

基于计算机 PCI 总线的虚拟信号源设计^{*}

仲元昌

(重庆大学通信工程学院 重庆400044)

摘 要 虚拟仪器是基于个人计算机(PC)作为硬件平台的一种新型仪器,为了充分利用 PC 机的硬件资源,设计通用性强的虚拟仪器,研究了 PCI 总线的典型特点及专用接口芯片,建立了基于 PCI 总线的 PCI 板卡设计方法,并设计了“多波形信号发生器插卡”。实验结果表明:该设计方法充分利用了 PCI 总线高速的特点,将直接数字式频率合成器(DDS)软件化,能产生正弦波、三角波、方波、噪声、扫频及其它复杂波形。由该 PCI 卡构建的虚拟多波形信号发生器可以方便、准确地选择波形和调节波形参数,具有较强的通用性。

关键词 虚拟仪器,PCI 总线,DDS,信号发生器

The Design of Virtual Multi-wave Signal Generator Based on PCI Bus

ZHONG Yuan-Chang

(College of Communication, Chongqing University, Chongqing 400044)

Abstract The virtual instrument is a new-style instrument based on computer as hardware flat roof. In order to design current virtual instrument with PC's hardware resource, PCI Bus and the interface chip are studied, the method of designing Add-In card is set up base on PCI bus, and the Add-In card of multi-wave signal generator is designed. Experimentation result show: the design method makes use of PCI bus' high speed features, DDS is realized by software. The signal generator can produce sin, triangle, square, noise, frequency sweep and others complex waves. The virtual multi-wave signal generator can be used to choose wave and to adjust its parameters conveniently and accurately. It has rather strong current.

Keywords Virtual instrument, PCI bus, DDS, Signal generator

1 引言

虚拟仪器(Virtual Instrument)是目前计算机仪器化发展的主要方向,它是虚拟技术与计算机技术在仪器领域的一个重要的应用,以计算机为基础,使用图形界面编程(GUI)技术,模拟实际仪器的面板、功能和操作,完成各种任务^[1]。

把具有测量功能的模块或仪器卡直接与个人计算机系统的总线联接起来,这样使用者操作这台计算机就像是在操作一台自己专门设计的传统电子仪器一样。虚拟仪器具有计算机与仪器的双重功能,这是传统仪器所无法比拟的。因此,虚拟仪器是计算机技术与仪器仪表技术日益融合的产物,已成为该领域研究的热点。

把一些功能模块制成数据采集卡的形式,用于信号的输入/输出,仪器的功能主要由软件来实现。因此虚拟仪器技术强调软件的作用,提出了“软件就是仪器”的概念^[1,2],这个概念克服了传统仪器的功能在制造时就被限定而不能变动的缺陷,摆脱了由传统硬件构成一件件仪器,然后再联成系统的模式。用户可根据自己的需要通过编制不同的测试软件来组合构成各种虚拟仪器。其中,许多功能直接就由软件来实现,打破了仪器功能只能由厂家定义,用户无法改变的模式,为广大的计量、电子测量和自动控制用户提供了一个技术先进、功能灵活、操作简便的开发环境,而且很容易与网络及其它外设连接。因此,虚拟仪器充分发挥了计算机的作用,便于组建复杂

的测试系统。笔者从研究虚拟仪器内核着手,以虚拟多波形信号发生器的设计为例,阐述虚拟器的设计原理。

2 PCI 总线结构

计算机与外部的信息交换是通过总线实现的,利用 PC 机作为数据采集平台,通过 PC 机的数据总线将采集到的数据高速传输到 PC 机内存中,是实现采集与数据存储的有效手段,也是虚拟仪器系统的重要硬件支撑^[3]。随着 GUI (Graphical User Interface)、多媒体等技术在 PC 机上的广泛应用,传统的 PC 总线(如 ISA),由于其带宽、位数等因素的限制,已不能满足系统工作的要求而被逐步淘汰,新型主板和高版本操作系统已不再支持 ISA 等总线。PCI (Peripheral Component Interconnect) 线结构如图1所示。

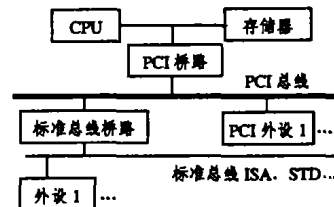


图1 PCI 总线结构

PCI 总线作为外部元件互连总线,被认为是最可靠、最灵活、高速的方案,具有众多独特的优点,使得多媒体插卡和数

^{*} 重庆市教委教改基金资助项目(0027);重庆大学学科建设基金资助项目(0321)。仲元昌 副教授,硕士,主要从事通信信号处理、载波通信、医用仪器、虚拟仪器等领域的教学与研究。

