工业和信息化部“十二五”规划教材

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

微型计算机接口技术

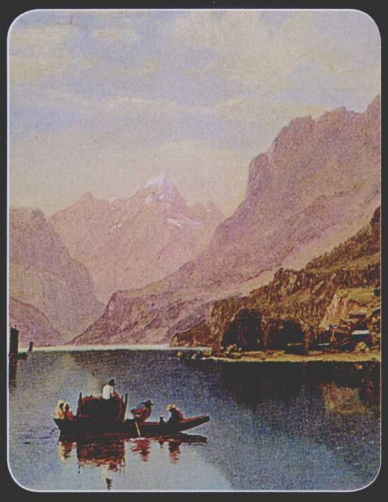
Microcomputer Interface Technique

刘乐善 陈进才 主编

刘乐善 卢萍 陈进才 李畅 编著

周可 主审

- 内容基础：以微机接口技术基础知识和设计接口电路基本技能训练为导向
- 方法具体：以编程模型和接口两侧分析方法为利器，使学习硬件知识不再难
- 应用可鉴：以简单而典型的设备接口为实例，阐述接口技术应用一般规律，举一反三



名家系列

中国工信出版集团

人民邮电出版社 POSTS & TELECOM PRESS

工业和信息化部“十二五”规划教材
21世纪高等学校计算机规划教材
21st Century University Planned Textbooks of Computer Science

微型计算机接口技术

Microcomputer Interface Technique

刘乐善 陈进才 主编
刘乐善 卢萍 陈进才 李畅 编著
周可 主审

名家系列

人民邮电出版社
北京

图书在版编目（CIP）数据

微型计算机接口技术 / 刘乐善，陈进才主编 ；刘乐善等编著. -- 北京 ：人民邮电出版社，2015.9
21世纪高等学校计算机规划教材
ISBN 978-7-115-40028-4

Ⅰ. ①微… Ⅱ. ①刘… ②陈… Ⅲ. ①微型计算机－接口技术－高等学校－教材 Ⅳ. ①TP364.7

中国版本图书馆CIP数据核字(2015)第186925号

内 容 提 要

本书根据当前微型计算机（简称微机）技术的发展与应用发生深刻变化的形势，对接口技术的教学内容进行了调整与更新，在保持传统接口技术基本内容的基础上，增加了无线通信接口等新技术。对接口电路的配置方式明确提出外置式接口与内置式接口两种布局，这两种布局各有优势，对台式微机与嵌入式微机是各得其所。对接口技术的应用开发，本书提出了接口技术分层次的概念，把接口技术划分为上层设备接口与底层总线接口两个层次，不仅厘清与延伸了接口技术的内容，也指明了在不同层次进行接口技术应用开发的要求与任务。在接口技术学习方法上，采用接口芯片与接口模块的编程模型方法，有力地化解了学习硬件的难度。

因此，这是一本内容实用、方法具体、应用可鉴、编写思路新颖与编写风格独特的微型计算机接口技术教材。本书内容组织遵循教学内容与教学方法并重的原则，具有可读性好、可操作性强的特点。

本书适用面广，既可作为普通高等院校和职业技术学院理工科所有专业的接口技术教材，又可作为广大从事微型计算机应用与开发人员的自学参考书。

◆ 主　　编　刘乐善　陈进才
编　　著　刘乐善　卢　萍　陈进才　李　畅
主　　审　周　可
责任编辑　邹文波
责任印制　沈　蓉　彭志环
◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路 11 号
邮编　100164　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京圣夫亚美印刷有限公司印刷
◆ 开本：787×1092　1/16
印张：17.75　　2015 年 9 月第 1 版
字数：469 千字　　2015 年 9 月北京第 1 次印刷

定价：42.00 元

读者服务热线：(010)81055256　印装质量热线：(010)81055316
反盗版热线：(010)81055315

前　言

目前对接口技术课程内容的讨论很热烈，主要集中在两个方面：一是讲台式微机系统还是讲嵌入式微机系统的接口技术；二是讲单个分立式的接口芯片还是讲多个接口并与处理器集成在一起的接口模块。主张后者观点的主要理由是，实际应用中（指在计算机的控制应用领域中）后者比前者多，尤其是在当今“互联网+”、物联网以及设备智能化应用中，ARM、MCU 有着明显的优势。

