



高等学校“十二五”规划教材 ■ ■ ■

微机原理与接口技术

(第二版)

毛红旗 刘 敏 杨洪亮 主编

W

WEIJI YUANLI YU
JIEKOU JISHU

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE



高等学校“十二五”规划教材

微机原理与接口技术

(第二版)

毛红旗 刘敏 杨洪亮 主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书在总结微机基本原理和技术特点的基础上，按照“原理与应用相结合、硬件与软件相结合”的原则，介绍了微机接口技术的基本要点。本书以 Intel 系列微处理器为背景，以 16 位微处理器 8086 为核心，分别阐述了 8086 微处理器、指令系统、汇编语言程序设计、总线技术、存储器系统、中断系统、DMA 控制接口、微机系统常用的通用可编程接口的应用实例分析等内容。

本书内容精练、实用易懂，结构层次合理，讲解深入浅出，适合作为高等学校计算机及其他相关专业学生的教材，也可以作为从事微机系统设计和应用的技术人员的参考书。

图书在版编目（CIP）数据

微机原理与接口技术/毛红旗，刘敏，杨洪亮主编。
—2 版.— 北京：中国铁道出版社，2012.7
高等学校“十二五”规划教材
ISBN 978-7-113-14885-0

I . ①微… II . ①毛…②刘…③杨… III . ①微型计算机—理论—高等学校—教材②微型计算机—接口技术—高等学校—教材 IV . ①TP36

中国版本图书馆 CIP 数据核字（2012）第 126520 号

书 名：微机原理与接口技术（第二版）

作 者：毛红旗 刘 敏 杨洪亮 主编

策 划：秦绪好 王春霞 读者热线：400-668-0820

责任编辑：秦绪好 彭立辉

封面设计：刘 颖

责任印制：李 佳

出版发行：中国铁道出版社（100054，北京市西城区右安门西街 8 号）

网 址：<http://www.51eds.com>

印 刷：航远印刷有限公司

版 次：2007 年 8 月第 1 版 2012 年 7 月第 2 版 2012 年 7 月第 3 次印刷

开 本：787mm×1092mm 1/16 印张：15.5 字数：370 千

印 数：5 001~8 000 册

书 号：ISBN 978-7-113-14885-0

定 价：30.00 元

版权所有 侵权必究

凡购买铁道版图书，如有印制质量问题，请与本社教材图书营销部联系调换。电话：（010）63550836

打击盗版举报电话：（010）63549504

第二版前言

本书是在毛红旗、王春红、杨洪亮主编的《微机原理与接口技术》一书（2007年8月由中国铁道出版社出版）的基础上修订而成。随着教学改革的发展，要求开设课程的内容少而精，因此本次修订的指导思想是：在保持全书知识体系结构不变的前提下，精选课程内容，突出应用性，降低了部分内容的学习难度。修订后的教学参考学时约为60~70学时。具体对以下几个方面进行了修改：

- (1) 基础知识部分删除了与“计算机文化基础”课程中重复的内容。
- (2) 8086微处理器部分删掉了8086支持芯片的详细介绍，只进行了简介。最大模式的介绍也进行了删减。总线操作部分，只对“最小模式下的总线读操作”进行详细介绍，其他总线操作只做简要介绍。
- (3) 8086指令系统和汇编语言程序设计部分，删除了侧重算法的较复杂的例子，替换成简单且有助于学生理解计算机程序的汇编、连接过程以及程序的存储及执行过程的例子。本书学习汇编语言程序设计的重点在于帮助学生理解计算机的工作原理而不在于练习程序设计。
- (4) 对总线技术部分进行了删减。
- (5) 存储器部分删除了ROM和RAM工作原理的介绍，增加了具体存储器芯片的介绍。删除了高速缓冲存储器和虚拟存储器工作原理的介绍，增加了对存储器系统的缓存-主存和主存-辅存层次的简要介绍。
- (6) 删除了与“电子技术”课程重复的模拟接口的有关内容。输入/输出接口技术部分，主要把例题替换成简单但是有代表性且易于实验的例题。
- (7) 原附录A和B精简后合为附录A，附录B为新增Proteus仿真软件的使用简介。

本书共分12章，每章开头介绍本章主要内容和重点，结尾有小结，并配有适当数量的习题以加深对本章内容的理解。

正文中带“*”的内容为选学内容。

本书由毛红旗、刘敏、杨洪亮主编，其中第1章由王春红编写，第3~5章由杨洪亮编写，第2、6、9章由刘敏编写，第7、8、10~12章由毛红旗编写，各章和附录由刘敏负责修订，毛红旗负责全书的统稿工作。

本书在修订过程中，参考了大量书籍，听取了许多授课老师与广大读者的意见，在此一并表示感谢。

由于微机技术发展迅速，再加上编者水平有限，难免会有不足之处，敬请各位同仁和广大读者批评指正。

编 者

2012年6月

第一版前言

本书是微机原理与接口技术是计算机专业的一门专业基础课，是一门涉及知识面广、技术性强的学科，也是计算机专业技术应用型人才必须掌握的一门专业技术。只有理解和掌握接口技术的内容，才能真正地了解计算机并且应用计算机。本教材从接口技术的基本原理出发，详细讲述了系统接口和应用接口的基本概念与工作原理、接口设计的关键技术和方法，并给出了相应的应用实例。这些应用实例大多是多年从事教学和科研工作教师的经验总结。为学习方便，每章后面都配有复习思考题，并把经常需要查阅的部分学习资料作为附录列于书后。

