



普通高等教育国家级规划教材

微型计算机原理与接口技术

WEIXING JISUANJI YUANLI YU JIEKOU JISHU

第6版

► 周荷琴 冯焕清 / 编著

中国科学技术大学出版社



普通高等教育国家级规划教材

微型计算机原理与接口技术

第6版

周荷琴 冯焕清 / 编著

中国科学技术大学出版社

内 容 简 介

本书是为中国科学技术大学工科电子类专业本科学生学习“微型计算机原理与系统”课程而编写的教材。从初版开始至每次修订再版,都是作者在参考国内外大量文献、资料的基础之上,吸取各家之长,并结合教学团队多年教学和应用研究的经验,精心组织编写而成的,可谓自成一统。全书内容丰富,图文并茂,讲述深入浅出,通俗易懂,并附有大量的实例和习题,部分习题还给出了解题提示,既可用于教材,也适合于自学,先后被列入“普通高等教育国家级规划教材”和“中国科学院指定考研参考书”。

全书 15 章,内容安排上注重系统性、先进性和实用性。前 5 章是基础部分,主要介绍微型机系统的组成原理、体系结构、8086 指令系统、汇编语言程序设计方法以及存储器的原理和电路设计。第 6~12 章讨论接口和总线技术,包括中断、DMA 和 I/O 接口,几个典型的大规模集成电路接口芯片(8255A、8253/8254、8259A、8251A、8237A 等),A/D 和 D/A 以及总线技术也被纳入其中。最后 3 章介绍高档微型机的工作原理,其中第 13 章包括 32 位微处理器的寄存器组成、保护模式下的内存管理、保护模式下的中断和异常以及任务切换等内容;第 14 章介绍 32 位机新增指令、浮点数、SIMD 技术和指令系统,并给出了许多编程实例;第 15 章简要介绍了 PC/XT 机的系统结构,主要对 32 位微型机的结构以及 64 位 CPU 和智能多核处理器进行了讨论,并概要阐述了 64 位机的系统结构和技术特点。

本书可作为高等学校电子类专业和其他相近相关专业本科教育的教材,也可作为从事微型计算机系统设计 and 应用等相关科技工作者的参考书。

图书在版编目(CIP)数据

微型计算机原理与接口技术/周荷琴,冯焕清编著.—6 版.—合肥:中国科学技术大学出版社,2019.2

普通高等教育国家级规划教材,中国科学院指定考研参考书

ISBN 978-7-312-04612-4

I. 微… II. ①周… ②冯… 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. TP36

中国版本图书馆 CIP 数据核字(2019)第 001742 号

出版 中国科学技术大学出版社

安徽省合肥市金寨路 96 号,230026

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

<http://zgkxjdsdxcs.tmall.com>

印刷 合肥华苑印刷包装有限公司

发行 中国科学技术大学出版社

经销 全国新华书店

开本 787 mm×1092 mm 1/16

印张 31.75

字数 832 千

版次 1996 年 9 月第 1 版 2019 年 2 月第 6 版

印次 2019 年 2 月第 47 次印刷

印数 378 001—383 000 册

定价 66.00 元

第 6 版前言

《微型计算机原理与接口技术》第 5 版自 2013 年 1 月由中国科学技术大学出版社出版以来,继续被全国许多高等院校选为教材,受到广大读者和同行老师的热情欢迎和支持,同时大家也回馈了不少有益的意见和建议,在此谨表谢意!

随着计算机和网络技术的高速发展,诞生了物联网、云计算和互联网+等新概念,人工智能应运而生,它将为“万物互联”之后的应用问题提供最完美的解决方案。这些新领域的发展不断地对芯片制造商提出高要求:更多的计算功能,更高的计算速度,更低的功率消耗,更小的芯片体积。然而芯片制造商正面临着摩尔定律的严峻挑战,10 nm 制程的微处理器姗姗来迟。尽管如此,计算机行业还是在不断创新,一款款高性能的智能酷睿处理器相继推出,传统 CPU 正在与 GPU、DSP、FPGA 等处理器深度融合,形成异构多核心处理器,微处理器的发展开始进入 CPU+ 的时代。为适应微型计算机技术这种飞速发展的形势和各高校不断深化的教育改革的需要,我们在 Intel 8086 微处理器诞生 40 年之际,对本教材进行了一次全面的修订。

