



面向

21世纪

高级应用型人才

中国高等职业技术教育研究会推荐
高职高专系列教材

计算机接口技术

庄琴生 编著

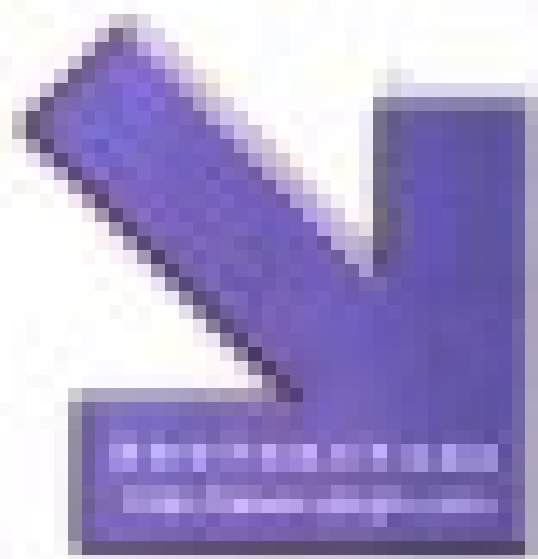
西安电子科技大学出版社
<http://www.xduph.com>



中国计算机学会推荐教材
高等院校计算机专业教材

计算机接口技术

张明华 主编



□ 中国高等职业技术教育研究会推荐

· 高职高专系列教材

计算机接口技术

庄琴生 编著

西安电子科技大学出版社

2004

内 容 简 介

本书的主要内容包括接口技术的基本概念、计算机与外设之间的信息传送方式、总线技术、中断接口技术、定时器/计数器接口、并行接口、串行通信接口、数/模和模/数转换、DMA 技术、标准键盘接口、CRT 显示器接口、AGP 接口、USB 接口、IEEE 1394 接口、IDE 接口、串行 ATA 接口、SCSI 接口等。

本书注重接口技术基本概念的讲解，强调应用性、实用性。书中提供了大量的基本应用实例和具体的实验内容，并对计算机接口的新技术进行了讲解和介绍。

本书主要以 PC 系列微型计算机为背景，适合作为计算机及相关专业的高等职业教育、大专教育、非计算机专业本科教育的教材，也适合作为具有一定计算机基础知识的工程技术人员学习和参考的资料。

★ 本书配有电子教案，需要者可与出版社联系，免费提供。

图书在版编目 (CIP) 数据

计算机接口技术 / 庄琴生编著. —西安: 西安电子科技大学出版社, 2004.6

(高职高专系列教材)

ISBN 7-5606-1391-8

I. 计... II. 庄... III. 电子计算机 - 接口 - 高等学校: 技术学校 - 教材 IV. TP334

中国版本图书馆 CIP 数据核字 (2004) 第 037419 号

策 划 马乐惠

责任编辑 王 瑛 马乐惠

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

<http://www.xduph.com>

E-mail: xdupfxb@pub.xaonline.com

经 销 新华书店

印 刷 陕西华沐印刷科技有限责任公司

版 次 2004 年 6 月第 1 版 2004 年 6 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 16

字 数 374 千字

印 数 1~4000 册

定 价 18.00 元

ISBN 7 - 5606 - 1391-8/TP · 0739 (课)

XDUP 1662001-1

如有印装问题可调换

本社图书封面为激光防伪覆膜，谨防盗版。

序

1999 年以来,随着高等教育大众化步伐的加快,高等职业教育呈现出快速发展的形势。党和国家高度重视高等职业教育的改革和发展,出台了一系列相关的法律、法规、文件等,规范、推动了高等职业教育健康有序的发展。同时,社会对高等职业技术教育的认识在不断加强,高等技术应用型人才及其培养的重要性也正在被越来越多的人所认同。目前,高等职业技术教育在学校数、招生数和毕业生数等方面均占据了高等教育的半壁江山,成为高等教育的重要组成部分,在我国社会主义现代化建设事业中发挥着极其重要的作用。

在高等职业教育大发展的同时,也有着许多亟待解决的问题。其中最主要的是按照高等职业教育培养目标的要求,培养一批具有“双师素质”的中青年骨干教师;编写出一批有特色的基础课和专业主干课教材;创建一批教学工作优秀学校、特色专业和实训基地。

