

微型计算机硬件技术及应用基础

(下册 接口与应用)

主编 邹逢兴

(下册 接口与应用)

微 型 计 算 机

硬件技术及应用基础

主编 邹逢兴 ○国防科技大学出版社

P340.3

Z96

2

字出版

微型计算机 硬件技术及应用基础

(下册:接口与应用)

主编:邹逢兴

编著:邹逢兴 何智勇 史美萍

国防科技大学出版社

· 长沙 ·

图书在版编目(CIP)数据

JS140.106

微型计算机硬件技术及应用基础(下册:接口与应用)/邹逢兴;何智勇,史美萍. —长沙:国防科技大学出版社,1997. 10

ISBN 7-81024-449-3

- I. 微型计算机硬件技术及应用基础(下册:接口与应用)
- II. 邹逢兴;何智勇,史美萍
- III. ①计算机 ②硬件 ③教材
- IV. TP36

责任编辑:卢天贶

责任校对:何 晋

封面设计:陆荣斌

国防科技大学出版社出版发行

电话:(0731)4505241 邮政编码:410073

新华书店总店北京发行所经销

长沙交通学院印刷厂印装

开本:787×1092 1/16 印张:14.5 字数:335 千

1997年10月第1版第1次印刷 印数:1—3000册

ISBN 7-81024-449-3

TP·94 定价:17.00元

内容简介

本书从计算机应用的需要出发,以 PC/486 为主要背景机,系统介绍了微型计算机的硬件技术及应用基础。全书分上、下两册。上册为微机原理篇,主要介绍微型计算机及其基本组成部分—中央处理器、存储器、输入输出接口的硬件结构与工作原理,以及总线与总线标准、中断与 DMA 机制、基于 80486 指令系统的汇编语言程序设计基础等内容。下册为接口与应用篇,着重介绍微机系统中各种常见 I/O 设备的接口原理与技术,常用接口芯片,386/486 微机系统接口等内容和微型机在测控系统中的应用。全书注重了选材的科学性、先进性和实用性,贯彻了模块化、结构化和原理、技术与应用并重的内容组织原则。

本书除可作为高等院校各理工科专业的本科教材外,也可作为研究生和各类计算机应用培训班的教材以及专科教学参考书,还可供从事微机应用开发工作的科技人员参考。

前 言

计算机技术的飞速发展和计算机应用的日益普及,对高等学校非计算机专业的计算机教育提出了越来越高的要求。计算机技术水平的高低和计算机应用能力的强弱,已成为今天衡量大学毕业生质量、水平的重要尺度。为此,全国各高等院校,特别是重点院校,正在国家教委的统一规划下,把非计算机专业的计算机基础教育放在十分重要的地位来抓,把普遍提高学生的计算机应用和操作能力作为一项重要工作,制定目标,强化措施,加大投入,促使计算机教育质量尽快上一个新台阶。

本书正是在这种计算机教育的大好形势下,为贯彻国家教委“面向 21 世纪教学内容改革”的精神,为满足高等学校工科本科计算机教学对新教材的迫切需求而编著的。

对于广大计算机应用人员来说,今天面对的主要是微型计算机。而当今在计算机世界占有绝对优势的主流微机是以 Intel 80×86 为 CPU 的微型计算机系统,其中又以 486 微机系统更具代表性、典型性和上下兼容性。因此,本书从计算机应用的需要出发,以 PC/486 为主要背景机,系统介绍了微型计算机的硬件技术及应用基础。全书分上、下两册。上册为微机原理篇,包括微机系统组成及工作原理、中央处理器和指令系统、汇编语言程序设计基础、总线与总线标准、存储器、输入输出接口、中断与中断控制、DMA 与 DMA 控制等 8 章内容,是全书的主体和基础。下册为微机接口与应用篇,在原理篇中 I/O 接口基本原理的基础上,进一步介绍了并行接口与串行接口、基本人机交互接口、模拟 I/O 接口、计数器/定时器电路及接口、386/486 微机系统接口等内容和最能反映计算机硬件技术本质的微型机在测控系统中的应用。根据非计算机专业学习、应用计算机的特点,本书总的组织、编写原则是:适当淡化微机内部和芯片内部原理,强化外部接口和应用。这里内部和外部的分界是“总线”,因此,把总线连接技术又作为接口和应用的关键。各章节力求贯彻原理、技术与应用并重,理论与实践结合的原则,注重选材的科学性、先进性和实用性;内容组织力求符合教学规律,由浅入深,循序渐进;论述尽量深入浅出,富有启发性。为便于教学和读者自学,各章后面均有适当思考题与习题。