这一话题的讨论，涉及两个层面的问题，一是接口技术所依附的微处理器，是选基于 X86 处理器的台式微机，还是选基于 ARM 处理器的嵌入式微机；二是微机接口电路本身及其布局，即接口电路的配置方式是采用分立式接口芯片并放在 CPU 外部，构成外置式接口电路，还是采用接口模块的形式与 CPU 集成在一块，形成内置式接口电路。

本书以讲台式微机系统的外置式接口为主。为什么要这样安排？首先，台式微机通用性强，体系结构完整，作为教材具有典型性，而嵌入式微机多为量身定制，典型性、代表性不够，作为教材不好选型。其次，台式微机与嵌入式微机配置的接口所起的作用与基本工作原理相同，只是接口电路的布局与组织结构不尽相同，各有所长。一般来说，前者接口电路的功能完整、全面，具有通用性，因而结构复杂，常常做成单独分立的接口芯片，布局在 CPU 外部，通过总线与 CPU 进行连接，形成外置式接口电路。后者接口电路功能单一，针对性强，因而结构简单，往往以接口模块形式与 CPU 放在一起，形成内置式接口电路，无需通过总线在外部进行连接，简化了芯片之间的连线，系统整体结构紧凑。所以，从系统结构上来看，台式微机采用外置式接口，嵌入式微机采用内置式接口，可以说是各得其所。但从教学的观点来看，采用分立式接口芯片，构成外置式接口电路，然后通过系统总线与接口连接，使接口与系统的边界清晰，而内置式接口模糊了接口与系统总线的边界，不利于对接口电路工作原理的理解。为此，在本书中，讲接口设计时，增加了接口电路设计解决方案的分析，以加强对接口电路配置方式问题的考虑。

对在教材中要不要讲那么多接口芯片，也有不同的看法，对此，还是来简单地回顾接口芯片的来历与作用。在早期的计算机系统中并没有设置独立的接口电路，对外设的控制与管理完全由 CPU 直接操作。随着微机技术的发展，其应用越来越广泛，外设门类、品种大大增加，且性能各异，操作复杂，CPU 再也无力直接操作如此庞杂的外部设备，需要在外部设置支持电路来协助 CPU 处理外设的问题，从而导致了接口的出现，有了接口也就有了实现接口功能的接口芯片，所以讲接口技术就回避不了接口芯片。问题是如何讲，为此本书提出接口芯片的编程模型方法，其基本思路是，分析一个接口芯片，不论它是简单还是复杂，主要注重其内部寄存器编程特性——寄存器格式、寄存器的端口地址以及装入寄存器的信息，而不深究芯片的逻辑结构，这样既降低了学习硬件的难度，又不失对芯片的应用。

从接口技术应用角度来看，做原创性开发与做二次应用设计所涉及的接口技术问题是不同的。为此，本书提出接口分层次的概念，把接口分为两个层次：上层设备接口和底层总线接口——总线桥。上层设备接口是面向设备的，因此也称设备接口，包括用户总线与设备之间的接口和用户应用程序。底层总线接口是面向总线的，也称总线接口，包括用户总线与 PCI 总线之间的总线桥和设备驱动程序，两者形成一个完整的接口。一般用户在已有的平台上进行接口应用设计，只做上层接口（因为底层接口已经由平台供应商给做好了），若要开发新设备的接口，则还需要进行总线桥和设备驱动程序设计。显然，开发底层接口的难度与复杂度比上层接口要高得多，由于篇幅原因，本书不展开讨论，可参见文献［7］～［11］，［23］，［24］。

