

微机接口与 输入输出过程通道

史福元 编著

科学技术文献出版社

微机接口与输入输出过程通道

史福元 编著

科学技术文献出版社

(京)新登字 130 号

内 容 简 介

本书以国内三种主流机型 IBM PC 系列 16 位微机、Z80 8 位机和 MCS-51 系列单片机为主要对象,阐述微机接口与输入输出过程通道技术。

全书共十二章,第一至第五章介绍微机接口技术;第六至第十一章介绍输入输出过程通道。内容涉及传感器、采样与放大、A/D 与 D/A 转换,以及功率驱动和执行装置接口技术等;第十二章介绍工业测控局域网络。本书内容系统全面,并结合科研和应用成果进行介绍,实例丰富多样,有利于建立微机工业应用的整体概念。便于教学、自学和应用。

本书可作为高等学校机、电类大学生的教材或教学参考书,也可供有关专业研究生、工程技术人员、微机应用与科技开发人员使用或参考。

图书在版编目(CIP)数据

微机接口与输入输出过程通道/史福元编著. -北京:
科学技术文献出版社,1995
ISBN 7-5023-2595-6

I. 微… II. 史… III. ①微型计算机-接口设备②微型
计算机-输入设备③微型计算机-输出设备 IV. TP36

中国版本图书馆 CIP 数据核字(95)第 10431 号

科学技术文献出版社出版

(北京复兴路 15 号 邮政编码 100038)

通县建新印刷厂印刷 新华书店北京发行所发行
787×1092 毫米 16 开本 14.125 印张 365 千字

1995 年 9 月第 1 版 1995 年 9 月第 1 次印刷

印数:1—3000 册

定 价:17.80 元

序

随着大规模和超大规模集成电路技术的飞速发展,微型计算机的应用已渗透到工业过程控制、智能化仪器仪表、数据采集与处理、家用电器和其它有关科技领域。然而,微机应用离不开接口,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。

从微处理机诞生到广泛应用的今天,已经有不少关于接口原理和应用方面的书。本书无论从编写指导思想上,还是从内容的组织和安排上,都作了较好的处理,一改旧貌,刻意创新。其主要表现在:面向广泛应用,针对国内三种主流机型展开对微机接口的介绍;在介绍微机接口技术的基础上,把内容扩展到输入输出过程通道的阐述;提出了微机与生产设备或过程进行“广义接口”的概念;并对过程通道的组成原理,包括从传感器技术、采样与放大、A/D与D/A转换技术、功率驱动与执行装置的接口技术,到工业测控局域网络等都分别作了详尽讨论。全书内容丰富全面,既有原理分析与设计方法介绍,又结合实际给出了应用实例。该书在理论联系科研与应用实际方面有创见,也是目前见到的一本比较全面系统地论述微机接口与输入输出过程通道的专门著作。

朱逸芬

1995.5.6

前 言

随着 70 年代微处理机的问世和 80 年代高性能微机与超级微型机进入市场,微型计算机已迅速发展到应用于工业生产过程控制系统、数据采集与处理系统、智能化仪器仪表、办公自动化系统和社会生活的各个领域。微机应用的普及和深入,必然会带来技术支持的广泛需求,而这种技术支持的重要内容就是微机与控制或应用对象之间的接口组成及其功能实现。本书正是为满足这种需求而写成的。

全书由两大部分组成,分为十二章。第一部分包括第一至第五章,在分析和论述了有关接口的共性技术——微机接口技术基础之后,分别对并行接口、串行通信接口和可编程计数器/定时器接口进行了较为深入的介绍,然后讨论了人—机通信设备(键盘、显示器和打印机)的接口技术。第二部分包括后七章内容,其中第六至第十一章是关于输入输出过程通道的组成与设计技术,第十二章对微机工业测控局域网络作了专门介绍。为了满足用微机组组成各种实际应用系统的广泛需要,第二部分内容将微机接口技术扩展到更为广阔的应用领域。其中,模拟量输入通道涉及到的主要内容包括:仪表测试技术、传感器技术和模拟信号处理技术等;模拟量输出通道涉及到伺服驱动、电机电器、大功率器件和控制工程等技术领域。将微机接口与输入输出过程通道贯穿起来,进行全面系统的介绍,是本书的特点之一。

本书涉及到的微型机机型,既有以 Z80 为代表的典型 8 位微机,又有 MCS-51 系列单片机,还有 Intel 8086/8088 16 位微机 IBM-PC/XT 及其兼容机。三种典型微机接口和应用实例并举,是本书的特点之二。

本书的特点之三,表现在既重视基本原理的阐述,又注重科研与应用成果的介绍,内容上兼顾了接口与通道的硬件组成及其软件实现。本书是一部颇具特色的微机接口与输入输出过程通道技术的专门著作。既可以作为高等学校机、电类专业大学生或研究生的教材或教学参考书,也可供从事控制工程和微机应用系统设计、使用和维护的广大工程技术人员使用或参考。

合肥工业大学朱逸芬教授详尽审阅了书稿,提出了不少宝贵意见,并为该书作序。安徽大学李仁教授、上海大学陈伯时教授和龚幼民教授都曾给予作者以热忱鼓励和支持,为此一并表示诚挚的谢意。中国迅达电梯有限公司合肥服务中心张宏国高级工程师、机械部北京自动化研究所季瑞芝高级工程师也曾给予热心帮助与支持;安徽大学马修水和赵永芳老师为书稿的图文整理付出了辛勤劳动。在此谨向他们,并向作者的长期科研合作伙伴温阳东、祖伟两同志,向有关参考文献的作者及关心和支持本书出版的朋友们表示衷心感谢。

