



高等学校“十二五”重点规划教材  
信息与自动化系列

1101 010 001 011010

# 微机接口原理及应用技术

主 编 蔡成涛 梁燕华 王立辉  
主 审 朱齐丹

HEUP 哈尔滨工程大学出版社  
Harbin Engineering University Press

# 微机接口原理及应用技术

主编 蔡成涛 梁燕华 王立辉

主审 朱齐丹

哈尔滨工程大学出版社

## 内 容 简 介

本书从应用的角度出发,系统全面地介绍了微机系统中的并行/串行接口、输入/输出人机交互接口、定时/计数器接口、A/D 与 D/A 接口、中断控制接口、DMA 接口以及微机系统中常用的功率接口的基本原理及应用技术,对接口技术使用的典型接口芯片原理及应用要点进行了介绍。全书重点讲述各种 I/O 接口的硬件、软件设计技术,注重理论分析与实际应用相结合,内容简明、实用、丰富。全书结构紧凑,语言通俗,具有一定的系统性和实用性。

本书可作为高等学校自动化、通信工程、机电一体化专业本科生和工科类其他相关专业的教材,也可供从事微机系统设计和应用的技术人员自学和参考。

## 图书在版编目(CIP)数据

微机接口原理及应用技术/蔡成涛,梁燕华,王立  
辉主编. —哈尔滨:哈尔滨工程大学出版社,2011.8  
ISBN 978-7-5661-0256-0

I. ①微… II. ①蔡… ②梁… ③王… 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(2011)第 182818 号

---

出版发行 哈尔滨工程大学出版社  
社 址 哈尔滨市南岗区东大直街 124 号  
邮政编码 150001  
发行电话 0451-82519328  
传 真 0451-82519699  
经 销 新华书店  
印 刷 黑龙江省教育厅印刷厂  
开 本 787mm×1092mm 1/16  
印 张 13  
字 数 307 千字  
版 次 2011 年 8 月第 1 版  
印 次 2011 年 8 月第 1 次印刷  
定 价 25.00 元

<http://press.hrbeu.edu.cn>

E-mail: [heupress@hrbeu.edu.cn](mailto:heupress@hrbeu.edu.cn)

---

# 前 言

“微机接口原理及应用技术”课程是自动化专业、电子通信工程专业、机电一体化专业本科生必修的一门专业基础课程,也是当今电子类和工程类各专业学生的一门重要的选修课程。本书适应面广,可作为工科相关专业的教材,也可作为计算机技术等级考试的参考资料,还可供从事微机系统设计和应用的技术人员自学和参考。

“微机接口原理及应用技术”是微机原理及微型处理器技术的后续课程,在本书编撰过程中,将重点放在接口技术原理及应用上,对 8086 处理器系统结构、指令系统、微机基本原理及应用等技术不作过多讨论,而是以在微机系统应用过程中使用最多的并行/串行接口、输入/输出人机交互接口、定时/计数器接口、A/D 与 D/A 接口以及功率接口的基本原理及应用为主线,全面而翔实地讨论各部分接口的结构、使用方法。同时,为了尽可能保证接口技术的完整性,对中断控制器接口和 DMA 接口也作了简要介绍。本书在使用过程中要求读者掌握 8086 微处理器、单片机或相关微处理器的结构组成、工作原理及使用技术,并能够运用汇编语言书写简单应用程序。

全书结构紧凑、章节编排合理,具有一定的通用性、系统性和实用性,内容介绍深入浅出、通俗易懂。全书共分 9 章,第 1 章为微机接口技术基础概念,第 2~9 章为接口技术原理及使用方法介绍,在各章节的编排中,每章以一项接口技术为核心,从基本概念、技术原理、接口芯片使用方法及典型应用几个方面进行介绍,每一章均可独立学习。本书由蔡成涛,梁燕华,王立辉主编。其中,第 1~3 章由蔡成涛编写,第 4~7 章由梁燕华编写,第 8 章、第 9 章由王立辉编写,全书由朱齐丹主审。

本书在编写过程中,参考或引用了国内一些专家学者的论著,在此表示衷心感谢!由于作者的水平有限,书中的错误及疏漏之处在所难免,敬请读者批评指正。

编 者

2011 年 6 月于哈尔滨工程大学

# 目 录

第 1 章 微机接口技术概述	1
1.1 微型计算机基础	1
1.2 微机接口概念	8
1.3 I/O 同步控制方式	11
1.4 微机接口技术特点及学习方法	15
第 2 章 并行通信接口技术	17
2.1 并行通信和并行接口	17
2.2 可编程并行接口电路 8255	18
2.3 8255 应用举例	33
第 3 章 串行通信接口技术	37
3.1 串行通信基本概念	37
3.2 串行通信数据差错控制	41
3.3 串行通信标准接口	44
3.4 可编程串行接口芯片 16550	52
第 4 章 人机交互接口技术	62
4.1 LED/LCD 显示器接口	62
4.2 键盘接口	76
4.3 专用键盘显示接口芯片 8279	85
4.4 专用键盘显示接口芯片 8279 应用举例	92
第 5 章 定时/计数器接口技术	97
5.1 定时与计数	97
5.2 Intel 8253/8254 可编程定时/计数器	99
5.3 8253/8254 的应用举例	113
5.4 实时时钟接口 Motorola 146818	115
第 6 章 D/A 与 A/D 接口技术	122
6.1 D/A 转换器	123
6.2 A/D 转换器	137
第 7 章 中断控制器接口技术	157
7.1 中断的基本概念	157
7.2 可编程中断控制器 8259A	158
7.3 8259A 应用举例	171
第 8 章 DMA 接口技术	173
8.1 DMA 技术概述	173