鉴于工科非计算机类专业很多,不同专业差别很大,即使是同类专业,不同院校的教学要求也不尽相同,本书在编写时遵循了“宽编窄用”的内容选取原则和模块化、结构化的内容组织原则,以具有较宽的适应面和灵活的选择余地,利于实施不同对象、不同层次、不同学时数

的教学。本书参考学时数为 60~100。对于 60 学时左右的教学,可只讲上册(微机原理篇);对于 80 学时左右的教学,可讲原理和接口;对于 100 学时左右的教学,可全书都讲。当然,根据各自具体情况和要求,也可作其它取舍,例如,可全书都讲,但目录中打“*”号的内容不讲。不管哪种取舍,都应把实验、上机放在重要的地位。

本书由邹逢兴主编,并编写了第一、四、五、六、七、八、九、十、十一、十二、十三、十四章;第二、三章由何智勇编写,邹逢兴修改。书中大部分插图由史美萍用计算机设计和绘制。

本书从编写到出版,自始至终得到国家教委理科计算机教学指导委员会委员、国防科技大学副校长齐治昌教授,国家教委工科计算机教学指导委员会副主任委员、国防科技大学教务部副教务长邹鹏教授,国务院学位委员会计算机科学与技术学科评议委员、电子工业部计算机教学指导委员会主任陈火旺教授和国家教委工科计算机教学指导委员会副主任委员麦中凡教授等知名计算机专家的指教。在此,对他们表示衷心感谢。吴盘荣、王淑英、杨心强、曹兰斌、秦恒玉、刘孟仁、朱参世、李续武、赵运轩、窦文华、王广芳、薛啸宇、董宏俊、陈文伟、陈怀义等校内外同行专家教授,应邀对本书的编写指导思想、内容选取和编写大纲进行了多次审查,提出了许多宝贵意见。尤其是国防科大计算机系王风学教授,对书稿进行了十分认真细致的审阅,提出了许多极其宝贵的真知灼见和修改建议。在此对他们一并致以深深谢意。

本书尽管是作者长期从事微机应用方面教学与科研的经验结晶,但由于水平和经验有限,加上它结构新、内容新,错误一定难免,敬请使用本教材的同行专家和读者不吝赐教,批评指正。

邹逢兴

1997 年 8 月于湖南长沙国防科大

下册提要

学习微型计算机原理的目的是为了应用。下册(微机接口与应用篇)正是从实用出发,在上册(微机原理篇)中 I/O 接口基本概念、原理的基础上,进一步介绍了一些常用接口的原理、方法和技术,包括并行接口与串行接口、基本人机交互接口、模拟 I/O 接口、计数器/定时器电路及接口、386/486 微机系统接口等五章内容,并以最能反映计算机硬件特征的微机在测控系统中的应用为背景,专列一章,介绍了微机应用的一般方法、步骤和相关关键技术。各章基本上是先讲述有关接口或应用原理,再介绍实现这些原理的典型方法或典型芯片,最后给出这些方法或芯片的一、二个应用举例,使读者有机会从理论和实践的结合上学习、理解有关原理与技术。鉴于非计算机专业学习、应用计算机的特点,本书总的组织、编写原则是适当淡化微机内部和芯片内部原理,强化外部接口和应用。而内部和外部的分界是各级总线,所以总线连接技术是接口和应用的关键,在学习下册时要注意抓住这个关键。对于微机或芯片内部,重点是了解其功能和基于功能的编程。

通过对下册的学习,要求读者不仅对各类微机接口的基本原理与技术有较深刻的理解和认识,而且能正确选用各种专用/通用接口电路芯片来组成不同功能的微机应用系统,能自行设计微机应用和开发中可能遇到的一些常见 I/O 接口的硬件和软件,为独立地从事微机应用工作奠定良好的基础。

目 录

第九章 并行接口与串行接口

§ 9.1 并行接口与串行接口概述	(1)
§ 9.2 可编程并行接口芯片 8255A-5	(2)
§ 9.2.1 内部结构与引脚功能	(2)
§ 9.2.2 控制字的确定和初始化编程	(5)
§ 9.2.3 三种工作方式和接口方法	(7)
§ 9.3 串行通信的基本概念	(16)
§ 9.3.1 数据传送方式	(16)
§ 9.3.2 波特率和收/发时钟	(17)
§ 9.3.3 信号的调制和解调	(18)
§ 9.3.4 误码率和串行通信中的差错控制	(20)
§ 9.3.5 通道的多路复用	(21)
§ 9.3.6 串行通信的基本方式	(23)
§ 9.3.7 串行通信的实现方法	(23)
§ 9.4 异步串行通信标准接口	(24)
§ 9.4.1 异步串行通信协议及标准接口	(24)
§ 9.4.2 RS-232-C 标准接口	(26)
§ 9.4.3 10mA(60mA)电流环标准接口	(31)
§ 9.5 可编程异步串行接口芯片 INS8250	(33)
§ 9.5.1 内部结构与引脚功能	(33)
§ 9.5.2 INS8250 的编程	(41)
§ 9.6 PC 系列微机中的串行通信接口	(42)
§ 9.6.1 串行通信接口硬件逻辑	(43)
§ 9.6.2 异步通信 I/O 功能及调用	(45)
§ 9.6.3 异步串行通信的实现	(47)
思考题与习题	(51)