《微机原理与接口技术》教材是在毛红旗教授主编的《微型机接口技术》一书的基础上，经过重新整理、修订、扩充而成的。本教材知识结构层次合理、内容实用易懂，主要有以下特点：

1. 注重基础、循序渐进

编者结合长期的教学实践，力求在微机的软、硬件技术结合上做到循序渐进、深入浅出地阐述其工作原理及实际应用。

2. 掌握原理、侧重应用

教材根据高职高专培养目标要求，侧重于对学生在微机接口的设计、开发和应用能力等方面加强培养。在介绍了每一种接口的基本原理和工作方式的基础上，以大量的应用实例分析说明应用技术的要点，使学生在牢固掌握微机原理知识的基础上，具有一定的设计能力和系统应用能力。

3. 重点突出、难点分散

教材遵循面向应用的教学目标、重点突出、内容详尽，力求在微机软件、硬件技术结合上由浅入深、从易到难、循序渐进，对内容的选取、概念的引入、文字的叙述、例题和思考题的设计等进行了精选。

4. 内容全面，风格良好

全书共分为 13 章。每章开头部分都有本章主要内容和要点，结尾部分都有本章小结，并配有一定的习题以加深对内容的理解。教学参考学时为 80~90 学时。在授课过程中，教师可根据实际课时适当安排教学内容，其中带有*号的章节可根据授课实际情况作为选讲内容。

本书由毛红旗、王春红、杨洪亮编著，其中第 1 章由金月光编写，第 3、4、5 章由杨洪亮编写，第 6、9 章由王春红编写，第 2、7、8、10、11、12、13 章由毛红旗编写，最后由毛红旗统稿。在编写过程中，参考了大量书籍，听取了许多专家和学者的宝贵意见，在此一并致以衷心的感谢。

由于编者的水平和经验有限，书中不妥之处在所难免，敬请读者和专家提出宝贵意见。

编者

2007 年 6 月

目 录

第 1 章 微机系统概述.....	1
1.1 微机发展概况	1
1.2 计算机中信息的表示方法	3
1.2.1 数制及其转换	3
1.2.2 计算机中二进制信息编码.....	5
1.2.3 计算机中数的表示.....	6
1.3 微机的基本结构	8
1.3.1 微型计算机系统	8
1.3.2 微型计算机的硬件系统....	10
1.3.3 微型计算机的软件系统....	11
*1.4 微机的主要性能指标和应用	12
1.4.1 微机的主要性能指标.....	12
1.4.2 微机的应用	13
小结	15
习题	15
第 2 章 80x86 微处理器	16
2.1 8086 CPU 的内部结构	16
2.1.1 8086 CPU 的功能结构.....	16
2.1.2 8086 CPU 的寄存器配置	18
2.2 8086 微处理器的引脚信号和 工作模式	20
2.2.1 8086 的引脚信号及功能	20
2.2.2 8086 的支持芯片	22
2.2.3 8086 的工作模式及 引脚特性	22
2.3 8086 微处理器的存储器组织	25
2.4 8086 微处理器的总线操作和 时序	28
2.4.1 8086 的复位和启动操作	28
2.4.2 8086 最小模式下的 总线操作	29
*2.4.3 8086 最大模式下的 总线操作	31
2.4.4 8086 的等待状态时序	32
2.4.5 8086 的中断响应周期	33
2.4.6 8086 的总线空闲周期 T	33
*2.5 Pentium 微处理器.....	34
2.5.1 Pentium 微处理器的 功能结构	34
2.5.2 Pentium 的寄存器组织	37
2.5.3 Pentium 的工作模式.....	40
2.5.4 Pentium 存储器系统.....	42
小结	42
习题	43
第 3 章 指令系统	44
3.1 寻址方式	44
3.1.1 立即数寻址	45
3.1.2 寄存器寻址方式	45
3.1.3 直接寻址方式	45
3.1.4 寄存器间接寻址方式	46
3.1.5 寄存器相对寻址方式	46
3.1.6 基址变址寻址方式	47
3.1.7 相对基址变址寻址方式	47
3.2 8086 指令系统	48
3.2.1 数据传送指令	48
3.2.2 算术运算指令	52
3.2.3 逻辑运算和移位指令	57
3.2.4 串操作类指令	60
3.2.5 控制转移指令	62
3.2.6 处理机控制指令	65
*3.3 80x86 与 Pentium 扩充和 增加的指令	66
3.3.1 80286 扩充和增加的指令	67
3.3.2 80386 扩充和增加的指令	67
3.3.3 80486 新增加的指令	68