8.2	DMA 控制器 8237 .....	174
8.3	8237 编程及应用 .....	183
<b>第 9 章</b>	<b>微机系统功率接口技术</b> .....	<b>187</b>
9.1	常用功率器件 .....	187
9.2	开关型功率接口 .....	192
<b>参考文献</b>	.....	<b>202</b>

# 第 1 章 微机接口技术概述

## 1.1 微型计算机基础

介绍微型计算机,就不能不谈微处理器。

微处理器本身不是计算机,它是小型计算机或微型计算机的控制和处理部分。

微型计算机,简称微机,它是具有完整运算及控制功能的计算机,它除了包括微处理器外,还包括存储器、接口适配器以及输入输出(I/O)设备等。如图 1-1 所示为微机的各组成部分。其中微处理器由控制器、运算器和若干寄存器组成;输入输出(I/O)设备与微处理器需要通过接口适配器连接(即 I/O 接口);存储器是指微机内部的存储器(RAM,ROM 和 EPROM 等芯片)。

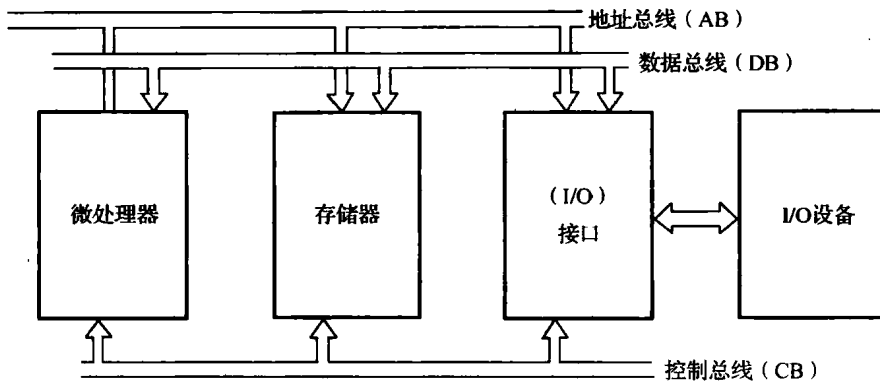


图 1-1 微机各组成及其连接关系

微机系统中常用的单片机是指将微处理器、一定容量的 RAM 和 ROM 以及 I/O 接口、定时器等电路集成在一块芯片上,构成的单片微型计算机。

### 1.1.1 微处理器的组成

微处理器包括两个主要部分:运算器和控制器。

#### 1. 运算器

运算器由运算部件——算术逻辑单元(Arithmetic and Logical Unit,简称 ALU)、累加器和寄存器等几部分组成。算术逻辑单元的作用是把传送到微处理器的数据进行算术或逻辑运算。算术逻辑单元具有两个主要的输入来源,一个是累加器,另一个是数据寄存器,它能够完成这两个输入数据的相加或相减运算,也能够完成某些逻辑运算。算术逻辑单元执行不同的运算操作是由不同控制线上的信号所确定的。运算器的两个主要功能如下:

(1) 执行各种算术运算。

(2) 执行各种逻辑运算,并产生逻辑测试,如零值测试或两个值的比较。

通常,一个算术操作产生一个运算结果,而一个逻辑操作产生一个判决。

## 2. 控制器

控制器由程序计数器、指令寄存器、指令译码器、时序发生器和操作控制器等组成,是发布命令的“决策机构”,即协调和指挥整个计算机系统的操作。控制器的主要功能如下:

(1) 从内存中取出一条指令,并指出下一条指令在内存中的位置。

(2) 对指令进行译码或测试,并产生相应的操作控制信号,以便执行规定的动作。比如一次内存读/写操作,一个算术/逻辑运算操作或一个输入/输出操作等。

(3) 指挥并控制 CPU、内存和输入/输出设备之间数据流动的方向。

相对控制器而言,运算器接收控制器的命令而进行动作,即运算器所执行的全部操作都是由控制器发出的控制信号来指挥的。

ALU、计数器、寄存器和控制部分除在微处理器内通过内部总线相互联系以外,还通过外部总线与外部的存储器和输入/输出接口电路联系。外部总线一般分布数据总线 DB、地址总线 AB 和控制总线 CB,统称为系统总线。存储器包括 RAM 和 ROM。微型计算机通过输入/输出接口电路可与各种外围设备连接。

## 3. CPU 的主要寄存器

### (1) 累加器(A)

累加器是微处理器中最繁忙的寄存器。在算术和逻辑运算时,它具有双重功能:运算前,用于保存一个操作数;运算后,用于保存所得的和、差或逻辑运算结果。

### (2) 数据寄存器(DR)

数据(缓冲)寄存器通过数据总线向存储器和输入/输出设备送(写)或取(读)数据的暂存单元。它可以保存一条正在译码的指令,也可以保存正在送往存储器中存储的一个数据字节等。