由于作者水平有限,时间仓促,错误和不足之处在所难免,恳请广大读者批评指正。

作 者

1995 年于合肥

目 录

第一章 微机接口技术基础	1	第一节 传感器概述	120
第一节 微机接口技术概述	1	第二节 温度测量及温度传感器	121
第二节 微机接口操作原理	5	第三节 压力测量及压力传感器	126
第三节 输入输出数据传送方式	9	第四节 流量与液位的测量	129
第四节 中断技术	12	第八章 测量放大技术	132
第五节 总线技术与标准	20	第一节 测量放大器	132
第二章 并行接口	29	第二节 程控增益放大器	134
第一节 并行接口概述	29	第三节 隔离放大器	137
第二节 Intel 8255A 及其应用	30	第九章 模拟多路开关与采样保持器	140
第三节 Z80 PIO 及其应用	38	第一节 模拟多路开关的作用与分类	140
第三章 串行通信接口	47	第二节 模拟多路开关的配置与选用	141
第一节 串行通信基础	47	第三节 模拟多路开关与微机的接口	143
第二节 串行通信接口芯片及其应用	52	第四节 采样保持器的作用原理	144
第三节 串行通信总线标准	63	第五节 集成采样保持器芯片及其应用	145
第四章 可编程计数器/定时器接口	70	第十章 模/数和数/模转换与接口	148
第一节 概述	70	第一节 D/A 转换原理与接口技术	148
第二节 Z80 CTC 及其应用	71	第二节 A/D 转换原理与接口技术	162
第三节 Intel 8253 及其应用	76	第三节 V/F 转换原理与接口技术	178
第五章 人一机接口	86	第十一章 驱动与执行装置及其控制接口	182
第一节 人一机接口概述	86	第一节 驱动与执行装置的作用与分类	182
第二节 键盘接口	87	第二节 常用功率驱动器与驱动接口	183
第三节 显示器接口	96	第三节 电动执行装置及其控制接口	189
第四节 打印机接口	106	第十二章 微机工业测控局域网络	203
第六章 输入输出过程通道	111	第一节 概述	203
第一节 概述	111	第二节 BITBUS 分布式局域网	207
第二节 模拟量输入通道	112	第三节 TDC-3000 集散控制系统	214
第三节 模拟量输出通道	116	第四节 远程智能分布式测控网络	217
第四节 开关量输入和输出通道	117	参考文献	219
第七章 传感器技术	120		

第一章 微机接口技术基础

接口(Interface)是指两个电路或设备之间的分界面或连接点。所以,接口可以是微处理器与输入输出设备(如键盘、CRT 显示器、打印机等)之间的连接部件,也可以是微型计算机与外部设备(如传感器、执行机构、智能化仪表等)之间的连接部件。这种两部件之间的连接既包括物理电路上的连接,也包括逻辑上的连接,即信息如数据、命令、状态等的交换。

接口技术是采用硬件和软件相结合的方法,研究微处理器和外部世界之间如何实现安全、可靠、高效的信息交换的技术。接口技术是直接影响微机系统推广应用成败的关键技术之一,熟悉与掌握微机接口技术也就成为当代科技开发与工程技术人员应用微机系统的基本技能之一。

第一节 微机接口技术概述

一、微机接口的功能

图 1-1 给出了一个典型的微型计算机系统组成框图。如图所示,存储器和不同种类的外部设备都是与各自的接口电路相连,而微处理器则通过公用的三总线(地址总线、数据总线和控制总线)与所有的接口电路连接。

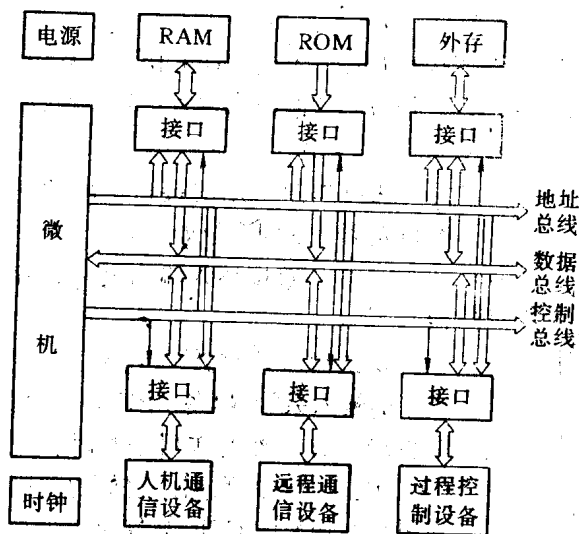


图 1-1 微机系统组成框图

微机接口一般具有以下几种功能:

(一) 信号电平的转换

外部设备大多数为复杂的机电设备,有各自的电源系统。而微机电源系统为 TTL 电平或 MOS 电平。两者电气规范不一致,需要由接口电路来完成信号电平的转换。有时,接口还可以兼作电气上的隔离。

(二) 数据格式的转换

微机系统内部数据传送为多位(8 位、16 位或 32 位等)并行方式,而外设大都采用位串行数据传送方式。故要求接口具有数据“并→串”和“串→并”的转换能力。对于输入输出过程通道来说,还有 A/D 和 D/A 的转换问题需要解决。

(三) 数据的暂存与缓冲

由于外设的工作速度一般都远低于微处理器的信息处理速度,所以,为了充分发挥微处理器的效率,并缓解两者在速度上的差异所造成的冲突,接口应具备一定的数据暂存能力。故接口内一般均设置有数据缓冲寄存器和锁存器。

(四) 对外设的控制与监测

接口接收 CPU 送来的命令、控制或定时信号,对外设实施控制与管理。外设的工作状态或应答信号也要及时返回给 CPU,以实现 CPU 与外设输入输出操作的同步。同时,由于微机系统中往往连接有多种或多台外设,CPU 也要借助接口中的地址译码电路对外设进行寻址。一般高位地址线用于接口芯片的选择,低位地址线用于芯片内部端口的选择,以选定需要进行信息交换的外设。