按接口技术的层次，本书只讲了上层接口，且以接口技术基本原理与接口设计方法为主，因为读者若真正掌握了这些接口技术基本知识，能举一反三，在实际应用中，遇到更复杂的接口问题就会知道如何下手了。在校学生作为初学者，学习一门课程时，还是应从基本的知识开始，逐步深入为宜。例如第 5 章例 5.1 和例 5.2 中断技术举例时，只举了用按键申请中断的例子，设备虽然简单，但可以从中领会到可屏蔽中断的全过程以及学习编写中断服务程序的方法，突出了中断技术知识点的核心。然后到第 9 章 A/D、D/A 转换器接口中，采用中断方式的数据采集的例 9.2 举例时又把中断技术的应用提升了一些，进一步加深对中断技术的认识。这种以基本原理与方法为牵引，循序渐进、逐步深入的安排，可能更易被读者接受。

微机接口技术是一门实践性很强的课程，除了课堂理论学习之外，还需要强有力的实践性环节与之配合。对于接口技术课程，如果只有理论知识学习，而无实验实践，其学习效果与收获会大打折扣。为此，我们编写了实验教材，研制并推出了“微机接口与汇编语言实验平台”，适于配合课堂教学实验和课程设计、毕业设计、实习和实际动手能力的实训等多种实践环节。实验系统和本书的内容紧密配合，相互补充，书中例举的接口实例，可以通过实验平台进行实际操作和实验，真正做到课堂原理讲授和实践环节一脉相承。

本书共 13 章，其中第 12 章无线通信接口和第 13 章基于 FPGA 的接口电路设计是新增加的教学内容，其余各章均在原有教学内容的基础上进行了大幅度修改和补充，尤其是在本书编写的思路和内容的组织方面有新的提升，以期为读者提供一本（对读者）可读性好、（对教师）可操作性强、内容基本、方法具体、应用可鉴的微机接口技术教材。

本书由刘乐善、陈进才主编，周可主审。其中，卢萍、陈进才编写第 13 章，李畅编写了第 11 章和第 12 章，其余章节由刘乐善编写。全书由刘乐善统稿。本书的出版得到了华中科技大学计算机科学与技术学院大力支持，在此表示衷心的谢意。同时要特别感谢参考文献的作者，他们提供了丰富多彩的技术文献。

由于计算机技术发展神速，加之编者水平有限，书中难免存在不足之处与错误，诚挚希望读者赐正，不胜感激。

编　者

2015 年 5 月于华中科技大学

目　录

第1章 概述

在微机系统中，微处理器的强大功能必须得到外设的支持才能实现，而外设与微处理器之间的信息交换是通过接口来实现的，接口技术已成为直接影响微机系统功能和微机推广应用的关键技术之一。因此，微机接口（以下简称接口）技术已成为当代理工科大学生必须学习的基本知识和科技与工程技术人员必须掌握的基本应用技术。

本章将对接口技术的基本概念进行介绍和讨论。

1.1 接口的基本任务与接口技术的发展概况

1.1.1 接口的基本任务

在微机系统中，接口处于总线与I/O设备之间，负责CPU与I/O设备之间的信息交换。接口在微机系统所处的位置决定了它在CPU与设备之间所起到的桥梁作用。因此，接口技术是随CPU技术及总线技术的变化而发展的，也与被连接的I/O设备密切相关。

在实际应用中，人们总是利用接口来加入用户自己的设备或模块构成应用系统，可见接口技术是应用系统开发必不可少的关键技术。

微机接口技术的基本任务有两个：一是实现I/O设备与总线的连接；二是连接起来以后，CPU通过接口对I/O设备进行访问，即操作或控制I/O设备。因此，接口技术的研究就是围绕I/O设备与总线如何进行连接以及CPU如何通过接口对I/O设备进行操作展开的。这涉及接口的连接对象及通过什么方式与途径去访问设备等一系列问题。

例如，对I/O设备的连接问题，涉及微机的总线结构是单总线还是多总线；对设备的访问问题，涉及微机采用何种操作系统。这些都是接口技术需要进行分析和讨论的内容。

1.1.2 接口技术的发展概况

接口技术的发展是随着微机体系结构和被连接的对象，以及操作系统的发展而发展的。当接口应用环境发生了变化，作为中间桥梁的接口也必须变化。这种变化与发展，过去是如此，今后仍然如此。