### (3) 指令寄存器(IR)及指令译码器(ID)

指令寄存器用来保存当前正在执行的一条指令。当执行一条指令时,先把它从内存取到数据寄存器中,然后再传送到指令寄存器中。指令分为操作码和地址码字段,由二进制数字组成。为执行给定的指令,必须对操作码进行译码,以便确定所要求的操作。指令译码器就是负责这项工作的。指令寄存器中操作码字段的输出就是指令译码器的输入。操作码经译码后,即可向操作控制器发出具体操作的特定信号。

### (4) 程序计数器(PC)

为了保证程序能够连续地执行下去,CPU 必须采取某些手段来确定下一条指令的地址。程序计数器正是起到了这种作用,所以通常又称其为指令地址计数器。在程序开始执行前,必须将其起始地址及程序的第一条指令所在的内存单元地址送入 PC,当执行指令时,CPU 将自动修改 PC 的内容,使之总是指示出将要执行的下一条指令的地址。由于大多数指令都是按顺序执行的,所以修改的过程通常只是简单的加 1 操作。

### (5) 地址寄存器(AR)

地址寄存器用于保存当前 CPU 所要访问的内存单元或 I/O 设备的地址。由于内存和



CPU 之间存在着速度上的差别,所以必须使用地址寄存器来保持地址信息,直到内存读/写操作完成为止。

显而易见,当 CPU 和内存进行信息交换时,都要使用地址寄存器和数据寄存器。同样,如果把外围设备的地址作为内存地址单元来看待的话,那么,当 CPU 和外围设备交换信息时,也需要使用地址寄存器和数据寄存器。

### 1.1.2 存储器的组成及分类

存储器按大类,主要分为 RAM 和 ROM。这里论述的 RAM 和 ROM 是半导体存储器,是一种采用大规模或超大规模集成电路工艺制成的存储器芯片,具有体积小、质量轻、集成度高等特点。

RAM(Random Access Memory)存储器是一种在正常工作时既能读又能写的存储器,故它通常用来存放原始数据、中间结果、最终结果和实时数据等。RAM 中存入的信息不能长久保存,掉电后便立即消失,故它又称为易失性存储器。

ROM(Read Only Memory)存储器是一种在正常工作时只能读不能写的存储器,故它通常用来存放固定程序和常数。固定程序和常数是利用特殊手段存放进去的,一旦写入便能长久保存,需要时便可读出使用。

目前,随着科学技术的发展,不断出现新的存储器类型,图 1-2 对各种存储器按照工作原理进行了分类。

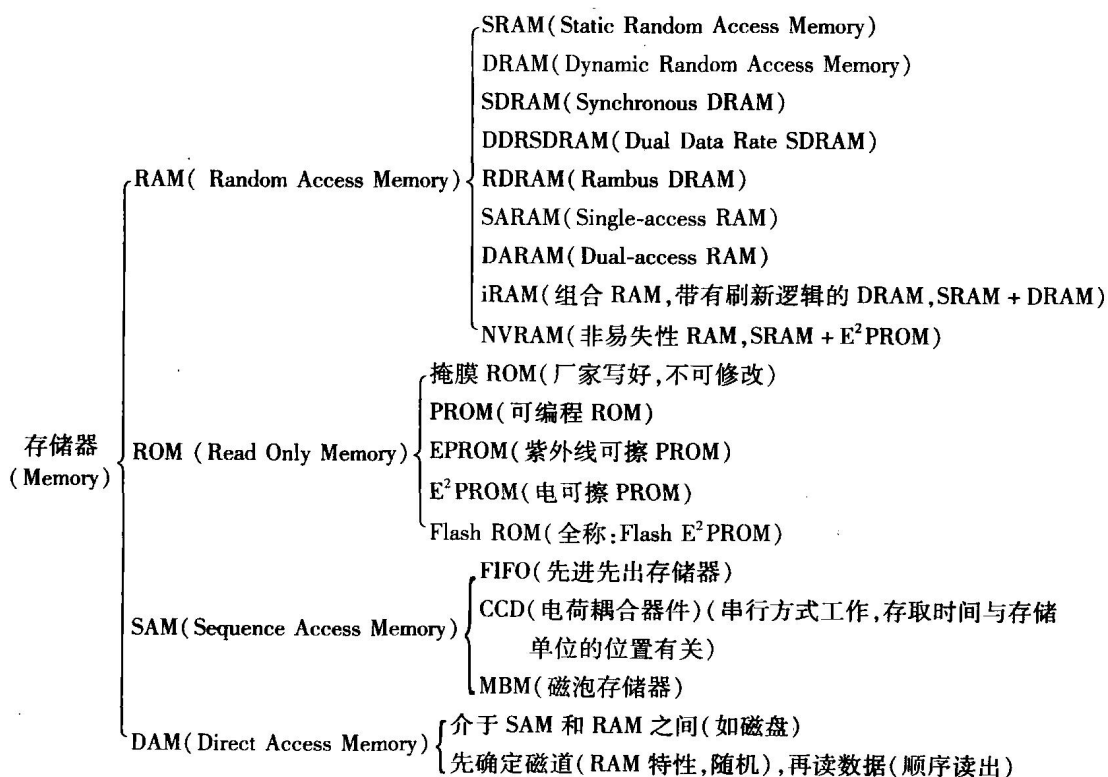


图 1-2 按照工作原理的存储器分类

由图 1-2 可知,存储器可以细分为 RAM(Random Access Memory)、ROM(Read Only Memory)、SAM(Sequence Access Memory)和 DAM(Direct Access Memory)四大类。