为解决当前信息及机电类精品高职教材不足的问题,西安电子科技大学出版社与中国高等职业技术教育研究会分两轮联合策划、组织编写了“计算机、通信电子及机电类专业”系列高职高专教材共 100 余种。这些教材的选题是在全国范围内近 30 所高职高专院校中,对教学计划和课程设置进行充分调研的基础上策划产生的。教材的编写采取公开招标的形式,以吸收尽可能多的优秀作者参与投标和编写。在此基础上,召开系列教材专家编委会,评审教材编写大纲,并对中标大纲提出修改、完善意见,确定主编、主审人选。该系列教材着力把握高职高专“重在技术能力培养”的原则,结合目标定位,注重在新颖性、实用性、可读性三个方面能有所突破,体现高职教材的特点。第一轮教材共 36 种,已于 2001 年全部出齐,从使用情况看,比较适合高等职业院校的需要,普遍受到各学校的欢迎,一再重印,其中《互联网实用技术与网页制作》在短短两年多的时间里先后重印 6 次,并获教育部 2002 年普通高校优秀教材二等奖。第二轮教材预计在 2004 年全部出齐。

教材建设是高等职业院校基本建设的主要工作之一,是教学内容改革的重要基础。为此,有关高职院校都十分重视教材建设,组织教师积极参加教材编写,为高职教材从无到有,从有到优、到特而辛勤工作。但高职教材的建设起步时间不长,还需要做艰苦的工作,我们殷切地希望广大从事高等职业教育的教师,在教书育人的同时,组织起来,共同努力,编写出一批高职教材的精品,为推出一批有特色的、高质量的高职教材作出积极的贡献。

中国高等职业技术教育研究会会长

李家亮

IT类专业系列高职高专教材编审专家委员会名单

主 任：高 林 （北京联合大学副校长，教授）

副主任：温希东 （深圳职业技术学院电子通信工程系主任，教授）

李卓玲 （沈阳电力高等专科学校信息工程系主任，教授）

李荣才 （西安电子科技大学出版社总编辑，教授）

计 算 机 组：组长：李卓玲(兼) （成员按姓氏笔画排列）

丁桂芝 （天津职业大学计算机工程系主任，教授）

王海春 （成都航空职业技术学院电子工程系副教授）

文益民 （湖南工业职业技术学院信息工程系主任，副教授）

朱乃立 （洛阳大学电子工程系主任，教授）

李 虹 （南京工业职业技术学院电气工程系副教授）

陈 晴 （武汉职业技术学院计算机科学系主任，副教授）

范剑波 （宁波高等专科学校电子技术工程系副主任，副教授）

陶 霖 （上海第二工业大学计算机学院教授）

徐人凤 （深圳职业技术学院计算机应用工程系副主任，高工）

章海鸥 （金陵科技学院计算机系副教授）

鲍有文 （北京联合大学信息学院副院长，副教授）

电子通信组：组长：温希东(兼) （成员按姓氏笔画排列）

马晓明 （深圳职业技术学院电子通信工程系副主任，副教授）

于 冰 （宁波高等专科学校电子技术工程系副教授）

孙建京 （北京联合大学教务长，教授）

苏家健 （上海第二工业大学电子电气工程学院副院长，高工）

狄建雄 （南京工业职业技术学院电气工程系主任，副教授）

陈 方 （湖南工业职业技术学院电气工程系主任，副教授）

李建月 （洛阳大学电子工程系副主任，副教授）

李 川 （沈阳电力高等专科学校自动控制系副教授）

林训超 （成都航空职业技术学院电子工程系主任，副教授）

姚建永 （武汉职业技术学院电子信息系主任，副教授）

韩伟忠 （金陵科技学院龙蟠学院院长，高工）

项目总策划：梁家新

项 目 策 划：马乐惠 云立实 马武装 马晓娟

电 子 教 案：马武装

前 言

“计算机接口技术”是计算机及相关专业高等职业教育、大专教育的重要课程之一。为了适应计算机科学与技术的飞速发展,本书不但注重计算机接口技术的基本概念和基本原理的讲解,而且注重结合具体教学内容对计算机接口的最新技术进行介绍。本书不仅强调基本概念和理论的学习,更注重应用性、实用性和实践性。为了方便学生学习,巩固所学的知识,书中提供了大量的基本应用实例,并结合具体章节安排了相关的实验。

本书共分9章,各章内容安排如下:

第1章讲解计算机接口技术的基本概念,计算机与外设之间的数据传送方式。为了更好地理解本章的内容,本章安排了无条件传送接口的实验。

第2章讲解总线技术。该章主要讲述了总线的功能与分类,以及微机系统中的重要总线:PC/XT总线、ISA总线和PCI总线。着重讲解了PCI总线的结构、信号及总线读/写传输周期。总线技术对于学生来说比较抽象,因此本章安排了I/O地址译码实验来帮助学生理解总线概念。

第3章讲解中断接口技术。结合可编程中断控制器8259A,讲述了Intel公司的支持前端总线为800 MHz的Prescott核心的Pentium 4处理器及Hyper-Treading(超线程)技术的新型865、875系列主板芯片组中的ICH5系列芯片中的中断控制器功能。

