



教育部高职高专规划教材
Jiaoyubu Gaozhi Gaozhuan Guihua Jiaocai

微机原理与接口技术

汪吉鹏 主编
马云峰 于乃功 穆效江 副主编

高等教育出版社
HIGHER EDUCATION PRESS



教育部高职高专规划教材

(电气、自动化系列)

电路基本分析	石 生	主编
电路与电工技术	陆国和	主编
电工实验与实训	陆国和	主编
模拟电子技术基础 (第二版)	周良权	主编
模拟电子技术	胡宴如	主编
数字电子技术	杨志忠	主编
电子电路及电子器件	郭培源	主编
电子技术实践与训练	廖先芸	主编
电子技术	付植桐	主编
电机及拖动	许晓峰	主编
电机应用技术基础	牛维扬	主编
电力电子技术	浣喜明	主编
自动检测技术	宋文绪	主编
自动控制原理及其应用	姚为正	主编
交直流传动控制系统	杨 帆	主编
工厂电气控制技术	黄 坚	主编
可编程控制器应用技术	钱 平	主编
可编程控制器技术教程	张运波	主编
微机原理与接口技术	胡学林	编著
工厂供电技术	吕景泉	主编
	汪吉鹏	主编
	陈小虎	主编

ISBN 7-04-009834-2



9 787040 098341 >

定价 16.90 元

教育部高职高专规划教材

微机原理与接口技术

汪吉鹏 主 编

马云峰 于乃功 穆效江 副主编

高等教育出版社

内容提要

本书是教育部高职高专规划教材。

本书以 MCS-51 系列单片机为主线,从应用角度出发,阐述了单片机的硬件结构、指令系统、程序设计的基本方法和应用、系统扩展和接口技术。全书共分十四章,内容包括:微型计算机的基础知识;MCS-51 单片机的硬件结构;半导体存储器及其扩展技术;MCS-51 的指令系统;汇编语言程序设计方法;中断系统;定时器/计数器及其应用;微型机的输入/输出;MCS-51 与可编程接口的扩展;MCS-51 系统扩展;单片机串行通信与接口;数模、模数转换器的接口设计;单片机应用系统的设计和单片机应用系统的开发工具。本书注重将理论讲授和实践训练相结合,强调应用性和实践性,并结合大量应用实例对单片机应用系统的设计、开发、调试和故障诊断等进行了讨论。每章配有丰富的例题、习题和思考题。

本书可作为高等职业学校、高等专科学校、成人高等学校及本科院校举办的二级职业技术学院和民办高校电气、电子、通信、自动化、机电一体化等专业的教材,也可供从事单片机应用的工程技术人员参考。

图书在版编目(CIP)数据

微机原理与接口技术/汪吉鹏主编. —北京:高等教育出版社,2001
教育部高职高专规划教材
ISBN 7-04-009834-2

I. 微… II. 汪… III. ①微型计算机—理论—高等学校:技术学校—教材②微型计算机—接口—高等学校:技术学校—教材 IV. TP36

中国版本图书馆 CIP 数据核字(2001)第 07818 号

微机原理与接口技术
汪吉鹏 主编

出版发行 高等教育出版社

社 址 北京市东城区沙滩后街 55 号

电 话 010-64054588

网 址 <http://www.hep.edu.cn>

<http://www.hep.com.cn>

经 销 新华书店北京发行所

排 版 高等教育出版社照排中心

印 刷 北京印刷二厂

开 本 787×1092 1/16

印 张 19.75

字 数 480 000

邮 政 编 码 100009

传 真 010-64014048

版 次 2001 年 7 月第 1 版

印 次 2001 年 7 月第 1 次印刷

定 价 16.90 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

出版说明

教材建设工作是整个高职高专教育教学工作中的重要组成部分。改革开放以来,在各级教育行政部门、学校和有关出版社的共同努力下,各地已出版了一批高职高专教育教材。但从整体上看,具有高职高专教育特色的教材极其匮乏,不少院校尚在借用本科或中专教材,教材建设仍落后于高职高专教育的发展需要。为此,1999年教育部组织制定了《高职高专教育基础课程教学基本要求》(以下简称《基本要求》)和《高职高专教育专业人才培养目标及规格》(以下简称《培养规格》),通过推荐、招标及遴选,组织了一批学术水平高、教学经验丰富、实践能力强的教师,成立了“教育部高职高专规划教材”编写队伍,并在有关出版社的积极配合下,推出一批“教育部高职高专规划教材”。