### 1. RAM(Random Access Memory)

#### (1) SRAM 和 DRAM

SRAM 是英文 Static Random Access Memory 的缩写,DRAM 是英文 Dynamic Random Access Memory 的缩写。这是根据内存的工作原理划分出的两种内存。DRAM 表示动态随机存取存储器,是一种以电荷形式进行存储的半导体存储器。DRAM 中的每个存储单元由一个晶体管和一个电容器组成,数据存储电容器中。电容器会由于漏电而导致数据丢失,因而 DRAM 器件是不稳定的。为了将数据保存在 DRAM 类型的存储器中,DRAM 器件必须有规律地进行刷新。而 SRAM 是静态随机存取存储器,因此只要供电,它就会保持一个值。一般而言,SRAM 比 DRAM 的读写速度要快,这是因为 SRAM 没有刷新周期。但每个 SRAM 的每个存储单元一般由 6 个晶体管组成,可以断定在给定的固定区域内,DRAM 的密度比 SRAM 的密度要大。因此,SRAM 常常用于高速缓冲存储器,因为它有更高的速率。而 DRAM 常常用于 PC 中的主存储器,因为其拥有更高的密度。

#### (2) SDRAM, DDRSDRAM 和 RDRAM

SDRAM 是英文 Synchronous Dynamic Random Access Memory 的缩写,DDRSDRAM 是英文 Dual Data Rate Synchronous Dynamic Random Access Memory 的缩写,RDRAM 是英文 Rambus Dynamic Random Access Memory 的缩写。上述三种存储器是针对当前微机内存种类划分的,它们都属于上面提到的 DRAM。

SDRAM 中文名字是“同步动态随机存储器”,意思是指理论上其速度可达到与 CPU 同步。自 Pentium 时代以来,SDRAM 一直都是微机系统中内存名称的代名词。由于其最初的标准是采用将内存与 CPU 进行同步频率刷新的工作方式,因此,基本上消除了等待时间,提高了系统的整体性能。DDRSDRAM 从名称上可以看出,这种内存存在技术上与 SDRAM 有着密不可分的关系。事实上,DDRSDRAM 就是 SDRAM 的加强版,它主要是利用时钟脉冲的上升沿与下降沿传输数据,相当于原来两倍频率的工作效率。RDRAM 原本是 Intel 强力推广的未来内存发展方向,其技术引入了 RISC(精简指令集),依据高时钟频率来简化每个时钟周期的数据量。RDRAM 也是采用类似于 DDR 的双速率传输结构,同时利用时钟脉冲的上升与下降沿进行数据传输。由于这种内存是全新的结构体系,需要兴建专用的生产线才能进行大批量生产,而且生产这种存储器还必须按产量向 Rambus 公司交纳一定的专利金,因此在一定程度阻碍了 RDRAM 的发展。

#### (3) SARAM 和 DARAM

SARAM 是英文 Single Access Random Access Memory 的缩写,DARAM 是英文 Dual Access Random Access Memory 的缩写。这种分类是按 CPU 每个机器周期能对存储器进行访问的次数来划分的。SARAM 在一个机器周期内只能被访问一次,而 DARAM 则在一个机器周期内能被访问两次。

#### (4) iRAM

iRAM 为组合 RAM,是一种在片内集成了刷新控制电路的 DRAM,这种存储器芯片既集

中了 DRAM 存储器密度高、功耗低的特点,又避免了刷新控制接口的设计,同时又具有 SRAM 的特性。一般来说其芯片引脚和用法与 SRAM 一样,故又称其为 PSRAM (Pseudo Static RAM)。

### (5) NVRAM

NVRAM 是一种非易失性的随机读写存储器,既能快速存取,系统断电时又不丢失数据。实际上,它是把 SRAM 的实时读写功能与 EEPROM 的可靠非易失能力综合在一起。在芯片数据保护下降到一定数值时,NVRAM 存储器芯片能迅速将 SRAM 中的内容并行地转移到 EEPROM 中,当电源恢复后,EEPROM 中的内容又自动放入 SRAM 阵列中,对于一般的 NVRAM 存储器芯片,这种转存操作能可靠进行数万次,而 EEPROM 中的内容能保证数据存储 10 年以上。

## 2. ROM(Read Only Memory)

### (1) 掩膜 ROM

掩膜 ROM 的编程是由半导体制造厂家在生产过程中进行的,掩膜 ROM 制造完成后不能再更改其内容。

### (2) PROM

PROM 是英文 Programmable ROM 的缩写,称之为可编程程序只读存储器。PROM 内部有行列式的熔丝,视需要利用电流将其烧断,写入所需的资料,但仅能写入一次。

