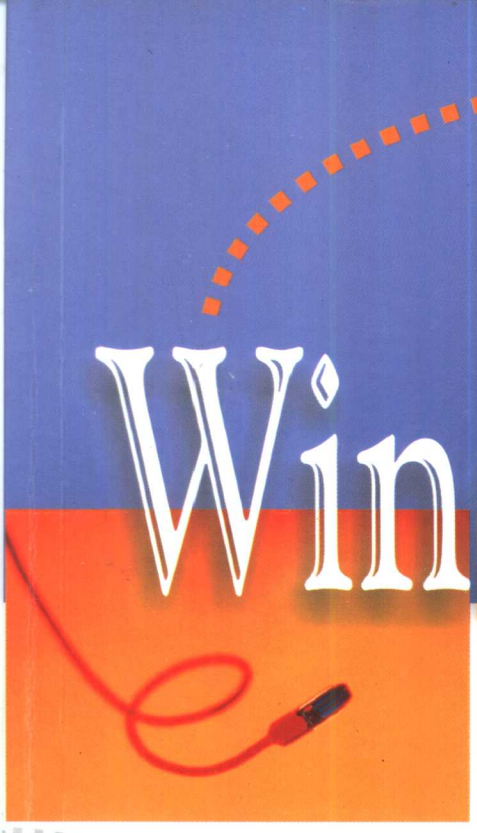
Windows环境下
软硬件接口技术

李圣怡　戴一帆　王宪平　等编著



国防科技大学出版社

Windows 环境下软硬件接口技术

李圣怡　戴一帆　王宪平　罗兵　彭小强　编著

国防科技大学出版社

·长　沙·

图书在版编目（CIP）数据

Windows 环境下软硬件接口技术 / 李圣怡，戴一帆，王宪平等编著. —长沙：国防科技大学出版社，2001. 1
ISBN 7-81024-690-9

Ⅰ. W…　Ⅱ. ①李… ②戴… ③王…　Ⅲ. 电子计算机-接口　Ⅳ. TP334.7

中国版本图书馆 CIP 数据核字（2000）第 56449 号

国防科技大学出版社出版发行
电话：（0731）4572640　邮政编码：410073
E-mail：gfkdcbs@public.cs.hn.cn
责任编辑：张静　责任校对：石少平
新华书店总店北京发行所经销
国防科技大学印刷厂印装
*
787×1092　1/16　印张：25.25　字数：583 千
2001 年 1 月第 1 版第 1 次印刷　印数：1—4000 册
*
定价：34.00 元

内 容 提 要

本书结合软件和硬件的新技术，全面、深入地介绍了新型计算机接口系统的原理和应用设计，着重讲解了 32 位 Windows 操作系统下，计算机接口系统的诸多新问题、新技术。其内容包括：总线技术、I/O 端口技术、并行通信端口、串行通信端口、内存管理和映像接口技术、中断技术、DMA 技术、Windows 设备驱动程序。至于已经不太常用或者其它同类书中已有的内容，本书中就略去了。本书的主要特色是结合新技术，面向接口系统，使读者可以很快地把这些新技术应用到实际工作中去。本书包括很多实用的例子，读者可以通过它们，尽快了解新技术、新器件，缩短学习和工作所需的时间。

本书可作为计算机应用、通信与系统、仪表与测量、信号与电路、监测与控制等专业的高等院校高年级本科生、研究生的教材，也可供广大工程技术人员阅读和参考。

前 言

相信很多读者正在为计算机接口系统发愁。现今的接口系统大多要求有漂亮的Windows界面、友好的人机接口、快速的数据传输等功能。使用DOS下的debug、速度极慢的标准并行端口和RS232串行端口以及一大堆的74系列逻辑器件来满足现今的接口系统要求确非易事，甚至于不可能。具体来说，虽然在32位Windows（包括Windows 9x，Windows NT，Windows 2000）中实现端口操作并不那么困难，但对于进行内存读写、处理硬件中断、实现DMA操作、在Windows下最方便地实现准确的时序操作，甚至编写Windows设备驱动程序这些典型的接口问题，就不是那么容易实现了。令人遗憾的是，以前很少有书籍从接口系统的角度全面地、实用地介绍这些问题的解决方法。