第4章讲解定时器/计数器接口电路,主要讲解了8253定时器/计数器电路以及利用8253实现诸如发声和乐曲演奏等应用功能。在本章中,还结合Intel公司的新型865、875系列芯片组,讲解了ICH5系列芯片中的定时器/计数器功能。

第5章讲解并行接口,为了使学生易于理解并行接口技术,书中利用一定的篇幅讲解了简单并行接口电路的实现。本章主要讲解了Intel 8255A并行接口芯片及其应用,其中包括8255A在LED数码管显示和非编码键盘等电路中的应用。

第6章的内容为串行通信及串行通信接口8250。

第7章的内容为模/数转换和数/模转换。

第8章讲解DMA接口技术及Intel公司的新型865、875系列芯片组中的ICH5系列芯片中的DMA控制器功能。

第9章讲解人机交互设备接口与常用标准接口。本章的主要内容有PC系列微型机键盘接口及键盘I/O调用;CRT显示器接口、图形加速卡及AGP总线;并行接口标准Centronics及并行打印机I/O调用;USB接口;IEEE 1394串行总线接口(包括IEEE 1394a和IEEE 1394b);硬盘驱动器接口(包括IDE接口、串行ATA接口以及SCSI接口)。

本书主要以PC系列微型计算机为背景,适合作为计算机及相关专业的高等职业教育、大专教育的教材,也适合作为非计算机专业本科教育的教材,本书还适合作为具有一定计算机基础知识的工程技术人员学习与参考的资料。

在本书的编写过程中得到了西安电子科技大学的温有奎教授、南京工业职业技术学院的李虹老师、清华大学的李桂老师等人的指导和大力帮助,在此表示衷心的感谢。

编者

2004年2月

目 录

第 1 章 计算机接口技术概述	1
1.1 接口技术的基本概念	1
1.1.1 接口的功能与基本结构	1
1.1.2 接口信息与端口的编址方式	3
1.2 计算机与外设之间的数据传送方式	4
1.2.1 无条件传送方式	5
1.2.2 查询传送方式	6
1.2.3 中断传送方式	10
1.2.4 DMA 传送方式	13
1.3 无条件传送接口实验	14
习题 1	16
第 2 章 总线技术	18
2.1 概述	18
2.1.1 总线的功能与分类	18
2.1.2 总线的组成	19
2.1.3 总线的传输过程和传输方式	20
2.2 PC/XT 总线与 ISA 总线	23
2.2.1 PC/XT 总线	23
2.2.2 ISA 总线	26
2.3 PCI 局部总线	28
2.3.1 PCI 总线的结构与特点	28
2.3.2 PCI 总线信号定义	29
2.3.3 PCI 总线周期简介	34
2.4 其它微机总线简介	37
2.4.1 MCA 总线	37
2.4.2 EISA 总线	37
2.4.3 VL 总线	37
2.4.4 PCMCIA 总线	38
2.4.5 STD 总线	38
2.5 I/O 地址译码实验	38
习题 2	40
第 3 章 中断接口技术	41
3.1 中断的基本概念	41
3.1.1 中断的概念	41

3.1.2 中断的优先级与中断嵌套	41
3.2 80x86 系列微型计算机的中断系统	42
3.2.1 中断的分类	42
3.2.2 中断矢量表	44
3.2.3 8086/8088 中断响应过程	45
3.3 Intel 8259A(PIC)可编程中断控制器	46
3.3.1 Intel 8259A 的功能与结构	46
3.3.2 8259A 的中断过程	49
3.3.3 8259A 的初始化命令字和工作方式命令字	50
3.3.4 8259A 的初始化编程举例	55
3.3.5 主程序与中断服务程序的关系	57
3.4 ICH5 中的可编程中断控制器功能	58
3.4.1 ICH5 系列芯片与 Intel 865、875 系列芯片组	58
3.4.2 ICH5 中的 8259 中断控制器功能	59
习题 3	60
第 4 章 定时器/计数器接口电路	62
4.1 基本概念	62
4.1.1 定时与计数	62
4.1.2 定时方法简介	62
4.2 可编程定时器/计数器	63
4.2.1 Intel 8253 的功能与结构	63
4.2.2 8253 的控制字	64
4.2.3 8253 的工作方式	66
4.2.4 8253 的编程	71
4.2.5 8253 的应用	72
4.3 ICH5 中的定时器/计数器功能	82
4.3.1 Intel 8254 定时器/计数器	82
4.3.2 ICH5 中的 8254 定时器/计数器	83
4.4 可编程定时器/计数器 8253 实验	83
习题 4	85
第 5 章 并行接口	86
5.1 并行接口概述	86
5.1.1 概述	86
5.1.2 简单并行接口	88
5.2 可编程并行接口芯片 Intel 8255A	90
5.2.1 Intel 8255A 的结构与功能	90
5.2.2 Intel 8255A 的控制字	93
5.2.3 Intel 8255A 的工作方式	95
5.3 8255A 的编程与应用	101