### (3) EPROM

EPROM 是英文 Erasable Programmable Read Only Memory 的缩写,称之为可擦除可编程只读存储器。EPROM 可利用高电压将资料编程写入,擦除时将线路曝光于紫外线下,则资料可被清空,并且可重复使用。通常在封装外壳上会预留一个石英透明窗以方便曝光。

### (4) EEPROM

EEPROM 是英文 Electrically Erasable Programmable Read Only Memory 的缩写,称之为电子式可擦除可编程只读存储器。其原理类似 EPROM,但是擦除的方式是使用高压电场来完成,因此不需要透明窗。

### (5) Flash ROM

Flash ROM 称为快闪存储器,是一种长寿命非易失性的存储器,它是 EEPROM 的变种。它与 EEPROM 的不同点是:它能在字节水平上进行删除和重写而不是整个芯片擦写,因此,Flash ROM 就比 EEPROM 的更新速度快。目前常用的 U 盘、SD 卡、CF 卡等均为快闪存储器,简称为闪存。

## 3. SAM(Sequence Access Memory)

SAM 为顺序存储器,存储器所存信息的排列、寻址和读写操作均是按顺序进行的,并且存取时间与信息在存储器中的物理位置有关。在这种存储器中,如磁带存储器,信息通常以文件或数据块形式按顺序存放,信息在载体上没有唯一对应的地址,完全按顺序存放或读取。

### (1) FIFO

FIFO 是英文 First In First Out 的缩写,是一种先进先出的数据缓存器,它与普通存储器

的区别是没有外部读写地址线,这样使用起来非常简单,但缺点就是只能顺序写入数据,顺序读出数据,其数据地址由内部读写指针自动加 1 完成,不能像普通存储器那样可以由地址线决定读取或写入某个指定的地址。FIFO 一般用于不同时钟域之间的数据传输,另外,对于不同宽度的数据接口也可以用 FIFO。FIFO 的宽度和深度是其两个重要参数:FIFO 的宽度指的是 FIFO 一次读写操作的数据位;FIFO 的深度指的是 FIFO 可以存储多少个 N 位的数据(如果宽度为 N)。根据 FIFO 工作的时钟,可以将 FIFO 分为同步 FIFO 和异步 FIFO。同步 FIFO 是指读时钟和写时钟为同一个时钟,在时钟沿来临时同时发生读写操作。异步 FIFO 是指读写时钟不一致,读写时钟是互相独立的。

### (2) CCD

CCD 是英文 Charge Coupled Device 的缩写,称之为电荷耦合器件,目前广泛应用于数字成像相机中。它由一种高感光度的半导体材料制成,能把光线转变成电荷,通过模/数转换器转换成数字信号,数字信号经过压缩以后由相机内部的闪存存储器或内置硬盘卡保存,因而可以轻而易举地把数据传输给计算机,并借助于计算机的处理手段,根据需求和想象来修改图像。CCD 由许多感光单位组成,通常以百万像素为单位。当 CCD 表面受到光线照射时,每个感光单位会将电荷反映在组件上,所有的感光单位所产生的信号加在一起,就构成了一幅完整的画面。将 CCD 作为顺序存储器的一种,是由于 CCD 数据的读出是按照一定顺序输出的。如根据数据输出格式不同分为全帧转移 CCD(Full Frame CCD)、帧转移 CCD(Frame Transfer CCD)及行间转移 CCD(Interline Transfer CCD)。

### (3) MBM

MBM 是英文 Magnetic Bubble Memory 的缩写,称之为磁泡传感器。利用在磁性薄膜的某一位置上“有”和“无”磁泡的两种物理状态代表“1”和“0”,可实现信息的存储。控制磁泡的产生、消灭、移动和检出等可实现信息的写入、传输和读出。利用磁泡间的排斥作用还可以实现逻辑功能。用磁泡存储、处理信息的技术称为磁泡技术。磁泡存储器的优点是具有非易失性,存储密度高,可靠性高,无高速旋转的机械部分,适合在运动条件下工作;缺点是速度慢,取数时间是数毫秒,比磁盘稍快,但较半导体存储器慢得多。

## 4. DAM(Direct Access Memory)

直接存储器既不像 RAM 那样能随机地访问任何存储单元,也不像 SAM 那样完全按顺序存取,它是介于 RAM 与 SAM 之间的一种存储器。目前广泛使用的磁盘就属于直接存取存储器,当要存取所需信息时,它要进行两个逻辑动作,第一步为寻道,使磁头指向被选磁道,第二步为在被选磁道上顺序存取。

### 1.1.3 微型计算机系统结构

以微处理器为核心的微型计算机,尽管种类繁多,型号各异,但它们的内涵基本相同,与一般的计算机并无本质上的区别。无论是简单的单片机,还是较复杂的个人计算机、各种专用系统和超级计算机,从硬件系统结构来看,它们都是由微处理器 MPU、存储器 RAM 和 ROM 以及输入输出设备 I/O 三大部分组成,它们之间用地址总线 AB,数据总线 DB,控制总线 CB 联系在一起,如图 1-3 所示。因此微型计算机的系统结构也称为三总线结构,简称