而且，随着计算机硬件和软件的快速发展，一贯以高可靠性，高价格为特征的工业控制领域和其它应用中的计算机接口技术发生了很大的变化。工业领域中的应用软件逐步向Windows平台转移，Windows NT也因其优异的稳定性在很多项目中被选做工业应用操作系统；各种计算机插件板也逐步向PCI总线转移，以期获得更高的数据传输能力和更强的智能。然而，大量的计算机接口系统仍然被局限于DOS系统和旧的ISA总线结构，滞后流行技术3～4年。造成这种滞后的原因包括Windows具有保护机制、Windows本身具备非严格实时性、缺乏应用新总线和新器件所需要的专门化知识以及专门化开发工具等多方面，其中很重要的一点就是缺乏合适的书籍。

虽然目前已经有了很多讲述计算机接口技术的书籍，但本书的特色之处可以使读者紧跟当前软件技术和硬件技术的发展，解决32位Windows操作系统下的计算机接口问题。以第九章“Windows 设备驱动程序”为例，这章没有像软件书籍中那样长篇大论，而是在整体尺度上让读者掌握设备驱动程序，细致地解析复杂易混的术语，然后提供了一个仔细注释了的设备驱动程序实例，让读者可以很快地掌握这项“深奥”的技术。

本书的其它章节也有这样的特色。即使是“传统”的技术，如8250、8255、8259的使用，一方面把它们与新技术交织在一起，如Windows下对串行端口的编程、Windows下的中断实现及其性能分析，一方面对这些传统技术进行详尽的叙述，这样，读者就可以在日常工作中非常方便地查找需要的内容。当然，本书更是详尽地介绍了IEEE 1284（EPP、ECP）、RS485、RS422等较新的技术。

感谢国防科技大学出版社为本书的出版和发行所做的卓有成效的工作，尤其感谢本书编辑的细致工作。另外还要感谢各位好友，感谢他们无私提供的好素材和不断的支持。

限于作者水平，疏漏之处在所难免。恳请各位同行和各位读者不吝批评指正。

作者

2000．10

目　录

第一章　微机接口技术概述

第二章　PC 系列计算机的总线标准

第五章 串行通信接口

第一章　微机接口技术概述

1.1　接口与接口技术

1.1.1　概述

自从 20 世纪 70 年代初第一个微处理器 Intel 4004 问世以来，微型计算机的发展极为迅速。在短短 20 多年内，经历了 4 位机、8 位机、16 位机、32 位机几个大的发展阶段，目前 64 位微机也已研究出来。微处理器和微型计算机的性能提高了成百上千倍，功能一代比一代增强，其发展速度远远超过了大型机、中型机和小型机。但是再好的微机，其强大的计算功能往往是由接口外围设备的能力和处理外界信息的能力表现出来的。其道理是显然的。首先，任何计算机必须有一条接受程序和数据的通道，才能接收外界的信息来进行处理，这就必须有输入设备，如键盘、鼠标等。而处理的结果还必须送给要求进行信息处理的人或设备，才能为人或设备所利用，这就必须有输出设备，如 CRT 显示终端、打印机、绘图仪等。其次，为了将计算机应用于数据采集、参数检测和实时控制等领域，必须向计算机输入反映测控对象的状态和变化的信息，经过中央处理器处理后，再向控制对象输出控制信息。这些输入信息和输出信息的表现形式是千差万别的，可能是开关量、数字量，更可能是各种不同性质的模拟量，如温度、湿度、压力、流量、长度、刚度、浓度等等，因此，需要把各种传感器和执行机构与微处理器或微型计算机连结起来。

由此可见，为了完成一定的实际任务，微型计算机都必须与外部客观世界进行广泛的信息交换和传输，即与各种外部设备相联系。但是，微型计算机并不可能直接与任何外部设备相联系，而必须通过一定的规范接口才能对外部设备进行检测与控制，从而与它们交换信息。

所以我们可以把微机接口定义为微型计算机与外部设备之间的公用边界，是把微型机与外界各种检测、控制对象联系起来的纽带和桥梁。接口是任何微机应用系统必不可少的重要组成部分。所用微机组成一个实际应用系统的关键技术是接口技术，任何微机应用开发工作都离不开接口的设计、选用和连接。可以说，任何一个微机应用系统的研制和设计，在确定了算法之后的主要工作就是微机接口的研制和设计，需要设计的硬件是一些接口电路，所要编写的软件是控制这些电路按要求工作的驱动程序。

1.1.2　微机接口的两个特征

从学习和掌握微机接口技术的角度看，微机接口技术有两个特征。

首先，微机接口技术是一种用软硬件综合来完成某一特定任务的技术。为构成某种特定功能的微机系统而进行二次开发时，都不可避免地要遇到硬件设计、连接和软件编程控制等问题。因此，要求从事接口技术工作的人员必须具备计算机硬件和软件两方面的基本知识与能力，既能熟练地分析、设计一般接口中常见的数字逻辑电路，又能熟练地用汇编语言或其它高级语言编制各种接口所需的驱动程序。