全书的总体结构没有改变,依然是 15 章,分成 3 个部分,即第 1~5 章的基础部分,第 6~12 章的接口和总线技术,第 13~15 章的高档微型机原理。

本次修订的主要内容包括:

1. 适当进行了篇幅压缩,去掉一些陈旧的或较难掌握的内容,所有的表述尽可能简洁精练。例如,简化了 Cache 地址映射方案,略去了影响 Cache 性能的因素,删除了 8255 方式 2 的详细介绍及其在 PC/XT 中的应用、硬件 UART 框图、8251A 与异步 MODEM 的连接、DMA 控制器简介、DAC0832 直通方式、USB 的设备枚举、采样保持原理、附录 A 的指令一览表以及少数难度较大的例题和习题。

2. 对部分内容进行了梳理和归并。例如,删去了部分已过时总线规范的介绍,并对 12 章的内容做了归并;把 13.1 节“32 位微处理器的结构和工作模式”前移到了第 2 章。

3. 加进了反映计算机技术最新发展的内容。例如,增加了这几年推出的DDR4内存,PCI-E 4.0、USB 3.1、USB 3.2接口标准等方面的内容,特别是在第15章中,对高档机的内容做了较多的修改和完善,系统地介绍了9代智能酷睿多核处理器的功能与特点,并对Intel的“Tick-Tock”策略做了概括。

作者谨向本书编写和修订过程中参考过的著作和资料的作者以及为本书编写和修订作出重要贡献的所有老师和学生表示衷心感谢!

编 者

2018年10月于合肥

编者注:应众多兄弟院校要求,我们专门制作了本书的教学课件,由中国科学技术大学出版社免费提供给所有选用本书作为教材的授课老师,以方便教学。联系方式:press@ustc.edu.cn 或 sjzhang@ustc.edu.cn。

第 5 版前言

本书是为中国科学技术大学工科电子类本科生学习“微型计算机原理与系统”课程编写的教材,被列入“普通高等教育‘十一五’国家级规划教材”和“中国科学院指定考研参考书”。自 1996 年出版以来,本书先后被很多高校选作本科教育教材,得到了广大读者和同行老师、学者的热情支持,同时大家也反馈回不少宝贵的意见和建议,在此谨表示感谢!为适应微型计算机技术飞速发展的形势和教育教学改革不断深化的需要,我们再次对原书进行了全面的修订。

自 20 世纪 70 年代第一代微型计算机问世以来,计算机技术以惊人的速度发展,涌现了数十个品种、几百个型号的微处理器,数据宽度从 8 位、16 位、32 位发展到了 64 位,处理器芯片的 CPU 核心发展到了双核乃至 4 核、6 核和 8 核,当前微型计算机的发展已经进入了智能多核时代。希望通过这次修订,本书能更系统地归纳和清晰地展示已经发展了 40 多年的计算机技术,能更深入浅出地讲清楚那些看似深奥的计算机知识,从而真正有助于教师们的课堂教学和学生们的课后阅读。通过努力我们顺利完成了修订工作,新版《微型计算机原理与接口技术》将以全新的面貌与读者见面。

全书共 15 章,内容包括 3 部分:第 1~5 章是基础部分,仍以 8086 为主要对象,包括微型计算机的基础知识、8086 CPU、寻址方式、指令系统、汇编语言程序设计和存储器。第 6~12 章讨论了接口和总线技术,包括中断、DMA 和 I/O 接口,典型的大规模集成电路接口芯片(如:8255A、8253/8254、8259A、8251A、8237A),A/D 和 D/A 以及总线技术等。第 13~15 章介绍了高档微型机的原理,包括 32 位微型机的工作原理、指令系统与程序设计以及 64 位 CPU 和智能多核处理器,也包括 16 位、32 位和 64 位机的系统结构和技术特点。

本次修订的主要内容包括:

1. 对第 1、2、4、5、8、12、15 章的全部内容以及第 6、13、14 章的部分内容重新进行了组织和编写,尽可能删繁就简,循序渐进,将现今最新的技术包含进来,表述上没有冗余,文字简练顺畅,全书一气呵成。

2. “存储器”一章不仅包含了最新的存储器技术,还对目前在嵌入式系统领域广泛应用的串行 EEPROM 做了较详细的介绍;考虑到 Cache 技术在现代 CPU 技术中的地位越来越重要,对其工作原理的讨论也比较深入。

3. 将 I/O 接口技术与并行接口芯片 8255 合并形成了第 6 章,并将几个简单接

口芯片的介绍前移到第2章。

4. “总线技术”一章对已淘汰的总线标准只做了简单回顾,重点讨论了当前流行的系统总线 PCI、PCI Express 以及串行总线 USB 和 IEEE 1394。

5. 第13章“32位微型计算机原理”,对保护模式下的中断和异常进行了较详细的描述,并增加了任务切换的内容。

6. 第14章“32位微型计算机的指令系统与程序设计”,在原有的32位微型计算机新增指令和程序设计的基础上,增加了 IEEE 标准 754 浮点数的表示方法、奔腾处理器的 SIMD 技术、SIMD 指令系统、SIMD 程序设计实例等内容。

7. 第15章在原有 PC/XT 机以及 32 位微型计算机系统结构的基础上,增加了 64 位 CPU 和多核处理器的内容,并着重介绍了 64 位微型机的系统结构、芯片组和主板,内容涉及最新的 i7、i5 和 i3 智能多核处理器。

8. 对第4章和第14章中的16位和32位微型计算机的所有汇编语言程序实例都进行了精心设计,并全部进行了上机验证。

9. 对原书保留章节中的绝大部分插图进行了修正。

10. 修改了原书保留章节中的部分内容和习题,并为许多习题增加了解题的提示信息,读者根据例题和提示信息,不难得出正确答案。

吴秀清教授为本书此前的版本做出了重要贡献,特此感谢!

在本书撰写过程中,青年教师李峰、何力以及研究生潘剑锋、刘冰啸、乔赫元、袁非牛、王鹏、刘勃、刘学亮、王恒良、郭永刚、赵何、黄庆华、卢胜、陈立群、梅涛、武海澄、陈功等在资料的收集、例题的验证、程序的调试、插图的绘制、多媒体课件的制作等方面做了许多工作,并对书中的内容提出了不少有益的建议,在此一并表示衷心的感谢!

此外,在本书编写过程中,我们查阅、参考了大量国内外相关的文献以及网上资料,在此特向相关作者表示深切的感谢!

由于作者水平有限,书中错误和不当之处在所难免,敬请读者批评指正,以便日后再版时予以修正。

编 者

2013年1月于合肥

目 录

第 6 版前言	(i)
第 5 版前言	(iii)
第 1 章 微型计算机的基础知识和发展概况	(1)
1.1 计算机中数的表示方法	(1)
1.1.1 进位计数制	(1)
1.1.2 二进制编码	(3)
1.1.3 带符号数的表示方法	(4)
1.2 计算机的基本结构和软件	(6)
1.2.1 计算机的基本结构	(6)
1.2.2 计算机软件	(8)
1.3 微型计算机结构和系统	(10)
1.3.1 微型计算机基本结构	(11)
1.3.2 微型计算机系统	(15)
1.4 微型计算机的发展概况	(16)
1.4.1 计算机的发展	(16)
1.4.2 微型计算机的发展	(18)
第 2 章 微处理器的结构和工作模式	(22)
2.1 8086 CPU 的内部结构和存储器组织	(22)
2.1.1 8086 CPU 内部结构及工作过程	(22)
2.1.2 8086 CPU 内部寄存器	(24)
2.1.3 8086/8088 CPU 的引脚功能	(27)
2.1.4 8086 的存储器组织	(31)
2.2 8086 的工作模式和总线操作	(35)
2.2.1 最小模式系统	(36)
2.2.2 最大模式系统	(40)
2.2.3 总线操作时序	(42)
2.3 32 位微处理器的结构与工作模式	(45)
2.3.1 32 位微处理器结构简介	(46)
2.3.2 32 位微处理器的工作模式	(50)

