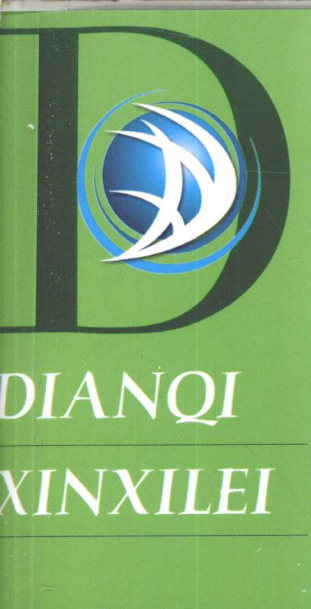




普通高等教育“十二五”电气信息类规划教材

微型计算机原理 及接口技术

■ 林志贵 主编

免费电子课件



机械工业出版社
CHINA MACHINE PRESS



普通高等教育“十二五”电气信息类规划教材

微型计算机原理及接口技术

主 编	林志贵	
副主编	严锡君	袁臣虎
参 编	王 敏	李现国
主 审	王宜怀	



机械工业出版社

本书以 Intel 80x86 CPU 为核心,系统介绍微型计算机基础知识、原理、指令系统、汇编语言程序设计、总线、存储器、输入/输出接口、中断系统、模拟量输入/输出、人-机接口及应用等。本书将系统的硬件和软件融为一体,突出微型计算机的3个关键技术:堆栈、中断、描述符。在掌握微型计算机基本体系的基础上,进一步介绍新型微型计算机的体系结构、现代存储器、USB、IEEE1394、以太网等总线技术。

本书内容新颖全面、重点突出;在充分考虑教学与自学的基础上,叙述由浅入深、通俗简洁,并配备了例题和习题。

本书可作为高等院校学生学习微型计算机原理和接口技术的教材,也可以作为有关科技人员的自学教材和参考书。

本书配有免费电子课件,欢迎选用本书作教材的老师登录 <http://www.cmpedu.com> 注册下载或发邮件到 yaxin_w74@126.com 索取。

图书在版编目(CIP)数据

微型计算机原理及接口技术/林志贵主编. —北京:机械工业出版社, 2010.8

普通高等教育“十二五”电气信息类规划教材
ISBN 978-7-111-30798-3

I. ①微… II. ①林… III. ①微型计算机—理论—高等学校—教材②
微型计算机—接口设备—高等学校—教材 IV. ①TP36

中国版本图书馆CIP数据核字(2010)第098023号

机械工业出版社(北京市百万庄大街22号 邮政编码100037)

策划编辑:王雅新 责任编辑:王雅新

版式设计:张世琴 责任校对:陈立辉

封面设计:赵颖喆 责任印制:杨 曦

保定市中国画美凯印刷有限公司印刷

2010年9月第1版第1次印刷

184mm×260mm·25.25印张·622千字

标准书号:ISBN 978-7-111-30798-3

定价:39.00元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

社服务中心:(010)88361066

门户网:<http://www.cmpbook.com>

销售一部:(010)68326294

教材网:<http://www.cmpedu.com>

销售二部:(010)88379649

读者服务部:(010)68993821

封面无防伪标均为盗版

前 言

本书是编者在多年从事微型计算机原理及接口课程教学实践和科研经验的基础上,参考了大量相关文献资料,组织编写而成。本书主要目的是使理工科非计算机专业的研究生或本科生了解微处理器发展的新技术和应用领域,掌握微型计算机的基本结构、工作原理、接口技术及汇编语言程序设计等内容,具有初步的微型计算机硬件和软件开发能力。同时,本书也可供从事微型计算机硬件或软件技术工作的工程技术人员参考。

本书在编写过程中,结合国家教委提出的加强本科生实践能力培养的精神,重点讲述了微型计算机的基本原理以及与外设的接口技术。本书共 14 章:第 1 章讲述微型计算机的发展过程、数制、编码以及二进制数运算知识;第 2 章介绍了 Intel 80x86 处理器的结构、工作方式及特点;第 3 章详细介绍了 8086/8088 指令的寻址方式及其指令系统;第 4 章简要介绍了汇编语言的特点及伪指令,重点介绍了汇编语言的程序设计及应用实例;第 5 章详细阐述了 ISA 总线、PCI 总线,同时,也介绍了 USB、IEEE1394 等总线技术;第 6 章介绍了半导体存储器的分类、管理及特点,详细阐述了 Flash 存储器以及存储器与 CPU 连接时要注意的问题;第 7 章介绍输入/输出技术及总线控制器 8237A,重点讲述了 I/O 地址的分配问题;第 8 章介绍了中断的概念、8086 中断特点以及中断控制器 8259A;第 9 章介绍了计数/定时的概念、可编程计数/定时控制器 8253A 的工作原理及其应用实例;第 10 章介绍了可编程并行接口芯片 8255A 的工作原理及应用实例;第 11 章介绍了串行通信基本概念、串行通信接口以及可编程串行通信接口芯片 INS 8250/NS 16x50 的工作原理与应用;第 12 章介绍了数/模转换技术(D/A 转换器)及应用,以及模/数转换技术(A/D 转换器)及应用;第 13 章简要介绍了微型计算机人-机接口技术;第 14 章列举了微型计算机的几个应用实例。

本书在内容安排方面,从理工科非计算机专业的特点出发,注重应用,叙述由浅入深,逐层递进。由于微型计算机原理及接口技术课程是一门实践性很强的课程,在采用本书作为教材时,还应注意加强实践环节,通过大量的上机实验,培养学生基于微型计算机的实验研究能力以及软、硬件方面的动手开发能力。