“教育部高职高专规划教材”计划出版500种,用5年左右时间完成。出版后的教材将覆盖高职高专教育的基础课程和主干专业课程。计划先用2~3年的时间,在继承原有高职、高专和成人高等学校教材建设成果的基础上,充分汲取近几年来各类学校在探索培养技术应用型专门人才方面取得的成功经验,解决好新形势下高职高专教育教材的有无问题;然后再用2~3年的时间,在《新世纪高职高专教育人才培养模式和教学内容体系改革与建设项目计划》立项研究的基础上,通过研究、改革和建设,推出一大批教育部高职高专教育教材,从而形成优化配套的高职高专教育教材体系。

“教育部高职高专规划教材”是按照《基本要求》和《培养规格》的要求,充分汲取高职、高专和成人高等学校在探索培养技术应用型专门人才方面取得的成功经验和教学成果编写而成的,适用于高等职业学校、高等专科学校、成人高校及本科院校举办的二级职业技术学院和民办高校使用。

教育部高等教育司

2000年4月3日

前言

本书是教育部高职高专规划教材。

本书以 MCS-51 系列单片机为主线,从应用角度出发,阐述了单片机的硬件结构、指令系统、程序设计的基本方法和应用、系统扩展和接口技术。全书共分十四章,内容包括:微型计算机的基础知识;MCS-51 单片机的硬件结构;半导体存储器及其扩展技术;MCS-51 的指令系统;汇编语言程序设计方法;中断系统;定时器/计数器及其应用;微型机的输入/输出;MCS-51 与可编程接口的扩展;MCS-51 系统扩展;单片机串行通信与接口;数模、模数转换器的接口设计;单片机应用系统的设计和单片机应用系统的开发工具。本书注重将理论讲授和实践训练相结合,强调应用性和实践性,并结合大量应用实例对单片机应用系统的设计、开发、调试和故障诊断等进行了讨论。每章配有丰富的例题、习题和思考题。

本书的编写特色主要表现在以下几个方面:

1. 以 MCS-51 单片机为核心讲述微机原理及其接口技术,阐述单片机应用系统的开发和应用软件的编制。把单片机内部结构和外围接口结合起来,建立一个较完整的单片机应用系统。
2. 在接口技术部分的内容中,适当增加串行接口芯片的硬件连接及编程设计方法,使其具有一定的先进性和实用性。
3. 简要介绍了 8051 系列用 C51 语言开发单片机应用软件,使学生初步掌握用高级语言设计单片机应用系统程序的方法。
4. 全书内容根据电类高职高专教学计划和教学要求而编写,内容组织新颖,应用举例实用,概念条理清晰,便于学生的学习和掌握。

本书承东南大学胡仁杰副教授主审,他仔细审阅了书稿并提出了许多宝贵意见,在此表示深切的感谢。

参加本书编写工作的有汪吉鹏(前言、第 2 章、第 4 章、第 5 章、第 6 章、第 7 章、附录 A、附录 B、附录 C),于乃功(第 1 章、第 12 章、第 13 章),穆效江(第 3 章、第 10 章、第 11 章),马云峰(第 8 章、第 9 章、第 14 章)。汪吉鹏任主编,负责全书的组织编写、统稿、定稿和审阅。马云峰、于乃功、穆效江任副主编(按姓氏笔画排列),负责本书的审阅工作。