第 3 章 8086 的寻址方式和指令系统	(54)
3.1 8086 的寻址方式	(54)
3.1.1 立即寻址方式	(54)
3.1.2 寄存器寻址方式	(55)
3.1.3 存储器寻址方式	(55)
3.1.4 其它寻址方式	(61)
3.2 指令的机器码表示方法	(62)
3.2.1 机器语言指令的编码目的和特点	(62)
3.2.2 机器语言指令代码的编制	(63)
3.3 8086 的指令系统	(66)
3.3.1 数据传送指令	(66)
3.3.2 算术运算指令	(74)
3.3.3 逻辑运算和移位指令	(87)
3.3.4 字符串处理指令	(92)
3.3.5 控制转移指令	(96)
3.3.6 处理器控制指令	(112)
第 4 章 汇编语言程序设计	(117)
4.1 汇编语言程序格式和伪指令	(118)
4.1.1 汇编语言程序格式	(118)
4.1.2 伪指令语句	(122)
4.1.3 完整的汇编语言程序框架	(127)
4.2 DOS 系统功能调用和 BIOS 中断调用	(131)
4.2.1 DOS 的层次结构	(132)
4.2.2 DOS 系统功能调用	(132)
4.2.3 BIOS 中断调用	(137)
4.3 汇编语言程序设计方法与实例	(141)
4.3.1 顺序结构程序设计	(141)
4.3.2 分支程序设计	(143)
4.3.3 循环结构程序	(146)
4.3.4 代码转换程序	(149)
4.3.5 过程调用	(152)
第 5 章 存储器	(158)
5.1 存储器分类	(158)
5.1.1 内部存储器	(158)
5.1.2 外部存储器	(160)
5.1.3 存储器的性能指标	(163)
5.2 随机存取存储器 RAM	(163)

5.2.1 静态 RAM(SRAM)	(164)
5.2.2 动态 RAM(DRAM)	(165)
5.2.3 内存条	(169)
5.3 只读存储器 ROM	(171)
5.3.1 可编程可擦除 ROM(EPROM)	(172)
5.3.2 电可擦除可编程 ROM(EEPROM)	(175)
5.4 存储器与 CPU 的连接	(178)
5.4.1 设计接口应考虑的问题	(178)
5.4.2 存储器接口设计	(179)
5.5 高速缓冲存储器	(186)
5.5.1 高速缓存的原理	(186)
5.5.2 高速缓存的基本结构	(188)
5.5.3 主存与 Cache 的地址映射	(189)
5.5.4 Cache 的基本操作	(191)
第 6 章 I/O 接口和并行接口芯片 8255A	(194)
6.1 I/O 接口	(194)
6.1.1 I/O 接口的功能	(194)
6.1.2 I/O 端口及其寻址方式	(195)
6.1.3 CPU 与外设间的数据传送方式	(197)
6.1.4 PC 机的 I/O 地址分配	(201)
6.2 8255A 的工作原理	(204)
6.2.1 8255A 的结构和功能	(204)
6.2.2 8255A 的控制字	(206)
6.2.3 8255A 的工作方式	(208)
6.3 8255A 的应用举例	(214)
6.3.1 基本输入输出应用举例	(214)
6.3.2 键盘接口	(217)
第 7 章 可编程计数器/定时器 8253/8254	(222)
7.1 8253 的工作原理	(222)
7.1.1 8253 的内部结构和引脚信号	(222)
7.1.2 初始化编程步骤和门控信号的功能	(226)
7.1.3 8253 的工作方式	(227)
7.2 8253/8254 的应用举例	(231)
7.2.1 8253 定时功能的应用举例	(232)
7.2.2 8253/8254 计数功能的应用举例	(234)
7.2.3 8253 在 PC/XT 机中的应用	(238)