早期的计算机系统，接口与I/O设备之间无明显的边界，接口与I/O设备控制器做在一起。到8位计算机系统，在接口与I/O设备之间有了边界，并且出现了许多“接口标准”。8/16位计算机系统，接口所面向的对象与环境是XT/ISA总线、DOS操作系统。现代微机系统，接口所面向

的对象与环境是 PCI 总线、Windows 等操作系统。这使得接口技术面临许多新概念、新方法与新技术，而且出现了层次结构。下面简要地说明接口技术的变化发展过程。

在早期的计算机系统中并没有设置独立的接口电路，对外设的控制与管理完全由 CPU 直接操作。这在当时外设品种少、操作简单的情况下，是一种简单可行的方法。然而，随着微机技术的发展，其应用越来越广泛，外设门类、品种大大增加，且性能各异，操作复杂，从而导致了接口的出现，其原因如下：首先，如果仍由 CPU 直接管理外设，会使 CPU 完全陷入与外设打交道的沉重负担中，导致 CPU 工作效率低下；其次，由于外设种类繁多，且每种外设提供的信息格式、电平高低、逻辑关系各不相同，因此，主机对每一种外设都要配置一套相应的控制和逻辑电路，使得主机对外设的控制电路非常复杂，不易扩充，这极大地阻碍了计算机的发展。为了解决以上问题，开始在 CPU 与外设之间设置简单的接口电路，后来逐步发展成为独立功能的接口和 I/O 设备控制器，把对外设的控制任务交给接口和 I/O 设备控制器去完成，这样极大地减轻了主机的负担，简化了 CPU 对外设的控制和管理。同时，有了接口之后，研制 CPU 时就无须考虑外设的结构特性如何，反之，研制外设时也无须考虑它是与哪种 CPU 连接。CPU 与外设按照各自的规律更新，形成 CPU 和外设产品的标准化和系列化，促进了微机系统的发展。

接口经历了固定式简单接口、可编程复杂接口和智能接口几个发展阶段。各种高性能接口标准的不断推出和使用，超大规模接口集成芯片的不断出现，以及接口控制软件固化技术的应用，使得接口向更加智能化、标准化、多功能化及高集成度化的方向发展。市场上还流行一种紧凑的 I/O 子系统结构，就是把接口与 I/O 设备控制器及 I/O 设备融合在一起，而不单独设置接口电路，正如高速 I/O 设备（硬盘驱动器和网卡）中那样。

由于微机体系结构的变化及微电子技术的发展，微机系统所配置的接口的物理结构也发生了变化，以往在微机系统板上能见到的一个个单独的接口芯片，现在集成在一块超大规模的外围芯片中，也就是说原来的那些接口芯片在物理结构上已“面目全非”。但它们相应的逻辑功能和端口地址仍然保留下来，基本上与原来的兼容。因此，尽管微机系统的接口的物理结构发生了变化，但用户在编程时，仍可以照常使用它们。

值得注意的是，尽管外设及接口有了很大的发展，但比起微处理器突飞猛进的发展，差距仍然很大，尤其是数据传输速率方面，还存在尖锐的矛盾。近年来，工业界推出了不少新型外设、总线技术、接口标准及芯片组，正是为了解决系统 I/O 瓶颈问题，相信今后还会出现功能更强大、技术更先进、使用更方便的外设及接口。

CPU、外设及接口在微机系统中所起的作用不同，因而对它们的要求也不一样。例如，8 位数据宽度，基本上可以满足一般工业系统对外设和接口的要求，而微处理器内部数据处理则要求 32 位或 64 位甚至更高。集成度的提高与物理结构上的改变，并不意味着否定接口在逻辑功能上的兼容性。有人说，现在的主板上已找不到分立的接口芯片，它们都已集成到超大规模的接口芯片中去了，因此，现在还讨论单个的接口芯片没有什么意义。而实际上，一些接口电路的逻辑端口及命令格式均得以沿用，初学者最好从基本接口电路开始，充分理解了分立的接口芯片的工作原理、方法及特点之后，才能更好地了解并掌握高集成度接口芯片的工作原理与使用方法。