3.3.4 Pentium 新增加的指令	68	5.3.2 AGP 总线	98
小结.....	69	5.4 设备总线	99
习题.....	69	*5.4.1 IEEE-488 总线	99
第 4 章 汇编语言程序设计	71	5.4.2 RS-232C 总线.....	100
4.1 汇编语言程序设计概述	71	5.4.3 USB 总线	101
4.1.1 汇编语言源程序的结构 ...	72	小结.....	102
4.1.2 汇编语言语句类型及格式....	73	习题.....	103
4.1.3 汇编语言的数据与表达式....	74		
4.2 伪指令	76	第 6 章 存储器系统	104
4.2.1 符号定义伪指令	76	6.1 存储器概述.....	104
4.2.2 数据定义伪指令	77	6.1.1 存储器体系结构	104
4.2.3 段定义伪指令	78	6.1.2 半导体存储器的分类	105
4.2.4 段寻址伪指令	78	6.1.3 半导体存储器的主要	
4.2.5 过程定义伪指令	79	性能指标	107
4.2.6 模块定义与连接伪指令 ...	79	6.2 读/写存储器与只读存储器.....	108
4.2.7 其他伪指令	79	6.3 半导体存储器接口技术	109
4.3 汇编语言程序设计基本方法	80	6.3.1 存储器地址译码	109
4.3.1 程序设计概述.....	80	6.3.2 存储器与 CPU 的连接	111
4.3.2 顺序结构程序设计	81	小结.....	112
4.3.3 分支结构程序设计	82	习题.....	113
4.3.4 循环结构程序设计	84		
4.3.5 子程序设计	85	第 7 章 输入/输出接口技术	114
4.4 中断调用	87	7.1 接口的基本概念	114
4.4.1 DOS 系统功能调用.....	87	7.1.1 接口电路	114
4.4.2 BIOS 中断调用	89	7.1.2 用接口的原因	114
小结.....	90	7.1.3 接口的功能.....	115
习题.....	90	7.1.4 CPU 与外设之间的信号 ..	116
第 5 章 总线技术	92	7.1.5 接口的基本组成	117
5.1 总线概述	92	7.2 I/O 端口编址方式	118
5.1.1 总线的分类	92	7.3 CPU 与 I/O 设备之间的数据	
5.1.2 总线的组成	93	传送方式	119
5.1.3 总线的性能参数	93	7.3.1 程序传送方式	119
5.1.4 总线标准	94	7.3.2 中断传送方式	122
5.1.5 采用总线结构的优点	94	7.3.3 DMA 传送方式	122
5.2 系统总线	95	小结.....	123
5.2.1 ISA 总线	95	习题.....	123
*5.2.2 EISA 总线	96		
5.3 局部总线	97	第 8 章 中断系统	124
5.3.1 PCI 总线	97	8.1 中断的基本概念	124

8.3 可编程中断控制器 8259A	131	11.1.3 串行通信的信号 传输方式	180
8.3.1 8259A 的引脚信号和 内部结构	131	11.1.4 串行通信的接口标准 ...	181
8.3.2 8259A 的工作方式	133	11.1.5 传送速率与发送/ 接收时钟	181
8.3.3 8259A 的初始化命令字 ...	134	11.1.6 信息的检错与纠错.....	182
8.3.4 8259A 的操作命令字	137	11.1.7 异步通信和同步通信 ...	183
8.3.5 8259A 的初始化流程	139	11.1.8 串行接口的功能.....	184
8.3.6 8259A 在微机系统中的 应用	139	11.2 可编程串行接口芯片 8251A.....	184
小结.....	141	11.2.1 8251A 的基本性能	184
习题.....	142	11.2.2 8251A 外部引脚与 信号功能	185
第 9 章 DMA 控制接口	143	11.2.3 8251A 的内部结构	186
9.1 DMA 控制器概述	143	11.2.4 8251A 的编程.....	187
9.1.1 DMA 传送特点	143	11.3.5 8251A 的初始化	190
9.1.2 DMA 传送机制	144	11.3.6 8251A 应用举例	192
9.2 可编程 DMA 控制器 8237A	146	小结.....	194
9.2.1 8237A 的引脚信号和 内部结构	146	习题.....	194
9.2.2 8237A 的寄存器格式及其 编程命令	150	第 12 章 定时/计数器	195
*9.2.3 8237A 的 DMA 周期	154	12.1 定时/计数概述	195
9.2.4 8237A 的使用和编程	155	12.2 定时/计数器 8253.....	197
小结.....	159	12.2.1 定时/计数器 8253 的 外部特性	197
习题.....	160	12.2.2 8253 的内部逻辑结构与 功能	198
第 10 章 并行通信接口	161	12.2.3 8253 的编程命令与读/写 操作	198
10.1 并行通信与并行接口	161	12.2.4 8253 的工作方式及其 特点	201
10.2 简单并行接口芯片	162	12.2.5 8253 应用举例	205
10.3 可编程并行接口芯片 8255A	164	小结.....	211
10.3.1 8255A 的外部特性和内部 结构	164	习题.....	211
10.3.2 8255A 的编程命令	167	附录 A 汇编语言的开发方法	212
10.3.3 8255A 的工作方式	168	附录 B Proteus 仿真平台简明应用 ...	217
10.3.4 8255A 的应用举例	173	附录 C 8086/8088 指令系统	224
小结.....	177	附录 D 常用 DOS 功能调用 (INT 21H)	228
习题.....	178	附录 E 常用 ROM-BIOS 功能调用	232
第 11 章 串行通信接口	179	附录 F ASCII 码字符表	236
11.1 串行通信的基本概念.....	179	参考文献	237
11.1.1 串行通信的特点	179		
11.1.2 串行通信的数据 传送方式.....	180		