(五) 实现中断管理

为满足 CPU 与外设并行工作的要求以及实时性要求,许多外设以硬件中断方式与 CPU 进行信息交换。为此,接口中应有中断请求信号的产生与屏蔽逻辑,有的还具有中断优先权排队电路。对于可采用 DMA 方式传送数据的外设,接口还应具有 DMA 请求信号的产生与屏蔽逻辑。

(六) 可编程功能

所谓可编程,即可以用程序来改变接口的工作方式。大多数接口芯片是可编程的,这样便可以在不改动硬件的情况下通过修改驱动程序改变接口的工作方式,从而大大增强了接口的灵活性和适应性。

应当指出,并非所有接口都具备上述六种功能。对于不同应用和不同配置的微机系统,其接口功能一般有所不同。但是,数据缓冲功能、输入输出操作的同步功能和可编程功能,是各种接口都应具备的基本功能。

二、微机接口的基本结构

根据接口的基本功能要求,为协调并实现数据在 CPU 和外设之间进行正常传送,接口电路主要由数据输入输出缓冲寄存器、控制寄存器与状态寄存器、数据总线缓冲器、地址总线缓冲与译码器,以及内部控制逻辑等几部分组成。

接口电路通常做在一块大规模集成电路芯片上,称作接口芯片。不同功能的接口芯片,其内部结构各不相同,但大都包括上述几部分电路组成。典型接口芯片的内部逻辑结构框图,如图 1-2 所示。

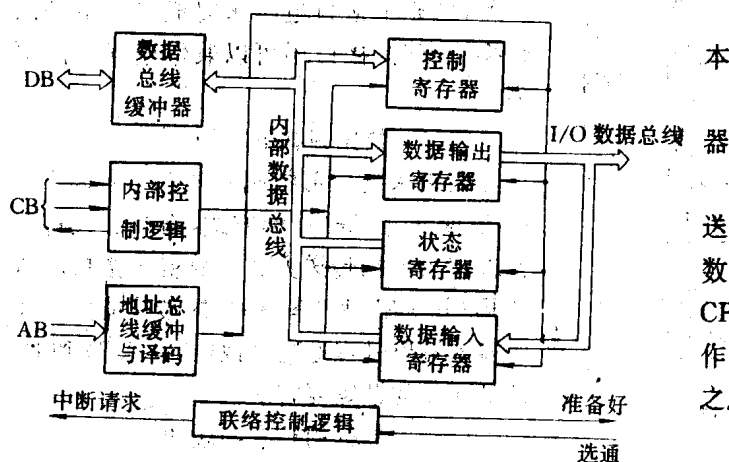


图 1-2 典型接口芯片内部逻辑框图

现对如图所示的接口电路的基本组成部分分别介绍如下:

(一) 数据输入输出缓冲寄存器

数据输入寄存器用于暂存外设送来的输入数据,以备 CPU 来读取。数据输出寄存器的作用是暂时存放 CPU 送往外设的数据。它们的共同作用是在快速的 CPU 与慢速的外设之间进行协调与缓冲。

(二) 控制寄存器

控制寄存器用来存放由 CPU 写入的决定接口电路工作方式和应有功能的控制字或命令字信息。这些信息是通过对接口芯片编程写入的,由它们决定接口应处于

何种工作方式或状态。

(三) 状态寄存器

这是用来寄存外设当前所处的各种状态信息的寄存器。CPU 通过读取该寄存器的内容,及时了解数据传送过程中已经发生或正在发生的事情,以便作出正确判断,并能安全可靠地与接口完成交换信息的各种操作。

(四) 数据总线缓冲器

由于 CPU 的系统数据总线是输入输出双方向的,而且对所有外设接口来讲是公用的。所以,各个接口电路与 CPU 之间所进行的输入输出数据传送,一般都经过该数据总线缓冲器,而不是将系统数据总线与接口内部数据总线直接相连。

(五) 地址总线缓冲与译码器

经 CPU 数据总线取入或送出的数据,可能是外设送来或 CPU 送出的数据信息,也可能是 CPU 送给接口或外设的控制信息或者是由接口取入的外设状态信息。为了进行正确区分,对上述靠外设一侧的数据输入输出缓冲寄存器、控制寄存器和状态寄存器分别各分配一个地址,称为端口地址。此类端口地址的形成便是地址总线缓冲与译码器的功能。

(六) 内部控制逻辑

内部控制逻辑电路用来产生或形成芯片内部的控制信号、中断申请信号,以及系统的控制和应答信号。所有这些控制信号确保了 CPU 正确无误地与外设进行信息交换。

三、微机接口的分类

微机接口分类的方法很多,常用的有两种:一种是按照接口与外设之间交换信息的方式来分类,另一种是按照接口电路的功能来分类。

(一) 按接口与外设交换信息的方式分类

按照这种方式分,接口有并行接口和串行接口两种。

并行接口可使接口和外设之间同时传送若干位数据(比如说 8 位)。传送速度快,需要信号线数多,适合于近距离数据传送。

串行接口是外设和接口之间实行一位接一位的串行传送,传送速度较慢,但数据线只要一根,适合于远距离数据通信。

值得指出的是,不管是并行接口还是串行接口,在 CPU 系统总线一侧与微处理器进行数据传送都是并行的。

(二) 按照接口电路的功能不同分类

按照接口电路和功能不同一般可分为辅助操作接口、人一机通信接口、过程控制接口和智能 I/O 接口等。

1. 辅助操作接口

辅助操作接口是使微处理器能发挥最基本的处理和控制功能的接口部件。它包括数据总线和地址总线的接收器、驱动器、微处理器的时钟电路,以及存储器部件的接口电路等。

总线接收器具有总线加载缓冲、总线滤波和阻抗匹配等三种功能。总线驱动器是一种功率放大器,用以增加微处理器地址总线和控制总线的驱动能力。

随机存取存储器 RAM 的接口电路包括总线接收器、总线驱动器,以及存储器译码和读/写控制逻辑。只读存储器 ROM 的接口电路,因不用 CPU 对其进行数据写入操作,所以其接口电路除数据总线土不用接收器外,基本上与 RAM 接口电路类似。然而对于 PROM 和 EPROM