据采集卡都挂在 PCI 总线上。PCI 总线是一种靠近系统处理器的局部总线,故有很高的传输速度。当以 33MHz 总线时钟步操作时,32位的总线带宽可使数据通道的传输率达 132MB/s。PCI 总线独立于 PC 机的系统处理器,不会受 PC 机微处理器性能、速度的影响。因为,为 PCI 设计的器件是针对 PCI 的,而不是针对处理器的,因此,设备的升级独立于处理器的升级。PCI 总线为 32 位,可扩展为 64 位,由于采用了地址、数据总线复用结构,减少了管脚个数和 PCI 部件的封装尺寸,从而使板卡小型化,使板卡嵌入计算机系统中成为可能。PCI 总线具有即插即用功能,支持即插即用的操作系统,能够自动配置参数并支持 PCI 总线扩展板,使用方便。PCI 部件的驱动程序可以跨平台,兼容性好。

但受以下因素的影响,真正实现 PCI 总线接口是有一定难度的。

(1) PCI 总线是一种局部总线,与常见的 ISA 总线、IS 总线有着较大的差别,它需要通过接口芯片转化为用户总线。

(2) PCI 总线接口规范较为复杂,而且时序规定严格,一般芯片不能满足要求。

(3) 由于结构和时序的特点,需要 PCI 总线开发工具。比如基于软件的总线接口模型和基于硬件的 PCI 总线实习器、逻辑分析仪等。

目前,实现 PCI 接口的有效手段主要有专用接口芯片和 PLD。专用接口芯片放置于 PCI 插卡与 PCI 总线之间,是传递数据和控制信号的接口电路。专用接口芯片是解决设计难点最有效的手段,可以使开发者免于直接面对复杂的 PCI 总线规范,而致力于板卡功能的实现。这类专用芯片主要有 PLXTECH 公司的 PLX90XX 系列的 AMCC 公司的 S5933,还有专门针对 DSP 等接口的芯片^[3,4]。

3 PCI 专用接口芯片 S5933

S5933 就像一个桥梁,把 PCI 局部总线(PCI Local Bus)转换为用户总线(ADD-ON Data Bus)。用户总线可以方便地通过单片机(Intel i960)和可编程逻辑器件(GAL、CPLD、FPGA)控制^[3,4]。

该芯片提供了三种数据通道:邮箱通道(Mailbox)、直接映射通道(pass-Thru)和先进先出通道(FIFO,有寄存器和直接读写两种方式),和两组总线控制寄存器。PCI 总线控制寄存器和本地总线控制寄存器,分别用来控制 PCI 总线和本地总线的操作,完成中断、DMA 等相关内容的设定。

S5933 的 PCI 配置空间可以通过“串行/8 位并行 EEPROM”来设置,也可以不配置而采纳默认值。AMCC 提供了一个 AMCCPCI3(即 NVBUILDER)小软件,通过它设定基地址、寄存器、DMA 传输的控制方(主机/本地逻辑)、FIFO 通道的同步/异步工作等参数。

4 多波形信号发生器的设计

采用 DDS(Direct Digital Synthesizer)即直接数字频率合成器来构造多波形信号发生器^[5]。DDS 的基准频率(取样时钟)是固定不变的,它靠改变频率控制字来控制频率,实质是靠改变一个周期内对 ROM 中数据的(成比例增大或减小)取样点数来控制频率。

要用 DDS 产生任意波形,需要对它做些改造,至少要将正弦表 ROM 改为 RAM,并和计算机(DSP、单片机等 CPU 处理芯片)相连以传输数据。但这些方法要受到诸如累加器位

数、存储器 RAM 容量和 RAM 内的波形数据必须为整周期等的约束。我们利用 PCI 的高速特点,将 DDS 软件化,总体设计框图如图 2 所示。

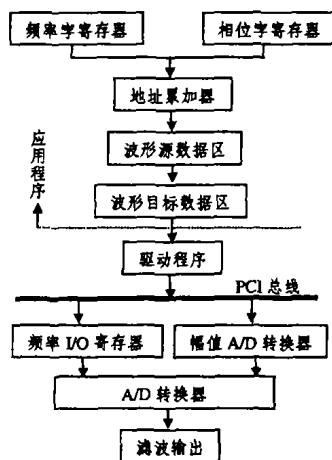


图2 系统框图

4.1 硬件设计

硬件用 S5933 处理接口逻辑,通过直接 FIFO 通道的 DMA 方式传输大量波形数据,邮箱传递少量的控制命令(主要为电压型幅值 DAC 的输入数据)。通过 GAL 芯片控制传输和单片机来处理用户中断、解析控制命令(修改并锁存幅值 DAC 的数据)。

