

文章编号: 1001-2486(2001)04-0088-05

一种六通道高速数据采集系统的设计*

付强, 李保国, 张旭东

(国防科技大学 ATR 重点实验室, 湖南 长沙 410073)

摘要: 提出并实现了一种基于 PCI 总线六通道高速采集系统的结构。该系统采用微机作为采集主控单元, 兼容三种体制雷达的采集要求, 可以实现六通道同时采集和大量数据的存储; 为了实现高速和大量采集, 系统采用 S5933 作为 PCI 总线接口, 利用 PCI 总线的高速传输能力, 满足主控单元与采集单元的数据传输要求; 同时利用双口 RAM 内部切换保证连续采集和连续传输; 控制电路采用 FPGA 实现, 简化了电路板设计, 提高了灵活性。软件系统在 Windows NT 平台上实现, 保证了整个系统的安全性与稳定性。

关键词: PCI 接口; 高速采集; 驱动程序

中图分类号: TN274⁺2 文献标识码: A

Design of a High Speed Six-channel Data Acquisition System

FU Qiang, LI Bao-guo, ZHANG Xu-dong

(ATR Key Lab, National Univ. of Defense Technology, Changsha 410073, China)

Abstract: This thesis presents a system architecture of six-channel high speed data acquisition on PCI bus and implement it. This system, using a PC as the control unit, can be used to get data from three different kinds of radar, and implement data acquisition through six channels, and store large amount of data. In order to realize high speed and large amount acquisition, we use S5933 as PCI interface, using the high transfer speed of the PCI bus, satisfying the required data transfer rate between the control unit and the acquisition unit. At the same time we use dual-port RAM to provide continuous acquisition; and use FPGA to implement the control circuits, simplify the circuit board design and improve the flexibility of the design. In order to ensure the security and stability of the whole system, we implement the software system under Windows NT operating system.

Key words: PCI interface; high speed acquisition; device driver

数据在精确制导自动目标识别算法研究中占有十分重要的地位, 它不仅是研究的基石, 而且是检验研究结果的准绳。由于实验条件限制, 常常采用计算机仿真产生数据。但仿真数据较难准确反映现代战场目标与环境的复杂性, 只能用作初步的验证。因此有必要研制一套用于外场实验的数据采集系统, 获取真实的测量数据。

本文研制了一种能兼容三种雷达制导体制的数据采集系统, 三种体制分别是圆锥扫描阶跃变频体制、线性调频脉冲压缩体制和脉冲多普勒体制。

1 硬件系统

1.1 系统要求

阶跃变频体制要求有 30 MHz 的采样率, I、Q 两路数据采集, 信号幅度 0 到 0.5 V, 在帧同步和脉冲同步信号控制下采样, 其中帧同步信号用来标识圆锥扫描周期的起始。

线性调频脉冲压缩体制要求有 32 MHz 的采样率, 由于采用单脉冲测角, 包括和、差三通道的 I、Q 信号共六路数据, 信号幅度 0 到 1 V, 在脉冲同步信号控制下采样。

脉冲多普勒体制要求有 4 MHz 的采样率, 单脉冲测角, 包括和、差通道三路数据, 信号幅度 0 到 1 V, 在脉冲同步信号控制下采样。

* 收稿日期: 2001-01-08

基金项目: 国家部委项目资助

作者简介: 付强(1963-), 男, 副教授。

这三种体制数据采集要求能采集特定距离窗内的数据，因此必须能根据需要设置距离窗。距离窗的产生是通过控制两个参数实现，控制相对于同步脉冲的延迟计数确定距离窗的起始，通过控制采集计数确定窗的宽度。此外，需要控制同步脉冲记数，阶跃变频体制还要求满足帧同步要求。

经过分析可以发现，脉冲多普勒体制的 4MHz 采样时钟可以通过 32MHz 时钟 4 分频得到；阶跃变频和脉冲多普勒体制通道数都小于线性调频体制；三种体制信号都可以通过放大满足 0 到 2V 的量程；都是通过脉冲同步进行触发采样。