5.3.1	8255A 初始化编程举例	101
5.3.2	8255A 应用举例	103
5.3.3	8255A 在 LED 数码管显示电路中的应用	105
5.3.4	8255A 在非编码式键盘电路中的应用	112
5.4	实验	120
5.4.1	可编程并行接口 8255 方式 0 实验	120
5.4.2	8255 与七段数码管实验	121
5.4.3	交通灯控制实验	122
习题 5		124
第 6 章	串行通信接口	125
6.1	串行通信概述	125
6.1.1	串行通信的特点	125
6.1.2	串行通信的原理	126
6.1.3	EIA RS-232C 串行接口标准	128
6.2	可编程串行通信接口 8250	131
6.2.1	8250 的功能与结构	131
6.2.2	8250 中的寄存器	133
6.2.3	8250 的编程	138
6.2.4	8250 的应用举例	140
6.3	8250 串行通信实验	145
习题 6		147
第 7 章	模/数(A/D)与数/模(D/A)转换	149
7.1	模/数(A/D)转换	149
7.1.1	模/数转换器的原理与主要技术指标	149
7.1.2	模/数接口芯片 ADC0809	150
7.1.3	ADC0809 与系统的接口及应用	153
7.1.4	16 位模/数转换芯片 ADC1143 简介	158
7.2	数/模(D/A)转换	160
7.2.1	数/模转换的工作原理	160
7.2.2	数/模转换器的主要技术指标	161
7.2.3	数/模接口芯片 DAC0832	162
7.2.4	DAC0832 与系统的接口及应用	164
7.3	实验	167
7.3.1	模/数转换器实验	167
7.3.2	数/模转换器实验	169
习题 7		171
第 8 章	DMA 接口技术	172
8.1	DMA 概述	172
8.1.1	DMA 的作用	172

8.1.2 DMA 传送的工作原理	172
8.2 DMA 控制器 Intel 8237A 简介	173
8.2.1 8237A 的工作周期	173
8.2.2 8237A 的内部结构与引脚功能	174
8.2.3 8237A 的工作模式与传送类型	176
8.2.4 8237A 的寄存器组	177
8.2.5 8237A 的应用	182
8.3 ICH5 中的 DMA 控制器功能	183
习题 8	185
第 9 章 人机交互接口与常用标准接口	186
9.1 PC 系列微机键盘接口	186
9.1.1 键盘接口的工作原理	186
9.1.2 键盘 I/O 操作	189
9.2 CRT 显示器接口	191
9.2.1 CRT 显示器接口的基本原理	191
9.2.2 图形加速卡与 AGP 总线	193
9.3 并行打印机接口	196
9.3.1 并行接口标准 Centronics	196
9.3.2 并行接口的逻辑结构	198
9.3.3 并行打印机 I/O 调用	200
9.4 USB 接口	201
9.4.1 USB 接口概述	201
9.4.2 USB 总线拓扑结构与系统连接	204
9.4.3 USB 的应用	207
9.5 IEEE 1394 串行总线接口	208
9.5.1 IEEE1394 总线的拓扑结构与主要特性	209
9.5.2 IEEE 1394 和 USB 的比较	211
9.6 硬盘驱动器接口	212
9.6.1 IDE 接口	213
9.6.2 IDE 接口的类型	214
9.6.3 IDE 接口的传输模式	215
9.6.4 Serial ATA 硬盘接口	216
9.6.5 SCSI 接口	218
习题 9	223
附录 A 计算机接口实验	225
附录 B 汇编程序的使用方法	242
附录 C 国内外常用二进制逻辑元件图形符号对照表	245
参考文献	246

1.1 接口技术的基本概念

在计算机系统中除了 CPU 以及基本硬件之外, 还有各种外部设备, 比如显示器、键盘、打印机、磁盘驱动器、光盘驱动器等。这些外部设备需要与 CPU 进行大量的信息交换, 根据 CPU 的要求进行各种操作。进行信息交换时, 由于各种原因(比如电平、时序等的不同), 这些外部设备不能与 CPU 直接相连, 它们之间的信息交换必须经过一个中间电路, 也就是接口电路才能实现。

外部设备多种多样, 可以是机械式、电子式或其它形式的; 输入/输出的信息不相同, 可以是数字量、模拟量或开关量; 信息传输的速度也不相同, 键盘输入的速度是秒数量级, 而磁盘输入的速度最高可达百兆位/秒以上。