4.2 软件设计

应用程序主要有数据处理、波形显示和数据输出三大部分。数据处理模块负责计算或调用文件,得到波形源数据,当数据区长度小于 1M 时,一次性调入程序缓冲区,大于 1M 时分批调入。数据输出模块完成和驱动程序(自编动态库)打交道,响应硬件中断,并执行本次 DMA 传输,为下一次 DMA 准备数据,即根据软件界面上设置的频率和相位,从源数据区抽取数据并计算得到目标数据;响应界面的幅值变化请求,传递一次幅值 DAC 的数据。波形显示就是把要输出的数据和幅值信息结合起来,按界面的设置,显示在示波窗口(见图 3)^[6]。

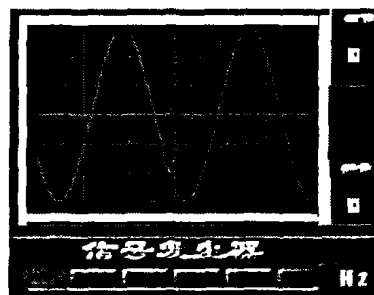


图3 虚拟多波形信号发生器界面

(1) 波形控制:波形源数据是产生波形的基础,它可以通过函数计算得到,也可以从文件中读取,其长度可根据波形的复杂程度或用户对输出波形的精度要求而改变。通过频率控制字和相位控制字,循环地从源数据中获得目标数据,通过 PCI 总线接口传到 D/A 转换器,就得到了所要的波形了。

(2) 频率控制:地址累加器在软件中即为指向波形数据区的地址偏移量。由于外部时钟频率 fclk 已固定,我们就可以通过调节频率控制字 q 来控制输出波形的频率 fout,假设数据区长度为 m,则:

$$q = m * f_{out} / f_{clk} \quad (1)$$

频率分辨率 Δf 和最小输出频率 $f_{\min}$ 相等,即:

$$\Delta f = f_{\min} = f_{\text{clk}}/m \quad (2)$$

往往在要求较低频率输出时,数据区可能会较长,采用了类似硬件相位截断的方法,也就是将同一数据重复输出多次 r (由波形的复杂程度、D/A 转换器的位数决定),这样可以成倍减小数据区。

(3) 相位控制:通过启用相移寄存器,就可以实现多通道间的相位关系控制了。如果需要 θ 角的相移,那么相移字为:

$$w = \theta * m / 360 \quad (3)$$

(4) 幅值控制:为了保证幅值改变时不改变波形质量,采用了双 D/A,分别用于波形输出和幅值调节。幅值 D/A 的输出电压作为波形 D/A 的参考电压 V_{ref} 输入,这样可以快速控制波形幅值。

结束语 将 DDS 软件化,波形、频率和相位等控制更加容易,硬件电路也大大简化。高速的 PCI 总线,再加上 DMA 方式,为产生长周期或非周期波形提供了保障。该多波形信号发生器可以产生标准的正弦、方波、三角波、锯齿波、白噪音、扫频正弦和其它非规则波形,其主要性能指标为:(1)外部时

钟:8MHz;(2)输出幅值:±5V 可调;(3)输出频率 0.1Hz~500KHz 可调,频率分辨率 0.1Hz。

参考文献

- 1 Speedler H T W. Virtual Instruments and Virtual Environments [J]. IEEE Instrument. Meas. Mag, 1999, 2(9):14~19
- 2 Cristaldi L, Ferrero A, Piuri V. Programmable instruments, virtual instruments, and distributed measurement systems: what is really useful, innovative, and technically sound [J]. IEEE Instrum. Meas, 1999, 2:20~27
- 3 The Measurement and Automation Catalog 2001. National Instruments
- 4 Benetazzo L, Bertocco M, Piuri V. A Web-based distributed virtual educational laboratory [J]. IEEE Transaction on Instrum. And Meas, 2000, 49(2):355~394
- 5 Bertocco M, Ferraris F, Offelli C, Parvis M. A Client-servers architecture for distributed measurement systems [J]. In: Proc. IMTC98, Str. paul, MN, May 1998. 67~72
- 6 Autodesk Inc. Object ARX Developer's Guide and Reference Manual. 1997

(上接第229页)

由公式 $C[i] = P/P[i]$ 可求出 $C[i]$ ($i=1, 2, 3, 4, 5, 6$)。

又 $C[i] \times D[i] \equiv 1 \pmod{P[i]}$ 且 $C[i]$ 与 $P[i]$ 互素,于是可通过扩展 Euclid 计算出 $D[i]$:

$D[1] = -1/224; D[2] = 1/2160; D[3] = 1/126; D[4] = 1/1890; D[5] = -1/240; D[6] = -1/3360$ 。再利用中国剩余算法求得:

$$R_1(x, x_1, 2) = \gcd(f_1(x, x_1, 2), f_2(x, x_1, 2)) = \sum_{i=1}^6 C[i] \times D[i] \times R_2(x, x_1[i], 2) = x^3 - 34x_1^2x + 52x_1x + 2$$