由于编者水平有限且成书仓促,难免有疏漏、欠妥和错误之处,恳求读者批评指正。

编者

2000 年 8 月

目 录

第1章 微型计算机基础知识	1	2.4.4 低功耗操作方式	44
1.1 微型计算机数制及其转换	1	思考题与习题	45
1.1.1 微型计算机常用数制的特点	1	第3章 半导体存储器及其扩展设计	46
1.1.2 微型计算机数制间的转换	3	3.1 半导体存储器	46
1.2 微型计算机中数的表示方法	6	3.1.1 半导体存储器概述	46
1.2.1 机器数与真值	6	3.1.2 常用半导体存储器	48
1.2.2 原码、反码、补码	7	3.2 单片机外部存储器的扩展	52
1.2.3 微型计算机的二进制编码	11	3.2.1 扩展外部存储器的一般方法	52
1.3 微型计算机系统组成原理	14	3.2.2 扩展程序存储器	54
1.3.1 计算机的基本结构	14	3.2.3 扩展数据存储器	55
1.3.2 微型计算机的基本组成	14	3.2.4 扩展串行存储器	57
1.3.3 微型计算机的基本工作原理	16	3.2.5 存储器芯片的选择	60
1.3.4 微型计算机系统组成	17	3.3 存储器的若干实用技术	61
1.4 单片微型计算机概述	19	思考题与习题	62
1.4.1 单片机的历史及发展概况	19	第4章 MCS-51 的指令系统	63
1.4.2 单片机的发展趋势	20	4.1 概述	63
1.4.3 单片机的应用	21	4.1.1 指令和程序设计语言	63
1.4.4 8位单片机的主要生产厂商和 机型	22	4.1.2 MCS-51 指令系统及符号说明	65
1.4.5 MCS-51 系列单片机	22	4.2 MCS-51 指令系统的寻址方式	66
思考题与习题	24	4.2.1 寄存器寻址	66
第2章 MCS-51 单片机的硬件结构	26	4.2.2 直接寻址	66
2.1 MCS-51 单片机内部结构	26	4.2.3 寄存器间接寻址	66
2.1.1 MCS-51 单片机的基本组成	26	4.2.4 立即寻址	67
2.1.2 MCS-51 单片机内部结构	27	4.2.5 基址寄存器加变址寄存器间接寻址	67
2.2 MCS-51 单片机引脚功能	38	4.2.6 相对寻址	68
2.3 MCS-51 单片机时序	41	4.2.7 位寻址	69
2.3.1 机器周期和指令周期	41	4.2.8 寻址空间	69
2.3.2 MCS-51 单片机指令的取指、 执行时序	41	4.3 MCS-51 指令系统	69
2.4 MCS-51 单片机复位及复位电路	43	4.3.1 数据传送类指令	70
2.4.1 复位操作	43	4.3.2 算术操作类指令	73
2.4.2 复位信号和复位电路	43	4.3.3 逻辑运算指令	77
2.4.3 程序执行方式	43	4.3.4 控制转移类指令	79
		4.3.5 位操作指令	83

思考题与习题	84	7.4 定时/计数器的4种工作模式及应用	126
第5章 MCS-51 汇编语言程序设计	87	7.4.1 工作模式0及应用	127
5.1 汇编语言程序设计步骤与技巧	87	7.4.2 工作模式1及应用	128
5.1.1 汇编语言程序设计步骤	87	7.4.3 工作模式2及应用	129
5.1.2 汇编语言编程技巧	88	7.4.4 工作模式3及应用	131
5.2 伪指令	88	7.5 应用举例	132
5.3 汇编语言程序设计	90	思考题与习题	133
5.3.1 顺序程序设计	91	第8章 微型机的I/O接口技术	134
5.3.2 分支程序设计	92	8.1 有关接口的基本概念	134
5.3.3 循环程序设计	94	8.1.1 输入/输出及硬件接口	134
5.3.4 子程序设计	100	8.1.2 接口信号的分类	135
5.3.5 查表程序设计	102	8.1.3 端口的概念及操作	136
5.3.6 散转程序设计	106	8.1.4 接口电路的功能概述	136
5.3.7 模拟组合逻辑的程序设计	107	8.1.5 主机对I/O的寻址方式	137
思考题与习题	109	8.2 接口数据的传送方式	138
第6章 中断系统	111	8.2.1 同步传送方式	138
6.1 中断概述	111	8.2.2 异步传送方式	138
6.1.1 中断概念	111	8.2.3 中断传送方式	139
6.1.2 中断类型	112	8.2.4 DMA传送方式	139
6.1.3 中断优先权及CPU响应中断的原则	112	8.3 I/O接口的简单扩展	140
6.2 中断处理过程	113	8.3.1 I/O接口的类型	140
6.2.1 中断请求	113	8.3.2 单片机的I/O口特性	140
6.2.2 中断响应	114	8.3.3 简单I/O口扩展举例	142
6.2.3 中断处理	114	思考题与习题	146
6.2.4 中断返回	114	第9章 MCS-51与可编程接口的扩展	147
6.3 MCS-51单片机中断系统	114	9.1 并行I/O接口芯片8255A	147
6.3.1 MCS-51单片机的中断结构	114	9.1.1 8255A的主要特性	147
6.3.2 中断源	115	9.1.2 8255A的内部结构	147
6.3.3 中断的控制	115	9.1.3 8255A引脚及其功能(外部特性)	148
6.3.4 中断处理过程	118	9.1.4 8255A的控制字	150
6.4 中断程序举例	120	9.1.5 8255A与单片机接口实例	154
6.4.1 主程序	120	9.2 RAM/IO接口芯片8155	155
6.4.2 中断服务程序	120	9.2.1 8155引脚及其功能	155
思考题与习题	123	9.2.2 8155内部寄存器及其编址	156
第7章 定时/计数器及其应用	124	9.2.3 8155内部定时/计数器	158
7.1 定时/计数器的基本功能	124	9.2.4 8155与单片机的接口实例	159
7.2 8051片内定时/计数器概述	124	9.3 键盘/显示器接口芯片8279	161
7.3 定时器的控制	125	9.3.1 8279的内部结构及引脚	161
7.3.1 工作模式寄存器TMOD	125	9.3.2 8279的操作命令字	163
7.3.2 控制寄存器TCON	126	9.3.3 8279的状态字	167