其次，微机接口的基本任务是要把微处理机和外部设备连接起来，使两者之间正确地交换和传递信息，这就决定了接口技术总体上的复杂性。一方面必须对微型计算机的硬件和软件，包括它的总线标准、组织结构及其指令系统和汇编语言编程等知识有较深入的了解。另一方面，还必须对被接口的外部设备的原理和特性，特别是电气工作特性、信息类型和格式、控制信号及其相互之间的时序关系等等有较准确的认识。

综上所述，微机接口技术是计算机应用领域中一种软硬结合、机电结合、以电为主的综合技术，是多种技术交叉的新兴边缘学科，具有较强的理论性和工程实践性。

1.1.3 接口功能

各类外部设备和存储器，都是通过各自的接口电路连接到微机系统总线上去的，因此用户可以根据自己的要求，选用不同类型的外设，设置相应的接口电路，把它们连到系统总线上，构成不同用途、不同规模的应用系统。

微机接口一般要具有如下功能：

一、 *数据缓冲功能*

为了解决 CPU 高速与外设低速的矛盾，避免因速度不一致而丢失数据，接口中一般都设置数据寄存器或锁存器或其它的数据缓存设备，如一些较高级的设备都提供 FIFO 缓存区。另外，为了实现 CPU 与外设之间的联络，接口电路还要提供数据缓冲区“空”、“满”、“准备好”、“忙”、“闲”等状态信号，以便向 CPU 报告接口或外设的工作情况，实现主机与外设之间的握手。

二、 *接收和执行命令代码的功能*

对于较复杂的外部设备，来自 CPU 的控制命令一般均以命令代码的形式送到接口的命令寄存器，再由外设的接口电路对命令代码进行识别和分析，形成若干个控制信号，以产生相应的具体操作。

三、 *信号转换功能*

由于外设所提供的信息类型、所需的控制信号和它所能提供的状态信号往往与微机的总线信号不匹配，信号变换就不可避免。如测温的热电偶给出的是模拟信号，而计算机只能接受数字信号，中间必须进行模拟信号到数字信号的 A / D 转换。信号转换也包括 CPU 的信号与外设信号的逻辑关系、时序配合以及电平匹配上的转换，它是接口设计中的一个重要内容。

另外，信号的串行－并行转换和信号具有一定的位数宽度也是某些接口必需的功能。

四、 设备选择与设备管理功能

比较复杂的微机系统中一般带有多种外设，同一种外设也可能配备多台，所以必须正确地选择相应的外设。在存在复杂的外设网络、外部设备总线或是高速数据传输的情况下，必须对外部设备进行有效的管理，典型的例子如仪器中常用的 IEEE-488 接口就具有设备管理的功能。

五、 中断管理功能

当外设需要及时得到 CPU 的服务，特别是在出现故障时，在接口中设置中断控制器，为 CPU 处理有关中断事务（如发出中断请求、进行中断优先级排队、提供中断向量等），这样既做到微机系统对外界的实时响应，又使 CPU 与外设并行工作，提高了 CPU 的效率。

六、 可编程功能

现在的许多接口芯片基本上都是可编程的，这样在不改动硬件的情况下，只修改相应的驱动程序就可以改变接口的工作方式，大大地增加了接口的灵活性和可扩充性。

在芯片技术的发展推动下，可编程接口也取得了巨大发展。如现在所广泛采用的固件（FirmWare）技术，在不改变设备的硬件电路的前提下，通过改变设备中 BIOS 里的控制程序、改变 FPGA 可编程芯片和一些在线可编程芯片中的逻辑程序，就可以实现设备的升级或功能更新。

1.2 主机与接口之间信息传送的方式

在接口电路设计时，根据应用系统的要求，在 CPU 与外设之间采用适当的信息传送控制方式是非常重要的。传送的方式不同，CPU 对外设的控制方式也不同，从而使接口电路的结构及功能也不同。传送控制方式一般有三种，即程序控制方式、中断方式和 DMA 方式。

1.2.1 程序控制方式

程序控制方式（Program Polling）通常又分为无条件传送方式和条件传送方式两类。

一、 无条件传送方式 （同步方式）

无条件传送方式是一种最简单的输入 / 输出控制方式，它使用方便，所需的软硬件都较简单，其所有的操作由执行程序来完成。采用这种传送方式，要求外设和 CPU 始终是准备好的，CPU 直接执行输入或输出指令，便可实现数据传送。在传送变化的数据时，无条件传送方式必须保持 CPU 与外设之间的同步。无条件传送方式一般只需要数据端口。