本书第 1、2、7、8、10 章由林志贵编写;第 3、14 章由李现国编写;第 4、13 章由严锡君编写;第 5、6 章及附录由王敏编写;第 9、11、12 章由袁臣虎编写。林志贵负责全书的大纲拟定、组织编写与统稿工作。

本书由苏州大学计算机学院王宜怀教授主审,他对书稿进行了认真的审阅和指导,提出了许多宝贵意见,在此对他表示衷心的感谢。

对本书编写过程中所参考的书籍和有关文献的作者表示衷心感谢。

由于编者水平有限,加之时间比较仓促,书中难免有不妥或错误之处,恳请读者提出宝贵意见和建议。

编 者

目 录

前言

第1章 微型计算机基础知识 1

1.1 概述 1

1.1.1 微型计算机的发展历史 1

1.1.2 微型计算机系统的组成 2

1.1.3 微型计算机的分类 4

1.1.4 微型计算机的主要性能指标 4

1.2 微机中的数制和数的表示 6

1.2.1 数制的基与权 6

1.2.2 数制的转换方法 7

1.2.3 二进制数的表示 8

1.2.4 微机中数的表示 11

1.3 微机的编码 12

1.3.1 ASCII 码 13

1.3.2 BCD 码 13

1.4 二进制数的运算及其电路 15

1.4.1 二进制数的运算 15

1.4.2 加法器 18

1.4.3 二进制数的加法/减法电路 19

习题 23

第2章 Intel 80x86 微处理器 24

2.1 8086/8088 CPU 的结构 24

2.1.1 执行部件 25

2.1.2 总线接口部件 26

2.1.3 8086 总线的工作周期 27

2.2 8086/8088 CPU 的引脚信号和工作模式 27

2.2.1 8086/8088 CPU 的引脚信号和功能 28

2.2.2 最小工作模式 31

2.2.3 最大工作模式 34

2.3 8086 存储器系统 36

2.3.1 存储器结构 36

2.3.2 堆栈的概念及 8086 堆栈 38

2.4 8086/8088 的主要操作功能 40

2.4.1 系统的复位和启动操作 40

2.4.2 总线操作 41

2.4.3 最小工作模式下的总线保持 44

2.4.4 最大工作模式下的读/写操作 45

2.5 80286 微处理器 45

2.6 80386/80486 微处理器 48

2.7 Pentium 微处理器 54

习题 58

第3章 8086/8088 指令系统 60

3.1 汇编语言指令格式与寻址方式 60

3.1.1 汇编语言指令语句格式 60

3.1.2 寻址方式 61

3.2 指令系统 67

3.2.1 数据传送指令 68

3.2.2 算术运算指令 76

3.2.3 逻辑运算和移位指令 88

3.2.4 串操作指令 96

3.2.5 控制转移指令 102

3.2.6 处理器控制指令 111

习题 112

第4章 汇编语言程序设计 115

4.1 汇编语言的基本语法 116

4.1.1 伪指令的语句格式 116

4.1.2 常数、变量和标号 116

4.1.3 运算符 119

4.2 伪指令 120

4.2.1 符号定义伪指令(赋值语句) 120

4.2.2 内存数据定义伪指令 120

4.2.3 段定义伪指令 120

4.2.4 过程(子程序)定义伪指令 123

4.2.5 模块定义与连接伪指令 123

4.2.6 列表伪指令 123

4.2.7 其他伪指令 124

4.3 系统功能调用 125

4.3.1 概述 125

4.3.2 DOS 系统功能调用	126	5.6.4 USB 数据传输	189
4.3.3 BIOS 中断调用	132	5.7 其他串行总线标准	189
4.4 汇编器的功能及汇编过程	138	5.7.1 IEEE1394	189
4.4.1 汇编器的功能	138	5.7.2 以太网	191
4.4.2 程序的编辑、汇编及连接 过程	138	习题	195
4.5 汇编语言程序设计及举例	140	第 6 章 存储器	197
4.5.1 程序设计步骤及格式	140	6.1 半导体存储器的性能指标	197
4.5.2 顺序结构程序设计	142	6.2 随机读写存储器 (RAM)	198
4.5.3 分支结构程序设计	143	6.2.1 静态 RAM (SRAM)	198
4.5.4 循环结构程序设计	146	6.2.2 动态 RAM (DRAM)	201
4.5.5 子程序设计	151	6.2.3 新型 DRAM	203
4.5.6 查表程序设计	158	6.3 只读存储器 (ROM)	204
4.5.7 综合应用举例	159	6.3.1 掩膜 ROM	204
习题	166	6.3.2 可编程 ROM	205
第 5 章 PC 总线	168	6.3.3 可擦写 ROM	205
5.1 概述	168	6.3.4 电可擦写 ROM	206
5.1.1 总线的概念及分类	168	6.3.5 新一代可擦写存储器 Flash	210
5.1.2 总线标准	169	6.4 存储器的体系结构及扩展	212
5.1.3 总线主要性能指标	169	6.4.1 存储器的体系结构	212
5.2 总线的数据传输过程和控制 方式	170	6.4.2 高速缓冲存储器 (Cache)	213
5.2.1 总线的数据传输过程	170	6.4.3 虚拟存储器	215
5.2.2 总线仲裁	170	6.4.4 存储器扩展方式	216
5.2.3 总线数据传输方法	172	6.5 CPU 与存储器的连接	217
5.3 PC 总线	173	6.5.1 连接时应注意的问题	217
5.4 ISA 总线	175	6.5.2 存储器的译码方式	218
5.4.1 ISA 总线的特点	175	6.5.3 CPU 与存储器的连接和地址 分析	219
5.4.2 ISA 总线系统结构	175	6.6 存储器管理	223
5.4.3 ISA 总线信号	176	6.6.1 内存分段的基本思想	223
5.5 PCI 总线	178	6.6.2 描述符与描述符表	223
5.5.1 PCI 总线的特点	179	6.6.3 段间保护	227
5.5.2 PCI 总线系统结构	180	6.6.4 段页式管理	228
5.5.3 PCI 总线信号	180	习题	229
5.5.4 PCI 总线周期和地址空间	182	第 7 章 输入/输出接口	231
5.5.5 PCI 总线配置空间	183	7.1 概述	231
5.6 通用串行总线 USB	185	7.1.1 I/O 接口的信号	231
5.6.1 USB 概述	185	7.1.2 I/O 接口	231
5.6.2 USB 总线的构成	186	7.1.3 I/O 编址	234
5.6.3 USB 系统的接口信号和电气 特性	188	7.1.4 I/O 接口的控制方式	234
		7.2 可编程 DMA 控制器 8237A	236

