

## 基于USB2.0的虚拟仪器设计

徐崇德, 朱荣新

(空军工程大学电讯工程学院, 陕西西安 710077)

摘要: 介绍了基于USB2.0的数据采集卡的设计与实现, 并运用该数据采集卡组成信号发生/采集设备, 成功开发了虚拟示波器、频谱分析仪。借助计算机的强大控制、计算能力, 使测试系统可提供面向用户接口、面向总线接口和面向数据处理的虚拟仪器功能。

关键词: USB2.0; 虚拟仪器; 固件程序: LabWindows/CVI

中图分类号: TH702 文献标识码: A 文章编号: 1009-3516(2006)03-0073-04

虚拟仪器技术是近几年发展起来的新一代仪器技术, 实现的关键在于外界数据的采集。运用高速数据采集卡和虚拟仪器技术, 可实现在信号测量、图像处理、音频视频信号处理等领域数据的高速、高精度的测量和处理。现在通用的基于PCI或ISA的数据采集卡存在很多缺点, 比如价格昂贵, 受计算机插槽数量、地址、中断资源的限制, 可扩展性差。基于USB接口的数据采集卡传输速度快, 且支持热插拔, 双向、同步传输, 具有高性价比。LabWindows/CVI是美国NI(National Instruments)公司开发的Measurement Studio软件组中的一员。它是32位的面向计算机测控领域的虚拟仪器软件开发平台。它以ANSI C为核心, 将功能强大、运用广泛的C语言与测控专业工具有机的结合起来, 实现了数据的采集、分析和显示。

## 1 USB数据采集卡原理

### 1.1 USB简介

USB即“通用串行总线”, 它支持在主机与各式各样即插即用的外设之间进行数据传输。一个完整的USB总线系统可以分为3个部分: USB总线的主机、USB总线的设备以及它们之间的连接。USB总线连接外设和主机时, 采用菊花链的形式对端点加以扩展, 有效地避免了PC机上插槽数量对扩充外设的限制, 减少PC机I/O接口的数量。

根据设备对系统资源需求的不同, 在USB规范中规定了4种不同的数据传输方式: 等时传输方式(Isochronous)、中断传输方式(Interrupt)、控制传输方式(Control)和批(Bulk)传输方式, 这些传输方式各有特点, 分别用于不同的场所。

USB协议支持两种传输速度, 即低速1.5 Mbps和高速12 Mbps, 而USB2.0协议的最高传输速率可以到480 Mbps。并且USB端口可以输出最大500 mA的电流, 可用长达5 m的传输线进行设备连接<sup>[1]</sup>。

### 1.2 系统框图

高速数据采集具有系统数据吞吐率高的特点, 要求系统在短时间内能够传输并存储采集结果。因此, 采集数据的快速存储能力和容量是制约加快系统速度和容许采集时间的主要因素之一。通常在数据采集系统中增加存储器件充当缓存器来满足要求, 常用存储器有FIFO、双端口RAM以及静态RAM等, 但容量小, 已经不能满足高速数据采集系统的需求。目前市场上的SDRAM和DDR SDRAM具有工作频率高、容量大、功耗低的特点, 数据线位宽可以达到64 bit, 完全适用于高速数据采集系统。因此, 在本设计中采用一个FPGA和SDRAM构成一个大容量的FIFO, 从而可有效的避免FIFO溢出, 同时也提供了一个低成本的满足高速实

收稿日期: 2005-12-01

作者简介: 徐崇德(1983-), 男, 云南永胜人, 硕士生, 主要从事测试计量技术及仪器仪表研究。

时流数据传输的解决方案(系统框图如图1所示。)