第十章 基本人机交互接口

§ 10.1 键盘接口	(54)
§ 10.1.1 键盘与键盘接口原理	(54)
§ 10.1.2 PC 系列机键盘及接口	(56)

§ 10.2 LED 显示器接口	(60)
§ 10.2.1 LED 显示器及显示原理	(60)
§ 10.2.2 一位LED 显示器接口	(62)
§ 10.2.3 多位LED 显示器接口	(62)
§ 10.3 CRT 显示器接口	(65)
§ 10.3.1 概述	(65)
§ 10.3.2 CRT 显示器及显示原理	(66)
§ 10.3.3 CRT 控制器接口	(67)
§ 10.4 打印机接口	(70)
§ 10.4.1 打印机及打印控制原理	(70)
§ 10.4.2 打印机接口方法	(71)
§ 10.4.3 PC 系列微机的打印机接口适配器	(75)
§ 10.5 鼠标器接口	(79)
思考题与习题	(80)

第十一章 模拟 I/O 接口

§ 11.1 DAC 及其与 MPU 的接口	(83)
§ 11.1.1 D/A 转换器原理	(83)
§ 11.1.2 DAC 的基本参数	(87)
§ 11.1.3 典型的 DAC 集成芯片	(88)
§ 11.1.4 DAC 芯片与 MPU 的接口技术	(93)
§ 11.2 ADC 及其与 MPU 的接口	(96)
§ 11.2.1 A/D 转换的四个步骤	(96)
§ 11.2.2 A/D 转换器原理	(97)
§ 11.2.3 ADC 的性能参数	(99)
§ 11.2.4 典型的集成 ADC 芯片	(100)
§ 11.2.5 ADC 芯片与 MPU 的接口技术	(109)
思考题与习题	(114)

第十二章 计数器/定时器电路及接口

§ 12.1 概述	(116)
§ 12.2 可编程计数器/定时器芯片 8254	(117)
§ 12.2.1 8254 的基本功能	(117)
§ 12.2.2 8254 的内部结构与引脚信号	(118)
§ 12.2.3 8254 的工作方式	(120)
§ 12.2.4 8254 的编程	(125)
§ 12.3 8254 在 PC 系列机定时系统中的应用	(131)
思考题与习题	(134)

第十三章 386/486 微机系统接口

§ 13.1 高性能多功能外围集成芯片 82380	(139)
§ 13.1.1 内部结构	(139)
§ 13.1.2 内部寄存器的分配	(146)
§ 13.1.3 82380 同 CPU 的接口	(146)
§ 13.2 外设控制器集成芯片 82C206	(148)
§ 13.2.1 主要特点	(148)
§ 13.2.2 主要功能部件说明	(149)
§ 13.3 高集成度外围接口芯片 82360SL	(151)
§ 13.3.1 主要特点	(151)
§ 13.3.2 主要功能部件说明	(152)
§ 13.4 EISA 总线 82350/82350DT 芯片系列	(154)
§ 13.4.1 概述	(154)
§ 13.4.2 EISA 总线外围芯片 82357	(155)
§ 13.4.3 总线判优	(159)
思考题与习题	(161)

第十四章 微机在测控系统中的应用

§ 14.1 计算机测控系统概述	(162)
§ 14.1.1 测控系统一般结构	(162)
§ 14.1.2 测控系统的实时性概念	(163)
§ 14.1.3 测控系统对计算机的要求	(164)
§ 14.1.4 构成测控计算机系统的方案选择	(165)
§ 14.2 实现计算机测控系统的关键技术	(169)
§ 14.2.1 传感器及测量	(169)
§ 14.2.2 模拟 I/O 通道建立	(172)
§ 14.2.3 数据处理	(184)
§ 14.2.4 执行机构及驱动	(192)
§ 14.2.5 可靠性与抗干扰技术	(202)
§ 14.3 计算机测控系统的设计与开发	(203)
§ 14.3.1 设计开发步骤	(203)
§ 14.3.2 实际测控系统举例	(207)
思考题与习题	(218)