9.3.4 8279 与键盘/显示器的接口实例	168	11.6.3 SMOD 位对波特率的影响	238
9.4 串行口扩展芯片 8251A	170	思考题与习题	238
9.4.1 8251A 的基本特点	170	第12章 数模、模数转换器的接口设计	240
9.4.2 8251A 的内部结构和引脚功能	170	12.1 DAC 及其与 MCS-51 单片机的接口	240
9.4.3 8251A 的控制字	173	12.1.1 DAC 的性能指标和选择方法	240
9.4.4 MCS-51 和 8251A 的接口实例	176	12.1.2 MCS-51 与 DAC 的接口	241
思考题与习题	178	12.2 ADC 及其与 MCS-51 的接口	249
第10章 MCS-51 系统扩展	179	12.2.1 ADC 的性能指标和选择方法	249
10.1 MCS-51 与键盘接口	179	12.2.2 MCS-51 与 ADC 的接口	251
10.1.1 键盘及其接口	179	思考题与习题	259
10.1.2 单片机应用系统中的典型显示/键盘接口	185	第13章 单片机应用系统的设计	260
10.2 MCS-51 与显示器的接口	188	13.1 单片机应用系统设计概述	260
10.2.1 LED 显示器及其接口	188	13.1.1 系统设计步骤	260
10.2.2 液晶显示器(LCD)及其应用	196	13.1.2 总体设计要点	261
10.3 MCS-51 与 TPμp-40A/16A 微型打印机的接口	205	13.1.3 硬件设计	262
思考题与习题	207	13.1.4 软件设计	262
第11章 单片机串行通信与接口	208	13.1.5 数据采集系统	263
11.1 串行通信基础	208	13.1.6 系统的调试、运行和维护	265
11.1.1 串行通信的分类	208	13.2 应用实例——溴素生产物料调控自动监视系统	265
11.1.2 串行通信的制式	210	13.2.1 系统的技术指标及功能	266
11.2 串行通信的接口标准	211	13.2.2 方案论证	266
11.2.1 RS-232C 接口	211	13.2.3 硬件电路设计	267
11.2.2 RS-422A 接口	213	13.2.4 软件设计	271
11.2.3 RS-485 接口	214	13.3 系统可靠性设计和故障诊断技术	277
11.2.4 各种串行接口性能比较	215	13.3.1 系统可行性设计	277
11.3 MCS-51 的串行接口	216	13.3.2 系统抗干扰技术	278
11.3.1 串行口控制寄存器 SCON 和 PCON	216	13.3.3 系统故障诊断	280
11.3.2 串行口的工作方式	217	思考题与习题	281
11.3.3 串行口的通信波特率	219	第14章 单片机应用系统的开发工具	283
11.4 双机串行通信技术	220	14.1 开发系统简介	283
11.4.1 双机通信接口设计	220	14.1.1 开发系统的功能	284
11.4.2 双机通信软件设计	221	14.1.2 开发系统的分类	284
11.5 多机串行通信技术	227	14.1.3 有关开发系统性能的几个基本概念	286
11.6 串行通信中的波特率设置技术	236	14.1.4 开发系统的基本组成	286
11.6.1 MCS-51 单片机串行通信波特率的确定	236		
11.6.2 波特率相对误差范围的确定方法	236		

14.2 用户样机开发调试过程	287	14.3.4 C51 程序结构	298
14.2.1 用户样机硬件调试	287	思考题与习题	299
14.2.2 用户样机软件的设计、调试	289	附录 A ASCII 码字符表	300
14.3 C51 开发工具	291	附录 B MCS-51 系列单片机指令集	301
14.3.1 8051 的编程语言	291	附录 C 指令助记符与机器码速查表	306
14.3.2 C51 编译器	292	参考文献	307
14.3.3 Keil/Franklin C51 编译实例	292		

第1章 微型计算机基础知识

1.1 微型计算机数制及其转换

人们在日常生活和工作中习惯用十进制数,而迄今为止,所有数字计算机都是以二进制形式进行算术运算和逻辑操作的,微型计算机也不例外。因此,通常人们还是用十进制数与计算机打交道,然后由计算机自动将用户从键盘上输入的十进制数和符号命令转换成二进制形式进行识别、运算和处理,最后再把处理结果还原成十进制数字和符号在 CRT 上显示出来。为了使使用计算机的人员弄清机器的这一工作机理,首先讨论一下常用进位制数的特点及它们之间的转换方法。

