发

网络实验室系统中可能存在多种服务器,如实验服务器,Http 服务器,数据库服务器等等。其中,最主要的是实验服务器,即 Java 开发的应用服务器。实验服务器主要作用是实

验的管理,用户的管理,实验的调度安排,用户与实验间通讯的建立。Http 服务器为网络实验室提供一个门户网站,便于大家使用,了解网络实验室,为网络实验室客户端程序的发布提供一个平台,为 Applet 方式的客户端提供最为基本的 Http 服务。数据库服务器用于存储用户信息,实验配置信息,具体实验数据,由应用服务器访问。

网络实验室采用双 C/S 的网络结构,为了保证内部网络的独立性和安全性,避免控制端和客户端之间的相互影响,实验服务器分为客户端服务模块和控制端服务模块,在不同的端口接受客户端和控制端的访问,同时,和大多数服务器一样,如果要处理多个请求,实验服务器应该使用多线程。

实验服务器的数据存储模块利用 JDBC 技术与数据库相连构建实验管理系统,实现对数据的存取,提供了强大的查询功能。

由于 Http 协议本身是一个无状态的协议,而虚拟网络实验室的实验过程需要状态的存在。所以,服务器端必须提供基于数据库和基于内存的两种会话服务,包括两个接口:会话服务接口和用户状态服务接口。会话服务接口主要用于提供创建用户会话和销毁用户会话。创建的会话以用户状态服务接口的形式供用户使用。用户状态服务接口保存了用户的所有会话状态,通过该接口用户还可以进一步得到网络实验室主服务接口和实验服务接口。用户接口提供了登陆、注册和注销等操作。客户端通过此接口,进行虚拟网络实验室的访问。服务器端在此接口之上进行用户管理。实验管理接口提供实验管理的各种功能。实验服务接口提供实验登陆、实验注销、发送命令和读取命令等方法。

3.4.3 客户端的开发

客户端以 Web 服务作为通讯方式,浏览器自动将 Java Applet 下载到客户端。Java Applet 与服务器端应用程序发送消息,建立一个 Socket。

客户端提供用户登陆、注销和用户管理等功能。登陆操作通过两个命令完成,即登陆命令和用户查询命令。登陆命令用于登陆服务器,创建用户线程;查询命令用于查询用户权限。用户注销发生在程序时,发送注销命令。服务器收到注销命令后,则立即进行用户注销处理,如关闭 Socket,线程中止等。用户管理主要包括用户信息管理和用户权限管理,提供用户的创建、删除、查询或修改等功能。用户信息包括用户名、密码和邮箱等信息,用户权限包括系统权限和操作权限。系统权限是指用户级别,如分为超级管理员、管理员和普通用户等。实验权限是指用于设定实验能力的权限。

3.4.4 控制端的开发

控制端与客户端类似，相对于服务器而言，它同样是一个客户端。控制端由 LabVIEW 编写，它负责硬件电路的搭建和信号的采集。所以，控制端有控制程序与监控程序两种程序，控制程序负责控制继电器，用来搭建不同的电路；监控程序负责监视控制程序状态和网络的状态，一旦发现异常则重新启动控制程序。控制程序与监控程序都通过 Socket 与服务器连接。

3.4.5 数据存储

数据存储系统采用 MySQL 或 Oracle 作为后台数据库，所有数据均以表的形式存在于数据库中。MySQL 数据库和 Oracle 数据库都属于关系数据库，有关数据库的知识不在此介绍。

3.4.6 客户端/服务器通讯模式

在网络实验室中，客户端与服务器端的通信存在四种模式：

1. 客户端主动发消息至服务器，服务器返回数据。
2. 客户端主动发消息，无数据返回。
3. 服务器端主动发消息，客户端数据返回。
4. 服务器端主动发消息，无数据返回。

这 4 种方式中，以 1 最为常见，3 最为少见。而且四种方式可能相互交叉，例如客户端主动发消息至服务器后，服务器在返回数据前，开始了模式 4 的通讯，即主动发送消息至客户端，然后返回数据，从而方式 1 和 4 实现了交叉。