的接口电路因要满足编程写入的需要,必须加上高压驱动器等辅助电路。

对于大容量的外存储器,如磁带、软磁盘和硬磁盘的接口,也可以归入辅助操作接口一类。这些设备与微处理器的接口一般是由专用的并行或串行数据通信接口控制器来实现的。因限于篇幅,本书不作具体介绍。

2. 人一机通信接口

人一机通信接口,简称人一机接口,是指微处理器接收用户输入的信息,或向用户送出信息所需的接口电路。所以,有时亦称用户通信接口。这类接口主要包括:键盘接口、显示器接口、打印机接口、图形设备接口和语音输入输出接口等。

人一机接口的基本功能主要在于完成信息表示方法和数据传送速率的两种转换工作。一般做法是,信息表示方法的转换任务分配给外部设备去完成,而数据传送速率的转换工作由CPU来承担。CRT显示器就是一个典型例子,用户通过键盘输入数据,这是一种机械到电气的转换接口。输入数据被转换成ASCII码,并使用计算机的表示方法,即一串1或0的序列送给相关接口和CPU。CPU通过中断或软件查询等待方式与用户输入数据的速率取得同步,这样便实现了传送速率的转换。

人一机接口将在第五章进行具体介绍。

3. 过程控制接口

过程控制接口通常又可分为:传感器接口和控制接口。

当微机系统用于智能化仪表、实时数据监测或控制时,必须测量对象现场的各种数据,如温度、压力、流量等。这类系统中,作为微机的外部设备有各种传感器,如温度传感器、压力传感器和流量传感器等。这些传感器首先将各种现场非电量参数转换成相应的电流或电压,这些电量都是随时间连续变化的模拟量,而微处理器只能处理离散的数字量。因此,在这些传感器和微处理器之间需要有适当的接口电路,一方面解决模拟量到数字量的转换,另一方面将转换后的数字量传送到CPU中,这些就是传感器接口所要完成的任务。

当微机系统作为一个测控系统时,除了需要上述传感器接口之外,还必须根据现场实测数据和对象的控制要求,经过适当运算处理后,对现场对象的运行状态或过程进行控制。微处理器是通过控制接口向控制对象发出命令实施控制的。

控制接口应当具有两个方面的功能:一个是数/模转换功能,即要将微处理器经过运算得出的结果(数字量)转换成能推动现场执行机构动作的模拟量;另一个是功率放大驱动功能。这是因为微处理器送出的经过数/模转换器转换成的模拟控制信号电压或电流的功率很小,必须经过功率放大,才能驱动电磁线圈、步进电机、伺服电机等执行机构动作。

传感器接口和控制接口作为本书的重要内容,将在后续章节作详细介绍。

4. 智能I/O接口

智能I/O接口,即给I/O接口电路赋予某些智能功能,也就是在接口电路中设有专用的I/O处理器,形成所谓的“智能I/O子系统”,专司输入输出数据的传送控制和加工处理职能。这些职能原来是由微处理器完成的,这样就可以确保主处理器以最高效率执行内部程序和数据处理等任务。

智能I/O接口一般有局部结构和远程控制结构两种。目前,发展较快的是远程智能I/O系统,它是在集散控制系统DCS的插板式智能I/O技术的基础上,融合了可编程控制器PLC的现场应用处理技术,采用现场的控制机将现场信号就近进行处理,并运用数字通信技术将所

有现场控制机联成网络,再经现场总线和适配器与主微机相连进行运算处理、监督管理与协调控制。

远程智能 I/O 系统由于具有诸多突出的优点,因而在工业控制领域有着十分广阔的应用前景。

四、 常用接口芯片

随着大规模集成电路技术的发展,各种不同功能的集成电路接口芯片应运而生。用户在进行微机应用系统接口设计时,只需要深入了解和熟练掌握各类接口芯片的工作原理、性能特点、使用方法和编程技巧,以便合理地选择芯片,把它们与微处理器正确地连接起来,并正确地编写相应的驱动程序。所以,接口芯片在微机接口技术中起着相当重要的作用,应当给予足够的重视。

对于 8 位微机三种主流机型: Intel 公司的 Intel8080/8085, Zilog 公司的 Z80 和 Motorola 公司的 M6800,分别提供了与各自配套使用的不同种类接口电路支持芯片。值得庆幸的是,这些通用接口芯片的数据线虽然只有 8 位,但它们大部分均可用于 16 位 PC 机上的字节操作或两片合并进行字操作。

表 1-1 列出了三种常用 8 位微处理器的系列支持芯片,供读者使用时查阅。

表 1-1 常用集成接口芯片

CPU 芯 片 接 口	Intel 8080/8085	M6800	Z80
串行接口	8251	6850	Z80 SIO
并行接口	8255,8155	6820/6821	Z80 PIO
定时器/计数器	8253/8254,8155	6840	Z80 CTC
DMA 控制器	8257,8237	6844/68A44	28410/28410A,Z80 DMA
中断控制器	8259,8214	6828/8507	
CRT 控制器	8275	6845	
软盘控制器	8271/8272	6843	
键盘、显示接口	8279		
通用接口	8291,8292,8293	68488	