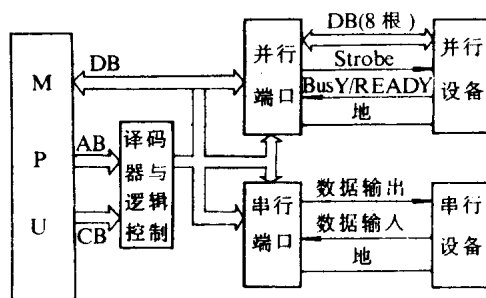
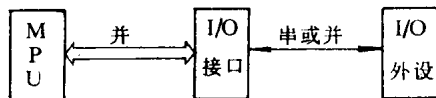
参考文献

第九章 并行接口与串行接口

§ 9.1 并行接口与串行接口概述

如前所述,微型计算机与 I/O 设备的接口按照数据传送方式的不同,可分为并行接口和串行接口两种。前者使传送数据的各位同时在总线上传输,后者则使数据一位一位地传输。并行传输又有字并行和字节并行之分,并行接口实现的一般是字节并行传输。

其实所谓的并行和串行,仅是指 I/O 接口与 I/O 设备之间的通信方式一个并行,一个串行。就接口与 MPU 之间的数据交换来说,无论并行接口还是串行接口都是一样的,均以并行方式传送数据,如图 9.1(a)所示。从该图不难想像,两者在结构和功能上的主要差别在于:串行接口需要实行并行和串行之间的相互变换,而并行接口则无需实现这种变换。至于在所用特定的 I/O 端口地址及其使用的地址译码,以及控制逻辑方面,对串行和并行两种接口来说都是类似的。当然,两种接口在接收到一次数据后的工作情况是大不相同的。



(a)并行和串行的概念

(b)并行接口和串行接口的结构、功能示意图

图 9.1 并行接口和串行接口的异同

并行接口和串行接口在结构、功能上的异同可从图 9.1(b)中大体看出。其中并行接口有 8 根数据线通向并行设备,还有两根应答控制线,一根地线,总共 11 根基本连线。串行接口将来自 MPU 的并行数据转换成位数据流,利用一根数据线,每次向串行设备输出一位数据;如果是输入设备,则接口可用类似方法,通过一根数据输入线一位一位地接收来自串行设备的数据,然后并行传给 MPU;公共地线是第 3 根连线。

显然,这两种接口比较起来各有所长。串行接口的优点是所需连线少,无论数据传输宽度是多少,一般只需一根 3 芯电缆,因此实现起来成本低。并行接口的优点则是传输速度快,一次即可同时传送 8 位,以至更多的位。究竟哪种接口更好呢?这不能笼统而言,而要根据计算机与外设之间距离的长短和对信息传输速度要求的不同而决定。通常在速度要求不高或传输距离较远的场合,例如外部设备是 CRT 显示器、磁带机或远距离数据采

集、控制装置时,常选用简单、经济的串行接口。反之,则常选用并行接口。

§ 9.2 可编程并行接口芯片 8255A—5

并行接口电路有不可编程和可编程两种。前者由数据锁存器和(或)I/O 三态缓冲器组成,电路简单,使用方便。但是,它们在硬件连接好了以后,功能就很难改变,因此,使用不灵活是其主要缺点。而可编程并行接口电路的最大特点恰恰就在于使用灵活,可以在不改变硬件的情况下通过软件编程来改变芯片的功能,因而其功能很强。

8255A—5 是目前应用最广的可编程并行接口芯片之一。它是 8255 和 8255A 的改进型产品,除在某些时间参数上有所缩短,使之能适应更高的工作频率外,其它参数基本相同,芯片功能和使用方法没有差别,相互间可完全兼容。所以,下面讲述时将它们统称为 8255。