目前，越来越多的接口设计人员，采用大规模可编程逻辑阵列芯片把多个接口电路集中做在一个芯片中。例如，用一个 FPGA 或 CPLD 芯片设计包含并行接口、串行接口，定时计数器以及 I/O 端口地址译码电路，这些都只是接口电路结构上的变化，而接口的功能与工作原理并未改变。

1.2　接口的分层次概念

从早期 PC 微机发展到现代微机，影响接口变化的主要有两大因素。一是总线结构不同，属于硬件上的变化。早期微机是单总线，只有一级总线，如 ISA 总线；现代微机是多总线，有三级总线，即 Host 总线、PCI 总线、用户总线（如 ISA）。二是操作系统不同，属于软件上的变化。早期微机上运行的是 DOS 系统，现代微机上运行的是 Windows 等操作系统。

这种变化使接口在完成连接和访问设备的任务时产生了根本不同的处理方法，形成了接口的层次概念，把接口分为上层设备接口与下层总线接口两个层次。这大大促进了接口技术的发展，丰富了接口技术的内容。

接口分层次是接口技术在观念上的改变，是接口技术随总线技术的发展而提升的新概念，对全面认识接口技术具有重要意义。在考虑设备与 CPU 连接时，不能停留在过去传统观念上，而必须面向两个不同层次的接口，这是学习现代接口技术与早期接口技术的差别之处。

1.2.1　硬件分层

早期微机采用单级总线如 ISA 总线，设备与 ISA 总线之间只有一层接口。现代微机采用多级总线，总线与总线之间用总线桥连接。例如，PCI 总线与 ISA 总线之间的接口称为 PCI-ISA 桥。因此，除了设备与 ISA 总线之间的那一层设备接口之外，还有总线与总线的接口——总线桥。在这种情况下，作为连接总线与设备之间的接口就不再是单一层次的，就要分层次了。设备与 ISA 总线之间的接口称为设备接口；PCI 总线与 ISA 总线之间的接口称为总线接口。与早期微机相比，现代微机的外设进入系统需要通过两级接口才行，即通过设备接口和总线接口把设备连接到微机系统。

1.2.2　软件分层

早期微机采用 DOS 操作系统，应用程序享有与 DOS 操作系统同等的特权级，因此，应用程序可以直接访问和使用系统的硬件资源。以 Windows 操作系统为例，现代微机在使用 Windows 操作系统时，由于保护机制，不允许应用程序直接访问硬件，在应用程序与底层硬件之间增加设备驱动程序，应用程序通过调用驱动程序去访问底层硬件，把设备驱动程序作为应用程序与底层硬件之间的桥梁。因此，在访问用户新添加的设备时除了写应用程序之外，还要写设备驱动程序。在 Windows 操作系统下，作为操作与控制设备的接口程序就不再是只有单一的应用程序了，程序也要分层次。访问设备的 MS-DOS 程序和 Win32 程序称为上层用户态应用程序，直接操作与控制底层硬件的程序称为底层核心态驱动程序。与早期微机相比，现代微机对外设的操作与控制需要通过两层程序才行，即通过应用程序和设备驱动程序才能访问设备。

1.2.3　接口技术内容的划分

按照接口分层次的概念，不难把接口技术的内容分为两部分：一部分是接口的上层，包括设备接口及应用程序，构成接口的基本内容；另一部分是接口的下层，包括总线接口及设备驱动程序，构成接口的高级内容。这两部分是现代接口技术的整体内容，或者说一个完整的接口是由基