另外, 与 CPU 进行交换的信息的来源也不相同, 它们可以来自于计算机的系统硬件(比如键盘、显示器、磁盘机等), 也可以来自于外部控制系统或现场采集到的信息。所以, CPU 与外设之间的信息交换是个复杂的问题, 这些信息的交换必须经过接口电路的协调才能实现。

目前, 在微型计算机系统中涉及到计算机系统硬件的接口电路很多, 比如键盘接口、磁盘机接口、打印机接口等已被集成到微型计算机的主板上, 而其它外部设备的接口常常制成适配卡(或称扩展板)插入到总线插槽中, 以便实现所需要的各种功能。

1.1.1 接口的功能与基本结构

接口(Interface)是两个部件之间的连接点或界面。计算机接口技术就是研究 CPU 如何与外部设备进行连接、耦合、匹配, 以便实现 CPU 与外部设备之间高速、可靠地进行信息交换的一门技术, 是计算机系统应用的关键课题之一。从某种意义上讲, 接口电路好像是一个转换器, 它把一种部件的信息转换成另一种部件所能接受的信息。在转换过程中要进行传输速率与时序的转换、信息表示方法的转换以及电平的转换。

1. 接口应具有的功能

1) 对外部设备的寻址功能

计算机系统中往往有多种类型的外部设备, 相同类型的外部设备也可能不止一台。CPU 在同一时刻只能与一台外设通信(当然可以通过分时、轮询等, 实现逻辑上的并行操作)。接口要能够根据 CPU 送出的地址信号及相关的控制信号做出正确的响应, 对 CPU 送出的地址

信号进行译码，根据 CPU 送出的地址选中所需的外设而进行相关的操作。

2) 信号转换功能

外部设备的信号电平可能不是 TTL 电平或 CMOS 电平，因此需要由接口电路来完成信号电平的转换。如果外设传送的是模拟信号，则需要接口电路进行模/数(A/D)或数/模(D/A)转换。如果外设采用的是串行数据传送方式(由于计算机内部是以并行数据传送方式工作的)，则接口电路需要具有并—串或者串—并转换的功能，如此等等。根据具体需要，接口要具有对信号电平、格式或类型等进行转换的功能。

3) 数据缓冲功能

外部设备的工作速率远远低于 CPU 的速率，为了提高 CPU 的工作效率并避免丢失数据，接口中必须有数据缓冲器。数据缓冲器作为数据交换的中转站，可以缓解计算机与外设的速度差异而引起的冲突。

4) 联络功能

接口应当具备握手信号或者能够提供状态信息。CPU 送来的控制信号、外部设备的工作状况以及应答信号都要通过接口与 CPU 以“握手联络”的方式进行交互。接口的联络功能可以保证数据传输正常有序地进行，并可用来检测外设是否工作于正常状态。

5) 中断管理功能

CPU 与外设之间的通信采用中断方式，有利于提高 CPU 的利用率。如果采用中断方式，则外设要申请得到 CPU 的服务，接口就应该具有发送中断请求信号和接收中断响应信号的功能；如果接口要处理多个外设的中断，则还应该具有中断屏蔽、中断优先级管理等功能。

6) 可编程功能

为了使接口具有较强的灵活性、可扩充性以适应多种工作方式或工作状态，接口应具有可编程的特性。现在大多数接口芯片都是可编程的，只需通过程序的修改或软件的设置就可以改变接口的工作方式，以满足不同的需求。

2. 接口电路的基本结构

接口电路的基本结构如图 1-1 所示。在与 CPU 相连的一侧，接口电路通过地址总线 AB、数据总线 DB 以及控制总线 CB 与 CPU 连接。地址总线 AB 用来提供访问接口电路的地址信息。数据总线 DB 用来与接口电路交换数据信息、状态信息和命令信息。控制总线 CB 包括 $\overline{IO/M}$ 、 $\overline{RD}$ 、 $\overline{WR}$ 、INTR 等信号线。 $\overline{IO/M}$ 用于确定是进行存储器访问还是 I/O