## 2 USB 数据采集卡硬件设计

### 2.1 芯片介绍

#### 2.1.1 EZ-USB 芯片

选用 Cypress 公司的 FX2 系列产品 CY7C68013, 它提供了对 USB2.0 的完整解决方案。该芯片包括带 8 kByt 片内 RAM 的高速 CPU、16 位并行地址总线加 8 位数据总线、I2C 总线、4 kByt FIFO 存储器以及通用可编程接口(GPIF)、串行接口引擎(SIE)和 USB2.0 收发器。CY7C68013 与外设有两种接口方式:可编程接口 GPIF 和 Slave FIFO。FX2 的 SLAVEFIFO 接口可以直接与外部设备“无缝连接”。它的一个主要的特点是:在 AUTOOUT 和 AUTOIN 模式下,数据流从主机传出或者向主机输入不需要经过 CPU 的处理。这样,低主频的 8051CPU 将不会影响高带宽的数据流传输。本系统内采用了 SLAVEFIFO 的 AUTOIN 传输模式。

#### 2.1.2 FPGA 芯片

设计中,FPGA 采用 Altera 公司的 CYCLONE 芯片 EP1C6-Q240。该芯片支持多种不同的 I/O 标准(包括 PCI 接口,可连接 ASSP 和 ASIC 设备的接口以及串行设备接口等)。实际应用中,FPGA 除了与 SDRAM 构成一个大容量的 FIFO 以外,还需要完成最为重要的 SDRAM 的读写控制功能。

#### 2.1.3 A/D 芯片

采用 Maxim 公司 MAX1180,MAX1180 是采用差分输入,带有宽带跟踪/保持(Trackand-Hold)的双通道 9 级流水线型结构的 ADCs,采样速率可达到 105 MHz。完全满足高速数据采样的要求。

### 2.2 硬件连接

设计方案中,SDRAM 控制器是数据流控制核心,负责数据的高速猝发读/写和定时刷新。当主机启动 A/D 转换后,SDRAM 控制器将采集结果存储到 SDRAM 存储器中,采集结束后,主机通过 USB 接口采用 DMA 传输方式将采集数据读到主机系统中。

设计中采用了 8 × 32 Mbit 的 SDRAM,采集数据先送入 A/D 数据缓存器,由 SDRAM 控制器读出并写入大容量 SDRAM 存储器。当 A/D 转换结束后,板卡修改状态标志或者发出中断请求,主机发出读取命令,SDRAM 控制器切换工作状态,把数据从 SDRAM 中取出,写入输出缓存器,由主机通过总线接口将结果读到系统中。其中,SDRAM 控制器负责对 SDRAM 的定时刷新。SDRAM 控制器根据设计,可以对 SDRAM 执行猝发读写、刷新等 12 种命令,完成对数据的访问<sup>[2]</sup>。SDRAM 控制器由 FPGA 编程实现。

A/D 转换器的采样时钟同时作为 CY7C68013 的 Slave FIFO 模式的读写控制时钟,即 CY7C68013 的接口时钟连接到 IFCLK 引脚。SLWR/SLRD 是 CY7C68013 Slave FIFO 的写使能/读允许信号。FPGA 向 CY7C68013 Slave FIFO 提供 Slave FIFO 输出允许信号 SLOE,仅在数据输出时有效。FD[15:0]为 16 位双向数据总线。FIFOADR[1:0]为端点 FIFO 选择信号。具体的硬件连接框图如图 2 所示。