总线结构。

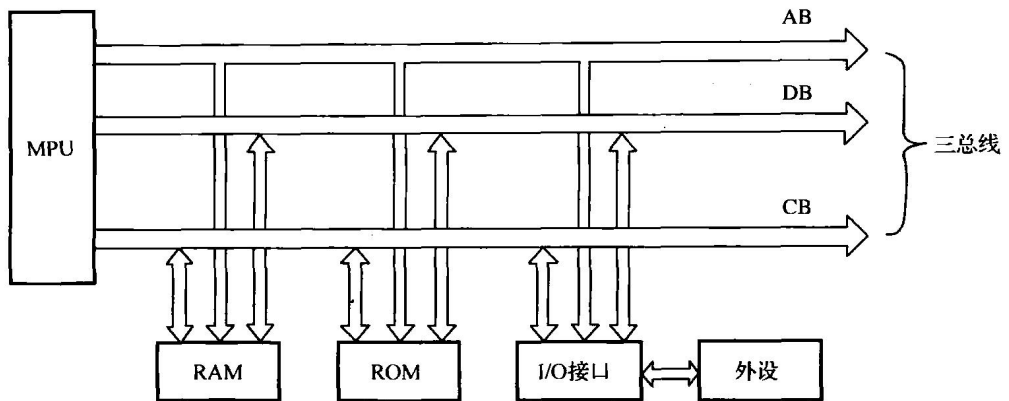


图 1-3 微机总线结构

所谓总线是指信息传送的公共通道,是沟通微机系统中各器件的桥梁。总线结构实质上是一种表示微机系统的各种硬件操作和复杂功能的精确模型,这些操作和功能如下:

- (1) 从 MPU 把数据写入存储器;
- (2) 从存储器把数据读入 MPU;
- (3) 从 MPU 把数据写入输出端口;
- (4) 从输入端口把数据读进 MPU;
- (5) MPU 中断操作;
- (6) MPU 控制的直接存储器访问 DMA;
- (7) MPU 内部寄存器操作。

#### 1. 地址总线 AB(Address Bus)

地址总线上仅传送 MPU 地址码。当微处理器 MPU 和存储器或外部设备交换信息时,必须指明要和哪个存储单元或哪个外部设备交换。因此,地址总线必须和所有存储器的地址线对应相连,也必须和所有 I/O 接口的地址线(或片选线)相连。这样,当微处理器 MPU 对存储器或外设读/写数据时,只要把存储单元或外设的地址码(或片选线)送到地址总线上便可选中它们工作。地址总线条数由所选 MPU 型号决定,表示此型号 MPU 的寻址能力。

#### 2. 数据总线 DB(Data Bus)

数据总线上传送的是数据和指令码,数据总线条数常和所用微处理器字长相等,但也有内部为 16 位运算而外部仍为 8 位数据总线的情况。由于 MPU 有时需要把数据写入存储器或从外设输出数据,有时又需要从存储器或输入设备输入数据,因此数据总线是双向的。数据总线的宽度表示系统数据的传输能力。

#### 3. 控制总线 CB(Control Bus)

控制总线用于传送各类控制信号。控制总线条数因机器而异,每条控制线最多传送两个控制信号。控制信号有两类:一类是 MPU 发出的控制命令,如读命令、写命令、中断响应信号等;另一类是存储器或外设的状态信息,如外设的中断请求、复位、总线请求和中断请求等。

## 1.2 微机接口概念

### 1.2.1 微机接口的作用

为了完成一定的实际任务,微型计算机必须与外部客观世界进行信息交换和传输,故必须通过接口来完成。但是从图 1-3 可以看出,存储器可以直接和系统总线相连,不需要接口,而输入输出设备(外设)却一定要通过接口与总线相连,这是因为外部设备的输入/输出操作和存储器读/写操作不同。众所周知,存储器都是用来保存信息的,功能单一;传送方式也单一,一次必定是传送 1 个字节或者 1 个字,品种有限,只有读类型和可读/可写类型;此外,存储器的存取速度基本上可以和 CPU 的工作速度匹配。这些决定了存储器可以通过总线和 CPU 相连,即通常说的直接将存储器挂在系统总线上。但是,外部设备的功能却是多种多样的。有些外设作为输入设备,有些外设作为输出设备,也有些外设既作为输入设备又作为输出设备,还有一些作为检测设备或控制设备,而每一类设备本身可能又包括了多种工作原理不同的具体设备。对于一个具体设备来说,它所使用的信息可能是数字式的,也可能是模拟式的,而非数字式信号必须经过转换,使其成为对应的数字信号才能送到计算机总线。这种将模拟信号变为数字信号或者将数字信号变为模拟信号的功能就需要 A/D, D/A 接口来完成。

大多数外部设备所用的信息是数字式的,不过,有些外设的信息是并行的,有些外设的信息是串行的。串行设备只能接收和发送串行的数字信息,而 CPU 却只能接收和发送并行信息。这样,串行设备必须通过接口将串行信息变为并行信息,才能送给 CPU;反过来,要将 CPU 送出的并行信息变为串行信息,才能送给串行设备。要实现并行数据和串行数据之间的变换就需要串行接口来完成。