第二节 微机接口操作原理

一、 典型 8 位微机的输入输出寻址方式

如前所述,外部设备是通过专用的接口电路,而不是直接地同微处理器进行连接的。每个外设通常都设有一个或几个端口,每个端口都有特定的端口地址,如同每个内存单元都有一个与之相对应的地址号码一样。CPU 同外设打交道时,首先要寻找该外设的端口号,这称作 CPU 对接口电路的操作,或 CPU 对输入输出设备的寻址。由于外设的端口地址设置方法不同,相应的寻址方式也就不一样。常用的输入输出寻址方式有存储器统一编址方式和 I/O 接口编址方式两种。

(一) 存储器统一编址方式

这种编址方式的基本做法是把每一个输入输出设备端口(简称 I/O 端口)都当作一个存储单元来对待,每个外设的每一个端口都指派一个地址号。因而,CPU 在对某一外设进行输入输出操作时,就象对某一内存单元进行读、写操作一样,仅地址号不同而已。这样,所有的访内指令均同样适用于对 I/O 端口的操作。因此,不必再设专门的输入输出指令,使指令系统得以简化。M6800 和 16 位的 M68000 都是采用存储器统一编址方式。

存储器统一编址方式的优点有二:其一,存储器访内指令可以用来处理输入输出操作。由于访内指令多,而且功能强,使用灵活,这给程序设计带来很大方便。例如,可以利用存储器指令直接对 I/O 端口中的数据完成算术运算,而不必把内容送入或送出中间寄存器。同样地可对 I/O 端口的内容完成逻辑运算和移位等多种操作;其二,I/O 端口可以有较大的编址空间,这在某些特殊场合很有用,例如在某些大型控制或数据通信系统中,可能需要多于 256 个输入输出端口。

这种编址方式的缺点在于:I/O 端口占用了内存单元地址号,因而减少了内存容量。另外,每个 I/O 指令操作需要全字长地址译码,指令的执行时间较长,而且程序较难区分 I/O 操作。

(二) I/O 接口编址方式

从组成原理上讲,可以认为计算机就是由 CPU、存储器和 I/O 设备所组成的机器。CPU 通过一个接口与存储器相连,通过另一个接口和 I/O 设备相连。为了访问存储器以及进行 I/O 操作,指令系统中分别设立了面向存储器和面向输入输出两类指令。

I/O 接口编址方式,就是将存储器地址空间和 I/O 接口地址空间分开设置,互不影响。但是,由于地址总线是为存储器和输入输出设备所共享,就产生了另一个问题,即地址总线上的地址究竟是给谁的。

Z80 采用这种编址方式,是这样来处理上述问题的。即用两根不同控制信号线区分存储器寻址和 I/O 端口寻址。一根为 I/O 请求线 $\overline{\text{IORQ}}$,一根是存储器请求线 $\overline{\text{MREQ}}$ 。如果 $\overline{\text{IORQ}}$ 线为低电平,表示地址总线和数据总线用作 I/O 操作。假如 $\overline{\text{MREQ}}$ 线为低电平,表示访问存储器。所以,利用地址总线 $A_0 \sim A_{15}$ 配合 $\overline{\text{MREQ}}$ 线为低,即可产生 64K 字节的内存寻址空间;利用地址总线低 8 位 $A_0 \sim A_7$ 配合 $\overline{\text{IORQ}}$ 线为低,便可产生 256 个外设端口地址,再配合读、写操作控制信号 $\overline{\text{RD}}$ 、 $\overline{\text{WR}}$,就可以对选中的 I/O 端口进行输入或输出操作。

I/O 接口编址方式也有两个优点:第一,由于采用了单独的 I/O 指令,使输入输出程序简单清晰,易于区分;第二,该类输入输出指令的地址较短,硬件译码电路简单,指令执行时间也短。

这种编址方式的缺点是:专用的 I/O 指令的功能有限,输入和输出数据都必须经过累加器,然后才能进行算术和逻辑运算。其次,由于采用了专用的 I/O 周期和专用的 I/O 控制线,增加了微机自身控制逻辑的复杂性。

目前,除 Z80 外,还有 8086/8088 和一些大型微机都采用 I/O 接口编址方式。

二、PC/XT 的 I/O 端口的地址分配与译码

16 位微机 8086/8088 采用 I/O 接口编址方式,将存储器地址和 I/O 端口地址分开设置。在进行外设连接或接口设计时,要弄清楚系统对 I/O 端口地址的分配情况,才能做到正确连接,合理选用。

下面以 PC/XT 微机系统为例,介绍其 I/O 端口地址的分配细则及其译码方法。

(一) PC/XT 的 I/O 的地址分配

16 位微机 8086/8088 的输入输出指令可以使用 16 位地址 $A_{15} \sim A_0$, 使 I/O 地址空间可达 64k。但是实际的 PC/XT 微机系统仅用 $A_9 \sim A_0$ 10 位地址, 可用 I/O 端口为 1024 个。

这 1024 个端口是这样分配的: 一部分已被系统本身占用, 另一部分已被配置的接口卡所用, 还有一部分被保留为今后开发使用, 最后留出一部分给用户使用。具体分为两大部分: 当 $A_9 = 0$ 时, 低端的 512 个地址为系统板所用; 当 $A_9 = 1$ 时, 高端 512 个供扩展插槽使用。具体分配细则见表 1-2 所示。由表 1-2 可见, 留给用户自由使用的只有 300H~31FH 地址区段。但原则上讲, 凡是未被占用的保留地址, 用户都可以使用, 只是应注意系统现有的配置情况, 并给生产厂家留有适当的余地, 以免发生 I/O 地址重叠和冲突。