本内容和高级内容构成的。因篇幅原因，本书只讨论接口技术的基本内容。

设备接口（Device Interface），是指I/O设备与本地总线（如ISA总线）之间的连接电路并进行信息（包括数据、地址及状态）交换的中转站。例如，源程序或原始数据要通过数据接口从输入设备送进去，运算结果要通过数据接口向输出设备送出来；控制命令通过命令接口发出去，现场状态通过状态接口取进来，这些来往信息都要通过接口进行变换与中转。这里的I/O设备包括常规的I/O设备及用户扩展的应用系统的接口。可见，设备接口是接口中的用户层的接口，是本书要讨论的内容。

总线桥（Bus Bridge），是实现微处理器总线与PCI总线，以及PCI总线与本地总线之间的连接与信息交换（映射）的接口。这个接口不是直接面向设备，而是面向总线的，故称为总线桥，例如，CPU总线与PCI总线之间的Host桥，PCI总线与用户总线（如ISA）之间的Local桥等。系统中的存储器或高速设备一般都可以通过自身所带的总线桥挂到Host总线或PCI总线上，实现高速传输。

早期的PC微机采用的是单级总线，只有一种接口，即设备接口，所有I/O设备和存储器，也不分高速和低速都通过设备接口挂在一个单级总线（如ISA总线）上。

现代微机采用多总线，出现了设备接口和总线桥两种接口，外设分为高速设备和低速设备，分别通过两种接口挂到不同总线上，使不同速度的外设各得其所，都能在一个微机系统中运行，大大增强了系统的兼容性。正是因为现代微机采用了多总线技术，引出不同总线之间的连接问题，使得现代微机系统的I/O设备和存储器接口的设计变得复杂起来。

1.3 设备接口

1.3.1 设备接口的功能

设备接口是CPU与外界的连接电路，并非任何一种电路都可以叫做接口电路，它必须具备一些条件或功能，才称得上是接口电路。那么，接口应具备哪些功能呢？从完成CPU与外设之间进行连接和传递信息的任务来看，一般有如下功能。

（1）执行CPU命令

CPU对被控对象外设的控制是通过接口电路的命令寄存器解释与执行CPU命令代码来实现的。例如，并行接口82C55A对外设的按位控制命令执行过程。

（2）返回外设状态

接口电路在执行CPU命令过程中，外设及接口电路的工作状态是由接口电路的状态寄存器报告给CPU的。

（3）数据缓冲

在CPU与外设之间传输数据时，主机高速与外设低速的矛盾是通过接口电路的数据寄存器缓冲来解决的。

（4）信号转换

微机的总线信号与外设信号的转换是通过接口的逻辑电路实现的，包括信号的逻辑关系、时序配合及电平匹配的转换。

（5）设备选择

当一个 CPU 与多个外设交换信息时，通过接口电路的 I/O 地址译码电路选定需要与自己交换信息的设备端口，进行数据交换或通信。

（6）数据宽度与数据格式转换

有的外设（如串行通信设备）使用串行数据，要求按照协议的规定，以一定的数据格式传输，如异步通信的起止式数据格式、同步通信的面向字符数据格式等。为此，接口电路就应具有数据并-串转换和数据格式转换的能力。

上述功能并非每种设备接口都要求具备，对不同的微机应用系统，所使用的设备不同，其接口功能不同，接口电路的复杂程度大不一样，应根据需要进行设置。

1.3.2　设备接口的组成

为了实现上述功能，就需要物理基础——硬件，予以支撑；还要有相应的程序——软件，予以驱动。所以，一个能够实际运行的接口，应由硬件和软件两部分组成。

1. 硬件电路

从使用角度来看，接口的硬件部分一般包括以下 3 部分。

（1）基本逻辑电路

包括命令寄存器、状态寄存器和数据缓冲寄存器。它们担负着接收执行命令、返回状态和传送数据的基本任务，是接口电路的核心。目前，可编程大规模集成接口芯片中都包含了这些基本电路，是接口芯片编程模型中的主要对象。若采用 FPGA 自行设计接口电路模块至少也必须包含这几个寄存器。

（2）端口地址译码电路

它由译码器或能实现译码功能的其他芯片，如 GAL（PAL）器件、普通 IC 逻辑芯片构成。它的作用是进行设备选择，是接口中不可缺少的部分。这部分电路不包含在集成接口芯片中，要由用户自行设计。