第 8 章 中断和可编程中断控制器 8259A	(244)
8.1 中断	(244)
8.1.1 中断概念和分类	(244)
8.1.2 中断的响应与处理过程	(248)
8.2 8259A 的工作原理	(252)
8.2.1 8259A 的引脚信号和内部结构	(252)
8.2.2 8259A 的工作方式	(254)
8.2.3 8259A 的命令字及编程	(256)
8.3 8259A 应用举例	(263)
8.3.1 8259A 的级联使用	(263)
8.3.2 中断向量的设置和中断处理程序设计实例	(266)
第 9 章 串行通信和可编程接口芯片 8251A	(273)
9.1 串行通信的基本概念和 EIA RS-232C 串行口	(273)
9.1.1 串行通信的基本概念	(273)
9.1.2 EIA RS-232C 串行口	(276)
9.2 可编程串行通信接口芯片 8251A	(278)
9.2.1 8251A 的内部结构和外部引脚	(279)
9.2.2 8251A 的编程	(283)
9.2.3 8251A 应用举例	(288)
第 10 章 模数(A/D)和数模(D/A)转换	(294)
10.1 概述	(294)
10.1.1 一个实时控制系统	(294)
10.1.2 采样、量化和编码	(295)
10.2 D/A 转换器	(297)
10.2.1 D/A 转换器原理	(297)
10.2.2 D/A 转换器的主要性能指标	(299)
10.2.3 D/A 转换器 AD7524、DAC0832 和 DAC1210	(300)
10.3 A/D 转换器	(306)
10.3.1 A/D 转换器原理	(306)
10.3.2 A/D 转换器 ADC0809 和 AD574A	(309)
第 11 章 DMA 控制器 8237A	(322)
11.1 8237A 的组成和工作原理	(323)
11.1.1 8237A 的内部结构	(323)
11.1.2 8237A 的引脚功能	(324)
11.1.3 8237A 的内部寄存器	(326)
11.2 8237A 的时序	(334)
11.2.1 外设和内存间的 DMA 数据传送时序	(334)

11.2.2 空闲周期、有效周期和扩展写周期	(335)
11.3 8237A 的编程和应用举例	(336)
11.3.1 PC/XT 机中的 DMA 控制逻辑	(336)
11.3.2 8237A 的一般编程方法	(338)
11.3.3 PC/XT 机上的 DMA 控制器的使用	(340)
第 12 章 总线技术	(342)
12.1 总线概述	(342)
12.1.1 总线的分类	(342)
12.1.2 总线的性能指标与总线标准	(344)
12.1.3 几种典型的计算机总线	(345)
12.2 PCI 总线	(347)
12.2.1 局部总线	(347)
12.2.2 PCI 总线简介	(349)
12.2.3 PCI 总线的应用	(350)
12.3 PCI Express 总线	(351)
12.3.1 PCI-E 总线简介	(352)
12.3.2 PCI-E 总线的发展	(353)
12.4 USB 总线	(355)
12.4.1 USB 总线简介	(355)
12.4.2 USB 的数据编码和信息传输	(360)
12.5 IEEE 1394 总线	(363)
12.5.1 1394 总线简介	(363)
12.5.2 IEEE 1394 规范的主要内容	(365)
第 13 章 32 位微型机的基本工作原理	(369)
13.1 寄存器	(369)
13.1.1 用户级寄存器	(369)
13.1.2 系统级寄存器	(372)
13.1.3 程序调试寄存器	(378)
13.2 保护模式下的内存管理	(378)
13.2.1 段内存管理技术	(379)
13.2.2 分页内存管理技术	(388)
13.3 保护模式下的中断和异常	(391)
13.3.1 中断和异常	(391)
13.3.2 保护模式下中断和异常的处理	(399)
13.4 任务切换	(403)
13.4.1 任务结构和任务切换数据结构	(403)
13.4.2 任务切换方式	(407)
13.4.3 任务调用、链接和切换过程	(409)