1.2 硬件系统实现

系统分为数据采集和 PCI 数据传输两个模块，分别用一块电路板实现，然后将它们复合为一个 PCI 插卡。这样的实现省略了数据的长线传输，而在高速并行长线传输会导致信号畸变和引入噪声。由于计算机内部的电磁环境十分复杂，在设计时需要考虑抗干扰措施。

系统的框图如图 1 所示。

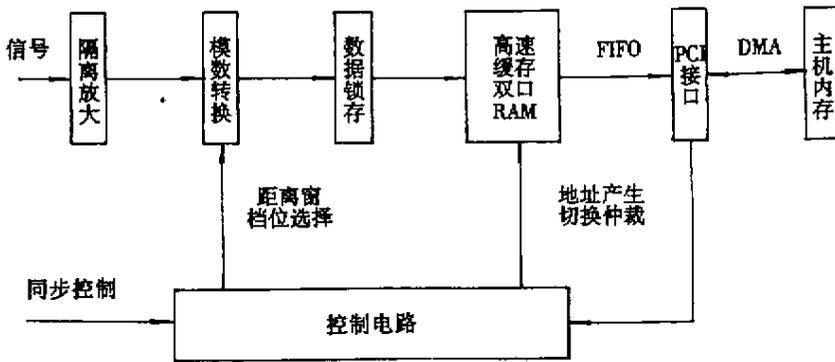


图 1 系统框图

Fig.1 System diagram

1.3 单元电路设计

我们采用基于 PC 的设计方案，用 PC 对采集系统进行控制并完成数据存储。

1.3.1 采集电路

前端模拟电路采用 AD812 作为输入放大和隔离。AD812 在 $\pm 5V$ 的电源下 0.1 dB 平坦度带宽达 40 MHz，接正向输入端，输入电阻为 15 M Ω ，完全满足本系统的要求。为了适应不同的输入电压范围，采用多路开关切换反馈电阻网络，多路开关的切换由主控机通过控制字选择。

AD 芯片采用 AD9058。AD9058 的最高转换速率为 50 MHz，模拟带宽为 175 MHz，满足系统要求。

1.3.2 数据传输存储电路

在采集系统中，数据的传输存储始终是困扰设计者的难题。存储介质的容量、速度、价格与采集传输要求之间的平衡是问题的核心。

通常基于计算机的高速采集系统都采用三个阶段来完成数据的记录。首先将采集得到的数据缓存到高速缓存中，然后通过 DMA 将数据转移到计算机的内存中，最后将内存中的数据存成文件。

三种体制中线性调频体制要求的传输速率是最高的，因此主要考虑满足这种体制的要求。由于线性调频体制采样频率为 32 MHz，共有六个通道，8 位的精度，所以在全速采集情况下，采集的数据率为 $32 \times 6 = 192\text{MB/s}$ ，但是实际算法要求只是在限定的距离窗内采集。经过计算，最高的传输要求为 40MB/s。

由于高速缓存容量有限，所以采用“乒乓”方法进行数据传输。具体方法为：将缓存划分为相等的两个部分，即高端和低端的 16K \times 16 空间，在同一时刻分别用于采集数据存储和 DMA。两个空间的切换是一个需要考虑的问题。由于本系统最高的传输要求为 40MB/s，所以只要当采集空间满时启

动传输即可。但是我们希望系统有更广泛的用途,所以对切换机制进行了修正。

这样可以自动根据采集速率调整传输,数据存储就只受到主控计算机的内存限制,但是在传输数据率较高时,会出现采集数据的损失。

1.3.3 PCI 接口

PCI接口的实现有许多方法,我们选择了AMCC公司的S5933芯片实现PCI接口。S5933提供了PCI配置寄存器,实现PCI总线规范2.1版本。它提供了以下三种传输数据的方式。