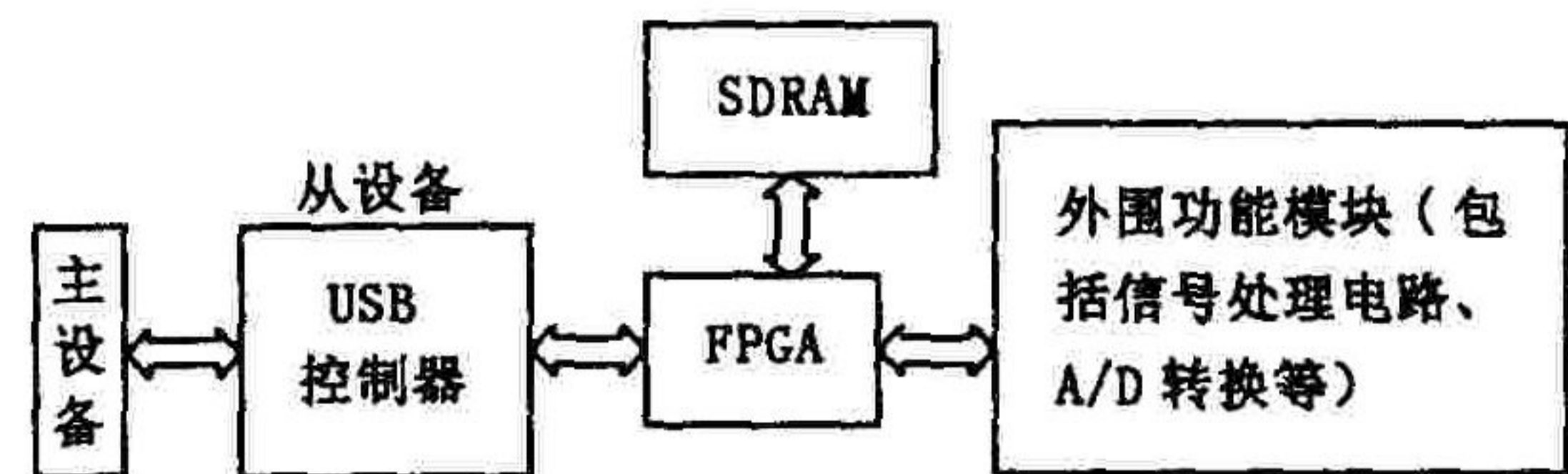


图1 USB 数据采集卡系统框图

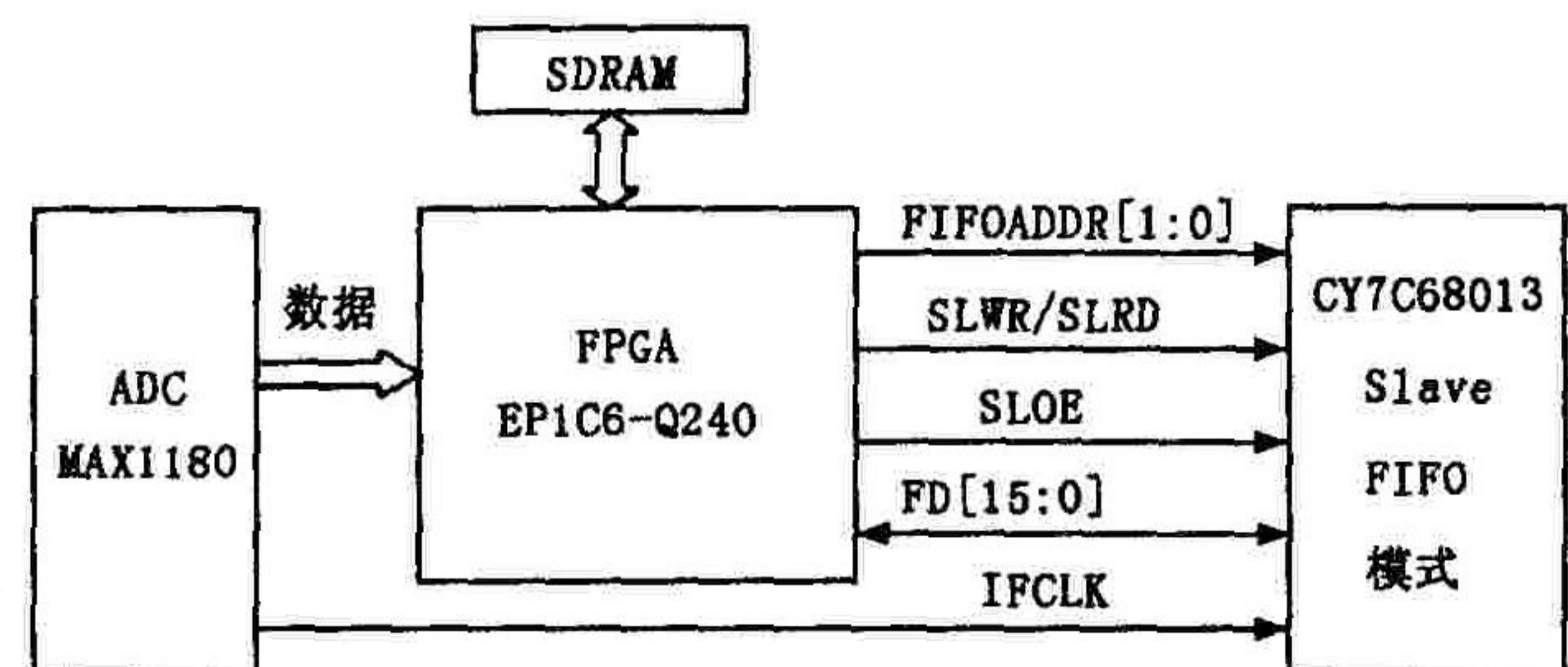


图2 硬件连接框图

## 3 系统软件设计与实现

本系统软件设计包括:固件程序、驱动程序和应用程序的设计。其中,固件程序是整个程序设计的核心。

### 3.1 USB 设备固件程序设计

固件程序相对比较复杂,需要用到大量的函数,但基本结构却相对简单,包括下面 3 个过程:①初始化:包括处理器和外围电路的初始化;②主函数:包括完成符合设备特定要求的代码;③中断处理:包括处理各种中断的程序代码。

设计中采用 CY7C68013 的端点 4 接收从主机发来的命令字,然后根据命令字进行相应操作,当端点 4 接收主机发来命令并进行完相应操作后,在端点 8 返回主机要求的操作结果或者仅仅返回控制命令执行的状态。接收数据端点 6 被设置为 SLAVEFIFO 模式,所以接收 FPGA 发来的数据,等待主机提取。由于端点 6 的 SLAVEFIFO 接口使用了 AUTOIN 模式,所以数据流由外部 FIFO 控制器控制,直接传入端点 6 的 FIFO,然后由主机提取。这里 CY7C68013 仅作为一条数据通道,而 CPU 并不参与这个过程。所以只要在固件程序的初始化程序段内配置好端点 6 的 SLAVEFIFO 接口模式,剩下的传输控制和其它的工作则由 FPGA 来做。

以下为部分固件程序代码,主要给出了端点 6 SLAVEFIFO 接口的 AUTOIN 模式配置程序。

```
void TD_Init( void ) //FX2 固件的初始化程序段,设置 FX2 的相应寄存器
{
    .....