第1章

微机系统概述

计算机是一种高速信息处理和传递的工具，其操作对象是“信息”（或称“数据”），处理（或称“加工”）的结果也是“信息”（或数据）。

计算机之所以获得广泛的应用，是因为其运算速度快、精度高、可以不间断地自动连续工作。完整的计算机系统包括硬件系统和软件系统。硬件是计算机的物质基础，而软件是计算机的灵魂，这两者缺一不可。计算机硬件，只有在相应软件的控制下，才可完成给定的任务，成为对人们真正有用的工具。

本章要点

- 微型计算机（简称微机）的发展过程及各代微处理器的特点；
- 计算机中信息的表示方法；
- 微处理器、微型计算机以及微型计算机系统的组成；
- 微型计算机系统的主要性能评价指标及应用。

计算机是一种能够自动、高速、精确地对数字信息进行加工、处理、存储和传输的电子设备。自 1946 年第一台电子计算机 ENIAC 问世以来，计算机的发展主要经历了电子管、晶体管、中小规模集成电路、大规模和超大规模集成电路等 4 个阶段。进入 21 世纪后，随着生物科学、神经网络技术、纳米技术的飞速发展，生物芯片、神经网络技术进入了计算机领域，计算机的发展进入第 5 个发展阶段。

按照体积、性能和价格来分，计算机可分为巨型机、大型机、中型机、小型机和微型机 5 种。20 世纪 70 年代初期，随着微电子技术和超大规模集成电路技术的发展，促使了以微处理器为核心的微型计算机的诞生。微型计算机不但具有计算机快速、精确、程序控制等特点，而且还具有体积小、质量轻、功耗低、价格便宜等特点。微型计算机现已渗透到国民经济的各个领域，极大地改善了人类的工作、学习和生活方式，成为信息时代的主要标志。

1.1 微机发展概况

微型计算机是指以微处理器为核心，配以存储器、输入/输出（I/O）接口电路及系统总线等设备的计算机。微型计算机采用超大规模集成电路技术，将运算器和控制器——微处理器（Microprocessor）集成在一片硅片上。

随着微电子与超大规模集成电路技术的发展，微型计算机技术的发展基本遵循摩尔定律，微处理器集成度每隔 18 个月翻一番，芯片性能随之提高一倍左右。通常，微型计算机的发展与

微处理器的发展紧密相关，微型计算机的性能主要取决于微处理器的性能。人们通常按微处理器的字长和功能划分微型计算机的发展阶段：

① 第一阶段（1971—1972年）：采用4/8位低档微处理器（如Intel 4004和Intel 8008）为微型计算机的CPU。微处理器主要采用PMOS工艺，集成度为2300元件/片，基本指令执行时间为20~50s，主频在500kHz以下，基本指令48条。

② 第二阶段（1973—1977年）：采用8位中档微处理器为微型计算机的CPU。微处理器的代表芯片是MC6800、Z80、Intel 8080/8085等，采用NMOS工艺，集成度较上一代提高4倍，基本指令执行时间为2~10μs，主频高于1MHz，基本指令70多条。其流行机种是TRS-80和Apple II。

③ 第三阶段（1978—1984年）：采用16位微处理器为微型计算机的CPU。微处理器代表芯片是Intel 8086/8088、MC6800、Z8000，采用HMOS工艺，集成度2~7万元件/片，基本指令执行时间为0.5μs，主频4~8MHz，采用这代微处理器的计算机指令系统完善，采用流水线技术、多级中断、多种寻址方式、段寄存器等结构，能够与协处理器相配合进行浮点运算。其流行机种是IBM PC和IMB PC/XT。

④ 第四阶段（1985—1992年）：采用32位微处理器为微型计算机的CPU。代表芯片是Intel 80386、Intel 80486、MC68040等，采用HMOS/CMOS工艺，集成度100万元件/片，基本指令执行速度25MIPS，主频16~25MHz，引入了高速缓存，采用精简指令集，其体系结构较16位机发生了概念性变化。流行机种是PC386和PC486。

⑤ 第五阶段（1993年至今）：采用32位Pentium微处理器P5为微型计算机的CPU。芯片采用0.6μm的静态CMOS工艺，集成度350万元件/片，基本指令执行时间为0.5μs，主频60MHz以上，采用扩展总线，设置高速程序缓存、数据缓存、超流水线结构。1995年推出的Pentium Pro系列微处理器P6，主频133MHz，设置两级缓存，采用动态执行技术，性能大大提高。而后又推出了Pentium MMX、Pentium II、Pentium III、Pentium 4。目前，Intel系列的微处理器中，主频已达3.8GHz。表1-1给出了80x86/Pentium系列部分CPU的主要性能参数。

表1-1 80x86/Pentium系列部分CPU的主要性能参数

微处理器	推出时间/年	生产工艺/μm	时钟频率/MHz	集成度/（万元件/片）	寄存器位数/位	数据总线宽度/位	最大寻址空间	高速缓存大小
8086	1978	10	8	0.040	16	16	1 MB	无
80286	1982	2.7	12.5	0.125	16	16	16 MB	无
80386DX	1985	2	20	0.275	32	32	4 GB	无
80486DX	1989	1、0.8	25	1.200	32	32	4 GB	8 KB L1
Pentium	1993	0.8、0.6	60	3.100	32	64	4 GB	16 KB L1
Pentium Pro	1995	0.6	200	5.500	32	64	64 GB	16 KB L1/256 KB L2
Pentium II	1997	0.35	300	7.500	32	64	64 GB	32 KB L1/256 KB L2
Pentium III	1999	0.18	500	9.500	32	64	64 GB	32 KB L1/256 KB L2
Pentium 4	2000	0.13	1300	42.00	32	64	64 GB	128 KB L1/512 KB L2