第一种是通过4个32位的MAILBOX与1个状态寄存器,给用户提供了灵活的命令和消息传输方法。

第二是通过一组32位的FIFO实现DMA传输。

第三种是在PCI总线和ADD-ON端建立多个直通(PASS-THRU)区域进行数据传输,每个区域可以独立定义存储区大小和总线宽度。这种方式适合于外接处理器和主机进行数据交换。

在本系统中,采用MAILBOX方式将控制参数送到采集板上,利用FIFO方式实现DMA。

1.3.4 系统外围控制

系统参数控制与传输控制是采用一片FPGA来实现。

FPGA设计灵活,修改方便,非常适合新系统的开发。我们采用的是Xilinx公司的SpartanXCS30芯片,它的容量最高为30000逻辑门,满足要求。FPGA简单时序如图2所示。

2 软件系统

本软件分为两大部分:NT下的驱动程序(.sys文件)和用户界面(WIN32应用程序)。其中驱动程序是本软件的主要部分,负责驱动采集卡,包括写采集参数到相应的寄存器,写控制寄存器,读取DMA数据,进行中断处理等功能;用户界面负责用户与驱动程序的接口,包括设置采集参数、利用读取的数据画波形、存储DMA数据等功能。

开发本控制软件所应具备的软件条件:

- Windows NT4.0 零售版本 (free build)
- DDK (DEVICE DRIVER KITS) for Windows NT4.0
- WIN32 SDK (software development kits) for Windows NT4.0
- visual C++ 6.0 企业版
- Numega Softice for NT

2.1 驱动程序

设备驱动程序实际上是指管理实际数据传输和控制特定物理设备的程序,为了保证操作系统的稳定和安全,NT规定设备驱动程序必须按照一定的规范来编写。WindowsNT实际上是一种基于对象的操作系统,内核模式的NT设备驱动程序通过一个叫做IO管理器的NT对象来和设备之间进行通信。在本系统中,驱动程序对PCI接口芯片S5933编程。具体步骤如下:

- STEP1: 设置S5933的工作方式;
- STEP2: 设置采集参数,形成控制字;
- STEP3: 设置S5933 FIFO工作参数;
- STEP4: 将控制字写入MAILBOX,
启动数据采集;
- STEP5: 传输数据,传输完毕后向主
机发出中断INTA;
- STEP6: 处理中断,将数据存入硬盘。

① DriverEntry 例程

采用了DDK(驱动程序开发工具包)函数HalGetBusData,对总线号、设备号、功能号作循环来查找PCI设备,一旦查到PCI设备,再用AMCC公司提供的厂家标识号、设备标识号来确定所找到的