1.1.1 微型计算机常用数制的特点

数制是指数的制式,是人们利用符号计数的一种科学方法。微型计算机中常用的数制有十进制、二进制、八进制和十六进制,存储在微型计算机中的信息是二进制形式的信息。人们常常用书写起来比较容易的八进制数或十六进制数来描述机器内的二进制数。

1. 十进制数

十进制数的特点有两个:一是它有 0~9 十个不同的数码,即 0、1、2、3、4、5、6、7、8 和 9 十个数字符号,这是构成所有十进制数的基本符号,这十个数字符号又称为数码;二是它是逢十进位的。十进制数在计数过程中,当它的某位计满 10 时就要向它邻近高位进一。因此,每个数码在数中最多可有两个值的概念,任何一个十进制数不仅和构成它的每个数码本身的值有关,而且还和这些数码在数中的位置有关。例如:十进制数 36 中数码 3,其本身的值为 3,但它实际代表的值为 30。

任何一个十进制数都可以展开成幂级数形式。例如:

$$(138.38)_{10} = 1 \times 10^2 + 3 \times 10^1 + 8 \times 10^0 + 3 \times 10^{-1} + 8 \times 10^{-2}$$

式中,基数为 10 的幂 10^2 、 10^1 、 10^0 、 10^{-1} 和 10^{-2} 在数学上称为权,整数部分中每位的幂是该位位数减 1,小数部分中每位的幂是该位小数的位数的负值。

十进制数 N 的一般形式为:

$$N = \pm \sum_{i=-m}^{n-1} a_i \times 10^i$$

式中: i 表示数中任一位,是一个变量; a_i 表示第 i 位的数码; n 为该数整数部分的位数; m 为小数部分的位数。

2. 二进制数

二进制数有两个主要特点:一是它有 0 和 1 两个数码,任何二进制数都是由这两个数码组成

的;二是二进制数的基数为 2,它遵循逢二进一的进位计数规则。

同十进制数一样,任意一个二进制数 N 同样也可以展开成幂级数形式,2 为计数制的基数,其一般形式为:

$$N = \pm \sum_{i=-m}^{n-1} a_i \times 2^i$$

式中: a_i 为第 i 位数码,可取 0 或 1; n 为该二进制数整数部分的位数; m 为小数部分位数。

$$\begin{aligned}\text{例如:}(1011.011)_2 &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} \\ &= 1 \times 2^3 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-2} + 1 \times 2^{-3} \\ &= (11.375)_{10}\end{aligned}$$

3. 八进制数

八进制数的特点有两个;一是它有八个不同的数码,即 0、1、2、3、4、5、6 和 7 八个数字符号,这是构成所有八进制数的基本符号;二是它是逢八进位的。

同十进制数一样,任意一个八进制数 N 同样也可以展开成幂级数形式,8 为计数制的基数,其一般形式为:

$$N = \pm \sum_{i=-m}^{n-1} a_i \times 8^i$$

式中: a_i 为第 i 位数码,取值为 0~8; n 为该八进制数整数部分的位数; m 为小数部分位数。

$$\begin{aligned}\text{例如:}(207.2)_8 &= 2 \times 8^2 + 0 \times 8^1 + 7 \times 8^0 + 2 \times 8^{-1} = 2 \times 8^2 + 7 \times 8^0 + 2 \times 8^{-1} \\ &= (135.25)_{10}\end{aligned}$$

4. 十六进制数

十六进制是人们学习和研究计算机中二进制数的一种工具,它是随着计算机的发展而广泛应用的。十六进制数的主要特点也有两个:一是它有 0、1、2、...、9、A、B、C、D、E、F 等十六个数码,这是构成十六进制数的基本符号;二是十六进制数的基数为 16,进位计数为逢十六进一。

十六进制数也可展开成幂级数形式,其一般形式为:

$$N = \pm \sum_{i=-m}^{n-1} a_i \times 16^i$$

式中: a_i 为第 i 位数码,取值为 0~F; n 为该数整数部分位数; m 为小数部分位数。

$$\begin{aligned}\text{例如:}(70F.B1)_{16} &= 7 \times 16^2 + F \times 16^0 + B \times 16^{-1} + 1 \times 16^{-2} \\ &= (1807.6914)_{10}\end{aligned}$$

在微型计算机内部,数的表示形式是二进制的,这是因为二进制数只有 0 和 1 两个数码,人们采用晶体管的导通和截止、脉冲的高电平和低电平就能分别表示它们。此外,二进制数运算简单,便于用电子线路实现。