同理可得:

$$R_1(x, x_1, -7) = x^3 + 119x_1^2x - 474x_1x + 2$$

$$R_1(x, x_1, 3) = x^3 - 51x_1^2x + 117x_1x + 2$$

$$R_1(x, x_1, -3) = x^3 + 51x_1^2x + 117x_1x + 2$$

继续运用中国剩余算法:

$$P = (x_2 - 2)(x_2 + 7)(x_2 - 3)(x_2 + 3)$$

$$C[i] = P/P[i] = >$$

$$C[1] = (x_2 + 7)(x_2 - 3)(x_2 + 3)$$

$$C[2] = (x_2 - 2)(x_2 - 3)(x_2 + 3)$$

$$C[3] = (x_2 - 2)(x_2 + 7)(x_2 - 3)$$

$$\text{又由公式 } C[i] \times D[i] \equiv 1 \pmod{P[i]} \Rightarrow$$

$$D[1] = -1/45; D[2] = -1/360; D[3] = 1/60; D[4] = 1/120;$$

此时求得的公因式

$$R(x, x_1, x_2) = \sum_{i=1}^4 C[i] \times D[i] \times R_1(x, x_1, x_2[i]) = x^3 + 13xx_1x_2^2 - 17xx_1^2x_2 + 2$$

经验证,该多项式确为原多项式 $f_1(x, x_1, x_2)$ 和 $f_2(x, x_1, x_2)$ 的最大公因式。

总结 本文研究了利用中国剩余算法对多元多项式进行提升从而求得最大公因式的算法,理论研究和实验结果表明用该算法来提取多元多项式的最大公因式是可靠和高效的。下表是运用本算法运行的时间开销,实验环境为:CPU: PIII

450;内存:SDR 512M;Windows 2000 professional 操作系统,实验数据为10次运算的算术平均值。由实验结果可知,计算所需的时间与多项式的次数有密切关系,因此如果能在化元期间对变量的随机取值方面作一些优化,将会使得该算法对高次多项式的公因式提取有更高的效率。

mygcd(f_1, f_2)	0.055(s)
mygcd(f_3, f_4)	0.079(s)
mygcd(f_5, f_6)	0.112(s)

其中:

$$\begin{aligned} f_1 &= (123x^2 + 23x_1^2x_2 + 11x_2^2)(58x^2 - 17x_1x_2 + 3x^3 + 23x_1) \\ f_2 &= (11x^2 + 5x_1^2x_2 + x_2^2)x^3x_1^2(123x^2 + 23x_1^2x_2 + 11x_2^2)(58x^2 - 17x_1x_2 + 3x^3 + 23x_1) \\ f_3 &= (123x^2 + 23x_1^2x_2 + 11x_2^2)^2(58x^2 - 17x_1x_2 + 3x^3 + 23x_1)^2 \\ f_4 &= (11x^2 + 5x_1^2x_2 + x_2^2)x^3x_1^2(123x^2 + 23x_1^2x_2 + 11x_2^2)^2(58x^2 - 17x_1x_2 + 3x^3 + 23x_1)^2 \\ f_5 &= (123x^2 + 23x_1^2x_2 + 11x_2^2)^7(58x^2 - 17x_1x_2 + 3x^3 + 23x_1)^3 \\ f_6 &= (11x^2 + 5x_1^2x_2 + x_2^2)^2x^3x_1^2(123x^2 + 23x_1^2x_2 + 11x_2^2)^3(58x^2 - 17x_1x_2 + 3x^3 + 23x_1)^3 \end{aligned}$$

参考文献

- 1 Parikh S N, Matula D W. A redundant binary Euclidean GCD algorithm [J]. Computer Arithmetic. In: Proc. 10th IEEE Symposium on, 1991. 220~225
- 2 Mansour Y, Schieber B, Tiwari P. Lower bounds for integer greatest common divisor computations [J]. Foundations of Computer Science. In: 29th Annual Symposium on, Oct. 1988. 54~63
- 3 Wu P-Y, Chen C-L, Parallel J. Extended GCD algorithm [J]. Parallel Processing Symposium. In: Proc. Eighth Int., April 1994. 357~361
- 4 Brown W S. On Euclid's Algorithm and the Computation of Polynomial Greatest Divisors [J]. J. ACM, 1971(18):476~504
- 5 Calvez L C, Azou S, Vilbe P. Variation on Euclid's algorithm for polynomials [J]. Electronics Letters, 1997, 33(11):939~940
- 6 Singh B, Siddiqi M U. Multivariate polynomial products over modular rings using residue arithmetic [J]. Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on], 1995, 43(5):1310~1312
- 7 袁仁保, 马建. 求解孙子问题的算法及符号计算软件[J]. 小型微型计算机系统, 1996, 17(12):20~25
- 8 丘维声. 高等代数[M]. 高等教育出版社, 2000. 14~15