    //设置 SLAVEFIFO 时钟由外部控制器提供。之前 FPGA 向 IFCLK 引脚提供接口时钟 IFCONFIG =
    0x43; //FIFO 接口时钟由外部控制器提供,最后两位设置 FIFO 接口模式为
    SLAVEFIFO
    SYNCDELAY; //同步延迟,由 FX2 技术手册规定
    EP6FIFOCFG = 0x0D; //设为 AUTOIN 方式,允许 0 长度包,最后一位为 1,设置接口为 16 位
    SYNCDELAY;
    EP6CFG = 0xE0; //EP6 方向为输入,块端点,包大小为 512 字节,双缓存。
    SYNCDELAY;
    PINFLAGSAB = 0x00; //,FLAGB = FF,为 FIFO 满标志
    SYNCDELAY;
    PINFLAGSCD = 0x00; //FLAGC = EF,为 FIFO 空标志,FLAGD = EP2PF
    SYNCDELAY;
    FIFOPINPOLAR = 0x00; //选择引脚极性,0 为低电平有效
    SYNCDELAY;
    EP6AUTOINLENH = 0x02; //和低位寄存器确定传输包大小为 512 字节/包
    SYNCDELAY;
    EP6AUTOINLENL = 0x00;
    SYNCDELAY;
    .....
}
void TD_Poll( void ) //此函数在设备运行时反复被调用,主要完成外部 FIFO 状态检测和数据传输。
{
}
BOOL TD_Suspend( void ) //该函数在设备进入挂起状态前调用
{
return( TRUE );
}
BOOL TD_Resume( void ) //设备重启后调用
{
return( TRUE );
}
```