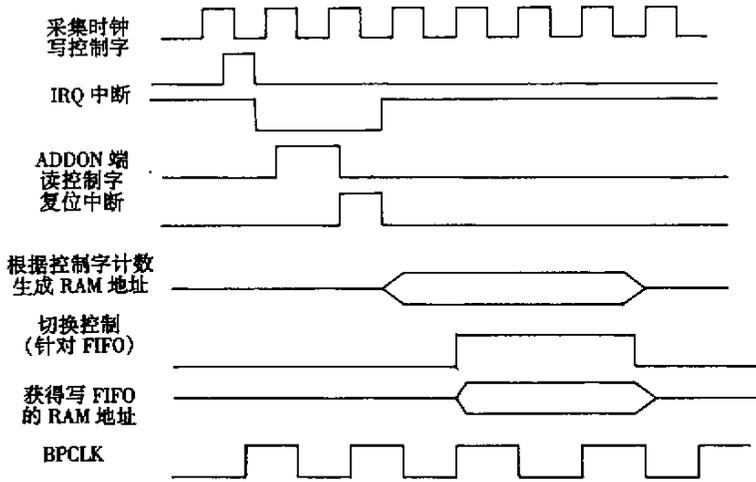


图 2 FPGA 简单时序

Fig.2 FPGA simple timing relationship

PCI 设备正是我们所要找的采集卡。HalGetBusData 返回指向配置空间结构的指针，这个结构中有驱动程序后面要用到的系统为采集卡分配的一些资源。

② Dispatch 例程

当 IO 管理器得到一个 IO 请求时，它使用请求的函数代码调用驱动程序中几个 Dispatch 例程中的一个。Dispatch 例程验证该请求，并让 IO 管理器把它发送到设备进行处理。本采集系统的 IO 请求函数代码共有四个：IRP_MJ.CREATE，IRP_MJ.CLOSE，IRP_MJ.DEVICE_CONTROL，IRP_MJ.READ。它们分别对应用程序的 CreateFile（打开文件句柄）、CloseHandle（关闭文件句柄）、DeviceIoControl（设备 I/O）和 ReadFile（读文件）请求。其中后两者是主要部分，用来实现读写寄存器和读取 DMA 数据。处理函数中主要用到了 READ_PORT_ULONG 和 WRITE_PORT_ULONG。

③ startIO 例程。

④ AdapterControl 例程

⑤ ISR（中断服务例程）

⑥ DpcForIsr 例程

后四个例程作者就不再赘述了，进一步了解可参看文献 [5, 7]。

2.2 用户界面（Win32 应用程序）

作者采用 Visual C++ 6.0 作为开发平台，使用 VC6.0 提供的 MFCAppwizard 向导来生成一个单文档的应用程序，在控制菜单里设了三个子菜单：设置参数、启动采集、显示波形；设置参数对应一个采集参数对话框，用户可以选择不同的参数设置；

显示波形则对应在主窗口同时显示六个通道的波形。整个界面非常简单。设置采集参数在对话框类的 ONOK 函数中进行，其中用到了 DeviceIoControl 函数，形式如下：

```
result = DeviceIoControl ( hDevice ,
                          IOCTL_WRITE_MAIL ,
                          &mail ,
                          sizeof ( mail ) ,
                          NULL ,
                          0 ,
                          &returnlength ,
```

NULL)

解释如下：result：BOOL 型变量，当 DeviceIoControl 函数访问成功时为 TRUE，反之为 FALSE；
 hDevice：用 CreateFile 打开的设备句柄；
 IOCTL_WRITE_MAIL：IO 控制功能代码；
 Mail：MAIL 结构的一个对象，MAIL 结构是在对话框类头文件中和驱动程序中都定义了的三个 Outgoing Mailbox 值的一个结构；
 returnlength：返回的字节数。

启动采集的响应函数是在文档类中定义的，为的是方便数据的存取。这个函数执行完后数据就已经从 S5933 的 ADD.ON 端读到了用户缓冲区中了。ReadFile 的函数形式如下：

```
result = ReadFile ( hDevice ,
                  buffer ,
                  readlength ,
                  &returnlength ,
                  0 )
```

解释如下：readlength：用户定义的需要读的数据长度；
 buffer：调用者缓冲区，事先申请了（readlength）字节长的空间；
 returnlength：实际读出的数据长度。

3 结束语

利用本采集系统进行了实际测量外场数据，结果证明这个系统完全满足所提出的指标要求，能够做到大容量高速连续采集，而且稳定可靠，测得的数据真实地反映出了目标与环境的特性，已经用于检验各种目标检测、鉴别、识别、跟踪算法，取得了良好的效果。

参考文献：

- [1] Fu Qiang. Automatic Target Recognition Based on Incoherent Radar Returns [A]. NAECON '96 USA. 1996.
- [2] 付强. 雷达目标回波数据采集系统的设计与实现 [J]. 系统工程与电子技术, 1995 (9): 71-76.
- [3] Liu Yang, Fu Qiang. A High-Speed and Adaptive Multichannel Data Recording System for Radar Target Analysis [A]. NAECON '95 USA. 1995.
- [4] 沈兰荪. 高速数据采集系统的原理与应用 [M]. 北京: 人民邮电出版社. 1995: 52-54, 59-62.
- [5] Windows NT4. 0 设备驱动程序设计指南 [M]. 北京: 机械工业出版社, 1997.
- [6] AMCC S5933 PCI Controller Data Book [R]. 1998.
- [7] Windows NT 技术内幕 [M]. 北京: 清华大学出版社 1993.