微型计算机的发展之所以如此迅速，主要取决于微型计算机具有运算速度快、计算精度高、集成度高、造价低廉等特点。又由于微型计算机硬件平台开放、易于扩展、适应性强，因此微

处理器的配套应用芯片和软件丰富，更新也很快。此外，微型计算机还具有体积小、质量轻、耗电少及维护方便等特点。

当前微型计算机和微处理器朝以下几个方向发展：

- ① 发展高性能的 64 位微处理器；
- ② 发展专用化的单片微型计算机；
- ③ 发展带有固件的微型计算机；
- ④ 发展多微处理机系统；
- ⑤ 充实和发展外围接口电路。

1.2 计算机中信息的表示方法

目前使用的计算机是一种电设备，由千千万万个电子元件（如电容、电感、二极管、晶体管等）组成，这些电子元件一般都只有两种稳定的工作状态（如二极管的截止和导通），用这两种状态对应的高、低两个电位表示 0 和 1，这在物理上最容易实现。0 或 1 可用二进制数的一位表示，称为 bit[1 个二进制位称为 1bit；8 个二进制位称为一个字节 (B)；多个字节组成一个字 (Word)]。因此，在计算机中，任何信息都是 0 和 1 的数字组合形式，即计算机存储和处理的全部是二进制信息。但是在实际应用中，需要计算机处理的信息是多种多样的，如各种进位制的数据，不同语种的文字符号和各种图像信息等。计算机中用各种二进制编码来表示这些信息。

为了便于书写和记忆，在计算机中数的表示还广泛采用十进制、八进制和十六进制等。为了区别所使用的数制，常用下标或数制代号标注，例如用 $(216)_{10}$ 或 $(216)_D$ 表示十进制数。如采用代码标注，十进制用 D 表示；二进制用 B 表示；八进制用 Q 或 O 表示；十六进制用 H 表示。十进制数的下标 D 可以省略。

1.2.1 数制及其转换

1. 进位计数制

(1) 计数符号

每一种进位计数制都有固定数目的计数符号。

- ① 十进制：10 个计数符号，0、1、2、…、9；
- ② 二进制：2 个计数符号，0 和 1；
- ③ 八进制：8 个计数符号，0、1、2、…、7；
- ④ 十六进制：16 个计数符号，0~9、A、B、C、D、E、F，其中 A~F 对应十进制的 10~15。对于字母开头的十六进制数，还必须在数据前加个 0，以表明它是十六进制数而不是其他符号。

(2) 权值

在任何进制中，一个数的每个位置都有一个权值。例如，十进制数 57892 的值为：

$$(57892)_{10} = 5 \times 10^4 + 7 \times 10^3 + 8 \times 10^2 + 9 \times 10^1 + 2 \times 10^0$$

从右向左，每一位对应的权值分别为 10^0 、 10^1 、 10^2 、 10^3 、 10^4 。

不同的进位计数制由于其进位的基数不同，其权值也是不同的。例如，二进制数 101101，其值应为：

$$(101101)_2 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

从右向左，每个位对应的权值分别为 2^0 、 2^1 、 2^2 、 2^3 、 2^4 、 2^5 。

(3) 基数

在任何进制中，基数是指这种进位计数制中计数符号的个数。例如，十进制的基数为10，二进制的基数为2，八进制的基数为8，十六进制的基数为16。每种进位计数制遵循“逢基数进一”的法则，如十进制数“逢十进一”、二进制数“逢二进一”等。

2. 不同数制的相互转换

(1) 二、十六进制转换为十进制

将二、十六进制转换为十进制采用“按权展开求和”的方法，即将每位数码乘以各自的权值并累加。

【例 1.1】 将 $(1101.1)_2$ 和 $(A3C.F5)_{16}$ ，分别转换成十进制数。

$$\begin{aligned}(1101.1)_2 &= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} \\&= 8 + 4 + 1 + 0.5 \\&= (13.5)_{10} \\(A3C.F5)_{16} &= 10 \times 16^2 + 3 \times 16^1 + 12 \times 16^0 + 15 \times 16^{-1} + 5 \times 16^{-2} \\&= 2560 + 48 + 12 + 0.9375 + 0.01953125 \\&= (2620.95703125)_{10}\end{aligned}$$

(2) 十进制转换为二、十六进制

十进制数转换为其他进制数时整数部分和小数部分必须分别遵守不同的转换规则。整数部分采用“除以基数取余数”的方法，小数部分采用“乘以基数取整数”的方法。假设将十进制数转换为 R 进制数。

① 整数部分：除以 R 取余数，即整数部分不断除以 R 取余数，直到商为0为止，最先得到的余数为最低位，最后得到的余数为最高位。

② 小数部分：乘以 R 取整数，即小数部分不断乘以 R 取整数，直到积为0或达到有效精度为止，最先得到的整数为最高位（最靠近小数点），最后得到的整数为最低位。

【例 1.2】 将 $(75.453)_{10}$ 转换成二进制数（取4位小数）。

整数部分	取余数		小数部分	取整数	
2 75	1	低	$\times \frac{0.453}{2}$		
2 37	1		<u>0.906</u>	0	
2 18	0		$\times \frac{0.906}{2}$		
2 9	1	↑	<u>1.812</u>	1	
2 4	0		$\times \frac{1.812}{2}$		
2 2	0		<u>1.624</u>	1	
2 1	1	高	$\times \frac{1.624}{2}$		
0			<u>1.248</u>	1	低