在程序开始执行后,固件程序会执行以下步骤:第一步,初始化所有的内部状态变量;第二步,调用用户的初始设置函数 TD\_Init(),初始化 USB 总线设备(即数据采集系统)接口为非配置状态,并同时打开中断;第三步,EZ-USB 固件程序开始重新枚举设备(ReEnumerate),直到收到 SETUP 包为止;第四步,一旦 SETUP 包被检测到,固件程序框架就开始进行任务分配。任务分配就是依次重复地执行下面的过程:①调用用户函数 TDPoll();②检测是否有标准的设备请求,如果有,则执行指令并做出相应的操作;③检测 USB 核是否

有 USB 挂起事件,如果有 USB 挂起事件,则调用用户程序 TD\_Suspend(),TD\_Suspend()返回为真时,USB 核检测是否有新开始事件;如果没有 USB 挂起事件,则使处理器处于挂起状态。当检测到有重新开始的事件,USB 核调用用户程序 TD\_Resume(),并继续执行步骤③;当 TD\_Suspend()返回为假时,则继续执行步骤③。

### 3.2 驱动程序及 PC 应用程序设计

在 WINDOWS 平台下,USB 驱动程序由 3 部分组成:USB 设备驱动程序、USB 总线驱动程序和 USB 主控制器驱动程序,它们必须遵循 WIN32 驱动程序模型(WDM)。其中,WINDOWS 操作系统已经提供了处于驱动程序栈底的 USB 总线驱动程序和 USB 主控制器驱动程序。在 USB 设备驱动程序开发时,可以利用 Windows DDK,也可以通过借助一些辅助软件工具,如 DriverStudio、WinDriver 等。本设计中采用 Jungo 公司的 WinDriver 进行驱动程序开发,WinDriver 提供的驱动向导,可根据用户的需要,自动生成代码框架,从而简化了驱动程序的开发,且和 DDK 的结合良好<sup>[3]</sup>。

在 PC 机应用程序设计时,采用 LabWindows/CVI 进行应用程序的设计,以完成虚拟示波器和虚拟频谱分析仪的设计。最终实现的虚拟仪器前面板如图 3 所示,该虚拟仪器可实现示波器和频谱分析仪的功能。