§ 9.2.1 内部结构与引脚功能

8255 是一种使用单一的+5V 电源、40 引脚双列直插式的大规模集成电路芯片,其内部结构与外部引脚分别如图 9.2 和图 9.3 所示。

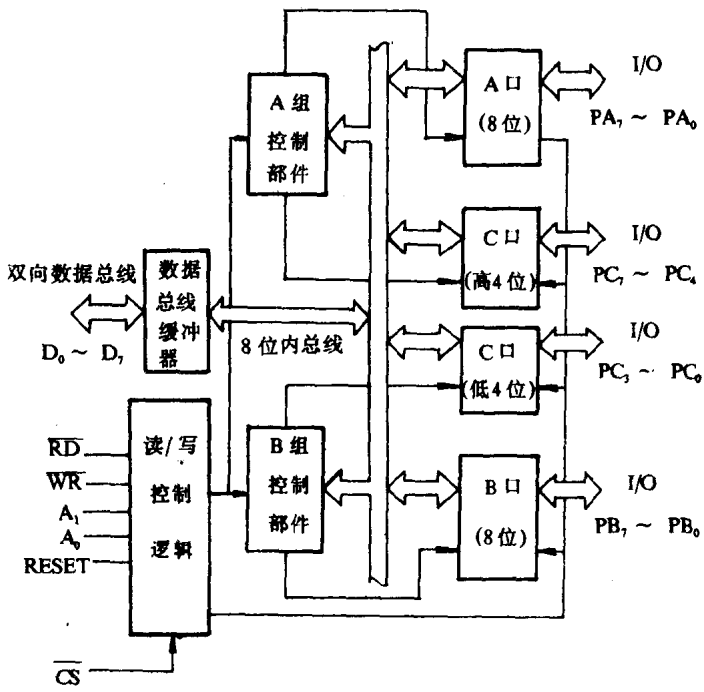


图 9.2 8255 内部结构

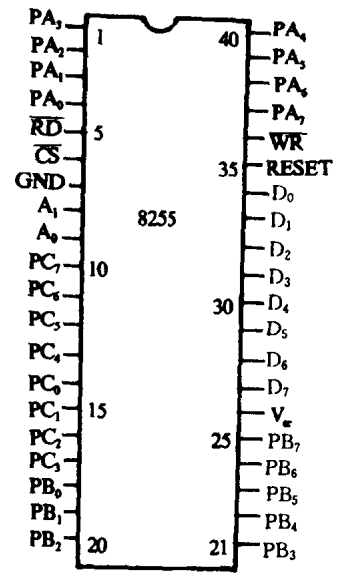


图 9.3 8255 引脚图

从图 9.2 可看出,8255 由三部分组成。

1. 与外设接口部分

这部分有 A、B、C 三个 8 位端口寄存器。通过 24 根端口线 $PA_7 \sim PA_0$ 、 $PB_7 \sim PB_0$ 、 $PC_7 \sim PC_0$ 与外部设备相连。其中 C 口又分为上半部(高半字节 $PC_7 \sim PC_4$)和下半部(低半字节 $PC_3 \sim PC_0$)。这 24 根端口线全部为双向三态。B 口和 C 口的引脚具有达林顿复合晶体管驱动能力,在 1.5V 时输出 1mA,所以通常最好选择 B 口或 C 口作输出端口。A、B、C 三个端口的工作方式和 A 口、B 口、C 口上半部、C 口下半部的输入/输出状态可通过程序来选择。8255 有 3 种基本的工作方式,定义如图 9.4 所示。

(1) 方式 0——基本输入/输出方式

这种方式下,3 个端口分为 A 组(8 位 A 口和 C 口高 4 位)和 B 组(8 位 B 口和 C 口低 4 位),每个端口都可通过编程作为输入口或输出口,作为输入口时都有三态缓冲器功能,作为输出口时都有数据锁存器功能。

(2) 方式 1——应答式输入/输出方式

这种方式下,A 口和 B 口作为 8 位输入或输出端口,C 口主要为 A、B 两口提供输入/输出的应答信号。这时的 A 口和 B 口,无论输入或输出都有数据锁存功能。

(3) 方式 2——应答式双向数据传送方式

这种方式下,A 口作为双向数据传送端口,C 口中的 5 位作为相应的应答控制信号,B 口和 C 口中余下的 3 位可工作于方式 0 或方式 1。

在上述 A、B、C 三个端口中,C 口还有位操作功能,即 C 口各位可以分别单独置“1”或置“0”,这就为形成所需的应答控制信号提供了很大的方便。

2. 与微处理机接口部分

这部分主要是保证微处理机对 8255 的编程、监视和提供数据通道。

(1)数据总线缓冲器 这是一个 8 位双向三态缓冲器。所有数据的输入输出,以及对 8255 发的控制字和从 8255 读的状态信息,都是通过这个缓冲器传送的。

(2)读写控制逻辑 有 6 根控制线,控制 8255 内部各种操作。RESET 用于 8255 内部复位, $\overline{CS}$ 和地址线 A_1 、 A_0 分别用于选片和片内选端口, $\overline{RD}$ 和 $\overline{WR}$ 用于控制 8255 数据的读/写。RESET 为高电平有效信号。芯片复位时,片内各寄存器都被清“0”,且 A、B、C 三个端口都被置为输入方式,24 条 I/O 端口线均为“高阻”态。RESET 信号通常与系统复位信号连在一起。 $\overline{CS}$ 、 A_1 、 A_0 、 $\overline{RD}$ 、 $\overline{WR}$ 等信号对 8255 内部各寄存器及其操作的选择控制作用如表 9.1 所示,8255 与微处理机总线的接口方法如图 9.5 所示。需要说明如下三点:

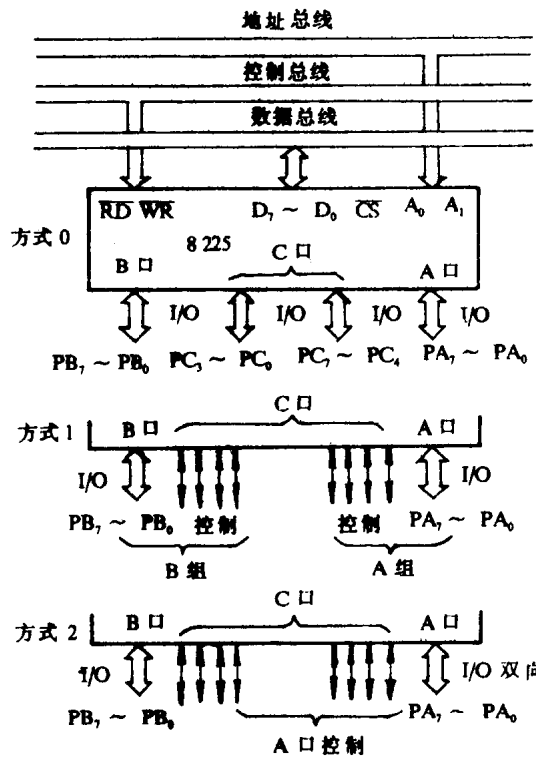


图 9.4 8255 三种基本工作方式

①对于采用专用I/O 编址方式的MPU,将 $\overline{\text{IOR}}$ 和 $\overline{\text{IOW}}$ 分别接到8255的 $\overline{\text{RD}}$ 和 $\overline{\text{WR}}$;而采用存储器映象式I/O 编址方式的MPU,则将 $\overline{\text{MEMR}}$ 和 $\overline{\text{MEMW}}$ 分别接到 $\overline{\text{RD}}$ 和 $\overline{\text{WR}}$.

②地址总线中的*i*代表I/O 端口地址线的根数,一般将AB 的最低两位 A_1 、 A_0 直接连到芯片的 A_1 、 A_0 . 作为端口选择线,而将其它高位地址线通过译码形成片选信号。

③因为8255 的RESET 信号是高电平有效,所以对于复位信号是低电平有效的MPU,如Z80/Z8000 系列MPU 和MC6800/MC68000 系列MPU 等,RESET 引脚必须经一反相器才能与8255 的RESET 引脚相连。

表 9.1 8255 的端口与操作选择表

A_1	A_0	$\overline{\text{RD}}$	$\overline{\text{WR}}$	$\overline{\text{CS}}$	操 作	
0	0	0	1	0	A 口→数据总线	输入
0	1	0	1	0	B 口→数据总线	
1	0	0	1	0	C 口→数据总线	
0	0	1	0	0	数据总线→A 口	输出
0	1	1	0	0	数据总线→B 口	
1	0	1	0	0	数据总线→C 口	
1	1	1	0	0	数据总线→控制寄存器	
×	×	×	×	1	端口输出为“高阻”	禁止
1	1	0	1	0	非法	
×	×	1	1	0	端口输出为“高阻”	

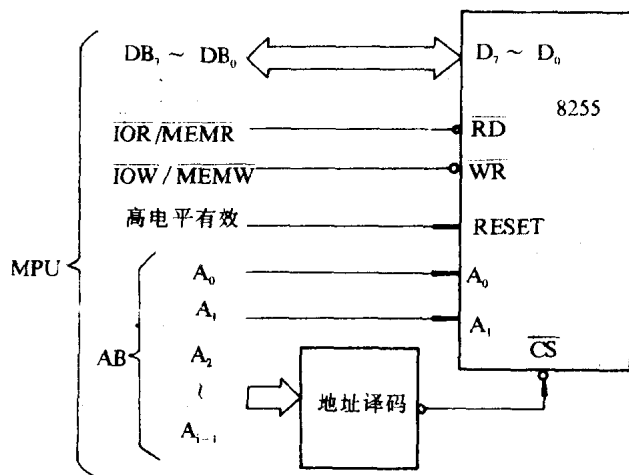


图 9.5 8255 与 MPU 总线的接口方法

3. 内部控制部分(A 组和 B 组控制部件)

这是两组根据CPU 送来的控制字控制8255 工作方式和输入/输出状态的控制部件。每组控制部件从读/写控制逻辑接收各种命令,从内部数据总线接收控制字并发出适当的命令到各自相应的I/O 端口。它也可以根据CPU 的命令字对C 口的每一位实现按位置“1”或置“0”控制。