得 $(75.453)_{10} = (1001011.0111)_2$

【例 1.3】 将 $(237.45)_{10}$ 转换成十六进制数（取3位小数）。

整数部分	取余数		小数部分	取整数	
16 237	13	低	$\times \frac{0.45}{16}$		
16 14	14	↑	<u>7.20</u>	7	
0		高	$\times \frac{7.20}{16}$		
			<u>3.20</u>	3	
			$\times \frac{3.20}{16}$		
			<u>3.20</u>	3	低

得 $(237.45)_{10} = (ED.733)_{16}$

(3) 二进制转换为十六进制

因为 $2^4=16$, 所以 4 位二进制数对应 1 位十六进制数。

将二进制数以小数点为中心分别向两边分组, 转换成十六进制数, 每 4 位为一组, 不够位数在两边加 0 补足, 然后将每组二进制数转化成十六进制数即可。

【例 1.4】 将二进制数 110111110.10001 转换为十六进制数。

$$(0011 \underline{0111} \underline{1110}.\underline{1000} \underline{1000})_2 = (37E.88)_{16} \text{ (注意: 在两边补零)}$$

3 7 E . 8 8

(4) 十六进制转换为二进制

将每位十六进制数展开为 4 位二进制数。注意: 可以将转换之后的二进制整数前的高位零和小数后的低位零取消。

【例 1.5】 将 $(B3E.A5)_{16}$ 转化成二进制数。

$$(B3E.A5)_{16} = (1011 \underline{0011} \underline{1110}.\underline{1010} \underline{0101})_2$$

1.2.2 计算机中二进制信息编码

所谓二进制信息编码, 是指用二进制代码来表示计算机所要处理的信息。在计算机里, 所有数字、字母、符号、操作命令等都是用二进制特定编码(一般表示为若干位二进制码的组合)来表示的。

1. 数字编码(BCD 码, Binary Coded Decimal)

BCD(二-十进制)码是一种常用的数字代码, 它广泛应用于计算机中。这种编码法分别将每位十进制数字编成 4 位二进制代码, 从而用二进制数来表示十进制数。

计算机中采用的是二进制数, 由于二进制数不直观, 人们不习惯, 因此计算机在输入和输出时, 通常仍采用十进制数, 只不过它要用二进制编码来表示, 这时使用 BCD 码就很方便。

最常用的 BCD 码是标准 BCD 码或称 8421 码(这是根据这种表示中各位的权值而定的, 其权值与普通的二进制相同)。表 1-2 列出了标准 BCD 码与十进制数的编码关系。

表 1-2 标准 BCD 码与十进制数的编码关系

十进制数	标准 BCD 码	十进制数	标准 BCD 码
0	0000	6	0110
1	0001	7	0111
2	0010	8	1000
3	0011	9	1001
4	0100	10	00010000
5	0101	100	000100000000

十进制数基数为 10, 它有 10 个不同的数码。因此为了能表示十进制数的某一位, 必须至少选择 4 位二进制数(4 位二进制数可以表示 16 种不同的状态, 所以用以表示十进制数时要丢掉 6 种状态)。在 BCD 码中, 0~9 之间的十进制数的 BCD 码与二进制数中的表示形式是一样的, 而 1010~1111 这 6 种状态不使用, 因此用标准 BCD 码表示十进制数时, 只要对每个十进制数字用适当的二进制数代替即可。例如, 十进制数 320 可以表示成 0011 0010 0000。

为了避免格式与纯二进制码混淆，通常在每 4 位二进制数之间留一空格，这种表示也适合十进制小数。例如，十进制小数 0.857 可以表示成 0.1000 0101 0111。

2. 字符的编码

计算机中最通用的字符信息编码为美国标准信息交换码(American Standard Code for Information Interchange, ASCII)，如附录 F 所示。这种代码用一个字节(8位二进制码)来表示一个字符，其中低 7 位是字符的 ASCII 码值，例如英文字母 A 是 41H。小于 20H 的是不可显示字符，通常是命令代码，如 0AH 是换行命令符。最高位一般用做奇偶检验位或者用于 ASCII 码的扩充。基本 ASCII 码有 128 个，包括大小写字母、数字、专用符号和控制符号等。扩充后的 ASCII 码有 256 个(扩充出了 128 个字符和图形符号)。

3. 汉字编码

计算机在我国应用时，要求其能够输入、处理和输出汉字。汉字通过输入设备将外码送入计算机，再由汉字系统将其转换成内码存储、传送和处理，当需要输出时再由汉字系统调用字库中汉字的字形码得到结果，这个过程如图 1-1 所示。