7.2.1 DMA 控制器的基本功能	236	9.4.5 方式4	279
7.2.2 8237A 的内部结构	237	9.4.6 方式5	280
7.2.3 8237A 的工作时序	239	9.5 8253A 的应用	280
7.2.4 8237A 的工作方式	240	习题	283
7.2.5 8237A 的寄存器组	241		
7.2.6 8237A 的编程及应用	244		
习题	247		
第8章 86 系列微型计算机的中断		第10章 可编程并行接口芯片	
系统	248	8255A	284
8.1 中断的基本概念	248	10.1 8255A 内部结构及引脚	284
8.1.1 中断的定义和作用	248	10.2 8255A 控制字	285
8.1.2 中断响应	249	10.2.1 工作方式选择控制字	286
8.1.3 中断的优先级	249	10.2.2 置位/复位控制字	286
8.1.4 中断嵌套	251	10.3 8255A 的工作方式	286
8.2 中断的处理	252	10.3.1 方式0	286
8.3 8086/8088 中断系统	253	10.3.2 方式1	287
8.3.1 中断矢量和中断矢量表	253	10.3.3 方式2	289
8.3.2 内部中断	254	10.4 8255A 的状态字	290
8.3.3 外部中断	256	10.5 8255A 的应用	291
8.3.4 中断矢量表的初始化	257	习题	295
8.4 可编程序中断控制器		第11章 可编程串行通信及接口	297
8259A	259	11.1 基本概念	297
8.4.1 8259A 的内部结构	259	11.2 串行通信接口	300
8.4.2 8259A 的引脚功能	261	11.2.1 RS-232-C 总线	300
8.4.3 8259A 的工作原理	262	11.2.2 RS-422A/423A 总线	301
8.4.4 8259A 的控制字及工作方式	266	11.2.3 RS-485 总线	302
8.4.5 8259A 的应用举例	271	11.2.4 其他串行接口	303
习题	271	11.3 可编程串行通信接口	
		INS 8250 和 NS 16x50	304
第9章 可编程计数/定时控制器		11.3.1 INS 8250 和 NS 16x50 的内部	
8253A	273	结构及引脚	305
9.1 基本概念	273	11.3.2 INS 8250 和 NS 16x50 的寄存	
9.2 8253A 的内部结构和引脚	273	器组	308
9.3 8253A 的编程	276	11.3.3 工作模式	315
9.3.1 8253A 的初始化编程	276	11.3.4 INS 8250 的初始化编程	317
9.3.2 8253A 的锁存命令	277	11.3.5 INS 8250 的应用	318
9.4 8253A 的工作方式	277	习题	320
9.4.1 方式0	277	第12章 模拟量输入/输出	321
9.4.2 方式1	278	12.1 模拟量的输入/输出通道	321
9.4.3 方式2	278	12.1.1 模拟量输入通道的组成	321
9.4.4 方式3	279	12.1.2 模拟量输出通道的组成	322
		12.2 D/A 转换器	322

12.2.1 D/A 转换器的工作原理	322	13.5 打印机及其接口技术	357
12.2.2 D/A 转换器的主要技术指标	324	13.5.1 打印机的工作原理	357
12.2.3 典型的 D/A 转换器芯片	325	13.5.2 打印机与微机的接口方式	358
12.2.4 D/A 转换器与微处理器连接 应用举例	329	13.6 多媒体计算机	359
12.3 A/D 转换器	333	习题	359
12.3.1 信号变换中的采样、量化和 编码	333	第 14 章 微型计算机的应用	360
12.3.2 A/D 转换器的工作原理	334	14.1 概述	360
12.3.3 A/D 转换器的主要技术指标	336	14.2 在控制系统中的应用	361
12.3.4 典型的 A/D 转换器芯片	337	14.2.1 微型计算机控制系统的组成	361
12.3.5 A/D 转换器与微处理器连接 应用举例	340	14.2.2 在多对象检测及控制系统中的 应用	361
12.3.6 V/F 转换器	346	14.2.3 在过程控制系统中的应用	362
习题	347	14.3 微机控制步进电动机的变速 系统	364
第 13 章 人-机接口	349	14.4 轧钢作业微机监测管理 系统	365
13.1 概述	349	14.5 在生物科学中的应用	367
13.1.1 常用的外围设备	349	14.6 在临床医疗仪器中的应用	367
13.1.2 外围设备的分类	349	习题	368
13.2 键盘及其接口技术	350	附录	370
13.2.1 键盘的工作原理	350	附录 A ASCII 码字符表	370
13.2.2 小型键盘接口	350	附录 B 8086/8088 指令系统	371
13.2.3 微机键盘及其接口	351	附录 C DOS 系统功能调用 (INT 21H)	384
13.3 鼠标及其接口技术	352	附录 D BIOS 调用	387
13.3.1 鼠标的工作原理	352	附录 E IBM PC/XT 的中断矢量表	391
13.3.2 鼠标与微机的接口方式	353	参考文献	394
13.4 显示器及其接口技术	354		
13.4.1 LED 显示器及其接口	354		
13.4.2 LCD 显示器	354		
13.4.3 显示卡	355		