A 组控制部件控制端口 A 和 C 口的上半部, B 组控制部件控制端口 B 和 C 口的下半部。实际上, A 组、B 组控制部件就是同一个 8 位的控制寄存器。有些位被分配控制 A 组, 有些位被分配控制 B 组。CPU 用一条输出指令写一个控制字到该控制寄存器, 即可选择和控制 A、B、C 各端口的工作方式。

§ 9.2.2 控制字的确定和初始化编程

8255 的工作方式和接口功能是 CPU 通过把控制字写入控制寄存器来实现的, 通常把这个过程称为初始化编程, 简称为初始化。初始化的基础是根据应用需要正确确定控制字, 因此必须了解 8255 的控制字格式。

1. 工作方式控制字

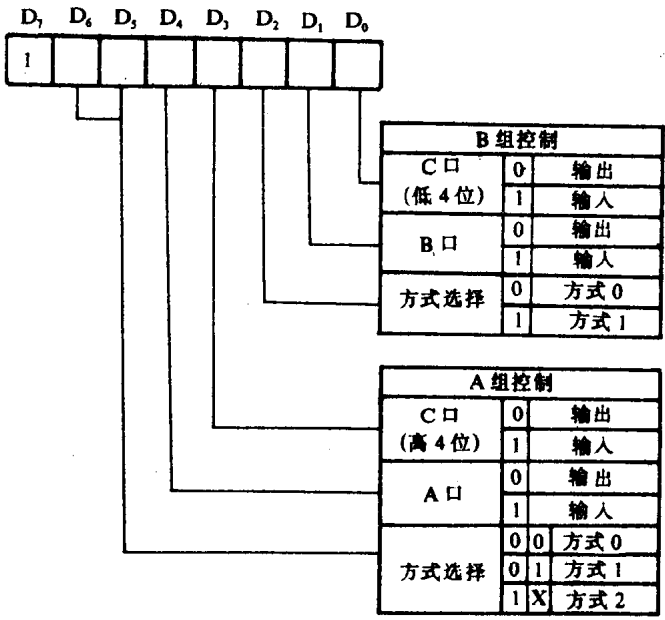


图 9.6 8255 工作方式控制字格式

工作方式控制字的格式如图 9.6 所示, 它规定了控制字各位的含义。 D_7 为方式控制字标志位, “1”为有效。 $D_6 \sim D_3$ 为 A 组控制位。其中 $D_6 D_5$ 用于设置 A 口的工作方式(A 口有 3 种工作方式), D_4 用于设置 A 口的输入/输出状态, D_3 用于设置 C 口高 4 位的输入/输出状态。 $D_2 \sim D_0$ 为 B 组控制位。其中 D_2 用于选择 B 口的工作方式(B 口只有两种工作方式), D_1 用于选择 B 口的输入/输出状态, D_0 用于选择 C 口低 4 位的输入/输出状态。工作方式控制字的这种格式表明:

①端口 A 和端口 B 要分别作为一个整体确定工作方式, 而端口 C 则是分成高 4 位、低 4 位两部分分别确定工作方式。端口 A 和端口 B 的工作方式可以不同, 端口 C 的上半部和下半部的工作方式也可以不同。8255 这四部分的工作方式可以任意组合, 这就使 8255 的 I/O 结构有很大的灵活性, 几乎能适应任何一种外部设备的连接需要, 还能满足同时连接几种不同 I/O 设备的需要。

②虽然 8255 的 I/O 有上述 4 个部分,每部分的工作方式又可以不同,但是所有各个部分的工作方式却是 CPU 用一条输出指令,通过一个控制字写入一个控制寄存器而确定的。这对于简化初始化编程是十分有利的。

2. C 口按位置位/复位控制字

这是专门用于对 C 口 8 位中任何一位实现置“1”或置“0”的控制字。该控制字的格式如图 9.7 所示,它只使用了 5 位有效位。 D_7 为按位置位/复位控制字标志位,“0”表示是本控制字。 $D_3D_2D_1$ 用于选择 C 口中要置“1”/置“0”的位,000 选 PC_0 ,001 选 PC_1 ,...,111 选 PC_7 。 D_0 用于确定所选 PC_i 是置“1”还是置“0”。 $D_0=1$,置“1”; $D_0=0$,置“0”。 $D_6D_5D_4$ 三位未使用,原则上可任意设置,但一般取 000。