常见的 Socket 编程模式都使用发送请求，等待接受的模式，但是，由于网络实验室中存在方式 3，4，所以无法直接套用该模式。

利用接收通知的特性，可以方便地处理方式 3 和 4，但是，若将发送和接收完全独立处理，对于方式 1 需要保留大量的状态变量，不利于程序的维护和进一步开发。

目前的网络实验室客户端通讯模块采用辅助线程，等待对话框等手段，使得即使四种方式相互交叉，4 种方式都可以方便地处理。

3.4.7 需要注意的问题

3.4.7.1 自定义协议

目前，互联网上所采用的协议是基于 TCP/IP 的协议簇。如较流行的 HTTP 协议、FTP

协议、SMTP 协议和 Telnet 协议。协议问题是网络通讯的核心问题。在远程控制虚拟网络实验室开发过程中，不可避免地要进行协议问题的讨论。在万维网中，我们不能逃离 TCP/IP 协议而任意自由的进行协议定义。所以，服务器端与客户端、控制端的协议定义都要基于 TCP/IP 协议之上（有关 TCP/IP 协议的内容将在附录中介绍）。在底层通讯上，使用 TCP 协议，采用 Socket 接口进行网络通讯。TCP 协议提供了无差错的，面向流的传输，所以应用程序必须规定流的格式，实现流的解析，即每次流传输由一个或者多个协议包组成。规定了协议包以后，应用程序面向协议包进行编程，隔离了底层通讯细节，为网络程序的开发提供了方便。

3.4.7.2 同步实验问题

由于此类实验属于远程控制类虚拟网络实验，同一个实验在同一时间内只允许一个用户拥有控制权，该用户可以自行搭建该实验电路进行实验。在这时间段内要求登陆的其它用户只能排队等待，但可以观看该实验进行的情况，不允许操作。只有等到当前操作实验的用户注销并释放控制权后，才允许下一个用户登陆操作。这种情况属于同步实验问题，开发时必须合理地考虑。

3.4.7.3 安全性考虑

三层的双 C/S 结构相对于双层 C/S 结构而言，系统具有更好的安全特性。在双层 C/S 结构中，控制部分直接连入 Internet，因而每个控制端都必须直接承受各种形式的网络攻击。而在三层的双 C/S 结构中，只有服务器端一个点联入 Internet，所有的控制端都处于局域网中，除了通过服务器端程序，外部程序无法以任何形式与控制端通讯，最大程度地保证了控制端的安全性。服务器是网络实验室的核心，其安全性是网络实验室正常运行的关键。服务器采用用户名/密码验证机制来增加服务器端的安全性。用户只有通过身份验证才能被服务器认可，建立用户线程，否则连接被关闭。客户端与服务器唯一的连接就是 SOCKET 通道，客户端对系统的破坏可以通过该通道完成，所以必须考虑防止通过实验模块窃取用户名、密码等重要信息。在服务器结构上，对每个正确登陆的用户在独立的线程中运行，当由于意外因素导致错误时，至多导致该线程的结束，而不会危及其他用户和整个服务器程序的运行，用户仅需重新登陆服务器即可。完善的权限机制也是保证服务器安全运行的又一有力机制。

第四章 小结

计算机技术和网络技术的飞速发展,多种优秀的虚拟仪器开发平台的推出,使得虚拟仪器技术渗透到国民经济与工农业生产中的各个行业,拥有非常广泛的应用前景。虚拟仪器技术在国内的研究还处于起步阶段,尚未开发出具有自主知识产权的虚拟仪器软件平台,所以还需要更多更优秀的专业人员来从事虚拟仪器的研究开发工作。

本研究课题,已经完成了五种电子技术实验中常用的虚拟仪器的开发,它们分别是虚拟双通道示波器、虚拟单(双)通道函数信号发生器、虚拟毫伏表、计数器和数字 I/O,并已经在电子技术实验中得到较好的使用。实验管理系统已经完成其基本框架的创建,目前也已经投入了使用之中。由于实验管理系统需要长期地维护,今后将逐渐地增加其它功能,使之不断完善。

在文中已经提到,LabVIEW 本身就是创建虚拟实验室一个很好的软件平台,将其与网络技术、数据库技术结合起来,实现虚拟仪器网络化。这样,可以将实验管理系统发展成基于虚拟仪器的虚拟网络实验室。所以,在论文的第三章详细地介绍了国内外虚拟网络实验室的研究情况,并提出了基于 LabVIEW 的具体几种实现方案及各种方案的优缺点。