二、 查询方式 （条件传送方式）

查询方式是指 CPU 在传送数据之前，要先检查外设是否“准备好”，若没有准备好，则继续查询其状态，等待外设，直至外设准备好，即确认外部设备已具备传送条件之后，才能进行数据传送。显然，在这种传送方式下，CPU 每传送一个数据，需要花费很多时间来等待外设进行数据传送的准备，因此 CPU 的效率很低，且 CPU 与外设不能同时工作。但实现这种传送方式的硬件接口电路简单，在 CPU 不太忙且传输速率要求不高时，可以采用这种方式。采用查询方式的接口一般需要两个端口，即数据端口和状态端口。

查询方式的程序控制方式要求外部设备有状态握手信号。

1.2.2 中断方式

采用中断方式传送信息时，不需要反复查询外部设备的状态。当外设已准备好，需要和 CPU 交换数据时，它就通过 I / O 接口给 CPU 一个中断请求信号。CPU 响应接口的中断请求，暂停正在执行的程序 (通常称为主程序)，插入中断服务程序，完成数据传输。由于 CPU 省去了对外设状态查询和等待的时间，从而使 CPU 与外设可以并行地工作，因此大大提高了 CPU 的效率。

这种方法的传输速率一般比程序控制方式高，但是在要求实时和快速响应情况下，必须考虑 CPU 进行任务切换和进入中断的时间消耗。而且中断方式仍然需要 CPU 参与控制每次的数据传输，实际上数据就是 CPU 通过执行 IN、OUT、MOV 等指令进行传输的，另外传输地址的形成也是由 CPU 来完成的。所以这种方法的传输速率也不是太高。

1.2.3 DMA 方式

采用 DMA（Direct Memory Access，直接存储器存取）方式，使 CPU 不参加数据 I / O，而是由 DMA 控制器 DMAC 来实现内存与外设之间、外设与外设之间的直接快速传送，从而减轻了 CPU 的负担。更重要的是，传输地址由 DMAC 通过硬件直接生成，可以大幅度地提高传输速率。

DMA 方式把 I / O 操作过程中外设与内存交换信息的控制交给了 DMA 控制器，实质上是在硬件控制下而不是在 CPU 软件的控制下完成数据的传输，大大提高了传输速率，这对大批量数据的高速传送特别有用。但是，此时 CPU 必须让出对计算机局部总线的控制。

1.3 接口设计的一般考虑

在进行接口分析和设计时，首先要做的就是充分的需求分析，然后在掌握新技术的基础上，考虑成本和时间等因素的约束，确定合适的解决方案。这也是工程设计的一般方法。

计算机接口实际上是以信号为中心的，因此进行相应的接口设计时应对完成功能所需的硬件和软件作统筹考虑，确定哪些功能由硬件完成，哪些功能由软件实现，作出合理的方案。在此基础上适当地选用I/O接口芯片，进行硬件接口电路设计及连接。然后，根据硬件连接的情况，进行接口驱动软件的分析与设计。

1.3.1 分析接口两侧的信号

计算机接口是以信号为中心的，所以在对两个不同的设备或系统进行连接时，首先应该分析信号的连接和变化。通常接口的两侧中，一侧是CPU或微机系统，另一侧是外设。对微机系统一侧，应搞清楚是什么类型的系统总线，什么类型的CPU，以及系统总线提供的数据总线、地址总线的宽度和进行信号传输时必需的控制总线信号。注意控制信号的逻辑定义（如有效电平、有效脉冲沿、信号方向、是否为三态信号等）、时序关系的特点及要求。数据总线和地址总线比较规整，对不同的系统总线或CPU其变化不大。而控制总线则因微机系统总线或CPU的类型不同，其信号线及有关时序的关系差别较大，因此，对控制信号线的逻辑定义与时序分析至关重要。

由于外设种类繁多，型号不一，所提供的信号线五花八门，其逻辑定义、时序关系、电平高低差异甚大，所以外设这一侧的情况更为复杂。一般来说，对外设一侧的分析重点应放在搞清有关外设的工作原理、性能特点及要连接的信号线的定义及时序上，从而找出需要接口为它提供哪些信号和功能才能正常工作，它反馈给接口哪些状态信号来表明其工作进程，以达到与系统总线交换数据的目的。一般来说，不管外设如何复杂，只要将它们的工作原理及与接口相连接的信号线的特性分析清楚，对接口电路的分析与设计也就不难了。