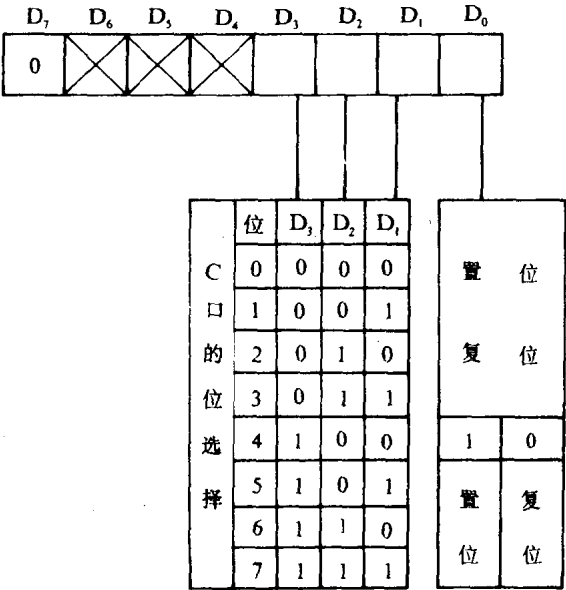


图 9.7 8255 C 口按位置位/复位控制字格式

C 口按位置位/复位的功能主要用于对外设的控制。利用这一功能,可使 C 口某一位输出一个开关量或一个脉冲,作为外设的启动或停止信号。当端口 A 或 B 工作在方式 1 和方式 2 时,利用这一功能,也可使作为应答控制线的 C 口有关位产生所需的联络信号(脉冲或电平)。这样,显然提高了应答线使用的灵活性,当然在程序上也增加了一些额外的控制步骤。在使用 C 口按位置位/复位功能时要注意如下三点:

①这一功能可使 C 口的任一位置“1”或置“0”,但一次(一条输出指令,一个控制字)只能使 C 口的一位置“1”或置“0”。如果使 C 口几位都要置“1”或置“0”,必须用几条输出指令,写入几个不同控制字。表 9.2 给出了按位置位/复位控制字中的 $D_3D_2D_1$ 取 000 时, C 口各位的置位、复位控制字。

②如果要在某位上输出一个开关量信号,对该位的置 1/置 0 操作必须成对使用。比如要使 PC_5 输出一个负跳变信号,可用下列 80X86 指令:

```
MOV AL,OBH          ;PC5 置“1”控制字送 AL
MOV DX,×××××11B    ;控制寄存器口地址送 DX
```

```

OUT  DX,AL           ;置“1”控制字送控制寄存器
MOV  AL,0AH          ;PC5置“0”控制字送 AL
OUT  DX,AL           ;置“0”控制字送控制寄存器

```

表 9.2 8255 C 口按位置位/复位控制字

C 口位	置位控制字		复位控制字	
	二进制	十六进制	二进制	十六制
PC ₀	00000001B	01H	00000000B	00H
PC ₁	00000011B	03H	00000010B	02H
PC ₂	00000101B	05H	00000100B	04H
PC ₃	00000111B	07H	00000110B	06H
PC ₄	00001001B	09H	00001000B	08H
PC ₅	00001011B	0BH	00001010B	0AH
PC ₆	00001101B	0DH	00001100B	0CH
PC ₇	00001111B	0FH	00001110B	0EH

如果要在 PC₅ 输出一个负脉冲,则在上列程序后面还要增加两条使 PC₅ 再置“1”的指令(因为各口初始化之前的状态是置“0”的):

```

MOV  AL,0BH
OUT  DX,AL

```

③C 口按位置位/复位控制字不是送到 C 口地址,而是送到控制寄存器地址。对此千万不要混淆。

综上所述可以看出,8255 只有一个控制寄存器地址,但需要写入该地址的控制字却有两种:工作方式控制字和 C 口按位置位/复位控制字。这两种控制字是以最高位 D₇ 作为区分的标志,D₇=1 为方式控制字,D₇=0 为按位置位/复位控制字。所以,控制字值大于、等于 80H 的是方式控制字;小于 80H 的是按位置位/复位控制字,而其中奇数值是置位控制字,偶数值是复位控制字。

§ 9.2.3 三种工作方式和接口方法

1. 方式 0 的功能及接口方法

这是一种基本输入输出工作方式,通常不用应答信号,或不使用固定的应答线,不使用中断。在这种工作方式下,三个端口都可以由程序选定作为输入或输出。

(1)简单的输入输出操作

简单的输入输出适合于:外设始终有数据可以给微处理器,以及始终能接受微处理器送来的数据。这种情况下可以进行无条件传送。典型的例子是以开关作为输入,以指示灯显示作为输出,如图 9.8 所示。

微处理器通过 8255 的 B 口从外部开关取得数据,进行处理后,再通过 8255 的 A 口用发光二极管将处理结果显示。此时的 8255 方式控制字为 10000010(82H)。实现此功能的流程图如图 9.9 所示。