第 1 章 微型计算机基础知识

1.1 概述

1.1.1 微型计算机的发展历史

自 1946 年诞生第一台电子计算机以来,在短短的几十年里,计算机已经历了电子管计算机、晶体管计算机、集成电路计算机和大规模及超大规模集成电路计算机发展过程。计算机按其性能、价格和体积的不同,一般分为五大类:巨型机、大型机、中型机、小型机和微型计算机。

微型计算机于 20 世纪 70 年代初研制成功。近年来,微处理器和微型计算机获得了极快的发展,几乎每两年微处理器的集成度翻一番,每 2~4 年更新换代一次。微处理器是微型计算机的核心芯片,简称为 μP 或 MP ,是将微机中的运算器和控制器集成在一片硅片上制成的集成电路。这样的芯片也被称为中央处理单元,简称为 CPU。

第一代为 4 位或低档 8 位微处理器,如 Intel 公司 4 位的 4004 及 8 位的 8008。它们均采用 PMOS 工艺,集成度约为 2000 个晶体管(或晶体片),只能进行串行的二进制运算,但用在各种类型的计算器中已经完全满足要求。这代微处理器的指令系统比较简单,运算能力差、速度慢,但价格低廉。软件主要使用机器语言及简单的汇编语言。

第二代为中高档 8 位微处理器,如 Intel 8085、Z80 和 MC6809。它们均为 8 位微处理器,具有 16 位地址总线,因此,最多可寻址 64K 存储单元。它们比第一代有了较多改进,如:采用 NMOS 工艺,集成度提高 1~4 倍,运算速度提高 10~15 倍,指令系统相对比较完善,已具有典型的计算机体系结构以及中断、存储器直接存取(DMA)功能。软件除汇编语言外,还可使用 BASIC、FORTRAN 以及 PL/M 等高级语言。但对于具有大量数据的大型复杂程序是不够的。另外,8 位微处理器每次只能处理 8 位数据,处理大量数据就要分成许多个 8 位字节进行操作,数值越大或越小,计算时间都很长。

第三代为 16 位微处理器,如 Intel 8086CPU、Z8000CPU 和 MC68000CPU。它们均为 16 位微处理器,具有 20 位地址总线。它们具有丰富的指令系统、多级中断系统、多处理器系统、段式存储器管理以及硬件乘除运算等。除此以外,还配备了功能较强的系统软件。

第四代为 32 位微处理器,如 Intel 80386/80486CPU 及 Pentium 系列 CPU。它们均为 32 位微处理器,具有 32 位地址总线。与协处理器协同工作,并采用高速缓冲技术,使其性能大大提高。

第五代为 64 位微处理器,如 Intel 80586CPU 及 Pentium Pro 等。80586CPU 的外围数据总线为 64 位,地址总线为 32 位,工作频率为 66MHz。Pentium Pro 的外围数据总线为 64 位,地址总线为 36 位,工作频率达到 200MHz。同时,采用了 RISC、超级流水线、超标量结构、MMX、动态分支预测等先进技术,使其运算速度及其吞吐量大大增强。

由于生产技术的限制,传统地通过提高工作频率来提升处理器性能的做法面临严重的阻碍,高频 CPU 的耗电量和发热量越来越大,这给整机散热带来了十分严峻的考验。双核技术即可解决这一问题。2006 年 Intel 公司推出 32 位全新 Core 架构的 Core 型微处理器。该处理器具有高级智能高速缓存、高级数字媒体增强、动态功率调节、深度睡眠、智能内存访问、先进的分支预测及宽位动态执行等。随着双核技术的进一步成熟,以及配套软件的开发及优化,双核/多核处理器将会成为市场的主流。

1.1.2 微型计算机系统的组成

微型计算机系统由硬件和软件两部分组成,如图 1-1 所示。

1. 微型计算机的硬件

微型计算机结构是指微机的硬件系统按照总体布局的设计要求将各部件构成某个系统的连接方式。至今为止,微型计算机硬件系统的组成无一不是采用冯·诺依曼结构,冯·诺依曼(出生在匈牙利的美国数学家)结构的基本内容是:

1) 用二进制形式表示数据和指令。

2) 将程序和数据预先存放在主存储器中,使微机在工作时能够自动高速地从存储器中取出指令和数据,并加以执行,这是“存储程序控制”的基本特点。

3) 确立了微机系统的五大基本部件:运算器、控制器、存储器、输入设备、输出设备。

微型计算机体系结构的主要特点就是采用总线结构,通过总线将微处理器(CPU)、存储器(RAM 和 ROM)、I/O 接口电路连接起来,而输入/输出设备则通过 I/O 接口实现与微处理器的信息交换,如图 1-2 所示。

总线是指微机中各功能部件间传送信息的公共通道,是微型计算机的重要组成部分。它们可以是带状的扁平电缆线,也可以是印制电路板上的一层极薄的金属连线。系统中各部件都是“挂”在总线上的,所有的信息都通过总线传送。根据所传送信息的内容与作用不同,总线可分为数据总线、控制总线和地址总线 3 类。

数据总线(DB):双向传输数据信息,指令码或数据信息通过数据总线送往 CPU 或由 CPU 送出。其宽度(根数)与 CPU 提供的数据线的引脚数有关,宽度越宽,传输数据的能力越强。

控制总线(CB):各种控制或状态信息通过控制总线由 CPU 送往有关部件,或者从有关部件送往 CPU。对于每一根来说是单向传送的,图 1-2 中 CB 作为一个整体,用双向表示。