第 14 章 32 位机的指令系统和程序设计	(414)
14.1 80386 新增指令和程序设计	(414)
14.1.1 80386 的寻址方式	(414)
14.1.2 80386 的新增指令	(416)
14.1.3 程序设计实例	(422)
14.2 浮点数的表示方法和奔腾处理器的 SIMD 技术	(429)
14.2.1 浮点数的表示方法	(429)
14.2.2 奔腾处理器的 SIMD 技术	(433)
14.3 SIMD 指令系统	(436)
14.3.1 数据传送指令	(437)
14.3.2 算术运算指令	(444)
14.3.3 逻辑运算指令	(448)
14.3.4 移位指令	(448)
14.3.5 比较指令	(449)
14.3.6 数据转换指令	(451)
14.4 利用 SIMD 指令进行程序设计	(452)
第 15 章 微型计算机系统结构	(460)
15.1 PC/XT 机的系统板	(460)
15.1.1 CPU 子系统	(460)
15.1.2 接口部件子系统	(462)
15.1.3 存储器子系统	(463)
15.2 32 位微型机的典型结构	(465)
15.2.1 主板的组成	(465)
15.2.2 Pentium II 主板	(466)
15.2.3 集成型主板	(469)
15.3 64 位微型机	(472)
15.3.1 64 位处理器	(472)
15.3.2 64 位操作系统	(474)
15.3.3 915 系列芯片组与主板	(475)
15.4 多核处理器技术	(478)
15.4.1 双核处理器的诞生	(478)
15.4.2 Intel 智能酷睿多核处理器	(480)
15.4.3 微处理器技术发展的新时代	(484)
附录 A ASCII 码编码表	(490)
附录 B 汇编语言上机过程	(491)
参考文献	(494)

第 1 章 微型计算机的基础知识和发展概况

本章首先介绍计算机中数的表示方法,然后就计算机的基本结构、大致的工作过程和计算机软件等内容展开讨论,接着阐述微型计算机的结构和微型计算机系统,使大家对 CPU、存储器、接口等概念有一个初步的了解,最后简要讲述计算机和微型计算机的发展概况。

1.1 计算机中数的表示方法

1.1.1 进位计数制

进位计数制是指用一组固定的数字符号和特定的规则来表示数的方法。在日常生活中,数多用十进制来表示,这是大家所熟悉的。有时也用别的进制来表示数,例如,表示时间的时、分、秒之间是六十进制,而小时与天之间为二十四进制。在微型计算机中,则采用只有 0 和 1 两个数字的二进制来表示数。这是因为计算机是一种电子设备,由大量只能识别电信号的逻辑部件组成,可以用电平的高低、开关的通断、晶体管的导通和截止来表示数字 0 和 1,运算规则简单,使用方便可靠。但二进制数的数位太长,不容易书写和记忆,而人们又习惯于使用十进制数,因此在计算机领域里使用多种进位制来表示数,常用的有二进制、十进制和十六进制的表示方法。

1. 十进制数 (Decimal)

十进制数具有 10 个不同的数字符号 0~9,其基数为 10,各位的权值为 10^i ,其实际值可按权展开后相加获得。在十进制数字后面可以加后缀 D,表示该数是十进制数,但 D 通常省略不写。例如,十进制数

$$347D = 347 = 3 \times 10^2 + 4 \times 10^1 + 7 \times 10^0$$

2. 二进制数 (Binary)

它只有 0 和 1 两个数字,其基数为 2,各位的权值为 2^i 。表示二进制数时,后面必须加后缀 B。例如,二进制数

$$10110B = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 22$$

3. 十六进制数 (Hexadecimal)

它由 0~9、A、B、C、D、E、F 共 16 个数字组成,其基数为 16,各位的权值为 16^i 。表示十六进制数时,后面必须加后缀 H。数字 A~F 分别表示十进制数的 10~15。例如,十六进制数

$$3A0FH = 3 \times 16^3 + 10 \times 16^2 + 0 \times 16^1 + 15 \times 16^0 = 3 \times 4096 + 10 \times 256 + 15 = 14863$$

每个十六进制数字都可用 4 位二进制数来表示,见表 1.1,如 0AH = 1010B, 0FH = 1111B。由于十六进制数与二进制数之间转换很方便,数位长度又只有二进制数的 1/4,因此,在编写汇编语言程序和打印程序清单时,广泛使用十六进制数。在计算机中表示存储器