本人在攻读硕士研究生阶段,结合本课题,在《江南大学学报(自然科学版)》期刊上发表有“基于 LabVIEW 的虚拟示波器设计”的研究论文一篇;

同时,还发表有《电动汽车用锌空电池电量预测方法的研究》的论文一篇,并与 2004 年 6 月参与了《集成电子技术基础》精品课程网络版课件的建设工作。

【参考文献】

1. 杨乐平,李海涛,赵勇等. LabVIEW 高级程序设计[M]. 北京:清华大学出版社,2003.
2. 刘君华. 基于 LabVIEW 的虚拟仪器设计[M]. 北京:电子工业出版社,2003.
3. <http://ni.com/china>.
4. 王敬喜,李裕能,牛玉新.基于 LabWindows/CVI 的网络化虚拟仪器.仪器仪表与分析监测[J].2003 年第 4 期.
5. 杨乐平,李海涛,肖相生. LabVIEW 程序设计与应用[M]. 北京: 电子工业出版社,2001.
6. 朱治国,郑建荣,刘小平.虚拟仪器及其常用开发软件.现代仪器[J].2004 年第 1 期.
7. 贾桂华.Visual C++实现虚拟仪器软件设计[J].信息与电子工程.2004 年第 2 卷第 1 期.
8. 朱晓华,章玉鉴.用 ActiveX 控件构造虚拟仪器[J].计算机应用.1998 年第 18 卷第 12 期.
9. 朱劼,马秦生,胡桂英.基于 HPVee 平台的虚拟仪器系统的建立.通信电源技术[J].1999 年第 3 期.
10. 耿详义,张跃平.JSP 实用教程.清华大学出版社.2003 年 5 月.
11. National Instrument Crop, PCI-6024E 数据采集卡用户手册.USA: 2000..
12. Aktan, B., Bohus, C.A., Crowl, L.A., Shor, M.H., 1996. Distance learning applied to control engineering laboratories. IEEE Transactions on Education, 39(3):320-326.
13. Sanchez, J., Morilla, F., Dormido, S. et al. 2002. Virtual and remote control labs using Java: a qualitative approach. Control Systems Magazine, IEEE, 22(2):8-20.
14. Guimares, E.G., Maffei, A.T., Pinto, R.P. et al.2003. REAL-a virtual laboratory built from software components. Proceedings of the IEEE, 91(3):440-448.
15. 李凌霄,刘志镜.网络虚拟电路实验室的 Java 实现.计算机辅助设计与图形学学报[J].2004 年第 16 卷第 3 期.
16. Ling KV, Lai YK, Chew KB, 2000. An Online Internet Laboratory for Control Experiments. IFAC Symposium on Advances in Control Education 2000.
17. Hahn, H.H., Spong, M.W., 2000. Remote laboratories for control education. Proceedings of the 39th IEEE Conference on Decision and Control, p.895-900 vol.1.
18. 张桦,王法玉,扬镇.远程虚拟实验台的 JAVA 实现.天津理工学院学报[J].2001 年第 17 卷第 3 期.
19. Gomez, F.J., Cervera, M., Martinez, J., 2000. A World Wide Web based architecture for the