地址总线(AB):CPU 执行指令时,用于单向传送地址信息,以便选中要访问的存储单

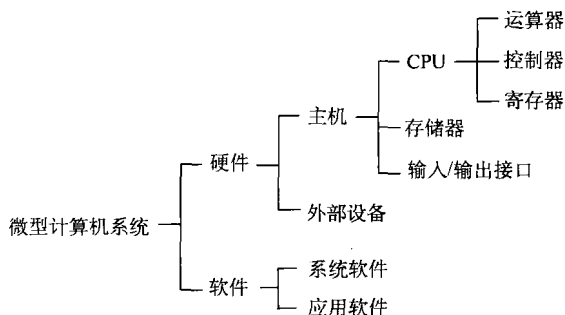


图 1-1 微型计算机系统的组成

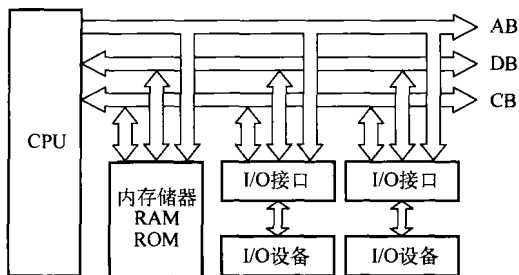


图 1-2 微型计算机的外部结构图

元或 I/O 端口。地址信息包括两种：指令代码在存储器中的地址信息和操作数在存储器中的地址信息。AB 的宽度决定了微机系统的最大寻址能力（寻址空间）。最大寻址空间 $= 2^N$ ，其中 N 为 AB 的宽度。例如，8086/8088CPU 的 $N=20$ ，则最大寻址空间 $2^{20}=1\text{MB}$ 。

总线结构中，系统中各部件均挂在总线上，可使微机系统的结构简单，易于维护，并具有更好的可扩展性。一个部件（插件）只要符合总线标准就可以直接插入系统，为用户扩充或升级系统功能提供了很大的灵活性。

(1) 微处理器（CPU） CPU 是微型计算机的核心芯片，它包括运算器、控制器和寄存器三个主要部分。运算器也称为算术逻辑单元（ALU）。顾名思义，运算器的功能是完成数据的算术和逻辑运算的。控制器是微机的指挥控制中心，由指令寄存器、指令译码器和控制电路组成。它负责把指令逐条从存储器中取出，经译码分析后，根据指令的要求，对微型计算机各部件发出相应的控制信息（取数、执行、存数等），使它们协调工作，以保证正确完成程序所要求的功能，从而达到对整个微机系统的控制。CPU 内部的寄存器用来暂存数据。

(2) 存储器 存储器（Memory）又称为主存（Main Storage）或内存，是微型计算机的存储和记忆装置，用以存放数据和程序。微型计算机的内存通常采用半导体存储器。

内存中存放的是数据和程序，从形式上看，均为二进制数。一般将 8 位二进制数记作一个字节（Byte，B）。为了便于对存储器进行访问，存储器通常被划分为许多单元，每一个内存单元中存放一个字节的二进制信息。

微型计算机为了对存储器进行管理，将每个内存单元都分别赋予一个编号，称为地址（内存单元地址）。这样，CPU 便能识别不同的内存单元，正确地对其进行操作，如图 1-3 所示。

显然，内存单元的地址和内存单元的内容是两个完全不同的概念。例如在图 1-3 中，第 6 号内存单元的地址是 00006H，其内容是 11001111B，即 CFH。

地址	内容
00000H	
00001H	
00002H	
	⋮
00006H	1100 1111
	⋮
FFFFFFH	

图 1-3 内存单元的地址和内容

CPU 对内存的操作有两种：读或写。读操作是 CPU 将内存单元的内容读入 CPU 内部；写操作是 CPU 将其内部信息传送到内存单元保存起来。显然，写操作的结果改变了被写内存单元的内容，是破坏性的，而读操作是非破坏性的，即，该内存单元的内容在信息被读“走”之后仍保持原信息。

按工作方式，内存可分为两大类：随机存储器（RAM）和只读存储器（ROM）。RAM 可以被 CPU 随机地读写，故称为读写存储器。这种存储器用于存放用户装入的程序、数据及部分系统信息。当机器断电后，所存信息消失。ROM 中的信息只能被 CPU 读取，而不能由 CPU 任意写入，故称为只读存储器，机器断电，信息仍保留。这种存储器用于存放固定的程序，如：基本的 I/O 程序、BASIC 解释程序以及用户编写的专用程序等。ROM 中的内容只能够用专用设备写入。

微机在执行程序时，CPU 会自动而连续地从内存中取出要执行的指令，并执行指令规定的操作。这就是说，微机每完成一条指令，至少有一次为取指令而访问内存的操作。内存是微机主机的一部分，一般把具有一定容量且速度较高的存储器作为内存，CPU 可直接用指令对内存进行读/写。在微型计算机中，通常用半导体存储器作为内存。

(3) 输入/输出 (I/O) 接口 外设是微型计算机系统的重要组成部分。程序、数据及现场信息是通过外设输入给微型计算机。CPU 计算的结果通过外设输出到外部。外设的种类很多,有机械式、电动式、电子式等。与 CPU 相比,外设工作速度较低。外设处理的信息有数字量、模拟量、开关量等,而微型计算机只能处理数字量。另外,外设与微型计算机工作的逻辑时序也可能不一致。鉴于上述原因,微型计算机不能直接与外设连接及交换信息,需设计一个“接口电路”作为微型计算机与外设之间的桥梁,这种接口电路又叫做输入/输出 (I/O) 接口或“I/O 适配器”。