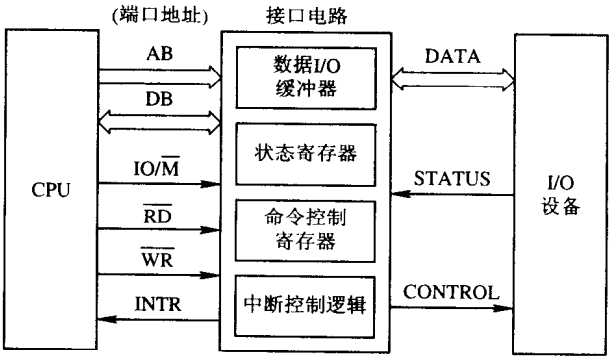


图 1-1 接口电路的基本结构

访问,当 $\overline{\text{IO}/\text{M}}$ 为高电平时表明是对接口电路(即对外部设备)的操作。 $\overline{\text{RD}}$ 是 CPU 发出的读命令信号,低电平有效,用于对接口电路进行读操作,CPU 在执行输入指令 IN 时会自动发出低电平的 $\overline{\text{RD}}$ 命令信号。 $\overline{\text{WR}}$ 是 CPU 发出的写命令信号,低电平有效,CPU 在执行输出指令 OUT 时会自动发出低电平的 $\overline{\text{WR}}$ 命令信号。INTR 线可用于外部设备通过接口电路向 CPU 提出中断申请。

在与 I/O 设备连接的一侧,接口电路通过数据线 DATA、状态线 STATUS、控制线 CONTROL 与 I/O 设备相连。数据线 DATA 用于在接口电路与外部设备之间进行数据交换。状态线 STATUS 反映外部设备的当前状态,CPU 或接口电路可根据状态线的当前状态决定下一步的操作。控制线 CONTROL 可用于对外部设备进行特定的控制。

接口电路处于 CPU 与外部设备之间,是 CPU 与外部设备之间进行通信的桥梁。接口电路内部一般包括数据 I/O 缓冲器、状态寄存器、命令控制寄存器和中断控制逻辑等部分。

进行输出操作时,数据 I/O 缓冲器用于暂存 CPU 输出的数据;进行输入操作时,数据 I/O 缓冲器用于暂存外部设备输入的数据。所谓输入或输出都是以 CPU 为参照点的,CPU 送出为输出,CPU 读入为输入。数据 I/O 缓冲器用于匹配 CPU 与外部设备之间因速率、时序等不同而引起的差异。状态寄存器用于寄存外部设备的状态,并把此状态提供给 CPU 或接口电路,以便 CPU 确定下一步如何进行操作。命令控制寄存器一般用于接收 CPU 送来的命令或控制字,命令或控制字可以控制外部设备的各种操作。

1.1.2 接口信息与端口的编址方式

1. CPU 与外设之间的接口信息

CPU 与外设之间传送的信息通常包括数据信息、状态信息和控制信息。

1) 数据信息

数据信息是最基本,也是最重要的一种信息,它包括数字量、模拟量和开关量。

数字量:通常为 8 位、16 位或 32 位的二进制数据或 ASCII 码数据。

模拟量:当计算机用于控制系统或数据采集系统时,大量的现场信息(比如温度、压力、流量和速度等)是连续变化的物理量。这些物理量经过传感器被转换成电信号,再经过放大、整形等处理后得到电流或电压,这些电流或电压仍旧是模拟量。计算机不能直接接收和处理这些模拟量,必须经过模/数转换,转换成数字量才能输入计算机;而计算机输出的数字量有时也需要经过数/模转换,转换成模拟量才能控制相关的外部设备。

开关量:是一些只有两个状态的量,如开关的闭合与断开、继电器触点的打开与吸合等,用一位二进制数“0”或“1”表示。一个字节的二进制数据可以表示八个这样的量。字长为 8 位的微机一次最多可输入或输出八个开关量。

2) 状态信息

状态信息反映当前外部设备所处的工作状态。输入时,CPU 常要先查询输入设备是否准备好(Ready=1?),准备好了才可以输入;输出时,CPU 常要查询输出设备是否空闲(Busy=0?),如果输出设备不忙(Busy=0),则 CPU 向该外部设备输出数据。CPU 根据接口电路提供的输入/输出设备的状态信息,决定是否进行数据的输入或输出。

3) 控制信息

控制信息是控制输入/输出设备启动、停止或执行某种特定操作的信息。

综上所述,接口电路传送三种不同类型的信息,即数据信息、状态信息和控制信息。三种信息的性质不同,要通过不同的端口分别传送。数据 I/O 寄存器(缓冲器)、状态寄存器和命令控制寄存器各占一个端口,每个端口都有自己的端口地址,用不同的端口地址来分别传送数据信息、状态信息和控制信息。一个端口的寄存器往往是 8 位的,通常一个外部设备的数据端口也是 8 位的,由于状态与控制端口往往只使用其中的一位或两位,因此有时不同外部设备的状态和控制信息也可以共用一个端口。

2. 端口的编址方式

输入/输出端口(即 I/O 端口)的编址方式主要有两种:内存和 I/O 端口统一编址及 I/O 端口单独编址。

1) 内存和 I/O 端口统一编址

I/O 端口和存储单元统一编址是在整个存储空间中划分出一部分地址空间给外设端口使用。CPU 访问 I/O 端口时所使用的指令与访问存储器时所使用的指令形式完全一样,仅根据访问的地址范围的不同来区分两种不同的操作。在这种方式中,I/O 端口和存储器占有不同的地址空间。比如,PDP-11 和 6800、6502 等系列的微机就采用这种方式。