（3）供选电路

这是根据接口不同任务和功能要求而添加的功能模块电路，设计者可按照需要加以选择。在设计接口时，当涉及数据传输方式时，要考虑中断控制或 DMA 控制器的选用；当涉及速度控制和发声时，要考虑定时/计数器的选用；当涉及数据宽度转换时，要考虑到移位寄存器的选用等。

以上这些硬件电路不是孤立的，而是按照设计要求有机结合在一起，相互联系、相互作用，实现接口的功能。

2. 软件编程

接口软件实际上就是用户的应用程序，由于接口的被控对象的多样性而无一定模式，但从实现接口的功能来看，一个完整的接口控制程序大约包括如下一些程序段。

（1）初始化程序段

对可编程接口芯片（或控制芯片）都需要通过其方式命令或初始化命令设置工作方式、初始条件以及确定其具体用途，这是接口程序中的基本部分。有人把这个工作叫做可编程芯片的“组态”。

（2）传送方式处理程序段

只要有数据传送，就有传送方式的处理。查询方式有检测外设或接口状态的程序段；中断方式有中断向量修改、对中断源的屏蔽/开放以及中断结束等的处理程序段，且这种程序段一定是主

程序和中断服务程序分开编写。DMA 方式有传输参数的设置、通道的开放/屏蔽等处理的程序段。

（3）主控程序段

这是完成接口任务的核心程序段，包括程序终止与退出程序段。如数据采集的程序段，包括发转换启动信号、查转换结束信号、读数据以及存数据等内容。又如步进电机控制程序段，包括运行方式、方向、速度以及起启/停控制等。

（4）辅助程序段

该程序段包括人-机对话、菜单设计等内容。人-机对话程序段能增加人-机交互作用；设计菜单使操作方便。

以上这些程序段是相互依存的，是一体的，只是为了分析一个完整的接口程序而划分成几个部分。

1.3.3 设备接口与 CPU 交换数据的方式

设备接口与 CPU 之间的数据交换，一般有查询、中断和 DMA 三种方式。不同的交换方式对接口的硬件设计和软件编程会产生比较大的影响，故接口设计者对此颇为关心。三种方式简要介绍如下。

（1）查询方式

查询方式是 CPU 主动去检查外设是否“准备好”传输数据的状态，因此，CPU 需花费很多时间来等待外设进行数据传输的准备，工作效率很低。但查询方式易于实现，在 CPU 不太忙的情况下，可以采用。

（2）中断方式

中断方式是 I/O 设备做好数据传输准备后，主动向 CPU 请求传输数据，CPU 节省了等待外设的时间。同时，在外设做数据传输的准备时，CPU 可以运行与传输数据无关的其他指令，使外设与 CPU 并行工作，从而提高 CPU 的效率。因此，中断方式用于 CPU 的任务比较忙的场合，尤其适合实时控制及紧急事件的处理。

（3）DMA 方式

DMA（直接存储器存取）方式是把外设与内存交换数据的那部分操作与控制交给 DMA 控制器去做，CPU 只做 DMA 传输开始前的初始化和传输结束后的处理，而在传输过程中 CPU 不干预，完全可以做其他的工作。这不仅简化了 CPU 对输入/输出的管理，更重要的是大大提高了数据的传输速率。因此，DMA 方式特别适合高速度、大批量数据传输。

1.3.4 分析与设计设备接口电路的基本方法

1. 接口芯片的编程模型方法

接口技术离不开或免不了与各种芯片、器件、设备打交道，这也是有些读者学习接口技术时颇感困难的部分，这主要有以下几点原因。一是硬件基础知识不够，如电子技术、数字逻辑等先行课没有学过或实践太少。二是有畏惧心理，一见到芯片，尤其是复杂的芯片就不知如何下手，觉得很难。遇到这种情况怎么办？首先，下定决心，要学习接口，就无法回避与硬件打交道，要学好接口就要去了解与熟悉相关的硬件知识。其次是不要畏惧硬件技术，其实它和软件技术一样，是完全可以熟悉与掌握的。其三是讲究方法。与接口技术息息相关的微机系统所包括的微处理器、存储器、接口芯片及总线桥，特别是微处理器和总线桥，其内部逻辑结构非常复杂，而且更新换代很快。面对如此庞大而复杂的硬件资源，应采用何种方法来学习，就成为了解与掌握现代接口