微型计算机与外设进行信息交换是通过接口电路中的端口实现的。每个接口可包含若干个端口,每个端口对应一个端口地址,可由指令按地址访问端口。接口的主要功能有:隔离主机与外设之间的数据;向外设传输控制信号和接收外设的状态信号;数据类型的转换。

2. 微型计算机软件

微型计算机软件是为了运行、管理和维护微型计算机而编制的各种程序的总和。软件和硬件是微型计算机系统不可分离的两个重要组成部分。微机是通过逐条地从存储器中取出程序中的指令并执行指令规定的操作而实现某种特定的功能。

微型计算机软件包括系统软件和应用软件。系统软件是指不需要用户干预的,为其他程序的开发、调试以及运行等建立一个良好环境的程序。系统软件主要包括操作系统 (OS) 和系统应用程序。操作系统是控制微型计算机的资源 (如 CPU、存储器及 I/O 设备等),使应用程序得以自动执行的程序。它是一套复杂的系统程序,用于提供人-机接口和管理、调度计算机的所有硬件与软件资源。其中最为重要的核心部分是常驻监控程序,微机启动后,常驻监控程序始终存放在内存中,它接收用户命令,并执行相应的操作。操作系统还包括用于执行 I/O 操作的 I/O 驱动程序,每当用户程序或其他系统程序需要使用 I/O 设备时,通常并不是该程序执行 I/O 操作,而是由操作系统利用 I/O 驱动程序来执行任务。

应用软件就是用户为解决各种实际问题而编写的各种程序。可用来编写用户软件的语言有机器语言、汇编语言和高级语言等。

1.1.3 微型计算机的分类

微型计算机的分类方法很多。按微处理器的位数可分为 1 位、4 位、8 位、32 位和 64 位机等;按功能和结构可分为单片机和多片机;按组装方式可分为单板机和多板机等。

利用大规模集成电路工艺将微机的 3 大组成部分——CPU、内存和 I/O 接口集成在一片硅片上,这就是单片机,使用专用开发装置可以对它进行在线开发。单片机在工业过程控制、智能仪器仪表和家用电器中得到了广泛的应用。

若将微型计算机的 CPU、内存、I/O 接口电路安装在一块印制电路板上就组成了单板机。单板机结构简单,价格低廉,性能较好,可用于过程控制、各种仪器仪表、机器的单机控制、数据处理等。

微型计算机是多板机,这是因为它是由主板及插在主板上的多个电路板 (如:显示卡、声卡、多功能卡、网卡等) 组成的。

1.1.4 微型计算机的主要性能指标

微机系统和一般计算机系统一样,衡量其性能好坏的技术指标主要有以下 6 个方面。

1. 字长

微机一条指令所能处理的一个基本信息单位,称为一个字。例如,一个数据(25、35.67、-0.0038……)、一个字符(A、a、#……)等均称为一个字。

字长是微机内部一次可以处理的二进制数码的位数。微机的字长决定于它的通用寄存器、内部存储器、ALU的位数和数据总线的宽度。例如,8086/8088 CPU内部寄存器为16位,所以字长为16位,称为16位机;80386、80486、80586(pentium)字长均为32位,故均称为32位机;近年来(2001年)新推出的“安腾”(Itanium)为64位机。

字长是微机的主要性能指标之一。字长越长的微机,其运算速度越快,数的表示范围越宽,数据的运算精度越高,机器的整体功能越强。然而,字长越长,微机的硬件代价也增大。为了兼顾精度/速度与硬件成本两方面,有些微机允许采用变字长运算。

一般情况下,CPU的内、外数据总线宽度是一致的。但有的CPU为了改进运算性能,加宽了CPU的内部总线宽度,致使内部字长和对外数据总线宽度不一致。如Intel 8088/80188的内部数据总线宽度为16位,外部为8位。对这类芯片,称之为“准××位”CPU,因此Intel 8088/80188被称为“准16位”CPU。

2. 存储器容量

存储器容量是衡量微机存储二进制信息量多少的一个重要指标。存储二进制信息的基本单位是位(bit)。一般把8个二进制位组成的基本单元叫做字节(B)。微机中通常以字节为单位表示存储容量,并且将1024B简称为1KB,1024KB简称为1MB(兆字节),1024MB简称为1GB(吉字节),1024GB简称为1TB(太字节)。

存储器容量包括内存容量和外存容量。内存容量又分最大容量和实际装机容量。最大容量由CPU的地址总线位数决定,如8位CPU的地址总线为16位,其最大内存容量为64KB;Pentium处理器的地址总线为32位,其最大内存容量为4GB。而装机容量则由所用软件环境决定,如现行PC系列机,若采用Windows,则内存必须在4MB以上;若采用Windows 95,则内存必须在8MB以上;若采用Windows XP,则内存必须在256MB以上等。

外存容量是指硬盘和光盘等的容量,通常主要指硬盘容量,其大小应根据实际应用的需要来配置。

3. 运算速度

微机的运算速度一般用每秒钟所能执行的指令条数来表示。由于不同类型的指令所需时间长度不同,因而运算速度的计算方法也不同。常用计算方法有:

1) 根据不同类型的指令出现的频度,乘上不同的系数,求得统计平均值,得到平均运算速度。这时常用品MIPS(Millions of Instruction Per Second,即百万条指令/秒)作单位。

2) 以执行时间最短的指令(如加法指令)为标准来估算速度。

3) 直接给出CPU的主频和每条指令的执行所需的时钟周期。主频一般以MHz为单位。

4. 系统总线

系统总线是连接微机系统各功能部件的公共数据通道,其性能直接关系到微机系统的整体性能。系统总线的性能主要表现为它所支持的数据传送位数和总线工作时钟的频率。数据传送位数越多,总线工作时钟频率越高,则系统总线的信息吞吐率就越高,微机系统的性能就越强。