对于采用并行接口和微机系统进行连接的设备仍然需要采用接口。因为 CPU 通过总线要和多个外设打交道,而在同一个时刻 CPU 通常只和一个外设交换信息,也就是说,一个外设不能长期和 CPU 相连,只有被 CPU 选中的外设,才接收数据总线上的数据或者将外部信息送到数据总线上。所以,即使是并行设备,也同样通过接口与总线相连。

除了上面这些原因外,外设的工作速度通常比 CPU 的速度低得多,而其各种外设的工作速度互不相同,这就要求接口电路对输入/输出过程能起到一个缓冲和联络的作用。

对于输入设备来说,接口通常起信息变换和缓冲功能,变换的含义包括模拟量到数字量的变换、串行数据到并行数据的变换,以及电平变换等,总之,目的是将输入设备送来的信息转换成 CPU 能接收的格式,并将其放在缓冲器中让 CPU 来接收。对于输出设备来说,接口要将 CPU 送来的并行数据放到缓冲器中,并将它变成外部设备所需要的信息形式,这种形式可能是串行数据,也可能是模拟量等。

可见,输入/输出接口电路是为了解决计算机和外设之间的信息变换问题而提出来的,输入/输出接口是计算机和外设之间传送信息的部件,每个外设都要通过接口和主机系统相连。接口技术就是专门研究 CPU 和外设之间的数据传送方式、接口电路的工作原理和使用

方法的。

### 1.2.2 微机接口的基本功能与结构

无论哪种接口,尽管连接的外设千差万别,与外设通信的方式也不一样,但其基本功能和基本结构是相似的。

#### 1. 微机接口基本功能

任何接口电路,基本功能都主要体现在三个方面:

- (1) 作为微型机与外设间传递数据的中途缓冲站;
- (2) 正确寻址与微机交换数据的外设;
- (3) 提供微机与外设间交换数据所需的控制逻辑和状态信号。

也就是完成数据、地址、控制三总线的转换和连接任务。

为了实现上述基本功能,作为接口电路,通常必须为外设提供几个不同地址的寄存器,每个寄存器即被称为一个 I/O 端口。所以,I/O 端口实际上就是一个很小的外部存储器,每个 I/O 端口和每个存储单元一样,对应着唯一的一个地址。端口寄存器和部分端口线被连接到外设上。通常所谓的 I/O 操作,是指 I/O 端口操作,而不是 I/O 设备操作,即 CPU 访问的是与 I/O 设备相连的 I/O 端口,而不是笼统的 I/O 外设。接口电路在微机系统中的位置示意图如图 1-4 所示。

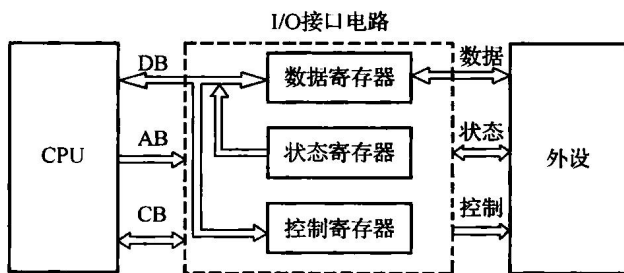


图 1-4 接口电路在微机系统中的位置示意图

根据图 1-4,这里给出微机接口的定义:微机接口指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。

#### 2. 基本结构

接口电路现在通常做在一块大规模或超大规模集成电路芯片上,因而常被称为接口芯片。当然,有时也有根据需要而用中、小规模集成电路做成的。不同功能的接口电路,其结构虽各不相同,但都是由寄存器和控制逻辑两大部分组成,每部分又都包含几个基本组成部分。接口电路基本结构如图 1-5 所示。

##### (1) 数据缓冲寄存器

数据缓冲寄存器分为输入缓冲器和输出缓冲器两种。

对输入端口而言,由于它的输出引脚直接挂在数据总线上,所以要求它必须有三态输出功能,当该端口未被选中时,其输出应处于“高阻”状态,与总线隔离;又由于外设输入到输

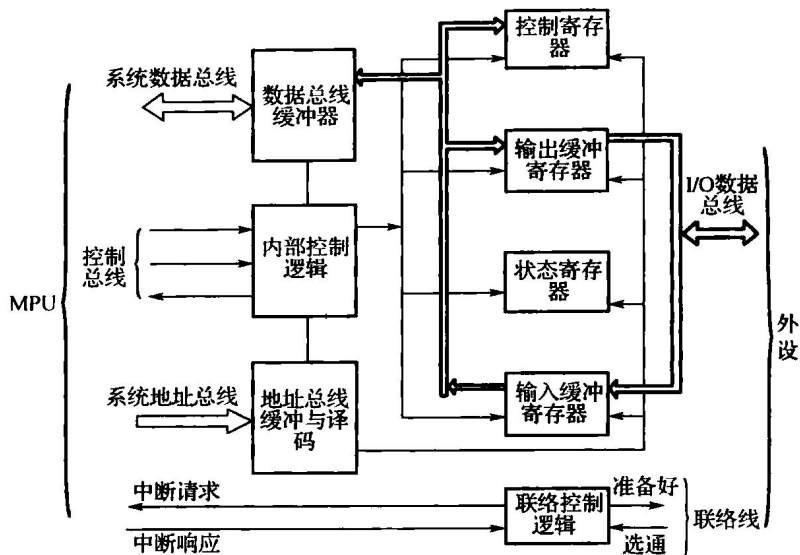


图 1-5 接口电路基本结构框图

入端口的数据与微处理器读取该端口数据往往不是同步的,因此输入时要求有相应的输入数据锁存器,利用它将来自输入设备的数据锁存起来,并一直稳定地保持到微处理器取走该数据。由此可见,输入缓冲器一般由锁存器和三态缓存器组成。