人们采用十六进制可以大大减轻阅读和书写二进制数时的负担。例如:

$$(11011011)_2 = (DB)_{16}$$

$$(1001001111110010)_2 = (93F2)_{16}$$

显然,采用十六进制数描述一个二进制数特别简短,尤其在被描述二进制数位数较长时更令计算机工作者感到方便。

在阅读和书写不同数制的数时,如果不在每个数上外加一些辨认标记,就会混淆而无法分清。通常,标记方法有二种:一种是把数加上括号,并在括号右下角标注数制代号,如 $(101)_{16}$ 、 $(101)_2$ 、 $(101)_8$ 、 $(101)_{10}$ 分别表示十六进制、二进制、八进制和十进制;另一种是用英文字母标记,加在被标记数的后面,用B、D和H大写字母分别表示二进制、十进制和十六进制数,如89H为16进制数、101B为二进制数等。其中,十进制数中的D标记也可以省略。

1.1.2 微型计算机数制间的转换

人们习惯使用十进制数,而微型计算机是采用二进制数操作的,因此要求微型机计算机能自动对不同数制的数进行转换。为弄清楚微型计算机对不同数制数间的转换方法,我们先来看看数学上是如何进行上述4种数制数间的转换的。

1. 二进制和十进制数间的转换

(1) 二进制数转换成十进制数:把二进制数按权展开后求和或从小数点开始向左或向右每4位作为一组按十六进制的权展开再求和。例如:

$$\begin{aligned}(11011.01)_2 &= 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-2} \\ &= (27.25)_{10} \\ \text{或者:} (11011.01)_2 &= (1B.4)_{16} = 1 \times 16^1 + 11 \times 16^0 + 4 \times 16^{-1} \\ &= (27.25)_{10}\end{aligned}$$

(2) 十进制数转换成二进制数:十进制整数和小数转换成二进制的整数和小数的方法是不相同的,下面分别进行介绍。

① 十进制转换成二进制整数的方法有很多种,但一般使用“除2取余法”。“除2取余法”法则是:用2连续去除要转换的十进制数,直到商小于2为止,然后把各次余数按最后得到的为最高位和最先得到的为最低位,依次排列起来所得到的数便是所求的二进制数。请看下面举例。

[例 1-1] 把 $(52)_{10}$ 转换成二进制数

解:

	十进制数	余数	
2	52	0	最低位
2	26	0	
2	13	1	
2	6	0	
2	3	1	
2	1	1	最高位

把所得余数按箭头方向从高到低排列起来便可得到: $(52)_{10} = (110100)_2$

② 十进制小数转换成二进制小数通常采用“乘2取整法”。

“乘2取整法”法则是:用2连续去乘要转换的十进制小数,直到所得积的小数部分为0或满足所需精度为止,然后把各次整数按最先得到的为最高位和最后得到的为最低位,“顺序”依次排列起来所对应的数便是所求的二进制小数。现结合实例加以介绍。

〔例 1-2〕 试把十进制小数 0.1875 转换为二进制小数。

解:把 0.1875 不断地乘 2,取每次所得乘积的整数部分,直到乘积的小数部分满足所需精度,相应竖式是:

0.1875		
× 2		
0.3750	·····取得整数 0	最高位
× 2		
0.7500	·····取得整数 0	
× 2		
1.5000	·····取得整数 1	
0.5000		
× 2		
1.0000	·····取得整数 1	最低位

↓

把所得整数按箭头方向从高到低排列后得到: $(0.1875)_{10} = (0.0011)_2$

对同时有整数和小数两部分的十进制数,转换成二进制数要分别转换,转换后再合并起来。

例如,把〔例 1-1〕和〔例 1-2〕合并起来便可得到: $(52.1875)_{10} = (110100.0011)_2$

应当指出,任何十进制整数都可以精确转换成一个二进制整数,但任何十进制小数却不一定能精确转换成一个二进制小数,只要转换过程中乘积的小数部分满足所需精度即可。

2. 八进制数和十进制数间的转换

(1) 八进制数转换成十进制数:方法和二进制数转换成十进制数的方法类似,即可把八进制数按权展开后求和。例如:

$$\begin{aligned}
 (148.65)_8 &= 1 \times 8^2 + 4 \times 8^1 + 8 \times 8^0 + 6 \times 8^{-1} + 5 \times 8^{-2} \\
 &= (104.828125)_{10}
 \end{aligned}$$