5. 外设扩展能力

这主要指微机系统配接各种外部设备的可能性、灵活性和适应性。一台微机允许配接多少外部设备，对于系统接口和软件研制都有重大影响。在微机系统中，打印机型号、显示屏分辨率、外存储器容量等，都是外设配置中需要考虑的问题。

6. 软件配置情况

软件是微机系统必不可少的重要组成部分，其配置是否齐全，功能的强弱，是否支持多任务、多用户操作等都是微机硬件系统性能可否得到充分发挥的重要因素。

1.2 微机中的数制和数的表示

数制是人们利用符号来记数的科学方法。数制可以有很多种，但在微机的设计与使用上常采用的有十进制、二进制、八进制和十六进制。

1.2.1 数制的基与权

数值所使用的数码的个数称为基；数值每一位所具有的值称为权。

十进制的基为“10”，即它使用的数码为0，1，2，3，4，5，6，7，8，9共有10个。十进制各位的权是以10为底的幂，用“D”表示，如十进制数523791 D：

5	2	3	7	9	1
10^5	10^4	10^3	10^2	10^1	10^0
十万	万	千	百	十	个

其各位的权为个、十、百、千、万、十万，即以10为底的0次幂、1次幂、2次幂等，故有时为了简便而顺次称其各位为0权位、1权位、4权位等。

二进制的基为“2”，即其使用的数码为0、1，共两个。

二进制各位的权是以2为底的幂，用“B”表示，如二进制数110111 B：

二进制	1	1	0	1	1	1
	2^5	2^4	2^3	2^2	2^1	2^0
十进制	32	16	8	4	2	1

其各位的权为1、2、3，即以2为底的0次幂、1次幂、2次幂等，故有时也依次称其各位为0权位、1权位、2权位等。

八进制的基为“8”，即其数码共有8个：0，1，2，3，4，5，6，7。八进制的权是以8为底的幂，有时也顺次称其各位为0权位、1权位、2权位等，用“O”表示。

十六进制的基为“16”，即其数码共有16个：0，1，2，3，4，5，6，7，9，A，B，C，D，E，F。十六进制的权是以16为底的幂，有时也称其各位的权为0权位、1权位、2权位等，用“H”表示。

1.2.2 数制的转换方法

我们习惯用十进制记数，在研究问题或讨论解题的过程时，总是用十进制来考虑和书写。由于微机只能处理二进制数，因此，需要把问题中的所有十进制数转换成二进制代码。这就需要用到“十进制数转换成二进制数”的方法。在微机运算结束时，又需要用到“二进制数转换成十进制数”的方法，才能把运算结果（二进制数）用十进制形式显示出来。

1. 十进制数转换成二进制数的方法

一般可用下列方法求一个十进制数的二进制代码：

用2除该十进制数可得商数及余数，则此余数为二进制代码的最小有效位（LSB）的值。再用2除该商数，又得商数及余数，则此余数为LSB左邻的二进制数代码。用同样的方法继续用2除下去，就可得到该十进制数的二进制代码。

例 1-1 求 13 的二进制代码。其过程如下：

$$\begin{array}{r}
 0 \\
 2 \overline{) 13} \\
 \underline{2} \\
 2 \overline{) 3} \\
 \underline{2} \\
 2 \overline{) 6} \\
 \underline{2} \\
 2 \overline{) 13}
 \end{array}
 \begin{array}{r}
 1 \\
 1 \\
 0 \\
 1
 \end{array}
 \left. \begin{array}{c} \\ \\ \\ \end{array} \right\} \begin{array}{l} \\ \\ \\ \end{array} \text{ (由上往下读, 可从左至右写出二进制代码)}$$

结果为：1101 B。

上面是十进制整数转换成二进制数的“除2取余法”（除以基数倒序取余）。

如果十进制小数要转换成二进制小数，则要采取“乘2取整法”（乘以基数正序取整）。

一个十进制的小数乘以2之后可能有进位使整数位为1（当该小数大于0.5时），也可能没有进位，其整数位仍为0（当该小数小于0.5时）。这些整数位的结果即为二进制的小数位结果。举例如下：

例 1-2 求十进制数 0.625 的二进制数。

用乘法的竖式计算，步骤如下：

$$\begin{array}{r}
 0.625 \\
 \times) \quad 2 \\
 \hline
 1.25
 \end{array}
 \quad \text{整数部为 1, 即二进制小数后第一位为 1.}$$

$$\begin{array}{r}
 0.25 \\
 \times) \quad 2 \\
 \hline
 0.50
 \end{array}
 \quad \text{整数部为 0, 即二进制小数后第二位为 0.}$$

$$\begin{array}{r}
 0.50 \\
 \times) \quad 2 \\
 \hline
 1.00
 \end{array}
 \quad \text{整数部为 1, 即二进制小数后第三位为 1.}$$

至此就不用再算下去了。如果小数位不是 0.00，则还要继续乘下去，直至变成 0.00 为止。因此，一个十进制小数在转换为二进制小数时有可能无法准确地转换。如十进制数 0.1 转换为二进制数时为 0.0001100110...B。因此，只能近似地以 0.00011001 B 来表示。

当然，十进制数转换成二进制数还有一个更直接的方法，即用最接近的权值逐次逼近。

如把 13.625 转换成二进制，与这个数最接近的权为 $8 (2^3)$ ，13.625 减去 8 余下 5.625；

与 5.625 最接近的权是 4 (2^2), 余下 1.625; 下一步取权 1 (2^0), 余 0.625; 接下一步取权 0.5 (2^{-1}), 余 0.125; 再取权 0.125 (2^{-3}), 余零, 结束。

权:	8	4	0	1	0.5	0	0.125
	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}
结果:	1	1	0	1	1	0	1
	1101.101B						

2. 二进制数转换成十进制数的方法

由二进制数各位的权乘以各位的数 (0 或 1) 再加起来就得到十进制数。

例 1-3 求二进制数 101011 B 的十进制数。

	1	0	1	0	1	1
权:	2^5	2^4	2^3	2^2	2^1	2^0
乘积:	32	0	8	0	2	1
结果:	43D					

二进制小数转换为十进制时也可用同样的方法, 不过二进制数小数各位的权是 2^{-1} 、 2^{-2} ...