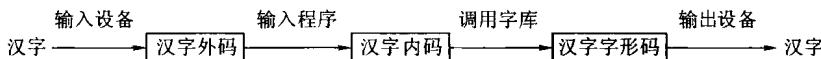


图 1-1 汉字处理流程

只有在中文操作系统环境下才能处理汉字，操作系统中有实现各种汉字代码间转换的模块，在不同场合下调用不同的转换模块。

1.2.3 计算机中数的表示

1. 数据格式

(1) 数的定点表示和浮点表示

在计算机中，对小数点的处理有两种，分别称为定点格式和浮点格式。

① 在定点格式中，小数点在数据中的位置固定不变。定点格式可表示成定点小数(小数点约定在符号位之后)或定点整数(小数点约定在最低位之后)。通常，小数点的位置确定后，在运算中不再考虑小数点的问题，因而小数点不占用存储空间。定点数表示简单，但数的取值范围小，精度低。

② 采用浮点格式的机器中的数据的小数点位置可变。浮点数的一般格式为：

$$N=R^E \cdot M$$

其中， N 为浮点数或实数； M 是浮点数的尾数；是纯小数； E 是浮点的指数，是整数；基数 R 是常数。

机器中的浮点数用尾数和阶码及其符号位表示。尾数用定点小数表示，尾数给定有效数字的位数并决定浮点数的表示精度；阶码用定点整数表示，指明小数点在数据中的位置并决定浮点数的表示范围。

(2) 带符号数和无符号数

对于一个数来说若最高有效位为符号位，则该数为带符号数；反之，若数的最高有效位为数值位，则为无符号数。当进行数据处理时，若不需要考虑数的正负，则可以使用无符号数。带符号数和无符号数的取值范围不同。对于字长为 8 位的定点整数，无符号数的取值范围是 $0 \leq X \leq 255$ ，有符号数的取值范围是 $-128 \leq X \leq 127$ 。

2. 机器数与真值

机器数是一个数在计算机中的表示形式，一个机器数所表示的数值称为真值。前面提到的二进制数，没有提到符号问题，因此是一种无符号数的表示。对于无符号数，机器数与真值相同，此时计算机的全部有效位都用来存放数据，它能表示的最大数值取决于计算机的字长。对于 n 位字长的计算机来说，表示无符号的整数范围为 $0 \sim 2^n - 1$ 。带符号数的习惯表示方法是在数值前用“+”号表示正数，“-”号表示负数。计算机只能识别 0 和 1，对数值的符号也不例外。对于带符号的数，在计算机中，通常将一个数的最高位作为符号位，最高位为 0，表示符号位为正；最高位为 1，表示符号位为负。例如：

真值	机器数
+83	0 1010011
-83	1 1010011

式中等号左边的 +83 和 -83 分别是等号右边的机器数所代表的实际数，即真值。

这种把正、负号也数字化的数称为机器数，即计算机所能识别的数。

3. 原码、反码与补码

计算机中常用机器数有 3 种不同的编码形式，即原码、反码和补码。

(1) 原码

上述以最高位为 0 表示正数，为 1 表示负数，后面各位为其数值的表示法称为原码表示法。

原码简单，与真值转换方便。但是若两个异号数相加或两个同号数相减时，必须做减法。在计算机内部，为了避免做减法，即用一个加法器来完成加减法运算，便引入了反码和补码。

(2) 反码

对于正数其反码形式与其原码相同，最高位 0 表示正数，其余位为数值位。

对于负数将其原码除符号位以外，其余各位按位取反，即可得到其反码表示形式。

(3) 补码

正数的补码与其原码具有相同的表现形式，最高位为符号位，其余位为数值位。例如：

$$X = +127, \quad [X]_{原} = [X]_{反} = [X]_{补} = 0\ 1111111$$

$$X = +0, \quad [X]_{原} = [X]_{反} = [X]_{补} = 0\ 0000000$$

负数的补码即为它的反码在最低位加上 1。例如：

$$X = -7, \quad [X]_{原} = 1\ 0000111, \quad [X]_{反} = 1\ 1111000, \quad [X]_{补} = 1\ 1111001$$

$$X = -0, \quad [X]_{原} = 1\ 0000000, \quad [X]_{反} = 1\ 1111111, \quad [X]_{补} = 0\ 0000000$$

$$X = -128, \quad [X]_{补} = 1\ 0000000$$

从以上几例可归纳出二进制补码的几个特点：

① $[+0]_{补} = [-0]_{补} = 00000000$ ，无 +0 和 -0 之分。

② 正因为补码中没有 +0 和 -0 之分，所以 8 位二进制补码所能表示的数值范围为 +127 ~ -128；同理可知， n 位二进制补码表示的范围为 $+2^{n-1} - 1 \sim -2^{n-1}$ 。在原码、反码和补码三者中，只有补码可以表示 -2^{n-1} 。

③ 一个用补码表示的二进制数，当为正数时，最高位（符号位）为“0”，其余位即为此数的二进制值；当为负数时，最高位（符号位）为“1”，其余位不是此数的二进制值，必须把它们按位取反，且在最低位加 1，才是它的二进制值。