技术必须考虑的问题。本书采用编程模型的方法。

编程模型也叫软件模型，是对任何一种硬件对象，如一个接口芯片（不管是复杂的还是简单的），主要是了解、掌握芯片的功能、外部特性和编程使用方法，而不在意其内部结构。芯片的功能是制定接口设计方案时选择芯片的依据，了解了芯片的功能后，就可以知道采用什么样的接口芯片更合适；芯片的外部特性（即芯片引脚的功能与逻辑定义）是接口硬件设计时如何进行连接的依据，了解了外部特性后，就可以知道芯片怎样在系统中进行硬件连接；芯片的编程使用方法是接口软件设计时如何进行编程的依据，了解了编程使用方法后，就可以知道怎样编程来实现芯片的功能。因此，更具体地说，编程模型是指芯片内部可访问的寄存器及其命令、状态，数据格式和分配给寄存器的端口地址 3 个元素。了解与掌握了一个芯片这 3 个方面的内容，也就可以利用它进行接口的软件设计了。

软件模型方法的实质是强调对硬件对象的应用，而不在意其内部结构，这大大简化了对硬件对象复杂结构的了解，而又不失对硬件的应用。因此，本书对接口芯片与外围支持芯片的内部逻辑结构不作介绍。

本书提出的这种从应用的角度了解硬件外部特性和编程用法而不在意内部硬件细节的方法，也是当前硬件系统设计（或硬件系统集成）与分析时常用的方法，接口技术课程更应该如此，因为它与电子线路、数字逻辑或计算机组成原理等课程的教学目的与要求不同，它更注重应用系统，而对系统中的各个模块只关心它的功能、外部特性、连接方法及其编程。因此，在接口技术课程中应该抛弃那种深究芯片内部工作逻辑和硬件细节的做法，把精力放到微机应用系统的构建和芯片的编程上来。

2. 接口两侧分析方法

设备接口是连接 CPU 与 I/O 设备的桥梁。在分析接口设计的需求时，显然应该从接口的两侧入手。CPU 一侧，接口面向的是本地总线的数据、地址和控制三总线，情况明确。因此，主要是接口电路的信号线要满足三总线在时序逻辑上的要求，并进行对号入座连接即可。

I/O 设备一侧，接口所面对的是种类繁多、信号线五花八门、工作速度各异的外设，情况很复杂。因此，对 I/O 设备一侧的分析重点放在两个方面：一是分析被连 I/O 设备的外部特性，即外设信号引脚的功能与特点，以便在接口硬件设计时，提供这些信号线，满足外设在连接上的要求；二是分析被控外设的工作过程，以便在设计接口软件时，按照这种过程编写程序，满足外设工作条件与要求。这样，接口电路的硬件设计与软件编程就有了依据。

3. 硬软结合法

以硬件为基础，硬件与软件相结合是设计设备接口电路的基本方法。

（1）硬件设计方法

对于台式微机的硬件设计主要是合理选用外围接口芯片和有针对性地设计附加电路。目前，在接口设计中，通常采用可编程通用或专用接口芯片，因而需要深入了解和熟练掌握各类芯片的功能、特点、工作原理、使用方法及编程技巧，以便合理地选择芯片，把它们与微处理器正确地连接起来，并编写相应的控制程序。

外围接口芯片并非万能，因此，当接口电路中有些功能不能由接口的核心芯片完成时，就需要用户添加某些电路，予以补充。

（2）软件设计方法

从整体来讲，接口的软件设计，应该包括上层用户程序和底层驱动程序。一般用户只需编写用户层的应用程序，而对原创性开发，就要涉及核心态的设备驱动程序设计。就用户应用程序而