的地址和存放的数据时,通常也都用十六进制数。

表 1.1 十进制数、二进制数、十六进制数、BCD 码的关系

十进制数	二进制数	十六进制数	BCD 码
0	0000	0	0000
1	0001	1	0001
2	0010	2	0010
3	0011	3	0011
4	0100	4	0100
5	0101	5	0101
6	0110	6	0110
7	0111	7	0111
8	1000	8	1000
9	1001	9	1001
10	1010	A	0001 0000
11	1011	B	0001 0001
12	1100	C	0001 0010
13	1101	D	0001 0011
14	1110	E	0001 0100
15	1111	F	0001 0101

4. 八进制数(Octal)

八进制数由 0~7 共 8 个数字组成,其基数为 8,各位的权值为 8^i 。表示八进制数时,后面必须加后缀 O 或 Q。例如,八进制数

$$753Q = 7 \times 8^2 + 5 \times 8^1 + 3 \times 8^0 = 491$$

每位八进制数可以用 3 位二进制数表示,例如, $627Q = 110\ 010\ 111B$,它与二进制数之间的转换十分方便。在 20 世纪 70~80 年代使用的小型计算机系统上,普遍采用八进制数编写汇编语言程序和打印程序清单。随着微型计算机技术的普及,十六进制计数法被用户广泛使用,微机中已不再采用八进制计数法了。

5. 不同进制数之间的转换

由上面的例子可以看到,将二进制或十六进制数转换成十进制数时,只要将当前位数值乘以该位的权值,再将各部分相加即可。如果要将十进制数转换成二进制数,则必须采用除以 2 求余数的方法来得到。

例 1.1 将十进制数 25 转换成二进制数,先用 2 去除,得到的商为 12,第一个余数 $B_0 = 1$,此位为结果的最低位。再将 12 除以 2,得到第二个余数 $B_1 = 0$ ……如此不断进行下去,直到商等于 0 为止,最后一次得到的余数为结果的最高位。这样,可以得到 $25 = 11001B$ 。

2	25	
2	12	余数=1 (B ₀) 最低位
2	6	余数=0 (B ₁)
2	3	余数=0 (B ₂)
2	1	余数=1 (B ₃)
	0	余数=1 (B ₄) 最高位

根据表 1.1, 将二进制数转换成十六进制数就很容易。例如, $1000\ 1010\text{B}=8\text{AH}$ 。这里要说明的是由于二进制数的数位较长, 不便于阅读, 所以本书在后面书写二进制数时, 从最低位起, 每隔 4 个二进制数, 前面加一个空格。但当二进制数在程序中出现时, 这个空格必须除去, 否则送到计算机中进行运算时就不正确了。

例如, 把二进制数 1011010011011001B 写成 $1011\ 0100\ 1101\ 1001\text{B}$, 它等于 B4D9H 。

6. 位、字节、字和字长等数据单位表示

- 位(Bit) 在计算机中, 二进制数的每 1 位(0 或 1)是组成二进制信息的最小单位, 称为 1 个比特, 简称位, 它是数字系统和计算机中信息存储、处理和传送的最小单位。

- 字节(Byte) 8 个二进制信息组成的一个单位称为 1 个字节, $1\ \text{Byte}=8\ \text{Bit}$ 。

- 字(Word) 一个字由 16 位二进制数也即两个字节组成。一个 16 位的字 $D_{15}\sim D_0$ 可分为高字节和低字节两个部分, 其中 $D_{15}\sim D_8$ 为高字节, $D_7\sim D_0$ 为低字节。

- 字长(word Length) 计算机中还用字长这个术语来表示数据, 字长决定了计算机内部一次可处理的二进制代码的位数, 它取决于计算机内部的运算器、通用寄存器和数据总线的位数。根据计算机的字长不同, 可将计算机分为 8 位机、16 位机、32 位机和 64 位机等不同的机型。显然, 计算机的字长越长, 一次能同时传送和处理的数据就越多, 运算速度越快, 精度也越高, 但制造工艺就越复杂。

1.1.2 二进制编码

在计算机中, 各种信息都采用二进制的形式来表示, 因此各种数字、英文字母、运算符号等, 都要采用若干特定的二进制码的组合来表示, 这就是二进制编码。最常用的编码有 BCD 码和 ASCII 码两种。

1. 8421 BCD 码(Binary Coded Decimal)