I/O 端口和存储单元统一编址方式的优点在于 I/O 端口的地址空间较大,对端口进行操作的指令功能较强,使用时灵活方便。缺点是由于 I/O 端口的地址也要占用整个地址空间的一部分,减少了内存单元的有效地址空间,I/O 操作指令执行的时间较长,由于 I/O 操作指令和访问存储器指令的形式完全相同,因此程序的可读性较差,端口地址译码器较复杂。

2) I/O 端口单独编址

I/O 端口的地址空间单独编址,与内存单元的地址空间相互分开,各自独立。8086/8088 系统中就采用这种编址方式。在这种方式下 CPU 要使用专门的输入/输出指令,例如,使用 IN 指令、OUT 指令来访问 I/O 端口,另外还要提供区分是存储器读/写还是 I/O 端口读/写的控制信号。

8086/8088 系统的 8 位 I/O 端口地址空间是 0~64 KB,地址总线中 $A_{15} \sim A_0$ 就可提供寻址 64 KB I/O 端口地址空间的能力。8088 CPU 的控制信号 $IO/\overline{M}$ 用来区别是内存单元的访问还是 I/O 端口的访问,所以地址空间 0~64 KB 不仅可以供 I/O 端口使用,还可以供存储器使用。

I/O 端口单独编址方式的优点是:不占用内存单元的有效地址空间,地址译码器较简单,端口操作指令长度较短,执行速度较快。由于端口操作指令与存储器操作指令不同,因此使得程序的可读性较好。缺点是输入/输出指令功能不强,一般除了进行传送操作以外无其它附加功能。

1.2 计算机与外设之间的数据传送方式

计算机与外设之间的数据传送方式可分为无条件传送方式、查询传送方式、中断传送方式和 DMA 传送方式四种。其中,无条件传送方式、查询传送方式和中断传送方式是在 CPU 的控制之下进行的,数据传送时由 CPU 掌握总线的控制权;而 DMA 传送方式则是在 DMA 控制器的控制之下进行的,数据传送时由 DMA 控制器掌握总线的控制权。

1.2.1 无条件传送方式

无条件传送方式是最简单的数据传送方式。在这种传送方式中，假设外设已处于就绪状态。无论何时，进行输入操作都要求输入设备的输入数据必须是准备好的；进行输出操作都要求输出设备不仅处于正常的工作状态而且必须空闲。数据传送时，程序不再查询外设的状态，直接用输入/输出指令进行数据传送。

图 1-2 为无条件传送方式输入/输出接口示意图。图中经地址总线把选中外设的地址信息送往端口译码器，当 $\text{IO}/\overline{\text{M}}$ 为高电平时可以进行输入/输出操作。当 $\overline{\text{WR}}$ 为低电平时进行输出操作，当 $\overline{\text{RD}}$ 为低电平时进行输入操作。当外设为输入设备时，要求输入数据的保持时间相对于 CPU 的处理时间长许多，所以输入设备使用三态输入缓冲器与系统数据总线相连。当外设为输出设备时，由于外设的速率相对于 CPU 来说慢得多，CPU 送出的数据应在接口中保持一段时间，以适应外设的操作，因此输出电路中使用了输出锁存器。

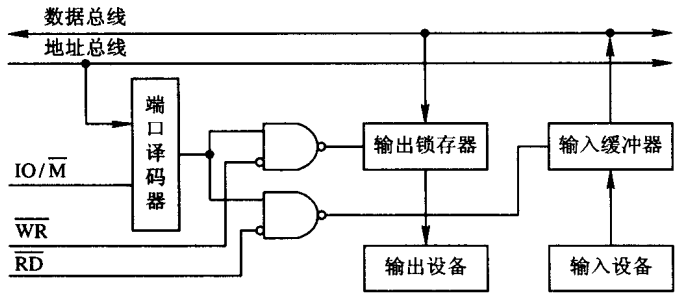


图 1-2 无条件传送方式输入/输出接口示意图

例 1.1 一个无条件传送输入和输出的具体实例如图 1-3 所示。图中八个开关可以看成是输入设备，八个发光二极管(LED)可以看成是输出设备，用集成电路芯片和与非门等组成接口电路。输入设备和输出设备具有相同的端口地址，通过 $\overline{\text{RD}}$ 和 $\overline{\text{WR}}$ 来区别是进行输入还是输出操作。假设输入设备和输出设备的端口地址为 218H。要求根据八个开关的通断状态点亮或熄灭对应的发光二极管，开关可以根据需要随时拨动。