表 1-2 PC/XT I/O 端口地址分配表

分 类	实际使用地址	外设 I/O 端口
系统板用	000~00F	DMA 控制器 8237A
	020~021	中断控制器 8259A
	040~043	定时器/计数器 8253A
	060~063	并行接口 8255A
	080~083	DMA 页面寄存器
	0A0	非屏蔽中断屏蔽寄存器
	0C0~1FF	保留
扩充插槽用	200~20F	游戏卡接口
	210~217	扩充部件
	218~2F7	保留
	2F8~2FF	异步通信卡(二)
	300~31F	实验卡(供用户使用)
	320~32F	硬盘卡
	330~377	保留
	378~37F	并行打印卡
	380~38F	SDLC 通信卡
	390~3AF	保留
	3B0~3BF	单色显示/打印卡
	3C0~3CF	保留
	3D0~3DF	彩显/图形卡
	3E0~3EF	保留
	3F0~3F7	软盘卡
	3F8~3FF	异步通信卡(一)

(二) I/O 端口地址译码

为了便于 CPU 识别和确认与之交换信息的 I/O 端口, 必须对 I/O 端口地址进行正确译码。译码方法较多, 通常使用固定式和开关可选式两种。

1. 固定式端口地址译码

所谓固定式译码是指端口地址不可更改。这是最简单最基本的端口地址译码方式。目前大多数接口卡中都采用这种译码方式。

按照译出端口地址多少不同又可分为：用门电路进行译码和用译码器进行译码两种方法。

如果仅需要一个端口地址，则可以采用与门、与非门及或非门等门电路来构成简单的译码电路。

用译码器进行端口地址译码时，可以译出多个端口地址。可供选用的译码器芯片型号和种类很多，如：3-8 译码器 74LS138 和 Intel 8205，双 2-4 译码器 74LS139 和 74LS155，4-16 译码器 74LS154 等。

图 1-3 所示为 PC/XT 系统板上 I/O 端口地址译码电路。该译码电路采用一片 74LS138 芯片，由 $A_5 \sim A_0$ 五根地址线参加译码，分别产生 DMA 控制器 8237、中断控制器 8259、并行接口 8255 和定时器/计数器 8253 等片选信号 $\overline{CS}$ 。地址总线的低 4 位 $A_3 \sim A_0$ 作为各芯片内部寄存器的地址选择线。由 74LS138 的译码原理和所连地址线可知，当进行 DMA 操作时，因 $\overline{AEN} = 1$ ， $A_5 = 0$ ， $A_6 = 0$ ，8237 芯片的端口地址范围为：000~01FH。同理，中断控制器 8259 芯片的端口地址范围是 020~03FH。与表 1-2 所列端口地址分配一致。

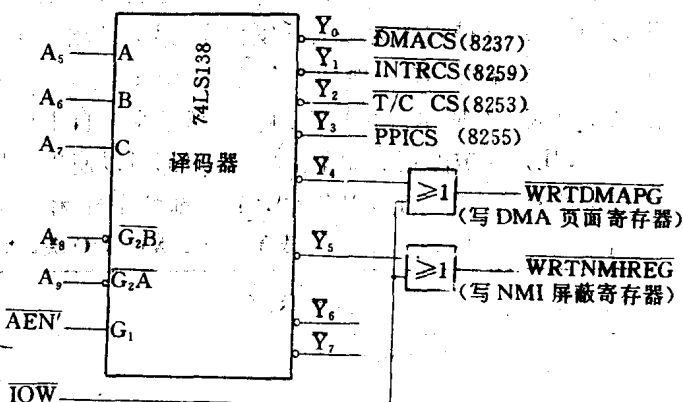


图 1-3 PC/XT 系统板上 I/O 地址译码电路

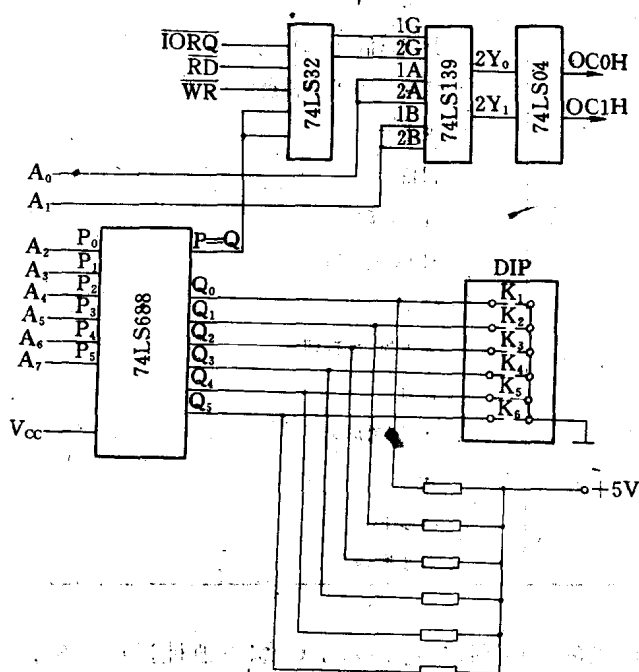


图 1-4 开关可选式译码电路

2. 开关可选式端口地址译码

如果应用场合要求 I/O 端口地址是可以变化的，则可采用开关可选式地址译码方法。这种译码方法可以不用改动译码线路，而只需改变跳线开关的组合状态从而达到更改 I/O 端口地址的目的。图 1-4 所示为一开关可选式端口地址译码电路。该电路中使用了一片 8 位比较器芯片 74LS688，当其输入端 $P_0 \sim P_5$ 的地址与设置端 $Q_0 \sim Q_5$ 的跳线开关的状态一致时， $P=Q$ 端才有选中输出。所以，只要改变跳线开关 DIP 的短路块 $K_1 \sim K_6$ 的是否插上，即可产生不同的地址输出信号。例如，按图中地址总线和有关芯片的接法，当短路块 K_1, K_2 插上 (ON)， K_3, K_4, K_5 和 K_6 均拔掉 (OFF) 时，由 74LS04 输出的两个端口