例 1-4 求二进制数 0.101 B 的十进制数。

	0	1	0	1
权:	2^0	2^{-1}	2^{-2}	2^{-3}
乘积:	0	0.5	0	0.125
结果:				

由此可得出两点注意事项:

1) 一个二进制数可以准确地转换为十进制数, 而一个带小数的十进制数不一定能够准确地用二进制数来表示。

2) 带小数的十进制数在转换为二进制数时, 以小数点为界, 整数和小数要分别转换。

此外, 还有其他各种数制之间的转换, 其方法和上述方法差不多, 都可以从数制的定义中找到转换方法。

1.2.3 二进制数的表示

1. 无符号数的表示法

二进制计数法的特点是: 以 2 为底, 逢 2 进位; 需要两个数字符号 0 和 1。

一个二进制数 N 可以表示为如下形式:

$$N_B = \sum_{i=-m}^{n-1} B_i \times 2^i$$

其中, n 为二进制数 N 整数部分的位数; m 为二进制数 N 小数部分的位数; B_i 为二进制数数字符号 0 或 1。

例如: $1101.1 B = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1}$ 。

上式中的后缀 B 表示二进制数。

2. 带符号数的表示法

日常生活中遇到的数,除上述的无符号数外,还有大量的带符号数。数的符号在微机中也用二进制数表示,通常用二进制数的最高位表示数的符号。把一个数及其符号在机器中的表示加以数值化,这样的数称为机器数,而机器数所代表的数称为该机器数的真值。机器数可以用不同方法表示,常用的有原码、反码和补码表示法。

(1) 原码 数 x 的原码记作 $[x]_{\text{原}}$,如机器字长为 n ,则原码的定义如下:

$$[x]_{\text{原}} = \begin{cases} x & 0 \leq x \leq 2^{n-1} - 1 \\ 2^{n-1} + |x| & -(2^{n-1} - 1) \leq x \leq 0 \end{cases}$$

例如,当 $n=8$ 时,

$$[+1]_{\text{原}} = 00000001 \text{ B}, [+127]_{\text{原}} = 01111111 \text{ B}$$

$$[-1]_{\text{原}} = 10000001 \text{ B}, [-127]_{\text{原}} = 11111111 \text{ B}$$

当 $n=16$ 时,

$$[+1]_{\text{原}} = 00000000 \ 00000001 \text{ B}, [+127]_{\text{原}} = 00000000 \ 01111111 \text{ B}$$

$$[-1]_{\text{原}} = 10000000 \ 00000001 \text{ B}, [-127]_{\text{原}} = 10000000 \ 01111111 \text{ B}$$

由此看出,原码表示法中,最高位为符号位,正数为 0,负数为 1。其余 $n-1$ 位表示数的绝对值。原码表示数的范围是 $-(2^{n-1} - 1) \sim (2^{n-1} - 1)$ 。8 位二进制原码表示数的范围是 $-127 \sim +127$,16 位二进制原码表示数的范围是 $-32767 \sim +32767$ 。原码表示法简单直观,但不便于进制加减运算。

(2) 反码 数 x 的反码记作 $[x]_{\text{反}}$,如机器字长为 n ,反码定义如下:

$$[x]_{\text{反}} = \begin{cases} x & 0 \leq x \leq 2^{n-1} - 1 \\ (2^n - 1) - |x| & -(2^{n-1} - 1) \leq x \leq 0 \end{cases}$$

例如,当 $n=8$ 时,

$$[+1]_{\text{反}} = 00000001 \text{ B}, [+127]_{\text{反}} = 01111111 \text{ B}$$

$$[-1]_{\text{反}} = 11111110 \text{ B}, [-127]_{\text{反}} = 10000000 \text{ B}$$

从反码表示法中可见,最高位仍为符号位,正数为 0,负数为 1。反码表示数的范围是 $-(2^{n-1} - 1) \sim +(2^{n-1} - 1)$ 。8 位二进制数反码表示数的范围是 $-127 \sim +127$,16 位二进制数反码表示数的范围是 $-32767 \sim +32767$ 。

从上例中还可以看出,正数的反码与原码相同,负数的反码只需将其对应的正数的反码(包括符号位)按位求反即可得到。

(3) 补码 数 x 的补码记作 $[x]_{\text{补}}$,当机器字长为 n 时,补码定义如下:

$$[x]_{\text{补}} = \begin{cases} x & 0 \leq x \leq 2^{n-1} - 1 \\ 2^n - |x| & -2^{n-1} \leq x \leq 0 \end{cases}$$

例如,当 $n=8$ 时,

$$[+1]_{\text{补}} = 00000001 \text{ B}, [+127]_{\text{补}} = 01111111 \text{ B}$$

$$[-1]_{\text{补}} = 2^8 - |-1| = 11111111 \text{ B}, [-127]_{\text{补}} = 2^8 - |-127| = 10000001 \text{ B}$$

补码表示法中,最高位仍为符号位,正数为 0,负数为 1。

补码表示数的范围为 $-2^{n-1} \sim (2^{n-1} - 1)$ 。8 位二进制补码表示数的范围是 $-128 \sim +127$,16