4. 溢出的概念

在计算机中，凡是有符号数一律用补码形式存放和运算，其运算结果也用补码表示。若最高位为 0，表示结果为正；若最高位为 1，表示结果为负。设 X 、 Y 是两个任意的二进制数，定点补码的运算满足下面的规则：

$$[X+Y]_{\text{补}} = [X]_{\text{补}} + [Y]_{\text{补}}$$

$$[X-Y]_{\text{补}} = [X]_{\text{补}} + [-Y]_{\text{补}}$$

采用补码运算可以将减法变成补码加法运算，在微处理器中只需要加法的电路就可以实现加法、减法运算。但由于计算机的字长有一定限制，所以一个带符号数是有一定范围的。当运算结果超过这个范围时，便产生溢出。显然，只有在同符号数相加或者异符号数相减的情况下，才有可能产生溢出（如果两个同符号数相加，结果的符号与之相反，则溢出；如果两个异符号数相减，结果的符号与减数相同，则溢出）。在计算机中也可以利用运算时的进位情况判断是否溢出：当数值部分向符号位的进位与符号位向高位的进位不相同时则产生溢出，因此可以用一个异或电路进行检测。

【例 1.6】两个 8 位补码表示的数相加没有溢出的例子。

$$\begin{array}{r}
 & & & & & C_{S+1} \\
 & & & & & \downarrow \\
 & & & & & C_S \\
 & & & & 11 \\
 & & & & \downarrow \\
 10001010 & & & & -118 \\
 + & 01111001 & & & +121 \\
 \hline
 & & & & +3
 \end{array}$$

令 C_S 为数值部分向符号位的进位， C_{S+1} 为符号位向高位的进位，此例中， $C_S = C_{S+1} = 0$ ，结果在 8 位二进制补码表示范围内，没有溢出。

【例 1.7】两个 8 位补码表示的数相加产生溢出的例子。

$$\begin{array}{r}
 & & & & & C_{S+1} \\
 & & & & & \downarrow \\
 & & & & & C_S \\
 & & & & 01 \\
 & & & & \downarrow \\
 01111110 & & & & +126 \\
 + & 00000101 & & & +5 \\
 \hline
 & & & & -125
 \end{array}$$

此例中， $C_S \neq C_{S+1}$ ，产生了错误的结果，发生了溢出。

1.3 微机的基本结构

1.3.1 微型计算机系统

1946 年，美籍匈牙利数学家冯·诺依曼 (John von Neumann) 等人在论文《关于电子计算机器逻辑设计的初步探讨》中，第一次提出了计算机组成和工作方式的基本思想。

其主要思想是：

- ① 计算机应由运算器、控制器、存储器、输入设备和输出设备这五大部分组成。
- ② 存储器不但能存放数据，而且也能存放程序。数据和指令均以二进制数形式存放，计算机具有区分指令和数据的能力。
- ③ 编好的程序事先存入存储器中，在指令计数器控制下，自动高速运行（执行程序）。

以上几点可归纳为“程序存储，程序控制”的构思。

数十年来，虽然计算机已经取得惊人的进展，相继出现了各种结构形式的计算机，但究其本质，仍属冯·诺依曼结构体系。

众所周知，计算机由硬件和软件两大部分组成。硬件是指那些为组成计算机而有机联系的电子、电磁、机械、光学元件、部件或装置的总和，是有形的物理实体。软件是相对于硬件而言的。从狭义角度看，软件包括计算机运行所需的各种程序；而从广义角度讲，软件还包括手册、说明书和有关资料。

微机系统是硬件和软件有机结合的整体。没有软件的计算机称为裸机，裸机如同一架没有思想的躯壳，不能做任何工作。操作系统给裸机以灵魂，使其成为真正可用的工具。一个应用程序在计算机中运行时，受操作系统的管理和监控，在必要的系统软件协助之下，完成用户交给它的任务。可见，裸机是微机系统的物质基础，操作系统为它提供了一个运行环境。在系统软件中，各种语言处理程序为应用软件的开发和运行提供方便。用户并不直接和裸机打交道，而是使用各种外围设备（简称外设），如键盘和显示器等，通过应用软件与计算机交流信息。

硬件和软件系统本身还可细分为更多的子系统，如图 1-2 所示。

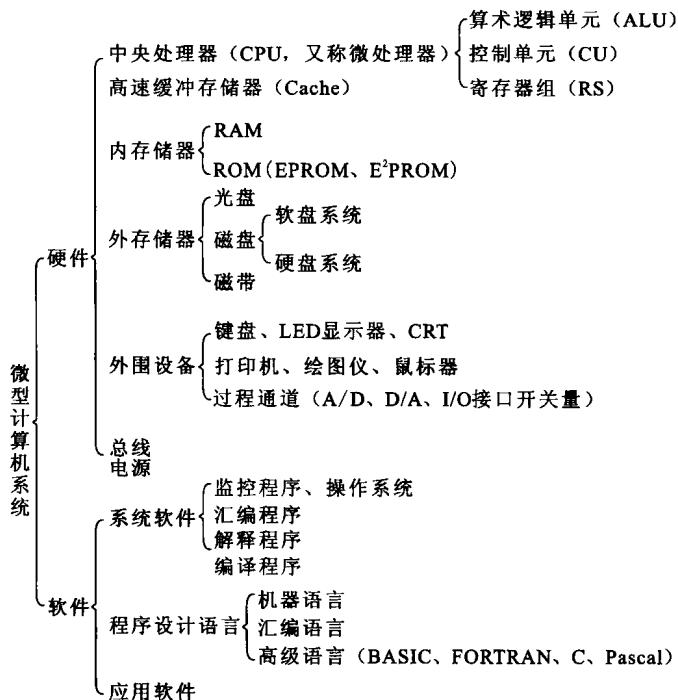


图 1-2 微型计算机系统的组成