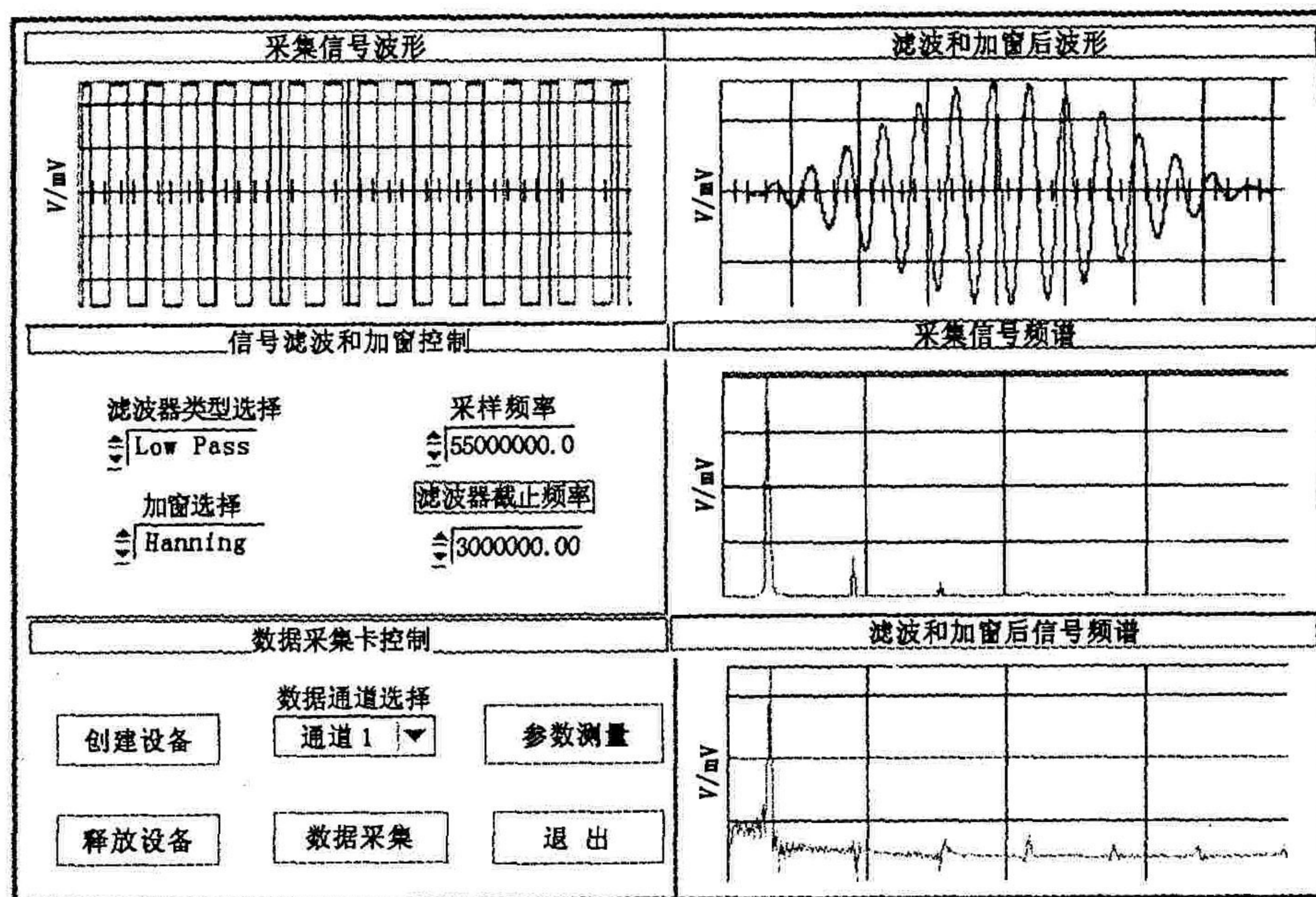


图 3 虚拟示波器和虚拟频谱分析仪前面板图

## 4 结束语

本文设计的数据采集卡解决了虚拟仪器在实现上的瓶颈制约,完成了外部数据与计算机的高速传输,理论上本数据采集卡可采集 50 MHz 的外部信号,可完全满足各方面的测量要求。本文中设计的虚拟示波器和虚拟频谱分析仪在实际应用中功能全面,使用方便,在具体运用中用户还可以根据功能需要设计 PC 机的应用程序,这也正是虚拟仪器的最大优势,即功能可由用户自定义。

### 参考文献:

- [1] Jan Axelson. VSB 大全[M]. 陈逸. 北京:中国电力出版社,2001.
- [2] 李卫,王彬,魏急波. SDRAM 控制器的 FQGA 设计与实现[J]. 电子工程,2004,30(10):29-31.
- [3] 刘映杰,张在峰,刘玮,等. 用 WinDriver 开发 PCI 设备驱动程序[J]. 信息技术,2004,28(2):78-80.
- [4] 张喜斌,白鹏,张启,等. 虚拟仪器技术在航空发动机稳定性测试中的应用[J]. 空军工程大学学报(自然科学版),2003,4(1):1-3.

(编辑:门向生)

(下转第 81 页)