(2) 十进制数转换成八进制数

① 十进制整数转换成八进制整数和十进制整数转换成二进制整数类似,十进制整数转换成八进制整数可以采用“除 8 取余法”。

“除 8 取余法”法则是:用 8 连续去除要转换的十进制整数,直到商数小于 8 为止,然后把各次余数按逆序排列起来所得的数,便是所求的八进制数。

〔例 1-3〕 求 $(199)_{10}$ 所对应的八进制数。

解:

	十进制整数	余数	
8	199	·····余 7	最低位
8	24	·····余 0	↑
	3	·····余 3	最高位

把所得余数按箭头方向从高到低排列起来便可得到: $(199)_{10} = (307)_8$

② 十进制小数转换成八进制小数方法类似于十进制小数转换成二进制小数,常采用“乘 8 取整法”。

“乘 8 取整法”法则是:把要转换的十进制小数连续乘以 8,直到所得到乘积的小数部分为 0 或达到所需精度为止,然后把各次整数按顺序排列起来所得的数,便是所求的八进制小数。

[例 1-4] 求 $(0.1875)_{10}$ 的八进制数。

解:把 $(0.1875)_{10}$ 连续乘以 8,直到所得乘积的小数部分为 0,相应竖式为:

$$\begin{array}{rcl}
 & 0.1875 & \\
 \times & 8 & \\
 \hline
 & 1.5000 & \cdots\cdots\cdots\text{取整数 } 1 \\
 & 0.5000 & \\
 \times & 8 & \\
 \hline
 & 4.0000 & \cdots\cdots\cdots\text{取整数 } 4
 \end{array}$$

按箭头所指顺序排列起来便可得到: $(0.1875)_{10} = (0.14)_8$

3. 二进制和八进制数的转换

由于八进制数的一位数相当于 3 位二进制数,从八进制数转换成二进制数,只需以小数点为界,向左和向右,每位八进制数用 3 位二进制数取代,即可分别转换成二进制的整数和二进制的小数。无论是向左还是向右,最后不足 3 位二进制数时用零补足 3 位。

[例 1-5] 把 $(70.521)_8$ 转换成二进制数。

解: $(70.521)_8 = (111\ 000.101\ 010\ 001)_2$

根据同样的道理,把二进制数转换成相应的八进制数是上面转换方法的逆过程。

[例 1-6] 把 $(1011011.00101011)_2$ 转换成八进制数。

解: $(001\ 011\ 011.001\ 010\ 110) = (133.125)_8$

4. 十六进制和十进制数间的转换

(1) 十六进制数转换成十进制数:方法和二进制数转换成十进制数的方法类似,即可把十六进制数按权展开后相加。例如:

$$(3FEB)_{16} = 3 \times 16^3 + 15 \times 16^2 + 14 \times 16^1 + 11 \times 16^0 = (16363)_{10}$$

(2) 十进制数转换成十六进制数

① 十进制整数转换成十六进制整数和十进制整数转换成二进制整数类似,十进制整数转换成十六进制整数可以采用“除 16 取余法”。

“除 16 取余法”法则是:用 16 连续去除要转换的十进制整数,直到商数小于 16 为止,然后把各次余数按逆序排列起来所得的数,便是所求的十六进制数。

[例 1-7] 求 $(3901)_{10}$ 所对应的十六进制数。

解:把 $(3901)_{10}$ 连续除以 16,直到商数为 15,相应竖式为:

$$\begin{array}{rcl}
 16 \overline{) 3901} & \cdots\cdots\cdots\text{余 } 13 & \text{写作 D} & \text{最低位} \\
 16 \overline{) 243} & \cdots\cdots\cdots\text{余 } 3 & \text{写作 3} & \uparrow \\
 15 & \cdots\cdots\cdots\text{余 } 15 & \text{写作 F} & \text{最高位}
 \end{array}$$

把所得余数按箭头方向从高到低排列起来便可得到: $(3901)_{10} = (F3D)_{16}$

② 十进制小数转换成十六进制小数方法类似于十进制小数转换成二进制小数,常采用“乘

16 取整法”。

“乘 16 取整法”法则是:把欲转换的十进制小数连续乘以 16,直到所得到乘积的小数部分为 0 或达到所需精度为止,然后把各次整数按顺序排列起来所得的数,便是所求的十六进制小数。

【例 1-8】 求 $(0.76171875)_{10}$ 的十六进制数。

解:把 $(0.76171875)_{10}$ 连续乘以 16,直到所得乘积的小数部分为 0,相应竖式为:

$$\begin{array}{r} 0.76171875 \\ \times \quad 16 \\ \hline 12.18750000 \quad \cdots\cdots\cdots \text{取整数 12 写作 C} \\ 0.18750000 \\ \times \quad 16 \\ \hline 3.00000000 \quad \cdots\cdots\cdots \text{取整数 3 写作 3} \end{array}$$