地址分别为:0C0H 和 0C1H。显然,当短路块按另一种组合插接时,则会输出另外的两个地址。

端口地址译码的方法很多,除了上述用门电路、译码器、比较器等芯片组成译码电路外,还可以用可程序逻辑阵列 PLA 和通用阵列逻辑 GAL 芯片组成译码电路。这时便不用通过改变跳线开关状态,而只需通过修改其中的程序便可改变译码器输出地址。使用灵活方便,特别适用于产品开发阶段和小批量的生产系统中。

GAL 芯片应用十分广泛,常用 GAL 芯片有 16V8、20V8 和 39V18 等。其中前两种称为普通型 GAL 器件,它们的“与”门阵列是可编程的,“或”门阵列是固定连接的。而 39V18 的“与”门阵列和“或”门阵列都是可编程的,被称之为新一代 GAL 器件。

第三节 输入输出数据传送方式

微机是通过接口电路实行与外设之间进行信息交换的。所以,CPU 对外设的控制,即对输入输出数据传送的管理,也可以说是 CPU 对信息流的调度策略问题。由于外部设备种类繁多,而且不同的输入输出设备的工作速率往往相差很大。这就要求 CPU 对不同类型的外设采用不同的输入输出处理方式。

通常 CPU 与外设之间进行数据传送有三种控制方式:即程序查询方式、中断控制方式和直接存储器存取 (DMA) 方式。图 1-5 分别简要地表示了它们的基本原理,下面将予以分别介绍。

一、程序查询方式

在这种数据传送方式中,CPU 与外设之间的数据传送,完全由程序控制进行。按 I/O 设备的特点不同,又可以分为:无条件传送方式和条件传送方式两种。

(一) 无条件传送方式

这种传送方式主要用于外设的各种动作时间为已知或固定的场合。也就是说,当 CPU 执行输入输出指令时,外设必定已准备好了。通常把有关输入输出指令插在程序中间,当程序执行到 I/O 指令时,便进行输入或输出数据。

图 1-6a 为无条件传送的输入方式

首先,认为外设已把数据送到了三态缓冲器的输入端。接着,CPU 执行输入指令,此时地址译码器输出与控制信号 $\overline{IORQ}$ 和 $\overline{RD}$ 共同作用,产生三态缓冲器的片选信号 $\overline{CE}$,从而把外设数

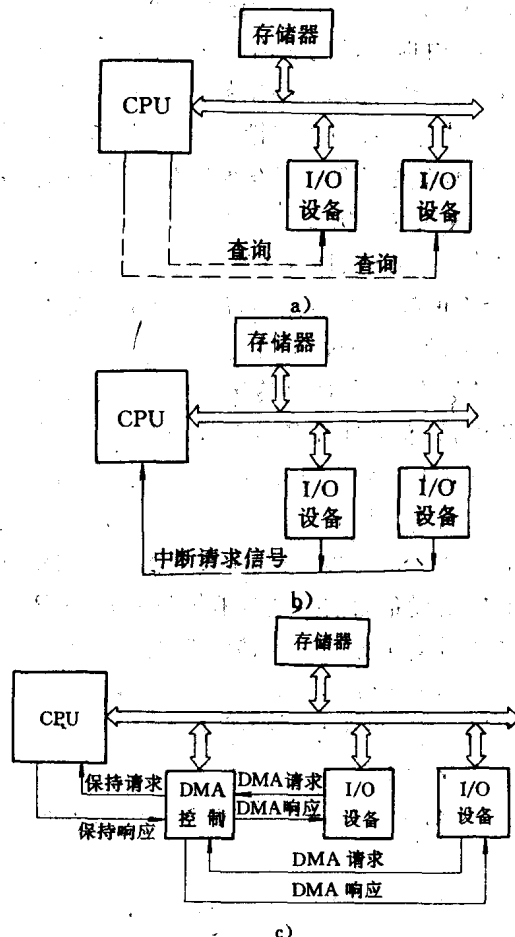


图 1-5 I/O 控制的三种方式

a) 程序查询 b) 中断控制 c) DMA

据经数据总线取入 CPU 中。

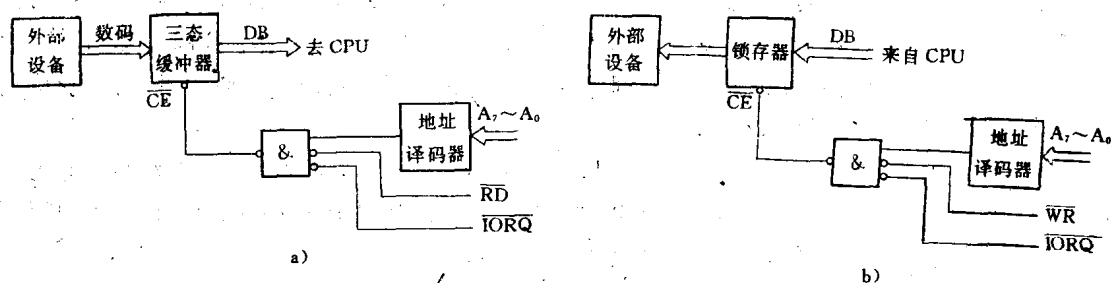


图 1-6 无条件传送方式

a) 输入方式 b) 输出方式

对于象继电器接点这样的输入设备,输入的数据一直处于有效的状态,CPU 随时可以用输入指令将接点状态取入 CPU 寄存器中。

图 1-6 b 给出了无条件传送的输出方式。首先,CPU 执行输出指令,将输出数据送到锁存器的输入端,由地址译码器输出和控制信号 $\overline{\text{IORQ}}$ 与 $\overline{\text{WR}}$ 共同作用,选通该外设接口的数据锁存器,实现将 CPU 送来的输出数据锁存,接着送给外部设备。很明显,CPU 执行输出指令时,必须确认所选外设接口数据锁存器是空的,表示可以接收新的信息。象字符显示器之类的输出设备,CPU 不需要查询外部设备是否处于准备就绪,随时可以用输出指令将信息送出显示。