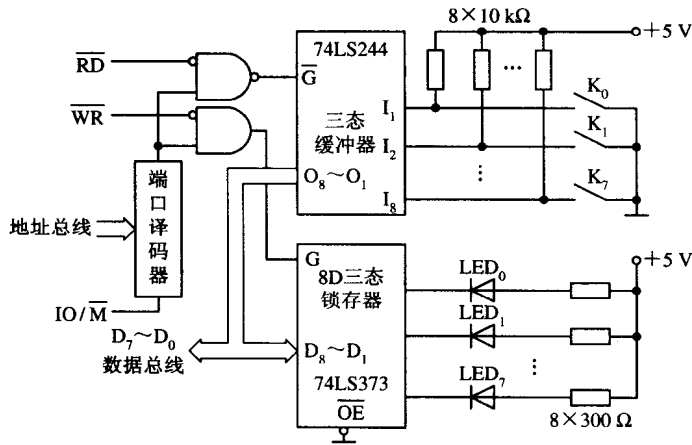


图 1-3 开关状态读取与 LED 显示接口电路

解 作为输入设备的八个开关连接由三态缓冲器(三态门)74LS244 构成的输入口。八个开关的另一端都接地,同时 74LS244 的输入端 $I_1 \sim I_8$ 通过电阻接到高电平。当 CPU 选通 74LS244 时,可读取各开关的状态。某开关打开时读入的值是 1,闭合时读入的值是 0。无论何时,CPU 经数据总线 $D_7 \sim D_0$ 读取各开关的状态,各开关实际都处于就绪状态。CPU 执行 IN 指令,使 $\overline{IO/\overline{M}}$ 为高电平,经端口译码器选中端口地址 218H,并使 $\overline{RD}$ 为低电平,经 74LS244 可以读入八个开关的状态。

8D 三态锁存器 74LS373 构成输出口,连接到作为输出设备的八个发光二极管(LED)。八个发光二极管的另一端都经电阻连接到高电平。CPU 选通 74LS373,经数据总线 $D_7 \sim D_0$ 把输出数据送到输出锁存器 74LS373 的 $D_8 \sim D_1$ 。74LS373 把数据总线送来的数据锁存,当 74LS373 的输出端某位为 0 时,使对应的 LED 被点亮;某位为 1 时,使对应的 LED 熄灭。无论何时,CPU 经数据总线 $D_7 \sim D_0$ 向八个发光二极管输出数据时,作为输出设备的八个发光二极管都处于空闲状态。CPU 执行 OUT 指令,使 $\overline{IO/\overline{M}}$ 为高电平,经端口译码器选中端口地址 218H,并使 $\overline{WR}$ 为低电平,经 74LS373 输出数据,可以点亮或熄灭发光二极管。

程序的执行过程是:CPU 读取各开关的状态数据,然后把读入的开关状态数据输出到 74LS373;重复执行上述过程以满足八个开关可以随时拨动的需要。程序如下:

```
LOP: MOV  DX, 218H      ; DX 指向端口地址
      IN   AL, DX        ; 从输入口读取开关状态
      OUT  DX, AL        ; 把读入的开关状态数据送到输出口显示
      CALL DELAY         ; 延时 10 ms 的子程序
      JMP  LOP
```

程序中 DELAY 是延时 10 ms 的子程序,未单独列出。

无条件传送方式的优点是程序编写和接口电路的设计都较为简单。缺点是由于无条件传送方式中程序不查询外设的状态,直接用输入/输出指令进行数据传送,因此在外设不处于就绪状态或外设发生故障时就会产生数据错误或数据丢失。并且,此时 CPU 还难以发现错误。所以这种传送方式使用并不广泛。

1.2.2 查询传送方式

无条件传送方式中程序不查询外设的状态,直接用输入/输出指令进行数据传送,在外设不处于就绪状态或外设发生故障时就会产生数据错误或数据丢失。为了解决这一问题,人们考虑采用查询传送方式。在查询传送方式中,CPU 与外设之间采用应答方式进行数据交换。CPU 不断地查询外设的状态,如果外设处于就绪状态,CPU 就可以和外设进行数据交换。CPU 与外设之间需要经过两个端口(即状态端口和数据端口)进行交互。CPU 执行程序,不断地通过读取状态端口的信息来测试外设的状态,当输入设备处于准备好状态或输出设备处于空闲状态时,CPU 就能执行输入或输出指令与外设交换数据。

1. 查询输入

查询输入的工作流程如图 1-4 所示。CPU 读入输入设备的状态信息,判断输入设备是否准备好,如果输入设备没有准备好,则继续查询输入设备的状态,直至输入设备准备好后才输入数据。