对输出端口而言,它是把来自 MPU 的数据通过系统总线输出到外部设备。MPU 的数据出现在总线上的时间很短,只是执行一条输出指令的时间,一般在毫微秒级,而外设通常不可能在这么短的时间内将总线上的数据取走,因此要求输出端口必须有数据锁存器,将输出的数据保持足够长的时间,至于输出端口是否有三态缓冲功能则无关紧要,因为一般每根输出端口线只对应于一个输出设备。

输入/输出数据缓冲器可以在高速工作的 MPU 与慢速工作的外设之间起协调、控制作用,即缓冲作用。

### (2) 控制寄存器

控制寄存器用于存放处理器发来的控制命令和其他信息,以确定接口电路的工作方式和功能。由于现在的接口芯片大都具有可编程的特点,即可通过编程来选择或改变其工作方式和功能,这样,一个接口芯片就相当于具有多种不同的工作方式和功能,使用起来十分灵活、方便。控制寄存器是写寄存器,其内容只能由处理器写入,而不能读出。

### (3) 状态寄存器

用于保存外设现行的各种状态信息。它的内容可以被处理器读出,从而使处理器“知道”数据传送过程中正在发生或最近已经发生的事情,供处理器作出正确的判断,使它能安全可靠地与接口完成交换数据的各种操作。特别当 CPU 以程序查询方式同外设交换数据时,状态寄存器更是必不可少的。CPU 通过查询外设的忙/闲、正常/故障、就绪/未就绪等状态,正确地与之交换信息。

以上三种寄存器是接口电路中的核心部分,通常所说的接口,大都指这些寄存器,它们



就是前面所说的 I/O 端口。但是,为了保证在处理器和外设之间通过接口正确地传送数据,接口电路还必须包括下面几种控制逻辑电路。

#### ① 数据总线和地址总线缓冲器

用于实现接口芯片内部总线和处理器外部总线的连接。如接口的数据总线端可直接和系统数据总线相连接,接口的端口选择端根据 I/O 寻址方式的要求与地址总线恰当地连接。

#### ② 端口地址译码器

用于正确选择接口电路内部各端口寄存器地址,保证端口寄存器唯一地对应一个端口地址码,以便处理器正确无误地与指定外设交换信息,完成规定的 I/O 操作。

#### ③ 内部控制逻辑

用于产生一些接口电路内部的控制信号,实现系统控制总线与内部控制信号之间的变换。

#### ④ 对外联络控制逻辑

用于产生/接收 MPU 和外设之间数据传送的同步信号。这些联络握手信号包括微处理器一边的终端请求和响应,以及外设一边的准备就绪和选通等控制与应答信号。

当然,并非所有接口都具备上述全部组成部分。但一般来说,数据缓冲寄存器、端口地址译码器和输入/输出操作控制逻辑是任何接口都不可少的,至于其他各部分是否需要、需要多少,则取决于接口功能的复杂程度和 I/O 操作的同步控制方式。

## 1.3 I/O 同步控制方式

### 1.3.1 I/O 同步控制概述

I/O 同步控制方式是微机基本系统与 I/O 外设之间数据传送的管理方法,是微机系统的一种调度策略。I/O 同步控制的目的是要实现 MPU 与 I/O 设备之间操作的同步,以实现两者之间正确有效的数据传送。I/O 操作和存储器操作有很大不同,这是由 I/O 设备与存储器的不同特性(主要是工作速度的不同)所决定的。由于半导体存储器与微处理器都采用 LSI 或 VLSI 工艺制成,它们的运行速度以至工作电平和数据格式等基本一致。尽管各种存储芯片的读写周期快慢不一,并且一般也比 MPU 的时钟周期长,但都是在同一数量级上。所以 MPU 与半导体存储器之间信息的交换除需要适当的电压或电流驱动的接口电路之外,可直接由 MPU 执行内存读或内存写指令操作来完成。在 MPU 向存储器发出读/写命令后,只要等待(有时还可重叠执行一些必要的微操作)若干固定的时钟周期,即可得到该次读/写操作的结果。而 I/O 外设则不一样,它们的数据传送速度比 MPU 相差好几个数量级,而且各种外设之间的数据传送速度也相差悬殊,加上 I/O 工作过程的定时完全独立于微处理器,所以它们之间的数据传送必须通过接口中的数据缓冲寄存器来进行,如图 1-6 所示。

输入一个数据的过程具体如下:

- (1) 外设把数据送入 I/O 接口的数据输入缓冲器;
- (2) 微处理器发出地址码通过地址总线寻址该输入缓存器;