(二) 条件传送方式

条件传送方式,又称程序查询传送方式。这种传送方式的特点是,所有的输入输出操作,都是处在正在执行的程序的控制之下,外部设备完全处于被动的地位。

条件传送方式下,数据传送前 CPU 必须对外部设备的状态进行测试,只有当外设确实准备好了才进行数据传送。如果没有准备好,CPU 便反复查询等待。程序查询传送方式的程序流程图如图 1-7 所示。

试看一个 Z80 CPU 用程序查询输出的简单例子。

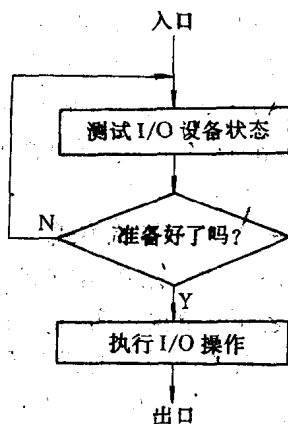


图 1-7 程序查询传送方式流程图

设某输出设备的数据端口地址为 92H。CPU 用程序查询方式向该外设输出数据。若该设备处于“空闲”状态,则 CPU 将内存 2500H 单元的内容输出给外设;若设备处于“忙”状态,则程序等待,直至设备“空闲”为止。

查询程序如下:

```
START: LD    A,      (2500H) ;取输出单元内容
        LD    C,      90H    ;状态端口地址为 90H
LOOP:  IN     B,      (C)    ;读入外设状态
        JP    Z,      LOOP   ;外设“忙”,等待
        OUT   (92H),  A      ;外设“闲”,输出数据
        HALT
```

上述程序中设定外设状态端口地址为 90H。假设当状态端口输入为 01H 时,表示外设端口数据寄存器已“空”,可以接收新的数据;若状态端口输入为 00H,则表示数据端口处于“忙”

态,不允许接收新数据。程序用 IN B, (C) 来查询外设状态, 而用 JP Z, LOOP 指令来决定程序是继续等待还是输出新数据。

二、 中断控制方式

程序查询传送方式的缺点在于: CPU 必须在程序循环中等待外部设备准备就绪后才能传送数据, 这就浪费了 CPU 许多宝贵时间, 大大降低了微处理器的效率。许多外部设备, 如光电输入机、打印机、键盘等, 都是毫秒级的慢速设备, 而 CPU 真正用于处理数据输入输出的时间, 只占用几十个微秒, 绝大多数时间都用于查询等待。所以, 为了提高 CPU 的利用率, 往往采用中断控制的传送方式。

中断控制传送方式的基本思想是: 当 CPU 需要输入或输出数据时, 先执行一条指令, 发出启动外部设备工作的命令。然后, CPU 便继续执行它原来的主程序。待被启动的外设准备好了(即输入时, 外设的输入数据已放入数据缓冲器; 输出时, 外设已把上一数据输出, 输出寄存器已空出来)以后, 便向 CPU 发中断申请信号。CPU 在接收到该中断申请信号以后, 便暂时停止正在执行的原程序, 转去对已经准备好的数据进行输入或输出操作。待输入或输出操作完成之后, CPU 再返回断点, 继续执行原来的程序。

在中断控制方式中, 如果有多个输入输出设备要和 CPU 进行信息交换, 则每一个设备的中断请求信号线, 都要连接到 CPU 的中断输入线上。每当输入或输出设备准备好了数据要进行输入或输出时, 便通过中断请求线向 CPU 发中断申请信号。因此, 这是一种以硬件方法来直接影响和改变 CPU 操作顺序的工作方式。外部设备能迫使 CPU 暂停它正在执行的现行任务, 转而去执行另外一种叫做中断服务程序的任务, 这个程序能满足外部设备的工作需要。

由上分析可以看出, 利用中断控制方式进行数据传送有如下几个优点:

(1) 实现分时操作 CPU 可以自己处理大量数据, 只有在外部设备数据准备就绪时, 才向 CPU 发中断申请信号。这样使用一台微处理器, 可以同时处理多台外部设备的信息。这既可以节省 CPU 的时间, 提高了系统的效率, 又可以使输入输出设备的服务请求得到及时的响应。

(2) 实现实时控制 微机用于生产过程控制时, 现场的各种参数、信息可以在任何时间内发出中断申请, 要求 CPU 处理, CPU 都能及时响应, 实现实时控制。

(3) 故障处理 微机在运行过程中, 会出现某些事先预料不到的情况或故障, 如电源突然掉电, 存储器出错, 运算溢出等。这些都可以利用中断系统自行进行处理。

三、 直接存储器存取(DMA)方式

利用中断控制方式进行数据传送, 一方面改善了 CPU 对输入输出设备的响应速度, 使在任何情况下不致丢失信息; 另一方面又大大提高了 CPU 的利用率。

但是, 从数据传送的角度看, 中断控制方式实质上仍是由 CPU 通过程序来进行的。而且每传送一个数据, 都要经历暂停主程序——保护断点和现场——执行中断服务程序——恢复现场——返回主程序的过程。这里每一步都要用到多条指令, 每条指令都要取指令和执行指令的时间。所以通常传送一个字节数据需要几十到几百微秒时间。这使得在有些情况下就不能使用这种方式来传送数据。一种情况是, 对于一些高速外设, 如读硬磁盘, 如果每次不能在规定的时间内把数据读完, 就会丢失数据; 另一种情况是, 要求成组地交换大批数据的场合, 如磁带机与内存交换数据时, 用中断控制方式也是不合适的。

所以, 必须寻求一种新的更高速度的输入输出数据传送方式——直接存储器存取方式, 简称 DMA 方式。