- implementation of a virtual laboratory. Proceedings of the 26th Euromicro Conference, p.56-61 vol.2.
20. Ko, C.C., Chen, B.M., Chan, K.P., et al, 2001. A webcast virtual laboratory on a frequency modulation experiment. Proceedings of the 40th IEEE Conference on Decision and Control, p.3236-3241 vol.4
 21. Gustavsson, I., 2002. Remote laboratory experiments in electrical engineering education. Proceedings of the Fourth IEEE International Caracas Conference on Devices, Circuits and Systems, p.I025-1 - I025-5.
 22. Bottazzi, S., Caselli, S., Reggiani, M., et al, 2002. A software framework based on real-time CORBA for telerobotic systems; IEEE/RSJ International Conference on Intelligent Robots and System, p.3011-3017 vol.3.
 23. Daponte, P., Nigro, L., Tisato, F., 1994. Virtual laboratory an object-oriented framework. Instrumentation and Measurement Technology Conference, IMTC/94. Conference Proceedings. 10th Anniversary. Advanced Technologies in I & M., p.11-16.
 24. Corradini, M.L., Ippoliti, G., Leo, T., et al, 2001. An Internet based laboratory for control education. Proceedings of the 40th IEEE Conference on Decision and Control, p.2833 - 2838 vol.3.
 25. Stonick, V.L., 1994. An update on the CMU virtual laboratory. IEEE International Conference on Acoustics, Speech, and Signal Processing, p.VI/53 - VI/56 vol.6.
 26. Dixon, W.E., Dawson, D.M., Costic, B.T., et al, 2002. A MATLAB-based control systems laboratory experience for undergraduate students: toward standardization and shared resources. IEEE Transactions on Education, 45(3):218-226.
 27. 高国华,张永忠.在 LabVIEW 中利用 ActiveX 技术访问数据库[J].国外电子测量技术.2004 年第 2 期.
 28. 周毅,江建军.基于虚拟仪器的网络虚拟实验室构建[J].测控技术.2003 年第 22 卷第 12 期.
 29. National Instrument Crop, 虚拟仪器白皮书.USA: 1999.
 30. 秦兴华,刘亚斌,孙涣鹏.基于 LabVIEW 的虚拟仪器网络化实现.测控技术[J]. 2004 年第 23 卷第 2 期.
 31. 陈兵舫,杨乐平.基于 LabWindows/CVI 平台的仪器驱动器开发.国外电子测量技术.1998

- 年第 2 期.
32. 吴志锋,陈汉平,吴伟亮.基于 Web 的虚拟仪器技术.仪表技术[J].2001 年第 5 期.
 33. 李亚录,郝应光,唐须安.基于 Internet 的远程虚拟实验室系统的开发.微型电脑应用[J].2001 年第 17 卷第 8 期.
 34. 邹宁,王天江.CORBA 在网络管理中的应用研究.计算机与数字工程[J].2004 年第 32 卷第 4 期.
 35. 宋波,董晓梅.Java 应用设计[M].人民邮电出版社.2002 年 8 月.
 36. 冯少荣.DCOM/CORBA 技术的研究及应用.信息与电子工程[J].2004 年第 2 卷第 2 期.
 37. 童恒庆,聂会琴,李锡品. CORBA/COM/EJB 三种组件模型的分析与比较.计算机应用研究[J].2004 年第 4 期.
 38. 王泽报,巴林凤,杨品. LabVIEW 与虚拟仪器设计.世界仪表与自动化[J].1999 年第 3 卷第 6 期.
 39. de Queiroz, L.R., Bergerman, M., Machado, R.C., et al, 1998. A robotics and computer vision virtual laboratory. 5th International Workshop on Advanced Motion Control, p.694-699.
 40. Irawan, R., Ooi, M., Yeung, G., *et al*, 2001. A virtual laboratory experience based on a double tank apparatus. Proceedings of the 40th IEEE Conference on Decision and Control, vol.3 p.2815-2820.

致 谢

本论文是在王小海教授悉心指导下完成的，首先我在此对王老师表示诚挚的谢意。在我的整个研究生学习阶段，无论在专业学习，还是在课题研究和论文撰写上，王小海教授都给予了极大的关心、指导和鼓励。非常感谢王老师三年来对我不倦的教诲与栽培。

在课题进展中，樊伟敏老师和付晓程老师给予了我无私和诚恳的帮助。在他们的耐心指导下，我的课题能得以较顺利地进展下去。在读研期间，教研组的陈隆道老师、章安元老师、祁才君老师等都给予了我非常大的关心和指导。在此对你们表示真心地感谢。

课题进行期间，我曾多次参加美国 NI 公司在上海、杭州举行的技术研讨会，得到了 NI 公司工程师的指导和帮助，在此对他们表示感谢！

衷心感谢已毕业的刘成尧、陈云和正在读博士的胡海兵等师兄，感谢赵云峰、阎磊、曹金男、何汪平等同学，他们在我两年多的学习过程中给予了极大的帮助和支持。

衷心感谢我的父母、妹妹，是他们在不断鼓励我，安慰我，使我能顺利的完成硕士学业。

最后感谢所有曾给我帮助的人、在我人生中相遇的许许多多匆匆过客，愿他们一切顺利，心想事成。

李翼

二零零五年二月于浙江大学