虽然计算机中的数都采用二进制表示, 但二进制数不但数位长, 而且不直观, 所以常采用 BCD 码来表示。BCD 码有 0, 1, 2, ..., 9 共 10 个不同的数字符号, 也是逢 10 进 1, 但它的每一位数字都用 4 位二进制来表示, 所以称为二进制的 BCD 数, 它是一种很直观的编码。4 位二进制数可表示 $0000\sim 1111$ 共 16 种码, 取前 10 个码作为 BCD 码。由于它 4 位的权值分别是 8、4、2、1, 因此也称为 8421 BCD 码。

例 1.2 用 8421 BCD 码表示十进制数 327。

$$(327)_{10} = (0011\ 0010\ 0111)_{\text{BCD码}}$$

反之, 由 BCD 码也很容易求得它表示的十进制数。

例 1.3 $(1001\ 0101\ 1000)_{\text{BCD码}} = 958$

十进制数、二进制数、十六进制数和二十进制 BCD 码之间的关系如表 1.1 所示。

BCD 码既照顾了人们使用十进制数的习惯,又考虑了计算机的特点,机器中还有专门的调整电路对数据自动进行处理,所以实际应用中经常采用这种编码。

2. ASCII 码

除了十进制数外,各种数字、字母及字符也必须用二进制编码来表示后,计算机才能进行处理。最常用的是 ASCII 码(American Standard Code for Information Interchange),即美国标准信息交换码。它用 7 位代码(00~7FH)来表示计算机中存储的字母、数字及符号,共可表示 128 个字符。如数字 0~9,字母 A~Z、a~z 等,其中数字 0~9 的 ASCII 码为 30H~39H,字母 A~Z 的 ASCII 码为 41H~5AH。键盘键入的数、字母以及送到显示器显示的字符都必须用 ASCII 码表示,标准 ASCII 码表见附录 A。一些控制字符也列入表中,它们在计算机中实现某些控制功能,如 CR、LF 和 BEL 分别表示回车、换行和响铃,STX、ETX、ENQ 等用于串行异步通信。

1.1.3 带符号数的表示方法

上面介绍的二进制数没有提到符号问题,因此是一种无符号数。但在计算机中,数有正、负之分,怎么来表示符号呢?通常用最高位作为符号位。若一个数的长度为 8 位($D_7 \sim D_0$),则用 D_7 位作符号位, $D_7=1$,表示是负数; $D_7=0$,表示是正数。

例 1.4 $0101\ 1101\text{B}=+93$

$1101\ 1101\text{B}=-93$

连同符号位在一起作为一个数称为机器数,它表示的实际数值称为机器数的真值。上面两个式子中,等式左边的数为机器数,右边的数为真值。超过 8 位的数可以用 16 位二进制数($D_{15} \sim D_0$)来表示,这时, $D_{15}=1$,表示负数; $D_{15}=0$,表示正数。

为了运算方便,机器数通常有三种表示方法:原码、反码和补码。

1. 原码

正数的符号位用 0 表示,负数用 1 表示,其余位为数值,这种表示方法称为原码。

例 1.5 $X=+105$, $[X]_{\text{原}}=0110\ 1001\text{B}$

$X=-105$, $[X]_{\text{原}}=1110\ 1001\text{B}$

原码简单易懂,与真值的换算也很方便,但若要进行两个异号数相加或者两个同号数相减的运算,就要做减法操作。然而在一般的计算机中是没有减法运算部件的,减法运算也要用加法部件实现,所以要引进反码和补码。

2. 反码

正数的反码与原码相同,最高位为符号位,用 0 表示,其余位为数值。

例 1.6 $[+4]_{\text{反}}=0000\ 0100\text{B}$

$[+31]_{\text{反}}=0001\ 1111\text{B}$

$[+127]_{\text{反}}=0111\ 1111\text{B}$ (最大值)

负数的反码为它的正数按位取反,即连同符号位一同取反。

例 1.7 $[-4]_{\text{反}}=1111\ 1011\text{B}$

$[-31]_{\text{反}}=1110\ 0000\text{B}$

$[-127]_{\text{反}}=1000\ 0000\text{B}$ (最小值)