把系统总线与外设两侧的有关信号线直接连接通常是不行的。这涉及信号电平、驱动能力、信号时序以及故障保护等几个方面的问题。所以，一般都需要对信号进行转换与改造，使之协调匹配，来同时满足接口两侧的要求。

1.3.2 接口方案与接口芯片的选择

微电子技术和集成电路技术的不断发展，造成了现在几代不同的接口芯片并存的局面。因此，接口芯片的选择不再是个局部问题，而是关系到了接口方案的确定、制作、维护和成本等各方面的问题。

通常可以采用下面几种方式进行接口电路的硬件设计：

1．采用传统的中、小规模的标准TTL，CMOS系列集成电路IC器件及传统的数字逻辑系统的设计方法进行接口电路设计。

2．利用现有的各种用途的通用或专用的可编程大规模集成电路接口芯片，并结合少量的中、小规模的IC进行接口电路设计。

3．接口的主要功能利用现有的各种用途的通用或专用的可编程大规模集成电路接口芯片，而接口逻辑、信号转换等功能不再使用中、小规模的逻辑芯片，而是利用可编程逻辑器件PLD（Programmable Logic Device）器件等各类可编程逻辑芯片（如GAL、

EPLD、CPLD、FPGA 等），并借助 VHDL 或众多的 EDA 工具进行相应的接口设计。例如，设计一块 ISA 总线的计数器插件卡，其计数功能由 Intel 8254 实现，而其译码等与 ISA 总线相关的接口由一片 GAL 来实现。现在，一般工程技术人员最常用的就是这种设计方案。

4．硬件上采用中、大规模的可编程逻辑芯片，如 FPGA 和各种现场可编程器件等，具体功能由可编程逻辑芯片中的软件模块实现。如 PCI 的插件卡同 PCI 总线的接口，可以利用专门的大规模接口芯片 S593X 实现，也可以做成软件核的形式加载到 FPGA 的一部分中去来实现这个接口逻辑功能和总线功能。目前，许多大批量生产的产品采用这种设计方案，比如网卡，就是一个主芯片再加一个大规模可编程逻辑芯片完成全部功能。

在接口电路设计中，往往不需要繁杂的电路参数计算，而是需要熟练地掌握和深入了解各类芯片的功能、特点、工作原理、时序关系、使用方法及编程技巧，以便根据设计要求和成本限制，合理选择芯片，把它们与微处理器或系统总线正确地连接，并编写相应的芯片初始化程序或驱动程序。

1.3.3 接口驱动程序分析与设计

硬件电路只提供了接口工作的基础，只有在驱动程序的控制下，接口才能发挥其作用。而且，随着系统复杂度的增加，驱动程序也越来越重要。在许多设备中，往往是硬件和驱动软件协同完成工作。硬件确定后，驱动程序对硬件性能的发挥就起着至关重要的作用。现在，计算机 DIY（Do It Yourself）爱好者经常到互联网上下载的各种设备（如显示卡）的驱动程序，就有可能使硬件的功能发生巨大的飞跃。更有甚者，随着新型可编程设备的不断增多，驱动程序对于硬件功能的更新和发展就起着更大的作用。最近的典型例子如 Creative 公司的 Sound Blaster Live！系列声卡，通过驱动程序的升级，更换声卡上 DSP 的程序，就从 256 个合成复音（计算机模拟现实乐器的能力）提高到了 1024 个合成复音。

接口驱动程序分析与设计离不开相应的操作系统。在 DOS 和 Windows 下，接口驱动程序的设计有非常大的区别，就是不同版本的 Windows，驱动程序也是不一样的。例如，16 位 Windows 3.x 和 16 位、32 位代码混合的 Windows 95 操作系统支持的虚拟设备驱动程序（VxD），在纯 32 位的 Windows NT、Windows 2000 中就不能使用；而 Windows 98、Windows NT 和 Windows 2000 支持的 WDM 类型设备驱动程序也不能在 Windows 3.x 和 Windows 95 中使用。

对于微机系统中的标准设备（如 CRT、Keyboard、Printer、HD、FD、串口、并口等），在 BIOS 中都有相应的功能块子程序供用户调用。在 Windows 下，系统也提供了支持。但是接口设计者常常碰到的是一些非标准设备，所以通常需要自己动手编制专用的接口驱动程序。因此，只有了解外设的工作原理和接口电路的硬件结构，了解软件操作系统的特点和驱动程序编写方法，才能正确编写相应的接口驱动程序。

总之，在接口设计中，接口硬件电路与接口驱动程序是紧密相联的，既要进行接口硬件分析与设计，还要对接口的软件进行分析和设计。通常，若接口硬件发生变化，接