按箭头所指顺序排列起来便可得到: $(0.76171875)_{10} = (0.C3)_{16}$

5. 二进制和十六进制数的转换

二进制和十六进制数间的转换十分方便,这就是为什么人们要采用十六进制形式来对二进制数加以表达的内在原因。

(1) 二进制数转换成十六进制数: 可采用“四位合一位法”。

“四位合一位法”法则是:从二进制数的小数点开始,或左或右每 4 位一组,不足 4 位以 0 补足之,然后分别把每组用十六进制数码表示,并按序排列。

【例 1-9】 把 $(10101011110100.00111000)_2$ 转换为十六进制数。

解: 0010 1010 1111 0100 .0011 1000

2 A F 4 3 8

$(10101011110100.00111000)_2 = (2AF4.38)_{16}$

(2) 十六进制数转换成二进制数: 这种转换方法是把十六进制数的每位分别用 4 位二进制数码表示,然后把它们排列起来。

【例 1-10】 把十六进制数 2AC.6B5 转换为一个二进制数。

解: 2 A C . 6 B 5

0010 1010 1100 0110 1011 0101

$(2AC.7B5)_{16} = (1010101100.011010110101)_2$

1.2 微型计算机中数的表示方法

机器数是微型机中数的基本形式。为了运算方便起见,机器数通常有原码、反码和补码三种形式。目前微型机系统中多采用补码形式,由于补码是在原码及反码的基础上演变过来的,故先介绍原码和反码,最后介绍补码。

1.2.1 机器数与真值

数学中的正负用符号“+”和“-”表示。计算机中是如何表示数的正负呢?在计算机中数据

存放在存储单元内,而每个存储单元由若干二进制位组成,其中每一位或是0或是1。在计算机中规定用最高位作为符号位,“0”表示“+”;“1”表示“-”。于是数的符号在计算机中被数码化了,即从表示形式上看符号位与数值位毫无区别。

设有两个数 N_1 及 N_2 :

$$N_1 = +1011011B; N_2 = -1011011B$$

它们在计算机中分别表示为:

$$N_1 = 01011011B; N_2 = 11011011B$$

为了区分这两种形式的数,我们把机器中以编码形式表示的数称为机器数(上例中 $N_1 = 01011011B$ 及 $N_2 = 11011011B$),而把原来一般书写形式表示的数称为真值(上例中 $N_1 = +1011011B$ 及 $N_2 = -1011011B$)。

若一个数的所有数位均为数值位,则该数为无符号数;若一个数的最高位为符号位而其它数位为数值位,则该数为有符号数。由此可见,对于同一存储单元,它存放的无符号数和有符号数所能表示的数值范围是不同的(如一个存储单元为8位,当它存放无符号数时,因有效的数值位为8位,故该数的范围为 $0 \sim 255$);当它存放有符号数时,因有效的数值位为7位,故该数的范围为 $(-127 \sim +127)$ 。

1.2.2 原码、反码、补码

1. 原码

微型计算机数的原码形式就是机器数形式,两者完全相同。它们的最高位为符号位,其余为数值位,符号位为0表示该数是正数;符号位为1表示它是负数。在微型机中,一个数的原码可以先把该数用方括号括起来,并在方括号右下角加“原”字来标记。

【例1-11】 设 $X = +1010B$, $Y = -1010B$,请分别写出它们在八位微型机中的原码形式。

解: $X = +1010B$ $Y = -1010B$

$$[X]_{\text{原}} = 00001010B \quad [Y]_{\text{原}} = 10001010B$$

用原码表示时,0这个数非常特别,它有+0和-0之分。例如,0在八位微型计算机中的两种原码形式为:

$$[+0]_{\text{原}} = 00000000B$$

$$[-0]_{\text{原}} = 10000000B$$

2. 反码

在微型计算机中,二进制数的反码求法很简单,有正数的反码和负数的反码之分。正数的反码和原码相同;负数反码的符号位和负数原码的符号位相同,数值位是它原码的数值位的按位取反。反码的标记方法和原码的类似,只要在括弧数方括号的右下角添加一个“反”字即可。

【例1-12】 设 $X = +1101101B$, $Y = -0110110B$,请写出 X 和 Y 的原码和反码形式。

解: $X = +1101101B$ $Y = -0110110B$

$$[X]_{\text{原}} = 01101101B \quad [Y]_{\text{原}} = 10110110B$$

$$[X]_{\text{反}} = 01101101B \quad [Y]_{\text{反}} = 11001001B$$

3. 补码

在日常生活中,补码的概念是经常会遇到的。例如,如果现在是北京时间下午3点钟,而您